

## Louvain School of Management

# Agile Project Management: motivations, implementation and evaluation.

Research Master's Thesis submitted by  
**Sabrina Deley**

With the view of getting the degree in  
Master 120 crédits en ingénieur de gestion, à finalité

Supervisors  
Supervisor's Manuel Kolp and Soreangsey Kiv

Academic Year 2017-2018



---

## Acknowledgements

---

I would first like to thank my thesis supervisor: Prof. Manuel Kolp. His door was always open when I ran into trouble or had any question related to my research. He let me write this paper in my own terms, and gave me the right insights to go forward.

In addition, I would like to express my gratitude to Mr. Samedi Heng and Ms. Soreangsey Kiv. Their contacts and encouragements were very helpful. Their contribution to the development of the scope of my thesis was very appreciated.

My sincere thank also goes to all the participants of the study: Mr. Cédric Folchi, Mr. Christophe Tilquin, Mr. Michel Herquet, Mr. Pierre-Emmanuel Dautreppe, Mr. Norman Deschauer, Mr. Stacey Buys, Mr. Mathieu Gérard, Mr. Louis Destrée, Mr. Franz Hegemann, Mr. Frederick Wellekens, Mr. Rudi Serron, Ms. Sze-Wai Leung and all the anonymous. They provided resourceful and valuable inputs to the study.

I finally want to express my deepest gratitude to my family and friends whom were always there for me in hard times and made this thesis happen. Especially Marianne Vanwymersch, Antoine Warnier, Simon Boland, Camille Druart, Ophélie Laforêt, Dieter Roosen and Edith De Cocqueaux for their stimulating discussions and their contribution to proofread this paper.



---

## Contents

---

Acknowledgements .....	1
Contents.....	i
Introduction .....	1
Chapter 1 – Software Project Management: an overview .....	5
1. IT project management.....	5
1.1. Project.....	5
1.2. Project Management.....	7
1.3. Conventional Software Project Management.....	11
1.4. Agile Software Project Management (APM) .....	13
2. Agile Software Development .....	16
2.1. Agile Manifesto.....	16
2.2. Agile Methodologies .....	17
2.3. Advantages of Agile project management .....	21
2.4. Limitations of Agile Project Management .....	22
2.5. Agile scaled methods .....	23
2.6. Selecting and Tailoring Agile methods.....	23
3. Management and Agile Software Development .....	25
3.1. VUCA world .....	25
3.2. Strategy, decision-making and operational .....	26
3.3. Change Management.....	28
3.4. Evaluation.....	29
Chapter 2 - Research Motivation .....	31
1. Purpose of the Research .....	31
2. Research questions .....	31
Chapter 3 - Research Methodology.....	33
1. Research Design.....	33
2. Selection of interviewees .....	33
3. Collection of qualitative data .....	37
4. Treatment and Analysis of Data.....	40
Chapter 4 - Results .....	43

1. Stand Point Before the Implementation .....	48
2. Goals.....	50
A. Strategic goals .....	50
B. Operational goals .....	51
C. Organisational goals .....	52
D. Strategic and operational goals.....	52
3. Change management .....	52
4. Assessment .....	55
5. Agile mind-set.....	56
Chapter 5 – Discussions .....	60
1. Research Question 1: Role of the goals .....	60
2. Research Question 2(1): Consideration of goals, way of adopting Agile.....	63
3. Research question 2(2): Consideration of goals, escort the change.....	64
4. Research question 2(3): Consideration, the assessment of the goals .....	65
Conclusion.....	68
Bibliography.....	73
Appendixes.....	79
Appendix 1 : Guideline of the qualitative survey .....	79
Appendix 2 : Interview Anonymous 1 .....	82
Appendix 3 : Interview Anonymous 2 .....	100
Appendix 4 : Interview Anonymous 3 .....	107
Appendix 5 : Interview Anonymous 4 .....	120
Appendix 6 : Interview Anonymous 5 .....	144
Appendix 8 : Interview Franz Hegemann .....	162
Appendix 9 : Interview Frederick Wellekens .....	171
Appendix 10 : Interview Louis Destrée .....	185
Appendix 11 : Interview Mathieu Gérard .....	199
Appendix 12 : Interview Michel Herquet .....	210
Appendix 13 : Interview Rudi Serron .....	220
Appendix 14 : Interview Stacey Buys.....	238
Appendix 15 : Interview Pyxis co-founders.....	255
Appendix 16 : Interview Sze-Wai Leung.....	273
Appendix 17 : Are you too busy to improve ?.....	287

---

## Introduction

---

Nowadays, using software has become a universal habit and an essential tool for everyone: from the multinational company willing to optimize its activities, to individuals using their smartphones. It is not easy develop a software from scratch. The requirements, scope, budget and deadlines are only few perspectives the project manager has to deal with. For an effective leading throughout the realization of a project, different project methods or frameworks are available. In theory, it seems easy to conduct a project but, in reality, how to choose the right framework can already be a challenging step to analyse.

The two best known frameworks are Waterfall and the Agile model. While Waterfall has been a known concept for centuries (even in situations such as for building a house, for instance), Agile appeared recently (in the late 20th century). Since then, there has been a switch of models from the first to the other. Even though there have been years of practice previous to this study, the maturity of the Agile implementation is still not satisfying. According to KPMG Advisory, 63% of Dutch and Belgian companies admits that the maturity of Agile is still lower than the maturity in their traditional methods (2017).

The following question is then asked: how does the management deal with Agile Project Management (APM) implementation? This implementation leads to a transformation of the company. It has the intent to change people's mind, which is one of the most challenging task in change management. This implies a lot of resistance. Motivations and goals are very important to determine before the change process begins. The decision-making for this initiative is important for people adhesion but also for long-term perspective. It affects culture, communication, opportunities and finances (Serie, 2018). The motivations and goals may be strategic, operational or organisational.

The goals mentioned in the literature are the typical advantage of Agile. The first one is the reduction of Time-to-Market. There is a need for results and delivery before the others. The adaptation to change possibility is also considered as a big goal that Waterfall lacked in. The third major goal is the transparency.

Different Agile methods exist. The best known are Scrum, Xp and Kanban. While Scrum gives a communication and prioritisation structure, Xp is more directed in the development quality. Kanban is a board that helps to visualize the work flow. Projects may be tremendous, and therefore require the use of programs. A program is a project composed by

smaller inter-connected projects. To manage programs, there also exists Agile methods, called scaled methods. The most commonly used scaled methods are SAFe, LeSS and Spotify.

The implementation of these methods needs guidance. The change management will then be the next step to handle. In order to gain a better handling of the resistance, some guidance may be given to coaches and change managers.

According to Kropp & Meier, firms are not getting through the whole potential of Agile project management (2013). The potential can be related to the Agile maturity. To evaluate the Agile maturity and the implementation, different assessments are needed: goal achievement, the setting up of KPIs and feedback. We must carefully consider the project success as well, because project success does not imply project management success and vice versa. The notion of feedback and improvements are consistent practices in the Agile models. The shortage of the Agile maturity was revealed in 2003 (Abrahamsson et al, 2003).

All these questions about goals before and after the implementation are at the core of this study. This is clearly a question of business decisions on implementation of Agile Project Management methodologies.

In the literature, the decision of Agile adoption in software development is not deeply analysed based on the characteristics of companies or projects. A lot of studies already found out that financials and a changing context are positive impacts to Agile adoption. But we do not know precisely whether companies determine the goals before the implementation and if they had specific expectations.

Throughout this thesis, we will answer the following questions: What role does the goal play in Agile methods adoption? Have they been taken into consideration? Why and why not? Is there a link with the agility position?

The objective of this thesis is to approach the reality in companies in terms of goals, implementation and assessment. Through the exploration of qualitative surveys, we will analyse the factors that companies consider in such an implementation. This will give tracks, advice and support future Agile adoption.

Agile intervenes everywhere in companies. Conference talks showed that it not only touched software development, but also Human Resources, Marketing, Communication, etc. The fact is that a lot of theory exists, but we do not know how the companies deal with that in reality. Each company follows a theoretical path and customizes following their needs or

problems. Everything begins with the software development. Hence, this thesis could help to see how it works in reality and what could be improved in their short and long-run. The Agile project management is a subject that is nearly an innovative mind-set, it appeared only in the beginning of the twenty-first century and has already a great expansion. This subject is not only interesting for the university in the understanding of companies' way of working. It is also a subject that a lot of companies go into blind and calls for guidance.

In order to have foundations of our analysis, the first part will deal with theoretical concepts in link with Agile and management.

The first chapter will first present what defines software project management. These words are often used but the structure and the requirements implied are not known by everybody, this we be dealt with in the first section. Its purpose is to clarify and act as a starting point. Moreover, we will review the two major project management frameworks.

Section 2 will go deeper into the Agile project management, explaining the mind-set and the methodologies. The main Agile methodologies will be presented briefly. Afterwards, advantages and limitations will be discussed. Then we will go further with a presentation of scaled method that have the specificity to manage several related projects. Finally, we will come to the application of these methods. The possibilities of tailoring will be discussed: contingency factors – having a portfolio of methods, method engineering - creating a new one or a partial adaptation – modification of some specifications in a method.

The third section focuses on management of the decision process and the change process. The first section will refer to the overall context and background to consider. Then, we will go deeper in the strategic, operational and organisational goals of the company. Next, we will consider the change management and the important elements of change. Then we will talk about the assessment of Agile, the different tracks that may exist to try to assess Agile maturity. Different concepts will be approached (KPIs, PDCA, Shu Ha Ri and Version One).

The second chapter of the thesis deals with the research motivation. The research is based on goals that a given company has when implementing an Agile methodology. We will focus on the role of motivations and its consideration that companies may have in their implementation.

Chapter 3 focuses on the research methodology. The qualitative interview motivations and its characteristics will be explained. Then the particularities of the interviewees will be

presented like functions and sectors. Afterwards, the collection of the data will be described with the elaboration of the sample, the construction of the interview and its realisation. Finally, we will speak about the treatment of the analysis.

Chapter 4 deals with the results of the qualitative interviews. The results are divided in five categories. First, we will talk about the project management circumstances before Agile implementation. Afterwards, the list of goals highlighted of the interviews will be synthetized. We will then speak about the change management put in place in the different large companies (The SMBs are not mentioned in this chapter because they implemented Agile directly). Then, we will see how companies assess their implementation. Finally, we will discuss about the mind-set, a subject that was very debated during the interviews.

The last chapter is the discussion. The aim of this part is to discuss the different topics linked with the research question mentioned in chapter 2. The different subjects of the discussions are the role of the motivations that companies may have and the consideration of this role. The consideration has been divided into 3 sections: the different possibilities of methods applications, the resilience that companies may have, and finally the rare evaluation of goals and methods during the assessment.

---

## Chapter 1 – Software Project Management: an overview

---

### 1. IT project management

This first chapter skims the scope of the thesis. Project and project management will be defined in the following section. Later on, the two most used project management models will be presented.

#### *1.1. Project*

There are several definitions of this concept, hence the explanation will be as explicit as possible. First, a project implies a change. It can be on different subjects: human, financial or structural resources. (Schwindt, 2005).

According to Hanniff & Salama, a project has characteristics, it is temporary and unique (because none is equivalent to another). It has organised resources and, last but not least, predetermined goals, which are the purpose of the project (2016).

There is a variety of types of projects depending on goals and social complexity. At first, if the project is **closed**, that means that the objective is known. On the contrary, an **open** project has various possibilities of ending. The objective is more conceptual, for example, improving something. Then, there are also two possibilities of social complexity: **low or high**. (Kuster et al., 2015).

Those **four categories** can be put in a matrix. The standard project is the simplest project to implement. Another type of project is the acceptance project. The goal is in this case well defined and it consists in formalised and standardised steps explained further in this paper. They often need the opinion of different departments, so they are complex repeated projects. Furthermore, there are potential projects where the final objective is not determined at all, but it concerns a small group of the organisation. It has very few impacts. And finally, the pioneer project has a high risk in the organisation because a lot of departments are involved, and the change will be radical and innovative. (Kuster et al., 2015)

After discussing about types of projects, there are also a flow of work. There are four consecutive phases to execute a project easily, this is called a project life cycle. The first step is the **initiation**. It consists in:

- the development of the idea through a feasibility study,
- the determination of scope and requirements,
- a WBS and OBS elaboration and establishment of needs.

Therefore, a close collaboration with the stakeholders is supported.

After that, the project needs to be planned in time which is also the case of a budget, the resources and communication needed for its execution. That is all the **planification**, the second step.

The third step consists is the **evaluation of the ongoing project**, which implies a monitor, a constant reporting, planning adaptation and seeing if the change keeps going.

And the last step is the **closure of the project**: the delivery of the finalised project to the customer and the bill (Hanniff & Salama, 2016).

The success of a project is determined by three constraints: **time, cost and quality**. These are the three criteria given by the customer at the beginning of the project. It creates the Iron Triangle (Atkinson, 1999). It is important, and not easy, to balance these three criteria during the project. An extreme in one of these will imply the unfeasibility of another.

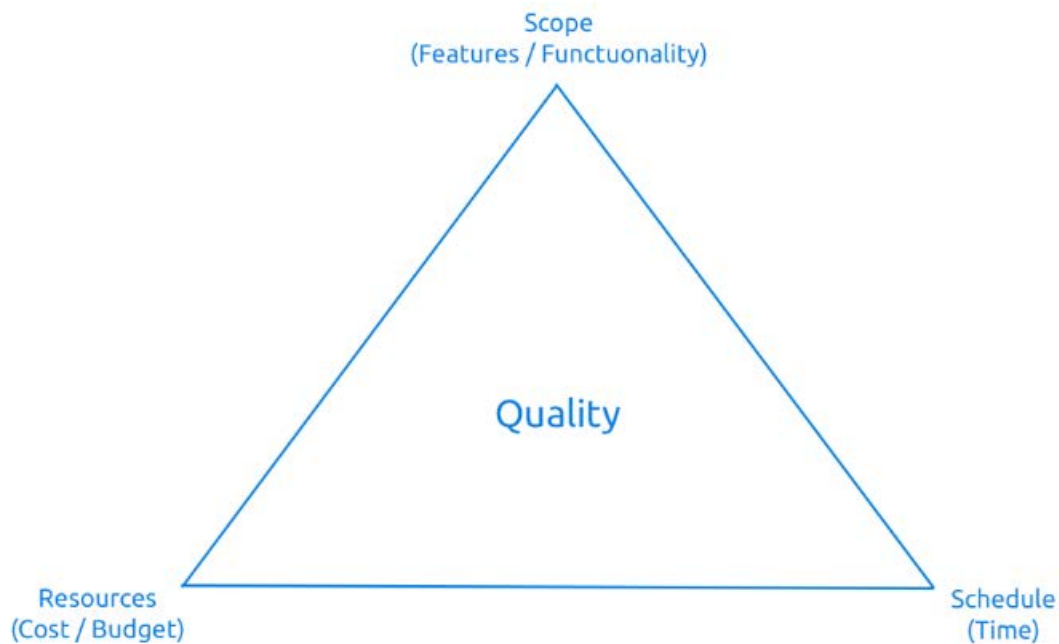


Figure 1: The iron triangle

Source : <https://www.easyproject.com/fr/contacts/la-gestion-de-projets-en-toute-simplicité-blog-conseils-ressources/673-methodologies-de-gestion-de-projets-waterfall-ou-Agile>

Three other constraints are important in the success:

- 1. Scope**
- 2. Resources**
- 3. Evaluation of risks**

Scope and resources are needed to take in consideration the three first. And then the risks, in each requirement, objective or change of those cause potential problems. Hence, it is essential to evaluate the risks (Hanniff & Salama, 2016).

### *1.2. Project Management*

Project management is quite a wide concept. The different tasks that a project manager has to go through will be explained in the following pages.

The notion of project management appeared in 1950. The development of project management was low compared to now, it only consisted in a mathematical planning and a control process for large extend projects. It evolved over time, as did its recognition. Its recognition was full when the PMI, the Project Management Institute appeared in 1969. Since

then, the world keeps improving the tools: they are constantly freshened and performed. Now it handles much more than only technical complex tasks. It expanded to several domains, covering more than products and services creation, it helps every management department (Kuster et al., 2015; Hanniff & Salama, 2016). It became known worldwide and in 1990, we first heard about “projectification of the society”, it is a way to handle our daily life as project management (Hanniff & Salama, 2016).

Project management is the deals with all project parts mentioned above. To summarize, here is a scheme of the project management of nine knowledge areas (which are explained below):



Figure 2: The project management knowledge areas

Haniff & Salama, 2016, p.13.

**The project integration management:** every project needs the right analysis for a good attribution of methodology and tools. It is all the meaning of the process and coordination.

**The project scope management:** it consists in the determination of the project limits. It is crucial to specify what is in the project and what is not. That includes the commitments, the Work Break-down Structure (WBS), and adapting the control processes.

**The project time management:** Time in a project is a critical objective and determining constraint. It is a great decisive of a project's success. It implies a good command of the estimation of the deadlines and the key resources needed to achieve the project scope. For each change in the project process, it evaluates the impacts with tools such as a schedule of the resources, a network analysis, a Gantt chart, and project crashing.

**The project cost management:** This is the second criteria mentioned before in the list of constraints (in section 1.1). For this knowledge area, we must know the necessary resources through all the project life cycle. The tools that can help are a good estimation of cost, cost management plans and an earned value analysis.

**The project quality management:** The conformity of the norms and the needs of the project are important. Hence, the good respect of the quality norms, the objectives and policies. To have the possibility to judge this quality, it is important to determine a classification of quality at the beginning of the project life cycle. The modes of quality determination are: the use of a quality plan, quality control measures, and change requests.

**The project human resource management:** In a project, the team is a key success factor. To handle a team at best, it is important to have a lead and a good overview of the team's work. Motivation, dispatch of responsibilities, conflict resolution, optimise of competences are examples of core activities for this knowledge area. Tools for the implementation of these activities are: the Organisational Breakdown Structures, responsibility charts, and team buildings.

**The project communication management:** It is all about information flows. The collection, distribution and storage of information with all the actors – which could be team players, stakeholders, customers, businesses – is huge. Hence, communication, project plans, evolution, update reports, and online system of information flows are used to optimise the communication flows in general.

**The project risk management:** There are a lot of risks in a project. Therefore, preventing the risks in avoiding them or propagating the impacts are tasks that have to be done in the beginning of a project. This prevention is very important, it can attenuate the proportion of a problem.

**The project procurement management:** Each project needs inputs such as material, products or services to deliver a project. For the purchase and acquisition, it requires an

inventory and a good relationship with suppliers and partners. Negotiations, administration and contracts will be a part of this dimension (Hanniff & Salama, 2016; Semal& Raucen, 2017).

Project management helps to optimise the organisation of a project, to encourage direct collaboration in teamwork, to give a precise responsibility to each member (Kuster et al., 2015).

A good project management methodology is a key competence to remain competitive on the market. It helps to reduce the costs, optimise the resources, and have a higher quality. But also, it is a way to prevent potential risks (Hanniff & Salama, 2016).

### 1.3. Conventional Software Project Management

Waterfall is the first project management technique encountered. It can be seen as the traditional way of thinking. This section will provide a view of this concept based on project management.

The popular traditional project management methodology (PMM) is the Waterfall model. This model works with steps, the next step will not be reached before the previous one is completed (Flora & Chande, 2014). It is based on a detailed scope, with which we can then start the introduction and planning phases. Based on the introduction phase, the planning phase elaborates a rigorous map of deadline output. You will find below the typical stages of a Waterfall model (Haniff & Salama, 2016).

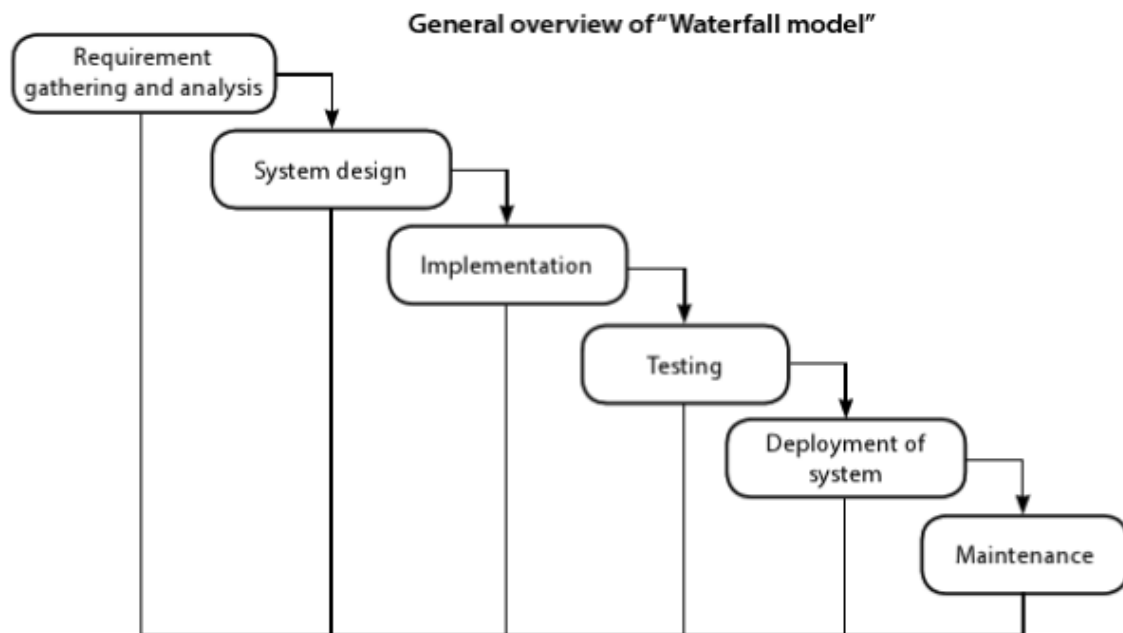


Figure 3: The waterfall model

Haniff & Salama, 2016, p.150.

In this model, we see that the first step is the “Requirement gathering and analysis”. This step shows the particularity of Waterfall. The scope constraint from the iron triangle is fixed. The project manager can have an impact only on the deadline and resources (EasyProject, 2018; Aljaber, 2018).

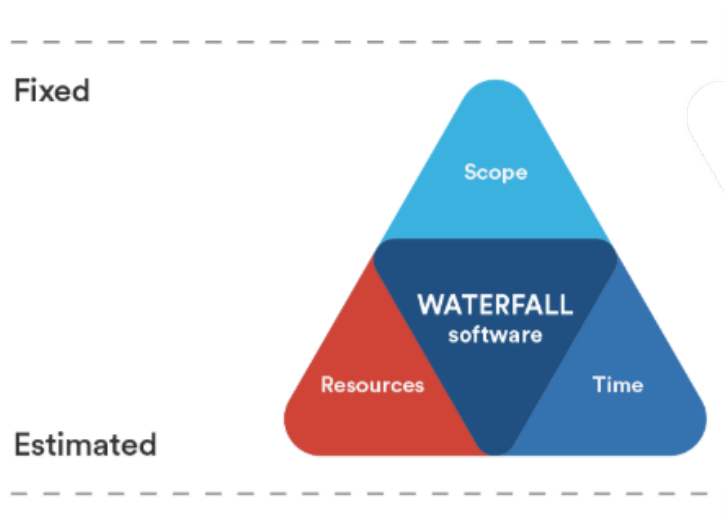


Figure 4: Waterfall's iron triangle

Source : <https://www.atlassian.com/Agile/Agile-at-scale/Agile-iron-triangle>

There are multiple advantages to the adoption of Waterfall. At first, it is easy to understand and to use. The strict process permits to be guided and be aligned in the team. Each phase is well separated. The overview plan can be replicated and adapted for next projects (Hanniff & Salama, 2016).

But there are also disadvantages, such as the inability to change or to make innovation. It will cause problems with changing requirements. Furthermore, it takes a long time before delivery and there are a lot of risks. It will not be adequate for the long term and substantial scale (Hanniff & Salama, 2016). Flora & Chande (2014) added that the issues of the Waterfall model are the overcharging project, the important delay and disrespected guidelines.

These disadvantages in nowadays' context, specifically in IT or software development domain, pushed companies to another project management methodology. It is in order to fill the gap that Agile Project Management appeared (Hanniff & Salama, 2016).

#### A. Prince 2

PRINCE 2 is a waterfall method, it means Project IN Controlled Environment. It is process-oriented and is based on a good documentation. The project is working for a product. It consists in 4 phases: pre-project, initialisation phase, delivery phases and finalisation. This setting up has the goal to pursue four objectives: direct, manage, deliver and plan (Guilmin & Nghunza, 2017).

## B. PMBOK

Another PMM is PMBOK. It is a framework helping to reach the fixed objectives. The communication is also more suitable thanks to the same language as the methodology (Kolp, 2017).

These two Project Management Methodologies are common and structured. Common (or generic) means that it can be applied to other domains than IT. The structured part is the particularity of the Waterfall model.

### *1.4. Agile Software Project Management (APM)*

This section is the first bases to know before choosing any Agile method. What does it mean to be an Agile company? The definition of an Agile company is rather complex.

When Agile comes in mind we think about quickness, easy move, problem solver (Macmillan dictionary, 2018). This word is more and more often used in business context. It began in project management focused in software development. Then, this mind-set was developed in other business department. Agile project management is, according to Macmillan dictionary: “a way of managing projects based on dividing the work into a series of small tasks and using regular feedback at each stage to make changes to how work in later stages is done. Agile project management is sometimes simply called Agile” (2018).

The Agile way of working is alternative to the traditional project life cycle. Agile appeared after several problems of the traditional methods, including the main one, the waterfall model. It is technology-directed and innovation-directed. Unlike the traditional way (based on a detailed scope and a predictive plan), Agile works on an incremental and iterative cycled system with an unspecific scope. The scope is detailed more and more during the process of the project and is only clearly defined at the end of the project (Flora & Chande, 2014; Hanniff & Salama, 2016).

APM permits the team to begin a project without a precise and closed definition of the project and the scope (Schön, Escalona & Thomaschewski, 2015). In the iron triangle, this implies that time and resources are defined and that the scope is free, as we can see on figure 5 (EasyProject, 2018). Detailed planning is not vital anymore in the Agile model (Haniff & Salama, 2016).

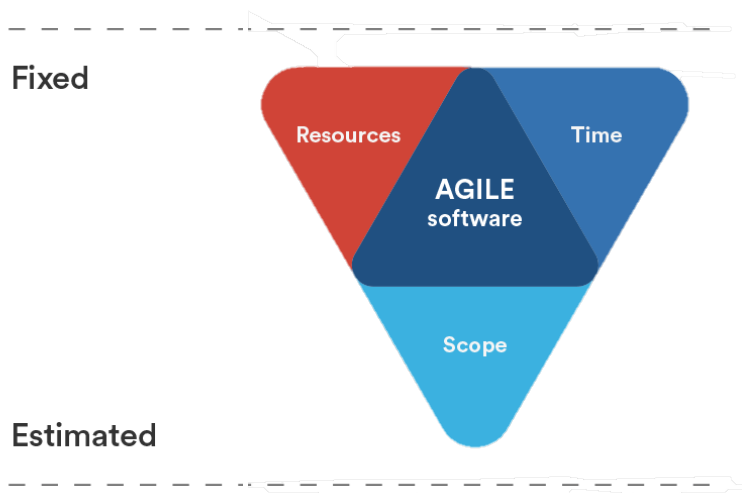


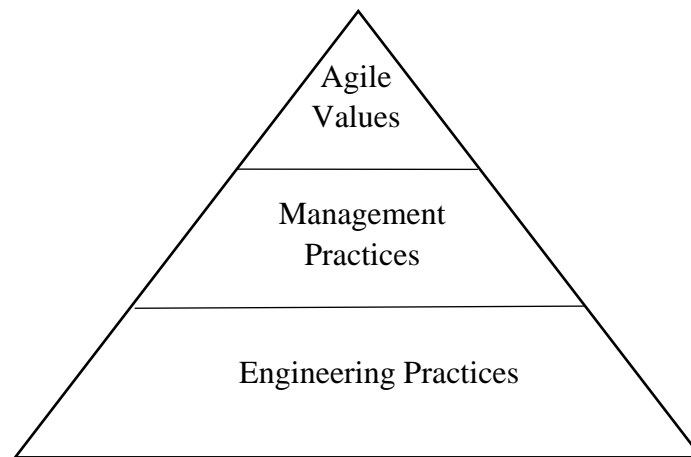
Figure 5: Agile's iron triangle

Source : <https://www.atlassian.com/Agile/Agile-at-scale/Agile-iron-triangle>

It gives more flexibility than the traditional models thanks to the iterative systems (Schön et al., 2015). Hence, we can define the Agile model like Flora & Chande as an: “iterative and incremental development approach in a highly collaborative and timely manner which allows the project to adapt the changes quickly” (2014). These benefits are encountered when the workers have to handle requirements changes or an alignment with the business but it also permits the engineers to optimise productivity (Campanelli & Parreiras, 2015). In that way, the changes can be handled throughout the whole process without beginning again. Hence, the definition of the project is developed during the project itself (Schön et al., 2015).

The Agile model satisfies the wishes of companies to produce quickly, at a lower cost in an unstable and constantly changing world. The phases do not have to be finished before starting another. The Agile model gives an output as soon as possible and then constantly performs it (Flora & Chande, 2014). Haniff & Salama added that the iterations give the opportunity to make an analysis on the delivery and have control on risks. It reduces them, and it controls the time-to market and the cost states (2016). This gives a flexible approach that can absorb technology, cultural or economic changes. This meets the expectations on all parties that work on it, every view is considered (Campanelli & Parreiras, 2015).

Agile software engineering requires some competences. Kropp & Meier explain Agile software engineering competences in a pyramid way that follow (Figure 6):



*Figure 6: Pyramid of Agile competences*

Source : Kropp & Meier, 2016, p.183

The bottom is the technical part, to sustain every work, techniques are needed. For this competence, eXtreme Programming (XP) is the suitable methodology including practices like unit testing, clean coding or collective code ownership.

Then the management practices level refers to the organisation and the way of working which refers to the Agile Project Management. It handles the planning, the cycles and the communication in the team. For this level, Scrum is a good support.

The last competence is the most difficult to attain. It is a change of attitude, it consists in following the Agile mind-set, its values and principles (Kolp and Meier, 2013; Kolp and Meier, 2014).

## 2. Agile Software Development

This chapter will go deeper in Agile project management. Therefore, we will first develop the first top of the pyramid of Kropp & Meier: the Agile mind-set. Then, we will explain the different methods briefly. We will discover the advantages and disadvantages after that. And we will finish with their application in companies.

### *2.1. Agile Manifesto*

For a better understanding of what the Agile model is, the Agile Manifesto will be used as a guideline. To respect the Agile manifesto, we have to take all the project elements globally and every part has to be considered. These elements are: team spirit, communication, the roles, the tasks and the process itself. (Schön et al., 2015).

Agile model implies four core values and twelve principles from the Agile Manifesto. The values are: the individuals and interactions before processes and tools, delivering working software before a detailed documentation, customer-oriented working and collaboration before contract negotiation, and finally, the response to change before planning (Campanelli & Parreiras, 2015; Flora & Chande, 2014; Haniff & Salama, 2016; Kolp, n.d.; Schön et al, 2015).

In the following, you will find the twelve principles of the Agile project management (Campanelli & Parreiras, 2015; Flora & Chande, 2014; Schön et al, 2015).

1. Fast delivery in a continuous manner.
2. Encourage changes whenever in the process.
3. Frequency of working delivery is high.
4. Interactions must happen between business and programmers.
5. Motivation is very important in such a model, it is the motor of the process in the project.
6. Communication such as face-to-face or different ways are required.
7. Running software is the key measure to determine if the project goes well, the project has to keep a rhythm of working.
8. Sustainable environment.
9. Technique must be perfect and the design almost as good.
10. Simplicity.
11. Self-organise team.
12. Continuous adjustments.

To follow these principles, the team must be autonomous to decide quickly about the resources needed and to be in the right environment. This leads to a better conduction of the project by motivated people. Blind confidence has to be put in place to give a self-satisfaction to bring a project to a higher level (Campanelli, & Parreiras, 2015; Schön et al, 2015). Then, it is better to accept any changes, even if it is late in the process. Adopting this sort of behaviour leads to know and deeply discover how to meet the requirements of the customers (Campanelli & Parreiras, 2015). Schön et al. added that the reaction to these changes are very important to be Agile. Therefore, communication has to be fluent between the parties (2015). The opinion of every stakeholder is important. Hence, it is capital that they give their meaning no matter their function in the process. Thus, the project implies a continuum of exchanges between the business and the client throughout the process. In order to make these exchanges possible, the frequency of work report has to be on short scale. It varies from one week to several months maximum, but the shorter the better for everyone (Campanelli & Parreiras, 2015).

The Agile Manifesto is not easy to teach because it implies a change of mind. It encourages qualities such as bravery, respect and openness. The difficulty lies in the fact that these values have to be in the bones, not only be read and then forgotten. The best way to instil the values to the learners is through the practical way (Kropp & Meier, 2013).

## *2.2. Agile Methodologies*

### *A. Scrum*

The scrum method is the most adopted in the Agile working environment. The goal of Scrum is to provide a lot of value to the product as soon as possible and to do so it gives the ability to deal with complex problems (Campanelli & Parreiras, 2015).

The principle is an iterative and incremental approach with a collaboration between different teams. The essence of this is the knowledge, the experience and the decision-making giving the context (Campanelli & Parreiras, 2015; Aubry, 2010).

Mikulenass et al. mentioned that Scrum is preoccupied by the process (2011). The process consists in several releases, depending on the project length, composed by 4 to 6 sprints. There is a “zero” sprint, which is the planification time before the first sprint. In that period, the deadline and costs are fixed, and some scope details are determined. The scope will evolve in time. The sprint is a fixed time bloc that leads to a delivery. The sprints always last the same time once determined. The budget is already fixed. At the end of each sprint,

there are a delivery and feedback, no matter if the work is completely done or not. A sprint is an amount of time and does not represent the achievement of a task (Kolp, n.d.; Aubry, 2010).

The story is based on a backlog which includes all the objectives, requirements and enables the easy access to the project view. This gives a clear view of what has to be done in production (Flora & Chande, 2014; Kolp, n.d.). The work and tasks are planned in stories, epic and feature (Kolp, n.d.). Each sprint receives a story characterised by INVEST charts: Independent, Negotiable, Valuable, Estimable, Small, Testable (Agile Alliance, 2018). Once a story has all these characteristics, this story is considered as done and ready to be launched in a sprint. Before a sprint, the team determined which story is prioritised and designated as epic. On these epics, the planning will be done, at the zero sprint, with the determination of what means “done”. The stories are separated in tasks and each team member take the responsibility of tasks (Kolp, n.d.).

During the sprint, there is a daily meeting which pursues a development of collaboration in the team. Hence, every day, during 15 min, each team member displays what is done, prevents what will be done and gives the difficulties encountered. At the end of a sprint, a review is planned. Every stakeholder is invited and even stimulated to have as much feedback as possible on the results of the sprint. This is called retrospective, it pours out a continuous improvement (Kolp, n.d.).

The different actors of a scrum product are the team, the stakeholders and the experts. The team has three major roles, the programmers, the Scrum Master and the Product Owner. The Scrum Master is the facilitator of the Scrum application in the organisation. It is not a manager, he favours the team and helps to eliminate difficulties (Kolp, n.d.).

Scrum facilitates learning curve and knowledge transfer (Petersen & Wohlin, 2009). It shows itself as a light, accessible but difficult to implement method. Unlike some company businesses think, it does not give the opportunity to change at any time or to interrupt a team for an emergency. It promotes good change but at the opportune time (Kolp, n.d.).

### *B. eXtreme Programming (XP)*

Extreme Programming (XP) is the method adapted for small teams with imprecise requirements and a high rate of changes. This method has the advantage of not being too complex and a gain of money. The unit tests are made throughout the code activities. All the system is often put together and the developers work in pair programming. The design is

simple, and the releases are often made. This method is suitable for simple projects but not for large, difficult or complex ones (Campanelli & Parreiras, 2015; Flora & Chande, 2014).

In addition, the main objective is to improve quality and to match the project with the constantly changing customers' needs. If the customer does have a lot of time for the project, it is not a problem for this method (Flora & Chande, 2014; Lin et al., 2013). The analysis of the needs must be supported by an in-the-field exploration, it is the development of the metaphor. This metaphor makes everyone see eye to eye on the terminology and scope. Some practices are paired programming, self-organise teams, simplicity in design. In such ways, the releases are regular, and the cycles are short. For the configuration of the worker aspect, it matches better with small to medium size team that are at the same place. The different actors are the XP coach, customer, Administrator, Programmer, Tracker, Tester, Consultant (Flora & Chande, 2014).

Mikulenias et al. concluded that XP adds some techniques and practices that make a complementary method to the Scrum method (2011)

### *C. Lean*

Lean management takes its origins from Toyota. Toyota sort out its new Production System which leads to the inspiration of the Lean. The Lean management takes the point of view of the customer and explores if there are possibilities of new ways of rethinking his values. The company wants to optimise the satisfaction of needs and not only gives one solution among others. Another principle is that they consider employees as resources. And the third one is the profitability of the company. This methodology is respectful to people, customers, employees, and promote social actions (Flora & Chande, 2014; Kolp, 2017).

Such as in manufacturing, the goal of the process is to avoid any waste. The performers will concentrate on the needs of the customer and that only (Campanelli, and Parreiras, 2015). Hence determine the value of the customer and challenging the process of eliminating waste are the first two main principles. The three others are a continuous delivery, a work done depending on demand and a constant evaluation and improvement (Flora & Chande, 2014; Kolp, 2017).

The lean includes methods like Genba and Kaizen. It also implies tools such as Just-in-time – minimizing the storage part, Jidoka – automation principle-, Kaizen, - continuous improvement and Kanban (Kolp, 2017).

#### *D. Kanban*

Kanban is a tool of Lean which enables to respect the goal (Campanelli & Parreiras, 2015). This tool is considered as a method of its own right for businesses since 2012 (Ngakam & Tchunte, n.d.). It does three things: having a clear vision of the work flow, restricting the work in production and determining the time left to get the product done. The qualities of the method are adaptable, visual and cost focused (Campanelli & Parreiras, 2015; Flora & Chande, 2014). Ngakam and Tchunte add the dimension of continuous and evolving change that we rediscover from its origins, the Lean management (n.d.).

#### *E. Feature-Driven Development (FDD)*

It is used only for the design and the prototyping phase. There are frequent deliveries to maintaining quality of the process. The different phases of the process are developing a feature, building a list for it, planning, designing and building the feature. It is the best fit for critical projects. The big quality of Feature-Driven Development is the capacity of adaptation of the method (Campanelli & Parreiras, 2015; Flora & Chande; 2014).

#### *F. Dynamic Systems Development Method (DSDM)*

The quickness is here the capital measure. The project is based on what we already have as resources and time and so predefined. The numbers of features in another hand is flexible (Campanelli & Parreiras, 2015). The customer plays a role all along the process (Lin et al., 2014).

#### *G. Adaptive Software Development (ASD)*

ASD is the best for solving problems large and complex. There is some guidance to avoid mistakes but not too wide to let creativity and it will not slow down or make the project not flexible. Concepts are iterative development, incremental for features and prototyping (Campanelli & Parreiras, 2015). It is the best Agile method to have a creation rapidly and a rapid evolution.

It consists in three phases: speculate, collaborate, and learn. The first one is the project set up, the second is the development of the feature and then the quality evaluation. It is feature-centered. There are four actors: executive sponsor, in charge of the product, participants, developer and customer (Flora & Chande, 2014).

## *H. Crystal*

It is a family of methodologies in which the right is chosen for the project needs and goals. It gives a list, but it also guides in the method selection (Abrahamsson et al., 2003). The project size and criticality are important here. The qualities of Crystal are the communication and collaboration and the time-boxed iterations (Campanelli & Parreiras, 2015). The projects are classified in colours depending on the project to extend or communication needs (Flora & Chande, 2014). Lin et al. add that Crystal is efficient and project safety preoccupied. It does not focus on the process but on people (2013).

## *I. Agile Unified Process (AUP)*

It is focused on Web-based development. It is an easy framework to understand and apply (Flora & Chande, 2014; Lin et al., 2013). It uses Agile techniques like Test Driven Development (TDD), Agile Modeling (AM), Agile change management. For each iteration, the team works on 7 different categories: model, implementation, test, deployment, configuration management, project management and environment. Each category has artifacts, activities and predetermined roles. The core principle is that the staff is autonomous, he knows what he is doing. The AUP gives freedom in the tools used for the project (Flora & Chande, 2014).

### *2.3. Advantages of Agile project management*

The advantages of the Agile methodologies compared to the traditional ones are the on-time delivery and the customer satisfaction (Flora & Chande, 2014; Schön, et al, 2015). Another advantage is that the transfer of knowledge is easier than to communication and feedback at the end of each iteration. The value of the customer is very high in the process and it is why the result is more adapted to the needs and the environment. We can also see transparency, control and quality are improved with the APM (Flora & Chande, 2014; Petersen & Wohlin, 2009). Campanelli & Parreiras add that there are other aspects. The delivery is not only on time, it is accelerated due to the model and, moreover, there is a bigger quality and productivity at the end of the project. These elements contributed to improve Information Technology, the project is rightly aligned with the business and finally is the large flexibility that makes the project changes possible. Moreover, the Agile model enables to keep an eye on budget and do not exceed it (2015).

#### *2.4. Limitations of Agile Project Management*

Benefits have been demonstrated by searchers of the Agile model, but it does not consider the complexity and the big efforts to put it in place. The context of the organisation such as culture, the dealing of resistance, self-estimation, the involvement and sponsorship of the leaders have a big role to play in here. Moreover, the company needs a lot of experience to handle the Agile model and therefore it takes time before the process can work efficiently (Campanelli & Parreiras, 2015).

For large projects, it is not ideal. The requirements are insufficient and unclear that does not give enough information for a new team player. Even if it adapts to every situation, it implies that the changes are time-taking (Flora & Chande, 2014).

Petersen & Wohlin pointed out in their survey that the continuous testing is hard and takes a lot of work. The communication between teams or the fear of the managers about power loss is not enough handled in the APM (2009).

Campanelli & Parreiras evoke the fact that some projects do not match with the Agile model because the organisation cannot permit themselves to put the business apart to put the model in place. The business must achieve the goals to sustain the company (2015).

For a project where the objectives and scope are clear with no ambiguity and there will be no changes, customer interaction is almost not vital, and the stakeholders are not needed, then the Agile model is not appropriate. In these circumstances, traditional model is preferred in that case (Mikulenias, Butleris & Nemuraite, 2011).

### *2.5. Agile scaled methods*

For an entire product development, Agile methods are not enough, management between projects are needed. It is for that purpose that larger-scale models appeared. The objective is to coordinate and handle the complexity and in order to fill the gap of agility for large-extend project (Kettunen & Laanti, 2017). Scalable Agile model is very hard to implement (Petersen & Wohlin, 2009).

#### *A. Large-Scale Scrum (LeSS)*

LeSS build a system that bounds the feature team together for coordination on the upper level (Kettunen and Laanti, 2017).

#### *B. Scaled Agile Framework (SAFe)*

SAFe builds a value flow but also an economic vision for the company. It implies a release of the system called ART, Agile Release Train. It optimises the bounding between the teams and the intervals of time that they work together. There is an ART-team that facilitates the application of the principle (Kettunen & Laanti, 2017).

#### *C. Spotify*

Spotify is a scaled method that was made by the firm Spotify itself. It adopted and adapted the Agile methods until they arrived at their own method. This method is not finished and will never be, the Spotify company insists on it. It is a continuous improvement. It is a success model, this is the reason why other companies try to copy it (Fernandes, 2017).

### *2.6. Selecting and Tailoring Agile methods*

There are two possibilities to implement Agile methods. The first one is to put in place Agile method without tailoring. In that case, the SAMI can be put in place. The SAMI (Sidky Agile Measurement Index) tries to be a guide in the adoption of Agile practices. It is composed of 5 steps, linked with core values of Agile Manifesto. The first step is upgrading communication and collaboration. The second step is focused on the delivering that should be early and in a continuous way. The quality is then the step to undertake, the delivery must work but also be integrated. The fourth step is the handling of change encourage by continuous feedback. And finally, the environment must support the Agile working. This adoption methodology is adequate for each Agile method because it is based on Agile adoption maturity process and the values of APM (Campaelli & Parreiras, 2015).

The second way of implementation is tailoring methods. The main idea of tailoring Agile methods is that no software development method is good enough to fit to the project. So, we pick the practices that we need. The company will spend efforts in putting in place only the facets of the Agile method that could help the company depending on the values, the way of working and the culture. This leads to a match with the business goals. It is less complex to put in place. To have an adapted method to a project, three main elements interfere. The first one is the portfolio in which we can choose the method. This portfolio depends on the company. The second one is the criteria on which we base the choice and finally the third one is the context of the project, the culture and goals of the organisation (Campanelli & Parreiras, 2015).

Two possibilities to fit projects, one or more methods in the portfolio predetermined by the organisation or create a new one. These two are called contingency factors and method engineering (Campanelli & Parreiras, 2015).

#### *A. Contingency Factors*

This approach considers all the context which means the organisation's and team's structure and culture, the methods available in the company, the context characteristics such as its level of uncertainty, its finality and the particular structure. This practice is not easy, it needs a lot of Agile project management knowledge and experience to choose the right method (Campanelli & Parreiras, 2015).

#### *B. Method Engineering*

The method engineering is the constitution of a new method based on practices of other existing methods. It is centered on the organisation or the project. The purpose here is to have more efficient methods due to the perfect match with the environment. Naturally challenges are huge. As the creator of the method the company has to face all the problems of the connexion of the different practices (Campanelli & Parreiras, 2015; Mikulenas et al. , 2011).

There therefore exists a process: the Strategic Analysis for Agile Practices (SAAP). It integrates the business goals in the Agile adoption. It uses the Method Engineering and the Balanced Score Card (BSC) together in order to include the strategy in the Agile project management (Campaelli & Parreiras, 2015).

### *C. Partial adaptation*

A third tailoring option is mentioned by Mikulenas et al. The partial implementation is an initialized adaptation project that contains several phases. The first phase, initialisation: the user identifies concerns as application areas, selects the related elements and their minimal levels. It is a trial of a new mixed model. The intermediate phases configure all the elements to match with the others (2011).

The different families of elements are Guidance Kind, Stage Kind, Producer Kind, Work Unit Kind, Work Product Kind, Language Kind, Category Kind, and independent Kind (Mikulenas, Butleris, and Nemuraite, 2011).

Companies have their own know-how, without any will to rebuild but willing to be more successful with a supplementation of their competences. But how to combine and put in place the methods for their partial methods? It is why people want to divide the Agile methods in elements to fit the partial adoption. Thanks to the framework of Mikulenas, the partial adaptation is possible because we respect the requirement to the solution of the sub-problem of making a new model, “pick pick practices” (Mikulenas et al., 2011).

Direct requirements are: agility requirements, method decomposition, application areas, element levels, criteria and prioritisation, implementation model. The indirect problems to deal with are abstraction levels, structuring, merge and generalization, coupling, pattern and finally extensible metamodel (Mikulenas et al., 2011).

## **3. Management and Agile Software Development**

The first section will explain the state of the world. The VUCA world is a big perspective of the context that project managers have to deal with. Then, some basics of business strategy linked with the interviews will be discussed, which is addressed for each decision that a company takes. To follow the process, after any decision comes the apparition of change, which includes fear and resistance. The evaluation of a decision is then approached.

### *3.1. VUCA world*

Since the beginning of the 21st century, it is stated that we are in a VUCA world, Volatile, Uncertain, Complex and Ambiguous world. A world with such a rapid rate change

but also with a high degree of difficulty may not take the liberty to follow a strict and rigid plan like a waterfall model. It is why Agile offers the possibility to fight against this difficult world with flexibility, rapidity, adaptability and visibility (Reeder, 2017).

Software development is one of the domains that contains the most uncertainty. It has a great impact on projects and therefore a good management is needed. Fortunately, Agile is considered as a good guidance in this uncertainty. There is then a possibility to make a state of the art and find opportunity in this phase (Dönmez & Grote, 2018).

### *3.2. Strategy, decision-making and operational*

The Strategy of a company is vast, it is why I only presented the elements that are approached in the interviews. These approaches are before all a diagnose of the environment and then the strategic goal and operational goals. Then there is an explanation on how a decision is made.

The strategy of a company is a manner to express the purpose, the mission and the long-term path to add value for their clients and all stakeholders, taking into account the environmental threats and opportunities. Based on their strategies, companies take decisions (Coeurderoy & Vas, 2016).

This decision is based on the SWOT analysis. The definition of the Strengths, Weaknesses, Opportunities and Treats are important for challenging a company. To establish the SWOT analysis, there are two diagnoses to do, the first one is the external diagnose based on the opportunities and treats. The second diagnosis for strategic analysis is the internal diagnosis including the Strengths and weaknesses (Coeurderoy & Vas, 2016 ; Guilmot, 2017).

To help analysis in the external world, the 5 forces of Porter can estimate the situation of the competitive world (see Figure 7). The parts that we will focus on in this thesis are the competition and the barriers to entry. These two forces are based on the differentiation of the services or products that a company may give (Coeurderoy & Vas, 2016). These will be the two motivations to acquire this technology.



Figure 7: Porter's five forces analysis

Source :<http://applyingconcepts.blogspot.com/2011/12/porters-five-forces.html>

In theory, every decision of a company will be based on the SWOT process to have a good strategy. This also includes the management of technologies. We can consider Agile methods as a form of technology that helps the companies' way of working.

Weil developed a chart that presents the management of technologies that a company owns. He gives a perspective of all the impacts and the assets that this includes. Technologies need a network that helps the company to learn and develop more these technologies. Therefore, it has to manage the research programs to have access to external technologies when needed and give the possibility to the company to learn and develop the assets linked to these technologies. In that way, it increases the technologies that can help and optimise the strategies of the company. To be up to date, it is important to evaluate the needs for future projects and strategies of the company. The evolution of the technologies is based on the awareness and constant analysis of the company. This analysis has to be linked to the strategies of the company but also the competitor and the technologies available on the market (Weil, 2008).

The Agile methods will help to manage the research and get more technologies with these projects and have keys in hand to improve the methods. Hence, you can always add value to them. These methods help in the development of the projects and help to learn about the projects with the feedback.

Drury-Grogan defines a decision as a commitment chosen, between other possibilities known or not, that a team or individual takes for a purpose. In the daily practice, the decision-

making is not a process where all options are evaluated and selected because of complex world or not enough available information. Hence the decision may differ from one decision to another depending on several internal or external factors (2013).

During the decision process, the business decides on goals. To have correct goals they have to be SMART: specific, measurable, attainable, realistic and temporal (Taskin and Dietrich, 2016).

The importance of determining goals should not be underestimated. It has at first an impact on performance in an individual way. But it has also an indirect impact on strategy (Abdel-Hamid, T. et al. 1999). It gives a right evaluation measures and, in that way, becomes more efficient. Another advantage of setting goals is having everyone in the company on the same page. Everybody knows what the direction is and how they have to work for it. To go further, knowing clearly the goals give the possibility to all to make the right decisions. And last of all, the goals can also be evaluated for a better repositioning of the company (McKinney, n.d.).

Behind strategy, there is also a need of “operational excellence”. To sustain a strategy, the company needs a support of the operational department, which is the daily application of the strategy otherwise it will not work (Weil, 2008). Strategic goals imply operational goals, that are short-term perspective (Petryni, 2018). Some examples of operational methods are the Just-In-Time, the Toyota system, the time-to-market (Weil, 2008).

### *3.3. Change Management*

The implementation of an Agile project management needs a good task board, a location for team gathering, and have an intensive period of time (Kropp and Meier, 2013). This thus implies a change management.

A change brings a lot of consequences, direct or indirect targets might not agree to these changes. Thus, they might resist to them or would be anxious and sceptical about them. In order to try to provide a change-friendly environment, it is recommended to have-managers who have psy-organisational mind or even people who are uniquely in charge of the change, namely, change managers. (Kuster et al., 2015).

The Agile implementation is a transformation due to the fact that it implies a cultural and strategic change in order to respond to the firm’s environment. The type of change can be a revolution or evolution, and this depends on the speed of change.

Like every change, a change management is needed because different resistance can appear. At first, there are individual resistance because of the personality, personal causes like knowledge or prejudice about this change, contextual cause like fear about his job. Then, there is the possibility of team resistance because of team norms or organisational context, structural reason or due to political reasons. The third resistance is linked with the change itself.

The business must always take into consideration that change implies a decrease of productivity. The targets need time to learn the change. Therefore, it is important that the targets receive support and guidance (Guilmot, 2017).

### *3.4. Evaluation*

After putting a new system in place, the decision-maker must analyse the results to evaluate if the understanding of the world was right and improve again. It is an integral part of the change process. But this learning-evaluating phase is not easy. Because the success or fails are not only based on the new system but also other factors that cannot be controlled (Guilmot, 2016; Weil 2008).

The assessment of the success of a project can be through the constraints like time, cost, quality, scope, resources and activity. But it can also be measured through models like the Project Management Performance Assessment (PMPA). It is focused on project success. Regarding the project management itself, it is a bit hard to determine precisely the benefits because they are both tangible and intangible (Radujković & Sjekavica, 2017).

It is important to know that a project's success does not imply a project management success and the contrary is also true. But a project management success may increase the project success at a substantial level. Project management success can be explained through three categories: factors of project management competences, factors of the organisation and factors of PMM, methods, tools and techniques. These three categories imply a lot of elements, you will find below its organigram (Radujković & Sjekavica, 2017).

Project management assessment (PMA) must be motivated and controlled by the managers. Otherwise it would not be done properly. They are not mature enough to be fully efficient, but it helps the team (Siriram, 2018). To determine the success of APM, there are many different criteria: very good communication, continuous delivery and on time, customer satisfaction. For the business meaning, the criteria of success are the short iterations, and the automation of the team (Radujković & Sjekavica, 2017).

This project management assessment is assessed during the Agile methods. The Agile methods are constructed on a PDCA Deming wheel. Plan, Do, Check and Act consists in four iterated steps. In the assessment part, the steps “check” and “act” are interesting because they analyse the results and they improve them. It consists in a continuous improvement of the method (Manager Go!, 2018). This improves the project management but it does not assess the implementation in the totality.

Shu Ha Ri is a method of martial arts in Japan. This method is learning process that can be applied to Agile. These three stages can show a type of maturity in Agile. This does not involve the strategy anymore, it is only based on the learning stage. The first level is the Shu: the student follows precisely what the teacher says. It does not try to see what lies after the theory. The Ha, is the level where you begin to understand the reasons of the theory. The final stage is the Ri, it is the level at which the student is a master and can adapt the theory to his needs. This method helps to get the Agile method but also to improve these methods (Novack, 2016).

To judge the implementation of a project, in our subject it is the Agile method adoption, determining KPIs is central for the operational performance. It is a support to the decision system (Issar and Navar, 2016). The KPIs are based on the Balance Scorecard, a performance management tool. There are 3 dimensions analysed in this strategic framework: finance, customer, processes and innovation (Samsonowa, 2012).

---

## Chapter 2 - Research Motivation

---

In the following section, we will discuss the scope of the research and what we try to find in this research.

### 1. Purpose of the Research

To respond to this type of research questions, the exploratory study is the best option. This study is the best way to discover a new view about a special thematic. The aim of the thesis is clearly to have a better understanding of the implementation of Agile methods, which is why the exploratory research has been chosen (Saunders, Lewis & Thornhill, 2016). This implementation is adopted by a large number of companies: 74% companies in Belgium use Agile to some extent (KPMG, 2017). The reasons might be hard to define; different business goals may be different like strategic, operational or cultural (Root, 2018). But for the same reason the quality of the research will depend on the participants and the structure of the information. The main types of goals will be more precise along the interview process (Saunders et al., 2016). The purpose of this study is not to generalize, but to give a clear view on the reality (De Ketele & Roegiers, 2009).

### 2. Research questions

In the literature, the goals of Agile adoption are too often not mentioned. The success and failure rate of this change is similar (Kamp, 2014). A lot of studies speak about the advantages and disadvantages of the Agile implementation, but the reason of the bad results should be discovered. The reasons why businesses make this decision is not analysed here. However, it is a major key of a change success. This is a track to explore. The determining goals may play a role in this undertaking and its success. It is why it is important to focus on that subject.

Because of this problem, we can state the following research questions:

1. What role does goal/intention play in Agile model implementation in software development?

The first research question will focus on the motivations of the companies. This will try to make a list of potential goals that determine to do the Agile adoption. It would be great if we could find the implications of this goal on the company. For some of them we will make links. For others, a deeper and more focused research will be needed.

2. Has this role been taken into consideration? Why has it and why not?

We will then analyse what the organisation put in place to make these goals happen but also if there is an assessment of the realisation of these in this second research question. This will then analyse by two approaches. The first one will focus on the change management, the way of Agile is implemented. The purpose is to find out what are the resources put in place for its success but also to reach the goals. And then the second one will work on the evaluation. The situation of goals after the implementation and finding if it is a success or not.

---

## Chapter 3 - Research Methodology

---

### 1. Research Design

The type of study recommends interviews that are semi-directive or not directive. To give some guidance during the interview, the semi-directive interview is better; the themes are easily understandable. This study wants to help the businesses in their implementation, but the scope of the study still has to be respected (Quivy & Van Campenhoudt, 2006; Lejeune, 2010).

The Face-to-face interview was chosen to have contact with the interviewee. There were three group interviews. This element will not impact the research because of the close relationship between colleagues and assertive collaboration. The quality could be higher in a one-to-one interview, but it balanced with the fact that their brainstorming gives more deepness. These group interviews were not impacted by the impression of obligation to do the interview in the group, because they proposed to do it at their facility (Saunders et al., 2016).

Due to some geographic problems, only one was made by phone. But this call turned on a non-directive interview because of its criticized meaning about Agile.

### 2. Selection of interviewees

#### ✓ Functions

The profile chosen for the interview were people who could have a direct contact with management or have business tasks and still have a great role to play in the IT department. The profile of the interviewees was another reason for the interview choice (Saunders et al., 2016). Two categories were selected, the first one is witness privileged. This category of people has a clear view of the problem due to their function. For example: externals of the enterprise who help implement Agile methods.

The second category is the direct public; all people that are directly concerned by the subject (Quivy & Van Campenhoudt, 2006). Both categories analysed, different functions came out. They will be explained one by one.

- Product Owner

It is the chief of a project in an Agile IT environment. He has a lot of responsibilities such as work dispatching, task prioritising, relation handling with customers but also communication between the team members. (Talenlyft, 2018; Elae, 2018)

He manages budget and time. His position is highly related to management. He oversees the technical aspect by keeping direct contact with customers and the business part. As project leader, he has management knowledge and has direct contact with the management.

- Scrum Master

It is a specific job when using one method, the Scrum method. He has a role of a facilitator in the development teams. According to Nedjma, the Scrum Master controls the application of Scrum method which includes the rules, concepts and practices. He is in close collaboration with the Product Owner, the team and the organisation. He is not a boss, he teaches, gives advice, oils the wheels of projects, and manages the problems. He plays the role of an asset or a guide in projects but gives no instructions or requirements. He gives an external, and maybe a more subjective, approach in projects. To sum up, he oversees the good working process (Lothon, 2017; Mountain Goat Software, 2018; Saidani, 2018).

- Business Analyst

A Business Analyst is a mediator between all stakeholders and determines the needs of them to build requirements, planning and the cost/benefit analysis. He has the role to adapt these requirements in the direction of the organisation's benefit. He looks for a solution for a business taking all aspects into account including people, existing processes, rules and costs. For a better understanding, he might be compared to a doctor. He analyses the patient (the business), makes a diagnosis and gives a solution (Bisk, 2018; Brandenburg, 2012; Modern Analyst, 2018; The Meta Business Analyst, 2018).

The Business Analyst who was interviewed worked in an SMB, in direct relation to the boss.

- Project Manager

The Project Manager is the end-to-end leader; he oversees the scope, planning, resources, budget, people and risks. He should keep an overall view of the completion and success of the project and by establishing the right objectives which must be clearly understandable and reachable. To attain these objectives team spirit and communication are very important. Here the Project Manager plays the mediator between all the team members. In that way, constant feedback and acknowledgement is a way to establish a good personal and team feeling but

also for change and innovation. His project is in his hands, which means responsibility implying that he should forecast the predictable and prepare the unforeseen. (Department of Finance, n.d.; Project Management Institute, 2018; Raucent & Semal, 2016; Techopedia, 2018).

- Program Manager

Program Manager is a Project Manager with experience which oversees different projects related items/subjects (Project Management Institute, 2018). He handles different parts of the business to guarantee the success of a product including different projects. He is more leading the project on a strategic level of the company. It implies a more long-term overview (Cohen, 2017; Hanson, 2018; Investopedia, n.d.).

- Co-founders of SMB and a CEO

They make the decision to apply Agile management in their company considering their budget, strategy and clients.

- Agile coaches

An Agile coach is a consultant coming outside the coming. When the people received their training, he guides the people adopt the new knowledge during their Agile change. He is there to help the people integrate the method and see all the impacts of the method on their environment.

- ✓ Sectors

The purpose was to reach a large panel of sectors to have a global view of the implementation of Agile methods in all IT departments. The study presented multilateral characteristics taking in consideration the implementation of Agile methods in different sectors and companies. The information, in that case, gives a view on different sectors, on different companies to understand the different needs or goals that an Agile implementer can have (De Ketele & Roegiers, 2009). To sum up different sectors:

- IT providers
- Telecommunication
- Banks
- Insurance companies
- Educational organisation

- Coaching organisation

The experience of the interviewees is in general between professional and senior in Agile management. Only one of them is a junior Agile coach, but he already has worked in Agile and now begins giving Agile instructions and implementation in a company.

### 3. Collection of qualitative data

#### ✓ Elaboration of the sample

As mentioned above, the main purpose was to have a related business and IT profile. The focus is on the implementation of Agile methods. The fact that the people were present at the implementation was the crucial element.

According to Saunder et al., the number of interviews for a semi-structured interview is between 5 and 25 (2016). And for this reason, the number of the interviews is fifteen with nineteen participants. It is an appropriate number because the information became repetitive and did not bring much new information in the scope of the thesis. We can see that their Agile implementation is relatively recent. The evolution of the implementation of the Agile model in the future may be interesting.

To find such profile, networking was major. Three ways were utilized. The first one was contacting every company with a minimum of IT. This approach gave me the opportunity to reach SMB but no big company answered. The reason would be the centralized way of contacting. The second way was the word-of-mouth. Thanks to my contacts, for example Cédric Folchi and above all Marianne Vanwymersch, the targeted people were easier to detect and to make an appointment with them. The last channel, in the same idea of networking, was the use of LinkedIn. Three SMB were already planned, this is why the research focus is on large firms filtered from the predefined function mentioned above.

The first contacts were made in their mother tongue, French or sometimes Dutch. For the Flemish, they were notified that the interview would be in French for a better comparison of the interview transcription. Their level of French was very good, so there was no problem for their approval or for the content of the interview. At the first contact, there was a brief description of myself, the purpose of my contact and the question if they would like to participate to my research. The exchanges were made by e-mail or by LinkedIn. It was the best way to fix an appointment.

The duration of the interview was fixed to one hour. Obviously, there were people who had a lot to tell, and some people went past the given time. The time varied from 35 minutes to 1 hour 35 minutes.

#### ✓ Conception of the interview guideline

Because of the type of study, which is to say an explanatory study, a semi-structured is one of the best methods to find causal relations (Saunders et al., 2016). This is why the guideline is composed of one presentation, a body of five themes and a conclusion (see Appendix 1). I based the structure of my interview on the structure of Petersen and Wohlin (2009).

At first, I tested the type of questions I would ask, with the collaboration of two people: Cédric Folchi, former Scrum Master, and Marianne Vanwymersch, a former Team Leader. These two people have a background in Agile methods in companies; one in a more technical way, the other in a business way. This permits to refine the inquiry and evaluate the degree of relevance of each question.

The interview begins with a presentation of myself, the framework of the research, the proceedings of the categories, an explanation about the confidentiality that will be discussed/talked about at the end of the interview and permission to record. This was the first part; it consists of informing the participant and let him prepare himself for the one-hour discussion. The fact that they are cognizant of the confidentiality matter at the end, gives them the possibility to say everything that they have in mind without thinking about confidentiality.

After the presentation of the context of the research, it was important to know more about the participant, his organisation and his function in his organisation. This preface gave the opportunity to begin the discussion in a relaxed manner and to start with easier questions. In general, a mix of open questions and probing questions were asked to have a lot of definitions and descriptions placed in their context. The probing questions are open ones with the particularity of focusing on specific subjects. The most important question words used are “how” and “why”. Hence, I began the body of the interviews with the history of the methods used for a process in the company to give some contexts of the transformation decision. This context will help to fix the main goals of the company later. After that the core subject began; exploring the reasons and the motivations of their implementation. It will answer the first part of the research question that speak about the role of the goals in this adoption. After that, it go further in the change and try to ask how Agile methods are/were implemented and the evaluation of the implementation itself. The first element will speak about the resources used for the reaching of goals and Agile success. The second element will and the potential expectations of the improvements in projects successes. These two elements answer the second part of the research question that was defined which deals with the consideration of the goals.

To close the interview, questions about their opinion were asked. It was important to ask these sorts of questions at the end of the participant to feel comfortable and not hesitate in taking his time to think about their advice on Agile methods but also on Agile manifesto (which was not really expressed before in the guideline). These questions have for purpose to know the reasons of the consideration of goals. We concluded the interview by asking if they have some remarks or questions and then thank the interviewee for their time and collaboration. After that, I asked the confidentiality part, to know if we could cite their name, the name of the company and the other companies mentioned when applicable.

✓ Realisation of the interviews

Concerning practical details, the interviews were done in French to ease communication and work. They were all done in Belgium, except for the phone call, the interviewee was in France. The place of the interview was mostly at their work, in a meeting room. The other interviews were at their house or mine. The chosen places were quiet places, as recommended by several research guides for a better reasoning of the subject and to feel comfortable in their responses (Saunders et al., 2006; Quivy & Van Campenhoudt, 2006).

To get the best result out of the exploratory interview, my intervention should be as minimalistic as possible. The participants receive some subjects and speak freely about it. The more interruptions there will be, the less the participant will talk, and therefore their answers will be brief. There will be less information and for the exploratory study that will not work. The conversation remains as free as possible (Quivy & Van Campenhoudt, 2006).

A second characteristic of the exploratory interview is that the interventions should remain the most open as possible. The lesser the interviewee is guided, the more he will give his experience and thoughts (Quivy & Van Campenhoudt, 2006).

This type of interview can lead to an unstructured interview depending on the participant's thoughts. Some questions will be answered differently depending on the state of the mind of the interviewee. Indeed, the face-to-face interview gives no time to think and analyse what to say. His reflections will also be interpreted. The interview will be unique because of personality, context and trust on that moment. After every interview, I will learn a bit more sensitivity and intuition in the conduction of the interview (Quivy & Van Campenhoudt, 2006).

All interviews were recorded and then transcribed for a better analysis (Quivy& Van Campenhoudt, 2006). Some of the interviewees asked for confidentiality because of a criticised view that may imply their employers or because of strategic reasons.

Two interviews move the scope away. The first one gave a critical view on Agile and that was the core of the interview. The second one gave a more steered conversation toward the following step after an Agile development implementation. This insight was the release management, the department in very close collaboration of the development one.

#### **4. Treatment and Analysis of Data**

The aim of the thesis is to understand better the managerial aspects of the Agile methods' implementation in companies. The companies don't have a specific Agile method guide to follow according to their environment. The results of the study will try to give a general manner that could give some clues and awareness about implementing the Agile methods. To do so, the analysis will be based on what is already done and will give some information about the specific context of this study. More interestingly the relevance of the management and IT department will be highlighted. Hence, it will not be some research for a better world, it is an applied research to improve the existing environment.

To this end, the exploratory approach was chosen because of little information about that subject specifically. Indeed, we know the advantages of this implementation. We know theoretically everything about how the methods are composed, are carried out. But the mysterious part is the manner how it is implemented in companies. There is no guidebook to find the way to become an Agile company. Hence, We used an inductive approach to answer the questions about motivations, manner of implementation and evaluation in the real business life. It is by the data collecting that we will find some elements of response. This data-driven study does not have the pretension to give the silver bullet (Saunders et al., 2016).

All interviews were transcribed. Some themes are emphasized in the interviews:

- Before
- Method
- Outbreak
- Goals
- Change management
- Assessment

The results and discussion part are based on these themes. With these teams, we could gather the ideas and analyse them.

The exploratory study's aim is not to confirm statements but rather to suggest trails and give a greater overview. There is a clear problem and context is important. The analysis of an exploratory study is a heuristic, trying to have given trails based on our own interpretation in the domain of working. There is no real methodology to analyse this data. The best way to deal with the interviews is to listen and listen again to the recordings, make some notes to highlight contradictions and similarities. It is through the information that hypotheses can come out. We will compare the different interviews on their business work. It could lead to keys to success for a future research. It is a first step for deeper research. (Quivy & Van Campenhoudt, 2006; De Ketele & Roegiers, 2009).



---

## Chapter 4 - Results

---

This part will be about the analysis of the interviews. To this purpose, their transcription was made, and recurrent themes quickly appeared, which makes us believe that a thematic analysis might be the best option. This method consists in giving a colour code for each theme which enables to find convergences and divergences and have a right inter-interview analysis (Blanchet & Gotman, 2015). A recapitulation of the results of the interviews (Appendix 2 to 16) can be found below.

*Table 1 : Summary of the interviews' results*

	<b>Before</b>	<b>Method</b>	<b>Outbreak</b>	<b>Goals</b>	<b>Change management</b>	<b>Assessment</b>
<b>B12 Consulting SMB</b>	/	Combination of Scrum and Kanban	Top-down	Strategic  In the DNA of the SMB	/	Feed-back  Customer satisfaction survey
<b>Bank</b>	Prince 2	Combination of Scrum, Kanban, LeSS and Spotify	Top-down	Reduce time-to-market  Reduce cost  Improve quality  Human  And 4 others.	Training  Coaches  Incubator  Several versions of change	Surveys  Start goals no more mentioned
<b>Cenotia SMB</b>	/	Kanban and Scrum	Intuitive	Could not work another way  Prioritising  Reactive during development and commercialisation	/	No formal assessment  Feedback

<b>Energy company</b>	Nothing	Portfolio of Scrum, Kanban and XP practices	Bottom-up	Reduce Time-to-market Delivering quality Productivity Visibility	Trainings Coach Method-metrics	Feed-back Survey compared to Version One report
<b>P&amp;V</b>	Waterfall	Hybrid Agile Combination of Scrum and Kanban	Middle-Top Middle-down	Cohesion IT and Business Quality of delivery Innovation Visibility Human	Trainings Coach	Evaluation of KPIs
<b>Pyxis Technologies Belgium</b> <b>SMB giving their experience with customers</b>	Combination of Waterfall and made-up	Spotify, Scrum	Bottom-up Top-down	Profitability Fast delivery Innovation Strategic (Rivalry) Human Cohesion of Business and IT Hype	Training Coach	Assess: Profitability Productivity Number of functionalities delivered

<b>SMB delivering applications</b>	/	Combination of Scrum and Kanban	Top-down, then bottom-up	Customer satisfaction Human	Improvements on their own	Feed-back
<b>Telecommunication company</b>	Waterfall	Hybrid Agile Scrum and Kanban	Top-down	Not mentioned	Training Coach	During Agile method only
<b>UCL</b>	Unstructured Agile	Scrum	Top-down	Frequency of delivery Give responsibilities Innovate the method of working Having visibility Structure	Training Coach	During Agile method only

To make this analysis, the different topics follow a predefined order. We will at first deal with what was used as project management methodology before implementing Agile. This chapter will explain the Waterfall or other methods brought about disadvantages. We will then discuss the outbreak of this new technology. Different means of outbreak in the company are explored.

After that, we will focus on the motivations of the Agile implementation. Firstly, the strategic analysis of the company is carried out. This chapter contains two perspectives: the environment and the internal diagnose. After that, we will speak about goals in general, it will be a mix of operational, cultural and human reasons.

In the implementation of Agile, a change management was needed. Such a transformation implies a certain structure to follow, a plan, training and people management. A synthesis of the way of change management is conducted by companies.

Are the goals evaluated at a moment of the process? That is the following question to which we will answer based on companies' cases. There will be different assessments: assessment of Agile itself, and then comparison of the predetermined goals, project success and unexpected benefits.

The last chapter is the presence of Agile mind-set. Following Agile principles, we can always improve. We will get some examples of improvements out of the interviews.

## 1. Stand Point Before the Implementation

This chapter has the purpose to explain the precedent project management methodologies and the reasons of Project Management Methodology changes. Agile appears like a solution of these problems. There are different manners of influence to adopt this technology in the firms.

The recent SMBs started directly with APM. For these companies, there will be no change management. Agile was a built-in part of their strategy.

As far as the other companies are concerned, they followed the Waterfall framework with methods such as Prince 2 or PMBOK before implementing Agile. There are companies for which it was impossible to say whether or not they followed any project management methodology (PMM). It is a mix of Waterfall and undetermined method. There is one large company which had no PMM because of structural changes: they came from one big group and then had to begin from scratch apart from them.

### ✓ Waterfall Framework Disadvantages

There were substantial disadvantages in using Waterfall. The projects were heavy, not profitable, not in adequacy with the demand, not fast enough on the market. Those are the key terms which were recurrent in the large firms' interviews.

The reason is that they had to make an analysis of needs and a bill of specifications before starting. This step could already take six months, and nothing was produced. Once the contract was signed, nothing could change from both sides, the customer may not say that it was not what he asked for, and it was too late to say that there was a mistake in the analysis. And this situation lasts for a period up to 3 years sometimes. We can discover a possible mistake only at the delivery, thus at the end of the process. Moreover, the analyst may be gone, and the knowledge of its analysis may be gone with him.

Another point is that in a period of 3 years, a lot of things have changed in this VUCA world (Reeder, 2017). Hence, at the delivery, the product is not what the user expects anymore. It will not be state-of-the-art. The product will already be outdated before users use it.

### ✓ Outbreak of Agile

When the interview went deeper into the reason of this choice of Agile undertaken and more in particular in the method choices, different perspectives of Agile outbreak came out.

The first outbreak mentioned is the utilisation by others and a trend in the literature. The environment of companies was a big influencer to use Agile and some methods. Other influencers were: the *rivalry*, the *renown*, and the fact that Agile *has proven its worth* in the past. These reasons came out before all for the justification of Scrum, Spotify and even Agile use. This outbreak may be related to the competitors and new technologies in the 5 forces of Porter (Guilmot, 2017; Coeurderoy & Vas, 2016). These reasons are in general related from top to down. The opposite is also possible, the workers had experience in another company or knew relatives that use this type of technology and brings the idea in the company.

✓ Top-down influence

In some companies a *top-down process* was made. The leaders consider the Agile implementation a *good step for improvement* and developed it through the company. They do it based on specific pre-analysis of the IT sector in the organisation and the impacts that it would have to the organisation. The time is optimised for a preparation of change.

That is the typical hierarchical way of giving orders. That makes the change difficult for some people could resist only because it is an obligation of the superiors. Communication, explanation of goals and a good change management would be needed (Guilmot, 2017).

✓ Bottom-up influence

Another company mentioned that *new developers*, after a while working there, sprout the idea of implementing Agile project management. They *had the experience* of it in another company, they liked it and wanted to pursue working in this way. They came up with the idea *to the directors* and *showed its benefits*. The seduced bosses accepted this undertaking in their company.

The disadvantage is that the process will take a long time before the implementation. The new workers will suggest this only after a while of experience there. After that, he would have to bring this idea but as a developer, the power is limited. He has to convince a lot of people before the implementation. And this process could take a long time.

✓ IT management influence

The outbreak may appear in *the IT leaders*, the problem is that this idea must then *expand in both ways*. The superiors have to be convinced based on a good analysis. Throughout the idea must be conveyed up to the bottom.

## 2. Goals

In this section, we will see which factors influence the different companies to go for the Agile project management. Firstly, we will discover the strategic goals based on the SWOT analysis. The SWOT analysis contains 2 parts: an environmental diagnosis and an internal diagnosis (Coeurderoy & Vas, 2016). After that, we will approach operational goals that are nearly linked to the strategic goals. And finally, we will cover the organisational goals; the link between strategy and operational will be explained.

### A. Strategic goals

#### ✓ Environment – Opportunities and treats

The SMBs found the opportunity to differentiate from the existing services by offering a quick delivery. One of them said “*It is part of our DNA, we could name our company B12 Agile Consulting*” (M.H., managing partner). The providers on the market promised so much but the customers saw nothing of it. Customers have to wait too long and the project has to be fast, and the requirement list is almost non-existent. Cenotia as well (another SMB) said that it *proclaimed from itself* also, it was *intuitive* and in fact, they could *not work another way* with the projects they have (C.T., co-founder and business analyst).

Regarding the large firms, they realised that there were threats about rivalry. Because the *competitors delivered before them*, they concluded that they had to *deliver quicker*. Rivalry delivers a long time in advance any product before they are in the production phase. It is a world where *you have to innovate otherwise you are out*.

They also found an opportunity in the outbreak of new technologies. This *innovation* called here Agile project management. The technology has been increasingly used in the world and it *seems to have great advantages* in the process.

#### ✓ Internal organisation – Strengths and weaknesses

Large companies highlight a major weakness in the company, it is the *gap between the business and IT teams*. The business explained the requirements of the project and then they did not see the IT teams before a year or sometimes even more. This made a lot of frictions

between these two parts. The communication between them is then strained. *Trust is gone*. They do not speak to each other, which implies that some important updates are not transferred.

*At delivery, misunderstandings appeared*. The product is not like it has to for the business. The translation of what business wanted, and the IT get was not good. The analyst may be gone with his knowledge. Moreover, time passed, and *the world changed*. In this VUCA world (Volatile, Uncertain, Complex and Ambiguous world), the requirements of yesterday are not the same as today (Reeder, 2017). Hence, a project of a year or longer without any consideration of the new state of the world makes the product inadequate for users. Agile gives a possibility of change and adaptation to the evolving world.

*Customer satisfaction* is a strength not to undervalue in delivering services. Otherwise the customer takes a hike to a competitor. In big companies, all IT providing is not in customer purposes but the profitability and performance are then taken into consideration.

#### *B. Operational goals*

There was a *need of prioritisation*, the requests have to be planned in a way that at each delivery, the product runs. The running product has an influence only if *it matters for the customer*. Thus, the important specifications have to be developed at first.

*Communication* in the overall is another goal, it is suitable for business but also in the team. It implies a process of *mutual assistance* in the team, and an *understanding of the function of every team member*. A lot of problems are solved quickly through daily meetings, for example. And the fact that they know what others do, *improve their knowledge* of the process and the *impact of their work*.

*Structure* was the most-used word. Companies need structure in the overall that brings *vision for all stakeholders*. Business and customers can know the stand point of the project through Agile, but the workers also know where they are in the planning. But the developers like also the structure, it brings a *view* of what they do, have done and have to do. They can also develop the core needs at the beginning instead of developing little things that are not vital.

To fulfil customer expectations, the delivery has to *respond to demand* and the *time-to-market* has to be reduced. The customers have, more often, results about the progress. And through Agile process, with the feedback cycles, these *expectations* are often assessed and/or

readjusted. It is also a way to consider the environmental change and to stay updated. The credibility of IT is improved by means of the visibility and transparency.

The “*rework*” of change in the large structure is very costly. It is why the cohesion of business is very important to require. The feedback and short cycle permit to achieve this and *reduce* the rework costs. There is a possibility of anticipating and putting things right.

### *C. Organisational goals*

Sometimes the *well-being of workers* is taken into consideration but unfortunately, it is not often the case. Thanks to the method, it is *less stressful* than the one before. But there is also a *discovery of skills in the people*. They are more *motivated* and take *initiatives*. In general, it is an additional benefit during the process, this goal is not the motivation of launching technology.

### *D. Strategic and operational goals*

Agile is the operational way to apply the strategy. It is also the way to realise the strategy. For instance, prioritisation is based on strategy, it implements quicker the strategy of the company.

Hence, this implementation is considered as *very strategic*. The goals are customer or business oriented. The first need is a *shorter time-to-market*. The second is *profitability* and the last one is the *improvement in quality* (by adapting to change and meeting customer requirements). That were the essential goals of management.

## **3. Change management**

This section deals with the transformation of the organisation to Agile. This does not concern SMBs that since the beginning implemented Agile. It consists in having a view of how the organisations implement Agile changes. And at the end, we will see that Agile process facilitates change.

### ✓ Planning

For one company, after months of elaboration, they applied the IT transformation. It first consists in a *system of incubators*. Small teams were sent to the incubator with new products to develop, and they had to do it with Agile method. They had training and coaches. After a team was ready to use the method on its own, another team took its place. This was

the VI, everyone kept their functions. For the second version, the middle management needed to find another function in the new hierarchy.

In other companies, they first put in place a *Hybrid Agile*. Because there are too many people and processes involved doing it at once. They have to work on *automated tests, reduce the dependence and teach the Agile mind-set* step by step. For P&V, Hybrid Agile was the first step that has now ended and is *going further to a full Agile*.

An alternative way was to give the possibility to teams to become Agile. Team by team, they had *lessons, coaching and adopted the method*. After that, they could choose what they wanted to do. New practices appeared, and it is *up to the team* to choose the *best practices* according to the product, and the team itself.

Then, some companies decided to implement Agile at *once*. They got a *book*, sometimes *training and they apply it the following day*.

#### ✓ Training

Training is generally given at the beginning of the change process. It is not always the case, some firms think *"It will be alright"* but then, after a while, *verbal ping-pong and several arguments* are triggered by the *training need*. Thus, the lack of training makes the actors suffer. It is why training is given at the end.

To go further on this point, following the same training together enables the same terminology and everyone has the same basis. When there is dissension, it has influence on the debate about the way of working: *"Are we Agile or not? If we are, we do it that way, it is the way to follow"*. It is better to *structure* and to have *background* to put everyone on an equal footing.

#### ✓ Difficulties

This section deals with the difficulties encountered in the change process. The investigators of the change management underestimated it because a lot more resistance or problems appeared than expected.

*Resistance* comes out of people. Some resistance appeared immediately after they heard the *word "change"*. It seems that young people have fewer problems with that. After that, you have the impression they want to monitor you all the time with a 100% of their

work. They have to determine the time for a task and they are scared of the consequences if they do not get the task done on the due date.

One thing is the *work obligations and work management* during a change. There are always obligations that have priority on the change. For example, there were legal obligations of deadlines; the project has to be done at a precise date. The change management process needed some free time slot to follow courses and be coached. Decisions had to be made, legal obligations prime first *so it was the change management that last longer*.

✓ People involved

A second difficulty is that change has a large scope. Agile changes the *culture of the organisation* it is why all members of the organisation will be evolved at one time or another. The shortage of extent implies discordances. All domains will be evolved by the time. It is necessary to *let Agile being Agile*.

The principal actor that must be involved here is the business manager. The business needs to follow this training because he does not have the keys in hand to instruct his subalterns to do something. But they *don't know the way of working and they misjudge the work*. For instance, in the iron triangle, also known as project management triangle, the resources and time are fixed, and the scope is mobile (EasyProject, 2018). *The directors* see that we can put *more content in it*. But the *contrary is also true* to respect the two other constraints. Since they do not know Scrum and they did not follow training, they *give orders to stand in the way of Scrum*. Another thing is that business has to know that they could not expect any improvement during the transformation. Because they think they will see the results directly, they give only bad feedback.

Middle management is a big field to handle. Indeed, it is *not considered enough* in the change. The presentation of the change is given, but there is no explanation of the future of their role. They have the fear of the change, and above that the *fear of losing their jobs*. *Agile flattens the hierarchical pyramid* and the function of middle management is suppressed. Big companies try to hold these functions in vain in *creating new functions*. But the problem is then that it is against the Agile mind-set. People are not controlled anymore, and these people were instructed to give orders to their people. This way of leading does not represent the Agile mind-set correctly. They have to behave like *servants for the people*. In such a way that they became themselves the power. They have to lose the control they had. It is really difficult

to do because they were trained that way. It is all a process to change this mind-set that is not taken into consideration at the moment in large firms.

I had the possibility to meet a release manager who is implementing an Agile way of working. There is *no real method to do it*, it is an adaptation of working based on the development flow. The automation testing is very important, they have to automate the testing. Based on the Scrum cycles of the development process, they planned gates where the projects will come through for a continuous check. In that way, the deliverables will always be checked during the process and not only at the end. At the beginning of a project, it will first pass by the release management that will verify different criteria like testing, timing, quality, scope and the implementation plan. If all those criteria are fine, it passes in acceptance. After that there is a follow-up during development. If it is fine, then it goes to production. For the production, there will be a clear implementation plan to be prepared, in such a way that every release worker knows his tasks.

After that, there is a monitoring and a report of the impediments encountered. And after 5 to 6 weeks of production, there is the closure. At the closure the project manager comes with lessons learned and closes the budget.

- ✓ Agile helps change management at the customers' organisation

The fact that *users get a continuous delivery makes them get used to it*. There is *less resistance* or at least *it is easier to handle*. In a project management, there is also change management to take into account. The utilisation of such a method implies a change but an easier change management.

#### 4. Assessment

In general, assessment is *done informally, by feedback*. As well in SMBs as in other companies. A lot of companies do not evaluate APM because there is already assessment in the method itself. They give priority to the work before the assessment.

Some companies do surveys to see *where people are in the implementation* but also if they are in the *Agile mind-set* or not. For the assessment, it is not easy to see which benefit comes out of the methodology because other factors might interfere.

For SMBs using tools, they find that the tools respond directly to their goals. Regarding large firms, the goals are maybe clearly defined at the beginning but when the assessment comes, it becomes more obscure.

P&V mentioned the KPI that they want to achieve, it took more time than expected but they finally achieved it. The KPI cited were, for example, *quality, delivery of business value and business satisfaction*. The project success was also improved through Agile. The project was almost never on time and on budget. Moreover, the satisfaction of business was not met with Waterfall. With Agile it is the case. These three elements determined the success of a project. The improvements of the methods are reported through the feedback during the method process.

An energy company and a big bank in Belgium do surveys to see the *Agile maturity* in the company. The energy company *makes reports of it and compared* to the precedent report and the rest of the world. The results are significant, the Agile implementation is on a good path for a beginner in agility.

The customers will *always be satisfied with such method compared to the Waterfall* model because they do not have to wait for a long time; they frequently have delivery. However, they also need to be aware of what Agile is.

Another SMB mentioned that due to the small structure of the company they did not formally evaluate the PM itself. Although, they consider the culture important in the use of the Agile methods. It is why two criteria are taken into account, namely *customer satisfaction*, and *employees' well-being*. They measured the customer's satisfaction and it was very positive.

Companies see the need for *constant improvement* of the method. The transformation will end someday, yet the improvement will never stop. It is *important to take time to wonder*: “*What could we improve in the way we work*”. The method has to be constantly reviewed. It is a point of the Agility definition itself. Such as the UCL, it would be great to encourage people to go to seminars to know even more about Agile and give the chance to improve.

## 5. Agile mind-set

The *Agile mind-set is important* for every interviewee, even for the one who do not know the Agile Manifesto. Adopting Agile is conducted in an Agile mind-set too. The Agile Manifesto is a synthesis of the way of conducting. It is cultural and philosophic. They do not tell what to do or what not to do. It is said “[Value] over [Other value]”, it means that the value on the left is more important than the value on the right (Agile Alliance, 2018). It founds a managerial school of thought.

- ✓ Agile Project Management and mind-set

The Agile Methods *can constrain the agility of a company*. Certifications are *heresy*, the only way to be Agile is the experience on the field. Certification constrained people to use one method, it pushed people to become dogmatic. Agile is an empirical study and needs to be based on mistakes and learn from them. It has to enable them to think and to learn.

- ✓ Importance of the mind-set?

The mind-set is very important. One *can apply* a method and *remember each step* that to do, but *to improve the method you have to acquire it*. Following step by step *is restrictive*, and thus it is not Agile. That makes the change management difficult to implement. There are people whose Agility is intuitive. But for others, new norms have to be learned and a lot of communication and coaching is needed.

- ✓ Mind-set is a long process

The mind-set of the people of some people don't change or the people do not want to change. There are people who kept the older value system. That could be due the fact that they keep secrets to be more indispensable or powerful. But some people want to work on their own, with no team working. To correct this, team building and communication are important to implement. All the companies they know that *the mind-set is the most difficult to get*. Regarding large structures, they state that the Agile mind-set is not truly exist in all the company, there is still a long way to go.

- ✓ Complementarity of mind-set and method

The Agile Manifesto helps to understand the construct of the method. It is a guide that gives pursued principles. When one gets the mind-set, one knows what brings every aspect of the method, you may explore your own Agile method in selecting the practices that you need.

The form of the method helps to construct in real what you think. The fact of continuous ameliorations with feedback could be done by another.

- ✓ Difference between state an Agile company and being Agile

There are companies that stated they were Agile but in truth, they only followed its methods, without having the value system. A lot of companies say that some state they are Agile but stay in a Waterfall logic. An example given is the *submission of the proposal*. With no scope, it is not possible to make a budget. And with no budget you cannot make a

proposal. An interviewee was very criticised on this problem. Companies can say they are Agile for the marketing but in fact might besmirched a lot of Agile principles.

*1. People have to be motivated*

This assumption is not applied in every company. Agile is used in HR to pay the employees less. For example, the developers have to work on Sundays to launch tests but HR refuses to pay them on Sundays. Thus, the workers do not want to work on Sundays.

*2. A team has to remain stable*

It is not the case, people are falling sick, women may be pregnant, and key workers will find better workplace elsewhere. All these factors do not warranty a continuous work, and the entire workflow is then in peril. Someone has a task but needs the task of one another to complete his, that is a problem.

*3. A systematic delivery*

There is no warranty of problem, in reality, when working with Agile there are always problems, it is why the delivery was always a week late. The unavailability of people, or their absence created this delay.

A solution is making the people more flexible in the tasks they can accomplish. This will require technical training to deal with specialised people.

*4. Teams are 100% available on one project*

It is not true in reality, developers are put on several projects at the same time. Hence, when there is a problem, the needed developer might on another project and he might not be available.

*5. Velocity*

At the beginning, it would have diminished of 50% and then increased. But in the real life, there were cases where the velocity went under 10% and only then it increased. But it would not be stable because of the sickness, departures and so on. Velocity will remain with peaks.

Agile works like special commandos works, it is 100% perfect only in some cases, for companies which have the assets and the resources to implement it.

It can work with innovations or for projects where bugs don't really have a big impact.

To certify an Agile working, we have to do a battery of tests, which leads to less agility because of long periods of tests.

✓ Following a method

Some companies consider that to follow strictly a method is the opposite of the Agile mindset.

In the interviews, there were people who made the point that the implementation of the Agile project management was tricky. In one of them we could hear “We think it will be alright”, they use the terms, they use the names, but it is not why the Product Owner knows what he has to do.

---

## Chapter 5 – Discussions

---

This five chapter will discuss different topics that went out of the interviews and linked with the research question. The discussions are there to give tracks about potential further research that could help a company to begin in Agile implementation.

The first section will discuss the first part of the research question. This research question concerned the role of goals in the Agile adoption in companies. A list of necessary or deduced motivations is constructed, and the bad impacts that may imply are mentioned. This is an attempt based on the interviews conclusion. A quantitative research would confirm or disprove these motivations.

After that, the consideration of Agile goals is still obscure, it is why a quantitative method of Agile method selection and tailoring would be needed as first point in this perspective. Therefore, it is opportune to see the different manners companies can select and tailor Agile methods in the database that we collected.

The third chapter is a hint of guidelines for an Agile change management. This follows the aim to answer the consideration of goals. It is not easy because the resilience is huge under big or little factors. We will talk about the factors received during the interview. But a deep case study will be needed focusing on the change resilience that may occur.

The last topic is the assessment, it is the final consideration of goals. The implementation assessment is not often done in our sample because they content with the assessment includes in the Agile cycles. Some of them did it well through different process. This discussion will allow to find the best method assessment that could make a ranking of Agile. A ranking would help for the companies to help them see where they are in their implementation.

### 1. Research Question 1: Role of the goals

This section will give an answer of the first part which is: *What role does goal/intention play in Agile model implementation in software development?* The motivations and goals organisations that should be considered when implementing Agile will be developed. These motivations and goals are linked to some problems that may arise in the organisation.

The first goal is to be the *first on the market*, or at a *competitive time*. If the need is to develop an innovation, it would suggest that you use Agile. It gives the possibility to offer your *new* products constantly and *improve* them continuously with the feedback of the users.

If we go further in innovation, Agile brings the opportunity to promote entrepreneurship. This goal was not mentioned but has the merit to be mentioned. The fact that there is no boss who is telling what you have to do enables this type of action. *Only guidelines are given* to the team and the *team thinks all together* for the project, which can bring more ideas on the project development. Moreau discovered Agile support entrepreneurship (2016).

Another goal is that Agile is considered as the technology of the moment in Agile Project Management. A lot of companies try to put it in place for that reason. It would give benefits and it is a form of *marketing label*. Naturally, the question is to know if it is better to do such as everyone else, be on the trend, not to lose customers or make the point of it. Maybe we could research if Agile could be some criteria for B2B services.

The *cohesion of IT and business* in large firms will be improved because of iterations and feedback. The business will receive quicker results and IT will be updated of changes during the process. It will then meet the business need and change, both parts will be happy.

The *visibility and structure* of the project gives a big advantage of using Agile. There is a clear view of the budget. This transparency is improved on both sides because the business side could easily give the priorities in projects to the IT side. Many companies find that all the stakeholders are more satisfied because they are aware of what is happening. Naturally, this motivation can also be considered negatively. The visibility gives an opportunity to control the developers and to use against them through the analysis of numbers without the reasons. This is not the objective of Agile either.

Due to the fact that the scope is not preestablished, customers must furnish available people that know the product and could give the expectations and the feedback. *Customers* must be an entire player in the project and his availability is necessary for the project success.

Agile is often considered as a method that enables to do better with fewer people, resources. This has to be studied. Following the interviews, on the one hand, the work is more *structured and optimised*. Moreover, the problems will not appear at the end, so it is

manageable. On the other hand, if all the requirements are known at the beginning, it would bring cost savings doing it in Waterfall.

Another aspect is that people will be more involved in the company. They will receive *more responsibilities* and this consideration will *ameliorate their well-being*. This will reduce cost because fewer managers are needed and the people, due to their increase motivation, will work better. It will be the case only under some conditions which are the respect of all the principles.

Structurally, one must be well aware that *the hierarchical pyramid will be flattened*. The middle management would find another role to play in the organisation but in the way that it will evolve it will be difficult for them to conform to Agile. They must accept this type of mind-set, people servant instead of people management.

This structure *may lead to a liberated company*. The responsibilities are shared by everyone and the hierarchy is reduced. People are at the centre of the company with responsibilities but also autonomy and participatory. From another angle, a liberated company is Agile.

Another goal is giving time for this adoption to be a success. Don't underestimate the time of implementation. The *change will take a long time*, well prepared or not. There will always be more problems or resistance than expected because people's set of values is the most difficult to change. Giving time to this change must not be a brake.

A cultural change is needed, and all the company will be impacted. Everyone has to communicate, for communication is vital in this mind-set. And this mind-set may survive if everyone follows the principles. It is why *not only IT has to change*, Business, Marketing, the Release management and HR for example have to follow the trend. Business has to be Agile as well, go at the same speed as world change and consider the way of working as important as the end if we do not want to constrain the IT agility.

There are so many people impacted in the change that the origins of the motivations and where destinations are questioned. Its role will be very different depending on the different flows. Would it better to have a bottom-up or a top-down influence? A track was that management gives the opportunity to develop in an Agile way but let them free to choose the method what is the most suitable for them. The reason is that developers know better which practices would help them to go faster, for example. Naturally, some structures have to be

given for business and management purposes. Top-down is easier to receive the budget needed and the correct analysis of the Agile method needs at a first time. Bottom-up will maybe have, on the one hand, less resistance because it will not be considered as an order or a negative change for them. It can have a snowball effect on motivation. It is then easier to adopt the mind-set. In another hand, it can lead to more resistance considering that people do not know what they are doing. A compromise from the beginning of Agile adoption may lead to a success with a continuous shift of top-down and bottom-up approaches.

## 2. Research Question 2(1): Consideration of goals, way of adopting Agile

This chapter will speak about the Agile methods choices. This make part of the resources that companies put in place that differentiate the most. Indeed, almost all of them had a training and coaches. We could previously see that there is a wide range of methods. The question is now: how have the company selected them? Thus, the Agile methods choices are interesting. And going further in the reflection, the motivations, the reasons of tailoring a method are also interesting.

Such as many of the interviewees, and it is good to mention the methods *must not be considered as rigorous and mandatory to follow*. They are frameworks, they gives a way of working that can easily be adapted following the needs.

As mentioned in the results part, the method selection is above all motivated by the environment. Scrum proved its worth, Spotify is *hype* and Agile is what every company use. There are thus good representations of Agile methods but also bad. For instance, Lean is seen as a method that “likes to supress” and before all people. This method will give bad information. Users of Lean consider it as a “*quick and dirty*” method because one develops only what the customer needs now but it engenders problems when the application has to be reviewed or adapted.

This leads to suggest that methods have to plug their advantages. A lot of methods that we mentioned in the theory were not even cited in the interviews. The companies do not know all the possibilities that Agile gives, and all the portfolio of its methods either.

We could see a form of analysis of which methods exist on the market when we know that companies do method tailoring, and more precisely method engineering. But they all used or have used the *well-known methods* such as Scrum, Kanban, Xp, ScrumBan and scaled methods such as SAFe and Spotify. For the SMBs, they based their choices *on their*

*experience* selecting what works for them and changing the practices that did not suit them. The energy company chose to let a portfolio of practices available and teams can pick what they need following their needs and specificities of the project. But these practices are also picked from the 6 methods mentioned earlier. It would be interesting to know if the decision takers of companies knew there exist other methods, and why they did not choose them. Therefore, a quantitative research will be needed.

The energy company has the particularity of not giving a method but a portfolio of practices. Everyone is formed following a method but then the team chooses depending on its maturity, availability and project requirements. This gives agility in the team and will select in an autonomous way what it needs.

When we speak about Agile, it is maybe the case of the energy company that will be approached. It is giving the keys to everyone and they select what they need as a responsible team that knows it needs. There is also a tool, available for Scrum at the moment, to give a metric of eligibility of the team. It explains if the team is eligible or not and, moreover, it reveals the help they would need depending on their situation (knowledge and experience of the method, technologies, team size...).

The energy company case could maybe go further, have a portfolio even more developed to have a panel of possibilities, the partial adoption. This can always be discussed because there is a need of knowledge of all these practices but also an important integration. And then integration from all teams of Agile practices will be challenging and risky. Where is then the limit of Agile?

### **3. Research question 2(2): Consideration of goals, escort the change**

This section is the beginning of giving a form of Agile implementation guide in the change management. The companies told me that there was a shortage of information of problems and resilience that one encountered in Agile implementation. It follows my research question 2(2) and furthermore to have a business purpose. For sure this discussion is only based on the problems that were pointed out during the interviews. Further research on all the situations that one may encounter when putting it in place would be a good follow-up and would even be asked by the companies.

The first resistance is that there will *always be opponents to change*. That was the first fact pointed out by the literature and the different participants.

After considering this fact, more particularly concerning Agile and Scrum is that the calculation of *velocity* makes the developers feel they are *constantly monitored* without considering that there are reasons behind the numbers. It is then needed to consider a better communication about this velocity and its utility.

Another problem encountered with Scrum is that the team members have to determine the tasks, but it is *not easy to evaluate how long you will work in advance for a task* that you have never done or when you don't know the problems you will meet.

You also have the Middle Management who does not know where its place is in the new organisation but also has to change of mind-set. The mind-set change is for everyone and this is the most difficult change.

In large firms, it is difficult to handle *the dependences* in Agile. The number of dependence is negatively correlated to the level of agility in the company. The dependence between two specialised people are huge and will imply a brake of agility when the required specialised person is absent. People need a new skill management that makes them multitask. Therefore, technical training is needed.

#### **4. Research question 2(3): Consideration, the assessment of the goals**

Section four discussed the poor APM assessment, the final consideration of goals. This conclusion is alleviated because the Agile method already includes an evaluation part. The discussion goes further to discover how we could make an Agile maturity ranking.

A conclusion that I could make is that the APM is not often evaluated. There are companies that collect more information *by a survey* about the stand of maturity in the organisation. The companies can see through feedback from the method that it has some benefits and it is enough for the majority of them. But with a method where feedback and improvements are including in the process, should there be a lot more assessments of the Agile method? Only the Agile maturity is not assessed in the process, the shortage that some companies compensate with surveys.

One of them, sure in any case, does a report with a clarification of the situation and an assessment of Agile Maturity. The Agile Maturity is compared with the Version One report. This report contains four parts: the benefits of Agile, the organisational culture, the extent of Agile in the company and the customer satisfaction (VersionOne, 2018). This explains

correctly how to compare your situation to others to see where you are and how to improve in the future to be more mature.

This made me interrogate about the why. Some interviewees spoke about PDCA, it is the model of construct of Scrum for example. It is Plan, Do, Check and Act. The Check and Act is the evaluation and improvement part are made by Scrum in the retrospective, for example. But the evaluation is not complete enough and following the experience of one interviewee it is a problem of culture. It is typical for Europeans. It is maybe a track to consider. In each case, the Agile based on PDCA framework allows to confirm and recheck the norms used before.

Pyxis Technologies Belgium gave funny illustrations of how people react to evaluation that you can find in appendix 17 (Forss, 2014). They do the feedback during the method process such as it is designed but then they don't take the time to improve because they are too busy. The feedback has then no need. You have to take the time.

The assessment made us think about how we could assess the Agile maturity in companies. Two interviewees brought me to the track of Shu Ha Ri. Shu Ha Ri are martial arts that consist in three steps. The Shu, you begin to learn the theory and you apply it word by word without question. The Ha you get the theory, you begin to understand the reason you do the things. The Ri, you are master, and you may adapt the theory with what you need. This model could work for companies who followed these steps. But there are companies, such as the bank interviewed that made an engineer method directly before the implementation.

Hence, an Agile maturity assessment should be more precise. The assessment should contain different categories such as how is the level of knowledge of the method, to what extent they integrated the methods and how they have modified it. That is based on the Shu Ha Ri map.

But then there are also other perspectives to reach like the level of communication, the level of freedom, automation, and motivation. Moreover, the success is a particle to consider, what progress it brought regarding time-to-market, customer satisfaction and expectations achievement.

All the principles of the Agile Manifesto should be measured in this maturity assessment to know where the participants are in their mind-sets.

To sum up, the good tools that may help are the Shu Ha Ri, the Agile Manifesto and then the Version one report that give a clear view of the organisation situation in comparison with the world.

---

## Conclusion

---

This thesis's framework includes Agile, decision-making process, Agile management and assessment.

For a better understanding of all the concepts, it was needed to understand the Software Project Management. In this chapter, after a context of these terms, two most notorious project management models are presented: Waterfall and Agile.

The second section spoke about Agile. The Agile mind-set was then explained, because Agile is not only a method, it is also a way to conduct. This mind-set opts for a focus on the delivery frequencies, quality, and the human aspect rather than the documentation or the planning. It is a way of being more active on the project. Then, we reviewed all the known Agile methods, whether they are scaled or not.

Agile has advantages such as seeing change as an opportunity, shortening the time to market and enabling good communication of team members and stakeholders. However, its drawbacks cannot be neglected. The lack of requirements and scope might scare and overwhelm the team. Another major limitation is the need for a huge change of the organisation. It implies a restructuration and the development of another corporate culture.

Afterwards, the application of Agile is shown. These methods are either applied as this or are tailored. The three tailoring methods are: contingency factors, method engineering and partial adaptation.

In section 3, after this perspective on Agile, knowledge management had to be discussed. At first, we have seen that the management context (the VUCA world) brings many challenges to companies, which has an impact on their strategy. We next spoke about the strategic, operational and organisational goals that companies can have. Then, we focused on the notion of change management and the element that it implies. We saw that there was resistance to handle but acknowledged strategies to implement change and to deal with the resistance. The evaluation prospect is then the last chapter reached in the theoretical part. We saw that assessment are mostly not efficient in companies and that it should be improved. Some Agile assessments are displayed.

Chapter 2 treats then about the research motivation. This explains the reasons and the framework of the research. The aim of this thesis was to answer the following questions: What role does the goal play in Agile methods adoption? Have they been taken into consideration? Why or why not? Is there a link with the agility position? Furthermore, it describes the objectives that it wants to obtain.

Chapter 3 developed the research method. In order to answer the research questions, we conducted an exploratory study. It did not have the purpose of validating hypothesis but rather discovering what is done in practice to lead on tracks for further research. To fulfil the exploratory research, semi-directive qualitative interviews were the best option. This type of interview enables to let the interviewee speak their mind on the studied topics freely. There were four different topics: the situation before the Agile implementation, the implementation itself with the goals and the change management, the assessment of this change and the interviewee's personal opinion on the matter.

In chapter 4, five aspects were drawn from these interviews. First, they all used Waterfall or a homemade Agile project management before. They were pushed to use Agile because the waterfall model was not efficient anymore in a complex environment.

In section 2, we saw that goals of the implementations were of three types. We could at first see strategic goals linked to the environment: the competitors were faster and used this technology for the established companies. SMBs saw an opportunity to enter the market thanks to this technology. The operational goals were prerequisite to satisfy the business/customer (reduce time-to-market, have a better adequacy with requirements and be able to change). Organisational goals were disregarded at the beginning of the implementation. Motivation, responsibilities and well-being were benefits that appeared during the process only. The employees are then more motivated, work on a participatory way and have responsibilities.

The third section of the results was touched on change management. The change plans of the companies were quite different from a company to another. Some companies did a preparation of the change process (including pre-analysis of goals or KPIs), others gave the instructions book of training a day before the implementation. All companies saw the need of training, mostly at the beginning, and some of them during the change process. Some resistance was encountered like the resistance of change and the resistance against the velocity that gave an impression of constant control on the developers. There was also

concerns from middle management, they saw that their function does not exist anymore and that they have to find another one. Moreover, they will need to change of mind-set because they have to be people-centered. Which is why it is important that this change management spreads the idea of Agile in the business and all its departments: Human Resources, Marketing, and Release management. This will bring less resistance or causes less agility shortage. Finally, agility improves the change management at the customer because of this iterative cycle.

The fourth section of the Results part dealt with Assessment. Assessment is often done through the Agile methods. But a real maturity assessment is rarely made. Moreover, goals were also rarely assessed. There were two alternatives: the survey the use of predetermined KPIs.

The last section dealt with the Agile mind-set. For all the interviewees, the mind-set is very important with or without the Agile Manifesto. The Agile manifesto helps to have a clear understanding of the mind-set. None of the companies could say that the organisation was Agile, there is still a lot of work to do at that point. This was a debated topic whether the companies were Agile or not with the perspective of mind-set.

The fifth chapter is the discussions part where we try to give tracks based on the interviews and literature in function on the research questions. The first discussion was the analysis of the goals' roles including the impacts that they may have. There is still research to be done on this topic, to know how many companies are influenced by the goals discovered.

Another discussion was about the different method applications. The application of this change in companies is a sort consideration of the goals. It is the way how they put in place the methods to satisfy their goals. Both the theoretical application and the tailoring method have pros and cons. Which one to opt for is still quite fuzzy and a deep research has to be made to compare the consequences of each adoption type.

The third discussion spoke about the perfect consideration of change management. Change management are tidy linked to goals. There will be fewer motivations without goals. We will see the types of resilience that occurs that are linked with goals. It is a start but needs a more analysed view of all the problems and difficulties that the company may encounter. Therefore, the analysis would be easier to do if the searcher was a member of the company.

The last discussion was about the assessment of goals and Agile method. There is a significant lack of project management assessment and the Agile maturity. We tried to give trails based on KPI's, PDCA, Shu Ha Ri and the Version One report to make an Agile Maturity ranking up.

There were also other topics that could be discussed. The Agile mind-set transfer still needs to be improved. The link between Agile and liberated companies could be explored. Moreover, Agile is spreading in the corporate world but it could also be applied in school or for a house moving.

The point to highlight through this research is that companies look for more guidance in the Agile implementation. A guidebook based on experience on the parameters to consider would be very appreciated. To provide somewhat of a trail to this problem. Some basic guidance in goals, resistance that one may encounter, and assessments are given. I tried to guide in goals some resistance that a company could encounter. Another topic worth researching further is the assessment. An Agile maturity ranking study would be relevant and appreciated. Some tracks already exist, but there is still a long way to go.

This study was made based on interviews, it implies that there is always a bias. The interviewees and I were influenced by our experience and knowledge. These biases were reduced at the minimum, but one cannot take all factors apart the redaction of the thesis. My interpretation of Agile, Agile mind-set and the key factors of management that may imply could have biased my analysis. These considerations apart, the research was made following well-known researchers' methods.

Agile gives huge possibilities that still need to be discovered. Personally, I was impressed about its scope, I never suspected Agile to be such a wide concept.



---

## Bibliography

---

Abel-Hamid, T., Sengupta, K., and Swett, C. (1999). The impact of goals on software project management: an experimental investigation. *MIS Quartely*, 23 (4). 531-555. Retrieved from <http://hdl.handle.net/10945/43630>

Abrahamsson, P., Warsta, J., Siponen, M. T., and Ronkainen, J., (2003). New Directions on Agile Methods : A Comparative Analysis. *Proceedings of the International Conference on Software Engineering*. May 3-5. Portland, Oregon, USA

Agile Alliance. (2018). *INVEST*. Retrieved from [https://www.Agilealliance.org/glossary/invest/#q=~\(filters~\(postType~\(~'page~'post~'aa\\_book~'aa\\_event\\_session~'aa\\_experience\\_report~'aa\\_glossary~'aa\\_research\\_paper~'aa\\_video\)~tags~\(~'invest\)\)~searchTerm~'~sort~false~sortDirection~'asc~page~1\)](https://www.Agilealliance.org/glossary/invest/#q=~(filters~(postType~(~'page~'post~'aa_book~'aa_event_session~'aa_experience_report~'aa_glossary~'aa_research_paper~'aa_video)~tags~(~'invest))~searchTerm~'~sort~false~sortDirection~'asc~page~1))

Agile Alliance. (2018). Manifesto for Agile Software Development. Retrieved from <https://www.Agilealliance.org/Agile101/the-Agile-manifesto/>

Aljaber, T. (2018). *The iron triangle of planning*. Retrieved from Agile at scale, Atlassian Agile Coach website : <https://www.atlassian.com/Agile/Agile-at-scale/Agile-iron-triangle>

*Analyse qualitative*. Université Catholique de Louvain, Louvain-La-Neuve.

Atkinson, R. (1999). Project management: cost, time and quality, two best guesses and a phenomenon, its time to accept other success criteria. *International Journal of Project Management* 17(6). 337-342.

Aubry, C. (2010). *Scrum* (6th ed.). Paris: Dunod.

Bisk. (2018). *Business Analyst Job Description*. Retrieved from Villanova University, Business analysis website <https://www.villanovau.com/resources/business-analysis/business-analyst-job-description/#.W0d-EdIzbiU>

Blanchet, A.& Gotman, A. (2015). *L'entretien*. Paris: A. Colin.

Bowes, J. (2015). *The definition of ready*. Retrieved from Manifesto, website : <https://manifesto.co.uk/the-definition-of-ready/>

Brandenburg, L. (2012). *What is a Business Analyst?* Retrieved from Bridging The Gap Webs website: <http://www.bridging-the-gap.com/what-is-a-business-analyst-role-anyway/>

Campanelli, A. S., and Parreiras, F. S. (2015). Agile methods tailoring – A systematic literature review. *The Journal of Systems and Software*, 110(2015), 85-100. doi:10.1016/j.jss.2015.08.035

Coeurderoy, R., and Vas, A. *Les fondements de la stratégie d'entreprise* [Online Course]. (2016). Belgique : Université Catholique de Louvain. Retrieved from <https://www.edx.org/course/les-fondements-de-la-strategie-dentreprise>

Cohen, E., (2017). *Program management: definition, roles, responsibilities & resources*. Retrieved from workamajig Website: <https://www.workamajig.com/blog/program-management-guide>

CollabNet. (2018). *The 13<sup>th</sup> Annual State of Agile survey*. Retrieved from <http://stateofAgile.versionone.com/>

De Ketele J.-M., and Roegiers, X. (2009). *Méthodologie du recueil d'informations: Fondements des méthodes d'observation, de questionnaire, d'interview et d'étude de documents*. Belgium, Brussels: 4<sup>th</sup> Ed. De Boeck.

Department of Finance. (n.d). *Roles and responsibilities of the Project Manager*. Retrieved from <https://www.finance-ni.gov.uk/articles/roles-and-responsibilities-project-manager>

Dönmez, D. and Grote, G. (2018). Two sides of the same coin – how Agile software development teams approach uncertainty as threats and opportunities. *Information and Software Technology*, 93. 94-111. doi : 10.1016/j.infsof.2017.08.015

Drury-Grogan, M., (2013). An investigation of the decision-making process in Agile teams. *International Journal of Information Technology and Decision Making*, 12(6), 1097-1120. doi: 10.1142/S0219622013400105

EasyProject (2018). *Méthodologies de gestion de projets: Waterfall ou Agile?* Retrieved from <https://www.easypj.com/fr/contacts/la-gestion-de-projets-en-toute-simplicité-blog-conseils-ressources/673-methodologies-de-gestion-de-projets-waterfall-ou-Agile>

Elaee (2018). Fiche métier: Product owner. Retrieved from <https://www.elaee.com/2017/04/05/27570-fiche-metier-product-owner>

Fernandes, T. (2017). *Spotify squad framework*. Retrieved from <https://medium.com/project-management-learnings/spotify-squad-framework-part-i-8f74bcfd761>

Flora H.K., and Chande S. V. (2014). A Systematic Study on Agile Software Development Methodologies and Practices. *International Journal of Computer Science and Information Technologies*, 5(3), 3626 – 3637.

Fors, H. (2014). *Are you too busy to improve?* Retrieved from <https://hakanforss.wordpress.com/2014/03/10/are-you-too-busy-to-improve/>

Guilmin, C.& Nghunza, N. (n.d.). *Gestion de projet : PRINCE 2*. Unpublished document, Université de Namur, Namur.

Haniff, A.& Salama, M. (Eds.). (2016). *Project management*. Oxford, United Kingdom: Goodfellow Publishers Limited.

Hanson, B. (2018). *What is a Program Manager vs. a Project Manager?* Retrieved from Wrike, Project Management website: <https://www.wrike.com/blog/program-manager-vs-project-manager/>

<https://smallbusiness.chron.com/difference-between-strategic-operational-objectives-24572.html>

Investopedia. (n.d.). *Program Manager*. Retrieved from <https://www.investopedia.com/terms/p/program-manager.asp>

Kamp, P.-H. (2014). Center wheel for success. *communications of the acm* 57(2). 52-54. doi: 10.1145/2556647.255665

Kettunen, P., and Laanti, M. (2017). Future software organisations – Agile goals and roles. *European Journal of Futures Research* 5 (6). doi: 10.1007/s40309-017-0123-7

Kolp, M. (n.d.). *Lean management*. Unpublished document, Université Catholique de Louvain, Louvain-La-Neuve.

Kolp, M. (n.d.). *Scrum*. Unpublished document, Université de Namur, Namur.

KPMG Advisory N.V. (2017). *Agile project delivery*. Retrieved from <https://home.kpmg.com/be/en/home/insights/2017/02/Agile-project-delivery.html>

Kropp, M.& Meier, A. (2013). Teaching Agile software development at university level : values, management, and craftsmanship. In 26th International Conference on Software Engineering Education and Training (CSEE&T) (pp. 179-188). San Francisco, USA: CSEE&T. doi: 10.1109/CSEET.2013.6595249.

- Kropp, M., and Meier, A. (2014). Qualitative Study of Successful Agile Software Development Projects. *IMVS Fokus Report 2014*. 5- 13.
- Kuster, J., and al. (2015). *Project Management Handbook*. Switzerland: Springer. DOI 10.1007/978-3-662-45373-5\_5
- Lejeune, C. (2012). *Séminaire d'accompagnement du mémoire (en sciences de gestion)* -
- Lin, J. et al. (2013). Goal oriented Agile unified process (GOAUP): an educational case study. *Proceedings of the 2013 International Conference on Software Engineering and Computer Science*. ICSECS. doi:10.2991/icsecs-13.2013.9
- Lothon F. (2017). *Le rôle de scrum master en 4 points*. Retrieved from Unow, Le coin des experts website: <https://www.unow.fr/blog/le-coin-des-experts/role-scrum-master-4-points/>
- Mckinney, P. ( n.d.). *What Are Business Goals? - Definition & Examples* [Online Course]. Retrieved from <https://study.com/academy/lesson/what-are-business-goals-definition-examples-quiz.html>
- Mikulenass G., Butleris R., and Nemuraite (2011). An approach for the metamodel of the framework for a partial Agile method adaptation. *Information technology and control*, 40(1), 71-82. <http://dx.doi.org/10.5755/j01.itc.40.1.194>
- Modern Analyst. (2018). *What is a Business Analyst*. Retrieved from <http://www.modernanalyst.com/TheProfession/Roles.aspx>
- Moreau, C. (2016). *Analyse des intérêts et des enjeux d'adopter les méthodes Agiles pour innover au sein d'une entreprise, cas pratique chez Colruyt group* (Master's thesis). Université catholique de Louvain, Louvain-La-Neuve. Retrieved from
- Moreau, Charlotte (2016). *Analyse des interets et des enjeux d'adopter les methodes Agiles pour innover au sein d'une entreprise, cas pratique chez Colruyt Group* (Master's thesis). Université catholique de Louvain, Louvain-La-Neuve.
- Mountain Goat Software. (2018). *ScrumMaster*. Retrieved from <https://www.mountangoatsoftware.com/Agile/scrum/roles/scrummaster>
- Ngakam, M.& Tchunte, L. (n.d.). *Kanban*. Unpublished document, Université Catholique de Louvain, Louvain-La-Neuve.

- Novack, J. (2016). Shu Ha Ri: An Agile Adoption Pattern. Retrieved from Solution IQ, website: <https://www.solutionsiq.com/resource/blog-post/shuhari-Agile-adoption-pattern/>
- Petersen, K, and Wohlin, C. (2009). A comparison of issues and advantages in Agile between state of the art and an industrial case. *The Journal of Systems and Software*, 82(2009), 1479-1490.
- Petryni, M. (2018). Difference Between Strategic & Operational Objectives. Retrieved from Chron, Strategic development plans Web site : <https://smallbusiness.chron.com/difference-between-strategic-operational-objectives-24572.html>
- Project Management Institute. (2018). What is Project Management? Retrieved from <https://www.pmi.org/about/learn-about-pmi/what-is-project-management>
- Quivy, R., and Van Campenhoudt, L. (2006). *Manuel de recherche en sciences sociales*. France, Paris: 3rd Ed. Dunod.
- Radujković, M.& Sjekavica, M. (2017). Project Management Success Factors. *Creative Construction Conference 2017, 196*, 607-615. doi: 10.1016/j.proeng.2017.08.048
- Raucent, J., (2017). LSMF 2021 – project management. Université Catholique de Louvain, Louvain-La-Neuve.
- Reeder, T., (2017). *Why Agile Management? Because it is an increasingly VUCA world*. Retrieved from Next Forge website: <http://nextforge.com/?p=2449>
- Rose, A. (Realisator). (2016). *What Are Organisational Goals? - Definition, Types & Examples* [Online course]. Retrieved from study.com, Business courses website : <https://study.com/academy/lesson/what-are-organizational-goals-definition-types-examples.html>
- Saidani, N. (2018). *Pourquoi un scrum master ?* Retrieved from Pyxis, savoirAgile.com website : <http://savoirAgile.com/2017/09/21/scrum-master-2/>
- Samsonowa, T. (2012). *Industrial Research Performance Management*. Walldorf, Germany: Springer-Verlag GmbH Berlin Heidelberg 2012. doi: 10.1007/978-3-7908-2762-0
- Saunders, M., Lewis, P., and Thornhill., A. (2006). *Research methods for business students*. England, Harlow: 7<sup>th</sup> Ed. Pearson Education Limited.

Schön, E.-M., Escalona, M. J., and Thomaschewski, J. (2015). Agile values and their implementation in practice. *International Journal of Artificial Intelligence and Interactive Multimedia*. 3(5). 61-99. doi: 10.9781/ijimai.2015.358

Semal, P.& Raucant, J. (2017). *Project management*, LSMF 2021. Unpublished document, Université Catholique de Louvain, Louvain-La-Neuve.

Serie, W. (2018). *How to Implement a Great Decision-Making Model at Your Company*. Retrieved from Vistage Minnesota website <https://www.vminnesota.com/blog/minnesota/2015/08/how-to-implement-a-great-decision-making-model-at-your-company>)

Siriram, R. (2018). Project management assesement (PMAs): an empirical study. *South African Journal of Industrial Engineering* May 2018, 29(1), 108-127. doi: 10.7166/29-1-1675

Talentlyft (2018). *Product Owner job description template*. Retrieved from <https://www.talentlyft.com/en/resources/product-owner-job-description>

Techopedia. (2018). *Project Manager (PM)*. Retrieved from <https://www.techopedia.com/definition/677/project-manager-pm>

The Meta Business Analyst. (2018). *How To Explain What A Business Analyst Does To Normal People*. Retrieved from <http://www.metabusinessanalyst.com/how-to-explain-what-a-business-analyst-does-to-normal-people/>

Weil, T. 2008. *Stratégie d'entreprise*. Paris : Presses des Mines. doi :10.4000/books.pressesmines.1322

---

## Appendixes

---

### Appendix 1 : Guideline of the qualitative survey

Bonjour Madame..., Monsieur...,

Je m'appelle Sabrina, je suis le cursus d'ingénieur de gestion à l'UCL Louvain-La-Neuve. Afin de finir mon master, j'ai choisi de travailler sur le sujet « les motivations, les objectifs et l'évaluation de l'implémentation des méthodes Agiles ». L'objectif de notre interview aujourd'hui est de découvrir les liens entre les méthodes Agiles et la prise de décision pour ces méthodes. Je voudrais déjà vous remercier de me recevoir et n'hésitez pas à dire tout ce que vous avez à l'esprit, même détails ou commentaires. Elle se déroulera avec d'abord une présentation de vous, votre entreprise et projets, ensuite plus spécifiquement sur l'historique des méthodes dans l'entreprise, ensuite l'implémentation de méthodes Agiles chez vous et leur évaluation, pour enfin finir avec votre opinion grâce à votre expérience. Puis-je vous enregistrer ?

#### 1. Présentation

Pouvez-vous vous présenter en quelques mots ?

Quelles études avez-vous suivies et quelle est votre expérience ?

Pouvez-vous présenter votre organisation ?

En quoi consiste votre fonction ?

Quel(s) type(s) de projets traitez-vous de manière générale ?

Comment est organisé votre département IT ?

Quelles sont vos objectifs stratégiques dans votre département IT ?

Quelle est la méthodologie de développement que vous utilisez actuellement ? Depuis combien de temps l'utilisez-vous ?

#### 2. Historique des méthodes dans l'entreprise

Quelle était la structure de travail dans votre département à votre arrivée ?

Quels changements avez-vous rencontrés dans la manière de travailler depuis ? (Processus)

Comment et dans quel contexte avez-vous décidé de mettre la/ les méthodes Agile en place ?  
Quelles étaient les motivations ?

Aviez-vous des attentes particulières grâce à ces méthodes ?

### 3. L'implémentation des méthodes Agiles

Quand ? Comment avez-vous implémenté les méthodes Agiles ? (Tout d'une fois ou en partie ? les difficultés, obstacles,) Pourquoi ?

Utilisez-vous les mêmes méthodes pour tous les projets ?

Comment le changement a-t-il été mis en place ? (Formations, mise en place de fonctions particulières...)

Quels étaient les objectifs attendus dans la mise en place du processus Agile ? Comment les déterminiez-vous ?

Est-ce que vous avez atteints ces objectifs attendus ?

Quelles modifications y a-t-il eu dans les méthodes Agiles depuis la première implémentation ? Pourquoi ? Comment ?

### 4. Evaluation

Comment avez-vous remarqué que ces changements étaient nécessaires ?

Comment jugez-vous qu'un projet soit un succès ?

Y a-t-il eu plus de succès (par rapport à vos critères) avec les méthodes Agiles ? Pourquoi ?

Quelles bénéfices les méthodes Agiles ont-elles donnés ?

Quelles sont les évaluations que vous effectuez après implémentation ? Comment se passent-elles ? (Fréquence,...)

Si les objectifs n'ont pas été atteints est-ce que certaines pratiques ont été supprimées / modifiées ? Pourquoi ?

Est-ce qu'il est possible de changer de pratiques lors d'un projet selon la situation ?  
Comment ? Pourquoi ?

Quel a été l'impact, en général, de l'implémentation des méthodes Agiles ? Etes-vous satisfait ?

## 5. Avis personnels

Pensez-vous que l'adoption des méthodes Agiles est dans le but de poursuivre certains objectifs ? Si oui, lesquels ?

Apporteriez-vous certaines améliorations au système en place pour l'instant ? Si oui, lesquelles ?

Quel est votre avis concernant les méthodes Agiles de manière générale ?

Comment présenteriez-vous les méthodes Agiles que vous utilisez selon votre expérience avec celles-ci ?

Connaissez-vous le Agile Manifesto ?

Est-ce que le Agile Manifesto joue un rôle dans votre entreprise ? Si oui, lequel ?

Quelles conclusions pouvez-vous tirer des méthodes Agiles en général ?

## 6. Conclusion

Dois-je rendre cette interview anonyme ?

Avez-vous des remarques à ajouter ou des conseils à me donner ?

Avez-vous des questions ?

Je vous remercie pour votre collaboration et votre temps.

## Appendix 2 : Interview Anonymous 1

M : Methods, process and tools responsible for the IT department in an energy company.

C : Agile coach in an energy company

*Pouvez-vous vous présenter en quelques mots ?*

M. : Je suis rentré chez une entreprise d'énergie il y a maintenant 7 ans. Avant, j'ai fait 4 ans dans le secteur bancaire en fait, toujours dans l'informatique.

Après je suis rentré chez une entreprise d'énergie, d'abord dans le département IT, et l'Agile s'est mis dans la société. Je ne vais pas dire pas par hasard, mais par opportunité. Un projet s'est lancé et des développeurs connaissaient Agile de projets précédents. Mais personne n'avait vraiment le bagage de connaissances. On s'est dit : « Qui connaît l'Agile ici? ». J'ai répondu que j'en avais fait dans mes 4 années précédentes. Durant mon expérience dans le bancaire.

J'ai fait des formations pour cette équipe, il y a eu un effet boules de neige et finalement je suis devenu le responsable Agile de l'entreprise à l'issue d'une réorganisation, même si j'étais rattaché à l'informatique.

On a entamé une démarche avec un coach Agile qui a été recruté et nous a rejoints il y a maintenant un an et demi. On accompagne les équipes qui veulent travailler en Agile et on a fait une formation interne propre à la société. On travaille également avec des fournisseurs externes qui nous donnent des formations certifiantes reconnues par le Scrum.org qui est un organisme certifiant. On a pas mal de contact avec le département RH, business.

On a un département dans l'entreprise qui s'appelle le département transformation et qui s'assure de la bonne application de la gestion de projets. Et donc, on fait l'interface entre l'IT et celui qui est owner du scrum.

*Ça fait combien de temps que vous êtes occupé ?*

M. : Ça fait 4 ans que j'ai reçu ce mandat Agile officiel. Avant j'ai fait 1 an et demi, 2 ans sur le côté par motivation, mais ce n'était pas mon job officiel.

*C'est quoi ta fonction par rapport à ce monde-là ?*

M. :Ma fonction, c'est responsable méthode et processus et outils pour le département informatique et maintenant dans cette fonction, j'ai un mandat d'agilité qui est plus large au niveau de l'entreprise.

C. :Je suis indépendant depuis un certain temps avec une formation d'informaticien. En résumé, à partir d'une fonction, au départ de développeur je suis passé par teamlead, project developer, architecte.

Donc au niveau de l'IT, quand j'ai été dans une phase de project manager, et par rapport à des aspirations personnelles je trouvais que ce type de management était relativement oppressant et violent par rapport à une équipe. Je n'étais plus du tout en phase avec la manière de faire même si c'était la manière dont on fonctionnait.

J'ai eu l'opportunité chez ING de commencer à faire en 2010, des tentatives d'agilité dans un programme que je gérais et qui était relativement ouvert à ça. Car c'était dans le monde du digital. J'ai fait de l'agilité au niveau de l'IT pour voir comment cela fonctionnait.

Ça a évolué et petit à petit, j'ai pris des fonctions de Scrum Master dans 2,3 boîtes et en 2014, je me suis endossé le rôle de coach Agile. J'ai accompagné quelques sociétés.

Je suis depuis 1 an et demi chez une entreprise d'énergie dans une fonction clairement de coach Agile au départ. Mais l'agilité ne peut fonctionner que si autour il y a une démarche d'intelligence collective. Avec mon collègue et d'autres partenaires, on essaye de développer, d'accompagner du change culturel, car l'agilité ce n'est pas que du savoir-faire, mais aussi du savoir-être et pour l'instant c'est un des focus.

*Chez une entreprise d'énergie, avant l'implémentation Agile, quelle méthode utilisiez-vous ?*

M. : J'ai commencé en 2007 chez SWIFT, domaine bancaire très sécurisé, où on utilisait Prince 2 avec du Waterfall. Ce qui était le trigger, ça a été la crise bancaire.

Ils se sont dit qu'on ne pouvait plus attendre de faire des releases, tous les 4 mois, comme on le faisait. On était dans un cadre Prince 2. Ces préoccupations étaient loin de moi, je faisais mon code et en passant au Scrum, je me suis dit que je devenais un acteur projet. Je suis beaucoup plus proche de mon client final, de la réalité du projet.

*Et par rapport à l'entreprise d'énergie, quelle était la méthodologie auparavant ?*

M. : On a commencé à faire une gouvernance Agile avant qu'il y ait une gouvernance Prince 2 qui s'installe. La gouvernance Prince 2 est venue du département de la transformation, qui a dit qu'on allait mettre du PRINCE 2. On a intégré Agile avec le Scrum Master qui est un chef Agile.

*Ça a dû être compliqué d'implémenter Agile en même temps que PRINCE 2 ?  
Comment cela s'est passé ?*

M. : Déjà, l'Agile, sans les considérations PRINCE 2, il faut savoir que c'est franchement compliqué, car on parle du savoir-être dans l'entreprise qui se vit quotidiennement. Le management va être extrêmement influençant. Il y avait au départ, des managers qui disaient je décide, tu exécutes. Ce sont des équipes qui vont prendre des décisions, et pour nous, c'était inconcevable, donc ça a été assez compliqué. Mais l'idée était d'arriver rapidement à des résultats et que ça fonctionne. Et donc, on a quelques beaux exemples qui ont légitimés l'agilité.

Ensuite le cadre Prince 2 est arrivé, plus pour des considérations budgétaires, mais pas propres à l'agilité. On avait du mal à piloter, à avoir une vue sur les projets et les gros échecs étaient non Agiles, donc le cadre Prince 2 a été mis en place.

En effet, en soi, il n'y a pas d'incompatibilité, mais parfois on rentre dans l'humain et ça devient presque une guerre de religion, car les personnes le mettent sur ce terrain-là.

Prince 2 est dans un cadre ou Scrum et Prince 2 ne peuvent cohabiter, mais où l'inverse est possible. On rentre dans l'humain, le personnel, avec certains ça se passe bien avec d'autres, c'est plus compliqué. Ça dépend du mind-set de la personne, de son bagage, de son expérience.

*Si je comprends bien, l'idée de l'agilité est venue bottom up, et puis, Prince 2 top down parce que le management ne s'y retrouvait pas, c'est bien cela ?*

M. : En fait, c'est parti bottom-up. Il y a eu des résultats, le business était satisfait. Quand on va vraiment donner le mandat, l'IT a un très mauvais bilan. Le directeur s'est dit qu'il devait changer les choses. Là, j'entends mon partenaire du business qui me dit qu'il était content, car ils étaient dans mes projets. On a vu que ça tournait, on avait de bons résultats.

Il y a eu un mandat top-down qui a été donné, mais après. Sans vouloir critiquer, chaque démarche est louable, mais la manière, c'est une façon compliquée d'amener les choses, car

très théorique. Sur un terrain de projet, il délivre, donc si on doit lire des PRINCE 2 by the book, qui est dire : « Aller, on va faire comme ça ».

En fait, il y a une croyance, pas spécifique à UNE ENTREPRISE D'ÉNERGIE, on pense donc ça se fait et ça va aller. C'est une croyance que de la théorie à la pratique, il n'y a plus qu'à faire.

*Ont-ils mis des formations en place ?*

M. : La spécificité c'est que des personnes ont été mandatées pour mettre en place PRINCE 2, mais n'avaient aucune expérience projet et aucune connaissance PRINCE 2. Ils ont eux-mêmes été suivre une formation et régurgité ce qu'ils avaient appris en formation et donc quand tu leur poses des questions, ils vont voir dans leur slide ou leur syllabus et donc niveau crédibilité, c'est limite. Et alors, il y a des expériences où on nomme un expert projet, mais on perd un expert business. Le chef de projet doit tout apprendre, mais n'est pas vraiment accompagné.

Ils ont fait ce qu'on appelle des formations externes, des lunch & learn. Dans ces lunch & learn, tu viens à midi, ils déroulent les slides, mais ça reste au niveau de l'accompagnement à notre avis, ça reste limité. Tu as suivi la formation, et l'aspect pratique leur manque. Donc, en Prince 2 on leur dit il faut remplir autant de template, il y a la liste à faire.

C. : Ils ont oublié à mon avis une étape importante c'est de customiser Prince 2 qui le permet. C'est dit dans l'introduction de Prince 2 qu'il peut être adapté à l'entreprise. Il y a plein de choses que l'on doit prendre et d'autres qu'on ne doit pas prendre. Ça je pense, que cette étape n'a pas été bien suivie. Ils ont fait des choix. Mais maintenant, c'est relativement jeune comme démarche. On se verrait dans 10 ans, on pourra raconter que le début était difficile, on a fait des erreurs, mais on a également appris de celle-ci. Beaucoup d'entreprises passent par là. Il faut de temps en temps faire des erreurs pour en apprendre.

Une entreprise d'énergie est très jeune dans cette méthode. On a commencé en 2016, ce n'est même pas 2 ans et la société n'a pas 10 ans.

M. : Quand je suis entré dans le département informatique, on était 50. Nous sommes à présent 250. Beaucoup de choses se mettent en place, mais niveau maturité, il faut du temps.

*En ce qui concerne l'agilité, quelles méthodes utilisez-vous ? Vous avez parlé de Scrum, y en a-t-il d'autres ?*

M. : On utilise aussi du KANBAN, on a fait un petit rapport des activités Agiles. Au niveau des pratiques employées, on a peu découplé des sprints, des product owner mais on a des équipes qui ne font que du daily. Ils n'ont pas tout le framework Scrum.

On a également des pratiques qui sont dérivées de l'eXtreme Programming, avec intégration continue.

*Donc, vous utilisez plusieurs pratiques de plusieurs méthodes comme : Scrum, Kanban, et eXtreme Programming, y en a-t-il d'autres ?*

C. : En fait, ce n'est pas une méthode, mais ce sont des cadres. C'est différent comme approche et donc il y a cette flexibilité-là de nos jours. Le cadre en fonction de la disponibilité l'équipe.

M. : Il y a une équipe support chez UNE ENTREPRISE D'ÉNERGIE, il y a des gens qui viennent tous les jours,

C. : Il y a Scrumban, en fait ce n'est pas une méthode, c'est quand même différent comme approche et donc il y a cette flexibilité-là aujourd'hui qui permet de coller le cadre en fonction de la maturité ou de la disponibilité de l'équipe et des rôles qui se font. Comme disait mon collègue, il y a une équipe support par exemple ici chez une entreprise d'énergie, il y a des gens qui viennent tous les jours ici qu'on n'a jamais vus. Ils sont dans une pièce quelque part.

M. : Ils sont support deuxième ligne, ils travaillent avec la clientèle, les problèmes inhérents à la clientèle, leurs clients. C'est-à-dire, par rapport à des produits qu'on met à la disposition d'une entreprise d'énergie, et un système de ticketing, etc. C'est une équipe dans tous les cas, c'est ce qu'ils ont dit : chacun travaille dans son espace et personne ne savait exactement ce que faisait l'autre, et il y en avait qui étaient débordés et d'autres qui n'avaient pas grand-chose à faire. Et donc là, fort de ces constats, du support du management et de la mise en place, et préparation plus un vrai workshop, etc. Ils ont mis en place au minimum le daily. Ils ont un peu traficoté le daily. Ce n'est pas ce que j'ai fait, ce qui se passe et que je rencontre comme obstacle. Cela laisse de l'espace beaucoup plus à la disponibilité et l'émotionnel parfois. Mais ils y vont. Mais c'est une étape. C'est une étape, mais ils ne sont pas orientés projets non plus. Ils sont plus dans de l'opérationnel. Cela existe aussi l'agilité pour l'opérationnel. Et donc l'agilité pour l'opérationnel c'est plus orienté Kanban et autre. Leur

présenter cela ils ne vont pas comprendre parce qu'ils sont encore trop dans leurs silos et ils débutent. Et ça, c'est une façon d'approcher l'agilité comme une autre.

Et ils font une rétrospective une fois toutes les 2 semaines parce que justement, ils ont eu du mal à collaborer. Ils font une rétro toutes les 2 semaines sans que ce soit, enfin c'est un des éléments du cadre Scrum qu'on a sorti et bien en fait on va quand même utiliser c'est quand même de la valeur même s'il n'y a pas tout le cadre. Et alors, on est aussi en train de réfléchir à déployer dans le cadre DevOps aussi.

C. :Donc voilà, il y a des projets digitaux donc qui ont été les précurseurs de l'agilité dont le web, entre autres. Voilà, nous on n'a pas envie de travailler comme cela, et après s'ils ne sont pas contents on verra. C'était l'étape. Et ceux-là, ils ont 4 ans de maturité pour l'agilité. Donc eux dans un modèle, ils peuvent se permettre de jouer dans le cadre à leur façon parce qu'ils ont la maturité, ils étaient autonomes. Donc ça c'est des candidats intéressants pour aller vers le DevOps. Et ce que je viens de dire, l'écart avec il y a 2 mois, c'est encore compliqué au niveau du savoir-être.

*Et en ce qui concerne vraiment Scrum, Kanban, vous avez choisi ces modèles comment ?*

C. :Je pense que Prince 2 c'est une méthode et le cadre Agile peut très bien travailler avec une méthode Prince 2. Itil, sans problèmes. La question c'est juste trouver comment est-ce qu'un cadre de travail, un savoir-être et un certain savoir-faire peut se marier avec une gouvernance méthodologique comme Prince 2. Il n'y a pas d'incompatibilité dans un cadre, on pourrait le faire avec n'importe quel cadre de travail à partir du moment, où à un moment la gouvernance on répond à certaines exigences. Parce qu'il y a des gens qui pilotent et qui paient donc cela se passe bien. C'est pour cela que j'insiste.

*Comment avez-vous sélectionné les modèles ? Pourquoi ces modèles-là et pas d'autres ?*

M. :Scrum c'est simplement par opportunité, parce que le projet qui devait démarrer avait un cadre projet qui s'y prêtait bien. C'est aussi ce que les membres de l'équipe avaient déjà un peu expérimenté. Ce que j'avais expérimenté aussi. Après, depuis lors, on a aussi développé des critères d'éligibilité. On va justement passer différents critères en revue en disant : est-ce que c'est un bon projet ? On pourra parler de produit, mais qui peut rentrer dans du Scrum. En

fait, on screener des critères équipes Product Owner, Scrum Master, des produits techniques, fonctionnels ou qualité et enfin l'organisation.

Est-ce que par exemple, le Product Owner a de l'expérience, a un niveau de maîtrise ? Est-ce qu'il connaît son produit ? Est-ce que c'est quelqu'un qui va découvrir son produit en même temps que l'équipe ? Est-ce qu'il va y avoir une vraie capacité de décision ? Ou est-ce qu'il va chaque fois remonter vers un organisme, etc. Donc cela peut freiner la vélocité de l'équipe. Est-ce qu'il va être disponible ? Présent pour l'équipe ? Là, en termes de produits techniques, bien voilà est-ce qu'on maîtrise les technologies ? Est-ce qu'en termes de complexité, est-ce que c'est aussi OK ? Les technologies, est-ce que c'est applicable ? Par exemple on se rend compte que tout ce qui est out net c'est assez facile à faire. SAP, moi j'ai connu Scrum avec SAP chez un autre client, mais ici ils sont plus frileux. Donc on n'a pas encore fait avec SAP. Au niveau fonctionnel, Built automatisé, est-ce qu'il y a des outils d'annexe de code ?

*Toute l'analyse qui est faite, mène à des conseils pour des pratiques à utiliser ?*

M. :Ici, c'est plus pour le Scrum. Quand la demande arrive, on va screener tout ça, on va faire une analyse de risque, en fait, avec les cotes les plus basses, on va faire un plan de mitigation. Voilà, ici je prends le Product Owner, il n'avait jamais fait d'Agile, et on a dit : « Tiens, Collègue va lui donner un coaching plus rapproché ». Donc on fait une sorte de plan pour mitiger tous les risques en fait qu'on passe en revue régulièrement.

Maintenant, pour les autres méthodes, là c'est plus s'il y a une demande en fait, une discussion entre la personne et nous en fait. On va aller plus vers l'un ou vers l'autre. Pour donner un exemple, il y a un nouveau service qui a été créé il y a quelques mois, avec lequel on va déployer du Kanban. Simplement, parce que c'est une équipe qui est plus opérationnelle et qui gère des petites demandes. On fait un mix de nouvelles petites demandes, d'incidents ou de problèmes qu'ils doivent délivrer à intervalle régulier. Donc là on se dit que le Kanban c'est ce qui nous semble le mieux. Plus de flexibilité, pas attendre la fin d'un sprint pour résoudre un problème.

*Toutes les équipes ont-elles reçu une formation Scrum, Kanban et autre ? Ou c'est via le bouche-à-oreille qu'elles ont eu envie d'en savoir plus ?*

M. :Donc, on a fait un peu de publicité, donc des sessions d'information, qu'on a faites nous-mêmes, à laquelle n'importe qui pouvait s'inscrire s'ils étaient intéressés, dans lesquelles on présente les différents outils qu'on a dans notre boîte à outils : Scrum ou Kanban aussi un peu

Lean, etc. Et alors les gens, c'est juste les Wireless en fait, voilà ce qui existe. Et puis on a souvent après des retours : moi je suis quand même intéressé d'approfondir un peu ça. Et donc on discute là, avec les gens pour l'implémenter. Alors ça, c'est pour la partie sensibilisation, ensuite quand on rentre vraiment dans une démarche concrète, là soit on fait une formation spécifique, Scrum ou Kanban donné par nous-mêmes ou pour certains facteurs-clé, on fait venir un formateur externe qui a la certification Scrum Alliance qui soit délivrée donc qui passe un test avec les questions, c'est important d'avoir une certification.

C. :Il faut 80 je pense.

*Pour Kanban, donnez-vous les formations vous-mêmes ? Pour la nouvelle équipe par exemple.*

M. :On a un petit desk de formation qu'on peut donner nous-mêmes. Maintenant, une des lessons learned de ces années, on s'est dit que les facteurs-clés doivent être certifiés. Donc, avant on donnait juste les formations nous-mêmes. Mais maintenant, il faut vraiment monter la barre, donc tous les Scrum Master doivent être certifiés. Alors pour le Kanban il y a une société qui fait passer des formations, certifiées aussi en Kanban certifié design qui est en fait une sorte de Lean appliqué au Kanban, justement tu vas minimiser le temps d'attente que le client a. C'est à partir de septembre. Jeudi de la semaine dernière, on a attribué le projet. Et donc à partir de septembre, il y a une société qui va venir donner une formation à ces gens qui va leur parler en vue de passer la certification et qui va aussi les suivre et les coacher.

*Kanban va suivre le même principe que Scrum : suivre les formations certifiées et puis appliquer ?*

C. :Kanban c'est un peu différent. C'est un peu la maturité d'une équipe qui passe du Scrum au Kanban, mais qui a besoin de garder un cycle itératif de delivery. En fait, c'est typiquement des équipes projets qui par exemple passe en mode de pré-production, et qui souhaitent délivrer toutes les semaines. Mais à ce moment-là, il n'y a plus toutes les cérémonies, ils délivrent au fil de l'eau tous les bugs par exemple, et toutes les corrections et améliorations et autres, et donc cela reste. Il n'y a pas vraiment une pratique Scrumban en tant que tel. Maintenant, il y a pas mal de littérature. C'est l'addition en fait du Scrum surtout dans son côté itératif, et le Kanban. Parce que le Kanban ne définit pas spécialement une période de livraison.

*Pour les pratiques qui viennent de eXtreme Programming, pourquoi avoir choisi ces pratiques-là ? Est-ce que vous en aviez entendu parler ?*

C. :Ça, c'est vraiment du pure bottom-up, c'est dans les cadres Scrum, développer plus senior. Les techniques ont en fait apporté par leur bagage, précédent à cette époque-là et les mises en place en plus. Donc c'était vraiment l'héritage des individus en fait.

*Cela se propage ?*

C. :Pas toujours, ici justement, on aimerait bien que cela se propage, on aimerait avoir plus pour répondre aux différentes demandes. Il y a des petits chantiers locaux qui sont les sommes des héritages, et on aimerait bien étendre que les expériences se mélangent en plus.

*Avez-vous déjà fait des modifications dans ces modèles pour les adapter à votre entreprise ? Et si oui lesquelles ?*

M. :Un exemple c'est, il y a les rôles Scrum Master, Product Owner, l'équipe, qui sont définies de manière assez high-level dans le Scrum. Et suite à différents évènements, on a décliné ces rôles, on a rédigé une fiche de rôles où il y a certains éléments qui bougent un petit peu, mais pour s'intégrer dans les différents rôles de l'organisation. Le project manager, dans le cadre pur scrum, il n'y a pas de project manager en fait. La responsabilité elle est où ? On a un IT project leader qui est en fait le bras droit du project manager pour les aspects IT. Du coup, Project Manager IT, Product Owner et Scrum Master cela fait beaucoup de monde. Donc qui fait quoi exactement ? Ça, c'est un des gros évènements. Après on n'a pas réinventé la roue, on a simplement fait que ce soit clair, pour éviter les tensions des différents rôles

C. :Donc la question, c'est qu'est-ce qu'on a adapté par rapport à la guidance classique ? Je pense que les équipes, d'elles-mêmes adoptent certaines pratiques qui adaptent après le Web. C'est une expérience. Par exemple, un meeting avec le Web où ils vont faire une démo, ils vont faire tout en même temps. Ils vont faire la rétrospective, et ensuite ils vont faire directement la démo et puis seulement après ils font le planning. Donc, en fait, ils font 2-3 heures de réunion, ils font tout le bazar en même temps, puis ils sortent. C'est quand même différent, déjà c'est une approche un peu différente de l'aspect classique, on se voit d'abord pour la review, pendant cette review on se fait une rétrospective, une fois qu'on a fait cette rétrospective, on passe un week-end, parce que voilà il y a déjà 3 heures de passées. Et on essaye de trouver de l'amélioration, et puis on fait le planning le lundi.

C'est une manière de faire, et puis alors, il y a peut-être quelques légèretés, voilà il y a quelques aspects, on oublie de respecter la définition of done. Quand je dis oublier, ce n'est pas oublier en fait, c'est délibéré. Mais parce qu'ils ont une réalité de terrain, avec les livrables qui font que parfois il faut pousser déjà lentement l'environnement d'acceptance alors que cela n'a pas été testé, ou suffisamment testé. Et puis au petit bonheur la chance, il y a des effets de board, des régressions. Cela entache la qualité qui est quand même l'objectif de l'agilité, c'est la qualité et délivrer de la valeur au client.

M. :Après il y a certains éléments que l'on a fait, on a essayé le Scrum-Scrum, mais au bout de 3 mois on s'est dit que c'est une bonne idée, c'est chouette de se voir, mais cela n'apporte pas vraiment de valeur donc, on a fait une petite rétro perspective et on a décidé d'arrêter, et voilà cela reste une expérience positive et on a essayé quelque chose.

Après ce qu'on a aussi, ce n'est pas une liberté, c'est l'application du Scrum si on fait du Scrum by the books, et on dit il n'y a pas de sprint 0. On commence, on construit et on y va. Ici, il y a quand même une phase préparatoire avant de démarrer les sprints, qui s'est appelé dans ce cadre-ci. Un cadrage, qui est fait pour délimiter le périmètre fonctionnel, avoir déjà l'architecture qui est quand même définie un minima, ce qui permet dans le cadre des projets d'avoir une validation budgétaire des enveloppes. Sinon on partait sans vraiment savoir où on allait, de quelle enveloppe on disposait, et quand même une phase de cadrage, qu'on peut appeler, sprint 0 ou pas, quand même une phase préparatoire qui a quand même démarré dans le cadre de Scrum.

*Est-ce que vous aviez des attentes particulières en mettant en place ces modèles Agiles dans l'entreprise ?*

M. :L'organisation avait des attentes, la principale c'était le time-to-market, il fallait une année calendrier avant de délivrer. Ici, on répond au business dans des temps plus rapides, et ce qui était à l'époque, une attente, c'était propre par rapport au bien-être des gens : en termes de collaboration et de bien-être ce qui n'était pas du tout à l'époque, c'est le paradis, mais ce n'est pas le Club Med non plus. En tous les cas, si je reprends les bénéfices, en fait c'est un sondage, tous les gens qui ont fait Agile ont reçus ce sondage, les points qui ressortent le plus, ce n'est pas tellement des points time-to-market, productivité, c'est plus de la collaboration 100% constaté de bénéfices, 77 % des répondants ressentent de la collaboration, de communication, de visibilité, et là on arrive avec les prises en compte de changements plus rapides, cela ne veut pas dire que l'on livre plus rapidement, et puis le moral, la motivation.

*Ces attentes du management, se sont-elles concrétisées au fur et à mesure du temps ?  
Avaient-ils davantage d'attentes au fur et à mesure du temps ?*

M. :Oui l'appétit vient en mangeant. Quand on a reçu le mandat, c'était un petit peu la panique donc il faut qu'on livre tel, tel projet, et on les a livrés. Puis il y a eu différents évènements qui ont fait que notre Directeur qui initiait ce mandat a été remercié, et le nouveau Directeur était un des manageurs qui initialement a vraiment soutenu la démarche. Parce qu'au début, je me suis senti un peu seul, les 3 personnes sur qui je pouvais m'appuyer dont cette personne-là qui était nommé Directeur. Et depuis 9 mois ici qu'il est directeur lui il a envie d'aller plus loin, c'est donc juste un petit travail des vacances d'été, c'est d'aller à la rentrée vers le Comité de Direction en disant : on a envie de passer en Agile 2.0, voilà nos recommandations. Il souhaite vraiment étendre les choses, ce que les gens souhaitent en bottom-up, l'Agile aussi. Donc dans les répondants, ceux qui ont répondu ils en veulent plus aussi. Donc 100% recommandent et 87% souhaitent qu'on étende l'agilité, 10% trouvent qu'il y en a suffisamment, et il y en a 3 % qui disent qu'ils ne souhaitent pas qu'on continue. Ce qui est bizarre parce que cette même personne a répondu qu'elle recommandait les pratiques Agiles et en même temps elle ne veut pas qu'on continue.

C. :Tu peux recommander l'agilité en général, mais peut-être pas pour Une entreprise d'énergie.

*Je vois que vous avez évalué cela par le feed-back des personnes qui mettaient cela en place. D'autres types d'évaluations ont-elles été faites ?*

M. :Ce qui n'est pas explicité comme évaluation, mais bon, c'est la réalité du terrain, c'est la réussite des projets. Qu'est-ce qu'on entend par réussite ? Est-ce qu'on est on time, on budget ? En tous les cas, que les projets délivrés aient la satisfaction du business.

*Donc là, on voit une belle amélioration avec les modèles Agiles ?*

M. :C'est un peu caricatural de dire cela, mais les seuls qui délivrent en fait, c'est les équipes en Agile en fait.

C. : Moi je connais le portefeuille Agile, mais il y a une disparité aussi entre projet et programme. Ici les gros programmes qui travaillent particulièrement avec des sous-traitants, des grosses organisations, et ceux-là l'agilité n'est pas vraiment intégrée. On n'en parle pas cela s'arrête là où il y a des petites tentatives.

*Et pourquoi ?*

C. : La maturité n'est pas encore suffisamment-là que pour dire que l'on peut embarquer dans notre modèle ce type d'approche. D'autant que parfois c'est même délocalisé. Il faudrait des fournisseurs qui soient à l'intérieur, qu'il y ait une demande vis-à-vis de nous qu'on accompagne l'agilité. Il faudrait que cela puisse se faire dans des conditions optima, qu'il y a un vrai sponsoring dans le programme. Comme c'est stratégique, c'est compliqué.

M. : On est dans du Waterfall, où tout ce qui décharge doit être rédigé, être out-sourcé à une société externe donc voilà.

C. : Tandis que les projets s'orientent maintenant vers l'agilité par défaut. Les autres projets ou programmes délivrent, mais souvent c'est le retard qui est pointé.

M. : Disons que l'on a eu des expériences il fallait faire quasiment le projet 2 fois, donc le projet a été arrêté à un moment donné, parce que hors budget, en retard, et puis en fait ils ont dû tout jeter et tout recommencer. Il y a eu malheureusement plusieurs cas dont celui-ci. Les projets en Agile, on a eu un projet qui ne s'est pas très bien passé, mais qui a fini par délivrer quelque chose, mais dans sa grande majorité on livre dans les temps encore la semaine dernière, on time budget, le business est content donc voilà. Tout n'est pas parfait, mais en tous les cas, i on compare les projets Agiles réussis et les projets Waterfall il y a une grosse différence.

Dans la façon d'évoluer, mon collègue coach Agile, point de contact de tous les coachs Agiles, alors, il est mis en place des comités de pratique. En fait, tous les Scrums Master se réunissent, tous les Product Owner aussi même si on a du mal à les rassembler pour les agendas. Et alors ils échangent un peu : voilà moi j'ai telle problème informatique, j'ai tel problème avec l'équipe, j'ai tel soucis, on pourrait peut-être faire cela. Donc une façon de prendre la température dans chaque équipe de manière centralisée, et envisager des plans d'action. Dans ce rapport-ci par exemple, on a fait une petite synthèse du feed-back des Scrum Masters. Il y a des gens qui connaissent très bien la vie dans les équipes, et aussi des gens qui ne connaissent absolument pas, qui n'ont pas le mind-set, donc c'est difficile de faire une cohésion d'équipe avec des gens qui ont une telle disparité. Il y a en plus de ressources qui changent dans la vie du projet, donc, il faut chaque fois construire, l'automatisation des tests pourrait être améliorée. Ils ont relevé le fait que le management n'était pas toujours très soutenant. Tout est synthétisé.

*Et votre évaluation est-elle encore en cours maintenant ?*

M. : On a fait le sondage, on l'a fait en juin, on est en train d'écrire le rapport. Il y a quelques petits trucs qui vont bouger, mais les grandes lignes sont là. C'est la 2ème édition qu'on fait cela, j'avais déjà lancé cela en 2016 en fait, ou les conclusions étaient quasiment similaires. Donc, 100% des gens recommandaient, naturellement la maturité de l'entreprise était très positive vu que cela faisait 1 an, 2 ans, il y avait énormément de débutants dans l'organisation. On utilisait aussi à l'époque la stratégie Map, qui était en fait un lien entre les goals et le kanban opérationnel. Mais cela n'a pas vraiment pris. C'était aussi principalement le Scrum qui ressortait. Donc on voit de plus en plus de bénéfices au fur et à mesure du temps. Par rapport au rapport de Version One, on est plus ou moins en termes de bénéfices en termes de répondants, dans les proportions, on est plus ou moins aligné avec la Version One là où il y a 2 ans on était beaucoup plus bas. En termes de moral, on avait un gap de 45 %. Rapidité de livraison 65%, mais ici on est beaucoup plus proche, on a un delta qui est bon. Maintenant, c'est la vue subjective des personnes qui a répondu.

Maintenant je vais me focaliser plus sur vos avis personnels, que pensez-vous des modèles Agiles en général ? Que pourriez-vous faire, comme conclusions sur ces modèles.

C. : Dans tous les cas pour l'expérience, c'est d'abord un savoir-être, un savoir-faire, donc pour moi c'est vraiment important de consacrer du temps, de l'énergie et donc de l'argent aussi au panier de change, ces changements, cet aspect type d'ouverture, transparence, engagement, responsabilisation, dynamique collective, innovation et oser faire des erreurs. Ça, c'est pour l'aspect du savoir-être. Alors par rapport au leader et au top, c'est les aider à prendre confiance, par rapport à cette démarche et cela demande un lâcher pris, de ne plus être dans le coupe-rôle, laisser les gens faire comme ils veulent, à partir du moment où ils sont engagés ils le font comme ils veulent, comme ils le sentent. Dans toutes mes expériences précédentes, toujours très négatif par rapport au projet. On ne peut que gagner du temps ou de la qualité. Au mieux, la qualité que ce soit sur le livrable et la qualité relationnelle humaine. Moi je travaille pour que les gens le matin disent yes et pas : encore cette équipe, ces projets. On voit la pression du manager qui a lui-même eu les pressions en cascades, voilà c'est pour cela que je travaille.

Après qu'on fasse du Scrumban, du Scrum, c'est bien utile parce que cela permet d'avoir un cadre, c'est important d'avoir un cadre. Au même titre, quand on fait des animations en agence collective, on a un cadre et on se rend compte que les gens dans une réunion, où ils ne sont pas spécialement productifs, parce qu'il n'y a pas vraiment un cadre dans la réunion, il y a un protocole, il y a un agenda, on prend des notes, le type qui est là et qui n'a rien à faire là,

il s'en va parce qu'il perd son temps. Je pense qu'on a un cadre, les gens sont rassurés et les gens se sentent bien.

*Et pour le savoir-être, vous avez déjà mis des outils en place ? Vous avez déjà commencé à faire ce change ?*

C. : Donc oui, il est en marche, il n'est pas encore soutenu suffisamment au point de vue de la Direction, mais voilà cela se passe et là-dedans on a un petit peu, on a développé des outils d'intelligence collective en tous les cas, on est en voie de les développer, de les incuber. Pas plus tard que la semaine passée, on a réuni 17 personnes, et il y a encore la même réunion qui va se produire au mois de septembre plus ou moins avec d'autres personnes, pour le même sujet. On sera 34-35 et ces personnes sont prêts, ont un besoin de ressenti, d'un mal-être. Qu'est-ce que l'outil va répondre à ce besoin, pour faciliter les choses ? Donc ces outils vont être définis, vont être couverts sur un planning, certaines personnes vont être aux normes de challenger le produit, de l'essayer, de le coter et de revenir avec un feed-back. Donc il va y avoir un certain nombre de produits, mais on applique déjà des techniques, animation de workshop en lead forum.

M. : Moi l'exemple que j'ai bien aimé, on a animé un séminaire IT et les gens dans la culture d'entreprise souvent arrivent en retard dans la réunion, on n'écoute pas vraiment, on arrive peu à des décisions à l'issue de la réunion. On a mis tout un cadre lors de ce séminaire de 3 jours, on a mis des logs de décisions et on a fait toute une animation pour cadrer les choses, et là on a été surpris de voir comment les gens étaient disciplinés, à l'écoute, vraiment le focus. Et le lundi qui a suivi, on a eu des réunions avec les personnes qui étaient au même séminaire et ont repris leurs habitudes parce qu'ils étaient sortis du cadre.

C. : Ils sont sortis là-bas et c'est l'attitude, mais bon c'est un pas. Il y a encore du chemin. C'est la vie de l'humain c'est comme cela.

*Est-ce que ce savoir-être, on peut le mettre en parallèle avec l'Agile Manifesto ? Est-ce que toute l'entreprise est au courant de l'Agile Manifesto ?*

C. : Il est devant la machine à café et après je ne sais pas dans quelle mesure les gens le comprennent, l'adoptent ou pas. Il y a une interprétation du Manifeste Agile. On parle souvent de la doc. Parce que la conclusion que du coup il ne faut plus de doc, alors que ce n'est pas cela qui a été dit. Il est dit, si on comprend, qu'il faut juste la documentation strictement

nécessaire. On ne fait pas la doc pour la doc. C'est un peu plus dans le savoir-faire. C'est vrai, que les aspects collaboratifs, qui sont mis en avant c'est fort important.

M. : Il y a un vrai tournant qui est pris pour le moment, où l'entreprise veut se lancer dans le monde du travail, dont l'Agile fait implicitement partie, en termes de participatif, des responsabilisations. La façon dont le chef va se comporter, auparavant tu te tais, tu exécutes, c'était assez abrupt on va dire, mais certains répliquaient le modèle.

Non, les décisions ce n'est plus moi qui doit les prendre, c'est vous qui êtes au contact du terrain. Petit à petit, les choses se mettent en œuvre ça part d'un sponsoring en tous les cas d'un exemple d'effort. En tous les cas, Benoît a eu la formation il connaît le manifeste. Mais ce n'est pas quelque chose qui est placardé sur les murs. L'entreprise fait un changement de culture, il y a différents éléments de culture qui sont communiqués, qui peuvent se rattacher pour certains à ceci, qui vont favoriser les choses, mais il y a un terreau qu'il faut nourrir en fait. Au niveau de l'humain, il y a tout un passif. Les gens ont des habitudes de fonctionnement, et puis pendant 20 ans, du jour au lendemain, ils ne vont pas s'intégrer dans un mouvement Agile.

*L'Agile Manifesto, joue-t-il un rôle dans votre entreprise ?*

C : Non c'est comme les croyants avec l'évangile. Il y a des gens qui vont le faire et comprendre ce qu'il y a derrière. D'autres se donnent bonne conscience en allant à la messe le dimanche.

En attendant, l'un et l'autre font un pas vers ça, mais je pense que le manifeste Agile en tant que tel est cité dans toutes les formations. C'est un guide qui pour l'instant, et je ne suis pas le seul à le penser devient obsolète. Je pense qu'il serait temps d'en faire un nouveau. Je pense que ça se travaille. C'est un code de conduite tellement interprétable. La réponse est plutôt oui et non dans l'entreprise.

M : Non, ce n'est pas une base de référence dont on va se servir. C'est trop implicite que pour pouvoir être utilisé.

*Apporteriez-vous certaines améliorations au système mis en place concernant les méthodes Agiles ?*

C : L'origine de l'agilité est principalement du cadre Scrum part d'un monde purement IT. C'était du Dev pour du Dev.

Aujourd'hui ce qui manque, ce qu'on devrait mettre plus en évidence dans le cadre du cadre Agile, c'est « on fait quoi avec les managers ». C'est pour moi le plus handicapant c'est le management, toute la couche middle management par rapport à une approche où au niveau de la direction, les cadres de base devraient porter la vision, la culture.

Toute la tranche du milieu, project manager, ... Tous ces rôles, ça pose problème, car ils sont très mal à l'aise dans la situation. On leur demande de devenir des leaders. Or, ils ne sont pas prêts ou pas nécessairement prêt à être des leaders.

Pour certains, c'est la libération, mais c'est vraiment très peu. Pour eux, cela s'explique par le fait qu'ils ont été formatés en tant que manager. Leur père a pu leur dire mon fils, tu vas faire des études d'ingénieur. Tu commences dans le management, tu vas grimper dans la hiérarchie. C'est son modèle, quand on lui dit lorsqu'il est au milieu de la pyramide qu'on va l'aplatir. La pyramide c'est fini, il n'y a plus de pointes, ou plutôt si, il y en a une, mais c'est plat.

La base va gérer et nous on a besoin de leads, de personnes qui vont diriger. On parle même de servant-leadership. Tu vas te mettre au service de l'équipe. L'équipe va gérer, va s'engager, s'autoresponsabiliser, va dire ce qui va ou ne va pas. Et quand ça ne va pas, il va te le dire et toi tu t'arranges pour trouver des solutions, en tout cas tu adresses les solutions. Tu soutiens ton équipe. Ça change complètement l'état d'esprit.

Imaginons, le gars, on lui a dit, un jour tu seras le boss d'une société, il y a cru. Il a fait ses études, il est rentré chez Deloitte, il a travaillé comme un acharné. Et puis il arrive dans une boîte à un certain niveau, et on lui dit maintenant c'est fini, et c'est ça qui est compliqué. Ça, je pense, que tant que ça reste dans un monde relativement fermé, du modèle du dev pour du dev et on va délivrer, ça a déjà dû être compliqué. Ils ont dû se heurter à des rôles de project manager, une certaine pression. Ils ont dû déjà probablement vivre ça.

Alors étendre ça, dans une entreprise où le business doit venir en action, où le politique est très puissant dans les entreprises. Le cadre ne dit rien à ce niveau-là.

Je pense qu'il serait nécessaire d'ajouter au cadre, peut-être un nouveau cadre de soutien pour le management. On a suffisamment d'expériences, de terrains. Je pense qu'on peut continuer à avancer comme ça, au niveau de l'agilité. Et je pense qu'il faut aider les gens là-haut. Ils ont

peur, ils ne sont pas à l'aise, ils cassent. Il y a des comportements totalement inadéquats, car les gens ont peur. Ils ont peur pour leur poste, carrière, qu'on ne délivre pas.

Le caractère égotique, il en prend plein la vue, car j'ai eu quelqu'un, au moment où je lui ai dit qu'il allait devenir Product Owner, il m'a répondu : « Qu'est-ce que c'est cette histoire ? ». Il était Product Manager chez ING, juste en dessous du directeur, il se demandait ce qu'il devait mettre sur sa carte de visite. Il ne voulait pas dire qu'il était Product Owner dans un cocktail avec des amis, car cela ne veut rien dire. Il y a vraiment un problème lié à l'ego et à la peur. Ce problème est légitime, donc il faut aider.

M. : Le cadre est fort agnostique, quand on se lance là-dedans, on ne s'en rend pas compte et le directeur qui m'avait mandaté, ne se rendait pas compte de tous les impacts que ça représentait en termes de culture et de mind-set.

Les premières discussions étaient compliquées. Je dis voilà, on va avoir un Product Owner, un gars du business, qui va nous dire ce que moi, je vais devoir faire (au développeur). On va avoir un problème. Et donc il faudrait, en effet aider les organisations. Il y a SAFe par exemple, qui essaye de rattacher la stratégie, le management. Je crois que la dernière Version ONE, c'est le dernier cadre pour l'Agile à grande échelle. Je serai curieux d'en voir une implémentation, car ça doit être lourd.

C. :Ce genre d'approche ne sert pas les grandes entreprises de consultance, de type, je cite Deloitte, mais il n'y a pas qu'eux. Et ce à cause du politique. McKinsey a l'expérience Lean, car il faut revenir à la genèse, le Scrum, l'agilité vient du monde du Lean. Donc, McKinsey a un grand savoir dans le monde du Lean, de savoir comment ça fonctionne et de ce savoir qui a été.

Si vous dites, voilà on travaille dans un modèle Scrum qui hérite du Lean, je peux déjà vous dire que le mot Lean pour certains, il y a un écho, licenciement massif. Ils se sentent du coup déchet, car l'idée c'est de faire, pas du nettoyage en tant que tel, il y a eu l'amalgame, mais on fait ce qui est nécessaire.

*Faudrait-il améliorer l'image de l'agilité pour ces personnes-là ?*

M. :En tout cas, il faut les accompagner, dans le processus du lâcher prise, et ça c'est un travail personnel. On ne se sait pas où on va avec ça. Si on prend 3 personnes qui ont les mêmes soucis, on obtiendra 3 résultats différents à des moments différents. Il y en a qui vont acheter puis revenir au contrôle, il y a un collaborateur avec qui j'ai travaillé pendant 1 an et

deuxième qui est parti. J'ai eu la surprise de voir qu'il est devenu projet manager. C'est peut-être pour faire de l'Agile à sa façon, mais il y a des retours, parce que le contrôle, ça donne l'illusion de résoudre un problème de peur : J'ai peur donc je vais mettre du contrôle, et je veux que ça délivre, donc il faut que je tiens la barre.

*Avez-vous d'autres améliorations en tête ?*

M. : Comme ça non. Ce qui m'aurait bien aidé, c'est un petit guide. Voilà les prérequis, voilà les problèmes rencontrés, de politiques, de communications, connaître les impacts que ça peut avoir sur les gens.

L'aspect retour d'expérience avec toutes les années maintenant qu'il y a de l'Agile, ce serait vraiment pas mal. Il y a sans doute des leçons qu'on pourrait reproduire. Parfois, on n'est surpris de certaines situations, mais elles ont dû être rencontrées des centaines de fois par des gens.

C. : À la fois, comment on adresse les points aux différents rôles et les typologies des personnes, et en même temps, ce qu'on pourrait leur donner comme bouquin pour qu'ils dorment. Bêtement dans le mind-set de certains, Agile c'est bon, on y va, mais ils ne se rendent pas compte de ce que ça représente. Bêtement aller frapper à la porte pour demander du budget, c'est tout un processus et il faudrait le petit guide à suivre quand une entreprise se lance dans ce genre d'aventures.

M. : Ça m'aurait épargné quelques nuits blanches.

*Pour terminer, si je vous demandais une définition sur l'agilité dans le cadre project management, que me diriez-vous ?*

M. : L'agilité, avec le recul, je dirais que c'est un état d'esprit. Dans un cadre project management, c'est plus un cadre de délivré itératif.

*Et pour l'état d'esprit, quelle serait-elle ?*

M. : itératif, mais plus de co-construction, de collaboration plus proche.

C. : Pour moi c'est un cadre de travail qui permet de délivrer de la valeur et la qualité aux clients, d'en récolter les expériences et de permettre l'amélioration du produit tout au long de sa vie.

### Appendix 3 : Interview Anonymous 2

T : project manager in a telecommunication company

*Pourriez-vous vous présenter et présenter votre entreprise en quelques mots ?*

T. : Je suis chef de projet et j'ai de l'expérience en Waterfall, mais aussi en Agile. Cela fait 11 ans que je travaille en tant que chef de projet dans des grandes entreprises comme une grande entreprise de télécommunication. Cela fait maintenant presque 3 ans que j'implémente Scrum. Comme vous savez en fait Agile c'est grand, c'est une collection de différentes méthodologies et Scrum c'est pour une livraison d'une manière rapide.

Alors, j'ai implémenté *Agile/Scrum* dans les petites boîtes, mais aussi dans les grandes boîtes. Maintenant, j'ai constaté que pour éviter l'augmentation des coûts et pour contrôler la période d'un projet il faut qu'on envisage une méthodologie hybride c'est-à-dire, une moitié *Waterfall* et une moitié *Agile*. Je ne parle pas des petites boîtes là tu peux implémenter Scrum sans problème. Dans les grandes boîtes comme une grande entreprise de télécommunication, dans laquelle on va implémenter une transformation, c'est dangereux d'implémenter Scrum en une seule fois. Pourquoi ? Il y a plein de dépendances. Plus il y a de dépendances, moins on sera Agile. Par exemple, ici on a des dépendances avec le release management, on a des dépendances avec les autres projets et les autres équipes de Scrum dans lesquelles il faut obtenir des données.

Voilà notre environnement est très limité et on ne peut pas être vraiment Agile. On fait des développements, on fait de l'*uni testune autre grande banque*, on fait des *testune autres grandes banques*, mais on n'est pas autorisé de faire des *déploiements*, on n'est pas autorisé de mettre des choses en UAT et production. Et souvent on obtient des données qui ne sont pas complètes, qui ont des problèmes avec des « *Quality* » donc ça cause en fait une *rework*, qui est une mise à jour de ce qu'on a déjà développé et cela cause des ralentissements dans la livraison. Et je crois dans les grandes entreprises, on a le *Waterfall* classique et toute la faisabilité puis l'architecture puis l'analyse puis le développement puis le *testune autre grande banque* puis la production puis les garanties.

Il faut implémenter Scrum dans les analyses, les développements et *testune autre grande banques*, mais après un périmètre, un scope qui est vraiment clair et on peut en fait définir un scope pendant l'étude de faisabilité, l'étude architecturale, mais pour analyse on peut aussi implémenter Agile. Le livrable final d'une analyse est soit un document soit plusieurs

documents d'études d'analyse. Donc, ce sont aussi des livrables et on peut aussi les implémenter dans des sprints et surtout après les développements, le test une autre grande banque c'est-à-dire unit test une autre grande banque pour le développement, test de régression, test d'intégration, test de fonctionnalité et tout cela. Mais aussi le développement dans la production cela doit rester dans une équipe d'Agile. Ce n'est pas le cas pour le moment chez Une grande entreprise de télécommunication, je travaillais chez une grande banque ce n'était pas aussi le cas. Donc, c'est-à-dire que dans les grandes boîtes, dans la chaîne des processus, il y a plein de processus, plein de dépendances, plein de complexités ça vaut la peine d'implémenter Agile dans Waterfall. Bon, il y a en fait plein de gens qui travaillent dans Agile et qui ne sont pas d'accord avec moi.

*Et avant d'avoir implémenté scrum, est-ce que vous aviez Waterfall comme méthode ?*

T. : Oui

*Aviez-vous rencontré des problèmes ? Quelles étaient les raisons qui vous ont dirigé vers Scrum ?*

T. : Le problème que je crois qu'on rencontrait en Waterfall c'est qu'il y a une distance entre le business et l'équipe de IT. Donc au début, dans Waterfall, on définit en fait le scope, on définit l'analyse et le business doit attendre jusqu'au moment où on va livrer et ils disent que ce n'est pas la chose qu'ils avaient demandée. Dans Agile, le business est plus impliqué dans le processus. Cela est un avantage. L'inconvénient c'est que pendant le développement ils disent « Ah oui, c'est comme cela, mais est-ce qu'on peut mettre aussi cela au-dessus », et on répond : « mais cela va ralentir en fait notre plannune autre grande banque et cela va coûter ». Ils disent : « oui, oui, oui, mais il faut... » Toutes ces discussions sont coûteuses, prennent beaucoup de temps et voilà. On a besoin d'un Scrum Master qui est assez fort et confiant de dire non au business, ou il va dire : « oui c'est possible, mais on va te dire combien de temps cela va prendre, combien cela va coûter et quel sera le délais de livraison » .

*Est-ce que l'entreprise avait certains objectifs stratégiques ou attentes particulières avant de mettre les méthodes Agiles en place ?*

T. : Tout à fait, car c'est une autre manière de travailler. Quand tu travailles dans une petite boîte avec des personnes plus jeune et dynamique, il n'y a pas de soucis. Quand tu travailles dans une grande boîte dans laquelle il y a en fait des gens qui travaillent dans la boîte depuis

30 ans, ils sont habitués à une façon de travailler, ils ne connaissent pas encore l'Agile, on parle de change management. Agile demande aussi une certaine flexibilité.

Donc pour implémenter Agile il faut en fait le communiquer très bien à l'entreprise, aux gens, voilà ça c'est la méthodologie. Il faut engager des coachs Agiles qui connaissent bien Agile. Il faut aussi être d'accord avec les syndicats. Peut-être eux ils n'aiment pas tous les jours les stand-ups, ils n'aiment pas d'avoir chaque 2 semaines un sprint plannune autre grande banque, peut-être qu'ils n'aiment pas la flexibilité pendant les releases et tout cela. Oui, il faut le communiquer très bien.

*L'entreprise a sûrement communiqué des avantages, des goals pour l'implémentation, quels étaient-ils ?*

T. : Chez Une grande entreprise de télécommunication, ce n'était pas encore communiqué. Je travaillais chez Une grande banque là c'était communiqué, mais le goal c'est en fait à partir de maintenant on va travailler comme cela. Soit tu restes, soit tu pars.

*Donc, si je comprends bien, il n'y avait pas vraiment d'explication pourquoi on changeait de méthode ?*

T. : Non, et il y a aussi un inconvénient c'est-à-dire que dans Waterfall il y a plusieurs hiérarchies. Donc, on a en fait programme manager, plusieurs project managers. Dans les grands projets. Le project manager peut avoir un, deux ou trois junior managers en dessous de lui, et un, deux ou trois project leaders. On travaille avec des team managers. Avant c'était horizontal, maintenant c'est vertical avec des tracks. C'est-à-dire, on a un track leader et un petit peu comme Program Manager, on a des Scrum Masters dans un track, mais en fait middle management, project manager, le team manager il faut qu'ils partent. C'est arrivé déjà chez Une grande banque et ce n'est pas chouette pour les gens. Et en fait il y a plein de connaissances qui partent, qui quittent l'entreprise.

Le Program Manager il s'occupe des choses high level. Ils travaillent avec des Scrum Masters avec des développeurs, il faut être plongé dans les détails des développements et tout cela et ils ne sont pas habitués et oui cela pose des problèmes.

*Cela fait combien de temps que vous avez implémenté l'hybride Agile chez Une grande entreprise de télécommunication ?*

T. : Cela fait un an.

*Et donc comment cela a été mis en place ? Avez-vous donné des formations à tout le monde dans l'entreprise ?*

T. : Oui

*Vous m'avez dit aussi que vous aviez engagé des coachs en Agile ?*

T. : Moi je suis le coach, car j'ai étudié et je connais le processus.

Donc vous êtes devenu coach pour cette implémentation alors ? Y a-t'il eu d'autres coach dans l'entreprise ?

T. : Oui

*Comment avez-vous appliqué la méthode ? Comment l'avez-vous mise en place ? Tout d'un coup ou cela s'est fait petit à petit ?*

T. : Petit à petit, oui. On commence par exemple par daily scrum, après tu commences le sprint plannune autre grande banque. Et après la rétrospective pour évaluer qu'est-ce qu'on peut améliorer, qu'est-ce qu'on peut faire mieux, et puis on commence à calculer la vélocité, donc ça se fait petit à petit. Chez Une grande entreprise de télécommunication, on n'est pas encore 100 % Agile.

*Est-ce qu'au fur et mesure de votre implémentation vous avez dû changer un peu la méthode, l'adapter à votre entreprise ? Vous avez des exemples ?*

T. : Oui, par exemple, on a un Product Owner qui ne savait pas c'était quoi son rôle. Il reste vraiment high level, sur sa tour d'ivoire. Il faudrait lui expliquer qu'à partir d'aujourd'hui tu es le owner du backlog donc tu ne peux pas me demander quelles sont les priorités, c'est ton rôle. Mais au début je l'ai aidé puisqu'il déterminait les priorités. Deuxièmement, pour un bon sprint plannune autre grande banque on a au minimum 2 heures et les gens n'ont pas le temps et on le fait en 1 heure. Polypoint, cela ne dit rien aux gens. Donc on travaille avec Man Days.

*Pendant et après l'implémentation Agile est-ce que vous attendiez certains résultats ? Est-ce que vous avez déjà évalué ces résultats si c'était à hauteur de vos espérances ?*

T. : Oui et non. Oui, mais pas vraiment sur papier. C'étaient surtout des feedback, des retours des gens, mais il y a plein de gens qui disent « Ah oui maintenant c'est mieux organisé, c'est

bien organisé , ça va plus vite ». Par exemple le Burn-down chart on a en fait des user story qu'on doit postposer au dernier sprint parce qu'il nous arrive en fait plein de choses : soit ils sont sous-estimés, soit il y a d'autres projets qui interviennent, soit les gens tombent malade voilà.

*Est-ce que vous évaluez aussi vos projets en eux-mêmes ? Comment déterminez-vous si un projet est un succès à la fin ?*

T. : Il y a plein de facteurs à considérer. Le facteur principal c'est la livraison du produit final. Mais à côté de cela on prend en compte la motivation de l'équipe, l'amélioration de la relation du business et IT, amélioration de la relation avec le management, le fait que l'on peut s'organiser, ce sont les facteurs d'un succès d'un projet. Ce n'est pas seulement le produit final. L'aspect people management est aussi important.

*Avez-vous vu une amélioration dans ces facteurs depuis l'implémentation de Scrum ?*

T. : Oui, tout à fait.

*Et quel a été l'impact général de l'implémentation sur l'entreprise ?*

T. : Des petits succès, c'est une grande boîte, mais dans mon environnement les gens sont plus motivés, le business est plus satisfait, on livre plus rapidement, il y a une vue claire sur la production.

*Et selon vous, est-ce que vous apporteriez encore quelques améliorations à la méthode mise en place aujourd'hui ?*

T. : Oui, alors comme je le disais en Agile il faut définir le scope dès le début. L'Agile cela sert à répondre au changement, le changement cela coûte, car il faut implémenter un change request. Il faut aussi payer et ça c'est l'impact. A côté de cela il faut, en Agile on parle de impediments ce sont les problèmes qui nous arrivent. Il ne faut pas attendre jusqu'à ce que les impediments, les problèmes arrivent. Il faut aussi déterminer les risques à l'avance et mitiger les risques. C'est plutôt Waterfall, mais c'est comme cela. Aussi les rôles et les responsabilités sont importants. Dans la théorie d'Agile, un développeur peut tester, un testeur peut développer. Quelqu'un qui s'occupe avec des déploiements peut aussi écrire l'analyse, mais ce n'est pas comme cela. En fait cela ne sera jamais comme cela. En fait tu peux peut-être travailler sur le long terme sur une équipe plus homogène, mais en réalité ce n'est pas

comme cela. Il y a un contrat dans l'équipe parce que chacun a sa propre responsabilité. Donc il faut bien déterminer qui est responsable de quoi.

*Et selon votre expérience avec Agile, quel est votre avis par rapport à ces méthodes Agile ?*

T. : Mon avis ? Comme je disais Agile, tu peux implémenter Agile dans une petite boîte avec des gens qui sont jeunes et dynamiques ça marche parfaitement. Dans une grande boîte comme ici avec plein de processus, avec des gens qui travaillent déjà depuis 30 ans, avec des gens qui n'aiment pas le changement en fait, ce n'est pas très évident. Donc il faut l'implémenter dans Waterfall. Agile cela doit couvrir l'analyse, le développement, le test une autre grande banque et le déploiement. Mais il faut avoir une étude de faisabilité, il faut avoir une étude architecturale et de ce fait le projet il arrive dans la production et après une période de garantie oui on se dirige vers l'Agile.

*Si vous deviez donner une définition sur les méthodes Agiles ce serait quoi selon votre expérience ?*

T. : Agile ou Scrum c'est une façon de nous organiser efficacement et aussi de livrer plus rapidement.

*Connaissez-vous l'Agile Manifesto ?*

T. : Oui.

*Quel rôle cela joue-t'il dans votre entreprise ? Est-ce que tout le monde suit ces principes ?*

T. : Il y a plein de choses par exemple : deliver grande banque software instead of documentation, et il y a plein de documentation et on ne peut pas éviter la documentation. Ce n'est pas le cas chez Une grande entreprise de télécommunication. Ou personnel interaction instead of ... On s'adresse oralement en face à face. C'est dur d'implémenter agilité quand on outsource. On travaille en fait avec les indiens. En Inde, il y a une différence de temps. Maintenant en Inde il est 4H30 par exemple je ne peux pas organiser une réunion à 3h ici parce que cela sera déjà 6h30 là. Ou bien il y a des gens qui travaillent chez soi. Ou bien avec les technologies comme Link ou Skype for business c'est presque possible. Mais bon en agilité on utilise les post-it et pour cela il y a des workshops, des ateliers donc tu as besoin d'une caméra pour visualiser, pour enregistrer tout cela quand tu mets des post-it sur les murs.

Du coup on utilise Gira, Excel comme si les gens ils étaient là. Agilité en fait c'est interaction, proactivité, des dynamiques dans des workshops voilà.

*Pensez-vous que si plus de personnes connaissaient l'Agile Manifesto il y aurait moins de problèmes chez Une grande entreprise de télécommunication ?*

T. : Non, on a des gens ici, qui ont peur des changements. Le changement c'est l'agilité ou quelque chose d'autre. On a des gens ici qui travaillent déjà dans cette boîte pendant des années. Ils ont vu plein d'évolutions, de révolutions et ils aiment en fait la stabilité. Ils ont en fait implémenté une façon de travailler et ils sont habitués de travailler dans leur coin. Ce n'est pas quelque chose que tu peux changer. S'ils lisent « Agile Manifesto », ils vont dire oui c'est chouette, et après 5 min, OK on travaille comme avant. Ils se trouvent dans leur zone de confort. Mais seulement en lisant « Agile manifesto », tu ne peux pas les changer. Donc ce n'est pas évident. Et dire à partir d'aujourd'hui on va travailler Agile, c'est quelque chose qui chez UNE AUTRE GRANDE BANQUE est propice, on implémente cela. A partir de maintenant on travaille Agile. En fait, ils ont viré plein de gens avec plein de connaissances qui ne voulaient pas s'adapter, mais maintenant ils ont des problèmes parce qu'ils ont pensé OK on va virer des personnes qui sont chères et on va engager des jeunes personnes qui sont meilleurs marchés et cela va rouler avec l'agilité. Mais ce n'est pas comme cela. Quelqu'un qui est junior il doit en fait apprendre le processus, les systèmes, les gens, il faut qu'il fasse des connaissances avec les gens. Cela prend du temps. Non, l'agilité c'est bien de l'implémenter, mais bon le monde ne changera pas parce que tu implémentes l'agilité.

### Appendix 4 : Interview Anonymous 3

M : R&D Director at an Intranet applications Company

J : Senior Developer at an Intranet applications Company

S : Developer at an Intranet applications Company

*Pouvez-vous vous présenter en quelques mots ?*

M : Je suis responsable de la R&D.

S : Je suis développeur ici depuis presque 2 ans. J'ai déjà eu un peu d'expérience avec Scrum dans le cadre de mon stage. C'est mon premier job.

J : Je suis développeur ici depuis 6 ans. On a eu quelques expériences avec scrum, mais aussi des méthodes un peu plus empiriques et aujourd'hui on peut dire qu'on a une démarche Agile.

*Pouvez-vous présenter votre entreprise ?*

M : Nous sommes 3 employés et un patron. Essentiellement, nous faisons de la consultance, pour ce qu'on appelle le domaine de la qualité. Nous travaillons dans des secteurs variés comme dans l'industrie ou les hôpitaux. On a une manière assez particulière de travailler qui implique que l'on fasse plus de R&D que de consultance chez nos clients.

Il y a beaucoup de changement en ce moment, nous allons d'ailleurs bientôt créer une « spin off », ce qui revient à séparer la société en deux structures. La nouvelle structure sera centrée sur une plateforme que nous avons développée dans le cadre de nos projets. Celle-ci permet de concevoir plus efficacement les applications dans un cadre Agile et nous souhaitons que les 2 activités (de consultance et d'éditions de logiciel) soient séparées. Les développeurs seront dans la nouvelle structure d'édition de logiciel, tandis que notre patron va développer la partie consultance. Dans les deux cas, nous allons développer des partenariats afin d'étendre nos activités.

*Vous êtes tous les trois envoyés dans des entreprises ?*

M : Nous nous rendons ponctuellement dans les entreprises, mais nous ne faisons pas de body shopping. Toutes les sociétés de consultance avec lesquelles nous parlons, - car on essaie de les démarcher pour qu'ils utilisent notre plateforme - font beaucoup cela. À l'opposé, nous

privilégions le télétravail, car celui-ci s'est révélé plus efficace et que nos clients ne nous imposent pas le body shopping.

J : Nous ne travaillons que sur des projets dans lesquels nous avons le lead, rarement en renfort d'équipes existantes.

M : En effet, on s'occupe de tous les cycles de développement, depuis l'analyse des besoins au déploiement avec leur informatique.

*Et vous vous occupez de tout à 3 ou par personne ?*

M : Non, les responsabilités sont réparties un peu différemment. J'interviens en coordination (soutien/analyse) dans tous les projets et chaque client a son développeur attitré. Tout le monde travaille sur la plateforme.

*Cela fait combien de temps que l'entreprise existe ?*

M : Depuis 2011.

*Avez-vous utilisé les méthodes Agiles depuis le début ?*

M : Nous sommes assez particuliers à ce sujet et c'est pour cela qu'il est intéressant de discuter avec vous aujourd'hui. Nous n'avons pas suivi les préceptes Agiles à la lettre, mais nous travaillons clairement de manière Agile, car nous n'entamons pas nos projets clients sur base de gros cahiers des charges. Nous travaillons plutôt par cycles. Nos cycles de développement sont d'ailleurs devenus de plus en plus petits. Pourquoi fait-on cela ? Parce que nous avons constaté que nos clients exprimaient un besoin au départ, mais que ce besoin n'était pas le besoin réel final. Faire des cycles permet d'affiner le besoin que le client ressent quand il teste les prototypes successifs. Donc c'est important de faire cela par petites étapes, le résultat est bien meilleur. Nous considérons cette méthode comme Agile spécifiquement par ce fonctionnement par cycle.

*Donc vous avez directement mis en place votre propre méthodologie, on peut considérer cela comme ça ?*

M : Il s'agit plutôt d'une philosophie de travail qui a émergé.

J : Puisque nous sommes très souvent à trois pour prendre les décisions concernant un projet, nous avons choisi au fur et à mesure des méthodes correspondant à notre manière naturelle de collaborer, plutôt que de les sélectionner a priori.

Aujourd'hui comme dit M on est en très petits sprints et on commence à adopter Kanban. Nous avons essayé et cela colle très bien à nos besoins.

*Avez-vous pris des idées dans chaque méthodologie, par exemple de Scrum, ... ?*

M : Il y a quelques années, nous avons appliqué scrum pendant un peu moins de 6 mois, avec un Scrum Master. C'était positif, car c'était structurant. Ce système nous permettait de cadrer les projets, mais ça s'est arrêté, car c'était justement trop structurant. Pas pour nous, mais pour les patrons qui eux ne pouvaient plus nous rajouter, au milieu des sprints, des demandes des clients et donc nous étions paradoxalement moins Agiles. Nous, développeurs, étions plus sereins parce qu'on savait ce qu'on avait dans les sprints, mais cela ne permettait plus d'être assez Agile. Ce n'est pas que cela posait des problèmes avec les clients, mais le but c'est qu'il y ait toujours une satisfaction maximale. Pour cela, nous avons arrêté le Scrum. Encore une chose à propos du Scrum, une difficulté était de déterminer les bonnes estimations pour la réalisation des tâches. Dans une boîte classique de consultance, c'est facile, car les tâches finissent par se ressembler et donc la précision des estimations s'affine avec le temps. Nos développements s'articulent davantage avec de la R&D plateforme. Lorsque nous rencontrons un type de besoin trois ou quatre fois chez des clients, il est généralisé (ou implémenté) dans la plateforme, pour qu'il n'y ait plus qu'à cocher une case ou à faire un tout petit code pour reproduire la fonctionnalité. En conséquence, une grande part de notre travail se trouve dans la R&D. Ce n'est, par nature, pas facilement estimable justement parce qu'on ne sait pas généraliser l'estimation d'implémentations si diverses et dont le but est de ne plus avoir à effectuer des implémentations du même type par la suite. Cela rendait donc le Scrum compliqué.

*Pourquoi avez-vous choisi Scrum au début ?*

M : C'était une décision de notre patron de l'époque.

*Après, avez-vous gardé certaines pratiques de Scrum pour continuer ou avez-vous complètement arrêté Scrum ?*

M : Nous continuons de fonctionner par cycle. Nous tentons également de ne pas avoir des cycles interrompus, par exemple: travailler deux jours sur un projet puis s'interrompre pour passer sur un autre devenu subitement plus urgent, pour quelques jours. On essaie toujours de caler 1 ou 2 semaines où on peut se concentrer sur quelque chose, sur un client ou sur la plateforme. Intercaler des petites phases de développement est inefficace parce que le switch demande beaucoup d'énergie. Cela fait perdre du temps et de la concentration. Comme nous l'avons expliqué, ce n'est pas toujours possible si on souhaite maximiser la satisfaction du client, mais nous gardons toujours cela à l'esprit dans notre organisation.

*Vous avez parlé de Kanban aussi, utilisez-vous la méthode dans son entièreté ou juste certaines pratiques ?*

J : Probablement pas dans son entièreté.

M : Nous utilisons un board (Trello) pour les tâches « Todo » et leur suivi, je pense que Kanban est plus large, une méthodologie complète.

*Et vous utilisez juste Trello comme outil ou vous utilisez d'autres outils également ?*

M : Depuis le début, nous avons utilisé de nombreux différents outils, dont certains ont été conçus en interne. Mais la plupart n'ont pas pris. Trello a cet avantage de permettre un encodage rapide et presque ludique. C'est psychologique à mon avis, pour encoder des tâches et faire le suivi, mettre à jour les statuts et avoir envie que cela avance. Ils ont réussi à faire une interface entraînante. En complément, nous utilisons un outil interne pour l'encodage de nos temps de travail. C'est essentiel pour notre organisation en télétravail, mais cela permet aussi de quantifier nos coûts R&D et de préparer la facturation.

J : Ce n'est pas un coût de gestion de faire une carte. Cela prend 2 secondes et cela se fait pour vous. Il s'agit plutôt de notre fonctionnement interne propre. Habituellement un développeur ne doit pas faire ses cartes.

M : Comme nous travaillons beaucoup en auto-gestion, nous avons effectivement la difficulté supplémentaire de nous auto-discipliner et de devoir prendre une grande part des décisions, ce qui signifie que nous sommes responsables de nos « cartes » de la création jusqu'au déploiement. C'est une adaptation que nous avons pu mettre en place récemment suite au changement de gérance.

*Quand vous rendez vous compte que vous devez changer votre méthodologie ou d'outil ?*

J : Nous l'estimons principalement à la charge ressentie dans la réalisation. Nous concluons généralement nos projets quelle que soit la méthodologie de management. Cependant, la charge ressentie et le stress éventuel sont grandement dépendants de celle-ci.

Si la conclusion d'un projet engendre une trop grande charge, par une mauvaise organisation, ou une communication peu claire avec le client, nous remettons en question la méthodologie, parmi d'autres facteurs.

M : L'origine de ces surcharges est généralement un manque de contrôle du client. Quand nos managers étaient nos contacts avec nos clients, ce qui n'est plus vraiment le cas maintenant, ils disaient : « Oui, oui » à toutes les demandes et on répondait : « Non, ce n'était pas prévu ». Au contact avec le client, celui-ci souhaite tout de suite savoir quand la mise à jour sera disponible. Si son contact n'a pas d'assez bonnes connaissances, il répondra ce qui fera plaisir au client, et cela nous met une pression inutile derrière. Le fait d'être, en tant que développeur, en communication directe avec le client, rassure tout le monde parce qu'on peut tout de suite lui fournir des réponses assez précises. Cela nous responsabilise par la même occasion et le client apprécie clairement avoir un contact direct avec le métier. Cela fluidifie tous les échanges, analyses, et l'organisation en général.

*Et le contact direct avec le client c'est vous qui l'avez demandé au manager ?*

M : Non c'est un changement qui s'est fait naturellement, en accord avec les clients.

*Au niveau communication, cela s'est-il amélioré d'avoir utilisé des méthodes Agiles ?*

M : Cela nous a permis d'obtenir de bien meilleurs résultats chez le client. Travailler par cycle et échanger avec le client à chaque étape permet au besoin de s'affiner en même temps que l'application et cela engendre une bien meilleure adoption par les utilisateurs. En post-projet, des adaptations nous seront naturellement demandées et pourront être réalisées dans un temps court, puisque c'est dans la continuité du projet. La communication est d'autant plus fluide qu'une relation de confiance se tisse, parce que les résultats sont là.

Nos clients nous rapportent les échecs de leurs précédents projets informatiques. On constate que l'absence de cycles (de développement et/ou d'échange) cause un problème majeur d'adoption des applications, puisqu'inadaptées, quand cela n'entraîne pas un abandon pur et

simple. Cela explique la méfiance que l'on rencontre dans les premiers rendez-vous. Après quelques cycles et les résultats positifs, ils montrent leur satisfaction des promesses tenues, ce qui reste pour eux assez inhabituel (malheureusement). Pour nous, c'est gratifiant.

J : Tout ce que tu dis est vrai, mais je ne sais pas si on peut tout imputer à la méthodologie. Il y a aussi le fait que nous ne soyons pas des commerciaux, mais des développeurs. Il y a un aspect très direct dans notre approche du client. Nous parlons rapidement de la finalité opérationnelle. Je pense que cela amène aussi la confiance. Nous ne pourrions cependant pas le faire sans une méthodologie Agile.

*Et le fait que vous êtes fort indépendants dans votre travail, est-ce que c'est cela qui va faire que vous pouvez adapter votre méthodologie comme vous voulez ?*

M : C'est certainement le cas, puisqu'il y a moins de personnes à convaincre.

*Quels types de projets vous occupez-vous essentiellement ?*

M : Des applications intranet que nous implémentons avec notre plateforme. Celle-ci nous permet d'être beaucoup plus Agiles parce qu'elle généralise tout ce qu'on doit usuellement faire dans ce type de projets. La généralisation des fonctionnalités rend leur implémentation très rapide et flexible, ce qui facilite le développement par cycle et les clients obtiennent les changements demandés en un temps record.

*Est-ce que je peux vous demander ce que vous voyez comme différence par rapport à votre stage par rapport à la méthodologie Scrum et ce que vous utilisez maintenant ?*

S : Scrum nous imposait une périodicité et des fenêtres de temps strictes pour les réunions. Ici, nous adaptons ces paramètres en fonction de la situation. Je pense que c'est la principale différence. Nous n'avons pas vraiment de personne désignée "Product Owner" ou "Scrum Master". En revanche, notre méthode de planification et suivi de tâches de développement est assez similaire.

*Est-ce que vous faites des réunions systématiques tous les deux jours ? Ou toutes les semaines ?*

M : Nous faisons des réunions régulièrement, oui, mais pas à horaire fixe. En fait, les réunions suivent le déroulement des cycles de développement, dont la durée varie selon les cas. Nous avons d'abord une réunion avec le client, qui inclut une review des précédents

développements, une analyse des besoins et une récolte des demandes. Nous avons ensuite une réunion de débriefing entre nous pour une analyse plus technique et le planning. Des réunions internes interviendront ensuite jusqu'à l'installation de la mise à jour, en fonction des péripéties de l'implémentation.

*Est-ce que vous avez certaines attentes, certains objectifs stratégiques par rapport à votre méthodologie que vous utilisez ?*

M : Nous n'avons pas d'objectif stratégique par rapport à notre méthodologie. L'objectif est plutôt que le client soit content tout en évitant tout stress dans l'équipe.

*Et éventuellement vis-à-vis de votre client ?*

M : Au-delà de sa satisfaction, il s'agit d'optimiser les contacts. Certains clients cherchent un contact quasi quotidien ce qui nuit à l'efficacité de développement. Il faut donc parvenir à rassembler les appels. Notre fonctionnement vise donc à structurer les contacts.

J : C'est ce que je suis en train d'essayer de mettre en place chez mon client. L'objectif est de mener chaque semaine un sprint meeting afin de garder l'approche par sprint court.

M : Nous souhaitons limiter la dispersion des contacts. Ce n'est pas nécessaire pour tous les clients. Mais globalement il faut équilibrer ces contacts.

*Évaluez-vous votre méthodologie d'un point de vue théorique ?*

M : Non, il nous arrive d'en discuter évidemment, mais pas d'un point de vue théorique.

*Si maintenant vous avez fait un changement d'aller vers Kanban par exemple, est-ce que vous évaluez l'avant/après changement ? Ou par rapport à votre utilisation de Scrum ?*

M : On en discute entre nous, mais pas formellement. On ne fait pas de réunions ou d'analyse approfondie autour de ça.

J : Je vois que, d'une manière générale, ça ne contraint pas ce qu'on réalise en termes informatiques. C'est principalement de la communication. On peut finalement mettre n'importe quoi dans un sprint meeting ou un kanban. Aussi, l'aspect connaissance du métier est de loin le plus important, pour éviter tout écueil de management. La méthodologie, sur un effectif si peu nombreux, n'amène pas de grands gains ou pertes de productivité.

*Comment déterminez-vous qu'un projet est un succès ?*

M : Si le client est satisfait (et qu'il le montre), si le projet prend de l'ampleur par rapport au besoin initial, mais aussi au retour des utilisateurs, qui voient leur charge de travail diminuée et leur performance augmentée. Ces indicateurs nous montrent aujourd'hui que tous nos projets ont réussi et cela valide notre méthode et notre technologie.

*Et au fur et à mesure de l'adaptation de votre méthodologie propre, vous avez vu ce succès augmenter, s'améliorer ? Est-ce que c'était plus facile peut-être d'arriver au succès ?*

J : Oui, mais la plateforme a tellement évolué, plus que notre méthodologie. Finalement, c'était grâce à l'outil.

M : Notre méthodologie a mûri avec nos expériences, mais c'est bien notre plateforme qui nous a permis d'être plus Agiles et d'obtenir de meilleurs résultats. Grâce à cela, nous avons pu davantage suivre notre philosophie de travail en étant moins perturbés par les problèmes d'implémentation ou les difficultés analytiques. Nous souhaitons faire passer ce message : la méthodologie Agile est très importante, mais elle doit être supportée par les outils adaptés. Des équipes de développement peuvent être très Agiles et performantes au sens organisationnel, mais elles ne pourront exploiter cette force sans être Agiles dans leur implémentation, à cause d'outils pas suffisamment flexibles voire d'une absence d'outils, les contraignant à un important supplément de conception. Il faut voir les deux côtés de la pièce.

En plus de son évolution, notre maîtrise de la plateforme et de ses possibilités nous confère une excellente capacité analytique lors de nos réunions avec le client, ce qui nous permet de l'orienter dans la bonne direction. Un problème peut souvent être résolu de diverses façons, mais on passe trop facilement à côté d'une façon plus adaptée, et en fait plus simple à implémenter, de le résoudre. Se lancer dans la mauvaise direction peut causer des difficultés d'implémentation et impacter négativement le résultat. C'est pour cela que ces réunions clients doivent être menées avec une maîtrise technologique.

*~~Selon votre opinion, est-ce que pour vous il faut avoir un certain objectif en tête pour pouvoir utiliser ou faire des modifications dans votre méthodologie Agile ou pas ? Objectif par rapport au projet, par rapport à la stratégie, par rapport au client, est-ce qu'il faut avoir un objectif bien déterminé ?~~*

J : ~~C'est une question de communication. Oui il y a toujours un objectif long terme on sait plus ou moins les applications que l'on peut faire, mais cet objectif peut changer parce qu'une application ne va finalement paraître moins importante. Et il y a un truc dont on a parlé et qui~~

~~finale~~ment est plus important, mais de voir plus ou moins le package final qu'ils veulent et comment cela va s'imbriquer, comment tout cela va se connecter et quels sont les priorités d'implémentation et puis par petits cycles. Oui il faut avoir une sorte de mini cahier des charges pour savoir exactement à quoi le client s'attend quand on a fini le développement avant de le voir la fois suivante.

*Est-ce que vous apporteriez encore des améliorations sur le système mis en place pour l'instant ?*

M : Actuellement non, car il est efficace et nos résultats sont excellents. Notre équipe étant de taille limitée, nous avons une bonne connaissance du travail de chacun. Dans cette situation, il faut au contraire éviter de surcadrer notre travail. Cela a eu des effets négatifs, en termes de flexibilité, à chaque fois que nous avons essayé.

Nous avons cependant la volonté de croître à travers notre activité d'éditeur de logiciel. Agrandir l'équipe impliquera de faire évoluer notre système organisationnel. Un cadrage minimal sera obligatoire pour garder une bonne visibilité et assurer la qualité de nos développements. Notre défi sera de trouver le bon équilibre.

J : On a la chance de pouvoir changer la méthodologie au besoin de manière concertée. Moi je peux communiquer avec mes clients via Trello. Ils ont un suivi de ce qui est réalisé et je n'ai pas besoin de plus, on est un effectif peu nombreux et même vis-à-vis de notre client, il n'y a qu'un point de contact donc il n'y a pas besoin de surorganiser. Quand on grandira, il faudra évoluer, mais pour l'instant je me sens bien comme cela, tant qu'on ne fait pas le Lean.

*Pourquoi ?*

J : Le Lean ne me semble pas adapté aux projets ayant de hautes exigences de qualité, de par la contradiction qui émerge de la mise en place rigoureuse de ses préceptes.

Lean prétend que tout ce qui n'a pas une valeur ajoutée directe pour le client est un déchet, ce qui pousse généralement les équipes à développer des quick-win rapidement et facilement accumuler de la dette technique.

Or, il invite également à repousser au plus tard les « décisions importantes », afin de promouvoir une souplesse architecturale devant permettre d'être plus robuste face aux imprévus. Vous êtes donc presque « invités » à développer vos applications sur des bases qui

se veulent mouvantes ou hypergénériques. Ceci se traduit soit par des réécritures fréquentes, soit par le besoin d'une architecture trop complexe, affectant la productivité globale.

Ces principes contredisent, à long terme, la notion de « Fast delivery » qui est vue comme un objectif et facteur de succès de cette approche.

Nous pensons a contrario que les décisions importantes doivent être prises et assumées tôt dans l'analyse, tout en laissant de la souplesse architecturale autour des inconnues qui demeurent. De manière générale, nous sommes assez critiques face aux méthodes qui prétendent contraindre la méthode de développement à des fins commerciales. Ce type de gestion de projet produit souvent un enlisement assez rapide, par l'accumulation de dette technique et le peu de qualité produit.

Nous sommes une entreprise de développement informatique, notre méthodologie doit soutenir le développement afin d'augmenter sa qualité, et pas le contraindre pour tenter de vendre plus. Notre approche met en avant la pérennité et les résultats à long terme. Nous nous félicitons d'avoir des collaborations interrompues depuis plus de 12 ans avec certains clients.

M : Notre premier client est toujours là parce que nos applications tiennent la route sur le long terme et parce que quand il y a du refactoring à faire, on le fait aisément. Quand ils demandent des modifications, quand ils demandent de migrer les applications vers de nouvelles versions ou plateformes, ça ne pose aucun problème, car tout est resté à jour et propre.

*Quel est votre avis de manière générale sur les méthodes Agile ? Cela peut être Lean, Scrum, eXtreme Programming... enfin toutes les méthodes que vous connaissez. Quel est votre avis en général ?*

M : Je crois que J l'a bien résumé. Nous apprécions la philosophie plus que les théories appliquées, qui peuvent être contraignantes et inefficaces sur le long terme. Mais la philosophie est géniale et donne d'excellents résultats, surtout comparé à Waterfall. Tout le monde devrait travailler de cette façon, par petits cycles. C'est une évidence. Maintenant, d'un client à l'autre, d'un projet à l'autre, d'une équipe à l'autre, il y aura un besoin différent dans l'organisation du travail donc imposer une manière de travailler via l'une de ces méthodes c'est non optimal.

*Connaissez-vous l'Agile Manifesto ?*

M : Nous l'avons lu.

*Est-il très présent dans votre entreprise ?*

M : Il n'est pas accroché au mur et nous ne le connaissons pas par cœur. Autant dans son esprit que dans ce que les méthodes qui en ont découlé, nous prenons et intégrons ce qui nous intéresse. Nous gardons ce qui fonctionne et abandonnons ce qui est contre-productif. Nous ne voulons surtout pas rester dans un moule, car c'est définitivement non optimal.

*Si vous deviez faire une conclusion sur tout ce que vous avez accompli, au point de vue structure d'implémentation, quelle serait la conclusion ?*

M : À chaque fois que nous avons voulu (trop) structurer, nous avons perdu en flexibilité. Il faut être Agile dans notre agilité : adapter notre fonctionnement à chaque nouveau projet et client, tout en continuant à travailler par cycle et sans imposer de fortes contraintes. Trop structurer ajoute de la lourdeur au développement et crée un risque en cas de pression du client ou d'absence des managers.

J : On estime beaucoup les méthodologies de manière théorique, mais on oublie que les acteurs de ces diverses méthodologies sont plus que primordiaux. Par exemple, des managers sans connaissances techniques vont rapidement trouver leurs limites, ou se cantonner au rôle de secrétaire de réunion, et ce quelle que soit la méthodologie. Il est important que l'espace d'interaction avec le management soit assez riche pour aborder au besoin les réalités techniques. Le management doit avoir une compréhension fonctionnelle du travail réellement accompli, afin de promouvoir une méthodologie qui fait sens dans cette situation.

Dans le cas contraire, toute tentative d'amélioration de la méthodologie ne sera que la mise en pratique hasardeuse d'un principe séduisant sur papier. De plus, le risque est important de désolidariser la prise de décision de la vraie maîtrise fonctionnelle, ce qui a en général des effets catastrophiques sur un projet.

Nous mettons donc l'accent sur la qualité des acteurs, et leur maîtrise technique, plus que sur l'amélioration méthodologique. À notre échelle, celle-ci reste un phénomène qui émerge de lui-même et n'a pas besoin d'être structuré formellement.

*Pensez-vous qu'il faut laisser les méthodologies souples, de modifier les méthodologies ?*

M : Certainement. Chaque équipe est différente (mission, profils, etc) et doit pouvoir s'organiser différemment selon ses propres expériences. Pour ce qui est de la communication vient un certain point où il est nécessaire de la standardiser pour éviter toute confusion.

*Faudrait-il que la sélection, modification de méthodologies se fassent bottom-up alors ?*

J : Tout à fait.

M : Sinon c'est déconnecté de la réalité.

*Avez-vous eu une expérience dans le lean ?*

M : Notre précédent manager était adepte du lean. Le « quick and dirty » primait clairement sur la qualité de l'implémentation.

J : Oui lean, c'est « quick and dirty ». Ce n'était peut-être pas le meilleur ambassadeur du lean. Cependant, la situation illustre bien notre propos précédent : la prise sur la réalité technique est primordiale.

Dans ce cas, nos arguments techniques contre certaines approches proposées étaient souvent disqualifiés par le simple fait d'énoncer « quick-win » ou la méthode lean en général.

La conjonction du 'Lean' et de l'absence de maîtrise technique par les managers a permis une accumulation de dettes techniques. J'ai, par exemple, récemment passé 2 mois à corriger une application remplie de « quick win ».

M : Je pense que le lean va de pair avec le body shopping où le développeur est placé chez un client, réalise un travail précis, est payé en régie et puis s'en va. Son travail n'est pas de sa responsabilité sur le long terme. Ce n'est pas lui qui paye la dette technique.

Nous ne travaillons pas de cette manière. Nous concevons notre plateforme de façon à réutiliser au maximum nos fonctionnalités en les généralisant. Mais nous réutilisons également les applications chez nos clients successifs. Nous avons par exemple une application pour suivre et gérer les compétences et l'apprentissage des compétences pour un client. Un autre client nous l'a demandée. Nous allons reprendre la même application en l'adaptant. De client en client, nous l'enrichissons, nous l'adaptons. Si ce n'est fait pas correctement, nous hériterions de cette dette technique. C'est un investissement au départ,

mais qui paie sur le long terme. En conservant un bon niveau qualitatif, nous pouvons offrir des solutions entièrement sur mesure à nos clients, en gardant une excellente rentabilité.

*Quelles conclusions pouvez-vous tirer sur les méthodes Agiles ? (À S.)*

S : J'ai pu constater qu'une méthode comme Scrum pouvait sembler plus rassurante, car elle offrait un planning prévisible et un sentiment de contrôle. Je suppose que Scrum facilite également l'intégration de nouveaux membres à une équipe existante puisque cela réduit la difficulté à s'adapter à méthode de travail différente. Mais une méthode comme Scrum me paraît trop rigide pour nos besoins particuliers. Nous devons être capables de nous adapter rapidement et quotidiennement à des imprévus avec une équipe réduite, ce qui semble difficile lorsqu'on suit strictement une telle méthode.

*Avez-vous encore des questions, des remarques ou des conseils ?*

M : Pour résumer, nous voyons la vraie l'agilité comme une philosophie plutôt qu'une méthode, mais elle ne peut être appliquée sans l'outil de développement adéquat. On considère souvent l'outil comme un élément périphérique, mais il est tout aussi important. Grâce à cela, nous réussissons tous nos projets clients en temps et en heure, avec des clients qui ont toujours le sourire et, dans ce type de projet (informatique), ce n'est pas vraiment une constante.

*C'est donc surtout l'outil qui aide beaucoup ?*

M : Il ne faut pas une méthodologie qui met des contraintes, il faut une philosophie qui nous permette de penser et de travailler en étant Agiles, mais également un outil qui nous permette d'exploiter ce potentiel au maximum.

## Appendix 5 : Interview Anonymous 4

L. C. : « Le Coach », Agile coach at a bank company.

*Pouvez-vous vous présenter en quelques mots ?*

L.C. : Je m'appelle Le Coach, je suis un freelance donc je n'ai pas vraiment une entreprise en tant que telle. Aujourd'hui, j'ai une fonction de coach Agile. Donc, j'aide les entreprises à transformer les gens. Parce que la transformation Agile, c'est la transformation de comportements et de mind-sets, au-delà des méthodes pures. Comment je suis arrivé à faire ça ? j'ai à peu près 21 ans de métier et je me suis toujours intéressé à: "comment améliorer la méthode de travail, comment améliorer les processus informatiques, comment est-ce qu'on peut automatiser, utiliser les choses et supprimer les choses superflues". Enfin voilà, l'efficacité, mais pas par rapport à la rentabilité, ce n'est pas mon souci premier. Ce n'est pas une question de performance ou de rentabilité-, mais on fait beaucoup de choses systématiques, qu'on répète tout simplement, les mêmes gestes, les mêmes choses, et on ne sait pas vraiment pourquoi on le fait. Et ça je trouve absurde.

Il faut qu'on fasse les choses pour de bonnes raisons, il faut que ce soit efficace, ça ne sert à rien de faire quelques choses pendant 3 jours alors qu'on pourrait le faire en un. Ça n'a pas de sens, et quand ça n'a pas de sens ça m'énerve. J'ai une formation d'ingénieur industriel, je suis électronicien de base, et j'avais pris la technique digitale parce que l'informatique est un dada depuis que je suis adolescent. Et donc, dès mon premier travail, j'ai essayé de chercher éliminer le superflu, comment éliminer les choses rébarbatives et répétitives. Il faut automatiser, on n'est pas là pour s'embêter, et c'est un peu le côté "ingénieur", trouver des solutions pour faciliter et pour améliorer. Donc je me suis orienté dans ce secteur-là. Et de fil en aiguille, de mission en mission, d'abord employé puis indépendant. Le but premier, c'était d'aider les gens pour améliorer le quotidien par rapport au travail. Car si ça améliore le quotidien, ça améliore la performance, les coûts, et les gens sont plus heureux, travaillent mieux et sont plus motivés. Quand on veut faire ça, on tombe inévitablement dans l'agilité à un moment donné, tout simplement.

*Vous travaillez actuellement dans une grande entreprise bancaire, et le fait d'être Agile coach de quoi vous occupez vous dans l'organisation ? De la structure, de la communication en particulier ?*

L.C. : C'est vraiment du coaching. Dans "Agile coach", il y a "coach" et en l'occurrence la transformation chez une grande entreprise bancaire (on parle de 2000 personnes en IT, rien que l'IT) donc ça fait des départements énormes avec plein de groupes. Donc cette transformation n'est pas l'affaire d'un coach, on était toute une armée, on était plus de 100.

*C'est un travail que vous faites tous ensemble ?*

L.C. : Réparties sur les 2000 personnes, il y a une vingtaine de tribes. Dans une tribu, il y a 7 ou 15 squads, chaque squad, c'est une équipe, entre 8 et 10 personnes. Certaines sont plus grandes que ça. Il y a des squads de 12-13 personnes (ça c'est un peu trop). Donc voilà, c'est une très grosse transformation, il y a eu beaucoup d'argent investi dans ce changement-là, et ça fait un an que ça roule, que c'est en train de se déployer.

*Ce changement a commencé il y a combien de temps ?*

L.C. : Alors moi, ça fait trois ans que je suis là, parce qu'avant de démarrer cette transformation massive d'informatique, l'entreprise bancaire avait démarré un incubateur. Donc c'est quoi un incubateur ? C'est une entreprise dans une entreprise, c'est un pseudo-département. Donc le but n'est pas de prendre un corps de métier et de travailler dessus, mais c'était un endroit où un département existant pouvait envoyer une, ou deux ou trois équipes. Des équipes de 8 personnes, c'était le chiffre un peu fétiche. Les envoyer dans l'incubateur pour qu'ils apprennent une autre manière de travailler. Et une fois que ces équipes sont rendues Agiles, qu'elles appliquent Scrum correctement qu'elle puisse tenir la méthode, le mind-set dans le temps, sans être chouchoutée par des coachs qui viennent voir comment ça se passe, etc. Alors elles pouvaient retourner dans leur département de départ.

*Et là elles continuaient à travailler en Agile ?*

L.C. : Voilà. Elles arrivent dans l'incubateur avec un projet, donc il y avait un but à atteindre. La plupart du temps, c'était des apps pour téléphone, des apps mobiles, ou des applications web.

*Avant les méthodes Agiles, quelles méthodes étaient mises en place ? Quelle était la structure ?*

L.C. : Dans l'entreprise, c'était de la méthode de travail standard, c'était la méthode de projet Prince 2 qui était appliqué partout. Donc c'était beaucoup de milestones, beaucoup de livrables, beaucoup de documentation, un grand plan que l'on déballe entre 6 et 12 mois. Les projets étaient très lourds chez eux. Parfois les projets duraient même plus longtemps que ça.

Donc le souci de la banque, c'était comment est-ce qu'on peut aller plus vite. Parce qu'on n'est pas assez rapide sur le marché. Si on veut rester compétitif, si on veut avoir des produits d'accroche pour qu'on n'arrive pas après tout le monde : « Tiens, pourquoi l'entreprise sort seulement une app maintenant alors que... ». On ne va pas citer tous les concurrents, mais il fallait changer c'est sûr.

*Donc les objectifs à la base de l'implémentation d'Agile dans l'entreprise c'était d'aller plus vite ? Y en avait-il d'autres ?*

L.C. : Alors quand une entreprise se lance dans l'agilité, c'est rarement pour que les employés se sentent mieux. C'est dommage. Mais la plupart du temps, c'est : on ne livre pas assez vite, on ne livre pas assez, et quand on livre, on est parfois à côté de la demande du client. Et on ne s'en rend compte qu'après avoir investi 6, 8 ou 12 mois et c'est un peu trop. On se rend compte que ce n'est pas tout à fait ce que le marketing voulait. Ou si c'était ce que le marketing voulait, c'était pour se rendre compte que les utilisateurs ne le voulaient pas. Donc il y a parfois des mismatches comme ça. Il y a beaucoup de projets qui ratent, les projets ne sont pas rentables, sont hors budget. C'est les grands classiques. Et l'agilité prône d'avoir le customer à portée de main, de l'inclure plus tôt. Dans les méthodes Agiles, (Scrum – c'est celle qu'on a implémenté dans l'entreprise-) l'idée c'est d'avoir des moments de livraisons très courts. Ce qu'on appelle en Scrum un sprint. On peut avoir des sprints de 2 à 4 semaines, ou même d'une semaine, c'est possible. Et de pouvoir livrer quelque chose, ou, en tout cas, pouvoir montrer dans tous les cas de l'avancement, pour avoir du feed-back. Alors ça, c'est un des grands principes, on livre souvent un tout petit bout de quelque chose, mais qui fonctionne et on essaye d'avoir du feed-back. Comme ça on sait si on est sur la bonne voie, si on va dans la bonne direction. Et donc on essaye d'avoir des représentants du clients, ou de l'utilisateur final. Pour qu'il essaye le prototype. Et il peut dire : bien, ça me plaît, ou ça ne me plaît pas, ce n'est pas ça que je voulais, ou en fait c'est ce que je voulais, mais maintenant que je l'ai, ce n'est pas ça dont j'ai besoin. Parce que les gens peuvent se tromper à beaucoup de niveaux. Donc ça c'est un des éléments fondateurs.

*Voulez-vous réduire le time-to-market à tout prix ? C'était considéré comme un objectif stratégique ?*

Le coach : C'était très stratégique "réduire le time-to-market", puisque justement ils sont tous d'accords pour dire qu'un an c'est trop long. Le business se plaignait parce que l'IT revenait un an après, au mieux 6 mois. Mais 6 mois c'est une demi-année. Il y a beaucoup de choses

qui se passent. On pense à quelque chose, on est les premiers, mais s'il faut 6 mois pour que ça arrive sur le marché, que ça soit accessible pour le client, on a des chances que ça se fasse déjà par quelqu'un d'autre. Il fallait d'ailleurs ce temps. Évidemment, ils ont un espoir aussi de réduire les coûts. (?). Donc le premier objectif, c'était le time to market, le deuxième c'était le coût. Le troisième c'est quand même l'amélioration de la qualité. Et puis viennent d'autres par la suite, parce que finalement ils avaient 7 ou 8 objectifs avec par exemple la satisfaction des employés.

*Avaient-ils des attentes bien précises sur le déroulement et la finalité ?*

L.C. : Il y a eu l'incubateur pendant deux ans. Sur deux ans d'incubateur, on a coaché 150 personnes ou à peu près. Ensuite, il y avait vraiment les responsables de départements qui voulaient aller plus vite. Le retour des employés qui rentraient dans l'incubateur, est qu'ils ne voulaient plus du tout, jamais, travailler dans l'ancienne méthode. Certains départements demandaient si ce n'était pas possible de rentrer dans l'incubateur, le département tout entier. Mais il y avait 100 et quelques places dans l'étage de l'incubateur, on ne pouvait pas accueillir un autre département de 100 et quelques personnes. Ce n'est pas possible, ça doublait la taille, donc ça n'allait pas. Cela dit on a fait un pilote la deuxième année, un deuxième pilote. Donc on a pris le département qui s'occupait des applications mobiles, web, etc. et il a été "placé" dans l'incubateur. Et on a commencé la transformation Agile dans tout ce département-là. Il y avait 8 équipes si je me rappelle bien, en une fois. On a essayé d'aller plus vite et de nouveau ça s'est bien passé – pas aussi bien qu'on pouvait encore l'imaginer au début, mais bon c'est normal, c'est un gros changement. Ils savaient qu'il y avait le coaching, qu'il y avait des cours, qu'il faut changer la manière de gérer les projets eux-mêmes, ce n'est pas juste "devenir" Agile. Le management doit aussi changer ses attentes par rapport au reporting, par rapport au plan, etc. Ça ne se passe pas de la même manière qu'un plan en Prince 2. Ensuite, les personnes qui ont lancé la grande transformation, le maître de transformation c'est la CIO qui venait d'une autre banque, qui a été responsable de la transformation là-bas. Elle s'est entourée de quelques personnes qu'elle connaissait déjà. Et donc ils sont arrivés avec une idée des choses à faire, des choses à ne pas faire puisqu'ils avaient une expérience. Ça a pris un petit peu de temps à démarrer, quand ils se sont dit qu'ils allaient le faire, ça a quand même pris quelques mois de réflexions dont l'établissement du plan, avec des conseils extérieurs. Et finalement, ils sont arrivés avec un plan de la transformation de l'IT. Et donc, ils avaient une feuille de route. On va faire des "vagues" de 3 mois, donc 4 vagues, 3 mois chacune. Un certain nombre de tribes vont commencer par vague

dans la transformation. Les autres auront juste des sessions d'awareness et des choses comme ça. Et donc sur le coup, ils n'auront pas un suivi de près, par des coachs, et donc moins de coachings que la première vague. Lorsque la première vague est finie, les coachs passent aux tribus de la deuxième vague, etc. Donc ils avaient un plan, qui semblait bien ficelé, bien huilé. Du point de vue, des coachs – dont je faisais partie – ce plan semblait trop serré. Basé sur l'expérience que j'ai racontée, de l'incubateur, de 2 ans, où les équipes étaient devenues Agiles, mais où il fallait encore se battre contre de mauvaises habitudes ça prend du temps : les feedback et les corriger. Et donc ici, ils avaient dit sur un an on va faire tout ça, on va faire du forcing, et puis les gens seront prêts pour continuer tout seul. Et ça, ça semblait, de mon point de vue, beaucoup trop court. On est rentré dans le plan, on a essayé de suivre, et puis très vite, ils se sont rendu compte que ça allait prendre plus de temps que ça. Et ça **devait** prendre plus de temps. Donc c'est simple, on est aujourd'hui un an après le début, et les tribus qui sont entrés dans la vague 1 étaient toujours coachés, et les coachs ne sont pas partis. On n'a pas pu le faire après 3 ou 6 mois, un an après, on n'a lâché personne, tout simplement parce que la banque avait décidé de faire la transformation, mais de ne pas réduire les livraisons et on ne peut pas demander à des gens de suivre des cours, d'être coaché (le coaching c'est quelque chose qui prend du temps) On ne peut pas leur dire: "tu vas travailler moins d'heures par jour à coder ou à faire tes analyses parce que tu dois être coaché, parce que tu dois suivre des cours" et en même temps: "on ne peut pas réduire les choses que tu devais livrer cette année".

Cela ne semble pas très logique, mais c'était parti comme cela. Donc l'IT avait promis à son business de ne rien changer en termes de livraison. Vous allez voir, on va se transformer et cela ne va pas avoir d'impact. Et bien, le problème, c'est qu'ils ont été quand même poussés à tenir cette promesse. Du coup, c'est la transformation qui a été moins vite. Elle aurait pu aller plus vite, mais les gens n'étaient pas assez disponibles. Dès qu'on disait : on va passer une demi-journée à faire un workshop sur ... « Ah non, ce n'est pas possible. On doit livrer cette chose-là et celle-là et on est déjà en retard. » Malheureusement une banque ce n'est pas une entreprise informatique comme une autre. Il y a des obligations légales. Donc une nouvelle loi change, on a 6 mois pour l'implémenter. On a 6 mois et après il y a des amendes. Et donc voilà, le département où je suis tombé,. On a dû implémenter des projets obligatoires, légaux. On ne peut pas changer la date, et on peut essayer de jouer sur le scope et dire finalement avec ce minimum-là on est dans la légalité, le reste viendra après parce que c'est un peu du « nice to have ». Et cela a été très chaud, mais on l'a livré, malgré la transformation, mais elle a

donc été plus lente que prévue. Et après ce projet, maintenant c'est le suivant donc tu vois et cela n'arrête pas. Ils sont sur le prochain truc et le suivant, est déjà sur les starting blocks. La seule question que l'on a maintenant c'est : quand est-ce qu'il y aura un moment de répit pour avancer dans la transformation. Et ça, pour eux, c'est très très compliqué.

*Quelle méthode avez-vous mise en place ?*

L.C. : C'est une méthode propre à l'entreprise. C'est ce qui a pris quelques mois avant de démarrer. Dans l'incubateur, on appliquait Scrum, c'est le numéro 1 de l'agilité Scrum c'est bien pour une équipe. Le modèle explique qu'il y a un Product Owner, un Scrum Master, il y a l'équipe, cela fonctionne bien. Mais si j'ai maintenant 200 personnes, comment je fais ? Scrum te dit qu'une équipe c'est autour de 8 personnes. Est-ce que je fais autant de fois une petite équipe ? Et comment ils se mettent d'accord ? parce que l'informatique de la banque ce n'est pas juste "une équipe peut produire un produit, donc l'équipe d'à côté en fournit un autre", non, non. Pour la banque, mettre un nouveau crédit en ligne, il y a l'équipe des crédits, il y a l'équipe des contrats, il y a l'équipe qui fait le web, plus toutes les couches des équipes intermédiaires. Donc, ce n'est pas juste une équipe qui peut dire : je mets ce nouveau produit en ligne. Ce n'est pas possible. Du coup, si on applique Scrum en tant que tel, dans le modèle il manque quelque chose. Comment les équipes interagissent ? On n'est pas les premiers à penser à cela. Même les pères de Scrum avaient pensé comment faire un scaling de Scrum. Cela dit, la banque est partie sur un modèle qui s'inspire à la fois de SAFe, et de Spotify. De là, les jargons de tribus, de squads, ça c'est un peu à la Spotify. Mais, par contre, la rigidité du modèle de livraison, de la gestion des opérations, c'était très ITIL et cela doit rester assez carré pour une banque donc SAFe aide aussi. Le modèle a été construit en interne, il y a des squads, des PO, etc. Une squad travaille en Scrum ou en Kanban, donc si c'est une équipe qui fait vraiment de l'opérationnel, des incidents ou des opérations au jour le jour, ou qui fait de l'infrastructure, pour eux, cela n'a pas de sens de parler de sprint, et de projets, qui prennent x sprints à mettre en place, un changement ici, un changement là, c'est plutôt de la gestion de tickets en plus. Donc ils font du Kanban.

Mais dans l'ensemble, les autres équipes appliquent Scrum et pour tout ce qui est mis à échelle, quand les équipes interagissent entre elles, c'est inspiré de SAFe. Donc il y a des rôles de "release engineer", "test engineer", de "tribe architect", des rôles qui sont autour des équipes.

*Pourquoi avoir choisi Scrum et Kanban ? Pourquoi ces méthodes-là et pas d'autres méthodes ?*

L.C. : C'est une bonne question. Je ne sais pas parce que je n'ai pas participé au choix. Mais je pense que Scrum est la méthode numéro 1 de toute façon en termes d'agilité, parce qu'elle a prouvé qu'elle fonctionnait bien.

*Et en ce qui concerne SAFe et Spotify ?*

L.C. : SAFe a beaucoup de succès dans les entreprises rigides, parce que SAFe est sans doute la moins Agile des méthodes Agiles. Ce n'est qu'un point de vue.

*Spotify, c'est pour donner le côté le plus Agile alors ?*

L.C. : Spotify, c'est parce que c'est un peu hype. Alors eux, c'est extrêmement Agile. J'entendais, il n'y a pas longtemps, les parents du modèle dire que finalement le modèle que l'on appelle Spotify partout, n'est plus le modèle qui est en place à Spotify. Ils ont évolué, ils ont changé le modèle, donc pour eux c'est un peu étrange d'entendre le modèle « Spotify ». C'était une des versions. L'agilité ce sont des principes, donc ce n'est pas une méthode. Scrum c'est un framework, oui c'est une méthode, mais elle est très peu descriptive, donc c'est très libre. Il y a des choses plus descriptives. SAFe est excessivement descriptif, il y a des rôles, des trucs, des moments qu'il faut faire ceci et il faut faire cela. Spotify c'est Agile, donc ils ont réfléchi à comment est-ce qu'on met en place une méthode qui fonctionne chez nous. Spotify est arrivé à un certain modèle, à un moment donné, ils ont fait une démonstration de leur modèle.. Elle a plu à beaucoup de gens, mais eux, ont continué. En fait, ils ont fait évoluer le modèle et ce n'est plus tout à fait ce modèle-là. Je pense que cela, par contre, même l'entreprise bancaire l'a bien compris, leur 1er modèle celui qui est déployé maintenant, c'est le premier pas. Etils ont déjà une V2. Le V2 est déjà prêt et ils travaillent à la road map V1 -> V2,. Il y a des changements qu'une entreprise peut supporter et d'autres où le gap est trop grand à franchir en une fois.

Dans une informatique comme l'informatique de l'entreprise bancaire, chaque département est très cantonné à une application, parfois à une partie d'application et pas à un métier. Donc une équipe au sein d'une tribu n'est responsable que de quelques composantes d'une application parfois. C'est extrêmement précis. Alors, si on veut appliquer l'agilité au sens le plus dynamique du terme, une équipe devrait faire une livraison end-to-end comme on dit. Cela veut dire du front-end, par la database jusqu'au mainframe parce que comme cela on

peut fournir de la valeur au client en une fois. On s'adresse à une équipe pour une fonctionnalité et elle la construit de bout en bout. Or ici à la banque, je vais peut-être dire une bêtise, mais, pour avoir une nouvelle fonctionnalité, pour un nouveau champ pour l'application banque sur le téléphone, Il va alors y avoir 7 à 8 équipes qui doivent intervenir. Mettre à jour le front-end, c'est une équipe, mais mettre à jour le middleware c'est une autre équipe et mettre à jour le mainframe, c'est une autre équipe et s'il y a une database, c'est une autre équipe. Voilà, cela en fait déjà 4 ou 5 et s'il y a un changement d'infrastructure à faire, c'est encore d'autres équipes. Cela demande beaucoup d'équipes parce que ce sont des responsabilités séparées. De dire, on va transformer et les équipes font du end-to-end, ça c'est de trop. Est-ce que c'est seulement possible, avec l'architecture d'aujourd'hui? Peut-être pas, parce que c'est trop de couches différentes, on ne peut pas prendre juste une personne de chaque couche, et la mettre dans une équipe. À la limite, ce ferait une équipe de plus de 8 personnes. Tu vois ce que je veux dire. Il y a des moments, c'est vraiment cela qui se pose parce que, par exemple, dans l'incubateur, je me suis occupé d'une équipe qui fait une nouvelle forme de gestion des montants sur la carte de crédit pour qu'on puisse faire des paiements étalés sur un achat fait avec une carte de crédit. C'est très sympa comme concept, mais pour faire cela, et on a réussi à le faire, dans une seule équipe, il fallait un développeur IOS - parce qu'en général les gens sont très spécialisés - un développeur Android, un développeur Web parce qu'il faudrait que ce soit disponible sur le mobile, mais sur le web aussi avec un laptop. Donc cela fait déjà 3 personnes. Puis, il faut des gens qui soient spécialisés dans les tests, mais parfois les gens qui font les tests, connaissent les tests sur application mobile, mais pas sur le reste. Et ensuite cela demande des changements sur les crédits, il y a des composantes du mainframe qui doivent intervenir. On avait une personne du mainframe. Cela fait déjà 5 personnes et puis finalement parce qu'il y a une vitesse de livraison, il fallait développer cela en quelques mois donc il fallait 2 développeurs Android. Donc on est arrivé à une équipe qui était plus grande que 10 juste pour avoir une personne avec de la connaissance de chaque domaine. Mais si un est malade, on ne sait plus travailler. Il part en congé, l'équipe s'arrête. C'est très compliqué. Donc il faut avoir des gens polyvalents. Il faudrait qu'une personne puisse être à la fois : développeur pour IOS, mais aussi Android. Et s'il le faut, qu'il sache aussi configurer une data base. Puis le deuxième dans l'équipe, il sait peut-être faire le web et le mainframe. Il faut être polyvalent et il faut qu'ils puissent se partager comme cela les skills. Et cela, ça demande soit énormément de training, soit de nouvelles personnes. Cela fait partie des choses que l'on ne peut transformer comme cela. La première étape de la transformation c'est de dire, que l'on va garder les spécialités comme

elles sont aujourd'hui, une équipe qui s'occupe du mainframe, une certaine partie de l'application X, etc. On va essayer de les faire travailler de manière Agile. La conséquence de cela c'est que les équipes ne sont pas indépendantes. Or l'agilité fonctionne mieux avec des équipes indépendantes. Au plus il y a des interactions, de dépendances comme on dit, au moins cela va fonctionner, au moins cela peut être efficace. Parce qu'il faut s'aligner sur des plannings, ce n'est pas le tout que cette équipe-là aille très vite, si l'équipe d'à côté qui doit fournir l'autre morceau en a pour 2 mois cela ne sert à rien de faire tout en 2 semaines. Ou alors, il vaut mieux le faire au moment où eux vont être prêts à livrer sinon ils ne savent pas intégrer et puis s'il faut revenir sur le code qu'on a fait il y a 2 mois, et qu'il y a un bug, cela va être plus compliqué parce qu'il faut se replonger dans le travail.

Ce sont des choses un peu compliquées qui font qu'on est obligé d'avoir un déploiement de la méthode Agile étape par étape. Donc d'abord, "agilifier" les gens, les équipes, ensuite, revoir l'architecture, aller maintenant vers des microservices. On peut commencer à développer de manière indépendante, puisque tout ne doit pas être synchro, on peut utiliser le service plus tard, on sait déjà qu'il fonctionne.

Donc il y a une version 2 du modèle de l'entreprise bancaire pour les années à venir qui demande un retravail de l'architecture, vers plus de connexions possibles.

Cela va diminuer le couplage ou les dépendances en faisant plus de choses qui soient "plugable", qu'on puisse connecter sans effort. Une fois qu'ils auront revu leur architecture, ils vont revoir leurs skills et disant et bien maintenant c'est plus facile, il y aura moins de couches et on pourra faire cela avec une équipe seule.

*En ce qui concerne les modifications des méthodes Agiles, y a-t-il eu d'autres modifications mis à part la mise en place de SAFe, Spotify et la V2 ?*

L.C. : elle est ajustée en permanence. Ça, c'est le côté Agile de l'implémentation Agile. Il faut aussi s'adapter en permanence. Le plan de départ, c'était x semaines de training, x semaines de ceci, et puis finalement il faut suivre les gens. Les gens, ce n'est pas comme une machine où je dis : « Tiens, je vais faire une nouvelle version de ma voiture. Je vais faire un plan, je vais changer la couleur, changer les jantes, la taille des roues, mettre un spoiler, je ne vais pas changer l'injecteur et cela va me prendre autant de temps ».

Ici, ce n'est pas cela, je vais travailler d'une certaine manière, ce n'est pas parce qu'ils ont suivi un training de 2 jours que, quand ils retournent à leur place, ils travaillent autrement. Ils

ont juste entendu un tas de concepts auxquels ils ne croient pas encore spécialement. Ils demandent à voir et puis, en pratique, comment je fais. Le changement n'est pas que de la cellule, du groupe, du team, mais c'est en plus le marketing, la communication avec le business. Maintenant, il y a un PO. Le PO n'écrit plus le cahier des charges comme avant. Je ne reçois pas 150 pages de spec qui ont été approuvées par 15 personnes. Maintenant, le PO vient et dit : je ne sais pas trop en détail, c'est encore en cours, mais on a l'idée de faire cela. Là, tout le monde regarde et on fait quoi avec cela ? Avant, cela se faisait en amont et de manière opaque par rapport à l'équipe. Aujourd'hui, l'équipe informatique est beaucoup plus au courant de ce qui se passe. C'est au bénéfice de tout le monde, parce qu'on fait moins d'erreurs, parce qu'on sait dès le départ les discussions en cours. Donc finalement l'idée du marketing c'est ceci, cela, c'est parti d'un truc embryonnaire et ça a été affiné. Plus on implique les gens, plus ils savent pourquoi et comment on arrive à une solution. Ils peuvent alors eux-mêmes donner plus de feed-back sur la solution en disant : « je ne suis pas sûr que cela réponde aux besoins du départ parce que si j'ai bien compris... »

Alors que dans la méthode classique, le business parle au project manager, qui parle avec une paire d'analystes séniors et un architecte et eux font un design qui est compréhensible pour le développeur et je suppose que chacun a bien fait son travail. Ici, on essaye de réduire tout cela à une communication directe. Il y a toujours un analyste et un designer, mais ils sont dans l'équipe. Et autant que faire se peut, on implique l'équipe aux discussions. Cela veut dire tout le monde tout le temps le plus souvent et le plus tôt possible.

*Utilisez-vous d'autres méthodes que Scrum et Kanban ?*

L.C. : C'est principalement ces deux-là.

*Est-il possible qu'au milieu d'un projet vous changiez de méthode ? Que vous passiez de Scrum à Kanban par exemple ?*

L.C. : En tant que tel, on ne parle plus autant de projets. On parle plutôt de produits que de projets. Les équipes qui sont passées de Scrum à Kanban, ou inversement, ce sont justement des équipes qui ne sont pas dans un projet, qui n'ont pas un produit à livrer, mais qui sont des équipes de support infrastructure, et des choses comme cela. Parce qu'ils ont essayé de faire Scrum, mais très vite cela ne colle pas. Les priorités par exemple, ce sont plutôt les premières arrivées, les premiers servis. C'est du first in first out.

Ensuite, il n'y a pas non plus le concept de livrer un incrément à chaque sprint qui soit fonctionnel et qu'on puisse tester. Mais si ce que l'on fait ne se prête pas à cela, si je ne dois pas livrer un morceau de produit, par exemple: je dois mettre un nouveau disque dans un serveur, je dois en même temps tirer des nouveaux câbles; à la fin du sprint le morceau ce sera : un tas de choses sans rapport les unes avec les autres.

Donc, pour eux parler de "Product increment", devient extrêmement virtuel, et moins c'est concret et moins ça tient la route. Les gens abandonnent, ils n'utilisent pas le concept correctement donc très vite cela ne correspond pas. En plus, il fallait essayer de planifier 6 ou 7 sprints à l'avance et de nouveau, pour une équipe qui travaille comme cela, parler 3 mois à l'avance, ça n'a pas de sens. Les demandes arrivent de semaine en semaine, je n'ai pas un backlog d'éléments à fournir pour arriver à une solution finale dans 6 mois. Ils n'ont pas cela. Je dois faire cela et cela et cela. Rien qui prend 3 mois ou plus, tout prend quelques sprints. Donc cela ne colle pas.

*Est-il possible que pour une raison ou l'autre, vous considériez qu'Agile ne fonctionne pas et que vous retourniez à Prince 2 par exemple ?*

L.C. : Alors là, cela dépend à qui tu parles et que tu poses la question.

*Quelles en seraient les raisons ?*

L.C. : Moi, je pense que la plupart des gens qui démarrent seulement se disent qu'Agile ne se prête pas à tout. Beaucoup de très grosses boîtes, déploient Agile en partie. En disant pour les choses un peu novatrices, avec fortes interactions avec l'utilisateur, l'agilité va marcher parce qu'on va montrer des choses, on va collecter du feed-back de l'utilisateur, on va s'adapter. Et cela donne une forte impression de bon sens de se dire qu'il y a des choses où ce n'est pas la peine de faire de l'agilité, on peut rester sur un modèle standard.

Si tu poses la question à un coach Agile, il te dira que l'agilité peut se faire à tout parce que c'est une question de mind set. C'est être proactif, c'est adaptatif, c'est les valeurs du Manifesto, plus les principes qui sont derrière. Où est le problème d'appliquer cela à quelque chose qui ressemble à une plus vieille technologie? Ou bien il n'y a pas de raison, on peut l'appliquer aussi. Maintenant, en toute honnêteté je pense qu'il y a des domaines, mais il faut les chercher, je n'en ai pas trouvé, où je peux comprendre que, par exemple, on a besoin d'un cahier des charges à l'avance, ferme, définitif signé, etc., mais avec tous les dangers que cela

comporte. L'interpréter de travers, mais je peux le comprendre. Je pense à la Défense, à un software médical pointu. Mais encore, pourquoi pas ?

Donc pour moi on peut l'appliquer à tout, mais je peux entendre que certains disent : je ne suis pas sûr. Mais de dire: "on a essayé la méthode Agile pour ce produit-là et puis on est revenu en arrière", je n'ai pas connu.

*Serait-il possible que certaines parties comme le cahier des charges soient fait en Prince 2 et continuer en Agile, est-ce que ce serait possible cela par exemple ?*

L.C. : Cela n'a pas de sens. Tout est possible, cela se fait par endroits, il y a des entreprises où il y a la gestion classique Prince 2 et puis quand on arrive à la demande d'implémentation dans l'équipe IT c'est une équipe qui fonctionne en Scrum. Mais on prétend que cela fonctionne en Scrum, mais cela ne peut pas puisque le cahier des charges est déjà établi or on est chargé de le construire ensemble. Le product backlog est censé être construit au fur et à mesure avec beaucoup de précision sur les choses à livrer en premier, on doit prioriser. Le backlog c'est une pile de choses à développer dont le plus prioritaire est en haut, c'est celui qui apporte le plus de valeur à l'utilisateur, au business puis vers les valeurs décroissantes. Ou bien ce qui comporte le plus de risques pour l'implémentation future. Mais si le cahier des charges est établi avant, alors on ne travaille pas sur base de valeurs, tout est déjà déterminé. Après si on veut adapter, si on se rencontre après 2-3 sprints que l'on est à côté de la plaque, il faut revoir le cahier des charges et on est reparti pour un round. On doit encore faire 3 mois d'analyse avec le client, tu vois cela ne tient pas la route. Donc ceux qui prétendent le faire, cela veut dire qu'ils ne sont pas tout à fait Agiles. Finalement, dans ce cas, le Product Owner n'est pas l'intermédiaire entre l'équipe et les stakeholders. Ils ne peuvent pas appliquer tous les principes. Donc, il y a des gens qui n'ont pas tout à fait le rôle attendu, et n'ont pas tout à fait le pouvoir attendu, le PO n'est pas celui qui décide de tout alors puisqu'il y a un chef de projet. Le chef de projet, s'il gère en Prince 2 il ne s'attend pas à avoir des délais dans ses livraisons, avec ses milestones. Il va vouloir passer ses package dans l'ordre qu'il a prévu. Cela ne colle pas. On peut essayer, on prétend, mais ce n'est pas tout à fait Agile. C'est vrai l'équipe fonctionne peut-être avec un délai de livraison de 2 semaines, et font peut-être du management visuel du travail, ils ont un board et des stories mais ils n'appliquent pas complètement le principe. Ce n'est pas possible. Alors ils font plutôt de l'itératif que de l'incrémental..

*Avez-vous évalué la mise en place des méthodes Agiles ?*

L.C. : Depuis le début, tout le temps. Il y a un assessment qui a été mis en place. Au début c'était une centaine de questions, sur les pratiques elles-mêmes : est-ce que l'équipe peut parler de choses simples: toutes les cérémonies Scrum; est-ce qu'elle fait son stand-up ? Est-ce qu'elle fait son sprint planning ? Les reviews? est-ce qu'elle fait sa rétrospective, etc. C'est parmi les questions. Après est-ce que le Scrum Master facilite les choses ? Il a besoin de coaching ? Qu'est-ce qu'il sort de la rétrospective ? Cela va-t-il bien dans un plan d'action ? Est-ce qu'il y a du suivi ? On essaye d'évaluer la maturité de l'équipe, des différents rôles-clés, de la tribu dans laquelle elle se trouve, est-ce que le management agit ainsi ou comme cela ? Je pense que dès la fin des 3 premiers mois, on a commencé à faire cette évaluation, tous les mois.

*Cela a donné quoi ?*

Le coach : Bien cela a donné que la période de progrès était plus longue qu'attendue, que prévue au départ. Mais cela a permis aux gens de voir qu'ils évoluaient. Cela a permis aussi de voir vers quoi ils devaient aller. Ce n'est pas le tout d'avoir des cours et du coaching, mais de dire finalement est-ce que j'ai une liste de choses où je sais que je dois changer ou agir ? Et quelle est la liste, est-ce qu'elle est courte ou longue, ou est-ce que j'y suis presque ? L'évolution classique d'une équipe qui devient Agile, c'est: au début, ils ne savent pas pourquoi on fait cela. Après, ils le font parce qu'ils sont obligés et puis une fois qu'ils le font par habitude, ils ont l'impression que ça y est. "Je suis transformé, je suis Agile". Oui, on fait les stand-ups, on utilise des post-it mais ils ne se rendent pas compte qu'ils doivent changer leur comportement, ce n'est pas juste parce qu'on est obligé. Il ne faut pas juste se dire : "je dois parler maintenant", ou "je vais écrire un post-it". Non, il y a beaucoup de choses à changer sur leur propre activité. Et à un moment donné, ils se rendent compte que, après tout cela, en fait, "on est encore nulle part". Il y a "faire" (de manière) Agile et "être" Agile. C'est un peu le jargon du coach Agile. Il y a l'étape où on se rend compte que là on agit encore comme cela, on pense encore comme cela, il faudrait qu'on pense comme ceci, qu'on agisse comme ça.

Là les gens passent à l'étape numéro 2 : prendre conscience, que finalement il y a encore beaucoup de choses à changer. Et donc commence à agir tout de suite.

*Mais est-ce que la transformation est finie un jour ?* Oui, la transformation est finie un jour, mais la remise en question est perpétuelle.

Donc la transformation aujourd'hui à la banque est arrivée à un stade où la plupart des gens - et cela a pris du temps - savent qu'il y a des choses à changer et se rendent compte qu'il y a des choses qui ont déjà changé. Mais il y en a qui ne sont pas encore là et c'est une étape capitale.

*La V2 était-elle prévue par rapport ce qui était mis en place au départ ? Savait-on directement qu'il fallait faire 2 étapes ?*

L.C. : C'était prévu depuis le début. À peine le V1 a été lancée, que le V2 était déjà au travail. Parce que justement quand on réfléchit, comment peut-on transformer une entreprise comme celle-là vers l'agilité totale, on conceptualise directement un modèle, mais on se rend compte que c'est beaucoup trop de changements en une fois. mais c'est un changement radical de l'informatique elle-même, l'architecture, l'infrastructure, on va changer les techniques, le métier doit changer. On ne peut pas faire cela en une fois. Alors on fait un pas en arrière. On se dit "qu'est-ce qu'on peut faire avec ce qu'on a ?" Eton met au point un modèle intermédiaire. Cela ne veut pas dire que le final était finalisé. Il a été affiné depuis, on est en train de travailler dessus, mais l'étape 1 a été établie parce qu'on savait que l'état final était inaccessible. Donc, dès le V1 sur pied on a commencé le V2 qui était là. On n'a pas dû attendre pour dire que cela allait aller moins vite que prévu.

*En ce qui concerne l'évaluation des produits, est-ce qu'il y a eu certains critères pour déterminer que les produits allaient plus vite ? Est-ce qu'on avait déjà des attentes, des objectifs à atteindre ?*

L.C. : Ça, c'est une grande différence entre le moment de l'incubateur et maintenant. Quand on rentrait dans l'incubateur, il y avait un produit déterminé. Parfois d'ailleurs, au début de notre introduction dans l'incubateur, il y avait des études de marché, il y avait des enquêtes, des prototypes, des choses comme cela. On savait ce qu'on voulait faire: une nouvelle app à créer sur le téléphone, une nouvelle page web, un nouveau produit. Ici, quand on dit on change toute la banque, il y a énormément de travail informatique et de la maintenance qui est juste des upgrades des choses qui existent déjà. De notre point de vue d'utilisateur, un nouveau produit ce n'est pas juste une nouvelle fonctionnalité ou une nouvelle page pour accéder à quelque chose.

Donc, quand on rentrait dans l'incubateur, la plupart du temps c'était un nouveau produit, quelque chose qui n'existait pas. Dans la transformation, il y a de tout. Donc dire, est-ce qu'il y avait un objectif à atteindre, par la transformation en termes de produits, c'est très difficile.

C'est vraiment au cas par cas et je ne pense pas qu'il ait eu une réflexion dans ce sens-là. En tous les cas, le département où moi j'étais, le timing était très mauvais parce qu'on était en plein project légal. Ce projecta pris 2 ans et on est arrivé avec la transformation au milieu de la 2ème année. Donc, autant te dire que le produit était déjà défini et presque développé complètement puis que c'était en mode Prince 2. Donc il'était presque en fin de tests, mais qui était censé durer 3 ou 4 mois. Tu ne peux pas arriver dans une équipe et dire, voilà on va faire un Product backlog avec ce que vous avez à faire pour le moment: des tests. Product backlog Tester. Quel est le produit ? Tester. Ce n'est pas un produit, c'est une activité, qu'est-ce qu'on fait maintenant ? Tu vois on était un peu mal pris. Donc c'était un peu virtuel, c'était compliqué. On a réécrit des user stories de test, c'était très fake au début pour eux, pas très tangible, cela n'avait pas beaucoup de sens. Ils se plaignaient beaucoup: mais c'est quoi cette story "L'utilisateur a besoin de voir le coût final sur un extrait de ...". Oui c'est déjà testé, pourquoi on doit écrire une story qui dit cela, car c'est déjà fait ?

Et donc voilà, "c'est pour apprendre à le faire". Tu vois donc pour ce département par exemple, ce n'était pas le seul truc qu'il y avait, mais c'était jusqu'au mois de janvier. De juin à janvier, il n'y avait presque que ce projet sur la table au centre des discussions. C'est très difficile de dire à ce moment-là de déterminer le produit par rapport à la livraison time-to-market. Cela fait déjà 1 an et demi qu'ils sont dessus. En janvier, on va être en retard et le projet suivant était déjà dans les starting block.

Donc dès que ce projet est livré, il faut faire le suivant, mais le suivant est déjà défini aussi, il est déjà obligatoire, on sait déjà que c'est pour juin. Compliqué. Et je pense qu'il y avait quelques départements qui avaient des produits, et donc la transformation en tant que telle n'est pas arrivée avec des objectifs en termes de produits, c'est beaucoup plus global.

*Du point de vue objectifs stratégiques, par rapport à la liste des 8 points dont vous avez parlé au début. Cela a-t-il été évalué ? Était-ce satisfaisant ?*

L.C. : Je pense qu'avec le temps, la transformation est devenue quelque chose de centrée sur elle-même et je n'ai plus beaucoup entendu parler des 8 points du début. Par contre, ce qu'ils ont fait - mais cela la banque avait l'habitude de le faire - il y a eu une enquête de satisfaction, il y en a eu plusieurs questions : comment sentez-vous la transformation ? Est-ce que vous pensez que c'est positif pour votre boulot ? Est-ce que cela va dans le bon sens ? Est-ce que vous avez les moyens de le faire ? Et je pense que dans l'ensemble, pour ce que j'en ai entendu, ainsi que pour la dernière évaluation encours, c'était positif. La culture du feed-back

fait aussi partie de l'agilité. Il faut se dire les choses de manière ouverte, en transparence, c'est important pour qu'une équipe fonctionne bien, et donc on demande toujours du retour aux gens, au coach, on demande le retour de l'équipe. On demande au Scrum Master le retour de l'équipe. On demande aux gens au sein d'une équipe de se demander du feed-back. On organise cela. Donc, le feed-back général des équipes après x mois de transformation, c'était : maintenant je sais mieux avec qui je travaille, ce que je fais, où est-ce que j'interviens, je sais mieux ce que mes collègues font, sur ce quoi ils sont en train de travailler. Les gens avaient l'habitude de recevoir du travail de leur chef, et de savoir ce que le collègue à côté d'eux faisait, on était sur la même chose, mais au-delà c'est beaucoup plus flou. Il n'y a pas besoin de savoir le chef de projet sait.

Ici on ne leur dit pas ça; c'est l'équipe qui doit savoir, et si tu fais quelque chose, cela doit être sur le board . Tout ce que l'équipe fait doit se trouver sur le visuel. Au début, cela a choqué certaines personnes: "mais cela n'intéresse personne!". Cela intéresse l'équipe parce que tu es occupé à faire quelque chose, il faut que les gens sachent, même si, j'entends: "je ne savais pas vous aider sur ce truc-là parce que c'est pas dans mes compétences, et j'ai pris du temps à lire ceci ou à apprendre cela en plus", et vous mettez une tâche: "John est en train d'apprendre l'application XY" alors qu'avant tout cela était un peu caché. De la même manière, puisqu'un objectif final est d'avoir des gens polyvalents, il faut savoir ce que l'autre fait. Et comme ça, j'apprends en partie. Certaines personnes ont commencé à faire du pairing et des choses comme ça, on se met ensemble toute la journée. Ou alors faire des sessions d'équipes où ils expliquaient leur métier aux autres. Cela paraît un peu surréaliste, au sein de l'équipe, les gens ne savaient pas ce que faisait l'autre. Si je suis développeur du composant AB et le gars à côté de moi faisait le CD. Cela n'a rien à voir avec moi. Je ne savais pas du tout ce que l'autre faisait. Il y avait une ségrégation très forte des métiers, et donc là tout d'un coup on dit peut-être qu'un jour tu pourrais l'aider à faire ces choses. « Ah bon ? Ce n'est pas ma responsabilité » et le message est très fort, est-ce que je suis responsable ou non. Ce n'est pas dans les jobs descriptions. Là, tout tombe à l'eau. On a vraiment cassé ces barrières-là. Maintenant, ils essaient de s'intéresser. Moi je ne sais pas développer cette chose-là peut-être je sais la tester et donc donner du renfort en test. Je sais mieux ce que je fais, ce que mon collègue fait, ce que l'équipe est en train de faire, sur ce quoi on travaille, quel est l'objectif à court terme, vu que le sprint est de 2 semaines, c'est un choix que la banque a fait. On a un objectif de 2 semaines et on avance tous ensemble et le fait que l'on promeut l'interaction et le partage, le dialogue, si quelqu'un a un problème il demande de l'aide et . Je ne vais pas

demander à quelqu'un qui connaît aussi. Non je vais dire que je suis bloqué et demander si quelqu'un a une idée. Parfois quelqu'un dans l'équipe dit : Ah moi je sais je vais venir te voir. Et si cette personne demande à une autre équipe peut-être que quelqu'un sait. Ça, cette dynamique-là les gens ont trouvés le truc que « Mais en fait c'est bien cela. C'est plus clair. »

Et on fait des plannings nous-mêmes. Ce n'est plus le chef de projet qui va dire pour le 27 mars tu dois m'écrire quelque chose. On avance et on sait vers quoi on va, même s'il y a toujours les contraintes légales. Tel truc pour telle date.

Ça, c'est le feed-back positif, on n'a pas encore eu des gens au départ réfractaires, puis "plus jamais je ne travaillerai avec ça"..

Dans la transformation générale, moi je n'en ai pas vu directement. Des gens qui faisaient au début "oui, mais cela ne marche pas dans tous les cas, pas pour notre style d'application, pas pour notre métier, pas pour...", et puis, finalement disent "oui !".

*Y avait-il une meilleure adéquation entre le projet rendu et ce qui était attendu ?*

L.C. : C'était beaucoup mieux, oui. On ne peut pas se tromper complètement sur quelque chose puisqu'on a donné le feed-back. Maintenant, ce qui arrive très souvent puisqu'on est orienté composant end-to-end et produit final, c'est que celui qui est dans le back-end fait quelque chose et le montre à ses stakeholders à lui et eux disent si ça va. Et de proche en proche on arrive au produit client. Mais on ne sait pas ce qui se passe totalement et si le client final ne va pas être satisfait puisqu'on ne le voit pas nous. Il n'y a qu'une équipe, celle du front end, qui travaille qui voit ce qu'au final.

Parfois, il y a des changements de spec qui semblent encore tardifs, par rapport à la méthode qui dit que l'on va le savoir dès le début. Parce qu'il y a des choses qui sont vues à la fin, à cause de toutes ces dépendances, tous les intervenants. Tout le monde n'est pas là en même temps avec un morceau de produit qui fonctionne. Il y en a un qui fait un morceau de back-end qui fonctionne, il y en a un autre un morceau de DB qui fonctionne. Mais ce personne ne voit tous les petits bouts ensemble. Ça, on ne peut pas faire à chaque sprint.

*Toutes les parties du projet sont regroupées toutes ensemble si j'ai bien compris, et à la fin s'il y a un souci, qui s'en occupe ?*

L.C. : C'est une bonne question. Tu vois le principe de l'équipe qui fait tout, qui teste, qui met en production, etc. c'est le but ultime. on ne se pose pas la question qui le fait. Si ça ne marche pas, c'est l'équipe qui est en train de le faire (le produit). Si tu en as beaucoup

(équipes), on est censé arriver chacun avec quelque chose qui fonctionne et qui est testé. Mais la demande d'intervention est tellement grande, et les problèmes de synchronisation étant ce qu'ils sont, la plupart sont obligés de dire " je livre alors que je ne sais pas si cela va s'intégrer avec le reste. Mais de mon côté, je l'ai testé en tout cas". Du coup à la fin, il reste une période de tests d'intégration. Cela demeure parce que ce qu'on a appelé les features sont souvent inter-squad et inter-tribe et donc quand chaque équipe a dit "mon morceau de la feature est prête et les stories de la feature sont prêtes", on pourra tester la feature dans son ensemble. Donc, il y a une période de quarter plan et il y a des périodes où on dit, pendant les sprints, on fait des tests d'intégration en général. Donc, s'il y a quelque chose qui ne va pas là-dedans, comme chaque squad, participe aux tests. ils savent que c'est dans leur partie puisqu'ils sont occupés, eux-même, à faire les tests.

*D'un point de vue impact, l'implémentation Agile dans l'entreprise bancaire en général, ils sont satisfaits ?*

L.C. : je pense. C'est très mitigé.

*Et en ce qui concerne la CIO, et l'équipe de base ?*

L.C. : Je ne sais pas exactement. Je pense qu'ils le présentent toujours comme un succès, mais d'un business à l'autre tout le monde n'est pas satisfait de la même manière. Certains disent : « Je ne vois pas la différence, je ne reçois pas le truc plus vite qu'avant », mais ils oublient que la transformation est en cours . C'est ce que je disais au début, ils ont dit, on va transformer en un an, et on ne va pas ralentir. Ça, ce n'est pas possible. Un message comme cela tout le monde doit dire : « Tu exagères un peu non ? ». Le problème c'est que c'est une promesse tenue et pour le business c'est "d'accord, très bien on va voir". Ils disent qu'ils n'ont pas reçu plus vite et ils ont même l'impression que c'est plus lent. Mais il y a des raisons pour que ce soit un peu plus lent. On aurait dû plutôt dit le contraire "on va se transformer pendant un an, cela va être un peu plus difficile. Mais après vous allez voir, on va bénéficier de ce qu'on aura appris", mais c'est un message difficile à faire passer à des gens qui peut-être depuis 10 ans disent que l'IT ne va pas assez vite, n'est pas assez réactive. Et donc, leur dire "et bien pour être plus réactive, on va être moins réactif pendant un an. Limite, on gèle tout". Ils n'ont pas pu faire passer ce message-là. Ils ont dit, ne vous inquiétez pas, cela ne va pas ralentir pendant un an, on va tout changer et après cela va aller mieux. Évidemment, ce n'est pas le résultat qu'ils ont eu pour certaines unités du business, ils ne sont pas contents.

*Pensez-vous que l'adoption des méthodes Agiles doit être basée sur certains objectifs stratégiques est-ce que c'est essentiel, selon vous ?*

L.C. : C'est philosophique. Je ne donne que mon avis. Je pense qu'une transformation Agile d'une entreprise doit faire partie de sa stratégie parce qu'encore une fois ce n'est pas anecdotique. Ce n'est pas "on va changer les templates, on va changer un peu les noms que l'on met sur les choses, mais l'entreprise tourne comme d'habitude". Pas du tout.

L'agilité est un changement de façon de penser, de penser le métier, le business, les interactions et donc le premier frein au changement c'est le management. Il ne s'en rend pas compte. Donc, il demande. Mais il demande parce qu'il entend "l'agilité c'est mieux", "ça va plus vite", ils espèrent plein de choses. Ils voient des chiffres, ils voient des clients contents, ils disent oui on va essayer. Et après, ce sont eux les premiers freins au progrès parce qu'à un moment donné, il y a le mind-set "command and control" qui n'est pas le mode Agile du tout.

L'agilité c'est: "empower the people", "shared responsibility". C'est juste à l'opposé (du command and control) en fait. De la hiérarchie avec le chef du chef du chef et puis les gens qui travaillent. C'est juste aux antipodes, Donc l'entreprise qui dit "je veux devenir Agile", doit se rendre compte qu'il y a beaucoup de choses qui vont changer. Comment je gère mes budgets, comment je gère mes gens, la gestion des ressources humaines doit changer, c'est énorme en termes d'organisation. C'est pour cela que c'est une transformation difficile et que souvent elle est faite à moitié. Même quand une entreprise prétend qu'elle a fait une transformation successful, elle dit : "nous on est Agile", et souvent, ils parlent de, comment, sur le terrain, dans l'atelier, les gens gèrent leur truc, mais l'entreprise n'est pas Agile pour autant, elle n'est pas complètement Agile. Il n'y a qu'une partie.

Qu'est qu'on va faire avec tous les middle managers, et tous les gens qui faisaient du people management ? Là, il ne faut plus faire du people management. Il faut servir les « people » pour qu'ils deviennent eux-mêmes empowered. Cela demande moins de gens puisqu'il n'y a plus autant de contrôle, il ne faut pas décomposer le travail, il n'y a plus des gens qui doivent faire des plans 8 h par jour. Les gens font leur plan eux-mêmes. Parce qu'un des principes c'est que, ce sont ceux qui font, qui sont les mieux placés pour planifier, ce sont les mieux placés pour estimer. Il n'y a plus des gens qui font les estimations pour eux, qui font les plans pour eux, des gens qui s'assurent que le plan est suivi. Cela fait beaucoup de managers, qui quand on a expliqué, disent : « mais qu'est-ce que je deviens moi dans ce modèle ? ». Il y a des Scrum Master, des PO et des tribe-PO, des tribe-Scrum Master mais moi je suis manager,

dans quoi je vais parce que je ne veux pas m'occuper juste d'une équipe, il n'y a personne qui s'occupe de l'ensemble. Il y a des gens qui commencent à avoir peur. Et très vite, sont apparus des rôles qui ne sont pas dans les Spotify, des rôles supplémentaires, les gens qui ont fait la gestion des ressources, mais même le mot ressources ne devrait pas exister. Il y a des gens, pas des ressources, c'est une machine une ressource, pas une personne. Et puis voilà, on a rajouté des postes, et bien finalement on a un tribeleader, il fait quoi lui ? Ce n'est pas un PO ce n'est pas un Scrum Master, ce n'est pas un test engineer cela veut quoi un try leader ? Dans le squad, il n'y a pas de chef. Le PO n'est pas le chef. Le PO décide des priorités du travail, mais il ne gère pas les gens, les gens se gèrent eux-mêmes et au niveau des gens celui qui regarde que tout se passe bien c'est le Scrum Master. Mais, ce n'est pas lui qui les gère... mais comment on fait pour les promotions, qui décide alors? Une entreprise qui dit qu'elle est Agile complètement souvent elle ne se rend pas compte de ce qu'elle demande. Je pense que c'est plus facile pour une jeune entreprise qui démarre d'avoir un créateur d'entreprise qui dise, moi je veux la démocratie, je veux management 3.0 et tu démarres comme cela. Tu ne crées pas une structure pyramidale, avec des chefs, des contrôleurs, des machins. Automatiquement, on est une équipe, on est des amis on organise le bazar et on grandit automatiquement et sans que ce contrôle soit mis en place. Là tu peux devenir et rester Agile. Mais une entreprise qui a un millier de personnes avec des dizaines et des dizaines de managers, qui ont pendant 10-15 -20 ans renforcés leur culture du contrôle, de la gestion, avec des comportements qui ont été récompensés pendant tout ce temps-là. Tu as reçu un bonus parce que tu as été promu parce que : tu es un bon contrôleur, bon manager, tu es quelqu'un qui gère bien les gens, qui dit ce qu'ils doivent faire, qui met les gens à leur place quand il faut le faire. Ce sont des comportements induits, récompensés et appris et ça c'est le plus dur. C'est dire : ok tout cela est du passé. Il y a des gens qui ont perdu, et ce n'est pas anodin, ce n'est pas anecdotique, c'est important, il faut gérer quoi. Qu'est-ce qu'on fait avec ces gens-là ? Est-ce qu'ils savent se reclasser dans quelque chose de plus dynamique, de plus Agile ? Est-ce qu'ils peuvent lâcher le contrôle qu'ils ont chéri pendant tout ce temps-là. Est-ce qu'ils peuvent l'abandonner, ils essaient : certains y arrivent, mais c'est difficile. Du coup c'est compliqué.

*Que sont devenus ces rôles dans la V2 ?*

L.C. : Dans la V2, il y a moins de rôles. Il y a des étages qui sont tombés. S'ils vont un jour vers la V2, il y a des gens qui devront se reclasser.

*Y a-t-il des choses mises en place pour l'instant pour les reclasser ?*

L.C. : Je ne sais pas pour le moment. Mais on le voit déjà aujourd'hui, il y a certaines personnes qui avaient pris des rôles, un an après se disent, je n'y arrive pas. Le fait qu'il y a un suivi de la transformation, et qu'il y a quelqu'un qui dit derrière, j'irai te voir après, là tu as refait le manager. Il faut que tu arrêtes de faire cela. Là, tu dois plutôt poser des questions, plutôt que de suggérer des solutions ou les imposer. L'équipe a donné son idée et toi tu viens la casser en disant : ce n'est pas comme cela qu'on fait. Avant, tu faisais cela maintenant il ne faut plus faire cela. « Oui, mais c'est parce qu'ils n'ont pas vu que... ». Dis-leur : Est-ce que vous avez vu qu'il y a ce problème-là, potentiellement, si vous faites comme cela ? Mais si tu continues à dire, oui, mais moi je sais, ils ne vont jamais prendre le leadership et c'est ce qu'on attend, qu'ils le prennent. Les gens qui après un temps disent : « Ecoute, je ne sais plus ce que je peux dire, pas dire, je suis perdu, je ne veux plus être Scrum Master. Ça ne colle pas. » Certaines personnes disent qu'ils abandonnent complètement, et retournent dans l'équipe, qu'ils redeviennent un analyste, certains sont revenus au métier qu'ils faisaient peut-être il y a 10 ans. En disant, finalement, c'est plus intéressant ça. D'autres disent, j'ai besoin de cours, ça ne va pas. Pas juste comme cela, il me faut plus de techniques, j'ai besoin, et d'autres disent : on change, je veux autre chose. Je veux devenir PO. C'est de nouveau se détacher des gens, parce que sinon ils restent dans la gestion des gens. Or le Scrum Master ne gère pas les gens. Il gère la dynamique entre les gens.

*Apporteriez-vous encore des améliorations dans le système mis en place ?*

L.C. : Moi je changerais encore beaucoup de choses. Ce n'est pas du tout fini. La transformation n'est pas terminée. Je le disais en le pensant très sincèrement. Là où est l'entreprise aujourd'hui, c'est l'étape où la plupart des équipes sont aujourd'hui, au stade où elles ont bougé, bon gré mal gré. Elles étaient là, elles ont fait un bout de chemin, et aujourd'hui elles se rendent compte du chemin parcouru, ce qui n'était pas encore le cas il y a 4 mois: "Ah oui, on ne fait plus comme cela et c'est beaucoup mieux et maintenant je comprends mieux et je vois que c'est nécessaire, je vois que c'est utile", c'est le bon moment pour eux, mais elles se rendent compte aussi que "oui, mais je ne peux pas aller plus loin, parce que le management ne nous a pas rencontré en chemin". "Oui coach, tu nous as dit que, et on a essayé de prendre le pouvoir, la décision et on s'était mis d'accord de manière pratique, coordonnée. Et puis soudainement, un ancien manager est arrivé, et a dit et pourquoi vous avez fait cela comme ça, pas question, il faut garder cela comme ça et comme ça."

*Ce sont surtout les personnes du management qui restent réfractaires ?*

L.C. : Et ne l'avoueront pas. Tu entends aujourd'hui dans certains meetings de management, tu vois des gens qui se plaignent des équipes qui doivent encore évoluer. Donc ils voient ce qui manque encore là-bas et ils ne se rendent pas compte qu'eux en fait n'ont pas fait la moitié du chemin que les autres ont fait quoi. Ils en ont fait, pour certains plus que d'autres. Certains ont vraiment changé, les managers dans le mode : "maintenant je demande aux équipes et ils doivent me dire, et je ne veux pas prendre des décisions pour eux". Tu dis : "lui a vraiment changé". Mais dans la majorité des cas et surtout quand ils se réunissent pour discuter entre eux, ils sont dans les anciens réflexes et donc ils disent : "les équipes, ils n'ont pas encore bien compris et ils attendent qu'on leur dise". Non, ce n'est pas vrai. Mais va les voir, tu ne discutes pas avec eux. Dans l'agilité, le management ne devrait pas être toujours dans son bureau, il devrait aller voir et communiquer, mais en live. "Et c'est quoi le problème?", "où est-ce qu'on est ?" "Est-ce qu'on est dans la bonne trajectoire, quelles sont les choses que je peux résoudre en tant que manager?". Finalement, ils ont changé les étiquettes, ils sont leaders managers, mais ils sont là pour servir l'équipe. Parce qu'une équipe ne peut pas changer l'organisation autour d'elle. Il faudrait des gens qui ont plus le pouvoir de le faire. Donc le rôle des managers dans l'agilité c'est ça. C'est de se dire tout ce que vous ne pouvez pas adresser dans votre cercle d'influence, moi je vais essayer. Pour que vous ne soyez pas freinés dans votre développement, dans votre livraison, etc. Mais c'est un autre mind-set que de dire, je viens voir et contrôler que tu fais bien ce que tu dois faire, pour atteindre l'objectif que tu m'avais donné. Là non, c'est de dire, il y a un objectif, vous avez établi un plan et je viens voir s'il y a quelque chose qui freine dans votre développement. Et s'il y a quelque chose qui vous freine je vais essayer de le résoudre et je reviens vers vous en vous disant si j'y suis arrivé ou pas. C'est une mentalité totalement différente et la plupart des managers n'arrive pas à sortir de leur bureau. N'arrive pas à aller, "je vais chez l'équipe et je fais mon petit tour une fois par semaine", "ça va les gars?". Ils se rendent compte de certaines choses, mais ils sont dans les vieux réflexes de dire : "on a besoin de savoir cela et donc je vais inviter un représentant d'équipe à venir me voir" tu vois? Ils ont eu l'impression d'avoir beaucoup changé en faisant cela malheureusement dans la tête des gens qui reçoivent cette invitation, tu vois c'est : "on me demande d'aller donner un statut à un chef". C'est à l'ancienne ça. Alors que l'on a expliqué pendant 8 mois que non, maintenant les chefs sont des serveurs leader. Ils doivent venir, **venir vous** aider. Jamais, ils ne me demandent d'aller, tu vois cela n'a pas changé.

Pourtant le manager a l'impression qu'il a changé parce que "je lui ai demandé de venir me voir et je ne demande pas qu'il fasse un rapport à son sous-chef pour venir me voir". Le chef a l'impression de voir qu'il a fait beaucoup: "J'ai bien changé, regarde!".

"Non, si tu avais été toi alors c'était top. Mais rien que cette petite nuance: tu l'as fait venir, pour lui rien n'a changé".

*L'Agile Manifesto, quel rôle ça joue dans l'entreprise ?*

L.C. : Le Manifeste c'est un peu comme un cadre sur le mur. Je ne crois pas que quelqu'un l'ait fait, mais donc souvent quand on veut corriger un comportement, on dit : "rappelle-toi que...". C'est la clé de voûte. On a expliqué cela à tout le monde au début donc, toute la banque, le marketing compris. C'est quoi les valeurs Agiles, c'est quoi le Manifesto ? Pourquoi il est là ? Dans le cours que l'on donnait pour expliquer cela, on essayait d'amener les gens avec un petit jeu avec des questions pour qu'ils se rendent compte que ce qu'ils ont expérimenté dans le passé, ce qu'ils voient comme le meilleur projet, la période de vie où ils se sentaient le mieux dans l'entreprise en fait reflétaient les valeurs Agiles. La plupart disent, "c'est quand j'avais les interactions avec le client, et là, on a pu montrer des choses", "on n'était pas figé par les outils". Inversement. La plupart des lieux communs erronés, comme "on ne fait pas de doc". "On privilégie le développement software à la doc, mais on ne dit pas qu'il n'y a pas de doc. Si tu dois choisir, aujourd'hui je n'ai que 2 jours à faire un truc, je vais d'abord faire le software qui marche et l'autre un peu plus tard s'il faut. Mais tu dois faire ta doc il en faut. Donc on revient au Manifesto, on rappelle. "Ce n'est pas qu'il n'y a pas de contrat, mais il vaut mieux que le customer soit là, qu'on discute, qu'on lui montre au lieu de de parler de contrat".

*Avez-vous d'abord présenté de l'Agile Manifesto avant les méthodes ?*

L.C. : Ensemble, à chaque fois qu'un élément de processus est discuté, s'il enfreint des principes on dit je vous rappelle que... Ce qui est plus important ce sont les interactions des gens plus tôt que...

*Quelles seraient les définitions des méthodes Agiles et de l'agilité, selon vous ?*

L.C. : J'aime bien de demander aux gens que je donne les explications : est-ce que vous savez d'où vient le mot Agile ? L'étymologie de l'agilité ? Cela veut dire vivant. Cela résume assez bien les choses.

Donc la définition de l'agilité, qu'est-ce qu'un être vivant, c'est quelque chose qui agit, qui est en mouvement perpétuel. L'agilité est le fait d'être réactif, d'être en communication, d'être en relation. Pour moi l'agilité ou les méthodes Agiles, c'est des méthodes centrées sur la réaction, l'humain, la cellule, le travail, plutôt que les méthodes monumentales. Qui sont, elles, centrées sur le livrable, le processus pour faire le livrable. Ça, c'est déconnecté de l'humain alors que l'agilité finalement s'est recentré sur l'humain par rapport au travail. Ce sont les interactions, les gens, avec le client, entre les membres de l'équipe c'est basé sur l'expérimentation, tout cela est centré sur les personnes qui font, et comment ils le font et comment ils interagissent pour le faire. L'agilité dans le software démontre qu'il y a une plus grande créativité et d'art que de méthodes.

L'agilité c'est être, être à l'écoute, être sensible, être actif, proactif, être responsable. Etre sans cesse en mouvement, en amélioration.

*Quelle serait la conclusion de votre expérience sur les méthodes Agiles ?*

L.C. : L'agilité c'est : l'essayer c'est l'adopter. Quand tu as goûté tu ne reviens pas en arrière.

## Appendix 6 : Interview Anonymous 5

X : worked as a Program Manager in a big insurance company and Program Director in a big informatic leader )

*Pouvez-vous vous présenter en quelques mots ?*

X. : J'ai une licence et maîtrise en informatique de gestion que j'ai acquise en 1991, ça doit me donner 30 ans d'expérience informatique bancaire/financière.

*Vous avez une expérience dans plusieurs entreprises, pouvez-vous m'en dire un peu plus ?*

X. : Oui, j'ai une expérience dans les domaines bancaires. Je fais aujourd'hui de grands projets en France et dans le monde de la fédération des prêts, on migre 250 milliards de dettes, ce qui est à peu près équivalent aux dettes de la Belgique.

*Quelle est la structure de votre organisation ?*

X. : En grandes lignes, il y a deux grands métiers. Il y a ce qu'on appelle le commerce, la vente de projets, et de l'autre côté la production et en-dessous de la production il y a la recherche et développement, et ensuite l'implémentation.

*Quand et pourquoi votre organisation a-t-elle décidé d'implémenter Agile ?*

X. : Pour moi l'Agile c'est du bullshit pour les entreprises, en gros c'est juste un moyen marketing pour avoir plus de clients. On va dire qu'on va livrer plus vite et donc ils sont intéressés. Mais en réalité ce n'est pas comme ça, pleins d'hypothèses de l'Agile Manifesto ne sont pas respectées. Les entreprises gagnent des sommes phénoménales par trimestres donc elles ne vont certainement pas dire qu'Agile n'est pas parfait. C'est devenu surtout un grand atout marketing.

*Quelle méthode avez-vous utilisée avant d'implémenter Agile ?*

La méthode Waterfall. Mais c'est comme dans Agile, c'est n'importe quoi. Tous les mots, anglais ce sont des mots qui font rêver mais ça n'exprime pas grand-chose au final.

On va rendre un projet, mais on est dans un monde tellement devenu complexe que d'office le projet à la fin ne sera pas un succès, il sera en retard parce que le monde a changé, les requirements ont changé, tout a changé dans le déroulement du projet. Donc les attentes d'il y a un an avant, ne sont plus les attentes d'aujourd'hui. Parfois il fallait un an pour mettre en

place un projet, pour développer la charte des besoins. Donc dans un sens, Agile apporte une amélioration, mais ce n'est toujours pas parfait.

*Donc Agile est quand même mieux que Waterfall ?*

Agile c'est pareil. On a beaucoup de mots en anglais dans Agile, on a backlog, user stories etc mais tout ça c'est de la poudre aux yeux, il y a 80% de mots anglais, et 80% de bullshits.

Agile dépend surtout du contexte pour que ça fonctionne. Et dans beaucoup de contextes ça ne fonctionnera pas. On se dit donc Agile pour vendre surtout, mais au final on est pas vraiment Agile.

*Pouvez-vous m'expliquer en quoi Agile n'est pas parfait ?*

Une première hypothèse c'est que les gens doivent être motivés, or dans une grande entreprise d'assurances, on utilise Agile pour moins payer les employés en IT. Du coup évidemment, les employés ne seront pas motivés. Un exemple très concret c'est : les RH qui décident de ne plus payer leurs employés pour leur travail le dimanche, alors qu'ils travaillent pour lancer les tests de projets. Du coup, évidemment un des employés me dit « Désolé Damien, mais moi je ne viens pas travailler dimanche comme je ne suis pas payé. » Du coup, j'arrange les choses pour faire en sorte qu'ils soient payés sauf que la RH refuse. Du coup, il n'a plus voulu travailler le dimanche. Et je ne pouvais pas lui en vouloir, il avait raison.

Une deuxième hypothèse par exemple, c'est d'avoir une équipe stable. Si c'était le cas dans l'entreprise ça pourrait bien fonctionner sauf que les gens tombent malades, les femmes peuvent être enceintes, les bons travailleurs s'en vont, vont voir ailleurs. S'ils ne sont pas motivés et qu'ils sont bons, ils vont d'office aller voir ailleurs. On peut très bien avoir un problème avoir besoin d'un DM pour régler ce problème, mais si le DM est absent pour maladie ou autre et il faut attendre qu'il revienne. Il y a aussi le fait qu'on va dire « voilà j'ai besoin de 7 user stories là, et j'ai besoin du langage SQL, sauf que celui qui sait coder du SQL n'est pas là, c'est pas possible. Du coup comme il est en congé, les autres ne savent pas travailler et c'est à l'arrêt.

Une troisième hypothèse c'est de livrer de manière systématique à telle date. Sauf qu'évidemment, on ne peut pas garantir qu'il n'y ait pas de problèmes donc en fin de compte, on livrait toujours avec une semaine de retard. Alors ces bugs, ce n'est pas grave pour un site internet pour un journal par exemple. On peut vite réparer en urgence, ça ne va pas

énormément impacter. Maintenant si c'est le cas pour une application de crédits par exemple, ça, ça n'ira pas. Ce ne sera pas accepté.

Et puis, il y a encore d'autres hypothèses. Il y a l'hypothèse que les équipes sont 100% fonctionnelles sur un seul projet. Or c'est faux. En général, dans toutes les boîtes les développeurs sont mis sur plusieurs projets en même temps. Du coup, s'il y a un problème dans un de ses projets, le développeur va arrêter le projet sur lequel il est occupé de travailler et va résoudre le problème. Sauf que c'est contraire à Agile justement. Agile dit qu'on va travailler sur un seul projet et se rendre à 100% disponible dans ce projet.

Il y a aussi l'hypothèse de la vélocité. On va dire qu'au début ça va diminuer jusqu' à 50% puis ça va réaugmenter. Sauf que chez moi par exemple, ça a diminué jusqu'à moins de 10%. Et après en effet ça a réaugmenté. Mais elle ne sera pas stable du tout, car comme je l'ai déjà dit, il y a des gens malades, des personnes qui s'en vont dans une autre boîte, et cetera. La vélocité restera toujours en pics.

Agile va promettre de monter un toit, sans même avoir les fondations, c'est insensé. Pour avoir une salle de bain, pour ça il faut un évier, et donc de l'eau et donc des raccordements qui viennent de la fondation. Et donc il faut que ce soit prévu à la base.

*Vous voulez dire que les méthodes Agiles ne fonctionnent pas ?*

Non, je n'ai pas dit ça. On ne peut pas dire qu'Agile ne fonctionne pas. C'est un peu comme les commandos à l'armée, un exemple d'expérience, car j'ai été moi-même commando. Les commandos spéciaux travaillent très bien, mais c'est parce qu'ils sont spéciaux que ça fonctionne très bien, ils sont bien payés, motivés. Ils ont des motivations comme le fait qu'ils veulent défendre leur pays, pour l'honneur ou pour d'autres choses. Et dans la réalité, il ne peut pas y avoir des commandos spéciaux partout, ce qui les rend justement spéciaux c'est qu'il y en a pas d'autres. Imaginons qu'on essaye d'appliquer ça partout, ça n'ira pas.

*Dans quels cas Agile pourrait fonctionner alors ?*

X. : Agile peut très bien fonctionner dans des boîtes qui sont motivées, qui ont toutes les ressources nécessaires. Là où les gens sont motivés, prêts à travailler n'importe quand, quand on en a besoin. Les employés de Google ou Facebook par exemple, ils voudront bien travailler le weekend car ils sont payés pour.

*Est-ce que les PME pourraient plus facilement réussir à mener à bien l'implémentation d'Agile ?*

X. : Non, ça n'a rien avoir. Ce n'est pas la taille de l'entreprise qui fait. Il faut que les hypothèses dont on a parlé soient respectées. Je dirais même que pour les PME, s'il doit y avoir quelqu'un absent c'est aussi difficile de le remplacer. Et puis, elles ont aussi des courts ou des longs projets. Ça pourrait moins aller pour des longs projets. Pour des courts, ça pourrait éventuellement aller, car on pourrait voir rapidement les erreurs. Et on peut aussi plus facilement déterminer où sont les erreurs. Tandis que pour de longs projets on ne peut pas tout prévoir, voir toutes les éventuelles erreurs.

*Pourquoi les entreprises se disent-elles Agiles alors ?*

X. : Voilà, on ne va pas dire au client non plus, client qui nous rapporte 10 million d'euros, que l'Agile c'est du bullshit, on va leur dire que l'Agile c'est génial, parce qu'il faut vendre. Car en tant que vendeur on doit aussi pouvoir nourrir nous, notre famille, subvenir aux soins des enfants, etc. Agile n'est pas implémenté à 100%, mais pour des besoins marketing on va d'office dire que c'est génial.

Agile est bien pour l'innovation. Mais pas pour tout. Par exemple pour Spar, les courses qui se font totalement sur une application, on ne peut pas se permettre d'avoir des bugs, car il faut éviter les piratages et le vol. Il faut également que l'application sache bien déterminer les codes-barres, c'est important. Donc il faut beaucoup de tests.

Donc oui Agile est bien pour l'innovation et où les bugs n'impactent pas énormément, ne sont pas grave. Dans le sens où, s'il y a un bug sur la plateforme de presse pendant 10 minutes ce n'est pas grave. Pour une banque par exemple, pour les taux de prêt, les garanties, cela a une toute autre ampleur. D'ailleurs un exemple, dans une des entreprises que je travaillais, et c'est véridique, on avait testé, tout va bien, sauf qu'on ne fait pas trop de tests non plus. Et pour finir, le client revient en disant voilà, ça fonctionne avec une garantie, mais avec deux garanties ça ne marche pas. Du coup, on revoit ça, on renvoie ça aux informaticiens indiens en Etats-unis, ils règlent ça et on renvoie tout au client en disant que c'est réglé. Sauf que qu'est-ce qui se passe après ? Tout fonctionne, ce problème-là est réglé, ça fonctionne avec une garantie, avec deux, mais qu'est-ce qui ne va pas ? Avec trois garanties ça ne fonctionne plus.

Alors oui de nos jours, tout le monde est formé, on suit des formations pour être Scrum master, ou autre. Et donc tout d'un coup on est Agile, on est bien mieux préparé. Mais au final, il y aura toujours des bugs.

*Vous avez parlé de tests, ça n'aide donc pas à être plus Agile ?*

X. : On a plein de différents tests automatisés. On appelle ça des environnements, on a un environnement de Built, c'est l'environnement des informaticiens, ensuite il y a l'environnement test – les testeurs informatiques qui font des tests unitaires, puis il y a l'acceptation – là ce sont les utilisateurs qui testent, c'est une simulation de la production, et ensuite ça passe en production. Alors même avec tous ces environnements, on va toujours oublier quelque chose surtout dans un monde de plus en plus complexe.

*Y a-t-il des cas pour lesquels les méthodes Agiles ne pourront jamais marcher selon vous ?*

X. : Les méthodes Agiles ne peuvent pas fonctionner pour certains cas, par exemple une plateforme pour compléter les taxes en ligne. Pourquoi ? Parce que cette plateforme va être disponible pour tous les citoyens en même temps, et il y a tellement de cas particulier que tous ces cas n'auront pas été considérés. Donc uniquement se baser sur Agile, on ne va pas penser à tous les cas. Et donc ce ne sera pas un succès puisque c'est censé être directement fonctionnel pour n'importe quel citoyen même pour les cas particuliers. Agile ça ne va pas marcher, car on va à chaque fois livrer certaines spécificités, mais pas toutes et on va donc en laisser de côté d'autres, car il faut qu'on livre. Avec Agile, on sait quand on va livrer, mais on ne sait pas la date de la finalité. Un autre exemple c'est Microsoft ou Tesla, on ne peut pas dire dans ces cas-là, voilà on va mettre 98% des fonctionnalités. Et puis on voit par exemple pour Tesla il y a un accident. Non, ça ne va pas. Pour ces cas-là on ne peut pas travailler totalement en Agile, car on ne peut pas se permettre de perdre une vie pour Tesla, ou pour Microsoft on ne peut pas se permettre qu'il y a un bug alors que des millions de gens l'utilise. Ça va donc rester des mois et des mois dans les tests. Mais ce n'est pas 100% Agile du coup puisqu'on ne va pas livrer pendant ce temps-là.

Agile a des avantages, des qualités par rapport aux autres méthodes, mais la complexité du monde aujourd'hui ne permet pas aux entreprises d'utiliser Agile correctement. Ça fonctionnera pour de petits projets, où l'impact des bugs est minime, ou alors il faudra beaucoup de tests et même avec des tests on ne va jamais arriver à 100% de la solution.

Agile ne va pas amener à l'échec, ça va amener au succès, on ne saura pas à côté de la plaque par rapport aux méthodes qu'on utilisait avant. Mais il va falloir faire beaucoup de tests.



## Appendix 7 : Interview Christophe Tilquin

C.T.: Co-founder and business analyst at Cenotia.

*Pourriez-vous vous présenter en quelque mots ?*

C. T. : Cenotia est une entreprise qui a été créée il y a 2 ans. C'était d'abord une entreprise qui était spécialisée dans les conseils en transformation digitale. Et puis, on s'est rendu compte qu'en fait, il n'existait pas sur le marché des logiciels qui répondaient aux besoins des clients que l'on avait. Et les clients que l'on avait, c'était principalement dans les professions libérales et des bureaux d'étude, d'expert immobilier et beaucoup de cabinets d'avocats. Ils avaient des logiciels très désuets et qu'ils cloisonnaient dans un environnement purement Windows. On a donc décidé de développer des logiciels par nous-même, en se disant que, comme on ne trouve pas sur le marché, et qu'on avait les compétences pour le faire, on s'est lancé là-dedans.

Cela a donc abouti maintenant à 2 produits que l'on a : un produit qui est spécifiquement pour les cabinets d'avocats et un autre qui est plutôt pour les professions libérales au sens large. Le premier s'appelle Legaclik, si vous allez sur [legaclik.com](http://legaclik.com) vous avez les informations. Et l'autre logiciel s'appelle Experta et lui est plutôt très généraliste pour les cabinets.

Pour nous, les méthodes Agiles, comme ça en fait, elles se sont imposées d'elles-mêmes. Car comme on est une petite équipe, on ne pouvait pas traiter tous les problèmes en même temps. On a à chaque fois brossé les grosses questions sur lesquelles on devait travailler et puis, chaque fois, on a pris les questions une à une et on a développé autour de cette question-là et on a fait un peu la méthode Scrum en faisant très régulièrement des réunions où on regardait à chaque fois pour ce petit paquet de tâches qu'est ce qui était fait. Et chaque fois revoir la priorité, car qu'on avait un ensemble de tâches le problème pour nous était surtout la gestion de la priorité. On se rendait compte qu'on mettait quelque chose en priorité 1, mais qui demandait trop de temps de développement donc on remettait en arrière et on essayait d'avancer sur des choses. On changeait les priorités sur des choses plus rapides à développer comme cela on avait à chaque fois des tâches de fonds et des tâches à court terme.

Donc pour nous ces méthodes Agiles, c'est juste intuitif. On ne voyait pas très bien comment on pouvait travailler autrement. Pour des plus grosses équipes, je crois qu'effectivement on peut imaginer des systèmes avec des projets où on voit toutes les tâches qui suivent avec des gens qui travaillent sur le long terme, sur le court terme et qui ne sont pas obligés de faire très

régulièrement des meetings pour pouvoir avancer et chaque fois réévaluer toutes les tâches qui sont en cours.

*Vous avez toujours utilisé les méthodes Agiles alors ?*

C.T. : Oui en fait on a toujours utilisé ces méthodes-là. Et c'est vrai qu'on a vu certaines formations sur les méthodes Agile, on doit voir un peu ce qu'on entend par méthode Agile. En effet, il y en a certaines que je ne connais pas du tout, mais les basiques ont l'idée d'être très réactifs tant pour le développement, mais aussi quand on a commencé à commercialiser l'application. Chaque fois, qu'un client nous demande d'ajouter quelque chose ou bien quand on fait une amélioration il y a du code qui n'allait pas ailleurs et qui avait un impact qu'on n'avait pas imaginé, on est obligé de réagir comme cela très vite. Donc, pour la méthode Agile c'est beaucoup de communication entre nous pour pouvoir réévaluer la priorité des tâches. C'est un peu comme cela que l'on travaille aujourd'hui.

*Et quand vous avez commencé à utiliser Agile, comment cela s'est-il passé ?*

C.T. : On a directement trouvé un outil qui nous convenait très bien en fait. C'est basé sur la méthode Kanban qui nous permettait chaque fois de pouvoir faire des listes de ce qui était à faire ou ce qui restait à faire, ce qui est en cours, ce qui est à tester et ce qui était terminé. On faisait passer les tâches comme cela d'un statut à l'autre. Et aussi dans la liste on pouvait chaque fois réévaluer en remplaçant les tâches. On a utilisé un outil - en fait on en a utilisé 2. On a un outil qui s'appelle Trello qu'on utilise tout le temps. Et alors aussi, mais cela c'est plus pour la partie développement, c'est un outil qui s'appelle Pick Bucket qui permet aussi de gérer un peu cela. Mais il ne donne pas des tableaux de bord, ça c'est plutôt ceux qui sont dans le développement pur. Moi je ne suis pas dans le développement qui utilise beaucoup l'outil Pick Bucket. Et alors, cet outil Pick Bucket, on l'utilise plutôt comme un outil de ticketing. Donc chaque fois qu'il y a une question ou un problème qui vient d'un utilisateur, directement cela entre dans ce système-là et comme cela on peut suivre ce ticket. Tandis que Trello, c'est plutôt pour la partie stratégie de développement. Donc quand on est sur un gros bloc et bien on essaye de le faire avancer et on se rend compte que bien non ça ne va pas. On doit diviser ce bloc-là et puis on doit le faire reculer. On est tout le temps sur une approche plutôt Kanban c'est ce qui nous convient pour notre équipe.

*Et quand vous avez commencé à utiliser Kanban, avez-vous eu des difficultés au début ?*

C.T. : Non, quand on s'est réuni on était déjà tous des profils seniors on connaissait tous ces outils-là et donc on a juste choisi la méthode que l'on voulait prendre. Mais comme je dis la méthode Agile pour des petites structures elle s'impose d'office. Et d'ailleurs, je ne sais pas qui a commencé à créer toutes ces méthodes Agiles, mais à mon avis ce sont des gens qui au départ étaient quand même dans des petites structures et qui étaient obligés de fonctionner comme cela.

*Et la méthode Scrum, quand l'avez-vous implémentée ?*

C.T. : Cela nous a aidé à structurer un peu notre démarche. On travaillait grosso modo comme cela, mais en fait on s'est rendu compte que cette façon de travailler ressemblait à la nôtre. Disons, on l'a plus structurée en disant : « Oui cela c'est la grosse pile de tâches, ça c'est la petite pile de tâches » et on s'obligeait en fait, suivant la méthode, à se voir plus souvent. Parce que c'est cela qui est bien, c'est organiser des mêlées et donc cela nous a juste aidé. Intuitivement, on a travaillé comme cela et cela nous a quand même aidé à nous structurer. J'ai suivi une formation quand même pendant 3 mois, c'était aussi organisé par la Louvain School of management qui était sur la gestion du changement. On s'est rendu compte, comme nous c'est aussi des logiciels ERP qu'on met en place pour des sociétés de service, c'est vrai qu'on ne gère pas les stocks de matières premières, etc., mais c'est un ERP. Depuis le premier contact avec le client jusqu'à la fin ils sont tout le temps dans l'application. Et pour gérer leur facture et pour leur comptabilité et pour envoyer leurs mails, pour gérer le dossier au niveau documentaire ils n'utilisent que cet outil là. Donc on s'est rendu compte qu'il y avait quand même de gros problèmes de gestion.

Et aussi, on avait pris toute une série d'initiatives suivant cette formation que j'ai suivi il y a 2 ans et je me suis dit qu'on était un peu dans le même cas que dans Scrum.

En fait c'est une discipline, et il y a des règles à suivre. Et comme ça, c'est structurant en se disant que bien on a une méthodologie à suivre. La gestion du changement, c'est la même chose, tout ce que je disais avant de manière un peu intuitive, je sais qu'il y a des sources derrière qui sont là pour bien justifier le fait qu'on est sur des bons rails.

*Et donc la gestion du changement c'est pour vous, ou c'est pour vos clients que vous en avez besoin ?*

C.T. : Pour les clients, pour surtout gérer les résistances au changement, qui sont énormes. Ils aimaient bien le logiciel avant et parfois ils ont des blocages au niveau du secrétariat. Parfois,

ce sont des blocages pour certains avocats « Qu'est-ce que c'est ce système-là, on ne sait même plus faire ça ». On a une équipe, on a un groupe de clients, j'ai encore eu le cas vendredi passé, il y a deux secrétaires, il y en a une qui est pour tout ce qui est nouvelles technologies. Et l'autre en revanche, m'a dit directement qu'elle a eu un problème pour envoyer un mail et elle m'a téléphoné en disant « Ça ne va pas, vous enlevez tout et vous remettez tout comme avant ». Pour bien montrer, que c'est quelque chose qui est très fort.

Pour Scrum, c'est intéressant de voir qu'on n'est pas les seuls à rencontrer ce type de problème, gérer ce genre de conflits, ce genre de résistances. Soit ça peut se gérer d'un training collectif, soit pour certaines personnes, il faut un coaching personnel. C'est grâce à ce type de formation qu'on peut voir qu'on est dans tel cas et qu'il ne faut pas commencer à chercher midi à 14h. Ces personnes on va les prendre en individuel, mais ça va nous prendre 3-4h, soit vous le faites vous-mêmes soit on le fait et puis on vous facture. Et quand même les utilisateurs sont assez réceptifs, parce qu'ils voient quand même que ce n'est pas aussi évident et quel que soit la qualité du système qu'ils vont mettre en place, ils vont toujours trouver des gens qui vont être réfractaires.

Dans la gestion du changement, on voit surtout que le pourquoi des gens qui sont réfractaires, ce n'est pas parce qu'il y a un nouveau système qui se met en place, c'est parce qu'il y a une perte de pouvoir dans l'organisation. Avant le système était très compliqué, il y avait qu'une secrétaire qui savait exactement tout ce qui se passait, maintenant tout le monde sait se débrouiller, et pour elle c'est une perte énorme de pouvoir dans l'organisation. Ce n'est pas le fait que le logiciel ne va pas bien. Et avec un ERP, même si c'est une équipe de 5-6 personnes on est déjà confronté à des choses comme ça.

Nous, on a des logiciels standards, on a une ossature de base et puis on peut adapter aux besoins du client. Vous avez certainement entendu parler de workflow, de flux de travail. C'est très spécifique de définir par exemple leur flux d'affaire. Quand ils ont un client par exemple et qu'ils ont un dossier qui s'ouvre, dans lequel ils créent un document pour des conclusions, ils encodent évidemment leurs prestations, le nombre d'heures qu'ils passent sur ce dossier. Mais ils peuvent définir aussi des flux de travail en disant « ce dossier nous il passe par tel type de statut quand, par exemple, ce sont des clients d'assurances, ou des clients particuliers, ou des entreprises c'est comme ça ». Et comme ça, chaque fois qu'ils changent les statuts des dossiers, il y a directement des choses qui se passent : un document de contrôle

qui est fait, ou pour un dossier il y a une lettre d'envoi d'ouverture de dossier qui s'envoie automatiquement. Mais ça dépend un peu comment chaque client veut voir les choses.

*Donc vous m'avez dit que c'est venu intuitivement de travailler en Agile, est-ce que vous aviez des attentes particulières ? Peut-être par votre expérience ?*

C.T. : C'est vrai que je n'ai pas été très étonné, ou enthousiasmé par les méthodes ou les formations que j'aie suivies, parce que je n'ai rien appris vraiment de nouveau qui nous permettrait d'aller plus vite. Mais c'est peut-être aussi par rapport à notre petite structure actuelle. Mais moi aussi ce que j'ai vu aussi bien pour la gestion du changement, que pour cette méthode-là c'est pour mieux nous structurer et d'avoir un background. Comme ça quand on me demande de quoi je parle je peux directement dire, voilà c'est dans tel bouquin, c'est expliqué clairement là.

*Vous avez pu évaluer vos projets ?*

C.T. : Non, ça on n'est pas encore à ce stade-là non. C'est vrai qu'on est systématiquement en retard et ça c'est un problème. Mais c'est vrai que nous mettons un calendrier un peu trop drivé par le client pour qui tout est toujours urgent. Et alors, on se fixe parfois des deadlines qui ne sont pas si faciles à suivre parce que dans notre équipe on a une personne qui est plus qu'extrêmement capable qui va à une vitesse extraordinaire, et d'autres qui développent à une vitesse normale. Alors voilà, je me base souvent sur celui qui a une vitesse extraordinaire, car on a quand même assez d'empathie pour les utilisateurs et on essaye de leur répondre au plus vite. Mais bon, on est aussi une jeune équipe, et donc pour nous chaque nouveau client est aussi une référence donc on veut l'avoir, et on veut la garder. Mais peut-être que dans un an on sera plus cool et on dira, on le fait, mais ce ne sera pas dans le mois, mais dans les 3 mois qui arrivent et on commence tout doucement à faire ça. Mais c'est vrai que s'évaluer aujourd'hui, oui on l'a fait, mais on est systématiquement en retard sur tous nos time flow et ça on le voit tout de suite. Quand on voit qu'on est en retard, ce que je fais c'est que je contacte le client et je leur dis : « Voilà on a eu ceci, et ceci, et donc voilà globalement le problème et donc on a un retard ». Donc cette gestion du changement, c'est surtout ça que ça m'a appris c'est qu'il ne faut surtout pas avoir peur de communiquer avec les gens. Peu importe, si le projet est en avance ou en retard par rapport au timing ou autre chose, ou des fonctionnalités. Mais il faut absolument au sein d'une équipe et entre équipes, externes, sous-traitants - parce que oui on a des sous-traitants qui développent pour nous - et clients, c'est être au top de la communication. Communiquer, communiquer, communiquer, c'est très

important que tout le monde a bien compris les informations et ça c'est harassant, parce que c'est beaucoup de boulot. Parfois il faut reformuler des choses d'une manière différente. Et parfois, si on pousse la communication très loin comme on l'a fait, bien moi j'ai encore avec un cabinet d'avocat qui ne veulent pas lire ou comprendre ce qu'on explique. Et même en le faisant très fort [communiquer], on peut avoir des problèmes. Oui on l'a fait, on sait qu'on l'a fait, mais c'est surtout au niveau des deadlines qu'on doit être plus confiant et ne pas se mettre des deadlines qui sont vraiment trop justes. Car il peut arriver vraiment n'importe quoi dans un projet. Par exemple, on a eu un développeur qui a eu une crise d'appendicite et quand on est une petite équipe qu'est-ce qu'on peut faire contre ça... Il y a des choses comme ça. Et puis il y a le fait aussi que logiciel il a certaines tailles. Au début, le logiciel est d'une taille respectable et puis quand on fait des modifications on peut bien maîtriser tous les impacts et maintenant on a une application qui commence à devenir très grosse et donc quand on impacte et on change les codes d'un point de vue de la facture on ne se rend pas compte que parfois les scans d'entrée – parce qu'ils peuvent être parfois reliés avec des factures fournisseurs- il peut y avoir un impact. On doit peut-être plus avoir de tests niveau évaluation, on ne fait pas assez de tests. Donc il y a les deadlines qu'on ne respecte peut-être pas assez, mais un problème qui devient de plus en plus récurrent c'est le problème de tests.

*Et donc les retards, c'est un impact dû à Scrum, dû à la méthodologie ou à autre chose ?*

C.T. : Ça n'a rien avoir avec les méthodes. On s'en rend compte avec les mêlées et cette communication intensive. Quand la tâche devient trop complexe on revient en arrière et on essaye de finir vite ce qui est facile et puis on revient sur la tâche plus complexe. Et parfois quand la tâche est super prioritaire, alors on remet en arrière les plus petites tâches. C'est ce système de communiquer souvent, et être très sélectif dans les tâches. Il y a quand même un screening au début dans les tâches qu'il y a à accomplir. On l'évalue, mais on ne le quantifie pas. On a peut-être encore beaucoup de choses à apprendre sur les méthodes Agiles qui pourraient nous aider un peu. Mais pour l'instant, on est un peu à saturation de ce qui a à faire au niveau développement. Et donc il faut peut-être qu'on ait, comme c'était le cas il y a 6 mois, un petit bol d'air pour pouvoir se demander : « Tiens qu'est-ce qu'on peut améliorer dans la manière dont on travaille ».

*Pour la communication, est-ce que Scrum ou Kanban vous aide beaucoup là-dedans ?*

C.T. : Scrum évidemment. Kanban, que nous allons d'ailleurs mettre dans notre application. Parce que je trouve que ce qui est très pertinent dans cette approche c'est que vous travaillez directement sur le tableau de bord. On en parle beaucoup en disant que voilà on peut directement déplacer leurs tâches. Et ce qui est intéressant, c'est qu'on travaille directement sur le tableau de bord par rapport à des logiciels de type standard ou c'est encoder des tâches de telle date à telle date et ainsi de suite, et puis seulement on voit ce que ça donne au niveau du planning. Tandis que là on travaille directement sur le planning. Maintenant, on est une petite équipe donc si on est 15 ou 20 on ne peut pas fonctionner comme ça. Donc en tout cas, pour des petites structures l'outil est très efficace. Et Scrum alors c'est plutôt pour nous obliger à communiquer sur les petites difficultés qu'il y a. Et parfois, quelqu'un dit bien voilà j'ai fait ça et parfois on teste et en fait le problème est plus grand que ça. Pour une personne, ce qu'il a fait est réglé et puis pour d'autres, le fait qu'on se voit ça permet de voir quels développements qui sont faits, ça s'appelle des pitches – quand on doit présenter de ce qui a été fait. Ça oblige quand même à chaque fois de retester et de se rendre compte des choses. Alors que si le développeur avait mis ça en attente pour quand le projet était terminé, on se serait rendu compte beaucoup plus tard qu'il y a une montagne de problèmes. Donc, le fait de faire ces mêlées systématiques ça aide quand même vraiment beaucoup. Et il faut s'obliger à le faire. Parfois on vient en me disant que ce ticket, c'est fait, c'est résolu. Et puis moi, je le passe à la moulinette et je me rends compte que ce n'est pas résolu. Il y a un petit truc qui ne va pas et ce genre de choses prises et correctement documentées, qu'on se dit qu'il y a quand même de la recherche derrière. Et même chose pour la gestion du changement si toutes ces approches existent c'est qu'il y a bcp de gens qui ont déjà rencontré ce type de problème. Ça ne consiste pas à réinventer la roue et se demander comment on va construire ça. Bien, on applique la méthode qui est proposée.

*Pour vous, est-ce que l'adoption des méthodes Agiles c'est suivre un certain objectif ?  
Est-ce qu'il faut avoir des attentes particulières ?*

C.T. : Oui. Moi j'ai été à l'école japonaise pendant plus de 10 ans. Et alors, je trouve que ce qui manque le plus en occident, c'est la méthode PDCA. C'est : Plan Do Check and Act. Et ça c'est quelque chose qu'on ne fait jamais en Europe. On planifie, et on fait quelque chose, ça marche très bien, ou ça ne marche pas et on recommence, on fait autre chose. Mais on ne met jamais en place quelque chose qui est monitoré et pour pouvoir avancer après. Et la méthode PDCA, la méthode que les japonais utilisaient chez eux et qui par une communication absolument lamentable n'a jamais réussi à passer en Europe. Parce qu'on est

resté sur ce même problème. Ça, c'est une petite parenthèse, il y a eu des conflits terribles à cause de cette différence de culture entre la partie japonaise et la partie européenne. Il y a aussi la partie américaine, mais alors là n'en parlons pas c'est encore pire, au niveau que c'est à chaque fois une culture différente. Et donc pour cette pratique PDCA, la partie check c'est la partie la plus importante et que justement en utilisant une méthodologie, on peut se fixer des indicateurs pour pouvoir checker et pourvoir améliorer ce qui a en place. Ça c'est ce que je vois par rapport à l'aspect encore une fois structurant des méthodes. C'est que si on a un objectif spécifique à suivre, ce serait de pouvoir mieux s'évaluer, donc il y a la partie structure, mais il y a la partie aussi, comme on suit un cursus, c'est de pouvoir mieux s'améliorer par après. Et ça, c'est un truc que je trouve important et que les européens encore une fois n'ont pas compris. J'avais de très bons collègues japonais et ils le reconnaissaient. Nous la seule consigne d'entreprise c'était : « Bon, maintenant on doit appliquer le PDCA ». Mais qu'est-ce que ça veut dire au quotidien ? Qu'est-ce que ça veut dire ? Donc je me suis renseigné et je faisais attention au quotidien quand on lançait un projet de mettre des indicateurs intéressants. Parce que quand tu lances un projet tu peux mettre des indicateurs, tu peux en trouver plein, mais ça demande quand même un exercice de trouver des indicateurs que tu peux suivre et que tu peux donner à des project leaders pour se rendre compte si le projet se réalise bien et surtout après quand on le met en marche – enfin le produit ça peut être une méthodologie, ça peut-être un tas de choses – de vérifier que les choses s'améliorent, que les choses avancent bien. Je pense que cette méthodologie ça apporte cet aspect-là. Et moi je me suis toujours fait traité par les européens de très formaliste. Parce que quand tu parles de méthodologies tu deviens formaliste. Alors que pour eux ils font une réunion, ils font ça comme ça. Et si ça n'a pas marché, aller on recommence. Alors que ce n'est pas pour rien si Toyota ou Yamaha, sont des leaders mondiaux sur leur marché. Et ils ont cette approche du check. Et pour la méthode PDCA j'ai déjà lu sur internet des articles et je me disais, le gars il n'a rien compris. Il disait que le PDCA vous devez checker. Mais en fait il faut voir le sens qui est derrière le check, le monitoring.

*Est-ce que vous avez également implémenté le PDCA ici ?*

C.T. : Le PDCA moi certainement, on le fait, mais comme je le disais, au niveau de notre approche, quand on suit la logique c'est surtout la partie structuration. Mais je voudrais qu'un des objectifs ce soit qu'on puisse mieux se mesurer et s'évaluer. Je vois bien que je suis le seul à le faire dans l'équipe. Parce qu'encore une fois, avec l'esprit européen de mes collègues, ils me voient trop formalistes par rapport à ça alors que pour moi, c'est justement

grâce au fait de monitoring qu'on a réussi à s'améliorer. Car j'ai pu dire des faits qui ne marchaient pas bien, mais j'ai aussi pu quantifier un petit peu. Et en fait ce travail de check, il n'y a aucun européen qui veut le faire. Pour la méthodologie, le gros problème qu'il y a c'est que trop de gens se disent « moi je suis dans l'action Christophe ». Tu veux être dans l'action ? Bien heureux pour toi, car tout le monde voudrait être dans l'action. Mais le PDCA justement, c'est dire que tout le monde doit être dans l'action, mais aussi faire le check de cette action. Et c'est justement ce check que les gens ne veulent pas faire. Et là, je pense que les méthodologies Agiles donnent l'impression qu'on est direct dans l'action, mais par contre, ces méthodes Agiles c'est également une méthode donc ça doit aussi permettre de s'évaluer et d'avancer et ça les gens ne veulent pas faire. C'est juste la partie activité qui intéresse. Je délire peut-être en disant ça, mais c'est un peu mon impression.

*Par rapport à votre expérience que pensez-vous des méthodes Agiles en général ?*

C.T. : C'est ce que j'ai dit au début, pour moi c'est intuitif. C'est assez intuitif de se rendre compte qu'on a une pile de tâches, on en sélectionne une partie et on essaye de démêler cette partie-là avant d'essayer de démêler une autre partie, voir travailler uniquement sur la stratégie. Ce qui importe sur les méthodes Agiles c'est se rendre compte qu'il faut travailler pour la stratégie. Les méthodes Agiles pour moi ce sont plutôt pour moi des tactiques. On est plutôt dans la tactique, et il faut être tactique, pragmatique aussi et il faut de la stratégie. Mais il faut aussi être dans le pratique.

*Est-ce que vous connaissez l'Agile manifesto ? L'esprit Agile, est-ce qu'on vous en a parlé ?*

C.T. : Je dois dire que je ne suis pas la personne la plus érudite pour parler des méthodes Agiles parce que je n'en ai pas suivi, je ne me suis pas documenté énormément. Pourquoi ? Parce que pour nous il y a une méthode comme le Scrum et des outils pour Kanban, je pense que ça nous permet déjà d'être très efficace. Comparé déjà quand je vois nos concurrents, parmi nos nouveaux clients nous avons beaucoup de transfuges. Des transfuges ce sont des gens qui utilisaient avant un autre logiciel et qui maintenant utilise notre logiciel. Donc on va récupérer, nous ce qu'on fait, ce sont des ERP ce qu'on appelle des logiciels intégrés, donc ce sont des fonctions qui sont intégrées dans une application et nos clients sont en grande partie des transfuges, qui donc utilisaient déjà des outils de gestion. Et non pas des gens qui ont des outils non pas atomisés, non intégrés au départ ou ils font des choses sur Excell, des choses dans Word, des bases de données dans Access et ils arrivent à chipoter comme ça avec des

duplications dans tous les côtés. Et ces clients-là nous disent quand même qu'on est beaucoup plus rapide que beaucoup de nos concurrents pour répondre à des besoins ou ne fut-ce qu'à des problèmes qui peuvent arriver. Je me rends compte maintenant que plus en plus, quand on fait une nouvelle release, on souffre. Parce que on ne teste pas assez.

*Pour ces projets de transfuges, est-ce que c'est plus difficile ou plus facile à utiliser avec la méthode Agile ?*

C.T. : Ici quand je parle de transfuge, je veux dire que l'équipe utilise un certain outil de gestion et nous ce que nous faisons, nous récupérons les données qui sont dans l'ancien outil, et nous le mettons dans le nouveau pour répondre davantage aux besoins que l'ancien outil n'arrivait pas à répondre. Mais oui, je me rends compte que quand on travaille par petit sprint comme ça que c'est quelque chose que les clients aiment. Et que nos concurrents ne veulent pas faire, ou ne savent pas faire. Ils se trouvent leaders sur le marché, enfin ils ont abaissé fortement le prix de leur abonnement donc ils se rendent compte qu'il y a un petit souci. Mais au niveau des méthodes Agiles, je me répète encore, mais c'est l'aspect structurant chez nous qui se justifie chez nous qu'on doit continuer comme ça. Et ce qu'on ne fait pas assez, c'est que ces méthodes doivent servir à monitorer. Nos clients voient qu'on oublie rien, c'est ça aussi grâce au Kanban et aux méthodes, c'est qu'il y a des éléments parfois qu'on ne travaille pas, mais ce n'est jamais oublié. C'est mis dans une case à côté, mais on sait comment y revenir très vite donc s'il y a un truc. Mais on n'est pas fort érudit sur toutes les méthodes qui devraient exister, peut-être qu'on devrait le faire dans un an, ou deux ans quand l'équipe sera un peu plus grande ou quand on aura moins de souci de tout ce qui est test de nouvelles releases, on doit trouver une façon et ça existe d'ailleurs, des programmes de tests pour rentrer toutes des procédures et en fait ils testent tout. « Envois d'un mail », et il envoie un mail ou, « créer un document » et il fait toute la batterie de tests et ça il faut tout configurer. Mais ça n'a rien avoir avec Agile.

*J'ai terminé avec mes questions, je ne sais pas si vous avez des remarques, conseils ou questions ?*

C.T. : Le conseil que je dirais c'est de creuser dans la littérature et voir le lien qu'il y a entre le PDCA et les méthodes Agiles, la manière dont fonctionne les japonais, les asiatiques en général, et surtout les japonais. Ça rappelle un petit peu, ça revient dans le ACT il y a une boucle qui se fait un petit peu dans les méthodes Agiles. Je ne sais pas si quelqu'un a déjà fait le rapprochement, mais ça m'a pris spontanément quand j'ai parlé avec toi donc je me dis que

c'est qu'il doit quand même y avoir un lien, et c'est intéressant de parler d'autres cultures comme de grandes sociétés comme Toyota. Parce Toyota c'est quand même les premiers au niveau mondial. Il y a une autre méthode utilisée par les japonais, mais ça c'est plutôt l'approche tableau de bord. Ce qu'il faut retenir c'est que les méthodes Agiles c'est pour améliorer le tactique, l'approche tactique quand on travaille avec des clients ou en interne ça doit être mêlé avec une approche stratégique qui est au-dessus. Tu ne peux pas faire que des méthodes Agiles, ça doit être complémentaire à une approche stratégique, il faut avoir les deux. C'est comme, quand tu as la gestion du changement, que tu gères un projet, tu dois avoir une gestion de projet technique classique, et en plus tu mets la gestion du changement au-dessus. Mais tu ne peux pas faire que de la gestion du changement ou que de la technique, c'est complémentaire. Et pour les méthodes étrangères, et les méthodes asiatiques que je connais un peu mieux. Parce qu'aux Etats-Unis ils ne m'ont jamais épaté, c'est « everything is fine, you are the best, what you propose is just unbelievable but just do nothing ». J'ai vu une vidéo extraordinaire sur les américains, d'un artiste, un chanteur non-américain qui disait que les américains ils ne connaissent qu'eux, ils ne connaissent pas les autres. Et donc quand tu vas là-bas c'est toujours « wonderful ». Eux, ils ne te disent pas vraiment comment ils travaillent, comment ça fonctionne. Je n'ai jamais rien appris d'eux, car ils ne veulent jamais rien apprendre des autres, donc les méthodes japonaises. Les européens ils veulent bien les asiatiques. Un second point chez les asiatiques, pour la partie stratégique, une méthode redoutable, mais bête, quand tu leur présentais quelque chose c'était toujours en format A3. C'est lié aussi au check, c'est la méthode Toyota.

Pour moi la méthode Kanban, c'est un tableau de bord. Ça aide à faire des connexions dans le management que tu ne t'attends pas. Ils sont très inefficaces, car ils doivent travailler des heures pas possibles, mais ils ont des outils de management très intéressants, efficaces. En management, il ne faut pas simplifier à outrance, c'est le problème des européens. Les japonais eux voudront des chiffres derrière. Il y a un manque de communication entre cultures aussi, mais ça dépasse largement les méthodes Agiles.

*Il y a également la méthode Lean qui vient des japonais et de Toyota.*

C.T. : La méthode Lean, je ne sais pas je vois ça avec éliminer, et pour les cadres des entreprises c'est éliminer les gens, et ça ne va pas.

*Qu'est-ce qui pourrait être une approche stratégique, que vouliez-vous dire par là ?*

C.T. : Une approche stratégique c'est le contraire de la tactique. Ça part d'un point de départ, quelle est notre vision, quel type de projet on a. Nous notre stratégie c'est d'être une entreprise qui fournit au prix du standard, du sur-mesure parce qu'on sait que chaque client a un besoin stratégique. Après tu dois déterminer cette vision. Un ERP va te permettre d'extraire énormément de chiffres, mais si tu n'as pas de stratégie alors quel chiffre est-ce qu'il faut sortir. La stratégie est essentielle pour voir les grands objectifs, que tu vas suivre pour les entreprises. Par exemple, est ce que tu veux faire des bières low-cost ou la meilleure bière du monde. Je dis que c'est complémentaire à une méthode Agile, tu auras un objectif, tu auras une série de tâches, et dans cette série de tâches tu vas choisir quelles tâches tu vas faire en premier lieu. Mais donc tu as toujours vision, objectif stratégique, objectif tactique/opérationnelle. Pour moi les méthodes Agiles c'est de l'opérationnel. Tu ne peux pas avoir d'opérationnel sans stratégie.

Et le lien entre les méthodes Agiles et les stratégies, c'est que les méthodes Agiles vont te permettre d'implémenter beaucoup plus vite la stratégie de ton entreprise.

Il y a des objectifs intrinsèques à la méthode comme livrer plus vite, mais il y en a aussi qui sont stratégiques dans l'entreprise.

## Appendix 8 : Interview Franz Hegemann

F.H. : Service Delivery and Change Manager

*Pouvez-vous vous présenter en quelques mots ?*

F.H. : J'ai été diplômé en '88 en informatique. J'ai commencé ma carrière comme développeur d'abord, après comme analyste, et après j'ai évolué comme projet leader. À ce moment-là, c'était la méthodologie Waterfall, c'était sur le mainframe. J'ai travaillé pour des grands clients, par exemple : « KBC », qui à l'époque était encore « Kredietbank ». À un certain moment j'ai changé ma carrière vers Ressources Humaines donc c'est tout à fait différente technologie vers des gens. J'ai travaillé pendant certaines années comme manager Ressources Humaines. Après j'ai travaillé pendant 5 ans comme Business Unit Manager donc c'était un peu plus commercial, mais toujours dans des sociétés informatiques et en fait toujours avec des gens qui travaillaient sur le mainframe et évidemment sur la méthodologie Waterfall. Et puis un jour, je suis devenu indépendant. En fait, j'avais décidé de devenir interim manager parce que j'aime les variations dans ma carrière. Et c'est ainsi que je suis arrivé chez P&V, comme interim manager. D'abord, c'était purement « at interim », il y avait un vieux manager qui allait en pension et on n'avait pas une bonne idée de comment gérer toute l'informatique. Parce qu'à ce temps, il y avait encore la partie Vivium à Anvers qui était séparée côté IT et on avait quelques grands projets pour migrer la société Vivium vers P&V ou l'autre direction. Et il y avait déjà une idée pour un grand projet pour gérer les Claims (les sinistres), mais on n'avait pas une bonne idée comment on allait faire cela. Parce qu'on savait bien que la manière où on travaillait là à l'époque n'était pas la bonne. Les projets prenaient trop de temps, c'était trop long, trop d'argent, les gens étaient un peu bétonnés dans leur situation. Donc beaucoup de questions, de bonnes idées, bonne vue sur les choses qui n'allaient pas bien, mais pas une idée comment on allait faire dans le futur. Je suis devenu, à ce moment-là responsable de tout ce qui est sinistre. C'était une équipe IT sinistres, c'était une équipe d'une centaine d'informaticiens uniquement. Pas les analystes, ils étaient séparés dans une autre équipe. Et là, j'ai commencé à développer des idées pour améliorer les choses. Évidemment, j'ai travaillé avec le manager des analystes Luc Daelemans. Et on avait les mêmes idées à propos de : « Comment peut-on améliorer les choses ? ». Donc, étape par étape, nous avons démarré, nous avons changé des petites choses, parfois des plus grandes étapes. Et un jour, le grand projet « Sinistres » avait décidé de faire avec un bureau externe. Donc on a décidé d'acheter un outil américain et avec l'outil il y avait une société de consultance qui a amené la méthodologie Agile. Et on s'est dit que c'est peut-être que c'est

cela qu'on doit faire dans le futur. Mais avec la totalité du personnel IT, c'était 200 – 250 personnes du côté IT, plus les analystes qui avaient l'habitude de travailler dans la méthode Waterfall, une manière très archaïque. Ce n'était pas évident. En fait, ça c'est un peu l'histoire. J'ai quitté P&V juste après la mise en place de la méthodologie Agile hybride, donc pas 100% Agile et les résultats maintenant je ne les connais pas.

Maintenant, aujourd'hui je travaille dans une société IT de nouveau, mais je suis seul dans ce projet et j'utilise un peu les idées Agiles. Mais parce que je suis seul je peux faire ce que je veux. J'ai beaucoup de stakeholder chez des clients, chez le fournisseur d'un outil que j'utilise, c'est moi qui décide tout, mais en fait c'est l'idée Agile que j'utilise aujourd'hui pour délivrer des petites choses, étape par étape. C'est ce que je fais maintenant.

*Donc chez P&V, lorsque vous utilisiez la méthodologie Waterfall, y avait-il des problèmes ? Pourquoi avoir changé de méthode ?*

F.H. : Un des grands problèmes que j'avais découvert assez vite quand j'étais chez P&V, c'est que la fin d'une tâche était toujours une semaine après la deadline donc et avant il n'y avait personne qui pouvait dire combien cela allait coûter, combien de temps ça allait prendre. Les grandes discussions, le business n'était jamais content, ils disaient toujours : « L'IT dit toujours non, ils ne savent pas faire ou ils n'ont pas le temps. Ils ont un grand budget, mais qu'est-ce qu'ils font avec ce budget ? Je ne vois rien ».

J'ai eu des discussions avec les membres du comité de direction : « Vous dites toujours non et vous ne faites rien ». Cela coûte beaucoup. Et en fait la transparence, c'était un grand problème.

Ils ne savaient jamais de quoi on s'occupait du côté IT, ils ont beaucoup d'argent, mais qu'est-ce qu'ils font ? Et à la fin d'un projet, il y avait toujours un gap entre ce qu'on avait délivré et l'idée que le business avait du résultat avant. Ça c'est un grand problème. Ils considéraient que l'IT ne faisait pas ce qu'ils avaient demandés. Alors quand tu dois déjà délivrer aujourd'hui et demain il faut déjà faire des changements, ça ne va pas. C'était donc un grand problème de la méthodologie Waterfall.

*Quand vous avez implémenté la méthodologie Agile, aviez-vous des objectifs particuliers ?*

F.H. : L'idée c'était d'améliorer évidemment. Mais on savait très bien que changer les méthodologies, mais surtout les gens, 250 personnes à une méthodologie qui était totalement

différente de ce qu'on faisait déjà pendant des dizaines d'années, ça allait être très, très, très difficile. Notre projet allait durer 1 an, 1 an et demi. Et à partir du premier changement qu'on voulait mettre en place et le moment où le changement était en place, ça n'était pas adapté. Le projet était en fait très lourd. Je ne sais pas si on avait vraiment des KPI pour déterminer si ce qu'on avait mis en place avait amélioré la situation d'aujourd'hui par rapport à avant.

*N'aviez-vous pas d'attentes sur ce qui allait se passer, sur d'éventuels impacts ?*

F.H. : Sur les résultats, je n'étais pas vraiment impliqué au niveau de la Direction donc peut-être. Le comité élabore, il avait des idées ou des KPI, mais moi je n'étais pas au courant. On avait un certain objectif, c'était mettre en place une nouvelle organisation, en plus de la méthodologie. La méthodologie, c'était une chose, mais toute l'organisation avait changé en même temps dans la méthodologie. On avait des sous-projets pour mettre en place certaines choses avec des formations évidemment. Mais aussi avec une équipe de « Magnificent Seven », avec une autre équipe de consultance aussi, chaque équipe était spécifique pour une partie du changement. Et pour chaque partie on avait une certaine attente et un certain objectif pour le mettre en place. Je ne pense pas qu'on avait une idée de l'objectif du grand projet NEO.

*Quelle méthode Agile avez-vous implémenté chez P&V ? Et comment cela s'est-il passé ?*

F.H. : La méthode Scrum. Il y avait Kanban aussi, pour gérer le day by day, ongoing business. Mais pour les nouveaux projets c'était Scrum. L'idée était d'avoir des équipes d'une dizaine de personnes, on appelait la méthodologie Agile : « Hybride ». Après, il y avait un business analyste qui suivait le projet, mais c'était surtout la partie développement qui était vraiment Agile. Donc ce n'était pas 100% Agile. J'espère que maintenant ils font un peu plus, mais à ce moment-là on avait vraiment le sentiment que c'était trop difficile d'avoir tout le monde. Des centaines de personnes impactées, d'avoir tout le monde dans le même projet de transition, c'était trop. Et on avait décidé de limiter ou de commencer avec la division IT, car on n'avait plus d'impact là et d'essayer de tirer des gens du business un peu plus vers la méthodologie Agile.

*Comment avez-vous fait pour que les personnes acceptent ce changement, est-ce que c'était un travail personne par personne ou cela se faisait en groupe de personnes ?*

F.H. : Par groupe de personnes, en fait on a commencé à former certaines équipes qui commençaient un nouveau projet. D'abord on avait le grand projet « Claims » où les gens

étaient obligés de travailler en Agile, car le bureau expert avait cette méthodologie, c'était décidé, pas de choix. Mais après chaque nouveau projet qui démarrait on formait les gens avant. Alors, on sélectionnait les gens un peu avec l'idée qu'ils allaient adopter cette méthodologie. On avait des gens qui avaient 50 ans et plus, on savait bien qu'il y avait des personnes pour qui c'était plus difficile de faire ce changement. On a donc décidé qu'on allait démarrer des nouveaux projets, avec des jeunes qui vont adopter assez vite la méthodologie et après les autres vont voir que ça a bien été, sont contents et c'est plus agréable. Mais même convaincre les gens d'avoir un laptop au lieu d'un PC fixe c'était déjà vraiment difficile. Ou le fait qu'ils ne pourraient plus rester à leur place. Car travailler ensemble veut dire se mettre ensemble. Par exemple, je travaillais sur ce projet avec quelqu'un qui était dans une autre salle, mais en Agile cela ne marche pas et convaincre les gens que c'était nécessaire, qu'ils ne pouvaient plus mettre leurs photos et autres, c'était vraiment difficile.

C'est pour cela qu'on avait pris des décisions, projet par projet. Un jour, on met en place une nouvelle organisation et à partir de ce moment-là tout le monde devait être formé et capable de travailler avec la nouvelle méthodologie.

*Et y a-t-il eu des problèmes lorsque vous avez commencé à mettre en place la méthode Agile ? Avez-vous dû faire des changements ou des modifications, par rapport à Scrum ou vous avez simplement suivi la méthode comme elle était et cela s'est bien passé ?*

F.H. : Je crois que personne ne suit la méthodologie complète à 100%. C'est plutôt apprendre, tomber et puis se relever. Un jour, on avait vu la nécessité d'engager des consultants externes « Agile coaches » pour aider à mettre en place la méthodologie, car d'une personne à l'autre c'était plus difficile ou plus facile, et ça dépendait aussi du projet.

*Au tout début, tout le monde recevait directement une formation ?*

F.H. : Oui, on avait organisé des formations. On avait les 7 sous-projets donc on avait organisé pour chaque sous-projet des formations, on avait des petits films, des présentations qu'on pourrait revoir sur l'intranet. On a bien donné les moyens de s'informer et d'organiser les formations strictes pour travailler avec la nouvelle méthodologie. On avait des check-lists.

Mais en même temps on avait les « brown bag session » le midi, ce n'était pas obligatoire, mais chaque consultant des 7 sous-projets donnaient une petite explication pendant une demi-heure pour donner des explications. Tout le monde avait l'occasion de sentir un peu, de goûter un peu c'est quoi l'« Agile ». Car on avait bien expliqué avant tout le projet pour avoir

tout le monde formé dans tous les sous-projets. C'était un très grand exercice et on avait donné plusieurs moyens. Au lieu de dire, « Maintenant on va former tout le monde sur ce sujet, et après sur ce sujet ». C'était trop de monde. Et en plus, une personne s'adapte plus vite qu'une autre. Et en fait, on avait des gens qui ne se sont jamais adaptés je crois.

*Et quel était le rôle des consultants ? C'était des coachs ?*

F.H. : Ils étaient des coachs, ils ont organisé les trainings sessions, mais ils étaient, en fait, activement dans les projets pour aider comme coachs. Ils n'avaient pas une fonction dans les projets, mais c'était vraiment pour coacher les gens dans un projet. Quand un coach n'était plus nécessaire, et qu'un autre projet avaient des problèmes, on donnait les services de ce coach.

*Avez-vous eu l'occasion de voir un peu les résultats de la mise en place de cette implémentation ? Vous avez eu des feed-back ?*

F.H. : Un peu, en fait j'ai suivi le projet sinistre où c'était bien fait. Car c'était un bureau extérieur, j'ai vu les résultats. J'ai vu que le business était plus concerné. Avant, on avait l'analyse et après on oubliait qu'ils étaient dans le projet. Il était en train de se développer et maintenant ils étaient plus investis et plus proches du projet, plus concerné. On pouvait adapter plus vite les demandes du business. Ça c'était vraiment clair, c'était visible.

*C'était un point positif pour le business alors ?*

F.H. : Pour le business oui. Pour les techniciens, ils n'avaient pas toujours le même enthousiasme. Mais des gens qui avaient 20-30 ans d'expérience pour une méthodologie, ils recevaient un petit papier avec les demandes et ils travaillaient pendant 2 mois et après ils disaient maintenant c'est prêt.

Dans la méthodologie Agile, dans le stand up meeting, chaque matin ils devaient dire ce qu'ils vont faire aujourd'hui et ça ce n'était pas toujours bien pour tout le monde.

*C'était un sentiment de contrôle ?*

F.H. : Oui, contrôle, ça c'était le mot que j'avais entendu le plus souvent, « on veut me contrôler ».

*Y avait-il encore d'autres feed-back sur l'implémentation ?*

F.H. : C'était un mix de positif et négatif.

*Et pour vous ?*

F.H. : Moi je suis pro, ce que je vois c'est que l'on a des résultats plus vite et même pour le business c'est toujours un trajet de changement quand on va développer quelque chose. Après 2 ans, on donne un nouvel outil, on a une certaine résistance. Or maintenant avec la méthodologie Agile, on a des petites pièces qui sont délivrées assez vite, on peut s'adapter assez vite, même le business a le nouvel outil ou le nouveau soft que l'on a. Même là la période de change n'existe plus, car ils ont déjà au début vu ce qu'ils vont recevoir et ils peuvent déjà travailler avec les nouvelles choses.

*Et pour les projets en eux-mêmes, y a-t-il eu des problèmes qu'il fallait changer un peu la méthode ou vous avez toujours suivi la méthode initiale ?*

F.H. : En fait, il y avait des problèmes de résistance. La personne qui ne voulait pas suivre strictement les règles ou qui ne voulait pas enregistrer dans l'outil qui supportait le projet. Avant, ils devaient enregistrer tout ce qu'il avait fait avant la fin du mois. Et c'est absurde parce que personne ne sait encore après 30 jours ce qu'il avait fait. Maintenant, c'était du quotidien qu'ils devaient faire et l'idée de contrôle de nouveau était là. Alors pour faire avancer le projet, on avait décidé pendant une certaine période on va limiter la méthodologie Agile jusqu'à un certain niveau pour essayer et après avec les coachs pour augmenter un peu le niveau pendant le projet.

Dans un projet spécifiquement, je me souviens qu'on a décidé, parce que le projet n'avancait plus. Ils pensaient qu'ils ne savaient pas comment ils devaient travailler. Et on avait décidé là OK, on va faire un pas en arrière pour garantir que le projet avance. On avait un peu les moyens contre les résultats, mais après le but était, même dans ce projet, avec ces gens spécifiques de continuer en Agile.

*Le business était-il content par rapport aux enquêtes que vous aviez faites ?*

F.H. : On avait réalisé plusieurs enquêtes pour voir si la satisfaction avait augmenté et c'était le cas. Je ne me souviens pas exactement des résultats et c'était chaque fois de meilleurs résultats.

*Est-il arrivé d'utiliser Waterfall en implémentant Agile ? Pour des raisons de résultats, de personnes ou de projets ?*

F.H. : Pas vraiment non. Mais je crois que c'était vraiment le down si on faisait cela. Car on prendrait le risque que certaines personnes disent ça y est on a gagné ». Donc non jamais.

*Pensez-vous que l'adoption des méthodes Agiles doivent poursuivre certains buts, selon vous ? Est-ce que vous pensez que pour implémenter Agile il vaut mieux avoir des objectifs, des attentes en vue ?*

F.H. : Oui, c'est plus pour défendre aux gens qui font de la résistance. On voit que ça va mieux. On a toujours besoin de KPI pour vérifier si le but est réussi. Mais en plus, comme change management oui pour défendre la décision de changer et de montrer que ça a amélioré les choses. C'est ce que j'ai entendu plus d'une fois sur le projet « cela ne sert à rien » et « c'est quelque chose de nouveau, quelqu'un a inventé cela, mais ... », « C'était mieux avant ».

*Quels objectifs donnez-vous aux personnes résistantes ?*

F.H. : D'abord on essaye de les convaincre que c'est mieux, et qu'il y a des avantages. Mais à certain moment on doit les obliger, c'est vraiment dire : « Pas question ! » et c'est tout. Souvent on a montré qu'il y a des avantages, il y a beaucoup de projets dans le monde qui ont utilisé Agile et cela a prouvé que cela marche, mais alors on doit accepter. Ce n'est pas la meilleure façon, mais parfois il faut le faire comme cela.

*Au moment où vous êtes parti, auriez-vous apporté des nouvelles améliorations à Agile et par rapport au système mis en place ?*

F.H. : Dans tous les cas, on doit adapter la méthodologie à la situation. Je ne suis pas un Agile coach. Je ne sais pas si ce qu'on a fait était tout à fait Agile, mais là je suis convaincu que l'on doit adapter la méthodologie au projet. On ne sait pas ou presque pas mettre en place une méthodologie théorique sans savoir si cela marche avec ces gens, ces projets, avec ce budget. Il y a certaines choses qui ont un impact sur ce qu'on fait et je crois pas que n'importe quelle méthodologie est vraiment 100% implémentable dans toutes les situations. Le fait que l'on avait choisi une méthodologie hybride c'était parce que on savait bien que suivre strictement la méthodologie dans une période limitée c'était impossible. J'espère qu'entre-temps on a un peu avancé, un peu plus loin dans l'implémentation de la méthodologie qu'à l'époque.

*Et quel est votre avis sur les méthodes Agile maintenant par rapport à votre expérience? Quelle serait votre conclusion sur la méthode Agile ?*

F.H. : J'en conclus que c'est mieux, que les résultats sont plus vite là, pour tout le monde qui est impacté par le projet avec le même but. Avant, on avait des conflits entre le business et les techniciens. Tout le monde travaille sur le même but. C'est difficile de changer les gens, donc

si on peut démarrer avec une équipe de personnes jeunes qui n'ont jamais connu quelque chose d'autre, cela va très bien marcher. Si on doit vraiment changer les gens qui ont un certain âge ou une certaine ancienneté dans leur méthodologie habituelle, cela prend du temps et il faut investir. Ce n'est pas impossible, mais pour certaines personnes, c'est impossible.

*Si vous deviez donner une définition de la méthode Agile ce serait quoi par rapport à votre expérience ?*

F.H. : En fait, il y a certains points, c'est difficile. C'est une méthodologie qui fait que le projet est plus gérable. On est obligé d'aller vraiment dans les détails, dès le début. Pour moi, c'est une caractéristique, ça fait partie de la définition, le fait d'aller dans le détail déjà au début du projet et on a besoin d'une idée du résultat final et on a une idée des sous-projets, de plusieurs parties du projet avant de développer déjà. Et parce que tout le monde est impacté en même temps, on fait partie d'une équipe. Pour moi, équipe c'est un mot-clé. On travaille au quotidien, tous ensemble avec le même but. Cela fait part de ma définition.

*Connaissez-vous l'Agile Manifesto ?*

F.H. : Cela donne une vue claire sur ce qui est Agile. J'avais acheté et lu le livre pour dummies au début de l'implémentation et le Manifesto explique clairement sur une feuille de papier l'idée de quoi il s'agit.

*Est-ce qu'il a un rôle sur l'entreprise ?*

F.H. : Moi je crois que non. Je n'avais jamais entendu avant. Mais je crois que cela pourrait aider. Si on avait eu cela au début du projet NEO c'est le nom du projet, je crois que peut-être cela pourrait aider les gens. On pourrait faire quelque chose, mettre un poster sur son bureau.

*Avez-vous quelque chose à rajouter ? Ou des questions ? Des remarques ?*

F.H. : Comme j'avais dit, maintenant je travaille seul dans un projet et j'applique Agile. L'idée d'Agile où je dois implémenter un outil et un service chez des clients. Et cela m'aide de convaincre les clients de travailler dans cet outil. Etape par étape. Je crois que la méthodologie ne doit pas être utilisée comme méthodologie, mais prendre des petites choses et pour améliorer les choses.

*Et cela permet au client d'avoir plus confiance en vous ?*

F.H. : Oui c'est certain.



## Appendix 9 : Interview Frederick Wellekens

F.W.: Release and project manager

F.W. : Tu parles de méthodes Agiles. Je ne suis pas quelqu'un de développement. Je suis occupé de méthodes dites agility plutôt opérationnel, release calendar. Si tu parles de scrum je connais, mais c'est déjà en place quand moi j'ai commencé à faire mes changements. Ça, c'est une petite nuance que je veux quand même donner et aussi le développement c'est une phase, mais seulement le scrum ce n'est pas suffisant. C'est pour cela qu'il y a des autres choses qu'on doit aussi faire attention et faire changer. C'est aussi pour toi utile comme cela tu ne te concentres pas seulement le développement, mais aussi sur le release et tout cela.

*Pouvez-vous vous présenter en quelques mots ?*

F.W. : Frederick Wellekens je suis free-lance IT proces owner, project manager depuis 2 ans. P&V c'est mon premier vrai client comme free-lance. Avant j'étais employé chez Bpost et encore avant j'étais IT coördinator in a school. J'ai finalisé mes cours en 2003, je suis ingénieur industriel d'informatique. C'était une des directions assez unique dans le temps c'était electro-mécanica, c'était pas vraiment fixé sur l'informatique. C'était à Gent et j'ai vu tout ce qui est Java, C#, cobol ça j'ai vu et j'ai aussi vu du côté opérationnel AD, tout ce qui est nécessaire pour les gens d'infrastructure. J'ai vu les 2 à l'époque dans l'école. Ça c'est l'avantage que j'ai eu et après je suis commencé comme ICT coordinator. Ça c'était mon premier job, car je n'ai pas trouvé de job en fait. C'était après 2000, on a eu le crash de 2000 et après tout le monde a eu besoin d'informaticiens après 2000 est passé et le marché était surchargé et j'ai pas trouvé facilement du travail et je me suis lancé comme IT coordinator. Ça j'ai fait un an c'était dans des écoles primaires, j'ai eu 6 écoles. Faire des cours avec des enfants, aussi donner des cours aux institutrices et supporter le secrétariat. Ça c'est en gros le job que j'ai eu. J'ai fait cela un an, un an et demie et après j'ai changé vers Bruxelles. J'ai fait le travail chez BPOST comme application engineer. Un application engineer, c'est quelqu'un qui est responsable d'une application spécifique dans l'entreprise. Ma responsabilité c'était les paquets, donc c'est taxi post. J'ai supporté cette application, c'était une plateforme assez grande. Ça c'était mon premier job, après j'ai évolué vers middleware engineer entretemps j'ai aussi fait des cours project management et j'ai fait des projets infrastructure et après je suis devenu release coordinator et après release manager chez BPOST . C'est comme cela que j'ai trouvé le chemin chez release management. Après release management chez BPOST je suis devenu team leader dans l'équipe opérationnelle à la fin de 20 personnes donc c'était

assez « challenging » et oui c'était un challenge et en 2015 ils ont tout reformé tout l'IT de BPOST et pour moi c'était le moment de faire le choix entre je reste et je continue le poste chez BPOST ou je quitte l'entreprise et je commence à faire autre chose comme free-lance. J'ai choisi de partir pour le free-lance et c'est comme cela que je suis venu ici comme release coordinator au début et proces owner IT Change&Release chez P&V. Ça c'est en gros le pad professionnel. J'ai fait des études post-graduat IT organisation & Management aussi c'était dans le temps de BPOST. J'ai fait pas mal de choses.

*Quelle était votre fonction chez BPOST et quelle est-elle chez P&V ?*

F.W. : Je suis proces owner IT Change&Release , cela veut dire qu'il y a des projets on a d'abord le stratégique, le board qui dit : ça c'est les projets que nous devons faire dans les années qui viennent. Donc ce sont des projets qui sont déjà sur le radar, mais pas encore actifs. Une fois que le projet commence à être actif, cela veut dire qu'il y a un project manager qui est assigné et il y a du budget et il y a des gens qui sont libérés. C'est à ce moment que le projet doit être connu chez nous aussi. Comme cela ça commence déjà dans notre radar pour faire le suivi. On ne va pas faire le travail du project manager, mais c'est pour garantir que la façon que le projet travaille c'est d'une certaine qualité et pour aider le projet et pour arriver à la fin sur la solution sur le produit qui était prévu dans le cadrage. Ça c'est en gros ce que l'on fait pour l'instant. Cela veut dire que l'on a les requiements que l'on est en train de définir et ça c'est quelque chose dans le projet et après quand les requiements sont définis, le projet vient chez nous, il y a certains « gate » que le projet doit passer chez nous, nous on va tchecker certains livrables par exemple si les gens sont planifiés au niveau testing, au niveau requiements, scope si cela est bien défini ou quoi, une fois que cela est assez bien pour nous, on donne notre « go » et cela veut dire que le développement peut commencer dans le projet. Ça c'est l'idée. Chez nous, on appelle cela le « ready for sprint » Cela veut dire qu'une fois le projet a passé ces « gates » cela veut dire que le projet peut commencer à développer. Le projet manager fait son travail et une fois que le projet a fini de développer il vient à nouveau chez nous et nous on commence à tchecker à nouveau, mais avec des autres livrables, si le testing est bien fait, si le scope est bien, si le timing est bien et si la qualité est bien et donc on fait aussi un certain tcheck et on commence déjà à regarder pour le mettre en production donc l'implémentation plan on doit avoir et donc c'est toute une liste de livrables que l'on tchecke à nouveau pour une deuxième fois et une fois que cela est bon, le projet peut se mettre en acceptance. Et que veut dire acceptance : cela veut dire que le business user peut commencer à valider le contenu du projet. Donc il y a un time line assez

connu et assez strict que le projet peut mettre en acceptance et peut valider le produit en fait. C'est à partir d'ici que cela peut « pondeler » il y a plusieurs projets qui travaillent en parallèle. A partir de UAT tous les projets sont mis ensemble sur le même environnement. A partir de built et test, ils font ce qu'ils veulent . C'est un poubelle, car il n'y a pas mal de projets qui sont en même temps, mais le timing est différent donc au niveau testing c'est pas facile. Mais une fois que tu es en acceptance, ce sont tous les projets qui sont mis en production pour le week-end, pour le même moment. L'environnement est assez clean, cela correspond le mieux avec l'environnement de production.

C'est ça que l'on fait, pendant la période d'UAT on fait le suivi à nouveau et à la fin quand la période d'UAT est finie je viens avant une avant dernière fois pour dire OK on est prêt, tout est bien fait et le projet demande le feu vert pour mettre en production. C'est l'avant dernier « gate » chez nous.

Ce qu'on fait aussi c'est un plan d'implémentation, on est sûr que tous les gens qui sont impliqués pendant le week-end sont bien au courant de ce qu'ils doivent faire pour éviter qu'on manque un truc pour mettre en production pendant le week-end. C'est ce qu'on fait aussi en parallèle.

Une fois le week-end même, on met en production, en fait ce qui se passe c'est à ce moment que le PM donne la responsabilité du projet à nous parce que c'est à nous de mettre le projet en production. C'est notre responsabilité de mettre le projet en total et bien en production pendant ce week-end. Ça c'est à nous et c'est une fois que le projet est mis en production, on regarde aussi l'incident après, le moins possible évidemment, on fait le suivi des incidents et on donne des rapports au management et je pense que c'est 5 ou 6 semaines après la mise en production il y a le dernier « gate » et cela c'est la clôture du projet. Cela veut dire que le project manager vient pour la dernière fois chez nous et il donne tous les lessons learn, les incidents qui sont fait et fait le handover vers le « day to day » c'est-à-dire le sema team. Cela veut dire que le projet est fini maintenant au niveau budget on va clôturer le tout, on va disperser les team lead, les team members. Donc le projet est fini et cela arrive dans le « day to day » où le produit qui est délivré par le projet vient dans l'équipe « day to day » et c'est à eux de gérer et c'est à eux d'avoir assez de connaissances pour quand il y a un incident, ou un changement qui arrive de traiter cela. Donc ça c'est la clôture du projet en fait. Ça c'est très vite sinon je peux te montrer un graphique c'est peut-être plus facile. Je ne sais pas si tu as compris ?

F.W. : Voilà une vue graphique. Oui ce sont tous les quality gate. Le start-up c'est au début on a un PM, mais on ne connaît pas encore les requirements. Le projet doit s'inscrire chez nous pour dire OK maintenant on est prêt pour faire l'inventaire des requirements. Le sprint, ça c'est quand on connaît tous les requirements, le développement sait ce qu'il doit faire en fait. Et c'est ici que tu vois par exemple ici c'est scrum, donc le développement méthodologie c'est ici que le développement commence à se faire dans une méthode assez Agile.

Ici c'est hybride Agile et donc le développement c'est fait sur une méthode Agile et on a des daily, des roaming sessions et tout ça, ça c'est entre le ready for sprint et le ready for UAT. C'est ici que tu vois vraiment la méthode Agile qui travaille.

*Est-ce pour chaque sprint ?*

F.W. : Par exemple, chaque projet il y a ici un sprint de 3 semaines, ça c'est le standard. Et le sprint, il y a des projets qui ont un sprint, donc c'est 3 semaines pendant la période UAT que ça doit être prêt. Mais il y a aussi des projets qui ont 5-6-7-8 sprints et c'est ici que tu vois vraiment tous les sprints, mais ici ce n'est pas nous qui font le suivi. Donc cela est quelque chose qu'on est en train de mettre en place, on doit aussi avoir un rapport ou une vue sur la fin de chaque sprint. On sait qu'à la fin de chaque sprint. Donc on a le ready for sprint, on dit OK on va le faire en 6 sprint. Donc, on connaît déjà le scope high level. Mais ce qu'on voit, c'est qu'après le sprint par exemple, tout ce qui n'est pas fini dans les premiers sprint, c'est rapporté vers les sprint après, mais la dead line à la fin des 6 sprint reste. Donc, à la fin, on voit qu'il y a des projets qui arrivent pour le « READY FOR UAT » oui on a fini, mais on a encore une backlog assez grand pour résoudre tout cela et ils demandent par exemple de prolonger la période de développement. Donc ça c'est quelque chose que l'on voit et qu'on essaie d'être plus vite au courant. Comme cela on sait déjà 3 semaines, un mois à l'avance, quels sont les projets qui sont, pas les mauvais élèves, mais qui sont en danger, parce que c'est ici que tous les projets arrivent ensemble en UAT. Une fois qu'ils sont en UAT il est très difficile de retirer un projet d'un release parce qu'ils sont assez imbriqués. Donc si un projet tombe dans l'eau ici, c'est le stress. Pour éviter tout cela, on essaie de tout faire à l'avance.

Le « ready for UAT » normalement le développement est fini donc le business peut commencer à valider. Ce que je te dis ici c'est tout de la théorie parce que dans la pratique c'est pas comme cela.

C'est pour cela que c'est très exigeant ce qu'on fait.

Ici tu as le « ready for prod » c'est-à-dire tout la période d'UAT est finie, le business a dit : on est suffisamment d'accord avec ce qu'on va mettre en production et on donne le feu vert.

C'est aussi un feu vert de nous pour dire OK tous les livrables c'est bon pour nous aussi donc on peut le faire.

Puis ici on a le release week-end même, et le closure c'est à la fin du projet, quand on dit OK on a tout livré, on n'a plus besoin de budgets, oui on clôture le projet.

Donc ça c'est en gros ce que nous faisons en release management.

*D'un point de vue Agile, vous suivez un modèle ou des méthodes ?*

F.W. : Au niveau Agile, ce qu'on arrive c'est qu'on a un release calendar que l'on doit suivre et le release calendar pour l'instant il n'est pas Agile. C'est ça que je voulais dire au début, on le développement qui est Agile, mais le mettre en production c'est pas du tout Agile. Donc c'est un hybride Agile comme on appelle cela ici, tu vas voir c'est aussi un peu waterfall, si tu vois, le ready for UAT doit être fini allé le développement doit être fini c'est pas Agile. Donc c'est un hybride Agile.

Donc le release calendar, pour l'instant, n'est pas du tout Agile. C'est quelque chose que l'on crée. Le release calendar de 2019-2020 existe déjà. Donc c'est quelque chose que l'on communique vers l'été. Donc maintenant on est prêt pour celui de l'année prochaine. Ici tu vois ça c'est le release week-end, les plages rouges et vert sont les périodes UAT. Ce qu'on a aussi ici c'est des regression testing et il y a un freeze aussi pour aligner. C'est plutôt pour les gens pour mettre leur vue sur le release week-end. Parce que si tu ne mets pas de freeze ici, les gens continuent à changer, continuent à fixer toutes les choses ici donc ils ne sont pas en train de regarder de mettre en production et c'est pour éviter que l'on oublie quelque chose qu'on a mis un freeze d'une semaine.

Cà c'est le release et l'UAT et tu vois qu'on rien qui est visible donc on a des projets avec un sprint, avec 5 sprint et tout cela ce n'est pas visible dans le release calendar. Mais le release calendar comme tu le vois ici c'est pas du tout Agile.

Avant, à l'époque il y a 3 ans, il n'y avait pas de release calendar ça c'était avant mon temps donc je ne peux pas juger, je ne peux pas te dire grand-chose de cela. Mais je sais que c'était très frustrant, ce qui se passe c'est qu'on travaille avec beaucoup d'équipes et chaque équipe travaille comme cela en fait. Donc on ne regarde pas ailleurs et on ne regarde pas avec son voisin en fait. Et quand j'ai dit, il y a une dizaine de projets qui arrivent en même temps, si ce

n'est pas coordonné, l'un projet va donner un coup de pied à l'autre projet qui cause des frustrations, qui cause du ralenti et tout cela c'est pour éviter tout cela qu'on a mis un release calender en place et pour être sûr que l'environnement acceptance c'est coordonné et bien. Donc c'est comme ça que l'on peut délivrer des projets encore, mais ça c'est un pas intermédiaire parce qu'on est en train de regarder pour être full Agile dans 2020. Et qu'est-ce que cela veut dire ? Et ça c'est quelque chose de tout nouveau tu as être la 2<sup>ème</sup> personne que je vais montrer cela. On a le release calender de 2018, on a le release calender de 2019 qui est une copie de celui-ci, mais il y a des initiatives qui sont en train d'être mis en route pour aller vers un calender full Agile. Et qu'est-ce que je veux dire avec full Agile? Et aussi ça c'est un pas en avant, ça c'est 2019 c'est une copie de 2018 sauf que en 2019. 2018 on a mis test automation parce que si tu veux être full Agile la partie devops pour nous c'est quelque chose qui est important, full automating testing , la façon comment on va déployer le code sur l'environnement c'est aussi quelque que chose que l'on doit automatiser donc le but de l'année prochaine c'est de mettre tout en place pour automatiser au maximum et pour créer une atmosphère assez devops. Je ne sais pas si tu connais devops ?

Le devops c'est plutôt une culture où on met tous les gens ensemble et tu donnes la responsabilité dans l'équipe. Donc c'est à l'équipe de développer, à l'équipe de créer, à l'équipe de résoudre tous les problèmes qui se passent dans leur scope dans leur produit.

Donc ça c'est quelque chose que l'on aimerait mettre en place, donc on est en train de regarder pour mettre cela en place à partir de l'année prochaine. Donc il on est en train de le préparer. Devops c'est un des choses, c'est le parapluie en fait, mais le code doit être assez Agile, doit être facile à déployer. Ça ne doit pas être manuel par exemple, aujourd'hui c'est encore quelque chose que l'on fait manuel et ça cause beaucoup de boulot et on peut faire des erreurs donc c'est pour éviter tout cela. Et la même chose que l'on fait avec le testing donc ça, ce sont les 3 piliers que l'on travaille l'année prochaine et tout cela c'est pour le shorter time to market et le quality. Donc ça c'est vraiment le mot clé pour les années qui viennent parce qu'on voit, et ça c'est pour moi et toi aussi on des clients dans le monde aussi, si on veut commander quelque chose on va sur bol.com ou coolblue et on veut que ce soit là. Tout le process qui est derrière on s'en fout, mais ce n'est quand même pas petit ce qui est derrière. C'est ça, c'est ce mind-set qu'on voit aussi dans toutes les entreprises, chez tous les gens en fait, sauf que par exemple comme BPOST, comme P&V on n'est pas prêt, Bol et Amazone et tout ça ils sont déjà beaucoup plus loin que les autres. Mais c'est ce mind-set qu'on est en train de voir chez chaque domaine. Chez Bpost, c'est le paquet c'est déjà comme cela, mais

chez l'assurance, chez tout en fait. Donc ça c'est le calendrier de 2020 donc on a un mélange des 2 donc c'est le même de 2018-2019, mais ici on a ajouté la partie Agile. Donc après chaque sprint on doit être capable de mettre en production. Donc cela veut dire qu'après chaque 3 semaines on doit être capable de mettre quelque chose en production. Et cela va être évidemment des petites choses et pas comme un corporate release parce qu'ici en corporate release on sort quelque chose que l'on a développé pendant des mois et des mois. Ça c'est des choses aussi que cela cause des risques. C'est un grand bloc et ce qu'on a développé en janvier par exemple pour le mettre en mars, les gens ne savent vraiment plus ce qu'on a développé si on a un problème avec cette partie de code, on doit encore relire la documentation et tout ça ça prend du temps. Pour éviter tout cela aussi c'est chaque 3 semaines. Donc des petits morceaux, l'avantage c'est que pour les gens qui développent on voit le résultat déjà après 3 semaines ou 6 semaines donc c'est beaucoup plus vite, c'est aussi motivant pour les gens qui développent. Et pour le business c'est le time to market, on voit au fur et à mesure des petits morceaux qui arrivent en production pour de nouvelles fonctionnalités donc ça c'est pour le business aussi un bon truc.

*Pour l'instant est ce que votre équipe est Agile?*

F.W. : Nous pour l'instant on n'est pas Agile. On est assez strict, tous les quality gate et tout cela parce que j'ai parlé des gate, ça ne marche plus comme ça. Si tu dois faire un gate chaque 3 semaines oublie ça on ne fait pas, on est déjà surchargé pour l'instant, mais dans les années qui viennent en 2020 on ne doit plus faire des gate et tout cela. Pour l'instant, nous sommes dans une situation où on n'a rien eu il y a 2 ans, on a mis tout le monde ensemble et on a aligné et dans le futur le but c'est que on est encore assez stricte, mais Agile c'est-à-dire que l'on est capable de traiter les demandes et les projets beaucoup plus vite qu'aujourd'hui. Donc aujourd'hui non on n'est pas Agile. On est seulement structuré et on est capable de donner des feed-back au management parce que c'est quelque chose qu'on ne pouvait pas donner dans le passé. Une des choses aussi que je crois vraiment c'est que devops et Agile pour moi ce n'est pas freedom c'est assez strict, c'est plus strict que ce qu'on a aujourd'hui. Aujourd'hui on n'est pas Agile, mais si tu veux être Agile tu dois être encore plus strict qu'aujourd'hui. Sauf que ce n'est pas très visible c'est dans le mind-set. Pour moi ce que j'ai vu c'est que pas mal d'entreprises disent qu'ils sont Agiles, mais si tu regardes vraiment derrière les rideaux ce qui se passe, c'est chaotique. Et ça pour moi ce n'est pas Agile. Agile est quelque chose de très structuré aussi sauf que le time line est beaucoup plus strict, mais si tu regardes derrière le rideau de l'entreprise qui disent qu'ils sont Agiles, c'est le chaotique. Tu n'arrives pas à

délivrer ce que le client veut et aussi ce que le management veut. C'est quelque chose qui va de main en main. Agile, le manifeste aussi, c'est très strict dans ce que tu fais, tu ne dis pas oui à tout ce qui se passe, tu sais ce que tu veux et tu arrives après 3 semaines.

Ce que je dis maintenant c'est de mon expérience. Si tous les autres disent autrement c'est à moi à regarder, mais c'est ce que je vois. Agile est un mot clé pour beaucoup de personnes cela veut dire qu'on peut faire ce qu'on veut. Ce n'est pas cela qu'Agile veut dire. C'est ça le message que je veux donner. Parce que à un moment on a eu des projets light, on a dit on va travailler un peu moins strict et on a vu que plein de PM plein de projets ont essayé d'utiliser cette façon de travailler et on a vu aussi que si tu regardes que c'est chaos. On a abandonné un petit peu la partie plus light et on a à nouveau réintroduit la structure. Agile c'est bon, mais ça doit être strict.

*Le projet d'introduire Agile dans le release c'est venu de qui ? C'est venu comment ?*

F.W. : On a 2 choses, j'ai fait la même chose chez Bpost et on a commencé à cause du business en fait qui a dit qu'on était trop lent et que ce qu'on a demandé on ne délivre pas à temps et la qualité n'est pas bon et tout cela. Et qu'est-ce qu'on a fait, on a fait un brown paper set. On a fait une réunion avec pas mal de gens avec tous les gens qui sont importants dans le sujet et on leur a donné des post-it. Sur le mur on met un brown paper et c'est aux gens de dire OK qu'est-ce quoi toi tu penses qu'est ce qui cause le problème qu'on a aujourd'hui. Qu'est-ce qu'on doit faire pour résoudre le problème du business. Donc on a mis tous des gens ensemble dans une salle de réunion, ils ont créé pas mal de post-it, on a collé cela sur le mur et après on a créé des clusters, on a créé des groupements et sur un des choses on a créé le calender et c'est sur base de cela qu'on a commencé à travailler. C'était moi le owner d'un des groupes et c'était le release calender qui n'était pas Agile, qui n'était pas assez bien. J'ai fait des interviews avec pas mal de gens aussi. C'était des one to one pour voir qu'est-ce qui était les possibilités, j'ai fait 4 ou 5 possibilités, j'ai fait un SWOT analyse et sur ces 4 solutions et à la fin j'ai donné un avis au management OK ça c'est pour résoudre le problème qu'on a au début, c'est ça la façon qu'on peut commencer pour Bpost. C'est vrai un des gens m'a dit si tu veux être vraiment bien et que tu veux répondre au problème tu dois travailler sur le full Agile. En fait c'est sur ça. À l'époque si tu fais des interviews et que tu regardes. Tu ne peux pas dire je vais créer un calender full Agile, mais si derrière tout n'est pas mis en place le test automatisé, le développement, le déploiement oublie ce n'est pas possible, mais chez Bpost j'ai dit c'est un très bon truc de le faire, mais ce n'est pas le

moment. Donc j'ai donné une autre proposition et sur base de cela management a créé un projet interne IT pour rouler le framework. Ça c'est le flow chez Bpost.

*Comment cela s'est passé ? c'est positif ?*

F.W. : Oui et non. Donc oui le projet est mis en production, on a travaillé 1 an, 2 ans sur cette façon de travailler et après c'est le Comité de Direction qui a décidé qu'on a commencé à regarder pour outsourcer le ITBpost. Et toutes les histoires que j'entends maintenant le release calender oublie. Maintenant, j'exagère ou les gens qui me disent cela exagèrent, on travaille pour faire un développement ils sont full Agile. Ils sont full Agile, mais ce n'est pas coordonné, il n'y a plus de gouvernance centrale qui monitore tout cela. Ça c'est un des trucs aussi est-ce que dans devops on a besoin de cela ? C'est vrai que peut-être que pas. Donc le release calender chez Bpost pour l'instant, d'après ce que j'ai entendu il existe encore, mais il n'y a pas mal d'exceptions et si tu arrives dans une situation où il y a une règle, mais il y a plus d'exceptions qui ne suivent pas les règles. Ça veut dire qu'il y a quelque chose qui ne marche plus.

*Savez-vous pourquoi ils ne le suivent plus très bien ?*

F.W. : Non, car j'ai quitté BPOST après, pour moi le seul truc que je vois c'est non je ne sais pas moi j'ai toujours pensé que c'était le grand out sourcing d'IT qui a foiré tout et les gens de gouvernance sont parti d'un jour à l'autre, mais ça c'est pour moi. Ça c'est pour moi, c'est très personnel.

*Et chez P&V ?*

F.W. : Oui ça c'est différent. Chez P&V qu'est ce qui se passe ici, c'est que à ma connaissance cela ne vient pas de business pour l'instant, mais c'est un bottom-up, c'est en interne que l'on voit que l'on peut travailler plus Agile. Donc ce sont les gens même qui vont poser des questions et qui commencent à commander pour être plus Agile pour travailler plus comme cela. Pour moi c'est une autre perspective, c'est une autre façon aussi parce que ça c'est le challenge que l'on a maintenant parce que j'ai parlé des initiatives qu'on est en train de faire, mais on doit persuader le management de le faire parce que comme toutes les entreprises il y a le budget et qu'on a un besoin de l'argent pour mettre cela en place et du temps au gens pour le faire donc c'est ça le challenge pour l'instant pour mettre cela en place. Ça c'est quelque chose qui bouge déjà depuis plus d'un an donc cela avance il y a 2 semaines j'ai entendu que le Comité de Direction veut aussi qu'on travaille comme Agile. C'est pour

moi une des choses qui peut avancer les initiatives par ce que cela veut dire que cela vient d'en haut aussi. Tu sais que dans une entreprise on travaille avec un organigramme à plusieurs niveaux et si le niveau plus bas dit, on veut travailler comme cela tu dois persuader tous les niveaux plus haut c'est plus lourd que si cela vient du haut vers le bas. Donc si quelqu'un comme Dirk dit on va travailler comme Agile, il a aussi le mandat de signer pour le budget aussi. C'est plus facile pour lui aussi. Mais maintenant je vois, et cela c'est quelque chose de nouveau depuis quelques semaines, il y a un pad top-down mais aussi un bottom-up qui vient donc il y a quelque chose sur ce sujet.

*Comment comptez-vous mettre ce changement en place ? est-ce que vous comptez mettre en place certaines choses particulières pour que tout le monde suive le mouvement ?*

F.W. : Pour le bottom-up on devait faire des tests estimation et la technique donc l'infrastructure et dans tous les 3 cas on a toujours fait un RFP pour regarder à l'extérieur pour avoir l'expertise des gens qui ont déjà fait. Donc il y a pas mal d'entreprises qui ont l'expérience, qui sont là pour donner leur expérience au client comme nous. Donc on demande toujours l'avis des autres entreprises pour donner leur expérience et pour avoir un way forward et après si on reçoit le go pour créer vraiment un projet on a besoin de ses gens aussi pour nous guider vers cette solution. Donc cela c'est différent de ce que j'ai eu chez Bpost. Là c'était quelque chose je ne vais pas dire de plus petit sans help de l'extérieur. Ici c'est quelque chose d'assez grand, car cela implique toute l'entreprise la façon de travail, la façon de process et tout cela donc c'est quelque chose d'assez grand et c'est quelque chose qui n'est pas nouveau. Il y a des autres entreprises qui ont déjà fait cela donc on ne doit pas inventer à nouveau la roue parce qu'il y a des gens pour t'aider. Ça va être plus vite si tu dois le faire toi-même, ça va prendre plus de temps.

*Et ses personnes-là ce sont plus des coachs ou plutôt des formateurs ?*

F.W. : Non coach, c'est plus que coach en fait parce que le coach pour moi c'est quelqu'un qui dit et qui te challenge. C'est toi qui reste dans le lead et c'est toi qui le fais. Ton coach il va te challenger et te poser plein de questions et pourquoi tu fais ça et ça et pourquoi ... il va te challenger, mais c'est plus que ça parce qu'il va donner de l'input, il va aussi dire, on a vu cela chez les autres clients et on propose que tu travailles aussi comme cela. Donc c'est plus que coach, formation non plus, car si tu fais comme cela les gens n'acceptent pas. Ce n'est pas comme un professeur qui vient, c'est quelqu'un d'entre les deux. C'est quelqu'un qui a la

connaissance, qui a l'expertise et qui donne cela vers nous pour nous aider pour arriver sur la bonne direction.

*Comptez-vous mettre certaines mesures pour voir si vos objectifs sont atteints. Ou cela n'est pas encore fixé ?*

F.W. : Je parle aussi à nouveau de l'expérience de moi. Comme par exemple, on a le closure du projet du business. L'expérience que je vois c'est dans IT en interne et c'est quelque chose pour nous on oublie ça. Donc on met quelque chose en place et oui on parle entre nous, c'est vrai c'est passé, ce n'est pas passé, mais à la fin si on fait des lessons learn, si tu arrives à la fin et si tu dis que si on a réussi ce qu'on veut, c'est bon et c'est passé. Si on arrive pas oui on commence à poser des questions et on commence à voir est-ce qu'on n'a pas à refaire quelque chose ou s'est moins strict ou c'est moins visible comme dans les projets business. Donc non. Non en fait si je dois être vraiment honnête, le paquet Bpost on a mis cela en place et c'est fait, cela a pris un an ou deux, mais on n'a jamais posé la question est-ce que c'est vraiment ça maintenant ce qu'on a demandé au début ? Donc c'est ça que tu veux demander ?

*Y a-t-il eu de l'amélioration ?*

F.W. : Oui il y a de l'amélioration ça il y a. C'est plus dans les feed-back pas vraiment un truc officiel où on dit par exemple dans un projet, ça c'est des close, on fait un close du projet, mais c'était assez informel et on a fait cela une semaine ou deux semaines après le livrable du projet. C'est trop tôt pour dire dans tout le bâtiment, à tous les gens si cela passe bien ou si cela ne se passe pas bien. On a fait une clôture du projet, mais on a fait un lessons learn, mais on doit regarder le travail du projet même, on a fait un process learn. Et c'est cela que je veux dire c'est : le problème que l'on a eu au début est-ce que c'est vraiment résolu, oui ou non ? Ça non.

*Quand vous avez mis ce calendrier en place, est-ce que vous l'avez modifié au fur et à mesure de ces deux années ? Par rapport aux attentes, d'un point de vue Agile ?*

F.W. : Ce n'est pas quelque chose de calendrier de cette année il est nouveau. C'est la première qu'on utilise ce calendrier. On introduit plusieurs slots pour mettre des choses en production. Au début on a travaillé avec un corporate release et un sema slot et un miner release. On a introduit plusieurs slots. La base de tout ça c'est pour ajouter plus de moments pour mettre quelque chose en production. Ça c'est quelque chose que l'on a avancé dans les années passées. Donc oui ça a avancé, c'est une virgule ou un point qu'on change dans le release

calender. Ce n'est pas quelque chose de vraiment structurel comme celui-ci de 2020. Celui-ci c'est vraiment autre chose. Ça c'est un grand changement. Celui de 2019 et 2018, on a fait un changement, mais ce n'est pas beaucoup. Ça avance, mais petit-à-petit.

*Comment jugez-vous qu'un projet soit un succès par rapport à votre équipe ?*

F.W. : À notre équipe, le plus beau paramètre que l'on a c'est que au moment que tu mets quelque chose en production, pendant le release week-end tout est bien passé et le jour après le release week-end, il n'y a pas beaucoup d'incidents. Ça c'est pour moi le KPI le plus important pour mesurer, mais ça c'est en global. Il n'y a pas mal de pas intermédiaires, mais en global c'est que si on a mis un corporate release en production et pendant le week-end ça s'est assez bien passé et on n'a pas eu beaucoup de stress et de problèmes et le jour suivant il n'y a pas beaucoup d'incidents, ça c'est pour moi un bon baromètre pour dire que c'était une bonne release. De l'autre côté, cela ne veut pas dire que c'était pour les projets bons. Pour le projet cela peut être oui on a eu un scope x et maintenant on était obligé de sortir avec un scope réduit assez fort. Pour le business cela veut dire autre chose encore.

*Pensez-vous qu'avec ce calendrier il y aurait encore davantage de succès dans vos projets?*

F.W. : Je vais le dire positif. Oui, mais parce que le calender as-us c'est pas ça qui augmente, il n'y a pas des choses derrière qu'on doit aussi faire et c'est pour cela que je donne aussi le message ici que on doit aussi faire le testing automation, des requiements c'est un des problèmes que l'on a aussi parce que je parlais des gate mais je peux donner un go sur la qualité inférieure aussi . Donc ça on sait parce que si par exemple tu dois passer par un gate, et tu as besoin des requiements tu peux voir cela de plusieurs manières. Tu peux dire OK les requiements il y a un document qui est là et qu'on appelle les requiements c'est suffisant donne le go ou on peut dire on a un document requiements, on voit qu'il y a pas mal de choses dedans et on voit qu'ils ont écrit assez bien et la qualité est beaucoup plus élevée que c'est un document requiements. Pour l'instant on donne des go assez faciles. Pour nous aussi c'est une chose que l'on doit travailler c'est les requiements. Les requiements qui arrivent ne sont pas toujours mature. Mais ça c'est quelque chose qu'on travaille aussi pour parler avec les gens de business et IT pour former les gens qui arrivent pour dire les requiements si tu veux commencer à développer tu dois savoir ce que tu dois développer. Et aujourd'hui on donne le go et si on voit un document requiements c'est suffisant, mais on donne le go assez facilement. Ça aussi c'est une chose sur laquelle on doit travailler. Chaque gate si on

augmente au fur et à mesure les livrables on peut arriver aussi vers le release calendar avec un succès livrable des projets parce que le release calendar pour moi ça c'est facile tout le monde peut le faire. Un release calendar comme cela tout le monde peut le faire, mais pour arriver sur un moment où on délivre quelque chose en production et les clients ou le business est content ça c'est autre chose. Oui et aujourd'hui on travaille sur la partie devops. On va suivre le chemin devops, test automation et aussi l'infrastructure la façon comment on va développer. On est en train de regarder si continuous integration, continuous deployment on est en train de regarder un cicd CI/CD (continuous integration/ continuous development) ça veut dire que tout est automatisé. Ça ne va pas être le cas en 2020. Si on a la base en 2020 je suis déjà content et après on peut encore augmenter la qualité, augmenter les choses.

*Est-ce tout cet ensemble qui fait que vous allez vers du plus Agile ?*

F.W. : C'est pour cela aussi que tout au début j'ai dit on travaille Agile au niveau développement, mais les clients à fin sont content parce qu'on travaille Agile. À la fin il vit le même résultat. Le produit arrive avec un corporate release et il doit attendre des mois et des mois. Mais IT dit quand même on est Agile. Je pense que maintenant on est assez conscient qu'on doit travailler sur plusieurs trucs pour augmenter la qualité et tu dois fine tuner à gauche, à droite, c'est donc comme cela que l'on va travailler les années qui viennent. C'est comme cela que je dis la structure est là parce que si tu n'as pas de structure tu ne sais pas sur quoi tu dois commencer à travailler pour améliorer. Donc c'est peut-être cela que je voulais dire tantôt pour moi Agile c'est plus strict, car si tu es strict tu sais sur quoi tu dois travailler pour améliorer le total.

*Selon votre avis personnel, pensez-vous que l'adoption des méthodes Agile est dans le but de poursuivre des objectifs stratégiques par exemple ?*

F.W. : Non pour le business si tu travailles Agile ou pas à la fin la stratégie doit être délivrée. La façon comment on le fait n'importe pas. Si tu veux travailler Agile ils disent OK ça va, débrouille-toi et amuses toi bien et travaille Agile. À la fin, on veut que la stratégie soit atteinte. Mais c'est vrai que moi je crois aussi que la façon de travailler more and more Agile cela augmente la chance de délivrer le projet stratégie, cela augmente la qualité. Ça je crois vraiment. Mais quand tu dis un CEO comme Hilde Vernaille chez nous, si on dit on va travailler Agile elle va dire bien pour toi, mais ma stratégie c'est ça et c'est à toi de te débrouiller comment tu vas y arriver pour délivrer cette stratégie.

*Faut-il changer certains objectifs pour les personnes qui les mettent en place ? Est-ce qu'elles doivent avoir un objectif pour les mettre en place ?*

F.W. : Par exemple au niveau d'un IT directeur il a un objectif pour par exemple dire le-time to-market doit s'améliorer. Donc les projets doivent être capables de délivrer plus vite quelque chose en production.

*Faut-il cet objectif-là pour pouvoir développer Agile ou qu'on peut tout simplement développer Agile sans des objectifs ?*

F.W. : il faut non, mais cela aide beaucoup. C'est comme aujourd'hui, mais ça aussi c'est mon avis personnel, on a le bottom-up qui dit oui on veut travailler Agile at on veut travailler comme cela et on a le management qui dit ce n'est pas dans mon objectif pour l'instant. Et je ne veux rien dire contre le management qui ont des autres objectifs alors donc ce n'est pas sur leur radar pour l'instant, mais c'est plus difficile d'attirer leur attention si ce n'est pas un des objectif de l'IT management. Donc est-ce que c'est nécessaire : non, mais cela aide.

*Que pensez-vous que des méthodes Agile en général avec votre expérience?*

F.W. : Je trouve cela une bonne évolution, mais si je parle avec des gens qui sont plus vieux que moi et qui sont déjà des années et des années dans le département IT eux commencent à dire que c'est un cercle, apparemment on a déjà travaillé Agile dans le passé sauf que c'était un autre nom. Donc pour moi c'est une bonne évolution sauf que on ne doit vraiment pas oublier que ce n'est vraiment pas chaotique. Travailler Agile c'est bon pour délivrer à la fin, mais la façon comment tu le fais ce n'est pas un chaos.

*Comment présenteriez-vous les méthodes Agile ? Si maintenant je vous demanderais une définition de qu'est-ce-que sont les méthodes Agiles. Pourriez-vous m'en donner une ?*

F.W. : Pour moi la base c'est toujours « customer oriented » pourquoi parce que c'est eux qui te paient donc c'est pour eux que tu travailles, tu dois toujours être « customer oriented » et avoir des short feedbacklooks et que tu dois être capable de montrer à ton client assez vite un progresse. Dans le passé on a dit OK on a un projet, le client donne cela à IT et IT dit OK on va entrer dans notre cave et on va travailler 6 mois sur le projet et on va revenir avec le produit. Et ça c'est quelque chose qu'on ne peut pas faire. Et c'est ça que toutes les méthodes Agiles ne font pas c'est qu'on travaille ensemble avec le client et c'est ensemble que l'on va arriver à un bon résultat. Pour moi agility c'est devops. C'est tous ensemble travailler pour le

même résultat et en bon esprit et aussi et vouloir vraiment que cela passe et que tu réussis. Etre motivé.

*Connaissez-vous l'Agile manifesto ? Est-ce que cela a un rôle dans l'entreprise au jour d'aujourd'hui ?*

F.W. : Cela veut dire que je dois connaître les stratégies d'entreprises aussi et ça je ne connais pas.

*Pensez-vous qu'aujourd'hui cela n'a pas un rôle spécifique ?*

F.W. : Non, mais je pense que cela va venir. C'est quelque chose je pense au niveau Agile, au niveau développement, au niveau project manager tout cela c'est déjà bien connu, bien intégré, mais au niveau higher management c'est quelque chose d'assez nouveau. C'est mon avis personnel. Je suis convaincu que cela va venir. Parce que cela vient des 2 sens time-to-market et le client qui est plus demandeur qui est aussi plus assertif parce que à l'époque ton assurance, tu parles avec quelqu'un, mais aujourd'hui tu mets sur Facebook ou sur Instagram et cela explose directement. Pour éviter tout cela aussi c'est quelque chose dont on doit tenir compte.

*Apporteriez-vous encore des améliorations au système mis en place et que vous allez mettre en place ? Est-ce qu'il y a encore d'autres choses que vous voyez que vous pourriez ajouter pour améliorer la manière de travailler ?*

F.W. : Pour l'instant, je pense que l'on a pas mal mis de track en route pour arriver vers cela. Peut-être oui quelqu'un de l'extérieur peut venir encore, mais on a déjà eu des RFP, on a déjà fait un assessment et je connais les résultats et il n'a pas de surprise dedans. Mais je pense avec les connaissances que l'on a aujourd'hui qu'on n'a pas oublié autre chose.

## **Appendix 10 : Interview Louis Destrée**

L.D. : Chef de projet à l'UCL

*Pouvez-vous vous présenter en quelques mots ?*

L.D. : Je m'appelle Louis Destree, je suis chef de projet à l'UCL. Je ne suis pas là depuis très longtemps, cela va faire un an. J'ai occupé maintes fonctions dans le monde privé pendant une bonne quinzaine d'années. Avant d'être chef projet, j'étais développeur aux différents échelons du métier d'informaticien. Ici maintenant je suis chef de projet IT où c'est une organisation de projets où il y a un chef de projet technique, un chef de projet métier et le tout

chapeauté par un directeur de projet. Le directeur de projet et le chef de métier, sont à mi-temps et donc je fais la fonction à temps plein.

Mon rôle c'est à la fois chef de projet et chef d'équipe, je gère le projet OSIS (Open Student Information System). On a 8 développeurs, 2 analystes, une personne pour le support DevOps ingénieur système pour ce qui est gestion des serveurs et ce genre de choses.

Le projet OSIS, c'est le projet qui consiste à réécrire toute la gestion de la partie administrative de l'UCL de l'application existante qui s'appelle EPC qu'on est en train de réécrire progressivement.

Je suis chef de projet, mais chef d'équipe aussi. C'est moi qui gère un peu la coordination entre les personnes dans l'équipe et les différentes équipes. Je suis aussi le chef d'équipe de l'équipe EPC. Les développeurs et analystes qui continuent à gérer leur application où là ils sont beaucoup plus restreints où il y a 3 développeurs, un analyste et une personne chargée du support.

*Cela fait combien de temps que vous utilisez les méthodes Agiles ?*

L.D. : Cela fait entre 7 et 10 ans, je dirais.

*Et avant cela, vous savez ce qu'ils utilisaient comme méthode ?*

L.D. : Moi, j'ai commencé à utiliser les méthodes Agiles quand j'étais encore à l'UCL ou dans des sociétés privées. Je pense que l'histoire est un peu similaire, mon expérience et ici à l'UCL. Avant, on était beaucoup plus dans des méthodologies Waterfall. Donc là, on part vraiment du principe qu'on va exprimer les besoins, analyser l'entièreté des besoins, établir un plan de développement, et puis passer à la phase de réalisation du plan. C'est le même constat qui est mené partout dans les agences de méthodologie Agile. Dans ces méthodes-là, on s'est quand même rapidement rendu compte qu'il y avait en général un délai énorme entre l'expression des besoins et les premiers retours vers les utilisateurs. Et justement, ce délai énorme faisait parfois que quand on montrait un premier quelque chose à l'utilisateur, on se rendait compte, beaucoup trop tard, que finalement on n'avait pas compris le besoin. Entre temps, le délai était tellement long que, finalement, le besoin de l'utilisateur avait changé. C'est des méthodes, qui par elles-mêmes sont excessivement contraignantes à mettre en place et à gérer, donc la gestion des projets beaucoup plus contraignants. Et pour le gestionnaire de projets et pour les équipes aussi. Dès le départ, penser à tout, c'est saucissonner le travail jusque dans les moindres détails. Le constat principal, c'est le temps nécessaire entre l'analyse

des besoins et la disponibilité d'imprimer quelque chose. Je pense que cela répond aussi à des impératifs par rapport au marché tout simplement. Dans tous les cas, dans le secteur privé, moins ici dans le public, le temps autorisé entre une idée et sa commercialisation pratique est de plus en plus court, et on a besoin de méthodes plus pragmatiques qui permettent d'aller beaucoup plus vite dans les développements. Dans tous les cas, arriver beaucoup plus vite à un premier résultat quitte à améliorer par la suite. Je crois que c'est aussi toute l'émergence des start-ups et ce genre de choses, sont liées aussi à l'urgence de ces technologies, pour rapidement être en mesure de remplir quelque chose pour après améliorer.

*Et dans le cadre de l'UCL, est-ce qu'il y a d'autres raisons qui poussent à utiliser des méthodes Agiles ?*

L.D. : Oui, mais ce sont les mêmes que l'on rencontre dans le privé. La principale raison c'est que la méthode de gestion de projet antérieure ne s'est pas révélée efficace. On met en œuvre une méthodologie Agile souvent sur des projets qui se sont plantés pour végéter dans une méthodologie plus traditionnelle. Ici, c'est clairement le cas. Ici, c'est encore assez particulier aussi. Historiquement, au niveau de l'UCL, on utilisait plus une méthode traditionnelle de Waterfall. Ils ont lancé le projet OSIS il y a 2 ans, en pensant faire de l'Agile. Donc, ils ont pris toute une série de concepts, d'idées, ils ont voulu adhérer à toute une série de principes Agiles, sans réellement mettre en œuvre une méthodologie. Pour moi, l'Agile c'est tout un ensemble de principes, et d'idées. Ce n'est pas une méthode en tant que tel, après il y a différentes implémentations de méthodes proprement dites, ça c'est du Scrum, ça c'est autre chose. Ici, à l'UCL pendant un an, ils étaient vraiment à mi-chemin donc ils adoptaient des principes Agiles, tout en prenant certains éléments de la méthodologie Scrum, mais pas tous. Ils travaillaient avec un Kanban, mais pas jusqu'au bout. Notamment, quand je suis arrivé, il n'y avait pas à proprement parlé de backlog. Le travail n'était pas coupé en user stories. Après il y avait une série de tickets qui existaient, c'est le principe de travailler sur des sprints de 2 semaines. Les tickets par exemple, n'étaient pas évalués. Aucune visibilité en termes de planning sur l'ensemble du projet.

Ça, c'est ce qu'on a fait depuis que je suis arrivé, c'est garder les principes d'Agile-même et d'appliquer à la méthodologie Scrum et toutes les différentes choses prescrites dans le Scrum.

*Pourquoi avoir choisi Scrum ?*

L.D. : Parce que cela correspondait le mieux à ce que j'avais moi-même pratiqué par le passé. Justement, par le passé quand je pratiquais cette méthodologie, dans les sociétés privées,

c'était dans un premier temps de manière très intuitive et sans suivre les règles jusqu'au bout. Quand je suis arrivé à l'UCL, j'ai vu qu'ils avaient aussi fait de la customisation de la méthodologie, mais parfois poussé à l'extrême ce qui rendait le projet inopérant. On aurait pu dire que ces principes-là étaient applicables, et qu'il fallait les pousser jusqu'au bout. Par exemple, ils travaillaient sur un Kanban, on peut se limiter à travailler simplement sur un Kanban, mais ça c'est plus pour des projets de maintenance. Je trouve pour un projet où on crée réellement quelque chose, on crée une nouvelle application, Scrum est a priori plus adapté, avec ses sprints, avec ses stories points. On peut vraiment faire un suivi d'un projet sur le long terme, Scrum est plus adapté qu'un simple Kanban. C'était vraiment la philosophie du projet, quand je suis arrivé. C'était de dire, on définit des tâches, on les met dans les to do au niveau du Kanban, et puis chaque développeur va naturellement sélectionner les tâches qui l'intéresse et les faire progresser. De mon point de vue, cela peut difficilement marcher à moins d'avoir une équipe excessivement mature. Cela peut difficilement marcher dans un contexte où on crée de A à Z une nouvelle application.

Cela marche très bien quand on fait la maintenance d'une application existante, où effectivement, sur le kanban, on a enregistré tous les incidents qui se produisent, et plus spontanément, les développeurs vont résoudre les incidents sur la gestion à la petite semaine. Cela peut fonctionner.

Dans une gestion de projet, réelle, le Scrum est plus adapté.

*Et pour implémenter Scrum, avez-vous mis en place certains postes, avez-vous donné des formations ?*

L.D. : Oui, ça a été une grande chance, qui est assez spécifique au contexte particulier de l'université. Et donc dans le contexte du projet en particulier, c'est un projet qui pour l'université est reconnu par tous comme étant un projet capital, donc important. Donc, il y a des ressources qui sont mis à disposition, dont des ressources financières non négligeables. J'ai vraiment eu de la chance, en arrivant. Quand je suis arrivé, j'ai dit, là ce que vous faites ce n'est pas tout à fait de l'Agile. Cela ressemble à de l'Agile, mais ce n'est pas de l'Agile. J'ai commencé à expliquer la manière dont moi je voyais l'Agile, mais surtout là où j'ai eu beaucoup d'aide c'est du fait d'avoir un budget de formation assez conséquent. On a fait appel à un coaching Agile. Donc il y avait, une personne qui venait un jour par semaine. Agile tout le monde connaît Agile, d'avoir entendu parler, ou lu, ou entendu de gauche à droite, après l'appliquer concrètement et aller jusqu'au bout, ça c'est autre chose. Cette personne

venait une fois par semaine, justement rappeler les grands principes, et l'accompagner dans la mise en œuvre des différentes réunions, des différents artefacts. On a fait ce coaching à raison d'une fois par semaine, pendant 3 ou 4 mois. On a aussi suivi toujours, par la même société, une formation de 3 jours sur ce que c'était l'agilité, tout ce qui était les Scrum Masters, les implémentations alternatives comme le Scrum, et on a terminé. Et ça c'était aussi très constructif. On a eu une formation de 3 jours et 17 personnes ont suivi cette formation de 3 jours. Et du coup c'était une formation certificative pour être Scrum Master de niveau 1. Donc tout le monde a été certifié Scrum Master.

*Et vous avez implémenté Scrum en une fois ou vous avez mis certaines pratiques en place petit à petit ?*

L.D. : Etant donné qu'il existait déjà toute une série d'éléments en place, le sprint était déjà en place. Avant même la formation, moi j'ai dit : il faut avoir une période. On avait fait l'effort de faire une analyse des user stories de l'application, car on parle d'un projet sur 4-5 ans, parce que on a un périmètre tout ce qui est formations, les cours et les programmes de formation. Dans cet objectif-là, on a redéfini toutes les user stories, on a recréé un backlog. Assez vite, on a réintégré l'ensemble de la pratique. On avait le backlog, on avait des estimations en stories points et on a commencé à mesurer, on a trouvé la vélocité de sprint en sprint. Cette vélocité, on a eu quelques ratés au début et puis aujourd'hui elle est stable. On a mis en place, les différents moments. Moi je suis arrivé, on avait une réunion d'équipe toutes les 2 semaines qui étaient un peu la clôture du sprint précédent et le début du sprint suivant. La réunion du sprint planning, existait, mais pas aussi précise que ce que l'on fait aujourd'hui. Ce qui n'existait pas et qui est vraiment très bien c'est le daily scrum, ça on a mis en place tout de suite. L'équipe était très demandeuse du daily scrum et puis on a mis en place la rétrospective. Donc toutes les 2 semaines, on fait une réunion d'une heure, d'une heure et demi, chacun s'exprime un peu plus sur son ressenti. Cette réunion aussi est très efficace et positive. Le sprint planning, il existait, mais on l'a formalisé. Vraiment un sprint planning en 2 temps. Un premier temps, participent les analystes, l'équipe de développeur va s'engager à faire une série d'user stories uniquement sur base des stories points correspondant aux différentes stories et à la vélocité moyenne. Cela dure en général, une heure, voire moins d'une heure. C'est aussi à ce moment-là que l'on fait la clôture du sprint précédent et l'ouverture du nouveau. Ce qu'on fait et qu'on ne faisait pas avant, c'est à la fois le plus douloureux, mais aussi très important, c'est le découpage, l'estimation en heures, des user stories. A priori, on va pouvoir faire tel et tel stories et puis dans un 2ème temps, ils prennent

chaque user story, la découpe en tâche. On estime chacune des tâches en heures, et on voit à ce moment-là, si cela matche avec le nombre d'heures qui est réellement disponible. 9 fois sur 10 cela convient. Il arrive exceptionnellement qu'on soit beaucoup trop bas, il reste encore plein d'heures, alors on en profite pour rajouter des user stories mais uniquement au niveau du scrum planning. Il arrive aussi que l'on se soit engagé d'en faire 5 et qu'en fait, ils ne puissent en faire que 4. Ça a pris sa forme beaucoup plus carrée. C'est au niveau des réunions, au niveau des artefacts, le product backlog est recréé au niveau des user stories. Il y a encore une réunion importante qui est le sprint review et ça c'est où on a encore un peu mal aujourd'hui. C'est la démonstration des nouvelles fonctionnalités toutes les 2 semaines. Au niveau du business, recueil du feed-back des utilisateurs. Et là, aujourd'hui, cela ne tourne pas encore comme il faut. C'est bien parce que quand je suis arrivé, assez rapidement, ils avaient perdu le contact avec les utilisateurs, ils avaient marre de venir aux réunions. Selon eux, cela ne servait à rien, pas assez de trucs à leur présenter. Ça on a quand même réussi à remettre en place. Aujourd'hui, ces réunions sont sous la responsabilité du chef de projet et pour moi ne sont pas toujours efficaces. On n'utilise parfois pas les bonnes réunions pour le recueil des besoins, de la validation d'analyses et de la validation du produit fini. C'est trop de trucs, normalement ça devrait juste être une réunion où on présente ce qui a été implémenté les 2 dernières semaines. Donc au niveau des artefacts, le produit backlog est mis en place et les user stories, c'est là où on travaille pour le moment. On a passé plusieurs jours avec des analyses pour définir les user stories après où on les a confrontés au business. Pour un certain nombre dans tous les cas, la réaction du business a été de dire : « Oui, mais, nous, on aurait pas du tout utilisé le même vocable ». Ça, je dirais c'est une leçon pour les prochains modules qu'on y attaquera. Pour les mois à venir c'est de formuler les user stories directement avec le business même. Si c'est des analystes, ce n'est pas les développeurs qui les formulent, mais même le langage des analystes n'est pas le même que celui du business. Voilà, ça c'est un point d'amélioration. Ou sinon, on a le rapport de sprint, c'est le diagramme qui montre l'évolution des story points sur la phase de 2 semaines. Donc il faut s'assurer que le sprint va bien pouvoir se terminer comme convenu. Et alors, on a la même chose, mais au niveau du projet, ça dans le cadre ici du projet c'est très positif, car cela nous permet vraiment d'avoir une communication très transparente, avec les autorités et les gens qui sponsorisent le projet, ce qui n'existait pas du tout avant. C'était un énorme problème parce que, avant, il y avait une tension entre l'équipe IT et les autorités. Parce que les autorités demandaient : « Vous en êtes où ? ». La réponse était du style, on avance, « Vous allez y arriver ? » On ne sait pas. Ici on a quelque chose quand même d'a priori plus précis.

*Quelles difficultés avez-vous rencontré pour mettre tout cela en place ?*

L.D. : Au niveau de l'équipe relativement peu. Quand j'ai débarqué, on m'a plus regardé comme de l'extérieur, je n'étais pas de l'université avant. Je débarquais en leur disant que ce qu'ils disaient n'était pas à 100% correct. Mais le fait d'avoir été tout de suite accompagné par un coaching, a vraiment appuyé la démarche. Après en général, il y a quand même un consensus assez large sur le fait qu'il faut faire de l'Agile. Dire que l'on va faire de l'Agile, cela effraye de moins en moins de gens et surtout les équipes jeunes. Or ici c'est une équipe relativement jeune. Je dois dire que j'ai eu relativement peu de résistance sur le principe qu'il faut faire de l'Agile, on en était conscient. Il fallait changer quelque chose, on m'a engagé parce qu'on reprend un projet qui végétait depuis un an, qui était clairement en danger. Or l'équipe le sentait aussi. Il n'y avait pas trop de résistance par rapport au changement parce qu'ils se rendaient bien compte que, oui, il y avait quand même quelque chose qui ne tournait pas tout à fait rond. Aujourd'hui, je ne dis pas que tout est parfait, mais factuellement, tout le monde se rend compte que cela tourne quand même mieux. Là où il y avait pas mal de distance quand même, c'est sur les aspects d'évaluation. Rien à faire, et ça j'avais le même problème dans le privé aussi, les gens quand on leur demande d'évaluer les tâches à faire, ils ont toujours l'impression que l'on fait cela pour les fliquer d'une manière ou d'une autre. Là, il y avait un peu de résistance. Moi, j'avais volontairement choisi la méthode d'estimation en story points plutôt qu'en heures parce que cela flique un peu moins. Parce que les estimations en heures, les développeurs vont toujours dire s'ils disent 10 heures et que j'en mets 15 on va me taper sur les doigts. Tandis que quand on les estime en stories point c'est très relatif comme info. Mais du coup, ils étaient rassurés je pense sur le côté on n'essaie pas de nous fliquer, en tout cas le chef de projet. Mais par contre, ils remettaient très fort en question la validité des estimations. Ils disaient : « Cela ne veut rien dire en story points, à quoi ça sert ? ». Alors là il a fallu un peu plus de temps. Je leur expliquais que c'est une valeur relative qui permet juste de voir en relatif où on en est au point de vue de l'avancement du projet. Du coup, c'était un peu mieux et cela est dû à la culture de l'université du coup. Mais alors les story points, la vélocité, tout ça c'est plus pour les autorités. On s'en fout, parce qu'ils étaient très fort dans une logique (et elle n'existe pas dans le privé) : « Mais bon le programme il sera prêt quand il sera prêt. On est à l'université, on ne risque pas notre job, on va faire le programme au rythme où on peut le faire, on le fera aussi bien qu'on peut le faire, il sera disponible ». Ça c'était une des réticences que j'avais, bien oui en faisant des story points et ce genre de choses, on essaye de s'inscrire dans un calendrier, on essaye de prendre

des engagements. Et ça c'est des choses que j'avais déjà eu par le passé dans les sociétés privées, ce sont des réticences qui tiennent un mois ou 2, et après, je pense que tout le monde rentre dans cette dynamique justement de voir la vélocité, voir que la vélocité augmente et justement de mesurer les tendances de la vélocité. Et finalement tout le monde s'approprie le bazar. Aujourd'hui, il n'y a personne qui remet en question le fait que l'on met des story points pour mesurer la vitesse à laquelle on va.

*Est-ce que vous avez déjà modifié ou adapté le modèle Scrum par rapport à votre organisation ?*

L.D. : Non, ça très peu. Régulièrement, tous les 2-3 mois, il y a des petits mouvements, des petites suggestions : « Est-ce qu'il ne faudrait pas scinder l'équipe en 2 ? est-ce que dans certains cas, il ne faudrait pas que les développeurs puissent quand même tester les tickets ? » Au tout début, la discussion sera : « On joue quand même un petit peu sur la durée des sprints, tiens là on sera quand même nombreux à être en congé, est-ce qu'il ne faudrait pas plutôt faire un sprint de 3 semaines au lieu d'un sprint de 2 semaines ? ». Grosso modo, j'essaye de dire non, on reste dans le truc de base. Le Scrum dit effectivement que la taille d'équipe idéale, c'est 5-6 développeurs pas plus. Ici, on est un peu trop nombreux par rapport à ça. Moi, au départ je ne voulais pas scinder l'équipe parce que le niveau de compétence de l'équipe n'est pas le même. Donc, j'avais peur qu'on se retrouve dans un truc, l'équipe des bons ou des moins bons donc il y a un phénomène de compétition entre les 2. Surtout je voulais d'abord qu'il y ait un élan positif pour se dire, voilà le projet avance, pour se rassurer sur les capacités, on le sait. Parce qu'évidemment le fait de scinder en 2 équipes permet de travailler sur 2 trucs parallèles d'accélérer au sein de l'équipe. Chaque partie va un peu moins vite, mais par contre est plus flexible sur le fait de travailler en parallèle. Au sein de l'équipe, on va y arriver progressivement.

*Aviez-vous des attentes particulières sur ce que ça allait donner après l'implémentation de Scrum ?*

L.D. : Oui, par rapport à ce contexte-ci, des attentes en termes de résultats tout simplement. On était dans un projet qui produisait des résultats, mais très très lentement et donc le premier objectif était de produire des résultats de manière plus fréquente. Donc, dans l'implémentation de Scrum c'était aussi de responsabiliser beaucoup plus les équipes. Ici, encore une fois, on est dans des contextes particuliers. À l'université, c'est une administration, les gens ne sont pas nommés, mais c'est tout comme. Et donc il n'y avait pas la pression comme dans le privé

et c'est bien. Un des buts recherchés était de restaurer une pression positive, pour les pousser vers un certain objectif, un certain résultat, et les responsabiliser. Et ce que je disais aussi c'est une équipe relativement jeune, et alors on est dans un contexte un peu protégé. Quand moi je suis arrivé, les développeurs ne se rendaient pas tous bien compte qu'ils travaillaient sur une application qui était utilisée par 60.000 étudiants. Les reconscientiser par rapport à tout cela, les inscrire dans la méthode, le fait d'être inscrit dans la méthode Scrum, les a intimement aidés. Pour se rassurer eux-mêmes qu'ils pouvaient produire un résultat. Car quand je suis arrivé, certains disaient : « Cela n'avancera jamais, parce que la gestion de projet est nulle, parce qu'on nous écoute pas ». Des tas de raisons. Le fait d'avoir mis une méthode très carrée je pense que cela a rassuré un peu le monde sur ça.

*Considérez-vous que ces attentes aient été atteintes ? Êtes-vous satisfait ?*

L.D. : En grosse partie, c'est ce que j'ai observé dans le privé aussi c'est que jusqu'à un certain point ça a marché. Ce qui est clair, qu'en temps de dynamique d'équipe, ça a créé une dynamique d'équipe très positive. Quand je suis arrivé, les gens, ça c'est aussi une particularité, contrairement au privé, ici on a encore, c'est un peu old school, c'est des analystes d'un côté, des développeurs de l'autre. On a très peu de gens qui sont des analystes programmeurs et donc quand je suis arrivé, l'ambiance au niveau de l'équipe n'était pas terrible. Il y avait un énorme fossé entre les développeurs et les analystes, les développeurs rejetant la faute sur les analystes, les analyses inexistantes ou mal faites et les analystes disant : « C'est quand même incroyable les développeurs ne sont pas foutus de comprendre ce qu'on leur demande ». Donc là, le fait de s'inscrire dans une méthodologie a vraiment bien aidé par rapport à ça. Ça a bien aidé, effectivement pour accélérer et avoir une production de résultat beaucoup plus régulière. Après, cela a énormément aidé pour donner plus de visibilité sur le projet, où en était le statut du projet. Je pense qu'ici, c'était aussi par rapport aux objectifs que l'on s'était fixé, l'objectif sur lequel on travaille était un objectif à 1 an et un peu plus. Je dirais qu'on va attaquer le 2ème tiers de l'objectif donc on rentre, de plus en plus dans la phase, c'est bien beau tout ça, on a déjà réalisé tant de choses. Si je prends ce qui est réalisé en story points, on a réalisé plus des 2/3 de ce qu'on avait prévu de réaliser. Donc on voit que cela avance, après c'est de vraiment aller jusqu'au bout du truc et ça je commence à le sentir ici et je l'ai déjà senti dans le privé, il y a parfois un gap entre : ok c'est bien beau toutes ces user stories, toutes ces story points, mais une fois que l'on a implémenté tout cela, est-ce que on arrive réellement à transformer ça en produit fini, réellement fini ? Qu'on a bien serré tous les petits boulons ? Donc je pense qu'on est dans cette phase-là qui n'est pas toujours facile.

Dans mes expériences précédentes, j'ai vécu plusieurs projets. C'est très carré sur la méthodologie, je prends un projet par exemple de 2 ans, pendant un an et demi on est très carré sur la méthodologie, et puis ici le dernier mois on se rend compte que l'on est quand même un peu short et alors là on rentre dans un mode de fonctionnement parfois moins cadré mais j'y vais jusqu'au bout. À mon avis, je voudrais éviter ce phénomène. Là, je voudrais qu'on arrive à aller le plus loin possible avec la méthode. C'est une des difficultés qui reste encore à franchir.

*Évaluez-vous votre implémentation de Scrum ?*

L.D. : Non, on pourrait, on s'était dit que l'on le ferait. Dans un premier temps, on a vu ce coach et on avait convenu qu'il reviendrait tous les 3 mois. Il est revenu 1 ou 2 fois c'était très sympa d'ailleurs. Je trouvais que ce coach était très bien dans l'animation du groupe. Après il nous avait fait un petit rapport sur la situation. Ça, par contre, je ne trouvais pas assez intéressant. Ce serait intéressant en effet, d'évaluer notre pratique, mais probablement par quelqu'un d'autre. Ce qu'on continue à faire, ça c'est chouette aussi dans le sens que l'équipe s'est vraiment approprié la méthode. L'équipe, même plus que moi en fait, va très régulièrement à des conférences de méthodes Agiles, organisées à gauche, à droite, ou des petites formations ici et là. Il y a un truc qui a été organisé à Bruxelles il y a 2-3 jours, de leur propre chef, ils ont été à cette formation. Donc ils continuent à avoir cette réflexion, sur la méthode, à venir avec des suggestions. On fait aussi une fois tous les 15 jours, un petit workshop en interne, l'un ou l'autre vient présenter un sujet à l'équipe et il n'y a pas longtemps c'était sur Spotify et leur implémentation Agile. Et voilà cela fait une émulsion d'idées par rapport à la méthode et donc effectivement l'avancement de cette réflexion sur le fait de quand on pourrait passer sur plusieurs équipes. Mais c'est vrai que plus formellement, on ne l'a pas fait, on devrait le faire.

*Avez-vous remarqué des améliorations par rapport à vos projets depuis la méthode Agile ?*

L.D. : Oui, l'amélioration c'est qu'il y a plus de résultats que l'on avait avant. Tout cela dans une bien meilleure ambiance, avec beaucoup plus de transparence. Et aussi, effectivement sur l'autonomie de la responsabilisation des développeurs. Avant, il y avait des développeurs qui se mettaient fort en retrait, ça je sais faire, ça je ne sais pas faire, il y a beaucoup moins ce genre de chose. Pour moi, ce sont des résultats concrets.

Il fallait un chef de projet et donc, notre chef de projet métier ayant des disponibilités limitées, ce n'était pas elle qui pouvait jouer pleinement le rôle de Product Owner. Donc moi j'ai pris ce rôle de Product Owner, mais ce n'était pas le rôle que j'ai préféré en fait. Parce que j'ai vraiment un background beaucoup plus technique. Mais il fallait quelqu'un pour le faire, du coup, j'ai pris le rôle de Product Owner. On a décidé de faire une tournante, pour ce qui était du rôle de Scrum Master, et ça je trouve cela assez chouette et vraiment cela contribue. Parce que ça dans mes autres expériences, j'étais systématiquement le Scrum Master tout en étant aussi le Product Owner. Et très régulièrement on se disait : mais c'est ce que la littérature défend : le Scrum Master doit être indépendant, autonome. Après ce n'est pas encore le dernier carat, parce que le dernier carat c'est vraiment d'avoir un Scrum Master en dehors de l'équipe. Et un Scrum Master qui est Scrum Master sur plusieurs projets, ça c'est vraiment intéressant, mais là on en est loin. On est une des rares équipes à l'UCL à fonctionner sur une dynamique Agile. C'est une des difficultés d'ailleurs, c'est que c'est très bien l'équipe de développement, qui fonctionne de manière Agile. Ils sont jeunes et se prêtent bien à ce genre d'idée. Mais par contre, les gens du métier, les utilisateurs ne le sont absolument pas. Idéalement pour les faire entrer dans un processus Agile, il faut qu'ils le soient un minimum et cela reste très compliqué.

*Est-ce que l'UCL voudrait que toutes ses équipes IT deviennent Agiles ?*

L.D. : Non, il n'y a pas de volonté particulière. Il y a des équipes qui ne pourraient tout simplement pas. Ici nous on a une grosse équipe, dans l'informatique à mon avis on doit être une des plus grosses équipes à former un bloc. Après, c'est énormément de petits blocs, 3-4 personnes je dirais que ce n'est pas efficace. 3-4 personnes cela peut être très bien pour fonctionner en Agile, parfois c'est souvent 2 personnes qui sont là depuis 20-30 ans il ne faut pas venir leur parler d'Agile. Même le Waterfall, ils ne connaissent pas. On leur demande, ils font. En même temps, cela fonctionne, dans une certaine mesure.

*Apporteriez-vous, personnellement, des améliorations par rapport à ce qui est mis en place ?*

L.D. : Oui je pense qu'il y a encore de la place pour des améliorations, globalement sur la manière de fonctionner, je pense. À terme, on serait plus efficace, dans 2 plus petites équipes, pour cela il faudrait encore se tasser un peu. 2 équipes de 4 c'est encore un peu léger, on risque des équipes pas tout à fait équilibrées. Avec 5 ce serait un minimum, d'avoir vraiment des tracks de 2 lignes de développement relativement en parallèle bien isolé. Au niveau des

analystes, là il y a un énorme boulot qui reste à faire. La qualité des analyses n'est vraiment pas au top. Parce qu'on a des profils très variés. On a une analyse très carrée, et qui va vraiment chercher dans le moindre détail. Par exemple, il y en a un qui ne va pas nécessairement être les plus doués pour rédiger un document bien clair. Par contre, il y en a d'autres qui vont rédiger un roman, mais qui sera incompréhensible. Donc la qualité de l'analyse n'est pas toujours top. Clairement, on essaye de mettre en place depuis 9 mois, une définition d'un vrai deal où on a une sélection de critères pour dire à partir de quand l'analyse est considérée comme complète et prête à mettre en développement. Et on n'y est toujours pas. Et ça c'est principalement un des gros freins que l'on a avec 2 analystes alors que le sujet est très vaste. En fait, 1 et demie, car ils sont tout le temps appelés sur différents fronts. Clairement, améliorer la qualité des analyses, cela passe par le fait de renforcer des développeurs. Si on peut renforcer encore les développements pourquoi pas, mais le plus urgent, ce serait de renforcer au niveau analyste. Ou de tout doucement attirer l'idée sur le fait d'avoir des gens qui puissent occuper les 2 rôles ce qui de mon point de vue pourrait être intéressant et gratifiant et en pratique ce serait utile pendant 2-3 itérations. On pourrait dire au sein de l'équipe de développement, systématiquement, il y a un développeur qui fait l'analyse, on fait une tournante, une fois par mois, une fois tous les mois, on change de manière à ce que chacun puisse monter en compétence. Parce qu'à mon avis, sur l'équipe de développeurs, au moins la moitié sont formés quand même comme analystes programmeurs. Après, c'est une question d'envie, s'ils veulent le faire ou pas. Voilà cela est sûrement un point à améliorer. Je l'ai dit tantôt un gros point à améliorer, c'est le sprint review et les interactions avec le business. Là cela coince à plusieurs niveaux. Ça coince d'un manque de disponibilité de la chef de projet métier, cela coince d'un point de vue disponibilité et de l'implication des utilisateurs. Ça, c'est par exemple un truc qui n'est pas confidentiel, on peut le dire, mais qui est très particulier au projet ici, c'est que l'on n'a pas d'utilisateur dédié au projet. Donc quand on doit poser des questions aux utilisateurs, quand vous voulez leur feed-back, on doit les solliciter, et donc on les interrompt dans leur travail. Oui ils nous aident, mais s'ils nous aident, ils ne font plus leur travail. Normalement, dans un projet de cette envergure, il devrait y avoir des ressources du business qui sont complètement libérés pour accompagner le projet. C'est ce qui se fait dans ce genre de situation normalement. On avait vu le cas de l'université d'Ottawa, qui avait remplacé complètement son staff informatique dans une autre formule puisqu'eux ont poussés vers des logiciels de gestion SAP et ce genre de choses. Mais c'est un projet où des personnes ont travaillé dessus, dont plus d'une vingtaine de métier, qui ont arrêté pendant 2 ans de faire leur métier juste pour faire la définition des besoins du projet. Au

niveau de l'équipe on peut faire tout notre possible pour fonctionner en Agile, si le business n'est pas Agile, cela ne simplifie pas les choses. Cela ne rend pas pour autant impossible, mais c'est vraiment un des gros freins.

*Quel est votre avis sur les méthodes Agiles en général ?*

L.D. : Je fais du développement depuis plus de 20 ans. Au début, avec le Waterfall et ce genre de chose, d'abord en tant que développeur et puis en tant que chef de projet ou gestionnaire de projet. Ces méthodes-là étaient excessivement lourdes à mettre en œuvre, procédurales, spépieuses, et très faillibles. Je me suis effectivement trouvé très souvent dans des projets, où effectivement après avoir passé beaucoup de temps de développement, on s'est rendu compte qu'on était à côté de la plaque. On avait fait quelque chose qui était très séduisant, mais qui ne collait plus aux besoins des gens. Donc, après les méthodes Agiles, si je prends mon expérience à moi, c'est l'expérience de beaucoup de chefs de projets. On a fonctionné un peu comme on le sentait, c'était le début d'Agile, en mettant des post-it sur les murs. Le fait de formaliser tout cela dans des concepts bien clairs, bien définis, on a recentré l'objectif des priorités ça c'est vraiment le manifeste Agile et la mise en œuvre dans du Scrum et autres : c'est se rassurer et cadrer une pratique commencée à mettre en œuvre. Dans mon expérience, cela a été très bénéfique et très opérationnel jusqu'à un stade. Le stade ultime dans tous mes projets précédents cela a été toujours un peu à l'arrache quand même sur la fin. Etre Agile et être vraiment Scrum jusqu'au bout du bout, ça, c'est un truc qui m'intéresse. Et après dire, est-ce que c'est la méthodologie ultime, ça je ne sais pas. Le Scrum certainement pas, après ce qui est intéressant ce n'est jamais qu'un cadre que l'on met sur les principes Agiles. Et cela, les principes Agiles sont relativement sensés et relativement immuables, il n'y a pas vraiment lieu de remettre en question les fondamentaux du manifeste Agile. Ce sera peut-être remis progressivement en question, ajouter des nuances, mais sur ces principes-là oui mettre l'accent sur tous les grands principes qui se trouve dans Agile.

*En ce qui concerne l'Agile Manifesto justement, quel rôle joue-t-il dans votre équipe ?*

L.D. : Je crois que c'est le socle commun sur lequel on est tous assez d'accord. Cela sert de base, de dire on adhère à cette façon de voir et c'est évidemment cette priorité-là qu'on essaye de mettre en avant.

*Si je vous demandais une définition de l'Agile, des méthodes Agiles, quelle serait-elle ?*

L.D. : Pour moi c'est une bonne approche, une série de pratiques à mettre en œuvre dans le but de mener un projet informatique à son terme. C'est vraiment cela l'idée. L'idée : qu'est ce qui fait qu'autant de projets informatiques partent en sucette ? Et quelles sont les grandes lignes directrices qu'on pourrait mettre en place et essayer de suivre pour que ce soit moins le cas à venir ? Pour moi, c'est cela.

Après c'est tout ce qui est l'ouverture au changement, l'activité et ce genre de chose, mais qui sont là aussi parce qu'ils répondent à une réalité et à un besoin du marché quand même.

*Quelle conclusion pourriez-vous tirer sur votre expérience Agile ?*

L.D. : Encore une fois très positive. Le point commun, c'est aussi l'aspect humain, cela redonne au boulot d'informaticien vraiment un côté très humain et l'interaction entre les gens. Cela ne tient pas nécessairement dans une approche plus traditionnelle où tout est très bien carré du départ, que chacun peut avancer dans son sillon sans trop faire attention aux autres, ça oui. C'est les interactions entre les gens. Le fait d'être à l'écoute aussi. Ici, quand je suis arrivé, c'était souvent : « Oui, mais comment voulez-vous qu'on y arrive ? ». Le métier vient toujours avec des nouvelles demandes, ça ne va pas, on ne peut pas accepter cela. Donc on était dans une situation de dualité entre le métier et l'IT. Le fait d'être dans cette méthodologie, dans cette approche de dire : on peut s'adapter, on peut les écouter et adapter le plan en conséquence. On pourrait être beaucoup moins dans un truc tendu et effectivement beaucoup plus ouvert. Dans un espace plus de négociation.

*Voulez-vous rajouter quelque chose ?*

L.D. : En déterminant le Scrum Master en roulement, et moi le Product Owner, c'est l'équilibre qu'on a trouvé pour appliquer Scrum, de manière non-traditionnelle. Ce qui était bizarre quand je suis arrivé c'est qu'il y avait deux chefs de projets, un directeur, et en Agile il n'y a pas cela. L'équilibre qu'on a trouvé c'est que le chef de métier, on le voit plus comme un Feature Owner. Et ça nous a pas mal aidé. Moi, avant j'avais travaillé uniquement en user stories, et quand je suis arrivé ici, on a travaillé en user stories. Et on a trouvé l'équilibre justement entre les deux chefs de projets : le chef de projet métier est devenu Feature Owner ou Epic Owner et on a donc commencé à travailler avec des epics. Et donc, elle travaille avec le business, plus au niveau de grosses fonctionnalités. Et nous, l'équipe technique, on travaille plus sur les user stories. Voilà, ça c'est un niveau supplémentaire intéressant. Et au-dessus du Feature Owner, on a le portfolio. Justement, l'université vient d'engager un portfolio manager, il y a 6 mois. Alors qu'avant, ce rôle n'existait pas. Pour répondre à la question si

l'université voudrait rendre toutes les équipes Agiles, non on en n'est pas là. Mais en tout cas, les équipes se rendent bien compte que c'est efficace, et que ça peut aider.

### **Appendix 11 : Interview Mathieu Gérard**

M.G. : Developer and partially Scrum Master).

*Pouvez-vous vous présenter en quelques mots ?*

M.G. : Je m'appelle Mathieu Gérard. J'ai étudié à l'EPHEC, donc j'ai un bac en informatique. J'ai terminé en 2007. Je travaille à l'UCL depuis 2007. J'ai travaillé pendant une série d'années avant d'arriver ici, pendant presque 10 ans en support informatique local - c'est les équipes en faculté qui dépannent les machines, ordinateurs, etc. Des professeurs pour la plupart et des administratifs évidemment. J'ai aussi fait un master en science du travail en cours du soir en cours de route. Voilà, maintenant je suis dans l'équipe de développement ici depuis mars 2017.

*Pouvez-vous présenter votre organisation ?*

M.G. : On est ici à l'UCL dans le service SIPS qui rassemble les informaticiens qui s'occupent du développement des applications à l'UCL. C'est donc très large, ça va des applications pour les cartes à puces aux SAP qui est tout le système de paie de l'UCL et on s'occupe principalement de OSIS qui est anciennement EPC que vous connaissez peut-être de nom. C'est le logiciel qui s'occupe de tout le corps business de l'UCL c'est-à-dire les cours, les inscriptions des étudiants, les inscriptions des étudiants aux examens, les encodages des notes, les délibérations, les productions des notes pour les étudiants, les productions des diplômes, toutes ces choses-là. L'équipe ici développe ça. C'est dans cette équipe là que je travaille depuis pour ma part un peu plus d'un an, donc depuis mars 2017.

On travaille en méthodologie Agile depuis septembre 2017 donc je suis passé de l'ancien modèle au nouveau. Je peux peut-être faire un petit topo là-dessus après. Voilà on est ici une équipe d'une dizaine de développeurs qui travaille dans ce domaine. Et une équipe d'analystes aussi qui travaille là-dessus. Et on a une partie de l'équipe qui travaille encore sur l'ancien système EPC et nous on est en train de développer vraiment on a commencé un autre système, système qui s'appelle OSIS de A à Z parce qu'on se rendait compte que le système EPC qu'on avait commençait à montrer ses faiblesses et donc on a commencé un nouveau logiciel.

*Quelle est votre fonction dans cette équipe ?*

M.G. : Je suis développeur. Et depuis la mise en place de la méthodologie Agile, on a décidé de mettre en place un système de Scrum Master, mais par roulement si vous voulez, donc on fonctionne par sprint de deux semaines. Je suppose que tous ces mots-là vous connaissez pour la méthodologie Agile. Par sprint de deux semaines et on est 3 Scrum Master qui prenons le rôle à tour de rôle par sprint en fait. Cela permet de garder un pied dans la technique quand on est pas Scrum Master on est « développeur » classique et donc la volonté à partir de ça, c'était de partir sur l'idée qu'on avait le rôle de Scrum Master sans s'enfermer là-dedans et sans perdre le contact avec la technique. Ce n'est pas ce qu'Agile préconise, Agile préconise un Scrum Master qui fait ça à temps plein quasiment si c'est possible, mais nous on a décidé de partir sur autre chose parce que cela nous convenait.

*Avant de développer cette méthode Agile, quelle méthode était mise en place ?*

M.G. : On fonctionnait par un principe de ticketing donc on avait une espèce du Kanban plus simplifié. On avait un board kanban avec les to-do, les in-progress et les done. On avait une série de tickets, de choses à faire définie en fonction de ce qu'on voyait, des contacts avec le business, des demandes qu'on recevait. Chaque développeur prenait un peu les tickets qu'il pensait pouvoir faire, il avançait comme ça, il faisait son ticket, il passait à la tâche suivante sans réel framework d'agilité là derrière. C'est un peu ça, à l'arrachée, je veux dire.

Ça ne fonctionnait pas mal, mais on s'est rendu compte à un moment donné qu'on manquait de structure si vous voulez, on avançait sur certains pans sans avancer sur des choses essentielles. On faisait plein de petites améliorations à gauche et à droite, sans réellement faire évoluer le cœur de l'application et ce dont nos utilisateurs avaient besoin et ce dont le business avait besoin pour pouvoir expérimenter.

*Est-ce que vous utilisez encore quelques pratiques du Kanban aujourd'hui ?*

M.G. : On a encore un board. C'est un board, je ne sais pas comment on peut le définir, mais c'est ça cale avec la méthodologie Agile. On implémente ça de manière Scrum quelque part donc on a un board Scrum. C'est-à-dire, qu'on a nos différentes colonnes : le backlog, les choses qui sont prêtes à être développées, les choses qui sont à analyser par les analystes, les choses qui sont en cours de développement, les choses qui sont testées par le développeur, testées par l'analyste, mises en production, ce genre de choses là. On a toujours ce principe-là

et à côté de ça on a greffé tout ce qu'on a implémenté côté Agile, c'est-à-dire les rétrospectives, les stand-ups meeting, toutes ces choses-là, qu'on fait d'une manière régulière.

*Et comment avez-vous mis en place ces changements ?*

M.G. : Par une formation en fait. Donc on a fait appel à la consultance pour la formation, en consultant Agile quelque part. Donc on en a eu deux. Un premier qui est venu petit à petit nous amener quelques idées de manières incrémentales, qui a amené quelques petites choses, quelques inputs dans le projet puis un moment donné on s'est dit qu'il faut une bonne base pour partir tous ensemble au même « niveau » par rapport au cérémoniel Agile, etc. ; notamment le b.a.-ba de ce qui est proposé par cette méthode pour après pouvoir éventuellement pouvoir l'assaisonner à notre sauce. On a suivi une formation de 3-4 jours, je pense, avec un formateur qui est venu ici en interne, nous lancer là-dessus. C'est un formateur de chez Wemanity qui est une boîte qui fait la consultance en Agile, en France aussi d'ailleurs.

*Aviez-vous engagé également des personnes particulières pour mettre ça en place ?*

M.G. : On n'a pas engagé de personnes particulières pour ça, si ce n'est Louis Destrée qui est arrivé plus ou moins au début de la mise en place de la méthode Agile dans l'équipe, mais on ne l'a pas engagé pour ça à la base, on a surtout composé avec les membres de l'équipe. Il est arrivé à ce moment-là à la base avec un rôle, mais avec un rôle de coordinateur d'équipes techniques, donc quelque part coordinateur de l'équipe de développement et des analystes. Maintenant, du coup de par la formation Agile, on s'est rendu compte que son rôle qui n'allait pas mal avec le rôle de Product Owner, de la méthodologie en fait. C'est un rôle qu'il assume tout en étant quand même assez large dans ce qu'il fait donc. On est à l'UCL dans une méthodologie, enfin, dans notre équipe en tout cas, dans une méthodologie Agile qui démarre.

On garde encore des anciennes habitudes de management assez stratifiée parce que l'UCL est comme ça. Nous, dans notre petite équipe on essaye de mettre en place les méthodes Agile maintenant. Il y a encore une série de choses qui nous sont demandées au niveau du management qui reste plus classique. Voilà, donc, Louis est arrivé à ce moment-là, il a pris ce rôle de Product Owner, mais on n'a engagé personne dans l'équipe pour la mise en place de la méthodologie Agile en tant que telle.

Comment avez-vous implémenté Scrum : tout en une fois ou c'était par petites étapes ?

M.G. : Non, on s'y est mis quasiment en une fois en fait. Donc après le suivi de la formation on s'est dit maintenant on va commencer à planifier les choses. Du jour au lendemain, on a démarré toutes les cérémonies en fait. Stand up meeting tous les matins, des sprints de 2 semaines, des rétrospectives, des sprint planning, etc. On a démarré d'un coup en fait, parce qu'on s'est dit : ici on a les bases, on va tout mettre en place d'un coup puis on va affiner en fonction des choses qui nous conviennent et des choses qui ne nous conviennent pas. Jusqu'à maintenant, cela fait depuis le mois de septembre, quasiment 9 mois. On a gardé une structure quand même assez stricte par rapport à ce qu'on avait démarré donc cela veut dire qu'on est toujours au stand up quotidien, ça nous convient on fonctionne bien comme ça. Maintenant, on a un peu arrangé à notre façon les choses comme la rétrospective que l'on fait selon ce qu'il nous convient, etc, avec parfois des choses qui décalent d'un peu d'Agile, mais qui nous permettent de travailler. Même chose pour le sprint planning par exemple, on fait ça à notre façon, mais on se rend compte que ça fonctionne.

*Est-ce dû à des difficultés que vous avez un peu changé par rapport votre façon ?*

M.G. : Oui, la volonté de passer en méthodologie Agile c'est qu'on démarrait un nouveau projet parce que finalement le développement d'OSIS tel qu'on le fait maintenant a démarré fin 2016. Je suis arrivé dans l'équipe en mars pour renforcer l'équipe de développement et petit à petit on s'est rendu compte que ça ne tournait pas comme on le voulait., on avait des contacts avec le business, mais qui ne nous permettaient pas, au fur et à mesure, d'amener de la plus-value pour eux. On se rendait compte qu'on commençait un peu à tourner en rond, à ne pas savoir où on allait exactement. Recadrer les choses et donc voilà l'équipe avait entendu parler de la méthodologie Agile. C'est une volonté qui est venue aussi de notre hiérarchie. On va essayé quelque chose comme ça puis ça a bien tourné.

*C'est venu plutôt de votre part ou plutôt celle de la hiérarchie ?*

M.G. : C'est venu plutôt de la hiérarchie, mais l'équipe était bien consciente qu'il y avait un problème aussi. On était tous plus ou moins alignés pour dire qu'il fallait essayer quelque chose de nouveau et donc on s'est lancé là-dedans.

*Est-ce que vous aviez des attentes particulières concernant la mise en place des méthodes Agiles ?*

M.G. : Je peux parler pour moi. Personnellement, j'étais intéressé parce que j'avais travaillé avec une équipe pendant presque 10 ans où dans le support informatique on ne peut pas

mettre en place ce genre de chose, et j'ai moi-même un intérêt particulier dans ce genre de méthode dans le fait de se dire tiens on va essayer d'innover un peu la façon dont on travaille. J'étais curieux et intéressé de ce qu'on pouvait faire de mieux avec cette méthodologie. Et effectivement, la formation qu'on a suivie nous a ouvert les yeux sur une série de choses qu'on faisait ou qu'on ne faisait pas et qui pourtant en les mettant en place permettent d'améliorer le flux de travail, etc. Ne serait-ce que la petite réunion quotidienne qui permet d'aligner chaque personne, savoir si quelqu'un a des difficultés, mettre en place des choses simples comme le pair programming, des choses comme ça, ce qu'on ne faisait pas du tout avant. Je me corrige : on le faisait, mais sans savoir exactement ce qu'on faisait, ici on le fait de façon un peu plus structurée. Oui, personnellement, moi ça m'a beaucoup intéressé et je me suis rendu compte par la force des choses qu'on a commencé à produire plus de plus-value business en travaillant comme ça. Pour ma part, ça a été positif en tout cas.

Y avait-il des attentes aussi de la part de votre hiérarchie ou de l'équipe ?

M.G. : Oui. Clairement, la hiérarchie attendait que grâce à la méthodologie Agile, même si on s'est rendu compte pertinemment que ce n'était pas le but de la méthodologie, mais il y avait une attente de pouvoir sortir un planning, de pouvoir dire : « Voilà le logiciel il sera à telle étape, à telle date, on pourra en sortir telle chose pour tel moment ». Et donc pouvoir planifier sur le plus ou moins long terme, l'évolution de l'équipe et du logiciel. Donc, maintenant on s'est rendu compte qu'Agile ça ne sert pas exclusivement à ça, ça peut aider. Mais l'attente de la hiérarchie était là aussi de voir l'équipe mieux s'organiser pour travailler d'une manière plus structurée, clairement il y avait une attente de leur part, et de la part de l'équipe globalement aussi. Peut-être pas de l'équipe en entier, mais je pense que globalement on peut dire que l'équipe est satisfaite d'avoir une structure de travail qui soit un peu plus efficace que ce qu'on avait avant, en tout cas.

*Avez-vous trouvé des avantages que vous n'aviez pas soupçonnés pendant l'implémentation de ces méthodes ?*

M.G. : Bonne question. Des avantages insoupçonnés. Oui, oui, parce que par exemple : le stand up meeting encore une fois, on en parle beaucoup, mais effectivement c'est quelque chose qui m'a marqué parce que quand on entendait parler de ça au début on se dit « 15 minutes tous les matins », qu'est-ce qu'on va dire ? Au final, on découvre là-dedans des choses qu'on aurait pas soupçonnées. On se rend compte très vite quand un développeur est bloqué sur quelque chose là où dans une méthodologie plus classique, il aurait peut-être passé

3-4 jours à tourner dans son coin à essayer de résoudre son petit problème. Là, le matin on en discute : « Tiens, tu n'arrives pas à avancer là-dessus, quelqu'un va venir t'aider », ou bien : « on va voir ensemble quelle solution technique on peut adopter pour régler ton problème ». Oui, ça c'était insoupçonné pour ma part. Je ne m'attendais pas à ce qu'une simple rencontre d'un quart d'heure quotidienne puisse finalement améliorer le travail de beaucoup de personnes et la communication dans l'équipe aussi. On sait bien ce que font les autres, on voit ou en sont les différentes choses de façon court terme sur un sprint de 2 semaines, mais aussi à plus long terme. On sait, voilà telle et telle personne va aller rencontrer telle personne du business pour avoir telles informations. On sait comment ça évolue à plus long terme et donc c'est vraiment à l'équipe d'être aligné aussi dans les connaissances de l'évolution du projet. Donc ça c'est un avantage insoupçonné. Bien sûr.

*Savez-vous pourquoi vous avez choisi Scrum et pas une autre méthode ?*

M.G. : Bonne question. Parce que c'est à la mode, à mon avis. Parce que tout le monde en parle parce que c'est un truc à essayer, parce que notre projet est chapeauté par le Professeur Yves DEVILLE qui est conseil au-recteur à l'université numérique et qui lui est sensible à cette méthode qui, finalement, a insufflé ça dans le projet. Et il fallait quelque chose et on s'est dit : « Pourquoi pas Agile, ça a l'air de fonctionner pas mal dans le coin, on va essayer ça » et ça a bien pris.

*Vous m'aviez parlé de modifications ou adaptations des pratiques que vous avez faites, comment en êtes-vous arrivé à faire cela ?*

M.G. : Quelque part en prenant les recettes d'Agile qu'on a reçu pendant la formation en se disant bon on va les appliquer bêtement, on va voir ce que ça donne et c'est ce qu'on a fait. Il y a des choses qui sont restées tout aussi habituel, il y a des choses qui se sont un peu estompées. Je n'ai pas d'exemple concret, mais moi le pair programming, j'en parlais, c'est assez régulier chez nous et il y a d'autres choses, par exemple, - j'ai oublié le nom de la réunion, ça prouve qu'on ne la fait pas souvent – c'est la réunion présentation qu'on fait normalement à chaque itération au business. Ça on en fait de temps en temps, mais ce n'est pas régulier. Le petit problème que l'on a chez nous par rapport à ça c'est qu'on construit un tout nouveau logiciel qui n'est, bien sûr, pas fonctionnel à 100%. Et le business, les gens de métier de l'autre côté, travaillent encore principalement dans l'ancien logiciel qui est EPC. Les choses que l'on est en train de développer actuellement sont pas encore pour l'instant, à ce jour-ci, utilisés par nos utilisateurs. Donc c'est très compliqué d'avoir un retour régulier et

concret de leur part, mais ici on arrive à une charnière parce que d'ici quelques temps la première partie du logiciel qu'on a développé va être mise en production et être utilisée et fermée dans l'ancien logiciel en fait cela va permettre qu'on ait des utilisateurs qui travaillent vraiment qu'avec ce que l'on a produit et d'avoir des retours plus réguliers. Et éventuellement de remettre en place ces réunions qui permettraient régulièrement de rencontrer le business, voir les besoins, incrémenter notre logiciel pour répondre aux besoins ou corriger les bugs, par exemple et très régulièrement pouvoir leur montrer, selon leur demande des fois précédentes, on a avancé comme ça et comme ça.

*Vous pensez que l'implémentation de la méthodologie suit des objectifs stratégiques dans votre équipe ?*

M.G. : Oui on est guidés par des objectifs. Parce qu'au début on tâtonnait un peu, on ne savait pas trop comment se lancer, on avait déjà développé, mais on tournait un peu en rond à certains moments. Puis, il y avait un objectif clair c'était de pouvoir fournir plus de valeur au business, pouvoir avancer dans le projet finalement. On a un logiciel EPC qui est vieillissant, qui commence à poser des problèmes, qu'il faut remplacer et donc l'objectif stratégique est celui-là. Le gros objectif stratégique c'est mettre de côté EPC, remplacer par OSIS. On est sur du long terme, on est ici actuellement dans les plannings sur un délais de 7 années plus ou moins. Donc on est à long terme, mais sans la méthodologie Agile, cet objectif stratégique on ne l'aurait peut-être jamais atteint, je pense. On serait resté avec cette idée, oui il faut remplacer EPC, il faut changer ce système qui pose des problèmes, mais comment on va y arriver ? Tandis qu'ici la mise en place de la méthodologie nous aide déjà à avoir déjà une planification. Mais déjà sur le long terme je peux vous dire qu'on est plus ou moins à 7 années grosso modo, c'est quelque chose qu'on n'aurait pas pu dire il y a un an. On n'aurait pas pu dire quand est-ce qu'on va finir, impossible. On savait qu'on avançait, mais, mais on ne savait pas à quel délai, on ne savait pas à quelle vélocité, on ne savait pas ce qu'on était capable de produire, en combien de temps, etc. On a maintenant la possibilité de faire. Ce n'est pas uniquement de la mesure, le but ce n'était pas de pouvoir donner des délais exacts, mais c'était de pouvoir avoir des avancées plus concrètes que ce qu'on avait.

*Comment jugez-vous qu'un projet soit un succès ?*

M.G. : Un succès ? Déjà par la satisfaction des personnes qui l'ont mis en place. Je ne vais pas juger d'un projet qui est déjà terminé parce qu'ici on est dans un projet qui est plutôt dans les débuts que dans la fin et dans la terminaison. Mais le succès actuel de notre projet c'est de

pouvoir dire que les développeurs sont appliqués, les développeurs et les personnes autour sont appliqués, on se rend compte qu'on avance. Maintenant, le vrai succès du projet c'est, je pense qu'il sera défini lorsqu'on l'aura terminé, mais c'est pour nous un horizon assez lointain. Donc c'est assez difficile d'évaluer de ce côté-là, mais un succès au jour d'aujourd'hui c'est de voir par exemple que les diffusions des notes d'examens se font de manière moins pénible qu'avant. On avait, il y a quelques années des étudiants qui ne savaient aller voir leurs notes d'examens le jour « j » et on a amélioré ça en répercutant une partie de la charge de travail dans le nouveau système. Des exemples de succès c'est le travail qu'on fait au quotidien c'est améliorer le travail des gens qui travaillent à l'UCL, améliorer la vie des étudiants dans la consultation de leurs notes, les inscriptions aux cours, les inscriptions aux examens, et il y a clairement la satisfaction de notre « client » même si on travaille en interne à l'UCL et qu'on n'a pas un vrai client. Mais on considère nous que notre client ce sont les membres du personnel qui travaillent et les étudiants qui étudient qui ont besoin de connaître leurs notes, d'encoder leurs inscriptions aux examens, de s'inscrire aux cours, etc. C'est ce qui nous fait avancer.

*La méthodologie Agile a-t-elle pu vous aider, a pu faciliter ce succès ?*

M.G. : Oui, je pense, clairement, dans le sens où on se rend compte que grâce à la méthodologie Agile on voit plus clair par rapport où l'on va et on fournit plus de valeur au business, même si actuellement comme je l'ai dit, le logiciel n'est pas tellement utilisé. En tout cas, pas pour la partie qu'on développe actuellement. J'ai parlé des inscriptions aux examens, par exemple, c'est une des choses qui se fait déjà. Mais la méthodologie nous aide à aller vers ça, vers quelque chose qui fonctionne et qui permet à notre « client » d'être satisfait.

*Est-ce que vous avez pu faire une évaluation de vos 9 mois avec la méthodologie Agile ?*

M.G. : On n'a pas encore réellement évalué, même si on le fait de façon incrémentielle, lors de nos rétrospectives. Systématiquement, on se demande : « Tien, qu'est ce qui fonctionne ? Qu'est ce qui ne fonctionne pas ? Qu'est qui dans la méthodologie Agile nous aide à avancer ? Qu'est ce qui nous freine ? ». Donc on n'a pas encore réalisé d'évaluations concrètes après un laps de temps fixé en se disant « Depuis les 9 derniers mois, qu'a t'on amélioré dans notre façon de faire ? ». On s'évalue de façon incrémentielle.

Donc je pense que tout ce que j'ai pu vous dire par rapport au changement de fonctionnement de l'équipe va dans ce sens-là. C'est quelque chose qu'on constate au quotidien. Ce n'est pas en évaluation formelle, mais on se rend compte de l'amélioration des choses en tout cas.

*Est-ce que vous avez pu encore découvrir d'autres améliorations par rapport à la méthodologie ?*

M.G. : Depuis la formation qu'on a suivie on est un peu en roue libre, c'est-à-dire qu'on a affiné les choses par rapport à notre besoin par rapport à la méthodologie, maintenant on a décidé d'essayer de se mettre à jour. Une partie de notre équipe va, par exemple demain, à une rencontre Agile qui a lieu à Bruxelles, par un organisme de formation donc on a quand même la volonté de pouvoir se mettre à jour pour discuter avec d'autres équipes qui mettent en place la méthodologie Agile. Maintenant, ça commence. On n'est à 9 mois, c'est-à-dire que jusque maintenant on a peu appliqué les recettes classiquement, puis on a affiné en fonction de ce qu'on avait besoin. Je pense qu'on va entrer dans une phase où on va essayer de trouver de meilleures idées. De voir ce qu'on va pouvoir faire de plus, quelles sont les choses auxquelles on ne pense pas, quelles sont les choses qu'on a zappées, qu'on ne met pas en place alors qu'on pouvait très bien faire pour améliorer encore notre façon de travailler. C'est en cours.

*Maintenant d'un point de vue personnel, est-ce que vous êtes globalement satisfait jusqu'à présent de la méthode Agile ?*

M.G. : Oui, ça c'est clair.

*Apporteriez-vous encore quelques améliorations ? Et si oui, lesquelles ?*

M.G. : Oui, j'espère, parce que si on n'apporte plus d'amélioration ça veut dire qu'on reste bloqué dans notre façon de travailler et il y a toujours mieux à faire, je pense. On a toujours des possibilités, on le voit régulièrement. J'ai un bête exemple : on avait des indisponibilités dans notre trio de Scrum Master, on a demandé à un autre membre de l'équipe : « Est-ce que tu ne veux pas, pour essayer pour voir ce que tu peux apporter, prendre le rôle de Scrum Master pendant un sprint ». C'est ce qu'il a fait, il a relevé le défi. Il a dit : « Oui je vois comment vous avez travaillé jusqu'à maintenant, je pense que je peux essayer pendant un sprint, et rien que de ce fait là on a pu trouver de nouvelles idées ». Le collègue est arrivé avec des petits changements dans la façon de fonctionner, mais il nous a donné des idées des choses que maintenant dans l'habitude quotidienne de l'équipe on a implémenté. Donc, un

petit exemple, ce qu'il nous a proposé de faire c'est lors des réunions, les petites réunions Agile, donc le stand up meeting quotidien par exemple, s'il y a des points qui sont soulevés dont il faut encore discuter, de les noter sur un post-it, et on le colle sur le tableau jusqu'à ce qu'il soit résolu. Comme ça on ne l'oublie pas. Alors qu'avant on disait qu'on en discuterait après, mais on n'en discutait plus. Le lendemain, on repasse dessus : « Est-ce que cette question est réglée, elle est réglée, on jette le post-it, elle n'est pas réglée, on en parle. Ça permet de concrétiser les choses et on n'est plus avec des points dont on dit on en reparlera après et puis on se rend compte après 2 semaines qu'on n'en a plus jamais parlé. C'est resté en suspens, ça peut poser problèmes et on a zappé. On continue à s'améliorer et je pense que d'office par la pratique au quotidien on va encore changer des choses et si on s'engluie dans des habitudes sans réelle remise en question, je pense qu'on arrête d'avancer. J'espère qu'on changera encore des choses.

*Quel est votre avis concernant les méthodes Agile en général ?*

M.G. : Je suis assez positif personnellement. Maintenant on se rend compte qu'il n'y a pas que du positif. Un des reproches qu'on entend souvent de la part de collègues proches, mais qui ne sont pas dans l'équipe directement, c'est qu'on passe énormément de temps dans des réunions ou aux séminaires Agiles : « Mais à quoi ça sert ? ». Mais je pense qu'en étant dedans au quotidien on se rend compte qu'il y a une utilité là-dedans.

*Est-ce que vous pourriez donner une définition des méthodes Agile ou du Scrumban que vous utilisez maintenant par rapport à votre expérience ?*

M.G. : C'est compliqué. Je pourrais dire un ensemble de conseils et de bonnes pratiques qui permettent à l'équipe de structurer son travail, de façon plus fluide que ce qu'on aurait pu faire avant, avec un avancement plus par tâtonnements. C'est une structure de la façon dont on travaille.

*Connaissez-vous l'Agile manifesto ?*

M.G. : Oui, plus par cœur, mais c'est le genre de chose qu'on a abordé lorsque qu'on a suivi la formation.

*Quel rôle cela joue dans votre organisation ?*

M.G. : Cela n'a plus un rôle vraiment concret, on en a pris connaissance, mais c'est quelque chose dont on ne se souvient pas au quotidien. Mais le fait que vous m'en parlez, je me dis

que je devrais aller le relire et peut être même l'afficher dans le bureau pour qu'on en soit conscient au quotidien puisque finalement c'est la base des choses.

*Pensez-vous que ça pourrait améliorer votre quotidien ?*

M.G. : Oui, je pense que si on a les yeux sur les principes de la méthodologie qu'on utilise quotidiennement ça peut recadrer parfois les choses. Parfois, on peut se demander, mais pourquoi on fait cette chose-là de telle façon et puis on se rend compte qu'il y a une base là derrière. On ne fait pas ça n'importe comment, ou sans fondement. Donc oui, le Manifesto on l'a un peu oublié, mais pourtant cela ne fait que 9 mois. Et je pense que ça peut fait du bien d'y retourner.

*Quelle conclusion pouvez-vous tirer de votre expérience, de ces 9 mois ?*

M.G. : La conclusion c'est un peu ce que j'ai dit jusqu'à maintenant. On se rend compte qu'on s'est amélioré dans la façon dont on travaille, qu'on peut encore s'améliorer. Agile c'est ça, c'est accepter le changement. Il y a ça aussi, je ne l'ai pas abordé jusqu'à maintenant, mais l'équipe pouvait avoir certaines réticences auparavant à se dire : « On a développé la chose d'une certaine façon, maintenant on doit la changer et puis après il faut encore la changer ». Et maintenant, on se rend compte que c'est comme ça qu'on peut travailler et évoluer. C'est-à-dire qu'on développe quelque chose et puis, c'est remis en question. On accepte le changement, et donc la conclusion est là : c'est quelque chose qui a permis à notre équipe de se remettre en question, tant de façon commune pour l'équipe que de façon individuelle pour certaines personnes et de se rendre compte qu'il y a toujours moyen de s'améliorer. Pour moi, c'est ça Agile en fait. On n'est jamais arrivé au bout de la façon dont on travaille, il y a toujours quelque chose de mieux à faire.

## Appendix 12 : Interview Michel Herquet

M.H. : Managing Partner B12 Consulting

*Pouvez-vous vous présenter en quelques mots sur vous et votre expérience ?*

M.H. : Je m'appelle Michel Herquet, je suis un des 3 managing partner de la société B12 consulting. C'est une société relativement jeune et relativement petite. On l'a créée en 2012. À l'heure d'aujourd'hui, je sais qu'on sera 17 en septembre. On est une société de services en informatique, on offre essentiellement trois grands types de services. Le service principal qu'on offre c'est le développement software custom, donc à la demande du client, exclusivement en B2B. On fait ce qu'on appelle du data science, c'est tout ce qui est analyse de données avancées avec des technologies avancées. Et on fait un peu de conseil stratégique classique, donc ce qui est tout autour de l'IT.

*Quelle est votre expérience et formation ?*

M.H. : Moi j'ai une expérience : fondateur, on a tous les trois un background plutôt en recherche scientifique. Donc on a fait tous les trois une thèse de doctorat ici à l'UCL. Et puis après la recherche, on a tous les trois un peu des backgrounds différents. Vincent a un background plutôt technique, dans tout ce qui est big data. Geof est plutôt opérationnel dans tout ce qui est développement de software dans le secteur bancaire, il a travaillé chez Sopra banking également. Et moi, j'ai travaillé chez Mc Kinsey comme consultant stratégique.

*En quoi consiste votre fonction dans l'entreprise ?*

M.H. : On est une société de conseil ce qui veut dire qu'on est structuré comme une société de conseil et donc on a des niveaux relativement classiques, qu'on observe essentiellement dans toutes les sociétés de conseil. On a des analystes développeurs qui sont à la recherche plutôt de l'exécution, ou plutôt de développement, on a des « associates » qui sont en charge de leur propre workstream, qui sont plutôt des indépendants et qui peuvent travailler avec grande indépendance sur les problèmes spécifiques prédéfinis. Des project managers qui organisent des projets et qui gèrent les projets. Et on a les trois partners, pour l'instant on est trois, il y en aura sûrement d'autres à l'avenir, on est surtout en charge de l'organisation interne et de la relation client.

*Est-ce que vous avez toujours travaillé avec les méthodes Agiles ou vous avez travaillé avec d'autres méthodes auparavant ?*

M.H. : Alors oui, malheureusement peut-être pour ton mémoire, mais oui on a utilisé que les méthodes Agiles parce que quand on a fondé la société... Donc moi, j'avais déjà eu l'occasion de travailler avec les méthodes Agiles en recherche scientifique. Et notamment avec eXtreme Programming qui est une méthode Agile qui est spécifique au software development et qui était une méthodologie que j'aie appliqué des années auparavant dans le cadre du développement d'un logiciel scientifique pendant mon doctorat. Et en fait les tous premiers problèmes qu'on a été confronté dans le développement on n'aurait sans doute pas su les faire autrement, simplement. Et c'est resté notre marque de fabrique, on travaille sur des projets relativement intenses, relativement court, avec des délais qui sont très challenging pour l'industrie. Notre tout premier développement software on a parlé de faire des e-learning custom from scratch donc à partir de rien, en 6 mois, pour ainsi dire. Et en partant de rien en termes de requirement list, on avait pas une analyse de 1 ou 2 ans, et il y avait des sales qui avaient promis monts et merveilles à des clients qu'on devait faire, mais en termes d'analyse fonctionnelle c'était très limité. Et donc tu n'as pas le temps de faire l'analyse fonctionnelle, parce que si tu fais l'analyse fonctionnelle complète comme en Waterfall classique bien tu passes 6 mois et puis tu n'as rien fait au bout de 6 mois enfin je veux dire tu n'as rien développé au bout de 6 mois. Donc le mieux de toute façon, c'est d'avancer en Agile. Comme on s'est spécialisé sur ces projets-là, en fait on s'est spécialisé sur tous les projets où Agile est crucial et donc de facto on ne sait rien faire d'autre qu'Agile. Donc voilà.

*C'est pour un objectif purement stratégique alors le fait d'avoir implémenté les méthodes Agiles ?*

M.H. : Pour nous, ça fait quasiment partie de notre ADN. C'est plus que stratégique, c'est presque notre signature si tu veux. On aurait pu mettre B-12 Agile consulting si tu veux, pour nous c'est dans l'essence même de la société.

*Comment vous avez- implémenté les méthodes ? Donc vous avez d'abord implémenté eXtreme Programming ?*

On n'a jamais implémenté formellement aucune méthodologie Agile. Alors, c'est un sujet qui a été longtemps débattu. Moi je trouve toujours un peu bizarre d'être extrêmement à cheval sur des méthodologies Agiles par rapport à une autre philosophie Agile. Quand je parle de

philosophie Agile je parle d'un ensemble de good practices qu'on met en place qui ont une certaine marge de flexibilité dans leur application en fonction avec les clients avec qui on travaille et ainsi de suite, plutôt que Scrum : « on fait comme ça et pas comme ça ». Je trouve qu'implémenter très rigoureusement, très rigide une méthode Agile c'est presque antinomique avec la définition même de méthodologie Agile. Je pense qu'Agile dans la façon qu'on implémente Agile. Donc dans la manière d'avoir une méthodologie Agile ou on est extrêmement à cheval, on a plutôt différentes approches qui ont des choses en commun c'est-à-dire la même culture, mais en fonction des clients ça va varier. Un point important pour nous c'est les outils d'implémentation. Parce que l'Agile c'est une culture, mais aussi des choses concrètes. Et une chose qu'on utilise c'est un outil qu'on appelle Pivotal Tracker. C'est un des outils de gestion de stories classique dans notre domaine, on en a plusieurs, mais Pivotal Tracker a le mérite d'être simple, d'être efficace, et de parfaitement correspondre à notre culture. Donc en fait notre implémentation d'Agile c'est de travailler autour de Pivotal Tracker. Donc c'est une logique. Pivotal Tracker est un mix entre Kanban et Scrum, un KanScrum ou un ScrumBan, il y a différentes appellations. Donc c'est un peu ça, ça tire un peu le meilleur des deux. Ce n'est pas que le Kanban parce qu'il y a aussi une notion de cycle, de release. Kanban c'est un peu du développement continu, donc tu as des releases, mais tu peux quasiment faire des releases tous les jours et donc ça c'est un peu trop extrême. Et en plus dans le Kanban il manque un rythme. Il y a plus la notion d'itération dans le Scrum que dans le Kanban, car la notion d'itération permet de donner un rythme. Mais ce qu'on n'aime pas dans Scrum c'est qu'on a un commitment un peu arbitraire que dans un sprint. Donc comme il y a des désavantages entre les deux on préfère suivre pivotal tracker qui est une méthode qui est à cheval entre les deux. Donc c'est essentiellement la logique d'un Kanban, mais avec des itérations ;

*Si j'ai bien compris, dès le début vous avez pris pivotal tracker ?*

M.H. : Oui c'est le premier qu'on a utilisé.

*Avez-vous dû faire des adaptations depuis la première utilisation ?*

M.H. : Pivotal Tracker a ça d'unique, c'est qu'il n'est absolument pas configurable donc on a travaillé sur pas mal de projets pour des clients avec Gira, ou d'autres outils qui sont très configurables. Mais trop configurables à notre goût. Ce qu'on trouve bien chez Pivotal tracker, c'est que comme il n'est pas du tout configurable, on ne sait absolument rien faire, on

ne sait rien absolument rien changer – enfin il y a juste deux, trois paramètres-, mais ça force un peu tout le monde à suivre la même logique

*Donc par rapport aux projets vous changez avec Gira ?*

M.H. : Non on fait toujours avec Pivotal Tracker. Après il y a des projets où va utiliser intensément Pivotal Tracker tout au long du projet, et il y a d'autres projets où l'on va « être plus relax » et le niveau de description des stories va être différent aussi. Il y a des projets où on travaille avec des descriptions relativement high level et ça va surtout être avant tout un outil de communication avec le client en disant : « voilà on a fait ça, et on a fait ça » et surtout être un outil de priorisation classique avec Trello ou ce genre de trucs. Alors que dans d'autres projets on va être super à cheval, tous les développeurs vont tous être sur Pivotal Tracker devant eux du matin au soir. C'est plus l'intensité de l'utilisation de l'outil qui va varier, mais la logique est toujours la même.

*Est-ce que vous avez utilisé certaines formations au début quand vous avez implémenté Agile ?*

M.H. : Non moi j'étais le plus exposé à ça, ça vient d'un bouquin Ken Becket je pense, quelqu'un qui a inventé eXtreme programming. Un bouquin qui est gratuit qu'on retrouve en PDF en ligne, et qui est un bouquin de 40-50 pages qui exprime vraiment la base. C'est un des signateurs de l'Agile Manifesto. Et voilà, c'est la base, c'est avant tout des réflexions culturelles sur le fait de préférer une façon de travailler par rapport à une autre.

*Selon vous, pivotal tracker atteint les objectifs que vous vouliez ?*

M.H. : Oui, parce qu'après on l'utilise avec un degré d'utilisation qui varie, mais en tout cas quand on l'utilise vraiment de façon extrême, quand on fait vraiment de l'eXtreme Programming, ça fonctionne vraiment hyper bien. Quand on a un problème avec pivotal tracker, c'est plutôt de notre faute que celui de l'outil. Je vais prendre un exemple très concret, on a des projets qui font trois semaines, ce sont des projets qui vont être très précis ou quoi. On sait que ce seront des projets qui ne fonctionneront pas vraiment en Agile pur, ce sont des projets à budget assez limité et donc on va être plus dans un genre de waterfall qu'on priorise un peu high level sur Pivotal Tracker. Mais voilà, donc on opère de façon vraiment Agile, en tout cas sur les projets long terme ça marche très bien.

*Est-ce que vous avez évalué vos projets par rapport à pivotal tracker ?*

M.H. : On est une petite société, donc on n'aura pas vraiment de processus formé d'évaluation. On a un petit packaging de 10 commandements et on essaye de s'y tenir. Commandements culturels de good practices si tu veux, on fait le point de façon informelle sur à quel point on a été bon dans tel ou tel projet. Par contre, ce qu'on a été amené à faire c'est, pour des clients, dans le cadre de conseils, des évaluations sur différents outils comme Gira, Pivotal Tracker et ainsi de suite. En général, c'est la meilleure solution pour ce client-là. Donc voilà, en interne, on n'a jamais vraiment fait ça. Encore une fois on insiste plus sur la culture et sur les good practices que sur quelque chose de très rigide. Nous ce qui compte avant tout c'est que 1. Les clients soient heureux et c'est la métrique numéro 1. Et deux que les développeurs soient heureux et ne soient pas surmenés du matin au soir ou irrités parce qu'ils n'ont pas l'impression de contrôler ce qu'ils font. Ce sont surtout ces deux métriques-là. Et ça on voit tout de suite si le projet marche ou marche pas.

*Et l'évaluation auprès des clients, quels résultats a-t-elle donnés ?*

M.H. : On a eu une évaluation un peu plus formelle il n'y a pas longtemps. Il y a une stagiaire en communication qui a fait une évaluation un peu plus formelle en discutant un peu avec tous nos clients. Et la majorité des clients qui jouent le jeu, pour eux ça se passe très bien. Après il y a des clients pour qui, qui vont avoir moins de temps à dédier à l'outil ou qui vont avoir moins de temps, ils seront réticent, avoir moins envie de travailler avec l'outil. Et là, pour qu'un processus Agile marche bien il faut un Product Owner, son rôle est important, mais il faut une contrepartie du côté du client qui connaissent vraiment le côté produit. Et ça si on l'a pas, il faut compenser, et c'est difficile de compenser dans le processus Agile. Et là, il y a un réflexe naturel de faire valider avec le client avant. Mais en tout cas avec nos projets long-terme ça se passe très bien avec nos Product Owner qui travaillent avec Pivotal Tracker et qui sont dans le même accord que nous.

*Je rebondis donc sur le fait que vous utilisez parfois waterfall, cela vous arrive quand même d'utiliser le modèle waterfall ?*

Non, sauf dans des circonstances plutôt pathologiques. Nous tous nos projets, notre partie pris ça va être Agile. Après il y a des projets qui sont tellement courts. Si tu as un projet de trois semaines, le plus petit que tu puisses faire en itération Agile c'est une semaine, en-dessous je n'en connais pas qui travaille. En général, c'est deux semaines. Mais trois semaines, si tu retires le kick-off ça fait deux itérations. Et en deux itérations, le client il n'a pas le temps d'interactions. Donc ce genre de projet-là, tu le fait mieux en Waterfall en disant : « voilà le

mock-up, pour une page voilà ce qu'on va faire ». Et à la fin du projet on fait le point et éventuellement si on rentre dans une logique d'amélioration continue pour ce premier prototype, là on rentre dans le processus Agile. Mais pour ce tout premier élément de travail, dire qu'on va faire des sprints ou quoi ça n'a pas tellement de sens. Donc quand un projet est extrêmement court paradoxalement Agile n'est pas la meilleure approche. Parce qu'on n'a pas le temps. Agile il y a quand même un overhead de gestion, la plupart des clients ne connaissent pas Agile. Ou ils pensent connaître, mais ils ne connaissent pas. Et puis la plupart des gens connaissent Waterfall, quand on construit une maison, c'est Waterfall, c'est l'exemple des architectes. Nous ce qu'on fait c'est que si le projet est très court, on dit : voilà ce qu'on va faire. À la fin du projet, il y a toujours un delta : on note ce qui va être fait. Mais à ce moment-là, il est toujours temps d'aller dans un processus Agile. Mais dans ces toutes premières phases on est toujours un peu Waterfall.

*D'un point de vue personnel, pensez-vous que l'adoption des méthodes Agiles doit poursuivre un certain but stratégique ? Ou est-ce que parfois c'est juste implémenter Agile ?*

M.H. : Implémenter Agile, implémenter toute technique de management, et l'implémenter juste parce que c'est à la mode, c'est forcément une hérésie. On suit toujours un objectif. Nous notre objectif c'est quoi ? L'essence d'Agile pour nous, c'est de faire du changement une opportunité plutôt qu'un problème en résumé. Donc le changement dans quasiment dans toutes les méthodes de management classique, que ce soit Agent Whart, Waterfall classique et ainsi de suite, le changement est vu comme un problème qu'il faut gérer. Et il y a des méthodologies aussi pour gérer ces problèmes. Ce que nous on voit dans Agile, c'est qu'on peut en faire une opportunité, pour faire des meilleurs produits et plus vite. Et de le faire de façon moins pénible pour tout le monde. C'est pour ça. Donc pourquoi on poursuit Agile c'est parce qu'on y voit une opportunité. A fortiori on voit, dans un monde qui s'accélère, le monde du changement est quelque chose qui permet de faire un ajustement, ce n'est pas juste de temps en temps qu'il y a du changement, mais tout le temps. Mais aussi dans des phases de go-to-market beaucoup plus rapide qu'avant. Un produit IT tu vas devoir développer 5 ans avant qu'il soit sur le marché. Maintenant plus personne ne ferait ça, ce serait suicidaire. Quand tu as un go-to-market qui varie maintenant de 3 à 6 mois c'est évident que tu dois être Agile.

*Donc pour vous, si j'ai bien compris, c'est mieux d'avoir un certain objectif quand on implémente Agile ?*

M.H. : L'objectif c'est toujours ces trois choses-là, c'est avoir des clients plus contents, d'avoir un produit qui répond plus à la demande et ça touche un peu les start-ups qui ont cette stratégie-là. Avoir un produit qui est mieux adapté à ce que le client veut, avoir une adaptation au changement et au vent contraire. Par exemple, on travaille beaucoup avec des start-ups, les start-ups leur priorité un mois n'est pas leur priorité 3 mois suivants. Et donc il faut pivoter deux fois entre-temps et voilà, ce qui était super urgent avant, peut tout d'un coup ne plus être urgent, car une autre urgence est plus importante. Donc il y a des clients plus contents, il y a une adaptation au changement et il y a aussi l'aspect bonheur interne. Un des problèmes de Waterfall c'est de subir quelque part, il y a rien de pire que d'avoir un projet Waterfall qui part en sucette après 1 mois. Parce que tout le monde subit la fin du projet avec des contrats qui viennent de nulle part, personne n'essaye de comprendre, des priorités qui semblent débiles. Donc il y a une part de bonheur interne aussi.

*Quel est votre avis sur les méthodes Agiles en général ?*

M.H. : Je ne pourrais pas dire qu'il est négatif. Pour moi elles sont clairement les meilleures pour beaucoup de situation mais pas pour toutes. Pour nous, dans 80,90% des cas. Il faut juste bien faire attention quand ça va pas avec les méthodologies parce que par exemple adopter Agile quand ce n'est pas la bonne méthodologie, c'est se planter à coup sûr de 1. Et surtout ne pas penser que c'est de faire des économies. Ça je pense que c'est le risque numéro 1. Il y a plein d'entreprises qui se disent : « Ah Agile, ça va coûter moins cher ! » Or c'est faux, Agile en moyenne ça va coûter plus cher, si tu regardes le même produit, le temps de vie de produit, 5 ans par exemple et que tu regardes la feature list au bout des 5 ans, et tu te poses la question, si j'avais eu cette feature list au premier jour combien ça m'aurait coûté de le faire en Waterfall en 5 ans, et combien ça m'a coûté de le faire en Agile en partant d'une feuille blanche. Ça coutera toujours plus cher en Agile, car tu auras toujours du refactoring, tu auras toujours des changements de code, tu auras toujours des choses à mettre à la poubelle. Tu as toujours une composante inutile, tu auras toujours du travail jeté à la poubelle qui est inévitable. Le point c'est que c'est souvent la seule solution puisqu'on a pas la feature list finale sous les yeux. Si tu veux Waterfall c'est la méthodologie idéale si tu as une feature list dès le début. Mais voilà le problème c'est qu'on en a pas. Donc il faut quand même rendre à César ce qui est à César je ne pense pas que les méthodologies Agiles soient moins chères ou même la plus efficace. C'est la seule méthodologie que quand tu n'as rien qui soit efficace. Ce qui fait que c'est la grande méthodologie la plus utilisée aujourd'hui.

*Connaissez-vous l'Agile Manifesto ?*

M.H. : Oui

*Et pour vous, quel rôle ça a dans votre entreprise ?*

M.H. : On ne l'a pas placardé sur tous les murs non plus. Moi ce que j'aime bien dans le Manifesto c'est la philosophie, les mots sont très bien choisis, c'est très court, c'est très facile à comprendre et c'est très culturel et philosophique que ce n'est un process défini qui donne une invasion culturelle. Et ce n'est pas non plus trop invasif. Si tu lis le Manifesto, c'est : quand c'est possible nous préférons faire ça plutôt ça que ça, la communication par rapport à la documentation. C'est ce genre de chose. Mais ils disent bien quand c'est possible, et quand c'est adéquat. Ils ne disent pas : oh la documentation c'est mal il faut faire la communication. Ce n'est pas dictatorial l'Agile Manifesto, voilà ce sont des développeurs qui se sont demandés, comment faire mieux ce qu'on fait actuellement d'une autre façon. C'est presque un manifeste culturel qui a fondé une école de pensée managériale en IT et c'est ça que j'aime bien. Je trouve très dommage que notre industrie est complètement dérivée d'un point de vue managériale quand je vois des certifications Scrum Master, excuse-moi, mais je vais dire chez nous personne n'est certifié quoi que ce soit parce que pour moi c'est une hérésie. Engager des personnes qui ont étudié 6 mois des bouquins, pour être certifié bidule. Pour moi, c'est de l'hérésie parce que justement c'est l'antithèse d'Agile. La seule façon de se former en Agile c'est de faire des projets en Agiles et de voir qu'est-ce qui marche, qu'est ce qui marche pas, voir aussi avec les équipes tu peux prendre une approche qui marche très bien avec une équipe qui ne marche pas avec une autre, ou la même équipe qui marche pas du tout avec un autre client. C'est aussi très humain, comme toutes les méthodes de management ce n'est pas une science exacte. C'est une science empirique. Qui, voilà en fonction des guidelines, fonctionne plus ou moins bien pour certaines situations et il faut les voir comme ça.

*Dans l'entreprise tout le monde arrive à avoir cet esprit Agile ?*

M.H. : Certains sont plus à l'aise, certains vont l'être moins et il faut justement jongler avec ça. Il faut aussi prendre chacun comme il est, il faut aussi donner les grands principes, s'assurer aussi que ces grands principes ne sont pas violés. Par exemple, pour moi ce qui est assez important, mais qui n'est pas assez mis en avant, c'est l'obligation to descend. C'est dans un processus Agile, la première chose que tu dois t'assurer, c'est que s'il y a un problème, c'est que ce problème soit rapidement remonté, mis sur la table et discuté. Pour

moi c'est un des fondements de l'agilité, c'est identifier quand tu as des problèmes. Quand on me dit qu'il a des objectifs bien définis, bien clairs, et que tout le monde sait ce qu'il a à faire, et bien ça c'est la théorie. En pratique il y aura une tuile, il y aura un problème quelque part, et toute l'agilité c'est de gérer le grain de sable. Si tu me dis que tout est coulé dans la marbre, ça je te dis c'est sûr ça va se planter. Après certains sont plus sensibles que d'autres. Certains aiment mieux d'autres moins, pourquoi ? Parce que certains aiment beaucoup échanger discuter, changer les priorités et puis il y en a d'autres qui aiment bien avoir un plan préétabli prêt pour la semaine. On essaye un peu de jongler avec ça.

*Si maintenant vous aviez une définition à donner sur l'agilité ce serait quoi ?*

M.H. : Pour moi, l'agilité c'est ce que je t'ai dit, c'est avant tout de faire du changement une opportunité, c'est la base de l'agilité. C'est un processus de management qui permet de faire du changement une opportunité. Et ça ne veut pas dire que d'autres processus managérial sont faux ou mauvais, c'est juste une façon différente de voir les choses. Le changement, pour nous c'est une opportunité presque commerciale, pour nous on offre des projets, on est les seuls à rendre, parce que plein de boîtes refusent d'essayer d'attaquer le problème de façon Agile. Et beaucoup de boîtes, quand bien même elles se diraient Agiles, restent dans une logique waterfall. Par exemple, tout le monde se dit Agile dans notre domaine, et tout le monde fait des appels d'offres et des remises d'offre. Les remises d'offres c'est intrinsèquement non Agile. Une remise d'offre c'est pour que tu puisses budgétiser un projet ce qui présuppose que tu sais scoper un projet complètement, ce qui est l'antithèse d'Agile. Tu as quelque part une contradiction dans ce qu'on dit et dans ce qu'on fait. C'est un grand écart assez intéressant et donc là nous on essaye de dire c'est qu'on va scoper dans un projet, mais le client va dire : « Ce n'est pas du tout ça que je veux, je veux ça en fait oublier tout ce que vous avez fait avant, il faut faire ça parce que mes clients viennent de me dire que... » Je pense qu'on est capable de le faire raisonnablement. Et en tout cas on ne va pas hurler que ce n'est pas dans le contrat. Et donc ça nous donne un avantage compétitif.

*Et si je vous demandais une définition des méthodes Agiles est-ce qu'elle serait différente ?*

M.H. : Non, l'objectif pour moi c'est embrace change, faire du changement une opportunité, reste l'objectif philosophique finale. Après elles ont toutes en commun certains traits, elles ont toutes la notion cyclique, peu importe si c'est un cycle court, continu ou long mais c'est la notion de cyclicité. Il y a la notion de priorisation qui est importante, une notion de vitesse,

une notion de combien on peut faire par cycle. Toutes ces notions sont importantes que tu vas trouver dans toutes les méthodologies Agiles, qu'elles soient plus ou moins marquées et aussi des rôles de Scrum Master, des rôles de chef d'orchestre tout ça tu les retrouves. Mais ça ce sont plutôt des traits communs. Pour une définition, toute méthodologie qui est désignée pour créer du changement une opportunité est essentiellement Agile.

### Appendix 13 : Interview Rudi Serron

R.S. : CEO at P&V company

*Est-ce que vous pouvez vous présenter en quelques mots ainsi que votre société s'il vous plait ?*

R.S. : Oui en quelques mots. Je suis Rudi Serron et je travaille déjà depuis longtemps chez P&V. Il y a plus de 30 ans,. À cette époque-là, j'ai parcouru l'IT j'ai fait tout ce qui était possible à l'IT. Et maintenant, depuis un an, je suis le CEO du Groupe P&V. depuis 11 ans je fait partie du management IT et avant ma fonction de CIO j'étais responsable architecture et la méthodology de development et gestion de projets.. Et avec l'équipe, on a implémenté, pas l'Agile, mais tout ce qu'on appelle l'hybride Agile. C'est pour cela que j'ai accepté l'interview pour vous donner nos expériences et pourquoi on l'a fait.

P&V groupe assurances, on est le 6ème assureur en Belgique. C'est la seule entreprise d'assurances qui est encore belge. On a plusieurs canaux de vente : Vivium tout ce qui est courtiers, P&V qui travaille avec des agents indépendants (tied agents), - ils ne peuvent vendre que des contrats P&V - et une partie directe qui s'appelle Actel. On fait des produits vie comme du non vie, i des prêts hypothécaires et du conseil. Je suis responsable pour les départements IT et pour toutes les branches on délivre les applications et les infrastructures.

C'est un peu en court la présentation de P&V, du département IT et de moi.

*Quelle est l'organisation de votre département IT ?*

R.S. : Au niveau organisation, il y a plusieurs départements. Il y a un département qui s'appelle « solution delivery » c'est tout ce qui est nouveau projet. Donc ce sont des nouveaux fonctionnalités ou modifications d'applications pour que le business peut réaliser ces ambitions. . Et puis, il y a un département qui est « service delivery » qui a avoir avec tout ce qui est le Run opérationelle, c'est keeping de lights on, c'est le maintien des applications, petites améliorations au applications et aussi la partie infrastructure hardware et operating systems ainsi que les aspect networking et information security..

Et puis, il y a une cellule qui s'appelle « project management et suivi budgétaire » qui est une troisième équipe.

Et puis en staff, il y a tout ce qui est « security officer et IT architecture lead.

*Avez-vous structuré votre organisation ?*

R.S. : Au total 260 FTE et on est réparti, parce qu'en fonction de la demande on fait up staffing, down staffing, 45% sont des externes donc c'est du staffing temporaire en fonction du nombre de projets qu'il faut réaliser.

*Quelle est la mission de l'IT ?*

R.S. : c'est faciliter et réaliser le change pour que le business peut réaliser ces ambitions.. keeping lights on, à des coûts raisonnables, et i être le partenaire préféré du business pour que le business puisse réaliser ses objectifs.

Pour avoir un TCO sous contrôle, il y a une gouvernance stricte, chaque euro que l'on dépense, il faut l'attribuer à quelque chose. Donc il faut du business value, une stabilité et il faut être Agile dans nos développements. Donc cela revient à dire être flexible et savoir changer en fonction du marché. Si le marché évolue, et bien on doit être capable aussi de changer très très très vite.

*Et cela fonctionne chez vous de changer si vite ?*

R.S. : C'est pour cela que l'on a appliqué le Hybride Agile methodology.

Avant C'était Waterfall. C'était vraiment rigoureux, donc quand l'étude est terminée, faire l'analyse fonctionnelle et puis il y avait des développements qui démarraient et à la fin il y avait une phase d'acceptation.

Sur des projets avec une ampleur relativement grande - c'est-à-dire qu'au début l'IT a des contacts avec son business pour capter tous les requirements. En phase d'analyse fonctionnelle, il a encore des contacts. Et puis l'IT était partie pendant des mois, années pour faire le développement et puis ils revenaient avec une solution, mais entre-temps les requirements avaient changés. Et donc la solution ne répondait plus avec leur nouvelles attente, mais avec ce qu'ils avaient demandé au début. Alors il fallait faire du rework à la fin cela coûte très cher parce qu'il faut faire des choses qui remontent au début du cycle de developement sont au début et puis cela crée l'impression sur les plus grands projets que tu ne finis jamais. En plus de l'augmentation du budget, pas de satisfaction du côté business parce que, pour eux, ce n'était pas cela qu'ils avaient demandés ou qu'ils voulaient à ce moment-là. Donc, à ce moment-là, il y a un misalignment entre le business et l'IT au niveau de l'organisation. L'IT coûte trop cher et on ne livre pas ce qu'on veut. Et pourquoi ? Deux raisons : les idées, requirements, changent entre-temps et l'IT n'était pas informé, et ça donnait l'impression que c'était l'IT n'étaient pas a l'écoute du business et que les

changements n'allaient pas assez vite. C'est à ce moment que l'on a décidé qu'il fallait changer, il fallait avoir une autre approche. Donc on n'a pas changé uniquement au niveau de la méthodologie de développement, mais aussi au niveau de la gouvernance. On a créé une transparence à chaque euro à quoi il était attribué. Chaque dépense, il fallait la justifier. Il y a 5 euro sur les 100 qui vont vers le run,, les 95 autres c'est ça, ça, ça qu'on doit faire. Et puis, il y a un reporting mensuel là-dessus et un suivi en toute transparence. S'il y avait cinq personnes qui travaillaient sur un projet, qu'est-ce qu'ils ont fait pendant une période- et comment ils avaient avancé en fonction du planning établi. Cette transparence, on a dû la créer. On a remis le business dans le lead pour définir toutes les priorités.. Et puis troisième phase, comment peut-on maintenant créer plus vite la valeur pour le business ? Et c'est là que l'on a appliqué le modèle hybride Agile. Les projets, on va les couper en petits morceaux et on va livrer de la business value tous les 3 semaines. Donc on démarre quelque chose, aujourd'hui on travaille comme cela, après une phase de pré-étude pour avoir un scope déterminé pour être certain : « les équipes de développement sont occupés pendant 2 ou 3 sprints .

Le role du product owner, qui est l'expert et qui déterminait les priorités des fonctionnalités qu'il faut livrer. Et à lui de déterminer, c'est quoi les fonctionnalités avec les plus hautes valeurs business qu'il faut développer, si c'est possible, en premier lieu. Et en fonction de cela, le projet avance. On démarre toujours maintenant avec un backlog avec des requirements qui sont fixes et pour un certain budget qui est estimé. Le product owner gère le user stories pour rester dans le budget. Des nouvelles requirement sont ajouter en supprimant d'autres pour rester dans la m^me enveloppe. En mode waterfall ce n'est pas possible, car le scope grandit uniquement.

Et ça, c'est l'agilité que l'on a créé. Parce que parfois, il faut de la flexibilité. Tu es dans un sens, le développement, puis le contexte change et il faut savoir aller dans l'autre sens et ne pas dire : « Non on fait le projet et on recommence ». Ça, on a obtenu en allant travailler vers un mode plus Agile.--> , on crée de la transparence. Après chaque sprint, tu vois ce qui sera livré et si cela convient. Deux, il a une vue sur les dépenses, son budget il peut le dépenser comme il veut et déterminer ce qu'on fera. Trois, dès qu'il a un ensemble qui est prêt pour aller en production, qu'on le met en production parce que cela doit être prêt pour mise en production après le sprint et cela gagne déjà de la valeur.

Et quand tout est à terme, on met en production et avant c'était big bang, c'était tout ou rien. Et ici, ce qu'on voit aussi, avant ils demandaient un budget pour un million et maintenant, ce

n'est pas souvent, mais on arrive aussi après une dépense de 80% de dire non c'est suffisant maintenant. Le reste je ne vais plus faire. Mais il est en production et cela crée de la valeur. Et ça c'est ce qu'on a travaillé en travaillant en Agile, dans un mode Agile pour le business.

Du côté IT, c'est la même chose aussi. Ce qu'on voyait dans les développements, les cycles, c'est un grand avantage c'est un projet classique cela donne l'impression qu'au début on a tout, tu travailles, tu travailles, mais c'est encore loin et puis tu vois le stress monter à la fin et c'était vraiment des pics et c'était difficile d'arriver.

Maintenant, c'est, on ne va pas dire qu'il n'y a pas de stress, mais c'est un peu du stress continu pendant 3 semaines, IT stress, mais tout ce qui est livré à ce stade, c'est oublié on peut commencer sur le reste et on n'a plus des pics à la fin.

On voit aussi maintenant avec le release management, qu'on a impliqué là-dedans, avant c'était toujours le dernier où on va y arriver, maintenant déjà 2 semaines en avant, c'est calme. On a une mise en production, il n'y a plus de stress, les gens partent à l'heure normale et tout cela parce que c'est autrement. On sait à l'avance où on est et le plus essentiel est déjà livré en production ce qui est important pour le business. Donc là, il y a eu un grand changement aussi pour travailler en Agile.

*Est-ce que c'était facile d'implémenter ce changement ?*

R.S. : Est-ce que c'était facile, non, parce qu'on crée de la transparence, mais c'est une transparence journalière et il y a une sorte de contrôle social qui se crée dans l'équipe. Parce que c'est une équipe qui, en travaillant comme cela, est responsable pour tout et le jour où il y en a un qui a donné un accord qu'il allait faire cette activité-là pour demain soir. Si ce n'est pas fait, c'est l'équipe qui est bloquée et c'est l'équipe qui doit l'aider pour y arriver et tout cela. Et si c'est chaque fois la même personne et bien c'est difficile. Avant c'était plutôt caché dans le gros volume, s'il y en avait un qui avançait moins vite, et bien, c'était moins remarqué. Ici c'est après 2 jours que l'on voit que la vélocité de l'équipe n'est pas là et cela traîne chez lui. Qu'est ce qu'il faut faire pour l'aider ? Si cela revient toujours sur la même personne, à la fin cette personne-là ne se sent pas bien et c'est difficile pour elle.

Moi j'appelle cela du contrôle social qui se crée, c'est plus facile pour le management donc cela se crée dans l'équipe. Et c'est l'équipe qui est responsable et cela met de la pression sur les gens aussi. Et pour certains, c'est une pression continue et ça, c'est difficile. Là, le change n'est pas facile. Pour le management, c'est plus facile. Tu as les dashboards, le Jira et tout

cela. Je ne dois plus demander au chef de projet : « Tu en es où ? », j'ouvre le dashboard. Ah cela avance bien et j'ai une vue journalière sur l'avancement des projets. Donc c'est du trust qu'il faut donner aux équipes. Ça, c'est un changement aussi. Et parfois, il faut intervenir pour les aider. Et tout cela, c'est un autre style de management aussi qu'on doit appliquer maintenant. Parce qu'on travaille dans ce mode-là, Agile, et on donne plus de confiance. Avant on donnait de la confiance aussi, mais c'est plus de responsabilités. Par rapport à avant, dans les équipes c'était plutôt non à la responsabilité, c'est plutôt par le project manager. Et maintenant, s'il y a des soucis c'est pour lui. Et ça on remarque aussi sur le terrain, qu'il y a ce sens de responsabilité qui est prise par les équipes parce qu'ils se sentent responsables avec le business et le résultat.

Le rôle de product owner est crucial et c'est le plus difficile de mettre en place. Prendre des décisions ce n'est pas donné à tout le monde.

Et avant, c'est aussi une leçon apprise, c'était business-contrat-IT. Si les projets n'étaient pas une réussite, IT disait « Oui, mais c'est les requirements qui n'étaient pas bons ». Et business disait : « Oui, mais IT, c'est toujours trop tard ». Maintenant, c'est un ensemble et ils sont ensemble responsables du résultat et cela marche beaucoup mieux.

*C'était cela la grande motivation d'implémenter hybride Agile ? Pour avoir cette cohésion le business et IT ensemble ? Ou y avait-il une autre motivation de départ ?*

R.S. : C'est parce qu'il y avait des soucis. Les soucis c'étaient : il fallait améliorer et c'était la seule chose comment on pouvait améliorer les choses. À ce moment-là, on n'avait pas d'autres possibilités. C'est aussi après une étude, on a dit : « Ok, qu'est-ce qu'on peut faire ? ». Et après cette étude-là, on a dit Agile way of working c'est le next step qu'on devrait prendre. Et puis aussi on ne l'a pas fait directement en full Agile. On n'a fait un pas dans cette direction-là, mais en gardant quand même encore une pré-étude pour voir dans quelle direction on allait. Une fois que la partie architecture est bien finie, comment on va le faire ? Une fois que ces questions-là étaient répondues, on commençait, pour éviter à la fin du rework, qu'on avait oublié quelque chose. Ce n'est pas qu'on prend quelque chose, on commence et après 3 semaines on dit : « Ah non, ce n'est pas ça, on fait autrement ». Là, on n'y est pas encore. Full Agile, c'est dans nos plans, mais il n'est pas appliqué aujourd'hui. C'est quelque chose que l'on met maintenant en place pour démarrer en 2019. Si c'est quelque chose qui est crucial, qui fait partie des fondations de l'IT, c'est encore de l'hybride Agile. Mais si c'est quelque chose pour des choses innovantes, ou pour des choses qui sont en

direct vers les clients, et bien là on va appliquer du full Agile. Et cette -étude, on ne va pas l'utiliser. Il y a une idée et on se met ensemble avec une équipe et on dit on fait le développement et on voit après 4 semaines quels sont les résultats. Et on continue dessus pour les faire évoluer. Et si après 6 semaines, on dit ce n'était pas cela, à ce moment-là, c'est aussi un changement de mentalité. Et même au niveau du board, on va avoir le courage de dire on arrête et on recommence de scratch, ou soit on abandonne carrément l'idée.

*Et en ce qui concerne votre implémentation, vous avez parlé du fait que vous alliez vers le full Agile, pouvez-vous m'en dire davantage ?*

C'est un change que l'on va faire sur 1 an et demi, 2 ans avant qu'il soit implémenté. Mais jamais pour la totalité. On restera toujours hybride Agile et certains projets, on fera en mode Agile full.

L'hybride Agile, il faut bien expliquer : on fait une pré-étude, on passe dans une phase de développement et il y a encore une petite phase de validation end-to-end avant la mise en production. La validation end-to-end c'est quand-même un minimum viable product un ensemble de données.

L'Agile, c'est vraiment on développe quelque chose et on met en production et ça c'est dans nos cartons pour l'année prochaine.

*Et vous utilisez quelle méthode Agile pour l'instant ?*

R.S. : Ça dépend. Donc il y a Scrum et il y a Kanban qui sont utilisés aussi. Kanban, donc on fait ça pour les projets et tout ce qui est service management. Service management ce sont des petits changements-là c'est plutôt du Kanban. Il y a un backlog et on prend des tickets et on essaye de finir le plus vite possible. Important éviter le travail en parallèle. Une activité à la fois par personne

Et puis, il y a aussi une méthode et c'est pour des plus gros projets, c'est du Scrum. Avec un dashboard et des dépendances et tout cela. Et il y a une certaine séquence qui est définie pour les exécutions. Les deux sont appliquées.

*Et est-ce que c'est possible que pour un certain projet vous commenciez par utiliser Scrum, et que par la suite vous avez changé vers une autre méthode ?*

R.S. : Non, pendant un projet on ne change pas. C'est aussi aujourd'hui certaines équipes qui ont l'habitude de travailler en scrum et d'autres en Kanban et le mixte on n'a pas encore

appliqué. On a déjà fait la différence, mais au du début projet, quand on travaille avec un intégrateur, eux préfèrent travailler en Kanban et l'autre en scrum, donc on suit un peu l'intégrateur. Mais c'est toujours au début du projet qu'on a fixé la technologie que l'on allait prendre. On n'a jamais changé au sein du projet parce qu'un projet ça doit aller vite et faire des changements ça aurait arrêté tout cela et on essaye d'éviter.

*Et quand avez-vous commencé à implémenter l'hybride Agile, vous avez fait cela comment ? Tout d'un coup ou par étapes ?*

R.S. : D'abord il y avait une étude et il y avait la proposition que l'on avait faite. Il y a eu l'acceptation du board qui nous a donné la permission de se lancer dedans. Et puis, il faut préparer les changements. Il faut commencer avec une indication et tout cela. On a appliqué une méthode, « la tâche d'huile » : C'est une tâche d'huile sur de l'eau, c'est petit et cela se répand. Et c'est un petit peu ce qu'on a appliqué aussi. On a commencé, on a donné la formation à une certaine équipe, puis ils ont bossé et on a regardé ce qu'il faut améliorer. Et en fonction de cela, ceux qui étaient dans les équipes, ils se sont scindés et comme cela on a grandi sur tout le département. On a grandi comme cela jusqu'à tout le département était sous hybride Agile et cela a pris un peu plus qu'un an avant que tout le département fût sous cette méthodologie-là. Et les gens qui viennent et les nouveaux et tout cela, ils s'intègrent dans cette équipe là et on avait aussi une équipe des Scrum coach et aussi au niveau analyse et architecture. Donc il y avait 4 coaches qui étaient là dans les départements qui aidaient les équipes et qui allaient voir dans les équipes pour améliorer les méthodes de travail. C'est aussi un peu ce continuous improvement qu'on appelle cela, les lessons learned, comment on peut améliorer, et le feed-back vers les autres équipes. Donc il y avait aussi une équipe sur le terrain de coach qui était en permanence là pour aider, pour donner des conseils. Et parfois aussi des équipes où cela n'allait pas, et il fallait prendre la main sur l'équipe et travailler avec eux pour les mettre sur les rails. Donc cela a quand même pris un an et demi avant que tout était mis en œuvre.

Et donc c'est le développement qui le fait, mais tout ce qui est infrastructure c'est encore la méthode classique, ce n'est pas du Kanban qui est appliqué, mais c'est comme un Kanban. Ils reçoivent un ticket et ils doivent les mettre en œuvre, mais ce n'est pas des stand-ups et tout cela, ils n'appliquent pas cela. Tous cela était accompagné par un programme de changement qui avait comme scope : communiquer, informer, parler et être à l'écoute des collaborateurs pour bien expliquer le pourquoi du changement, l'avancement et de capter en acte sur le feed-back des collaborateurs

*Et avez-vous dû rediriger certaines personnes dans certaines fonctions, je parle par exemple du product owner, du scrum master ? Comment avez-vous déterminé ces personnes-là ?*

R.S. : Donc product owner, ça c'est du côté du business et ça c'est aujourd'hui toujours compliqué, car jusque là c'est le product owner qui est le cœur de chaque équipe. C'est en fonction de son autorité que dépend aussi l'efficacité de l'équipe. C'est le cœur, c'est lui qui est le chef d'orchestre et qui doit bien savoir prendre des décisions quand il faut d'une manière indépendante. Et le Scrum master c'est le follow-up. Donc c'est plutôt l'aspect fonctionnel qu'il doit maîtriser, il doit maîtriser que toutes les activités présentes-là sont réalisées et organiser son équipe et aussi regarder s'il y a des impediments, les enlever pour que l'équipe puisse travailler d'une manière efficace. L'équipe, c'est une équipe homogène, mais indépendante qu'il faut créer cela aussi. Donc c'est important au niveau du product owner, au niveau du scrum master c'est important aussi, mais l'équipe aussi, il doit y avoir une bonne cohésion et tous les profils doivent être là. Il faut qu'une équipe arrive à travailler d'une manière indépendante parce que dès que tu as des dépendances avec des autres équipes tout tombe. L'équipe en soi c'est très important, mais le product owner c'est crucial au niveau prise de décision, autorité et savoir dire ce que l'équipe doit faire, et le scrum master c'est plutôt le follow-up de l'équipe. Ces personnes-là, product owner, on en cherche toujours dans certains domaines, parce qu'ils doivent connaître le business, mais pas uniquement le business le connaître, mais aussi comprendre la stratégie du business pour bien déterminer vers où on veut aller. Ce n'est pas facile cette prise de responsabilité, le ownership des priorités, ce n'est pas donné à tout le monde. Parce que les gens sont dans une certaine organisation qui était plutôt chez P&V d'une manière hiérarchique. Donc c'est le chef qui disait ce que l'autre devait faire et cette mentalité-ci, elle existe encore. Elle va à l'encontre du travail en Agile. Normalement, c'est l'équipe qui peut prendre leur responsabilité et eux doivent sentir que le management leur donne la confiance de s'organiser de telle manière. Et ça, ce n'est pas facile. Donc cela, c'est le plus compliqué. Et là, on voit encore que trop souvent cela dépend du manager qui doit intervenir sur des choses comme cela. Scrum master, en interne, on fait de l'up-staffing et c'est plutôt des profils comme cela, des experts que l'on a cherché sur le marché pour nous aider là-dedans. Donc c'est eux qui ont commencé avec les équipes et dans l'équipe, il y avait quelqu'un en interne qui avait dit : « Ah ça, cela m'intéresse de le faire aussi ».

Il a suivi les formations, il a reçu du coaching par les experts qui étaient là. Quand ils avaient assez de maturité, c'est l'expert qui part et c'est quelqu'un de l'interne qui prend. C'est de cette manière que l'on essaie de travailler, mais ça c'est pour la partie de Scrum master que l'on peut le faire. Mais un product owner, il doit connaître la boîte et il doit savoir vers où on va et on ne trouve pas si facilement à l'extérieur donc là c'est un peu complexe. Et au niveau du développeur, testeur, analyste et tout cela, c'est la méthodologie qui change, mais leurs compétences étaient là et il y a peu de changement. Sauf la partie dashboard qui était affichée tous les jours et s'il y en a un qui ne livre pas, il retrouve affiché quelque chose. Voilà, ça ne bouge pas et il doit s'expliquer, c'est ça qui est un peu difficile. Je ne sais pas par rapport aux autres, mais nous on constate que le rôle de PO c'est le plus difficile. Le staffer et aussi trouver les gens qui veulent le faire et prendre des décisions ce n'est pas facile. Et ça on demande aux gens maintenant et on voit que ce n'est pas évident.

*Aviez-vous des attentes particulières sur l'évolution avant la mise en place ? Aviez déjà des mesures, des indications de ce qu'allait donner l'implémentation ?*

R.S. : Oui, quand tu veux mettre quelque chose en place, il faut qu'il y ait un business case. Ce n'était pas moi qui l'avais mis en place. C'était Johan Dekens à l'époque qui était à ce moment-là le CIO, mais comme je dis, j'étais responsable pour tout ce qui était méthodologie et architecture, donc j'ai collaboré à ce principe-là. Donc quand il faut faire un changement, tu as besoin d'un budget et pour avoir un but comme cela il faut avoir un business case et en fonction du business case, il y a aussi un KPI qui est défini. Des KPI qui sont fixés sur la qualité du travail livré, sur le business value qui est livré donc il faut calculer que pour développer une fonctionnalité X il fallait autant d'euro avant qu'on avait du business value tu peux comparer comment c'était avant par rapport à ce qu'on espérait y arriver. Donc qualité, business value à livrer, et puis aussi la satisfaction du business. Donc c'est une baseline que l'on avait avant. Et puis à un certain moment, tu peux interroger à nouveau, on se trouve où maintenant ? Donc ça c'était un peu les keys success factors qui étaient définis à ce moment-là et par lesquels on a su mesurer. Maintenant, est-ce qu'on les a obtenus ? Oui, mais plus tard que ce qui était prévu au départ. Donc, on a pris du retard sur nos planning parce que le changement ce n'est pas uniquement du côté IT. C'est toute la compagnie. Donc c'est aussi le business qui doit se mettre dans ce mind-set. Un change ce n'est pas si facile que l'on pense. On pense toujours oui cela va aller, mais ce n'est pas facile pour les gens de s'impliquer ou de se mettre dans un autre contexte et c'est un peu sous-estimé. Mais à la fin, là où on est maintenant, on a réussi. Cela a pris plus de temps et c'est aussi la motivation pourquoi on veut

mettre maintenant le pas suivant vers du full Agile où c'est applicable. Il y a le « believe » comme on dit maintenant dans la nouvelle méthodologie, et pour nous on veut plus évoluer dans ce sens-là et on veut encore être plus Agile et délivrer dans certains domaines encore plus vite qu'on ne le fait aujourd'hui. C'est cela qui est la key aussi c'est le time-to-market qui est très important pour nous. S'il y a des changements qui sont nécessaires, il faut savoir livrer la valeur business le plus vite possible, mais contrôler, et cela doit fonctionner. Ce n'est pas à n'importe quel prix, c'est avec une qualité qui est demandée pour ce que cela puisse fonctionner.

*Par rapport au projet, comment déterminez-vous qu'un projet est un succès ?*

R.S. : C'est on time, on budget and client satisfaction.

*Avez-vous vu une grande différence entre quand vous suiviez le modèle Waterfall et maintenant en Agile ?*

R.S. : Oui

Et quelle était cette différence ?

R.S. : Oui parce qu'avant, c'était rare qu'un projet était à temps, dans le budget et que le client était satisfait. Pas pour les petits projets, mais un projet de 2 mois quelque chose comme cela on ne compare pas avec celui-là donc. Ce sont des projets dont on parle entre 500.000 € et satisfaction du business est plus haute, car ils sont avec nous au commandement de la réalisation et ils sont d'accord et content du produit qui est livré.

*Avez-vous découvert des avantages au fur et à mesure que vous avez implémenté Agile ?*

R.S. : Oui, des avantages, c'est au niveau des gens au niveau de la maturité des gens, on voit même certaines capacités que l'on n'a jamais vu chez des gens qui commencent à remonter. Et on voit des gens qui peuvent faire plus qu'avant, mais c'était moins visible. Et les gens prennent des initiatives et qui deviennent des moteurs alors qu'avant ils étaient un peu plus dans la masse. Et maintenant tu vois, chaque équipe est importante maintenant. Et ça se voit et il y a aussi plus de solidarité dans les équipes, et tout cela. Et ils s'aident plus qu'avant, avant c'était : « Mais moi je suis testeur ». Non, maintenant tout le monde est responsable dans l'équipe et cet esprit-là ça se voit beaucoup plus parce que les équipes sont plus petites. Et toutes les trois semaines, il y a une deadline en fait. C'est le niveau de stress au niveau de déploiement en production qui a fort diminué.

*Quel a été l'impact en général de l'implémentation dans l'entreprise ?*

R.S. : Dans la vue business for IT alors. IT est reconsidéré comme un partenaire du business, il est là pour le business et on arrive à livrer. On a maintenant la capacité de livrer. Avant il y avait des doutes. Avant c'était un alignement quand on faisait le budget et les plans et puis il y avait le désalignement parce qu'on ne se comprenait plus ou on avait mal compris. Maintenant, il y a cet alignement qui donne une transparence totale. Et parce qu'on utilise la méthodologie Agile, c'est visible pour tout le monde. Avant c'était un message donné par un chef de projet, maintenant, c'est tous les jours et tout le monde l'a. Donc moi je l'ai dans ma fonction. Mes sponsors, ils ont le même dashboard, ils voient tous les jours les projets qui les intéressent. Et donc, parce qu'il y a transparence, il y a aussi alignement et ça c'est un grand changement. Et c'est la crédibilité des équipes de projets qui a augmenté. Si on dit maintenant, commitment on va le faire, ils sont confiant qu'on va le faire, qu'on peut le faire. Et si on revient avec quelque chose, en interne, il y a quelque chose qui arrive encore on a aussi la volonté de trouver une solution. Avant c'était : « Mais c'est la faute de ... ». Maintenant on cherche des solutions ensemble. Et ça c'est pour moi un impact ou un changement que la manière de travailler en Agile a provoqué dans la société.

Pour nous c'est très important.

*Je voudrais revenir sur le fait que vous aviez fait une étude pour voir quelle méthode utiliser, comment s'est passée cette étude ?*

R.S. : Oui, donc c'est la société AE, c'est une société de Haasrode qui est spécialisé dans tout ce qui est architecture et méthodologie, qui a proposé la méthodologie Agile. En fonction de cela, on a contacté plusieurs consultants, Jef Cums qui, - à cette époque-là était fort reconnu pour tout ce qui est éducation, méthodologie et tout cela en Agile - est venu faire des workshops montrer comment cela fonctionne. On l'a fait aussi, pas uniquement en IT, mais aussi avec le business. Comment cela fonctionne ? Et bien expliquer ce sont quoi les concepts avec des jeux de rôle et tout cela pour montrer c'est quoi l'Agile et c'est quoi que ça pouvait apporter. Et on n'a pas comparé avec quelque chose d'autre. Avant c'était Waterfall, waterfall c'est terminé on doit changer maintenant. Donc ces principes-là on a quand même fait approved on technology ou approved on concept, mais sans des projets, rien qu'avec des jeux qui n'avaient rien à voir avec l'IT ou business et IT ensemble pour que tout le monde ait bien le mind-set. Qu'est-ce qu'on va faire ? Qu'est-ce que cela implique ? Comment on va s'organiser ? Une fois qu'on avait ces principes-là et un alignement là-dessus, on a commencé

le change avec la formation et tout cela et cette société là aussi, mais avec des consultants en plus qui ne sont pas d'eux. On a créé une équipe de méthodo, des formateurs et des coachs et en fonction de cela on a démarré. Ce qui était très important, c'était qu'il y avait un alignement avec le business. Donc on n'a pas fait en IT et puis on est venu, on fait tout de suite cet alignement-là et ils étaient aussi conscient qu'il fallait faire cela. Beaucoup de communication aussi.

Au début, c'est dire qu'est-ce qu'on va communiquer qu'est-ce qu'on va faire avec des plannings et puis aussi régulièrement donner du feed-back sur l'avancement. Et donc, ne pas lâcher sinon c'est parti comme Waterfall, on annonce et puis dans 2 ans on vient dire où on est. C'était chaque fois avec chaque directeur de business line, on fait cela, on fait cela, on a eu cela comme difficulté, et bien c'est ça et comment on va corriger et ça c'était très important. Un changement comme celui-là avec le business, il faut bien préparer, tout n'est pas couvert dans une préparation, on décide là-dessus et il faut communiquer, communiquer, communiquer sur l'avancement même sur le problème. Etre ouvert et dire les choses comme elles sont. Si des projets n'avaient pas réussi, on a toujours communiqué en toute transparence, comment on allait redresser la situation et ce qui est important aussi dans un track comme cela il faut prévoir du budget. On appelle cela de la contingence.

Tu feras peut-être moins vite qu'avant, ce n'est pas le premier le projet, tu es en cours d'apprentissage et tu feras moins vite, mais il faut s'assurer que la prochaine fois tu iras plus vite pour l'obtenir. Là, il faut prévoir du budget pour qu'on ait cette marge-là. Si tu n'as pas ce budget là, mais c'est moins bien et donc il fallait bien préparer tout ce projet-là dans tous ses aspects là.

*Quelles ont été les grosses difficultés à l'implémentation d'hybride Agile ?*

R.S. : C'était se dire qu'il y a quand même des autres scrum team qui étaient impactés par des changements ou quand il y avait un lien avec un autre département ou un fournisseur externe. Parce qu'eux sont encore en Waterfall en Agile c'est plutôt j'ai mon backlog, je connais mes deux sprints et puis je peux changer là-dedans. Mais si tu changes quelque chose, il faut communiquer aux autres aussi parce qu'ils sont encore peut-être en Waterfall et ils disent : « Ah cette fonctionnalité-là, je sais que je dois la livrer pour dans x mois », mais si dans le Scrum on a changé les priorités, il faut peut-être que la fonctionnalité soit là plus tôt. Si on oublie toutes ces dépendances, à ce moment-là tu arrives à des conflits. C'est ça qui au début était difficile, chaque équipe ne regardait que son backlog et pas les dépendances envers les

autres. Et à ce moment-là, on a ajusté en créant des scrums of scrum que ces scrum masters parlent tout le temps. Ce sont quoi les changements ? Ce sont quoi les besoins ? J'ai besoin de ça dans ça est-ce que tu es prêt ? Il faut qu'ils puissent s'aligner et ça ce n'était pas facile. Parce que c'était plutôt focus, ce sont mes points, je veux livrer mes points et s'ils sont livrés alors c'est bon. Ils devraient avoir cette vision end-to-end et c'est cela qui est très difficile. Même chose aussi au niveau architecture c'est qu'on travaille d'une manière hybride Agile donc il y a une certaine solution design qui est faite et c'est comme cela qu'on veut faire et puis quand le projet est livré et bien ils n'ont pas fait et il y a des choses qui n'étaient pas prévues et qui ne fonctionnent pas. Et ce sont les product owner qui avait fait un court-circuit. Pour rester dans ce budget en disant : « Mais on peut le faire comme cela aussi » et ce n'est pas communiqué donc au niveau architecture je n'étais pas au courant. Et puis quand il fallait le mettre en production, c'était une boutade, il fallait avoir 10 serveurs qu'on avait prévu et qu'on avait installé et qu'ils avaient changé et ce n'était plus des serveurs, mais le mainframe qui fallait à sa place. Oui, mais le mainframe n'est pas là alors tu ne peux pas aller en production. Ça ce sont des choses qu'on n'avait pas pensé qu'on a dû redresser en mettant scrum of scrum. Les architectes vont maintenant aussi dans les démos en fait et ils vont demander : « montre-moi ta solution design, est-ce que cela correspond encore ? Comment c'était fait ? Et ce n'est pas à chaque fois, mais c'est de temps en temps, donc on fait des échantillons. Ils ne savent pas encore quand ils vont arriver, mais s'ils sont là et qu'ils ne sont pas conformes, ils doivent recommencer. Et ça on ne faisait pas. On appelle cela un peu du out skipping. Mais c'est aussi qu'au niveau architecture cela reste conforme, mais pas au début uniquement, mais pendant l'implémentation. On avait vu, tu donnes l'autonomie aux équipes fixes, on doit arriver et pour nous cela fonctionne et le reste on n'a pas oublié. Et cette architecture comme cela et que tout est passé au peigne fin sur les autres sprints et tous les aspects sécurité, ce sont des choses qui sont vite oubliées quand ils n'ont qu'un focus fonctionnalités. Ça, ce sont des petites fautes qu'on a dû ajuster. Et qu'est-ce qu'on a encore ? On donne un bon équilibre dans les équipes, pas tout le monde a la même maturité, la même productivité. Il ne faut pas que les écarts soient trop grands. C'est mieux que ton équipe a la même compétence, et que tu peux estimer au même rythme que tu as 10 qui vont très très vite que 5 qui vont très vite et les autres, ils vont être frustrés parce que cela n'avance pas et les autres vont être frustrés parce que les autres vont trop vite.

Il faut trouver le bon équilibre dans la composition des équipes. Le project manager son rôle a changé. Il devrait moins s'occuper de la partie suivi fonctionnel parce qu'il est repris par le

scrum master, avant c'était le Project manager qui devait le faire, et il devait gérer cela un niveau plus haut; plus high level et couvrir tout ce qui est à voir avec le projet, ce sont les points les plus essentiels.

*Avez-vous fait des adaptations par rapport à ce que vous avez mis en place au début ?  
Est-ce que vous avez changé certaines choses ?*

R.S. : Oui, au début, on est parti sans outils, on faisait un tableau avec des post-it donc papier et tout cela. On avait un whiteboard avec des post-it. Et c'est bon quand tu démarres avec des petites équipes pour essayer tout cela. La même chose aussi, les testings étaient faits avec des Excel. Au niveau des requirements c'était aussi fait au niveau d'un Excel. Dans tout ce qui est built process, il y a peu d'automatisation et en fonction de travailler en scrum. Et tout cela, c'est à la fin qu'il y a des rétrospectives, il y a des remontées qui viennent des équipes qui disaient : « Ah mais ce serait quand même bien si on avait quelque chose qui reste tout le temps ». On avait déjà utilisé le dashboard à ce moment-là, on a acheté des outils comme Jira avec des tâches que tout le monde peut voir et on projette. On a des smart poort au lieu d'avoir des trucs de post-it papier. Et tout cela, et bien on a projeté le scrum. Chaque team a son propre dashboard pour faire le suivi et se déroule. Et ça c'est l'outil qu'on a mis en place. Et la même chose pour le test management, les test cases et tout ce qui peut, c'est déjà utilisé maintenant. Il y a des dashboard qui sortent automatiquement qui rendent plus efficace le travail. Tout ce qui requirements, il y a aussi un outil maintenant où on arrive à les documenter et ça c'est d'office chargé en Jira qui est l'outil de développement au niveau des users stories. Donc on a automatisé cette partie-là. Et puis ce qu'on a fait aussi c'est le built process cela veut dire que c'est développer, il faut en créer un exécutable. Avant c'était manuel et c'était chaque développeur qui devait le faire, maintenant on a créé aussi un processus automatique. Donc il est fait toutes les heures, pour qu'ils ne doivent plus s'en occuper. Et aussi au niveau de tout ce qui est testing on a automatisé avec des outils certaines choses qu'avant on faisait manuellement. Donc, il y a des activités répétitives, manuelles que par après on a remplacées par des outils, mais c'est venu comme input de rétrospectives. Ça, c'est important aussi l'équipe en Agile réfléchi comment il faut faire pour améliorer et qu'après cela suit. Parce qu'autrement, si en tant que management tu n'as pas d'écoute là-dessus, on veut s'améliorer, mais il n'y a pas de fundings et le management ne veut pas, alors ça ne va pas. Et on a choisi vraiment cette approche-là parce que à ce moment-là, d'abord le processus, on prouve que cela fonctionne et puis les outils nécessaires vont suivre sinon tu

achètes plein d'outils et puis ils ne seront pas utilisés. Ici, c'était vraiment les équipes qui étaient demandeurs d'avoir certains outils, on a même changé, car on est passé dans Jira.

Mais au niveau méthodologie même on a peu changé. Donc on n'a pas dit est-ce que c'est Kanban ou scrum, donc là on en a discuté au début c'est le choix qui était fait donc de ce côté-là, on a peu changé.

*Avez-vous laissé la méthode telle qu'elle est ? Ou n'avez-vous pas fait de modification dans la méthode telle qu'elle est théoriquement ?*

R.S. : Si, il y a toujours des petites modifications, mais pas pour dire quelque chose de fondamental. C'étaient des mises-à-jour ou améliorations pour être plus efficace. L'estimation des points et tout cela là, on a travaillé on a utilisé d'autres méthodologies tout ça pour voir qu'on pouvait être plus précis. Parce que cela reste encore difficile. Pour trouver de bonnes estimations pour le backlog et tout cela. Et là, les changements étaient plutôt des améliorations. Comment peut-on être plus précis ? Et là, on a travaillé. C'est encore parfois un try-out que l'on fait, on va une fois essayer avec cela et qu'est-ce qu'on fait par rapport à l'autre, on utilise une autre méthodologie. Le principe on a très peu changé parce qu'on a changé à cette époque-là toutes les méthodologies. Donc requirements, méthodologies, les analystes, les développements tout cela faisait partie du démarrage en Agile. Donc, avec toutes les qualités et tout cela, mais c'est un ensemble. Et à ce moment-là, c'est peut-être aussi pour cela qu'on a un peu changé par après parce que c'était beaucoup aussi pour les équipes. Il y avait beaucoup de changements en même temps et il ne faut pas éviter les changements parce qu'à un certain moment il faut avoir une certaine continuité et on a choisi pour faire des petits improvements à chaque fois sur base de ce qui existe parce que les équipes étaient confiantes et nous aussi que la méthodologie était la bonne. Un changement, c'est tu fais un pas en arrière pour peut-être faire 2 en avant pour continuer et puis c'est stop. Maintenant on change tout de suite et puis tu as ce changement. Mais c'est notre expérience, des gros changements on n'a plus fait.

*Maintenant, selon votre avis personnel. Est-ce que vous apporteriez encore des améliorations au système qui est mis en place maintenant ?*

R.S. : Oui du hybride Agile, on va aller vers le full Agile parce que ça c'est le next step. Time-to-market c'est dans le monde des assurances comme dans le monde financier, l'innovation. L'innovation c'est quelque chose d'inconnu, mais il faut l'essayer pour faire quelque chose de différent des autres, on ne peut même plus le faire en hybride Agile. Mais il

faut vraiment passer dans un mode Agile et ça c'est le plus grand changement c'est au niveau board que l'on doit accepter. Ce que moi j'appelle full Agile surtout en innovation c'est le fast fail il faut l'accepter. Et là c'est un changement aussi important c'est : il faut sélectionner les collaborateurs qui peuvent vivre avec cette éventualité. Parce que cela veut dire qu'on travaille fort sans avoir de résultat et ça il faut aussi avoir le mind-set. L'acceptation que c'est comme cela. On dit toujours c'est comme travailler dans un jardin, on sème quelque chose dans le potager et le plaisir c'est quand tu peux récolter. Dans un fast fail, cela se peut que tu doives recommencer beaucoup avant que tu vas récolter quelque chose. Il y a peut-être beaucoup de gens qui vont abandonner. Et aussi au niveau board, s'il y a des affichements qui n'apportent rien il faut savoir continuer. La digitalisation est importante et là on doit pouvoir aller vite. Et aller vite, en ayant bien étudié tout à l'avance, ça n'existe pas. Et là, c'est le gros changement que l'on va faire. Mais c'est dans la continuité d'Agile et on ne va plus jamais revenir en arrière ça j'en suis convaincu.

*Et vous parliez de l'Agile mind-set, cela joue-t'il un rôle dans votre entreprise ? Est-ce que cela a une place importante l'Agile mind-set, le manifesto ?*

R.S. : En IT plus que le reste du business. Mais le mind-set c'est quelque chose que le business est conscient aussi et ils ont compris que ce n'est plus une question de livrer des projets IT et c'est aussi la préparation avant que tu puisses commencer l'IT. Ils doivent appliquer ce mind-set aussi. Et ça vit maintenant qui est renforcée aussi lors des dernières présentations que j'ai faites. ça sert à rien d'aller vite avec les solutions IT si la préparation va encore prendre 2 ans avant que l'on commence quelque chose. Dans le monde des assurances par exemple, avant de lancer un nouveau produit, il faut que Risk product, policy gestions des contrats et tout soient d'accord sur qu'est-ce que le nouveau produit va être avant que IT va commencer. Si cette partie-là doit aussi se faire maintenant dans un concept Agile, que tu es trois semaines ou six semaines. Et non ce n'est pas le produit, on abandonne et maintenant on étudie pendant un an, on est pas certain du projet et puis on le donne à l'IT qui le développe dans un mode Agile et à la fin on dit quand même la conception du produit a quand même pris un an d'étude pour un développement de 15 mois. Non, il faut que cela puisse se faire demain sur 6 mois au total et ça c'est maintenant le business qui doit s'adapter.

*Peut-on dire que ce mind-set est occupé de se répandre alors un peu partout dans l'entreprise ?*

R.S. : Ils sont conscients qu'il faut, mais on n'en n'est pas encore là. Le management, le directeur et tout cela ils sont conscients qu'il faut évoluer surtout du côté produit. C'est important et ils savent qu'ils doivent aller pour cela. Mais tous les employés ne savent pas encore travailler avec ce mind-set-là. Et ça c'est le next step que l'on doit prendre autrement on peut encore faire du côté IT pour les projets le système full Agile, mais le reste doit suivre aussi. Sinon le bénéfice ne sera pas là. Moi j'appelle Waterset Agile et toute la société doit évoluer vers cela. Next step.

*Et selon votre expérience comment juger vous les méthodes Agile ?*

R.S. : Moi j'ai l'impression positive, je suis aussi une des personnes qui a toujours défendu parce que mon expérience, Agile n'existait pas encore. Donc j'ai une longue expérience en IT où j'ai commencé comme analyste programmeur et analyste fonctionnel puis j'étais chef de projet et j'ai toujours travaillé de manière très étroite et en collaboration avec le business. Pour moi, Agile, il n'y a rien de nouveau, c'est le nom qui a changé, mais tous mes projets que j'ai fait, tous les jours je discutais avec le business et tout ce qu'on faisait c'était en collaboration avec lui. On faisait une équipe analyse fonctionnelle et on allait à côté, c'est ça qu'on va faire, vous êtes d'accord et j'ai jamais remarqué que les projets délivrés c'était une surprise quand on revenait que c'était livré. Mais on n'appelait pas cela de l'Agile ça n'existait pas. C'était une méthode waterfall, mais c'était une interaction avec le business, mais on ne parlait pas de product owner. Puis Agile est venu, oui c'est comme cela qu'il faut aller maintenant. C'est une méthodologie que l'on dit, mais ce n'était pas loin de la manière que moi j'ai toujours travaillé. Je suis tout à fait convaincu oui il faut aller vers cela et il faut appliquer cela et l'appliquer en général. Pour moi, c'était plutôt le next step. Je ne vois pas beaucoup de différence avec la manière de travailler avant avec mes départements business. Mais ce n'était pas généralisé comme cela dans le groupe P&V.

P&V c'est aussi un historique de fusion donc tout ce qui venait d'une grosse banque ING c'était livré il fallait un mandat sur les requiements par 15 organes avant que tu puisses faire le reste. P&V c'était plus informel qu'on travaillait, mais cela ne s'appelait pas de l'Agile. Quand c'est venu, j'étais une des personnes qui poussait. Et c'est peut-être pour cela aussi que s'il y avait des défauts je ne les ai pas retenus, parce que moi je dis non. Il faut aller vers ça. Ça c'est l'avenir pour la société et les changements que l'on devrait faire. Au niveau du management, on était vite convaincu. Avec Dirk Geerinck qui était là il venait de l'infrastructure, même lui aussi il disait côté infra, mais non c'est le pas qu'il faut faire. On était vite aligné pour aller dans ce sens-là. Ceux qui ne le font pas je ne comprends pas

pourquoi ils ne l'ont jamais fait. Et je ne pense pas qu'il y en ait qui le font et il y a déjà beaucoup d'autres qui sont déjà à l'étape suivante.

*Et si je vous demandais une définition d'Agile ou de l'Agile mind-set sur votre propre expérience, quelle serait-elle ?*

R.S. : C'est flexibilité, rapidité, transparence avec qualité attendue.

## Appendix 14 : Interview Stacey Buys

S.B. : Agile Coach at Pyxis

*Pourriez-vous vous présenter en quelques mots ?*

S.B. : Moi je suis Stacey. j'ai 42 ans, j'ai commencé à travailler en '97 avec à la base un graduat informatique de gestion. Depuis je suis retourné à l'université faire un Master in Computer Sciences à l'UMONS aussi. Je suis développeur à la base, j'ai commencé à développer directement en sortant de mes études, j'ai utilisé le même langage pendant une vingtaine d'années, de manière fluctuante. En 2012, j'ai changé d'entreprise et je suis arrivé dans une boîte où j'ai eu l'occasion de passer sur les technologies Microsoft et autres. Et en même temps, j'ai recroisé un ancien collègue avec qui j'avais travaillé quelques années auparavant. Lui a eu un parcours un peu différent, il a eu l'occasion de toucher à l'agilité, de toucher à une méthode appelée Scrum à différents endroits et s'est impliqué dans la communauté et est allé à des conférences à Paris et Toulouse, etc. Et tout en continuant à s'impliquer, il a passé des certifications et est devenu trainer. Et c'est dans ce cadre que je l'ai revu et on s'est revu à une journée de conférences. Il m'a dit qu'il donnait des cours. Je me suis dit que cela m'intéressait. Ma première formation on me l'a donné en 2010 chez un précédent employeur. Ils avaient décidé de changer un petit peu l'organisation au niveau de l'informatique, le directeur informatique était intéressé par ce genre de choses, il a été suivi par la direction, et donc on était parti là-dessus et on a tous fait une belle formation et une des équipes a commencé par mettre cela en place, moi après je suis parti.

Donc je rencontre mon ancien collègue, et on discute, je suis intéressé, et je me paie la formation. Il faut savoir que ce sont des formations qui ne sont pas données, c'est quand même 1.800 € donc là c'est vraiment un investissement personnel et donc j'ai eu ma première certification et je n'ai de cesse de, comme on dit, l'évangéliser un petit peu. Partout où je suis passé j'ai dit voilà on peut changer ça et ça. Au fur et à mesure du temps, là je suis à la certification, mon but c'est de faire comme mon ancien collègue et de devenir traineur donc j'ai encore de longues années de travail là-dessus, il y a x certifications à passer et il y a beaucoup de travail. Je suis « tombé amoureux de la méthode » parce que c'est quelque chose qui me correspondait assez bien. J'ai été à un moment donné correspondant qualité, j'ai travaillé pour la cellule qualité la première société où j'ai travaillé donc tout ce qui est normatif comme normes ISO, etc. Améliorations, qualité et autres, c'est quelque chose qui me touche beaucoup. Donc, les méthodes Agiles, indirectement se tournent sur l'amélioration

continue, le PDCA c'est vraiment la base, en fait c'est un peu cela qui m'a plus. C'est que je voyais quelque chose de concret dans ce qui moi me touche et je suis parti là-dessus.

Il faut savoir qu'en informatique, et bien voilà les trucs importants dans les méthodes normales, on va parler de Prince 2 parce que c'est quelque chose de très lourd, très carré, ça a été fait par les anglais, il y a énormément de sociétés qui travaillent un peu au petit bonheur la chance. Tu as une analyse qui sort comme cela, on le donne au développeur qui se débrouille et on développe version « la débrouille ». Ça marche, ça ne marche pas, mais voilà et donc un informaticien quand il va passer son interview on va parfois lui demander : qu'est-ce que vous avez utilisé comme méthode ? et moi je me suis régulièrement retrouvé à dire : cela ne veut pas dire grand-chose. Et donc j'ai très bien accroché à ces méthodes, j'ai énormément investi, j'ai beaucoup lu, une bibliothèque absolument épouvantable. Et suite aussi a de nouveau une conférence, qui était organisée par mes collègues, maintenant je fais partie de l'équipe, j'ai croisé mes deux patrons et on a été en discussion pendant 4 ans. On a une opportunité, cela ne marche pas et finalement je suis arrivé dans cette société au 1<sup>er</sup> décembre avec tout mon background.

Je me suis occupé d'équipes, j'ai essayé de faire passer dans certaines boîtes ce type de méthodes, ça ne passait pas. Dans la dernière où j'étais, pendant une année, j'étais Scrum Master de l'équipe de développement tout en développant. Ce n'est pas la meilleure des places de se couper en 2. Et donc j'ai de nouveau eu contact avec eux puis cela s'est concrétisé. Ma première mission, j'y suis toujours, elle va se terminer fin du mois, donc 7 mois de mission, j'ai rejoint en fait mon patron qui avait besoin d'aide, car tout seul ce n'était pas assez. C'est une grosse boîte, il y avait 13-14 équipes à soutenir, alors on est allé là en coaching pour essayer de les aider d'améliorer le processus, les aider à performer un peu plus. Lui s'est détaché un peu au fur et à mesure du temps parce que forcément il a autre chose à faire. Il doit s'occuper du recrutement et d'autres clients. Et moi j'ai continué et je suis full time là-bas, mon travail consiste à les coacher Je suis en 4/5 chez Pyxis et le lundi je donne cours à Henallux à Namur à l'IESN,.

Ça veut dire quoi les coacher ? Les observer beaucoup, relever les petites choses qui pourraient être corrigées, les petits travers que certains développeurs peuvent avoir, conseiller les Scrum Master (Je reviendrai sur ce que c'est un Scrum Master), les soutenir, les aider dans les résolutions de problèmes. Même chose pour les Product Owner et s'il y a quelqu'un qui m'appelle à un moment donné et qui dit : « Ecoute viens voir, car là on sent qu'il y a quelque chose qui ne va pas », je suis là, je vais les voir. Il y a une équipe ici dernièrement,

eux ils avaient une méthode et avaient besoin d'en utiliser une autre. Ils ont eu besoin du coach parce qu'ils ne savaient pas trop quoi et comment faire, donner des évaluations sur les points à développer, ils ne savaient pas donc voilà c'est le genre de choses, on accompagne. En gros, c'est comme un coach sportif, tu es à côté du gars qui veut faire du sport et donc tu es là pour le motiver, tu es là pour le pousser au-delà de ses limites, t'es là pour lui donner les bons conseils pour qu'il ne se blesse pas. Globalement parlant, tu prends un coach sportif et tu translate cela vers l'informatique et tu as un coach Agile.

Alors, les méthodes Agiles. Au début de l'informatique, c'était au petit bonheur la chance puis, il y a certaines méthodes qui se sont mises en place, avec du bon et du mauvais.

Et donc, au début, après le chaos, on a vu apparaître Waterfall (et d'autres). Puis dans les années 90, les méthodes dites légères, les « méthodes » Agiles.

Pourquoi choisir une méthode plutôt qu'une autre ?

Il y a une question de feeling par rapport aux personnes qui doivent travailler, mais il y a aussi une question de contexte. Tu peux avoir des environnements extrêmement simples ou d'autres très complexes. Il y a des méthodes qui vont très bien s'appliquer pour quelque chose et d'autres pas du tout. Certains vont te vendre le framework Scrum comme étant la panacée, comme étant la solution pour tout. Moi j'estime que ce n'est pas vrai. Ça marche pour énormément de choses oui, mais il y a tout un contexte autour qui doit se mettre en place et c'est là que viennent les difficultés.

Bref, au début, après « au petit bonheur la chance », il y a au Waterfall...

c'est quoi les méthodes Waterfall ? On fait quelque chose d'un bout à l'autre et puis on passe seulement à l'étape suivante. Il y a l'expression du besoin, cahier des charges, etc. Et puis on fait toute l'analyse et une fois que l'analyse est terminée, on fait valider. Le client a signé c'est bon. Tu te trouves avec une grosse caisse avec plein de papiers et en gros s'il y a une erreur là-dedans, tu vas t'en rendre compte un an, deux ans, trois ans plus tard quand cela aura été développé et là le client va dire : « Ah non, moi je ne voulais pas de ça, non ça c'est pas ce qu'on vous a demandé, là ça ne fonctionne pas », car il y a un contrat établi longtemps avant et le problème de ces gros contrats, c'est que techniquement parlant il est cadenassé. Le client a beau dire qu'il n'avait pas demandé un élément A mais qu'il avait demandé un élément B cela, mais trop tard c'est signé donc vous payez et je ne veux plus rien entendre. L'analyse a été faite 2 ans avant. L'analyste est peut-être parti, il est peut-être mort... enfin soit, problème

de pertes de connaissances éventuel et aussi du contenu de l'analyse. Ce n'est pas parce que je te dis quelque chose que dans 2 ans ce sera encore vrai. Le marché peut changer, les technologies peuvent changer.

La lenteur de ce type de processus, a engendré certaines réactions chez les développeurs, les analystes, les chefs de projets, et le mouvement Agile vient de l'Amérique, x personnes au fur et à mesure du temps, dans les années '80-'90 ont remarqué qu'il y avait des choses qui ne fonctionnaient pas et ont testé certaines autres. Chacun de leur côté ils ont testé chez un client, chez un autre. Et ils ont adapté les manières de faire petit à petit : on va changer et on va faire une réunion d'une certaine manière plutôt qu'une autre, on ne va plus faire des documents de 300 pages et on va juste se limiter à un autre format plus compact ou plus de documents, mais moins épais sur des sujets plus précis. On va travailler plus vite, on va faire des petits cycles. Au fur et à mesure du temps, ils ont pu observer que leurs petits tests fonctionnaient mieux que ce qu'ils faisaient avant et un jour ils ont décidé de tous se voir. Ils se sont réunis en 2001 pour travailler et échanger leurs idées et ce qui en est ressorti c'est que tous avaient la même vision, en ce sens qu'il fallait travailler beaucoup plus vite (livrer plus vite, plus souvent de petits morceaux et non pas un mastodonte 4 ans après le début du projet), avoir un feed-back beaucoup plus important et ne plus seulement penser aux chiffres et aux contrats, mais revenir à l'essentiel qui est l'être humain. Et donc en est sorti ce qu'on appelle le « Manifeste Agile pour le Développement Logiciel » ou en plus court et plus mauvais, le Manifeste Agile parce que si tu enlèves développement logiciel cela peut engendrer plein de mauvaises compréhensions. C'est juste 4 principes enfin 4 oppositions : on préfère les individus et les interactions aux les processus et outils. On reconnaît les seconds, mais on préfère les premiers.

Donc un bête exemple que je donne toujours, si tu es au 3<sup>ème</sup> étage et que tu veux l'aide de quelqu'un qui est au 5<sup>ème</sup>, tu envoies un email. La personne peut être malade, être absente, dans une réunion ou autre et toi tu attends ta réponse. Si tu la veux tout de suite, le mieux c'est d'avoir la personne face à face. Tu attendras peut-être le lendemain, car voilà tu as d'autres choses à faire, mais toi tu attends la réponse et le lendemain tu te décides et tu vas voir et on te dit qu'elle est en congé de maladie une semaine, elle s'est cassé la jambe.

Et donc utiliser l'outil c'est bien, mais cela va beaucoup plus vite de se lever, prendre les escaliers ou les ascenseurs, mais tu montes jusqu'au bureau, il est où Jean-Paul ? Et tu as son collègue qui dit : « Il est en congé », donc tu sais que tu n'auras pas ta réponse tout de suite et

cela c'est ce qui est privilégié ce sont les interactions. L'exemple peut se répéter sur tout le manifeste, mais je suppose que tu les connais, les 4 phrases et les 12 principes derrière.

La base de l'agilité ce n'est que cela. Une réflexion qui a quand même duré quelques années de x personnes pour essayer d'améliorer le travail, pour essayer d'améliorer la réponse de l'IT à tout ce qui lui est demandé. Moi je l'ai vécu, j'ai travaillé sur un projet qui a été mis, après mon départ, en production un an en retard. Ça a provoqué quoi ? Le client a eu droit à un paiement d'astreintes du fournisseur et fourniture de secrétaires parce que c'était de l'époque de l'euro et de l'an 2000 et donc les secrétaires pendant un an ont fait du double encodage francs belges/euros pour pouvoir fournir les nouvelles factures en euro aux clients.

Ce genre de gros process peut coûter cher et donc au niveau de l'agilité, tu n'as pas que Scrum, on a commencé Rationnel Unified Process avant tout cela d'ailleurs c'est ce qu'ils ont utilisé à la Commission Européenne puis XP là c'est le développeur. XP c'est vraiment au niveau du développement, on fait ça comme ça, on écrit des tests avant de développer, etc... enfin bref. Scrum c'est un peu au-dessus, c'est plus de l'organisation de projets. On ne peut pas dire que c'est une méthode, c'est une boîte à outils (un framework) qui définit, un contexte, une manière de fonctionner, et dedans tu fais ce que tu veux. Mais souvent tu auras du XP qui se retrouve dans la vie des développeurs, parce qu'ils vont créer des tests avant d'écrire leurs codes et des choses comme cela. Je ne vais pas revenir sur tout le cycle Scrum, je crois que ça tu le connais.

*Pourriez-vous m'expliquer un peu l'organisation de votre client et la méthode qu'ils ont utilisés avant Agile ?*

S.B. : Alors sur l'historique de ce client-là, il s'occupe de paiements en ligne, je connais un coach qui a travaillé là. Avant, ils utilisaient d'autres choses et puis l'agilité est apparue dans cette société il y a quelques années, car depuis ils se sont fait racheter. Aujourd'hui ils essaient de reproduire le modèle Spotify. Comme le dit Spotify cela a marché chez nous, mais ne faites pas pareil. Tout le monde se dit que Spotify, l'idée est bien, mais ce n'est pas parce que cela marche chez Spotify que cela va marcher chez le voisin. Ils ont décidé de partir là-dessus, je ne peux pas dire que cela marche hyper mal à partir du moment où tu restes au niveau de l'informatique. Le problème de l'agilité, c'est bien si on cloisonne l'équipe de développement, à partir du moment où cela devient plus gros il faut vraiment que toute l'entreprise s'investisse. Le business doit comprendre ce que c'est l'agilité et doit s'organiser de manière Agile. Chez mon client cela ne se passe pas tout à fait comme cela. Donc cela

marche, mais il peut mieux faire. Il faudrait vraiment que les entreprises soient Agile dans l'entièreté. Dans l'entièreté cela veut dire du CEO jusqu'au petit développeur au fond de la cave. Ça, c'est beaucoup plus compliqué. Parce qu'être Agile, l'humain en premier oui, du coup les managers doivent lâcher du lest, doivent avoir confiance dans leur personnel. Et en général, les managers savent bien manager, savent bien serrer la vis, savent bien contrôler, mais relâcher c'est compliqué.

L'agilité au niveau de l'IT ça va, à partir du moment où tu dis aux développeurs vous allez pouvoir choisir ce que vous développez, ils sont tout heureux. Pour eux, leur travers, c'est qu'ils vont se dire : « Oui on est les maîtres de tout, on peut faire ce qu'on veut ». Non, il y a quand même des règles. Si je reprends Scrum, ça te donne un cycle, il ne te dit pas ce que tu fais dedans, tu dois développer parce que tu as des choses à développer. Les développeurs on ne leur dit pas : « Et allez-y les gars, travaillez 3 heures tous les mois ce sera suffisant », NON. Il y a une promesse faite par les développeurs. Et, à la fin de chaque cycle, que ce soit en XP, en Scrum ou peu importe, il y a une promesse qui a été faite et il faut sortir (= produire un morceau d'application) quelque part et il faut produire. Donc les développeurs ne sont pas laissés entièrement à leur bon vouloir. Allez-y les gars, produisez quelque chose comme et quand vous voulez, ce n'est pas vrai. Mais malheureusement quand tu essayes d'expliquer l'agilité, les gens voient d'abord : « Ah oui, les gars en T-shirt qui sont debout devant des papiers, des post-it ». Malheureusement, c'est le cliché que tout le monde a. Tu parles d'agilité, tu vas dire Agile, tu vas dire Scrum et puis si la personne que tu as en face de toi en a un petit peu entendu parler : Ah oui les post-it au mur. Mais les post-it au mur, c'est juste une manière de faire.

Pourquoi des post-it au mur ? Parce que tout le monde passe devant le mur, tout le monde peut voir ce qu'il y a au mur, cela facilite la transparence. L'Agilité c'est 3 choses en plus du manifeste, c'est la transparence, l'inspection et l'adaptation (méthode empirique, cycle PDCA). Donc la base de tout c'est d'être transparent. Si tu vas auprès d'un développeur et qu'il ne sait pas te répondre sur ce qu'il est en train de faire, non, il doit être parfaitement clair. Quand il a fini sa réponse, tu dois savoir exactement ce qu'il est en train de faire. Cela demande une autre manière de raisonner, cela demande du courage, cela demande de beaucoup d'implication. Et si c'est un manager qui vient, on a toujours la peur du manager. Peu importe l'entreprise, si l'agilité n'est pas globalisée, à partir d'un certain niveau tu sens un frein parce que devant toi tu vois un manager et la personne va penser : mon évaluation annuelle, mon bonus, mon truc. Il faut aller au-delà de ça.

Pyxis, c'est une entreprise libérée, on se dit tout, on a un problème ensemble tu vas me le dire. On ne se le dit pas en se tapant dessus. Communication non violente, etc. On se parle, on se dit les choses sans s'énerver. On se dit les choses. Tu as fait quelque chose qui ne me plaît pas, je me sens mal à l'aise, parce que tu as fait cela, il y a une communication qui doit absolument se faire et c'est tout un travail. Moi, comme je te disais, je suis là depuis décembre, j'ai énormément souffert en arrivant parce que ce n'est pas du tout la même manière de travailler. D'ailleurs c'est ce que moi je recherchais et en fait je ne me rendais pas compte du travail que tu dois avoir sur toi en entrant dans ce genre de processus, parce que tu es habitué depuis 20 ans, tu as un chef c'est carré et puis voilà.

Donc l'agilité ce n'est pas pour tout le monde. Il y a des gens qui pourraient tomber en dépression pour ce genre de truc. Tu dois prendre tes responsabilités, tu dois être transparent, c'est vraiment une autre manière d'être, de travailler. Quand tu es bien entouré cela marche bien. Et donc une transformation Agile s'il n'y a pas de coach cela risque de ne pas marcher. La société dans laquelle j'ai eu ma première formation Scrum, l'utilisation de Scrum aujourd'hui cela se résume à faire un daily meeting en envoyant un mail. Ce n'est pas Agile ça. Tu n'as pas de réunion, t'as pas des gens face à toi, ça ne va pas. Donc ce qui est à craindre c'est que les gens se disent : oui c'est chouette, c'est cool, cela va être sympa. Et malheureusement, ils ne sont pas accompagnés ou sont mal accompagnés et du coup oui forcément cela va mal se passer. Et là on va dire, de toute façon on va revenir au Waterfall vos méthodes Agiles là, les trucs ludiques cela ne marche pas. C'est comme dans tout, si tu apprends à nager tout seul tu es vachement doué. C'est la même chose pour les méthodes Agiles, si tu sais mettre cela en place tout seul, chapeau !

*Auprès de votre client, quelles étaient les principales raisons d'implémenter d'abord Scrum puis Spotify est que c'était comme vous avez mentionné, ça a l'air bien ou est-ce qu'ils avaient vraiment des objectifs, des attentes ?*

S.B. : Les objectifs premiers je ne les connais pas exactement, mais par rapport aux conférences du coach qui travaillait chez eux, il y a quelques années, quand il a commencé « à monter en puissance » en Agilité, à faire du coaching, etc. Il s'est mis à faire des conférences, et une des premières, c'était livrer son expérience justement. Comment lui est passé de rien à l'agilité.

Comme chez beaucoup, c'est le truc de geek. Le développeur qui dit : « Oui, mais chef ce ne serait pas mal si on pourrait travailler comme cela ». Et donc au fur et à mesure, on

commence à utiliser les méthodes, on utilise Scrum et cela ne marche pas mal, et puis cela commence à déborder du vase et cela se répand. Et au fur et à mesure, cela s'est répandu, tu as une équipe, deux équipes, trois équipes, les développeurs sont contents et puis au bout d'un moment il faut synchroniser ces gars. Je ne sais pas quand le modèle Spotify a été commencé. À mon avis, l'année dernière quand mon collègue est arrivé là-bas. Mais en tous cas, je sais que cela fait au moins 5 ans, qu'ils utilisaient Scrum.

*Ne savez-vous pas les raisons qui les ont poussés vers cette direction ?*

S.B. : Je sais pas vraiment les raisons, de ce que j'en sais c'est un désir de certains développeurs. La personne de tout à l'heure, l'a bien expliqué lors des conférences, il y a 3-4 ans, c'était vraiment cela, aller trouver les responsables et leur dire : « ça ne va pas, on devrait faire ça comme ça, ça comme ça ». Si tu es tout seul, on ne t'écoute pas, mais si tu commences à être 3,4..,10 tu commences à trouver une oreille attentive. Donc ils ont commencé à mettre en place Scrum dans 1-2-3-4 équipes et maintenant il y en a 14.

Il n'y a pas que Scrum. Scrum c'est bien pour les projets, où tu as « des choses bien définies » si je prends un exemple d'un support téléphonique où tu ne sais pas le nombre d'appels que tu as sur une journée, Scrum ce n'est pas super top. Il y a une méthode qui s'appelle Kanban où là c'est plus orienté flux continu. Dès qu'il y a quelque chose tu as un ticket et puis ils sont éliminés un à un, donc on a des équipes chez ce client principalement Scrum et puis 2-3 qui font du kanban, et il y en a une qui mixe les 2.

*Y a-t-il encore d'autres méthodes utilisées ?*

S.B. : Là-bas, non.

*Et comment ces méthodes sont-elles mises en place ? Ont-ils donné des formations ? Ont-ils pris des coaches Agiles ?*

S.B. : Ils ont pris des coaches, les employés ont été formés et encore aujourd'hui les Scrum Masters peuvent en ligne se former en vue de la certification et faire les examens pour obtenir les certifications. Mais, principalement aujourd'hui, les cours sont des cours en ligne. C'est plus facile.

*Et les méthodes, ont-elles été implémentées en une seule fois ? Ou cela a été par étapes ?*

S.B. : Je n'étais pas là à l'origine, à mon avis, c'était par étapes. Arrêter le vendredi et se dire allé lundi on commence Scrum, cela ne marche pas. Il y a énormément de travail et cela ne se voit pas beaucoup avant de mettre en place un cycle Scrum dans une équipe. Il faut bien préparer le backlog avec tout ce qui va devoir être développé donc cela demande de la réflexion il faut préparer l'équipe, pour trouver les développeurs, trouver le Scrum master, désigner les product owner. Ça c'est un processus qui peut prendre un certain temps il faut être accompagné par un coach, il faut être formé et ils n'ont certainement pas tout fait du jour au lendemain.

*Quelles modifications avez-vous faites chez le client et pourquoi ? Auriez-vous des exemples ?*

S.B. : Je ne dirais pas que ce sont des modifications, nous on apporte des conseils, s'ils ne veulent pas, ils ne veulent pas. On n'a pas de pouvoirs contraignants. On est là pour coacher. On dit : « Ecoute, tu es en train de faire cela de cette manière-là, tu es droitier, tu utilises la main gauche, si tu utilises la main droite cela ira mieux. », s'il répond : « Non, je veux continuer avec cette main-là », tu ne peux pas attraper la main et dire non utilise l'autre. Quand on est coach on conseille, on donne notre avis, on observe des choses, soit tu vois cela personne par personne, soit tu vois cela avec l'équipe complète, soit tu vois cela avec le Scrum Master pour son équipe. Si c'est le Scrum Master pour le Scrum Master, tu peux très bien lui dire : « Ecoute il se passe ça, essaie plutôt de leur faire faire de cette manière-là ou autre ». Je donne un exemple, le daily meeting, ce n'est pas du reporting. Ils font une promesse au début du sprint, voilà on va développer ça, ça, ça. Si le daily meeting c'est juste se limiter à voir les développeurs qui regardent les Scrum master : « Moi j'ai fait cela hier, aujourd'hui j'ai fait cela je n'ai absolument rien qui bloque ». Le deuxième dit : « Et bien, moi j'ai fait cela hier et le point est bloqué parce que j'ai besoin de telle et telle information, et tel et tel service, ou telle et telle machine, et il faut l'installer et en attendant je vais faire cela ». Ok le Scrum master est au courant et les autres développeurs sont au courant, mais on rate l'information la plus importante : est-ce qu'on est toujours sur les rails pour réaliser la promesse que l'on a faite au début du sprint ? Un des premiers conseils c'est de dire au Scrum Master, écoute, ce n'est pas du reporting essaie de faire comprendre à tes gars qu'ils devraient plus viser la promesse que plutôt « au jour le jour ». Si tu prends le Scrum guide, il va te dire vous devez répondre à 3 questions. Oui il faut y répondre, forcément, c'est bien que tout le monde soit au courant, de ce que chacun est en train de faire, ne serait-ce que pour prendre la main s'il y a un accident de voiture, une maladie ou un enfant malade. Mais le but à la fin du

cycle, à la fin du sprint, c'est de réaliser un agrément qui va être rajouté au produit final, et donc tous les jours, ils doivent se demander est-ce qu'on va pouvoir le livrer. Ça c'est la question la plus importante. Si jamais, il y a des petits changements à faire, tu peux éventuellement réduire le scope, pour faire une fiche client tu vas juste mettre le nom, prénom, adresse, mais pour l'instant tu vas pas faire les professionnels tu enlèves le numéro de TVA voilà. C'est un exemple bête, mais c'est restreindre un peu le scope du sprint, ils doivent s'en rendre compte. Si tu n'as que des gars qui viennent faire du reporting, c'est bien l'information est passée, on voit que les gens travaillent, mais si pendant 3 jours le gars est bloqué, il y a 2 choses, un le Scrum master ne fait pas son travail, là en tant que coach il faut bien les prendre par la main. C'est ce que j'ai dû faire dernièrement. Le Scrum master nouveau en place. J'étais là le vendredi, je n'étais pas là le lundi, j'étais là mardi et le blocage du vendredi était toujours là. J'ai regardé le Scrum master : « Et quoi ? pas résolu ? » - « Ah, mais c'était parce que c'était le service x qui devait faire cela... » j'ai donc répondu : « OK et bien vas-y tu vas avec les développeurs, tu débloques les choses, c'est ton rôle ». Donc on accompagne les gens de cette manière-là, on observe et on donne des conseils. Maintenant, parfois, on donne des conseils et on t'envoie balader clairement. Ce n'est pas des débutants en Scrum, donc les choses que tu peux leur demander, éventuellement, pour les aider à voir par certains travers qu'ils peuvent avoir dans leur manière d'appliquer la méthode, ils peuvent te dire non. Moi cela m'est arrivé. Il y a 4 mois j'ai demandé quelque chose, le Burndown chart. C'est une pratique qui est de moins en moins utilisée, mais pour leur montrer qu'il faut quand même garder cette vision du but en fin de sprint, je leur ai demandé de faire ce graphique. Le gars m'a regardé : « Tu veux le Burndown ? ». J'ai dit oui et il m'a répondu : « Et bien, tu le fais toi-même ». Voilà c'est quelqu'un avec de la personnalité. Il ne veut peut-être pas aider peu importe, dans son cas, je dirais que non, mais il est certain de travailler de la meilleure manière alors là en tant que coach OK. Ce n'est pas toi qui m'as demandé de venir, tu ne veux pas essayer de comprendre. À moi de trouver une autre manière de leur faire regarder leurs travers. Donc s'ils ne veulent pas les regarder de la face A et bien on va peut-être le faire regarder de la face B, tu vois. C'est une des choses qui est compliquée quand tu es coach, c'est que tu n'es pas là pour obliger les gens. Tu dois les « amener à », c'est comme un thérapeute. La solution est en vous. Il faut qu'elle monte et qu'elle sorte, en tous les cas, qu'elle vienne jusqu'au cerveau quoi. C'est compliqué en tant que coach.

*En tant que coach adaptez-vous les méthodes à leurs besoins ou ils suivent juste une méthode ?*

S.B. : On en discutait encore tout à l'heure, avec mes collègues, comme le dit mon chef tu ne dois pas être dogmatique. En gros, Scrum est très bien défini, c'est très carré, si je prends l'exemple très simple du daily Scrum. Daily Scrum = tous les jours. Là où je suis il y a des équipes qui le font 2 fois par semaine et cela leur va très bien. Parce qu'ils sont tous en train de travailler les uns à côté des autres et se parlent tout au long de la journée et donc le fait d'avoir une réunion pour se dire ce qu'on est en train de faire, si on est toujours bon devient un petit peu obsolète, parce qu'ils le savent. Ils sont tous en train de travailler ensemble. Une chose qu'il ne faut pas perdre, c'est la base de la méthode, c'est l'inspection, l'adaptation, si on te dit on ne fait pas de rétro, là ils cassent carrément la méthode. Le daily sert à garder un moment d'inspection pour tous les jours se dire, est-ce qu'on est encore dans les rails. Tu n'es pas obligé de le faire, mais dans le cadre de la rétro où là tu viens de faire 15 jours de développement, on n'a pas réussi, quelle en est la cause ? Qu'est-ce qu'on peut faire dans les semaines à venir ? S'ils ne font pas cette remise en question, ils ont tout perdu.

Il y a quand-même des choses sur lesquelles on ne peut pas les laisser faire, il y en a d'autres sur lesquelles oui on adapte. Tout le monde n'a pas besoin de l'entièreté d'une méthode, il y a des choses qui vont marcher pour certains et pas pour d'autres. L'évaluation de ce qu'il y a à développer, ils vont faire cela avec planning poker. Il y en a d'autres, cela ne leur plaît pas de jouer aux cartes alors ils vont utiliser des tailles de T-shirt pour dire cela c'est énorme, du XXL, ça c'est super short c'est du XS. Il y a des pratiques qui ne sont pas définies par Scrum, mais qui sont juste des pratiques, ils ont le choix. Il y a une base à respecter, oui maintenant tu peux en dévier de cette base, parce qu'il y a des choses qui ne sont pas applicables dans ton contexte, qui ne sont pas utiles, etc. On dit toujours qu'il faut utiliser le Shu Ha Ri cela vient des arts martiaux, tu commences par recevoir toute la théorie et tu l'appliques telle qu'elle à la lettre ensuite tu commences à l'assimiler et tu comprends les choses et à la fin tu es un maître, tu es un boss et tu peux dire ça je n'en ai pas besoin et ça je remplace. C'est une courbe d'apprentissage, c'est la même chose qu'une courbe d'apprentissage. Quand tu es au début, il vaut mieux quelqu'un qui conseille et qui te guide, et cette personne peut te dire, oui non, fais pas cela comme ça, ne le fais pas on n'a pas besoin voilà.

Au tout début, moi j'étais très, c'est écrit comme cela, on va faire comme cela. Plus maintenant, parce que j'ai pratiqué, j'ai vu. Voilà tout le monde ne peut pas se dire psychanalyste, tout le monde ne peut pas se dire coach Agile parce qu'il y a aussi un travail sur soi et c'est seulement en pratiquant qu'au bout d'un moment, tu te rends compte qu'il ne

faut pas suivre le manuel à la lettre, mais il faut l'adapter. Tu n'as pas d'autre choix que d'adapter parce qu'une méthode telle quelle il va manquer quelque chose.

*Ces adaptations qu'on fait par rapport à l'entreprise, ça se fait à un moment d'évaluation ou c'était au cours de l'implémentation ?*

S.B. : C'est au fur et à mesure, eux ils sont en Scrum, même les équipes qui font du Kanban s'autoévaluent régulièrement et au fur et à mesure du temps ils remarquent qu'ils ont besoin de dévier un petit peu, de faire autre chose. Il y a des équipes dans l'entreprise comme ils le disent si bien, ils étaient Agiles avant l'agilité, c'est-à-dire, ils n'ont jamais mis en place tous les processus Scrum, mais leur manière de faire, très start-up fait que c'était comme cela. Je disais, ils sont tous les uns à côté des autres très ensemble et là c'est à la bonne franquette qu'ils fonctionnent très bien parce que c'était tout petit. Et même si maintenant ils ont été rachetés, ils font partie de cette grosse boîte, ils ont eu l'opportunité de travailler de cette manière-là. Ensemble en comité restreint et l'un à côté de l'autre, les développeurs connaissent mieux le produit que les gens qui sont sensés leur donner le travail parce que ces gens-là ont changé et donc cela fonctionne très bien et très vite.

À la limite, maintenant c'est eux qui se plaignent que les personnes qui s'occupent de la production sont trop lents. Voilà il y a un niveau de maîtrise assez disparate selon les années d'expérience de chaque équipe. Donc, il y en a tu les laisses travailler et tu sais que cela va aller très bien, ils n'ont pas besoin de toi. Et d'autres, comme ils disent, ils viennent juste de mettre certaines choses en place ils ont besoin de la présence de quelqu'un d'être conseillés. En plus, tout le monde n'apprend pas à la même vitesse.

*Vous m'avez parlé qu'ils faisaient eux-mêmes des évaluations, quand et comment se font-elles ? Par laps de temps ? Par laps de projets ?*

S.B. : Les rétrospectives dans Scrum sont obligatoires une par cycle donc une par sprint. Les sprints, c'est de 1 à 4 semaines, et c'est le dernier évènement du sprint, donc s'ils travaillent du lundi au vendredi. ils commencent le lundi, tu fais des sprints de 15 jours le vendredi de la semaine qui suit tu termines par une rétro. Et là il y a plusieurs formats, ça dépend un peu de l'équipe, du Scrum master, un peu du jeu pour essayer de mettre un peu les gens dans l'ambiance parce que quoi qu'il arrive ta personnalité reste. Donc si tu as des gens timides, même s'ils sont frustrés par quelque chose, ils ne vont pas le dire, mais par contre, ils vont l'écrire sur un petit post-it ça, ça va. Et puis, il y a d'autres rétro où les gens se mettent autour de la table : « bon qu'est ce qui n'a pas été, moi là ça n'a pas été, là ça n'a pas été », la

personnalité joue vraiment sur l'expression que chacun va avoir. L'utilisation de jeux va aider les gens qui sont moins ouverts à justement s'ouvrir et à parler.

*Donc l'évaluation de la méthodologie se fait surtout avec des feedback ou il y a d'autres méthodes aussi ?*

S.B. : Des feedback parce que ce sont les développeurs qui manipulent et qui doivent maîtriser le processus. Et donc, c'est eux qui se posent des questions et naturellement il peut y avoir l'input de personnes extérieures, mais rares sont les moments dans une rétro où tu fais intervenir quelqu'un d'autre que l'équipe même. L'équipe avec le Scrum Master et le Product Owner, mais normalement, ils sont présents à la rétro. C'est uniquement ceux qui sont concernés par la rétro et par le processus qui sont présents. Donc en dehors de ces personnes-là tu n'as normalement jamais personne. Parfois il y a un manager qui voudrait venir, mais il faut que l'équipe soit d'accord parce que forcément la rétrospective c'est là où tu peux casser du sucre sur le dos de ton chef. Ça c'est un petit peu plus touchy mais c'est eux qui gèrent. C'est leur gestion. Clairement, c'est de l'autogestion.

*Est-ce que les managers et les éventuels clients ont aussi un pouvoir de feed-back sur les projets de comment ils se sont déroulés, par rapport à cette méthodologie ?*

S.B. : Tu as une présentation à chaque cycle de nouveau, pour ce qui est de Scrum, tu as la revue. Oui tu dois avoir les clients qui sont là pour dire : « ah non ce bouton-là je l'aurais voulu en bleu et vous l'avez mis en vert, cela ne va pas ». Tu leur présentes le produit dans l'état où il est à ce moment-là et tu as du feed-back directement, éventuellement au niveau de l'utilisation ils l'auront avant, après ou pendant peu importe. En tout cas le feed-back doit être régulier. L'introspection et le feed-back de l'équipe doivent revenir régulièrement et doivent être tenu compte régulièrement. C'est pour cela que Scrum travaille en sprint et dans le sprint tu fais tout.

*Par rapport à l'avant Scrum et Sportify est-ce que les feed-back des clients et des managers sont davantage positifs ?*

S.B. : L'entreprise où j'ai travaillé qui est passée de l'une à l'autre, c'est difficile à dire parce que là il n'y a que l'IT qui s'est penché sur Scrum pas le reste. Donc, le point de vue livraison au client, n'a pas forcément été plus rapide, on dépendait aussi des lois donc les commerciaux continuaient à travailler de la même manière. Le bénéfice n'était qu'au niveau de l'informatique et du coup il était très fortement réduit. Les clients, on n'a jamais eu de feed-

back par rapport au changement de méthode parce qu'ils ne voyaient pas. Eux, ils voyaient à intervalle régulier, un release 4 fois l'année et le release arrivait chaque fois au même moment. Ils avaient une nouvelle version, ils avaient une notice qui disait il y a ça et ça de nouveau dans l'application, mais ils n'avaient pas la vision de tout ce qui se passait à l'intérieur de l'entreprise. Là, il n'y a pas de bénéfice. Tu as juste le bénéfice des analystes et des responsables produits qui sont dans l'entreprise. Ce n'est pas le client final. Oui tu as un feedback mais ce n'est pas le client final donc il se pourrait très bien qu'ils se trompent, cela arrive. « Moi je voulais cela comme ça, mais les clients ils ont 3 clics en plus de souris ça va pas ». C'est seulement après que cela va chez les clients et les clients râlent, il y a des clics de souris en trop. « Moi j'ai 20 patients devant moi, je n'ai pas le temps de faire systématiquement pour chaque 3 clics sur l'écran », et du coup cela te revient, mais de manière indirecte. Ce n'est pas les personnes qui expriment ce problème qui viennent te le reporter. Ce sont les intermédiaires.

*Selon votre expérience, que pensez-vous des méthodes Agiles en général ?*

S.B. : Venant d'environnements très normatifs, j'apprécie ce cycle, où on fait, on demande et où on cherche constamment à améliorer. Au tout début c'est ça qui m'a fait accrocher à ce type de méthode donc je suis moi très attaché à ce genre de chose. Constamment se demander si on est dans le bon, comment s'améliorer, comment faire mieux, ça c'est ce qui me touche énormément. Ensuite en arrivant dans une entreprise libérée tout le côté humain moi m'importe énormément parce que forcément, on n'est pas des machines.

On peut aussi avoir des moments où on est super top et d'autres où on est prêt à étouffer, se coucher par terre, et donc avoir une bonne communication, une bonne collaboration avec les gens, c'est quelque chose qui, je ne vais pas dire qui fait passer la pilule quand tu as des choses difficiles à faire, mais, en tous les cas, l'ambiance est différente, la manière d'attaquer les choses est différente parce que tu sais que tu n'es pas tout seul. L'important c'est de travailler en équipe.

Donc les méthodes Agiles remettent l'être humain au centre des débats ce n'est pas pour rien, c'est l'être humain qui produit. Il y a peut-être des machines, des ordinateurs, mais pouvoir se reposer sur les autres parce qu'on sait qu'on n'est pas tout seul, cela aide à avancer. Voilà l'Everest, on t'attend en haut parce que nous on prend l'hélicoptère, là tu vas dire : « Oh non ». Et tout au long de ton ascension, il n'y a personne avec toi pour te dire allé vas-y. C'est la même chose que d'être tennisman ou footballeur s'ils doivent faire un match à huis-clos, ce

n'est pas terrible quoi. Si tu as les supporteurs qui sont autour, tu as une motivation supplémentaire, tu es boosté par ce qui se passe autour de toi. Et donc dans le cadre de Scrum ce n'est pas une personne à blâmer si un truc ne marche pas, c'est l'équipe parce qu'on est censé s'entraider. On est censé tous les jours se poser des questions, tous les jours. « Voilà les gars, j'ai un problème, il n'y a personne qui peut m'aider ? ». Il y a plus d'idées dans 2-3-4 têtes que dans une ». Du coup, il y a un soutien constant. Les autres te soutiennent et tu soutiens les autres. Cela est quelque chose qui m'importe énormément de ne pas laisser les gens dans leur coin dans leur problème, très clairement. Car moi j'ai connu ça dans un de mes précédents emplois : je dois faire des migrations, il y a un truc qui se passe mal tout le monde s'en fout. J'ai un client qui est à l'arrêt 4 jours. La seule personne qui pouvait m'aider travaillait à l'autre bout de mon bureau donc à 6-8 m de moi et le gars pendant 4 jours voit que cela ne va pas, je suis très expressif donc il m'entend râler, et il ne se pose pas de questions. Et au bout des 4 jours, il me demande : « Et ça va le client ? » Et je lui réponds que cela fait 4 jours qu'il est à l'arrêt. Et là seulement il est venu voir ce qui se passait. Au bout d'une journée on a réussi à débloquer le bazar. J'ai connu ce genre de situation, j'ai connu l'opposé où tu as un problème et on te trouve une solution tout de suite. Dans le premier cas, j'en avais parlé voilà il y a certaines personnes cela rentre d'une oreille et cela sort de l'autre. Ce n'est pas leur tâche ils ne se sentent pas impliqués.

Dans une équipe Agile, tout le monde est impliqué parce qu'on a tous ensemble quelque chose à faire, on a tous un but commun et on avance ensemble. Ça c'est quelque chose qui fait que moi je suis très orienté vers ce genre de chose.

*Si vous pouviez donner une définition par rapport à votre expérience sur les méthodes Agiles, Scrum par exemple, ce serait quoi ?*

S.B. : Travailler ensemble et mieux. Je ne vais pas dire plus vite, car ce n'est pas forcément plus vite, mais mieux. Cela induit une notion de qualité. Et la qualité est très importante. Pas dans mon parcours, mais dans les méthodes Agiles, il suffit de relire les principes, le feedback, fournir des logiciels de qualité. On essaie au maximum de réduire les bugs donc voilà. C'est un des points dans la définition qui joue beaucoup. C'est ensemble et mieux.

*Concernant l'Agile Manifesto, joue-t-il un rôle dans l'entreprise de votre client ?*

S.B. : Oui, dans le sens où, sans l'Agile Manifesto, nous n'aurions peut-être pas eu connaissance de toutes ces méthodes et non parce que jamais personne ne va se pencher sur le manifeste pour se dire : « Tiens notre méthode de travailler, cela correspond bien à cela ». Ils

ont mis un processus en place, Scrum, Kanban et puis voilà. Ils ne vont pas retourner en arrière pour aller voir les origines, est-ce qu'on est bien en adéquation avec ce qui a été dit ? Donc quand ils ont utilisé une méthode c'est censé être en correspondance avec l'origine (= le manifeste).

*Ont-ils été formé sur la philosophie Agile également ?*

S.B. : C'est compliqué comme question, on ne te forme pas sur une philosophie, mais sur une méthode. Et c'est au-fur et à mesure de le vivre, si tu fais toi-même une introspection, seul dans le coin, tu te rends compte du bénéfice que cela implique. Et tu te rends compte de ce que cela t'apporte. Ça a été mon parcours, j'ai 4 certifs, j'ai suivi 4 cours. On te dit : « la méthode elle est comme cela ». Un-deux-trois-quatre, voilà allez-y. Puis au fur et à mesure du temps, quand tu passes et repasses, tu dois t'impliquer. Mon parcours, du simple membre d'équipe, je suis allé un petit peu plus loin que cela, heureusement pour moi, en tant que coach. Mais tu te rends compte au fur et à mesure du temps, des bénéfices que cela implique. Et là, à ce moment-là, quand tu te rends compte des bénéfices, tu as un certain intérêt pour la philosophie de la méthode. Et moi c'est encore biaisé parce que c'est quelque chose qui me tient à cœur depuis longtemps. Par contre, lors de ma toute première formation, le fait que le formateur, et c'est le seul que j'ai vu ferait mis en avant la **roue de Deming**, cela tu le vois très rarement.

*Pensez-vous que votre client a encore des améliorations à faire ?*

S.B. : Oui aller plus loin dans l'agilité, c'est-à-dire, la faire déborder, ailleurs que dans l'IT. Le côté business, le côté RH aujourd'hui, il y a des entreprises qui mettent en place des processus Agiles dans les ressources humaines et cela leur apporte beaucoup. C'est quelque chose qui réhumanise l'entreprise. Là où je suis, comme je disais c'est un travail énorme et cela ne se fait pas tout seul. Il faut que les gens soient volontaires, impliqués. Il y a une entreprise ici en Belgique où mon chef est passé il y a 4 ans, on les a revus ici dernièrement et bien cela a mis 4 ans. Aujourd'hui, même la direction est à fond dans les méthodes Agiles et toute l'entreprise y est passée. J'y suis allé il y a quelque mois, c'était un paradis : l'engouement et la motivation des gens. Ils ont organisé un meet-up, ils ont fait venir des speakers et inviter des gens de l'extérieur comme une journée de conférence Agile. Tu viens, tu peux faire un jeu, et il y avait énormément de monde de la boîte et quand tu vois l'énergie qu'ils y mettaient, rare sont les employés qui vont mettre autant d'énergie dans un évènement d'entreprise et c'était franchement très sympa à voir. D'autant plus que j'ai discuté avec le

coach qui est employé là-bas, un coach interne, il a pu leur faire faire plein de changements entre autres et cela est assez rare, il a pu faire engager des Scrum Master full time. En général les entreprises rechignent toujours. Scrum Master cela sert à quoi, cela sert à rien, on prend un développeur, on prend un responsable ou l'autre et on lui colle en plus les responsabilités du Scrum Master. Cela freine un peu la mise en place du process, c'est compliqué. Là, il a réussi, il y a lui et 4 Scrum master pour s'occuper de la boîte.

Quand tu rentres, l'ambiance du bâtiment joue aussi, mais l'ambiance entre les personnes c'était enivrant. D'accord, c'était un event. Ce n'était pas pendant une journée de travail, mais tu les sens beaucoup plus impliqués. Dans les sociétés où je suis passé, ah oui tu fais du Scrum, oui il y a des post-it au mur, mais tu vas croiser quelqu'un dans le couloir il ne va pas te dire bonjour avec un grand sourire. Ça se ressent, quand c'est à un si haut niveau que cela, cela se ressent tout de suite. Mon client, oui ils ont des améliorations à faire.

*J'aimerais revenir sur la méthode Spotify. Est-ce que cela fonctionne assez bien dans leur entreprise ?*

S.B. : Oui cela fonctionne pas mal, mais Spotify est arrivé à ce modèle-là par le travail, par la réflexion, au fur et à mesure petite goutte par petite goutte le vase s'est rempli. Ici, ils sont arrivés en disant on va faire le modèle Spotify, on va trouver sur internet comment cela marche. Découper en équipes, organiser les équipes d'une certaine manière, ce qui est normal, ça a son intérêt. Au-delà de ça, Spotify a une manière de fonctionner que tu ne retrouves pas là où je suis parce que leur niveau de maîtrise de l'agilité n'est pas le même.

Les managers restent des managers. Quand il y a un manager qui vient et qui dit : « Tu dois faire cela comme ça », tu sens encore cette manière de faire dictatoriale du manager qui reste. Comme je te disais et là je parle de manager IT, l'agilité est bien au niveau des développeurs, mais dès que tu commences à monter c'est terminé. C'est comme des poissons dans un étang. Le pêcheur est au-dessus et les poissons se débrouillent très bien sans toi et puis de temps en temps tu mets un pied dans l'eau tu mets la canne à pêche.

Là c'est exactement la même chose l'agilité marche bien dans les développeurs et dès que t'essaie d'aller dans les niveaux supérieurs, tu sens bien qu'il y a un couac à gauche, à droite.

## Appendix 15 : Interview Pyxis co-founders

Interview Fondateurs Pyxis Technologies Belgium.

N.D. : Co-founder and Agile coach.

P-E.D. : Co-founder and Agile trainer.

*Pouvez-vous vous présenter en quelques mots ?*

N.D. : Je m'appelle Norman Deschauer, je suis coach Agile et avec mon associé Pierre Emmanuel, on a créé Pyxis. Moi j'ai un parcours de développeur logiciel puis analyste fonctionnel technique, chef de projet, burn out, et puis coach Agile. C'est un petit peu la renaissance.

P-E.D. : J'ai un parcours beaucoup plus technique. Cela fait 15 ans que je travaille principalement aux développements logiciels et puis dans le côté accompagnement technique. Les équipes à mettre en place, les automatiser, ce genre de choses là et je suis dans l'agilité depuis 2005, plutôt côté technique. Je travaille pas mal comme formateur aussi, mais pas en tant que coach Agile. Depuis une petite année, je suis en train de me réorienter en tant que coach professionnel, donc pas forcément dans le domaine de l'agilité, mais un point complémentaire.

*Pouvez-vous me parler de vos clients où vous travaillez ?*

N.D. : Actuellement, je termine une mission d'un an chez Ingenico, anciennement Ogone, qui font des paiements électroniques. Un collègue à la BNP, un autre chez Axa, Fédérale Assurances et surtout des grandes sociétés. Et cela va souvent dans la finance, banque.

P-E.D. : Je dirais qu'il y a les 2 parce que tu as travaillé aussi dans des PME auparavant, il y a un peu comme un shift qui s'est fait ces dernières années. On a beaucoup travaillé dans des PME, N.D. : a travaillé longtemps chez Delcampe qui par la taille est une PME une société de 20-25 personnes à Tubize spécialisée donc c'est un Ebay des objets de collections. Petite société, mais très grand nom dans le domaine de la collection. Et cela fait 2-3 ans que l'on se retrouve avec des plus grosses sociétés effectivement avec banque, assurances qui se tournent vers l'agilité de manière large. Il y a quand même des grandes banques qui se sont lancés dans des grandes transformations Agiles, depuis 3-4 ans.

N.D. : Oui, c'est ça. Mais qui parfois y vont un peu au bulldozer et engagent 200 coachs d'un coup.

Quelle méthode avaient vos clients avant d'implémenter Agile ?

N.D. : C'est-à-dire, soit ils n'ont rien soit ils ont inventé un truc, c'est un petit peu le mode empirique. C'est un mélange, c'est un peu ironique, de Waterfall - on voit souvent ce mot - et du grand n'importe quoi. Et finalement le grand n'importe quoi est souvent plus proche de l'agilité que des modèles beaucoup plus stricts. Et donc, c'est marrant de voir que les petites sociétés, souvent on leur donne d'abord une formation en introduction méthodologie, et ils disent : « Finalement, ce n'est pas très loin de ce qu'on fait ». C'est un petit peu en non structuré, c'est le côté émergent. Et un peu le fait qu'au jour le jour, là où les frameworks Agiles sont le plus structuré mais laissent peu de place à l'humain et l'émergence d'idées, la communication est aussi structurée. Et c'est la grosse différence entre le n'importe quoi et l'agilité, c'est qu'il y a une structure qui est basique et qui ne va pas être les grosses structures bien rodées comme: Prince 2, Itil et CMMI ; et qui vont plus être des recettes de cuisine qu'un framework.

*Quelles sont les raisons de leur implémentation Agile ?*

P-E.D.: J'aurais tendance à dire qu'il y aurait 2 ou 3 grosses raisons que l'on peut voir. Il y a un premier motivateur c'est un aspect financier où il y a une image que l'agilité va permettre de faire mieux, moins cher et avec moins de personnes. Il y a ce motivateur-là.

Petite parenthèse, dans les méthodes Agiles, il y a une méthode qui est le lean management. Quand on l'a entendu il y a quelques années, c'était pour une grosse restructuration dans l'acier et autre, ils lançaient derrière des plans lean. C'est des recherches d'économie financière poussée avec une vision assez spécifique, mais voilà c'est clairement des moteurs.

Un autre moteur, je pense que c'est une certaine remise en question par rapport à la concurrence. Ici, l'offre que je viens d'envoyer ici, c'est pour Ores. Ce qu'ils expliquent c'est assez typique de ce qu'on peut retrouver, c'est qu'ils se rendent compte que pendant un certain nombre d'années, ils étaient établis dans un monopole, dans une situation bien confortable et puis ils se rendent compte que le monde change. Il devient concurrentiel, que les clients demandent plus, différemment, etc. et donc ils ont un besoin de se réinventer, de se réorganiser. Et cela peut les amener vers l'agilité, car ils ont un besoin de transformation interne.

Un troisième élément potentiel je dirais, c'est d'aller davantage vers le côté responsabilisation, le côté humain, le côté autonomie, motivation. On parle beaucoup

aujourd'hui des entreprises libérées. En Belgique, il y a le groupe Decathlon qui a fait pas mal parler de lui dans cette idée-là où ils ont complètement transformé leur hiérarchie en interne pour mettre davantage l'employé au centre. Ils ont changé leur organisation pour donner beaucoup plus de responsabilités aux personnes. Et derrière cela, l'objectif c'est d'un côté de réduire les coûts et de l'autre, c'est de surtout retrouver une adhésion du personnel pour dire qu'on n'est plus juste des numéros dans la société, on est des acteurs importants. Et donner aux personnes la chance de s'investir dans la société.

N.D. : Et notre apport là-dedans, c'est que, souvent, on est amené sur les 2 premiers points qui sont l'économie : livrer plus vite au client, faire plus avec moins et parfois aussi se réinventer plutôt dans l'innovation. Et nous amenons souvent, on force un peu, de prendre conscience du côté humain de la transformation et faire comprendre que c'est très difficile de convaincre les gens s'ils ne participent pas à la création de ce qui les attend. Pour compléter aussi les exemples, ici en Belgique, il y a une société qui s'appelle Cosucra qui pratique aussi ce genre d'innovation managériale, qui est plus axée sur ses employés qui font, qui savent, on doit leur donner le pouvoir de décider dans l'entreprise. Parce qu'aujourd'hui dans le monde complexe que l'on est, même une PME entre la personne qui a l'info et la personne qui pourrait décider, qui est le CIO, CFO, s'il sait quelque chose il se passe trop de temps, il y a trop d'intérêts, trop de jeux de pouvoir et d'ego. Et donc finalement on ne sert pas l'entreprise, mais on sert une personne. C'est ça aussi qu'on essaie de casser, c'est renverser un peu la pyramide de l'entreprise. C'est le management au service de l'employé qui lui est au service du client, et finalement, tout le monde travaille dans le même sens. Je pense que oui cela se voit chez Decathlon. Ça, c'est les tendances actuelles.

*Ont-ils des objectifs stratégiques derrière l'envie d'implémenter Agile ? Ou cela vient-il plutôt des employés l'envie de devenir Agile ?*

N.D. : Ça de nouveau cela va dépendre, on est souvent appelé dans le secteur IT ou dans le département IT de l'entreprise, AXA vend des assurances, ils font de la banque aussi, mais ici dernièrement on a travaillé dans le département IT. C'est un vecteur propice à l'agilité, au développement rapide, itératif, incrémental.

*Les clients de manière générale, avaient-ils des objectifs particuliers ?*

P-E.D. : Pour reprendre, cet exemple-là, en fait ce n'est pas le top management qui nous appelle, mais c'est le management du département IT qui nous appelle. Je dirais qu'il y a peut-être 2 tendances, c'est dans les PME on a travaillé beaucoup dans les sociétés Agilys, ils

font l'application myShopi dans les téléphones. Et là, je dirais dans cette société-là, ils étaient peut-être 10 à l'époque. Je pense qu'il y a une envie des employés de changer leur façon de travailler et de changer aussi leur motivation et leur façon de collaborer dans l'entreprise. Donc là typiquement, je pense que c'est les employés qui étaient venus à une formation chez nous et puis qui ont demandé qu'on puisse avancer.

Par contre, cette logique-là dans des sociétés plus grandes, ce n'est pas un développeur qui va dire : « Tiens, j'ai envie d'agilité et on va appeler un coach. » Il y a peut-être une initiative du développeur qui va essayer de remplacer quelque chose, mais au bout d'un moment cela va être un manager qui va devoir reprendre la main pour lancer la transformation. Donc là-dessus, dans des sociétés plus grandes on va remarquer les deux : soit effectivement le côté management IT, qui va avoir cette envie, peut-être poussée par les employés, mais qui va amorcer ce changement-là, et puis cela va peut-être toucher les autres secteurs de l'entreprise ; ou alors dans d'autres cas, BNP c'est vraiment le top management qui va faire cela.

*Dans le cas du management IT, est-ce qu'ils ont des objectifs stratégiques précis ou ceux-ci viennent de plus haut ?*

N.D. : Chaque question que tu vas poser va amener à des cas, des branches. En fait, ce que j'ai déjà vu c'est qu'IT disait voilà, on voudrait être Agile. Du point de vue business, l'agilité est parfois perçue comme : ça coûte plus cher et ça prend plus de temps. Et parce qu'on aime bien les certitudes, c'est facile d'avoir des certitudes, on vit dans un monde complexe et complètement incertain, donc on fait des plans sur 2-3 ans et l'agilité c'est ramener ces plans à 2-3 ans à des morceaux beaucoup plus itératifs et rapides. Donc le danger et le challenge c'est pour l'IT de ne pas se faire mettre des objectifs. Ce que j'ai déjà vu sans citer de noms, c'est le management IT qui dit voilà on devrait passer en agilité et souvent le business qui chapeaute l'IT dit : « Ok, mais alors ce projet-là et ce projet-là, vous ne pouvez pas les louper ». Votre méthode miraculeuse, on va voir si cela marche, car déjà sans changer de méthode le fameux projet est irréaliste. Donc c'est ça le danger. Du coup à la fin, l'agilité ça ne marche pas parce qu'on n'a pas préparé le terrain qui permettrait que cela marche. On met trop d'ambition sur l'agilité. La recette que je vois aussi, pas forcément dans les grands groupes et dans les petits, on leur dit souvent : « Prenez un projet pas le plus stratégique, mais pas le plus petit non plus, et attendez-vous à ce que vous formiez une nouvelle équipe, une nouvelle méthode, un nouveau projet, et ne soyez pas trop ambitieux sur les résultats, laissez-leur la possibilité d'échouer et la méthode est là, justement, pour faire en sorte qu'ils soient créatifs, ils vont apprendre de leurs erreurs, il n'y a rien de certain ». Le fermetout est là pour

cela, trompe-toi, mais rapidement. Ne fais pas un projet de 2 ans, puis à la fin se rendre compte que finalement on s'est planté. Un projet Agile en quelques sprints, quelques semaines tu vas te rendre compte si tu es que la bonne voie ou pas. Après, à toi de voir si tu veux réajuster ou si tu veux que cela foire quoi. Donc, voilà ça, ce sont un peu les ingrédients que je peux avoir.

P-E.D. : Après, ce qu'on peut avoir comme objectif quand l'IT va démarrer le projet c'est aussi se rendre compte que cela ne marche pas avec le business. Ça ne marche pas dans le sens, la communication n'est pas bonne. Ou alors l'IT a l'impression de faire plein de trucs et puis le business a l'impression qu'ils ne font rien, il y a une différence de points de vue. Et donc un des objectifs potentiels de IT en démarrant l'agilité c'est de ramener de la transparence, de ramener une sorte de communication et de cohésion, entre le business et l'IT.

N.D. : Souvent on va utiliser des outils de visuel management qui soit numérique ou post-it sur les murs et donc c'est des outils de prise de décision et tu ne peux pas faire sans voir que le projet va foirer ou fonctionne bien. Tu vois ce que c'est Kanban ? Le business ne peut pas coller 1.000 post-it dans une to do liste en disant : « Oui, cela tient » et tu vas le faire. Alors qu'on voit très bien que l'on fait 5 post-it par semaine et on ne sait pas aller plus vite quoi. Donc, si tu en as mille, je te laisse faire le compte quoi. Il faudra x semaines avant d'arriver à ces post-it là. Et si, chaque semaine, tu viens et tu déverses 50 post-it et je n'en fais que 5, il y a un over-flow. Ça ne peut pas fonctionner quoi. On utilise aussi les principes Lean de visualiser le process de la demande business jusqu'à la mise en production, que ce soit un produit informatique ou un produit tout court, un service. Ça, c'est la cadence de l'entreprise. Donc tu visualises. Donc si le business change d'avis toutes les semaines, il y a une telle latence entre l'idée et la mise en production, que c'est rempli de déchets donc il faut soit raccourcir le délai soit qu'on puisse avoir des besoins plus raffinés avant de les envoyer dans la chaîne. Et donc c'est tout cela qu'on essaye de mettre en place.

Pour nous aussi un des axes d'améliorations ou d'attention qu'on essaie de mettre dans l'entreprise, c'est l'agilité. Ce n'est pas de l'IT ou du business c'est dans l'entreprise. On fait un produit qui répond à un utilisateur. Pour l'utilisateur c'est une black box comme google, Microsoft, Apple c'est une black box. Lui il veut un outil. Donc si l'IT et le business ne s'entendent pas, lui il va le voir direct dans son application, ça ne marche, pas ergonomique, ce n'est pas pensé pour lui. Ou pendant 6 mois il n'a plus de mise à jour et puis tout d'un coup, il a une nouvelle version majeure et il ne s'y retrouve plus. C'est toute la chaîne en fait qui doit être en phase. Pour moi, il n'y a pas de transformation Agile s'il n'y a pas une

collaboration propre dans toute l'entreprise quoi. Souvent, on est appelé par l'IT et nous on essaye de transpirer un peu dans le petit business, marketing, RH, c'est une transformation de l'entreprise et pas du département.

*Quelles sont les méthodes qui ont été implémentées chez vos clients ?*

N.D. : Souvent, donc on parle de l'IT, mais c'est même un Framework qui est utilisé ailleurs, c'est Scrum. C'est le plus populaire, c'est le plus utilisé, c'est maintenant 22 ou 23 ans qu'il existe donc c'est souvent ce qui revient.

*Ce sont vos clients qui demandent Scrum ou cela vient des conseils ?*

N.D. : En fait, l'idée est que Scrum est devenu très populaire, presque à la mode. Les agilistes conseillent Scrum parce que c'est populaire et les clients le demandent pour la même raison.

P-E.D. : Et puis souvent chez les clients, Scrum et Agile, c'est un peu la même chose, je pense qu'ils demandent Scrum parce qu'ils ont entendu parler de Scrum, mais cela s'arrête là. Dans les derniers clients que l'on a eu, ce qui revenait beaucoup c'était Spotify. Avant, dans les médias il y a quelques années, dans les sociétés un peu plus grosses, souvent c'est ce modèle-là qui revient probablement c'est ce qu'ils ont entendu et cela s'arrête là.

N.D. : Puis tu entends parler de travail, de squad, de guild, et donc tu as des nouveaux postes comme tribe leads qui apparaissent, avant c'était des chefs de projets, maintenant c'est des tribe lead donc cela fait un peu hype. Et je dirais même chez Spotify, ils en rigolent un peu parce qu'ils n'ont pas inventé un modèle. Ils ont juste pris de l'agilité ce qui les intéressait, ils ont épuré, nettoyé et fait 20.000 tests. Et puis c'est Henrik Kniberg qui a un peu distillé ce qu'on faisait chez Spotify, et le modèle Spotify est né. Donc les gens copient le truc, au lieu de se dire on s'en inspire, on assaisonne un peu notre mode de fonctionnement interne et on prend le meilleur quoi. Donc c'est devenu un modèle, il y en a d'autres comme SAFe, Scaled Agile Framework. Là c'est plus structuré et destiné à être un modèle. Encore que là, dans le vocabulaire, moi je trouve que c'est une erreur en fait. Je pense qu'il y ce besoin de faire du scaling de l'agilité, mais le fait de créer des modèles de scaling, pour moi cela a l'effet inverse en fait. On fait des silos horizontaux et verticaux en fait. On ne fait plus des départements, mais on fait des produits, des lignes de services. Tu regardes un organigramme avant et après Spotify les anciens managers sont des tribes lead on a juste renversé les couches, mais elles

sont toujours là en fait. Donc c'est toujours les gens qui avaient le pouvoir qui l'ont et donc tu sers toujours les mêmes intérêts en fait.

Sur le principe des transformations Agile c'est un parcours infini en fait, moi je ne connais pas aucun exemple de transformation Agile, pas qui a marché, mais qui est fini. Pour moi, c'est un parcours perpétuel en fait. Pyxis est sur la voie de l'agilité et peut être considéré comme Agile, mais on apprend tous les jours et on change tous les jours.

*Y a-t-il toujours des améliorations à apporter ?*

N.D. : C'est ça en fait. C'est pour cela qu'on insiste, la méthode Agile pour nous c'est une erreur de langage en fait parce que si j'applique cette méthode je suis Agile. Pour nous, c'est faux. Après si les gens veulent appeler cela la méthode Agile, OK ça va, on ne va pas faire les professeurs qui sont scandalisés.

*Comment les méthodes sont-elles implémentées chez vos clients ? Est-ce qu'ils sont formés, est-ce qu'il y a des postes qui se créent, comment tout cela se passe ? Tout se fait en un coup ou cela se fait-il par petites étapes ?*

P-E.D. : Il y a plein de façons différentes. Il y a une façon qu'on peut voir : Il y a un certain nombre d'entreprises, peut-être celle où ça vient du côté des employés, il y a des employés qui ont entendu parler d'agilité par des formations, des conférences, des livres ou autres et qui commencent à mettre en place des choses. Et là je dirais qu'en grosse caricature, on se retrouve avec 2 types d'équipes qui disent : « Et bien voilà, on a mis en place des trucs et ça roule très bien ou moyennement bien peu importe ça roule ». Ils considèrent qu'ils font de l'Agile et puis ils y en a d'autres qui vont continuer à s'interroger : « Finalement, on se rend compte d'avoir fait de l'agilité, mais en fait il y a des trucs qui coincent à droite, à gauche, on ne trouve pas, on n'a pas de réponses à nos questions ». Puis généralement, c'est dans ces équipes-là qu'on va arriver, c'est un peu comme cela qu'on a fait de l'agilité en fait. On avait mis en place notre façon de faire de l'agilité et puis on a commencé à aller voir les conférences et puis se dire en fait non. Et puis, peut-être à ce moment-là, il y a des personnes qui vont aller suivre des formations ou qui vont appeler des coachs pour venir. Tu demandais, est-ce qu'il y a des postes qui se créent, et bien pas forcément. Je dirais que tu as parfois, c'est juste un changement de casquette finalement. Tu as une équipe qui existe, qui met en place une façon de fonctionner et puis tu as une personne qui dit : « Et bien moi, je vais commencer à jouer le rôle de Scrum Master. Moi je vais prendre la casquette de Product Owner ». Ça va rester les mêmes personnes, mais qui vont redéfinir leurs rôles et leurs frontières. Il y a

d'autres cas où il y a des postes qui vont se créer, des coachs qui vont se rajouter dans les équipes de façon définitive ou pas.

N.D. : Chez Axa ils ont des coachs. Chez BNP, il y en a aussi en interne par contre, les petites structures cela devient tout de suite plus difficile.

P-E.D. : En fait, derrière la notion de poste à créer tu as un frein de se dire : « Mais toi tu prends le rôle de Scrum Master, mais tu ne vas rien faire, tu ne vas rien produire. » Donc effectivement pour une petite structure, tu as cette image-là. Mais non un Scrum Master, un coach, il ne produit pas donc c'est difficile de passer ce palier-là.

. Comme vous l'avez mentionné, les méthodes Agiles sont un modèle, les entreprises se rendent-elles bien compte de cela ? Font-elles des adaptations à ce modèle ?

P-E.D. : Il y a des entreprises qui sont très fortes dans l'adaptation du modèle. Tellement fortes que cela ne ressemble plus à rien, mais c'est une dérive qui est assez classique de se dire tu mets en place de l'agilité. Et disons, tu mets en place Scrum dans une équipe et puis il y a tel truc qui marche pas bien alors tu fais différemment ou tu l'enlèves, et puis tu l'adaptes, tu adaptes. Et au final, tu fais un truc, ce n'est clairement plus du Scrum. Et puis tu vas avoir certaines équipes, tu vas avoir des personnes qui vont amener l'agilité et qui vont venir avec un côté très dogmatique et dire le livre Scrum il dit cela, alors on doit faire cela. Et si on ne fait pas cela, ça ne marche pas. En oubliant le côté : c'est un framework. Et donc il faut s'adapter au contexte, au produit, aux personnes et on trouve un peu les 2 extrêmes.

N.D. : Donc il y a une valeur ajoutée en fait à utiliser le framework, à utiliser les pratiques clairement cela n'apporte rien autant s'en passer. Mais on a des outils d'assessment où sur les rôles et les meetings, les outils, ainsi de suite, est-ce que vous faites ? oui/non, et quelle est la conséquence de faire ou pas. Parce que ça en fait il y a plein d'équipes qui ne font pas de stand up, mais ils font une réunion de 2 heures par semaine. Qui finalement prend plus de temps que 4 ou 5 stand up et au bout duquel, au bout d'une demi-heure, j'ai perdu le focus et je n'écoute plus les autres. Tu vois remplacer ne veut pas nécessairement dire que c'est mieux ou que cela reste utile quoi.

P-E.D. : En fait, dans le côté adaptation, il y a un terme en agilité c'est Shu Ha Ri. Ce sont 3 mots que l'on utilise beaucoup dans les arts martiaux qui te donne en fait un processus d'évolution. On va dire, bien finalement tu as une première étape qui est le « Shu », c'est du karaté Kid, c'est tu ne poses pas la question, je te demande de faire cela et tu fais ça et c'est

tout. Le « Ha » c'est de dire : « Bien OK je fais cela, je continue à t'écouter, mais maintenant je comprends pourquoi je fais cela ». Et puis tu as seulement une dernière étape qui est le « Ri » qui est : ok j'ai compris pourquoi je le fais, je sais comment cela marche et donc je me permets de l'adapter. Tu as beaucoup d'équipes qui se disent je vais d'abord l'adapter en fait je n'ai pas compris pourquoi je le fais, pourquoi c'est nécessaire, mais je le change et du coup tu adaptes dans quelque chose qui n'a rien à voir.

N.D. : Puis tu vas avoir aussi le comportement naturel de l'immunité au changement. Tu viens avec des nouvelles méthodes et des noms, réunions, des nouveaux rôles. Et au final, les gens disent c'est du changement, donc je lutte et on trouve toutes les bonnes raisons pour ne pas faire selon le truc quoi. Avant c'était mieux, tu vois la rengaine habituelle. L'idée pour nous c'est de bien expliquer pourquoi on fait les choses d'une telle façon. Mais avant que tu as passé les phases du Shu et du Ha, tu ne permets pas de t'autoproclamer master de la méthode et de dire : « Oui nous on a compris, ça sert à cela, mais chez nous cela sert à rien. Par exemple, sans les nommer, mais chez un client, il n'y a quasi aucune équipe qui fait la sprint review. Tu pourrais la traduire par la démo de fin de sprint. Donc en gros, ils ne font jamais l'inspection ou l'adaptation du produit. Donc tu ne sais jamais en fait s'il est bon, s'il répond aux besoins de l'utilisateur. Le Product Owner, le client moi je trouve qu'il a loupé une bonne partie du feed-back qui fait qu'on fait de bons produits. C'est difficile de leur inculquer la culture du feed-back aussi, de la boucle rapide de feed-back.

*Où se trouvent les clients dans le Shu Ha Ri ?*

N.D. : Ici, chez Ingenico, par exemple il y a 12 équipes, il n'y a pas une qui est dans le « Ri », il y en a une qui en particulier en est pas loin en fait. Et vraiment dans cette phase, on a utilisé Scrum by the book, on a aménagé certaines parties. Et là je pense qu'ils sont en phase de mastering, c'est : ils maîtrisent quand même pas mal de trucs. Sinon pour moi il y en a qui sont à peine dans le Shu.

P-E.D. : Je pense que MyShopi, typiquement, ils avaient pas mal évolués aussi.

N.D. : L'avantage de MyShopi, au vu la taille, ce n'est pas forcément une équipe, c'est vraiment toute l'organisation qui a pas mal appris.

*Vos clients, ont-ils pris l'habitude d'évaluer le modèle mis en place?*

N.D. : Oui et donc souvent quand on nous appelle : « On voudrait mettre tel modèle en place » ou « Venez nous coacher sur Scrum », la première question que l'on pose c'est

pourquoi ? C'est quoi votre moteur de passer à Scrum ? Êtes-vous déjà Agile ? Et plus loin pourquoi ? Là on essaye d'identifier les espèces d'indicateurs qui font que si nous on vient on va actionner sur ces points-là. Il est important de pouvoir mesurer. Parfois c'est difficile de mesurer la confiance que l'on peut avoir dans les gens, les engagements des personnes, mais très clairement sur le nombre de fonctionnalités que l'on peut livrer par mois, par release, par sprint, oui cela on peut le voir facilement. Après, il y a des à-côtés, il y a eu une conférence de Laurence Vanhée sur quelques années d'expérimentation au service public fédéral de sécurité sociale. C'était le nombre de recrutements, le nombre de postulations pour le service public, le nombre de jours de maladie, le nombre de télétravail et les coûts en interne. Comme ils avaient mis en place aussi une transformation du cadre de travail avec du télétravail, du flexdesk, plus de bureau pour les managers, mais ils avaient aussi réduits drastiquement les coûts fixes de la société, ils louaient moins de bureaux donc tout cela peut être mis en place en fait. Cela va dépendre de ce qu'ils veulent mesurer en fait. Un principe simple c'est qu'on améliore ce qu'on mesure. Évidemment, si on ne mesure pas on ne sait pas améliorer. Si on s'améliore on ne le verra même pas donc il y en a qui le font et on accepte aussi qu'en cours de route il y ait des effets positifs ou même négatifs qui viennent s'inclurent dans les indicateurs, des choses auxquelles on ne s'attendait pas quoi.

*Quels indicateurs étaient importants pour vos clients ?*

N.D. : La rentabilité, la productivité, le nombre de fonctionnalités qui sont livrées dans le logiciel, dans le produit.

P-E.D. : La qualité, quand tu dis la productivité c'est vrai le temps mise sur le marché entre le moment où une idée va naître et qu'elle va être disponible cela va être un indicateur.

N.D. : Souvent un time-to-market. J'ai une idée, elle est en production. Combien de temps ? La qualité globale et il y a plein d'autres trucs, c'est plutôt des effets secondaires positifs que vraiment des indicateurs de base.

*Et ces indicateurs au moment de l'évaluation, sont-ils positifs dans tous les cas ?*

N.D. : Ah parfois cela ne marche pas.

*À quoi cela est-il dû ?*

N.D. : Comme je disais tout à l'heure, c'est limité au changement. Donc les gens ne jouent pas le jeu, le contexte changeant, la pression que l'on peut mettre sur la personne. Tu sais en gros : « on vous a fait le cadeau de mettre Scrum et payer un coach alors on y va quoi, on va

mettre plus de projets ». Ils n'ont pas compris le truc. C'est le côté un peu irréaliste qui parfois ne fonctionne pas.

P-E.D. : Parfois ce n'est pas les bonnes personnes non plus. Soit dans l'équipe, soit dans les personnes qui collaborent du côté business. Ce qu'on dit souvent en formation c'est que l'agilité ce n'est pas une méthode miracle. Ce n'est pas parce qu'on a une idée que cela va marcher, cela dépend du contexte, cela dépend des personnes et maintenant il y a des équipes où cela ne marchera pas.

N.D. : Et donc, comme il y a un feed-back rapide, comme Pierre-Emmanuel disait, cela ne résout pas les problèmes, cela les montre plus souvent. Donc, est-ce qu'on veut résoudre les problèmes avant de se dire on va faire mieux, ou alors on se dit, ça on le sait on le fera plus tard. C'est toujours une problématique, il y a une belle image qui résume, c'est des Lego : « No time to improve ». Il y a 2 gars qui poussent un chariot avec une roue carrée et il y a un à-côté qui a une roue ronde et qui dit : « je peux vous aider ? » Non, non. C'est un peu cela aussi.

*Le management, est-il content de l'évaluation ? Donne-t-il des feed-back positifs ?*

P-E.D. : Le management n'a pas toujours une position facile dans le domaine de l'agilité parce qu'on les remet en question en fait. Quand on met de l'agilité, on va transformer la hiérarchie. Donc, il n'y a pas mal de personnes, chefs de projets, qui disent : « Mais je suis où dans l'agilité ? » Il n'y a pas de case : « chef de projet » quand on regarde une équipe Scrum donc tu as effectivement des chefs de projets ou certaine partie du management qui sont assez inconfortables avec l'arrivée de l'agilité. Et effectivement, je pense que l'idée c'est d'amener de la transparence. Et donc les personnes qui vont passer au-delà de ça et qui vont se repositionner dans une équipe, elles vont voir au final des indicateurs qui sont plutôt positifs. Donc oui, cela va être bien accueilli.

N.D. : Puis il y a tous les côtés positifs, qui ressortent et qui ne sont pas forcément mesurables, comme le bien être dans l'entreprise. Je ne suis plus sous l'emprise d'un manager qui m'assigne complètement mes tâches, j'ai le droit de choisir. Je suis reconnu pour qui je suis dans mon intégrité totale. Donc il y a ces côtés-là qui apparaissent, qui n'étaient pas forcément voulus par la personne ou le groupe de personnes qui nous a mandaté. Maintenant, on est très clair aussi sur les dangers que cela peut poser. C'est du changement, il y en a qui n'en voudront pas et il y en a qui partiront et qui quitteront la société parce que cela ne leur va pas. Comme je disais, c'est un mouvement perpétuel, et il y a beaucoup de gens qui ne

supportent pas de vivre dans un monde qui change tout le temps. Dans le temps, dans certains services : « Ils ont encore décidé de changer tels trucs, tu te rends compte quoi ? ». Bien oui parce qu'avant cela ne marchait pas.

P-E.D. : Puis, tu as beaucoup de remises en question qui revient dans l'agilité. On parlait tout à l'heure du côté entreprise libérée mais derrière cela et derrière l'agilité, on donne une responsabilité aux personnes. Il y a des personnes qui n'en veulent pas. Se dire : « Si cela foire, c'est la faute du chef de projet », c'est très confortable. Eh bien non, si cela foire c'est de la faute à nous, en tant qu'équipe, et à moi aussi. C'est parfois une position que les gens ne veulent pas voir. Il y a des employés qui n'adhèrent pas à l'agilité pour cela.

*J'ai entendu dire que vous êtes une entreprise libérée, je me demandais : L'agilité mène à une entreprise libérée ou c'est plutôt l'inverse être une entreprise libérée mène à l'agilité ? Ou cela n'a pas de lien ?*

P-E.D. : J'aurais tendance à dire que ce qui me semble à peu près certain, c'est que dans tous les cas, l'agilité va avoir tendance à être sur le chemin de l'entreprise libérée.

On ne va pas forcément y aller parce que ce sont des étapes fort différentes, mais je pense que c'est une sorte d'évolution. Quand on va commencer à parler de responsabilité au sein d'une équipe, de transparence, on va avoir envie d'aller davantage dans cette direction-là. Et donc, quel est l'avantage d'aller dans cette direction-là, c'est qu'effectivement dans mon équipe et dans ma société je n'ai pas forcément un manager à qui je dois demander l'approbation, d'avoir envie de discuter du budget différemment, pas envie de prendre les décisions différemment donc c'est un cheminement qui va dans cette logique-là.

Après est-ce qu'une entreprise libérée va forcément dans le chemin de l'agilité ? Dans une certaine mesure dans une forme d'agilité oui.

N.D. : C'est une question de sémantique, ce que nous en entend par agilité : oui très clairement. Une entreprise libérée est Agile. Et je réfléchis à d'autres exemples entreprises dites libérées parce que c'est un peu pompeux de dire qu'on est une entreprise libérée.

P-E.D. : Surtout, on a la volonté de l'être

N.D. : C'est flatteur, mais ce n'est pas encore peut-être le cas. D'un autre côté, si pour moi je devais traduire l'agilité, autrement c'est participatif. Et dans ce cadre-là oui une entreprise libérée est très participative. Ce n'est pas un pharaon qui décide tout cela de sa pyramide, c'est des groupes de personnes qui ont un leadership sur l'entreprise. Mais du coup il n'y a

plus de niveau quoi. Il peut y avoir une hiérarchie dans les entreprises libérées parce que la hiérarchie est très forte à répéter des étapes, on verra plus une entreprise libérée comme un réseau, il y a des cellules qui sont responsables de certains trucs. Moi, personne physique, je peux faire partie de plusieurs réseaux. J'ai des compétences en agilité, en coaching, je vais les mettre au service de la cellule coaching, j'ai une compétence marketing, mais je vais aussi m'impliquer dans la cellule marketing.

*Qu'est que vous amélioriez chez vos clients pour l'instant ? Dans l'état où ils sont pour l'instant ?*

N.D. : En fait sur la dernière année ici, j'ai appris pas mal de choses sur les groupes de management. Et ce qui m'a marqué, il y a un modèle qui s'appelle les 5 dysfonctionnements d'une équipe et en fait c'est graduel si tu veux. Si tu cherches pourquoi une équipe, pas qu'une équipe de management, ne fonctionne pas, le premier niveau c'est le manque de confiance. Et très souvent tu peux juste arrêter là en fait. Parce que ça, cela va représenter la vision du truc. 80% des dysfonctionnements de l'équipe c'est le manque de confiance. Donc c'est peut-être cela en fait, le manque c'est montrer de la vulnérabilité, de transparence

P-E.D. : Et de revenir sur le côté humain en fait. Je pense que ce que tu vas voir de façon assez typique c'est qu'un manager n'a pas le droit d'être humain il est avant tout manager. Ce n'est pas chez un client, mais c'est un autre coach qui me partageait ça : il y a un jeu qu'on organise assez souvent, c'est ce qu'on appelle le marshmallow challenge, on joue avec des spaghettis, des marshmallows enfin bref on doit faire une construction c'est quelque chose d'assez connu. Et puis, le coach français a expliqué qu'il a mis cela dans une entreprise avec une dizaine de groupes qui jouent à cela et l'équipe de management a refusé de participer. Tu te rends compte s'ils étaient moins bons que les autres. Et ça tu trouves cela très souvent. Le management ne s'autorise pas d'être faible, d'avoir tort, de ne pas savoir. C'est donc beaucoup le côté humain, je pense.

*Je reviens sur les indicateurs de la méthodologie, y a-t-il eu certains indicateurs qui n'ont pas été atteints, mais le projet était tout de même un succès ?*

N.D. : C'est là où est la grosse difficulté. Si tu dessinais tout le flux entre la demande initiale et la mise en production, finalement, nous on est appelé là. Si cela marche ou pas, ce n'est pas le tout en fait. Donc du coup, si on ne se focalise qu'à un seul point de là à là, cela marche très bien. Mais en fait, après, cela s'est mal passé. Ou peut-être qu'avant ce n'était pas clair, on a renforcé le côté pas clair du téléphone arabe, en fait. Et puis finalement, quand l'utilisateur

reçoit le truc, dans les temps, dans le bon budget, c'est pas du tout ce qu'il a demandé. Tu vois la satisfaction client, ou certains indicateurs performants où tu peux livrer la fonctionnalité que l'utilisateur voulait et tout cela. Mais quand tu le livres, il clique sur un bouton et cela met 3 heures à venir quoi. Ça ce sont des trucs qui passent outre l'agilité. Mais là où tu vas voir si le modèle Agile fonctionne, c'est si on peut réagir rapidement à cela. Et aussi, surtout, est-ce qu'on a le droit à ce que ce genre de truc arrive. Donc tu vois, est-ce qu'on vire l'équipe, ou est-ce qu'on dit c'est votre responsabilité trouvez une solution. Pour moi l'agilité, c'est ça. Il y a un principe dans le modèle Spotify c'est « fast failure recovery over failure avoidance ». C'est finalement bâtir une structure qui permet de rapidement rebondir sur une erreur plutôt que de mettre tous les efforts dans : il ne faut pas que cela arrive. Parce que de toute façon, ça arrivera. Et quand cela arrivera, tu ne seras pas préparé parce que tu auras mis tellement d'effort à faire en sorte que cela n'arrive pas, que c'est inconcevable que cela arrive. Et donc tu as viré toute l'équipe, le manager, le responsable produit, en attendant l'erreur, elle est là quoi. C'est le genre de truc qui peut arriver. C'est ce qui me vient rapidement.

*Quel est votre avis concernant les modèles Agile en général selon votre expérience ? Qu'est-ce qui vous a peut-être le plus séduit ou justement interpellé par rapport à ces méthodes ?*

N.D. : Spontanément, ce qui me vient, c'est l'épaisseur du syllabus. Plus, il va être fin et plus ça va me plaire parce que ça donne beaucoup de place à ces étapes d'apprentissage et de customisation de la méthode. Tu prends Scrum c'est 21 pages, tu prends Prince 2 c'est 400 pages : là, tu te dis, j'essaye de répondre à toutes les questions en Prince 2. C'est des recettes de cuisine que Prince 2 propose, cela pose un cadre. Et donc en gros très facile à apprendre, mais difficile à maîtriser. Donc, c'est là que tu dois vraiment faire la différence entre le côté humain qui va s'approprier le truc et le côté déshumanisé qui va dire va voir à la page 602 le chapitre 4, la question est là et la réponse est là, c'est cette façon de faire. C'est cela qui me vient spontanément.

P-E.D. : Moi je pensais au côté humain, mais pas directement lié à la méthode, mais lié plutôt aux effets en fait. Pour moi, ce qu'on voit beaucoup dans l'agilité, c'est vraiment une séduction, le fait de se sentir bien, de se sentir mieux considéré, meilleur job.

*Vous utilisez également la méthode Agile, quelles améliorations vous apporteriez ici dans votre modèle ?*

N.D. : Le dernier sujet auquel on s'intéresse c'est cette notion de leadership. C'est qui est responsable de quoi sans pour autant dire, là, je suis manager de quelque chose. Il y a vraiment cette notion de responsabilité, et un cas concret, mais plutôt matérialisé c'est un Scrum Master pour nous est un leader dans le sens où il doit prendre la responsabilité pour ne pas laisser muer un problème. S'il y a un problème, il doit le dire, mais cela ne veut pas forcément dire qu'il doit le résoudre. Donc il va être attentif à ce que l'équipe soit informée qu'il y a un problème et stimuler l'équipe à trouver une solution. Plus que le modèle ancestral du manager qui, je vais vraiment élargir, grossir le truc, ils ne sont pas capables de s'améliorer, je vais trouver une solution et je vais leur imposer. Et c'est là finalement que tu perds l'engagement des équipes. Je n'ai qu'à appliquer ce que l'instance suprême a décidé pour moi. Ici, c'est un des problèmes que l'on vit actuellement, c'est Pyxis grandit, mais pas à une croissance folle. À 5 ans, on est 10 avec 3 freelances dont une huitième personne qui va arriver, ce n'est pas une croissance folle, mais on tient beaucoup au modèle qu'on tient à mettre en place. Mais le casting un peu difficile et peut parfois rendre inconfortable les personnes qui rentrent ici parce qu'on dit, mais tu n'es pas juste un employé tu fais partie de Pyxis et tu auras des responsabilités aussi. Ça, c'est ce que veut dire mon leadership.

P-E.D. : Et lié à cela, c'est ce que veut dire la notion de délégation. C'est dans quelle mesure nous on est confortable à déléguer, à notre pouvoir derrière ce leadership et d'accepter tout ce qui va derrière quoi. Donc c'est la combinaison des 2.

N.D. : Petite parenthèse, la confiance et la délégation. Autant la confiance dans un domaine précis peut être binaire, quand tu délègues ce n'est pas tout ou rien. Il y a un modèle management 3.0 qui définit sept niveaux de délégation entre je décide et cela se limite à cela en passant par je vous informe que j'ai décidé et je prends la décision avec vous, vous décidez vous m'informez, vous décidez et je ne suis même pas au courant. 2 extrêmes et tu passes par toutes les palettes des délégations et d'information.

*De quel modèle s'est inspiré votre modèle ?*

N.D. : En fait, on a des coach au Canada qui ont beaucoup travaillé à s'approprier des modèles, à réinventer de nouveaux modèles. En fait c'est un assaisonnement il n'y a pas un modèle qui est sacré, donc on va prendre la spirale dynamique, l'ennéagramme, la sociocratie, l'holocratie. Et ici, nous ce qu'on essaye de porter c'est de participer à l'agility qui est un modèle fractal, qui pour nous est intéressant. Parce qu'au sein de Pyxis il y a 3 cellules, 3 rôles qui sont : campus, conseil et studio. En gros, la formation, la consultance et le

développement qui fonctionnera chez Pyxis Belgique pour ces 3 modèles-là. Chez Pyxis Belgique en gros, au sein de l'organisation Pyxis. Donc tu vois c'est un modèle que tu peux zoomer, tu retrouveras toujours les mêmes formes et les mêmes termes. On va parler de Core Value Zone, de membrane et différentes étapes d'évolution dans une idée vers un produit. Tu vois il y a des cycles, donc c'est vraiment emprunt comme le management 3.0 est aussi un mélange de 100 bouquins de management bien ici notre modèle est issu de plein de modèles différents et il fonctionnera avec nous et peut-être pas avec d'autres. Moi ce que j'aime bien dans ce modèle-là c'est ce côté fractal. Tu peux zoomer tu retrouveras toujours la même forme.

*Continuez-vous à l'adapter régulièrement ?*

P-E.D. : Je n'en dirais peut-être pas à adapter le modèle. Je pense qu'on est plus actuellement dans une phase où nous nous adaptons au modèle. On l'appréhende, on le construit en fait, on est en train de se construire sur ce modèle, de l'utiliser dans notre construction.

N.D. : En même temps, comme on est dans le groupe qui construit le modèle, on donne notre feed-back aussi, comme une société qui utilise un modèle. C'est-à-dire les difficultés que l'on rencontre et nos coachs nous aident aussi : « Tiens, on a pensé à ça ». Ils nous donnent des grilles de lecture différentes sur le modèle.

*Au point de vue de l'évaluation, comment cela se passe ? C'est plutôt positif pour vous ou vous ne voyez pas encore spécialement les avantages ?*

N.D. : C'est un peu des 2 en fait. Le positif c'est qu'on a potentiellement un vocabulaire et quelque chose à se rattacher, plutôt que le n'importe quoi, l'émergeant, le day to day, donc c'est rassurant de ce côté-là. C'est un peu des deux.

*Connaissez-vous l'Agile Manifesto ? Quel rôle cela joue dans votre entreprise et dans les entreprises de vos clients ?*

P-E.D. : Je dirais c'est très clairement des valeurs que nous on va garder en tête et je pense que cela fait partie des premiers éléments que l'on présente au client quand on parle d'agilité. En fait, pour moi, c'est la plus simple et la plus courte définition de ce qu'est l'agilité qu'on présente au client. C'est un document important et on l'utilise beaucoup en formation aussi pour provoquer un peu. Comment vous sentez-vous par rapport à ça et le fait de comparer à gauche et à droite, c'est un bel outil pour provoquer la réflexion et la discussion.

*Avez-vous quelque chose à rajouter ?*

P-E.D. : Est-ce que tu t'intéresses au monde de l'agilité qui pourrait se transposer dans le monde de la famille ? Parce que tu as tout un volet de l'agilité au quotidien. De se dire, dans mon cercle familial, conjoint, enfant, organisation, vacances... Tu peux utiliser finalement les concepts de l'agilité, de Scrum, ou autre pour gérer cela. Tu retrouves beaucoup d'aspects humains parfois que tu vas mettre, on parle de confiance et transparence où tu vas peut-être mettre cela comme une grande valeur au travail et regarde finalement dans le quotidien ou la famille ce n'est pas nécessairement là.

N.D. : Le rapport entre parents et enfants, l'apprentissage à l'école et pas le droit à l'erreur. J'ai eu encore une discussion avec le Directeur de l'école de mes filles, où je lui parlais de Montessori : les classes inversées. Et d'apprendre aux enfants à être responsables, plutôt que juste télécharger du contenu toute la journée via le professeur. Et de leur apprendre à apprendre plutôt que de leur donner de la matière.

Au-delà dans le développement personnel, il y a un truc auquel je fais attention c'est les injonctions. Tu vois dans l'entreprise tu vas souvent entendre par exemple les managers qui ne peuvent pas être moins, c'est une injonction en fait. Tu es manager, déjà tu es pro donc tu vas pas jouer avec des spaghettis, car tu travailles et puis tu ne peux pas être moins bon que les autres.

P-E.D. : Et puis dans le monde de la famille, tu en as pleins, les parents qui vont dire à leurs enfants : « Mais arrête de pleurer, un grand garçon, ça ne pleure pas ; une grande fille, ça fait ses lacets, un grand garçon, il mange son assiette. » Donc si je veux être un homme, il faut que je lave mon assiette, il faut que je fasse mes lacets et je ne peux pas pleurer. Et donc tu vas construire un schéma mental et eux vont reproduire cela avec leurs enfants avec leurs amis, avec leurs collègues et c'est cela, tous ces ingrédients-là qu'on a dans l'entreprise, on ne distille pas nécessairement dans l'entreprise ou les personnes qui nous mandatent. Attention : on va voir le modèle par le développement du modèle, tu vois cela on ne fait pas. Mais quand on constate des situations comme cela, on l'explique et on parle avec la personne on lui propose, on ne lui impose pas. Donc ici, dernièrement c'est un gars avec qui j'ai travaillé, je rencontre, un de ses anciens collègues et qui me dit : « Ah machin, toujours en micro management ? ». Le micro management c'est vraiment gérer tâche par tâche, c'est tu ne donnes pas une mission à quelqu'un tu lui assignes ses tâches au quotidien. Et c'est marrant parce qu'au début quand on m'a présenté cette personne-là on me l'a quasi décrite comme cela. Et lui est persuadé que ce n'est pas vrai. Et je pense que la volonté qu'il a c'est être au service des gens donc de les aider. Juste, tu les étouffes toi en fait. Tu ne leur laisses pas

d'espace d'oxygène, pour qu'il se crée. Ce gars est drivé par les bullet-points, les tasks lists, il ne fait que ça. Et ici, dernièrement je lui donnais le feed-back que l'on m'a remonté. Mon temps est compté, je ne peux pas rester at vitam eternam dans cette boîte-ci, mais il faut que je te le dise et donc je propose des exercices d'attention. Tu vois comme tenir un journal et demander aux personnes pour voir comment ils interprètent cela, cette présence, etc. Voilà ce genre de truc qui n'est pas lié à Scrum, mine ou l'agilité dans le cadre d'un projet, plutôt lié aux personnes, les familles.

N.D. : Cela pourrait être une ouverture de gérer ton déménagement en Scrum.

P-E.D. : Ou plutôt que de dormir de 8 à 10 h, il y a des personnes qui dorment 4 cycles dans la journée. Réveil 2 heures de sommeil, réveil, 2 heures de sommeil. C'est un peu extrême, mais ça existe. On parle de sommeil polyphasique avec deux modèles : le modèle « uberman » (où ta journée ne sera coupé que par 6 siestes de moins de 30mn) et le modèle « everyman » (où ta journée va être coupée par une phase de sommeil de plus ou moins 4h, et entre 2 et 4 siestes supplémentaires)

## Appendix 16 : Interview Sze-Wai Leung

S.L. : Product Owner at Wolters Kluwer Transport Services.

*Pouvez-vous me parler un peu de vous et de votre entreprise ?*

S.L. : L'entreprise, je ne dirais pas que c'est une start-up parce qu'elle a 20 ans. C'est dans la software development, c'est un logiciel pour le secteur logistique. Wolters Kluwer, c'est une boîte à la base qui publie des publications légales. Mais notre département a été racheté par Wolters Kluwer pour faire des investissements et tout et c'est pour ça qu'ils ont emprunté Wolters Kluwer logistics services. À la fin justement ils remarquent que ça ne correspond pas du tout à leur stratégie et à leur marque non plus donc on nous a revendu. Tu peux garder ce nom, c'est le plus connu. Et au moment où on implémente ça, je dirais on est peut-être 100-120 personnes. Je ne suis pas sûre en tout cas ce n'est pas 50 et ce n'est pas 500. Donc on a une équipe, comme toutes les boîtes : on a marketing, on a finance, et cetera. Mais le but principal c'est la gestion de projet et aussi le côté informatique. Le côté informatique je dirais ça fait un tiers de l'entité de la boîte.

Et en fait, je rentre peut-être dans le détail, à la base je ne pense pas que c'est vraiment le directeur qui proposait Agile. En fait, souvent ce sont les informaticiens qui ont l'expérience avec Agile, c'est une autre façon de travailler. C'est plutôt les informaticiens qui ont une certaine expérience, manière de travailler et ils proposent ça. Et évidemment les directeurs qu'est-ce qu'ils retiennent ? Ils retiennent le côté voilà on va avoir des life-cycle, les time-to-deliver beaucoup plus courts. Donc ils pensaient qu'en implémentant Agile on peut faire des releases tous les mois, toutes les deux semaines. Parce que c'est ce qu'on propose dans Agile. C'est que voilà tu fais des moments de sprints, des cycles de sprints. Et puis, en fait après chaque moment de sprint tu peux délivrer quelque chose. Ça, c'est l'idée. Donc évidemment les chefs ils retiennent ça : « Ah on peut délivrer toutes les deux semaines, tous les mois, chouette ! » Ça, c'est une chose. Deuxième chose c'est que nous on souffre, on souffrait à l'époque, on doit attendre très longtemps pour insérer certaines fonctionnalités dans la feuille de route parce que si on a déjà planifié six mois de développement, on n'a pas des ressources extensibles non plus. Donc si le jour J, tu dois insérer quelque chose, on a déjà les choses remplies jusque dans six mois. Donc ça veut dire le plus tôt on peut commencer à travailler sur ce que tu demandes, c'est dans six mois. Donc en fait, ce qu'ils ont aussi retenus c'est que tu peux changer avec Agile après chaque sprint. Donc ça va être, toutes les deux semaines tu peux accepter ce que je demande. Donc en fait je pense que ce sont ces deux choses qu'ils retiennent vraiment. Donc chez nous, à l'époque, parce qu'Agile, enfin si tu connais bien, ce

n'est pas une méthode, c'est un mind-set. On a plein de méthodes qui peuvent implémenter Agile : on a Scrum, on a Kanban, peu importe... Mais le côté Agile ça commence déjà dans notre façon de penser et ça peut se propager jusqu'au client, jusqu'au contrat avec un client. Tu peux signer un contrat avec le client en proposant Agile. Mais ça on est pas du tout là. Donc voilà juste pour mettre en contexte même les directeurs ne sont pas Agiles. Il y a que les développeurs qui sont Agiles. Mais je pense que c'est dans 80 % des cas, sauf les boîtes qui ont un budget Agile, qui engagent des change managers, des Scrum Masters, des Agiles Masters pour gérer ça. Mais sinon dans 80, voire 90 % des cas c'est vraiment entre les informaticiens à la limite avec moi qui suis plus dans le côté business qui est plutôt Agile. Mais avant ça, on était pas du tout Agile.

*En ce qui vous concerne, quelles études avez-vous faites et quelle est votre expérience ?*

S.L. : Je suis diplômée en sciences cognitives, après j'ai fait un MBA chez Solvay et après en Master en supply chain management. Et après, donc juste pour que tu saches, les brevets en sciences cognitives ça ne mène nulle part. On ne trouve pas de boulot, je ne suis spécialisée en rien. À ce moment-là, je suis venue en Belgique dans le seul domaine où on peut parler en anglais, c'est informatique. C'est pour ça que j'ai étudié pendant quelques semaines en C++ pour dire que je savais programmer, je savais quand même faire un peu de programmation. Et c'est comme ça que je suis entrée dans le milieu informatique. Et après comme j'ai fait le MBA je ne voulais plus rester dans l'informatique. C'est comme ça que je suis passée plus du côté business et donc mon rôle de Product Owner dans SCRUM.

*Donc vous êtes SCRUM Master ou Product Owner ?*

S.L. : Non, je suis Product Owner et on n'est pas vraiment Scrum Master, parce que quand on est Scrum Master pour les sociétés qui regardent vraiment près dans le budget, c'est le rôle un peu de médiateur, facilitateur. Il ne produit rien, donc voilà. Il n'y a pas Productivité donc évidemment les rôles comme ça on enlève. Et du coup, bien, c'est la bonne volonté de bien mener les choses. C'est comme ça que l'on fait dans notre système, normalement tu implémentes ton business, on a un Product Owner, on a un project développeur. En fait, dans notre système c'est le business Product Owner et le technical Product Owner qui font en sorte qu'il y a une très bonne communication et on fait aussi le rôle de Scrum Master. Par contre, quand on commence à avoir plus que 5,6 groupes comme ça, on a engagé quelqu'un qui fait plutôt change management et c'est un background de Scrum Master. Donc lui, il ne va pas faire Scrum Master de chaque Scrum, mais il a lancé le Scrum of Scrum.

*Et donc toi en tant que Product Owner c'est quoi ton job, tes responsabilités ?*

S.L: En tant que Product Owner je gère la feuille de route, je gère la priorité. Strictement parlant je ne dois pas faire le planning. Dans Scrum c'est moi qui fais une wish list, c'est moi qui met la priorité des choses et puis c'est le technical Owner, le technical lead qui prend selon priorité, et qui met des choses dans chaque sprint. Mais moi je suis assez contrôle-freak donc c'est vrai que c'est moi qui le fais. Et une chose très importante c'est que moi je pense, ce n'est pas pour frimer, je suis un très bon Product Owner. Donc il y a 5-6 Product Owner et il y en a un autre qui fait très mal. Ce qui est vraiment mauvais c'est qu'on ne gère pas bien la priorité. Parce qu'en fait Agile c'est vraiment un mind-set. En fait Agile, on doit d'abord certifier le client avant tout, deuxième chose c'est la communication entre nous, troisième chose c'est l'adaptation. On s'adapte, on n'a pas peur de s'adapter. Donc moi c'est presque un boulot, je ne dirais pas tous les jours, mais le mind-set c'est vraiment tous les jours. Je change les priorités des choses une fois par semaine. Du coup on fait des sprints de deux semaines en fonction de ce qui est livré, en fonction du temps qu'il reste parce que des accidents arrivent, des surprises arrivent. Il y a certaines choses, qui peuvent prendre plus de temps. Et j'ajuste vraiment au niveau du contenu des choses pour, à la fin, qu'on livre toujours ce qu'on a promis au client. On ne perd pas dans les fonctionnalités, mais on peut perdre dans les détails. Et ça, c'est vraiment le rôle de Product Owner afin de garder le cap pour savoir : « ça je t'ai promis de livrer une maison, tu auras une maison. Dans le contrat, on a parlé d'avoir une cheminée, une fenêtre, une porte. Voilà tu auras. Peut-être au début on avait dit qu'il fallait sculpter et tout, mais à la fin je livre une porte qui n'est pas sculptée. Je livre donc une porte sculptable, mais non sculptée. » Voilà, je parle en métaphore. Si tu prends un menuisier, il fait la porte et commence déjà à sculpter et moi je vais lui dire tout de suite : « Non, non, non, au lieu de sculpter ça, va directement faire les fenêtres, parce qu'il nous reste deux semaines avant de livrer et il faut faire les deux fenêtres. » Je n'ai vraiment pas peur de mettre la priorité et de les préparer vraiment. Expliquer vraiment ce qui est le plus important.

Donc à la fin, avant quand on travaillait avec waterfall, c'était beaucoup beaucoup de boulot, je devais commenter absolument tout. Et comme on travaille avec les gens du Kazakhstan et Ukraine et Belarus, ils ne comprenaient pas le business. Et si je vais leur dire il faut faire un dot en moins ils vont faire un dot en moins. Parce qu'ils ne comprennent pas le business. Et avec le système de Scrum, la communication se passe super bien parce que je suis un peu une maman pour eux. Et en fait, j'explique tellement bien au technical Owner que le jour où il veut changer de bord ou faire commercial il est tout à fait capable. Parce que je transfère

tellement de choses à lui et aussi au développeur. Ça veut dire que si moi je suis absente pendant deux semaines, ils peuvent vraiment prendre les décisions parce que dès le départ, dès le kick-off j'explique ce qui est le plus important. Ils savent très très bien la priorité. Ils savent très très bien et en plus ils savent ce que le client pense. C'est pour ça qu'au début, pour tout le monde, un changement, ça fait peur, en plus c'est un changement qui vient d'un informaticien ça fait peur. Et maintenant, ça fait deux ans que je fais ça je ne peux plus changer, ou travailler autrement, ce n'est pas possible. Et en plus avant, toute la pression pose sur toi tout seul.

Tu proposais des choses tu faisais des analyses, etc. Comme je dis, quand tu oubliais une chose, tout était ta faute. Et personne ne pouvait t'aider avant. Et maintenant avec le système d'Agile et Scrum, ce qui est vraiment bien c'est qu'en quelque sorte je donne juste les guidelines, je ne donne jamais de détails, c'est toute l'équipe qui contribue. Mais c'est ça la mentalité d'Agile en fait. C'est toute l'équipe qui construit une chose ensemble.

Donc je fais uniquement les grosses boîtes, l'idée de grosses boîtes c'est que chacun peut les remplir. Et toutes les deux semaines j'insiste d'avoir une démo qui viennent d'eux et moi je suis très très attentive et ils se sentent très valorisés avec leurs travaillent. Et c'est là qu'on peut faire des ajustements. Ah là on a un peu de déviation. Ça se passe super bien.

Ceux chez qui ça ne se passe super bien, ils se croient supérieurs aux autres et ne travaillent pas avec l'équipe. Donc lui, il donne le message à une personne, le chef des développeurs, il a juste un cerveau, il a aussi d'autres choses en tête. Il passe comme ça et ne participe jamais aux sprints, ni aux démos, il s'en fiche en fait.

D'ailleurs j'ai vu plein de fois quand on fait des démos, il met en mute son Skype et il fait autre chose. Ce n'est pas un travail d'équipe.

Pour que l'Agile et le Scrum fonctionnent il faut un respect, il faut tout partager. Et aussi, c'est – je ne sais pas si c'est une question de supériorité - en tout cas il donne toute l'info au technical lead et il a fait sa priorité et il ne la change plus. Il a plus ou moins fait sa priorité et il laisse aux gars, au technique qu'ils se débrouillent. Il le blâme au lieu de travailler avec lui, de trouver une solution ensemble.

Ce qui est vraiment frappant, c'est qu'après on a eu un changement de rôle, en fait c'est le technical lead, le groupe qui ne travaille pas bien, pour moi c'est le Product Owner qui ne travaille pas bien. Et le technical lead, il a bougé, il travaille pour moi. Et mon technical lead, il travaille pour lui. Au début il était très défensif, il a eu peur. Je ne peux pas accepter.

En général, je ne challenge pas. Si on me dit ça prend deux mois ça prend deux mois. Si ça prend deux semaines, ça prend deux semaines. C'est le respect je trouve. Au début justement il prenait beaucoup trop de temps. Moi j'ai fait de la programmation, je sais très bien combien de temps ça prend. Ça prend deux jours, il me dit deux semaines. Il faut discuter quoi. Au début, j'ai dû vraiment le rassurer. On est vraiment en équipe, on sait toujours livrer des choses ensemble. Avec l'autre gars, bien j'ai eu plein de soucis, même tout à coup en Production on a des problèmes, on a perdu une semaine. Il faut tout rattraper. Il ne comprend pas, au début parce qu'il est mal éduqué par l'autre. Ça m'a pris vraiment 3-4 mois pour le rassurer, pour vraiment lui expliquer, je donne la priorité des choses, j'enlève carrément la priorité du scope si ce n'est pas possible. Et j'ai regagné sa confiance, on commence à travailler ensemble et puis après je suis partie.

*Donc tu m'as dit qu'avant vous utilisiez le waterfall ? Qu'est-ce qui a fait que vous avez été vers l'Agile ?*

S.L. Moi non, moi comme j'ai dit ce sont les développeurs qui ont fait ça. Sûrement il y a des architectes et tout ça qui ont travaillé dans l'Agile. En général, ce sont ceux qui aiment travailler en open source.

*Est-ce que ce sont les nouveaux arrivants dans l'entreprise qui ont donné l'idée ?*

S.L. Oui quand même. Pas forcément des gens nouveaux, mais on voit bien que ce sont des profils qui sont plus orientés open source, plus Agiles. Au début, ils font ce qu'on leur demande et puis après ils voient bien qu'on laisse plus de libertés et ils voient bien qu'on leur a lancé des trucs comme ça quoi. Et comme je dis, bien pour convaincre les directeurs ce n'est pas difficile parce qu'ils retiennent simplement qu'on peut livrer plus souvent.

*Ils retiennent les avantages et c'est parti ?*

S.L. Oui voilà c'est ça.

Attends que je réfléchisse. Pour commencer, ça c'est si tu lis un peu la littérature, c'est le problème dans beaucoup de boîtes, ils utilisent les termes Scrum, ils font des sprints, ils font des daily stand up... Ils utilisent tous les termes de Scrum et ils pensent qu'ils sont Agiles. Mais je pense que toutes les petites boîtes commencent toujours comme ça. Surtout quand ils essaient de se débrouiller. On a commencé par la forme plutôt que par l'esprit. Quand on engage quelqu'un on n'engage pas forcément quelqu'un qui connaît ce framework.

*Et comment avez-vous implémenté les méthodes ? Vous aviez simplement un guide à suivre, vous avez reçu des formations, ... ?*

S.L. Non, pas de formations, pas du tout. Les formations c'était quand j'allais partir. On a donné une formation à toutes les équipes produits, donc côté business et tous les informaticiens.

Et ce qu'on trouve très dommage, c'est que même les directeurs auraient du suivre les formations Scrum. Pourquoi ? Parce qu'on se retrouve un peu au milieu sans parler, ils retiennent que les avantages, mais ils ne respectent pas notre façon de travailler Scrum.

Déjà, il faut accepter qu'Agile, le triangle temps, ressource, scope, le triangle dans un cas classique, on fixe le temps, on fixe le contenu et c'est aux fournisseurs qui gèrent les ressources. S'ils disent deux personnes, on fait deux personnes, s'ils disent quatre personnes on met 4 personnes pour le faire. C'est comme quand tu rénoves une maison, mais tu dis, je veux que ça finisse fin août parce que j'ai une réception et je paye ce prix. Et mon contrat c'est 10000 euros. Et alors c'est l'entrepreneur qui doit mettre deux personnes, 3 personnes, ou 4 personnes. Ça c'est la façon de faire classique. Mais dans la façon Agile, on fixe l'équipe, on fixe éventuellement le temps parce que le commitment avec le client c'est quand même important.

S'il dit de délivrer quelque chose en août, on doit le faire. Mais on ne fixe pas le contenu. C'est comme ça que je dis qu'on peut mettre une priorité aux choses, on peut fixer les choses.

Ce que les producteurs et patrons ne comprennent pas forcément. Quand c'est comme ça, c'est le contenu qu'on peut changer. Donc eux, ils voient le côté on peut changer on peut ajouter. Mais ils ne voient pas le côté on peut enlever les choses.

La formation qu'on a reçue, est importante. Ce qu'on a reçu, chacun le connaissait déjà plus ou moins. Mais si tout le monde reçoit la même formation, on a le même langage. C'est parce il y a des gens qui connaissent plus, d'autres moins, il y a des gens qui s'en fichaient, il y a des gens qui y tiennent plus. Imaginons, moi j'y tiens beaucoup, on peut pas faire comme ça, on ne peut pas faire comme ça, on doit faire comme ça. Bien les autres, ils réagissent en disant : « mais qu'est-ce que tu dis, pourquoi tu fais ça ? ». Et maintenant que tout le monde a suivi la même formation tu as plus de poids quand tu dis, tu veux simplement exercer ce que tu dois faire et tu as plus de poids. « Parce que c'est normal, on est Agile ou pas ? Oui ? Alors on fait comme ça, c'est normal. »

*Et pourquoi croyez-vous que les formations n'ont pas été mises directement en place ? Souci de budget ? On croyait que c'était inutile... ?*

S.L. On pensait que ça allait fonctionner. Parce que, encore une fois, comme beaucoup d'erreurs d'implémenter Agile, même les patrons quand ils pensent, ils mettent la forme de Scrum, on est Agile. Mais ce n'est pas du tout ça.

On a fait des sprints, et tout. On utilise tous les termes, on utilise les squads, on a utilisé le nom, mais ce n'est pas pour ça que le Product Owner sait ce qu'il doit faire.

Comme j'ai dit, quand on a engagé quelqu'un, on n'a pas cherché quelqu'un de certifié. Même si, même certifié, je ne suis pas sûre que les gens sachent ce qu'il faut faire. Mais les gens qui à la limite ne connaissaient pas du tout, ou qui ont de l'expérience et qui suivent la moitié de ce qui est dans Agile donc... On fait un peu n'importe comment.

Je pense que le trigger qui dit qu'on doit avoir des cours, c'est quand on commence à avoir des ping-pongs, des disputes : « ah, ce n'est pas moi, c'est toi qui dois faire », et l'autre il dit : « ah, mais non ». Et la communication ne passe plus parce qu'on ne prend pas la responsabilité correctement.

*Et c'était surtout des problèmes de communication que vous aviez quand vous avez implémenté Agile ? Ou est-ce qu'il y avait d'autres problèmes ?*

S.L. Mais en fait, au niveau exécution, je pense que ça se passe plutôt bien.

Mais je pense que c'est vraiment le problème d'esprit qui n'est pas là. En revoyant en arrière, je ne comprends même pas, il faut qu'on me mette dans une forme Scrum pour que je devienne Agile. Je ne comprends pas. Et je pense que c'est parce que de moi naturellement je communique très bien, je ne garde aucun secret. Je sais que dans la boîte il y a des personnes qui disent carrément que : « plus je garde un secret, plus je suis indispensable, plus je suis important ». Je ne parle même pas de ça, je ne suis même pas quelqu'un comme ça. Mais il faut, c'est pour ça que la forme de Scrum, je pense qu'elle sert quand même à quelque chose.

Je ne sais pas expliquer. Naturellement, mon esprit est correct. Mais sans la forme de Scrum, ce n'est pas... ça n'irait pas. Plus la forme de Scrum est Agile plus ça marche bien, ça marche très bien. Et dans l'autre cas qui ne fonctionne pas bien, la forme est là, mais tout le planning... On suit strictement. Mais l'esprit n'est pas là.

En fait, ça ne sert absolument à rien, ça pose encore plus de problèmes qu'avant. Parce que les développeurs ils, parce que le problème avec Scrum, Agile et tout ça on monitor la vélocité

des développeurs, et ça devient presque une lame pour se faire blâmer. En fait c'est pire. J'ai entendu aussi les autres boîtes qui n'utilisent pas Scrum, ils utilisent Kanban, et ça passe très mal. Et en discutant, et c'est chaque fois... En fait Kanban, encore une fois sans esprit, ça ne marche pas. Je ne sais pas exactement ce que tu veux arriver comme conclusion, conseils machins.

*C'est un de savoir comment ce sont passés les objectifs ?*

S.L. L'esprit c'est le plus important. Mais c'est maintenant que je réalise que sans la forme, pourquoi on propose toute cette forme. Au final c'est vrai que sans la forme tu as beau avec cet esprit, mais... Je pense que la forme fait que tous les jours il faut se voir, c'est tout le monde développeurs qui... Oui voilà je sais pourquoi. Comme l'esprit Agile c'est adapter les changements, il faut livrer des petits morceaux pour pouvoir accepter le changement. Donc du coup la forme de Scrum qui insiste sur le cycle de sprints, qu'il faut livrer, c'est très important.

En fait avant c'était des projets qu'il fallait livrer dans 3 mois, 4 mois, 5 mois, 6 mois, mais à la fin ça dévie tellement fort que ça pose problème. Oui en fait c'est ça. L'esprit de toute manière, je suis certaine c'est le plus important.

*Pour être sûre d'avoir bien compris vous avez juste mis en place Scrum ou vous avez mis encore d'autres méthodes en place qui sont Agiles.*

S.L. En fait, dans certains cas on fait Kanban.

*Ça dépend de quoi ? Pourquoi et dans quel contexte vous mettez Kanban ?*

S.L. Ça dépend de la structure de l'équipe je pense. Donc c'est clair, les supports clientèle, il fait un peu Kanban donc, le premier téléphone, le premier répond.

*Et donc si vous commencez avec une certaine équipe, mais pour x ou y raison vous devez changer d'équipe c'est par exemple ça ?*

S.L. Quand je parlais, on était en train de mettre en place les squats, les différentes équipes qu'on a essayé de séparer en deux squats interchangeables.

*Et du coup qui changeraient par rapport à ça ?*

S.L. Je pense que ça marche, parce qu'en fait je ne sais pas parler d'expérience. En théorie ça marche. Parce qu'évidemment quand tu réévalues, la vélocité de l'équipe diminue, mais c'est le Product Owner, le Scrum Master qui mitige ce risque quand le changement d'équipe

intervient. Quand le moment est de 3 jours et qu'il faut mettre 5 jours, il doit y avoir une raison de changer. C'est pour moins de risques. Il y a des gens qui s'en font. Il faut quand même expliquer le pourquoi nous faisons ça. En connaissant, on ne discute pas sur la forme. Encore une fois on ne va pas tout changer. Maintenant il faut changer.

*Et vous à partir d'un moment est-ce que vous avez changé ? Est-ce qu'à partir d'un moment il y avait des choses qui n'allaient pas et vous avez changé des choses ?*

S.L. On n'a pas fait ça parce que la fluctuation de la quantité de travail ne survient pas en même temps. Parce qu'on imagine, ça c'est mon équipe, ça c'est ton équipe qui fonctionnait. Et tout à coup, c'est surtout ça en fait, c'est parce qu'il gère mal en fait. Tout à coup, il n'arrive plus à gérer, il ne s'en sort plus parce qu'il ne priorise pas, il peut voler des gens chez moi. L'idée c'était pour ça. Et aussi pour, comme ça aussi à la fin il peut grouper les deux et comme ça c'est un grand groupe malgré tout, il y a des spécialisations. Et pour au niveau de structure, être plus Agile.

*Donc ça c'est au début de l'implémentation ou à la fin ?*

S.L. Non ça c'était à la fin, c'était quand je parlais. Ça faisait déjà 5,6 ans qu'on avait Agile. C'est depuis 1 an.

*Vous avez donc trouvé un problème ?*

S.L. C'est planifié, il y a beaucoup de planification parce qu'il faut des transferts de compétences, il y a vraiment beaucoup de préparation avant de faire ce transfert de compétences.

Et en fait les gens, ils ont peur, mais ils sont contents en même temps. Ils ont peur parce que voilà, le changement ça fait toujours peur. Mais à la fin ils sont contents parce qu'ils voient qu'à la fin l'entreprise, on est tellement main dans la main, on est tellement lié. Parce qu'avant c'était presque un handicap, eux ils ne connaissent pas ce que les autres faisaient. En mergeant les deux équipes, en fait, ils voient : « oh, mais en fait quand je fais ça, il y a un impact là-dessus, et c'est pour ça que tu fais comme ça ! ». Mais encore une fois est-ce qu'il faut vraiment merger les équipes pour avoir ce résultat ? Peut-être pas. Si on a une bonne communication, si on a de bonnes formations régulières entre équipes ça peut aller.

*Quand vous avez mis en place Agile il y avait les deux principales attentes des directeurs ou il y avait encore d'autres attentes ? Est-ce qu'il y avait des*

*attentes particulières avec cette implémentation ? Est-ce que vous pensiez que vous alliez avoir un meilleur processus pour réussir les projets ? Ou que vous alliez avoir la possibilité d'avoir l'occasion de plus changer ?*

S.L. En fait, on ne croyait pas. Maintenant , que j'y pense, on a reçu une formation : « comment faire le poker planning ». Tu connais le poker planning ?

*Non*

S.L. Le pockup planning c'est avec, la forme c'est un jeu de carte, mais les cartes tu as zéro, un, c'est des chiffres de Fibonacci : 1,1, 2, 3, 8, 16, 24 après je ne sais pas donc voilà. Enfin c'est une technique pour faire estimer. Estimer, on ne parle pas de combien de jours j'ai besoin pour faire ça. Estimer c'est, je pense que la difficulté de ce task c'est numéro 8. Et si c'est un truc encore plus gros alors je dis 16. Si là, c'est beaucoup beaucoup plus petit, là je dis un. Au fur et à mesure tu mesures un patel. À chaque fois que tu estimes c'est un, ça prend trois heures de le faire, non ça prend 4 heures de le faire. Chaque fois qu'il estime 8, ça prend euh 4 jours pour le faire. On monitor en fait c'est ça la Velocity release après release, pour établir la vitesse de l'équipe. Le principal d'un Agile et Scrum qui fonctionnent bien c'est que l'équipe ne change pas. Et nous on utilise, on est déjà bien dans le sens que les personnes clés, le technical lead et le Product Owner il est chez nous, il n'est pas au Kazakhstan. Où je travaille maintenant il n'est pas du tout Agile, mais ce qui est pire de ce que je vois c'est que même ces positions clés on les mets à l'île Maurice.

Et ça c'est très grave, c'est une erreur. L'important est qu'il faut avoir des personnes clés dans l'entreprise. Et après les personnes clés tu peux les out sourcer au Kazakhstan ou... Pourquoi je parlais de ça ? Ah oui, parce que les développeurs on utiliser une entreprise qui fait des toutes boîtes et tout. Ces boîtes elles sont bien et tout. Ils ont des bons développeurs, pas chers et tout. Mais le problème c'est qu'il y a une rotation assez forte. Parce qu'eux, ce sont des jeunes, qui ont fait deux ans là-bas et sont contents d'aller ailleurs. En plus, ils ont l'expérience international, ils peuvent aller à l'étranger comme ça. Ils aiment bien. Et ... quand l'équipe n'est pas stable ça ne fonctionne pas du tout. Et donc nous on attend quoi ? On a reçu une formation poker planning je me souviens très bien on en rigole, on n'y croyait pas du tout.

*Mais vous n'aviez pas du tout d'attente particulière alors ?*

S.L. Encore une fois c'est un type ou deux dans la boîte qui connaissent ce genre de choses qui poussent. Ils ont convaincu... Mais je pense que les développeurs ils sont plus faciles à

accepter ça parce qu'eux ils voient l'intérêt d'avoir tous les jours un stand-up de cinq minutes. Ils voient que toutes les deux semaines leurs meeting. Pour eux, ça ne change pas vraiment, ils ont leur travail. Ce qui change vraiment c'est les analystes, c'est le Product Owner comme moi. Et là, au début, je suis très méfiante parce que je me dis ce n'est pas possible. Je dis, maintenant si j'oublie un dot, on va faire un dot en moins. Même si c'est un visage avec deux yeux, un nez, une bouche. Si moi dans la documentation je fais un œil, ils vont faire un œil parce que j'aie oublié l'autre. Et tu me dis maintenant je ne dois rien faire et je dois simplement faire un visage ? Mais eux, ils vont faire ça pour moi ? Je n'y crois pas du tout. Je veux bien essayer, mais c'est vraiment... Encore une fois l'esprit, l'attitude est vraiment très importante.

Au début c'est très dur parce qu'ils ne comprennent rien, ils ont l'expérience sur ça. Et je dois dire voilà je dois faire un visage, donc c'est combien le pick-up planning, c'est combien ? Je ne sais pas, je ne sais pas. Encore une fois je partage beaucoup. Je ne dois plus dessiner le visage, je dis implémente le visage. Par contre, je dois dire : bien cette fois-ci je dois faire un visage, l'important de pourquoi faire un visage c'est parce que en fait c'est un masque pour les enfants de la crèche pour le carnaval, et en fait en donnant ça eux ils comprennent : ah le visage doit être de cette taille-là parce que c'est pour les enfants, les yeux ça doit être plus rapproché parce qu'encore une fois c'est pour les enfants. Il ne faut pas faire un masque qui fait peur parce que c'est les enfants. En fait, ça change complètement. On est beaucoup plus proactifs, tout le monde participe plus en fin de la chaîne.

Mon équipe marche très bien, car je mets les testeurs à l'honneur. Alors que d'habitude en informatique ce sont les testeurs à la fin. C'est en général parce que tu n'es pas assez bon pour être développeur alors tu es testeur. Moi je fais dans l'autre sens. C'est eux qui doivent comprendre ce que le client attend de nous.

Je partage énormément avec les testeurs dans les meetings et ils ont beaucoup de choses à dire, par exemple : « Ah est ce que tu as pensé, tu as parlé d'un masque tu n'as pas pensé la ficelle pour attacher le masque, comment on va attacher ? » Ah bien mince c'est vrai, alors on va ajouter une tâche qu'il faut ajouter. Alors là comme on a ajouté un truc il faut évidemment que j'enlève quelque chose. Je dois enlever le truc pour mettre du rouge à lèvres sur la bouche par exemple parce qu'on n'a pas assez de temps. Celui qui ne fonctionne pas bien, il va dire qu'il a oublié la ficelle, ah il va la rajouter, mais il n'enlève pas quelque chose à la place.

*Comment vous avez évalué l'implémentation ?*

S.L. On n'a pas évalué.

*Les directeurs n'ont pas jugé utile d'évaluer ?*

S.L. Les directeurs ? Évidemment que non. On n'a pas engagé une personne spéciale pour implémenter ça, c'est quelqu'un qui sort de sa bonne volonté : on va faire ça, ça va être bien. C'est comme ça qu'on fait les choses, de la manière la plus pragmatique possible. C'est comme ça, on suit la forme, on fait des plannings, on fait des Daily Scrummer, on fait les sprints point. On n'évalue pas.

*Donc vous avez mis quelque chose en place, mais au final vous ne savez pas si ça a apporté du bon ?*

S.L. Non, maintenant est-ce que c'est important ? Oui peut-être, si on fait ça, ça peut aider l'autre équipe qui fonctionne mal.

*Est-ce qu'il faut avoir certains objectifs pour implémenter Agile selon toi ?*

S.L. Non, l'objectif c'est d'être meilleur. Par contre, il ne faut pas faire ça dans son coin, il faut amener tout le monde, avoir, pas un consensus, mais une compréhension commune sur Agile.

C'est comme je dis, à la fin on a eu une formation et ça a servi à quelque chose vraiment. Mais cette formation ne doit pas rester uniquement dans les gens qui travaillent, mais autour aussi, les directeurs.

On ne peut pas dire jusqu'au client, le client n'est pas forcément prêt à accepter ce genre de travail, mais selon mon expérience et celle du formateur, en fait les clients, ils sont toujours contents, toujours. Parce qu'au lieu d'attendre 4 mois, 6 mois que tu livres quelque chose, tous les, je ne dis pas toutes les deux semaines, mais tous les mois on livre quelque chose.

Par contre, il faut qu'ils comprennent vraiment c'est quoi Agile. Quand il voit que quelque chose ne va pas, ils corrigent, OK ils corrigent, mais on priorise toutes les demandes ensemble et il faut pouvoir enlever quelque chose. Si tu n'enlèves pas alors ça veut dire qu'on va prolonger. Mais ça c'est une étape plus loin, c'est plutôt les patrons, les penseurs de l'entreprise qui estiment si c'est un avantage compétitif pour la boîte d'être comme ça. Je pense qu'il y a certains clients qui s'engagent parce qu'ils ont cet esprit. Mais pour amener les clients, c'est peut-être une autre étape, mais c'est très important d'amener les patrons avec nous.

*Si tu devais donner une définition de l'Agile mind-set ce serait quoi par rapport à ton expérience ?*

S.L. Agile c'est vraiment une bonne communication, le respect entre chaque membre de l'équipe pour accepter, non seulement accepter, c'est encore plus, en anglais c'est embrace. Plus qu'accepter c'est on est contents d'avoir des changements et on fait tout pour. Parce que à la fin, pourquoi on accepte-t-on le changement ? C'est parce que si on change, c'est pour le bien. Donc ça veut dire on va rendre les clients plus heureux.

*Tu m'as parlé de l'Agile mind, est-ce que tu connais l'Agile manifesto ?*

S.L. Oui

*Est-ce que ça a joué un rôle dans l'entreprise ?*

S.L. Non

*Et avec les formations, ça a aidé les gens de l'entreprise ? Ou pas spécialement ?*

S.L. Malheureusement, au début, quand on implémente Agile, Scrum tout ça, c'est plus une contrainte. Dans le sens qu'il y a les technical leads, qui disent ah on est au milieu de Scrum, il n'y a pas de changement parce qu'on change tous les sprints.

Ça, on peut changer, mais au sein des sprints, on ne change pas. Ça à mon avis, c'est pour ne pas déranger la vitesse de l'équipe. Tous les deux sprints, on mesure la vitesse. Donc, si tu mets autre chose ce n'est plus bon. Donc au début c'est plus une contrainte.

Encore une fois au début on met la forme, avant de mettre l'esprit. Même si moi j'ai cet esprit, mais j'ai appris aussi à avoir cet esprit, je n'ai pas eu l'esprit tout de suite. Au début, comme on a mis la forme, ce n'est pas l'Agile manifesto dont on s'occupe, c'est d'être sûrs que toutes les deux semaines, on a bien réfléchi pour éviter les changements.

C'est plus, toutes les règles à suivre au départ. Mais ça, c'est plutôt Scrum, ce n'est pas Agile. L'esprit malheureusement, je pense que ça vient de la personne. L'erreur c'est qu'on n'a pas engagé quelqu'un. On a sous-estimé l'utilité de trouver des gens qui ont cet esprit. Et pour moi finalement c'est encore plus important.

On a certaines formes qui facilitent, on a certains outils qui facilitent l'implémentation de Scrum. Ça peut bien se passer. Mais quand tu as toutes les formes, mais pas l'esprit ça ne va pas du tout.

*Donc si tu pouvais faire une conclusion sur l'Agile ce serait quoi ?*

S.L. Je ne peux plus lâcher ça. Je ne peux plus travailler dans le sens inverse. En fait, on devient paresseux. Je rentre beaucoup moins dans les détails. Quand je réfléchis à un truc, je dis j'ai besoin d'un masque pour les enfants. En fait, à la limite on rentre plus dans l'essentiel des choses. Parce qu'on dit exactement ce que le client veut. Ce n'est pas ce qui imagine ce que le client veut.

Avant, je dessinais le visage avec les yeux et tout. J'étais limitée par ma créativité. Et personne ne challenge mon design, parce que moi je dis de faire comme ça.

Maintenant, je vais à l'essentiel, je parle de masque pour les enfants, c'est pour le carnaval. Ça, c'est l'essentiel. Alors on a une équipe, 8 personnes qui pensent avec moi. Et je deviens paresseuse, je ne sais plus faire comme avant. Avant j'étais fort dans les détails, je ne loupais pas une virgule. Maintenant je suis quelqu'un comme ça, ça m'embête d'entrer dans les détails.

Alors la littérature sur Agile ça vient d'IT, c'est rare que ça vienne des patrons. Comme c'est un esprit, ça doit être Bottom up. Si c'est top down, un peu comme moi j'ai suivi les formes.

C'est l'équipe qui sent le besoin de faire mieux. On a un problème de communication, on veut faire bien, on veut ce changement, c'est ça qui marche le mieux. Est-ce que l'entreprise a des objectifs ? Donc les patrons ? En général c'est le contraire, tu demandes des formations, tu dois convaincre les patrons et quand tu fais ça, tu dois uniquement dire les points positifs, que tu peux délivrer beaucoup plus rapidement, que tu peux faire des changements rapidement. Mais le problème c'est qu'après on est coincé parce qu'on retient ça pendant que tu implémentes Agile. Ils retiennent c'est ça l'avantage et ils commencent à détruire le bon fonctionnement.

*Tu veux dire quoi par-là ?*

S.L. Je pense que je veux dire que l'entreprise a pour objectif d'implémenter ça. Je trouve ces questions bizarres parce que les patrons s'en fichent (dans mon expérience). Ça ne vient jamais d'eux. Maintenant tu parles d'IT manager, souvent il faut quelqu'un qui est vraiment convaincu. Il faut un croyant Agile pour amener ça. Quand tu es un croyant d'Agile, mais alors il croit en tous les objectifs d'Agile. On rend les clients heureux à la fin. Mais il comprend ça, mais ce n'est pas l'objectif de l'IT manager.

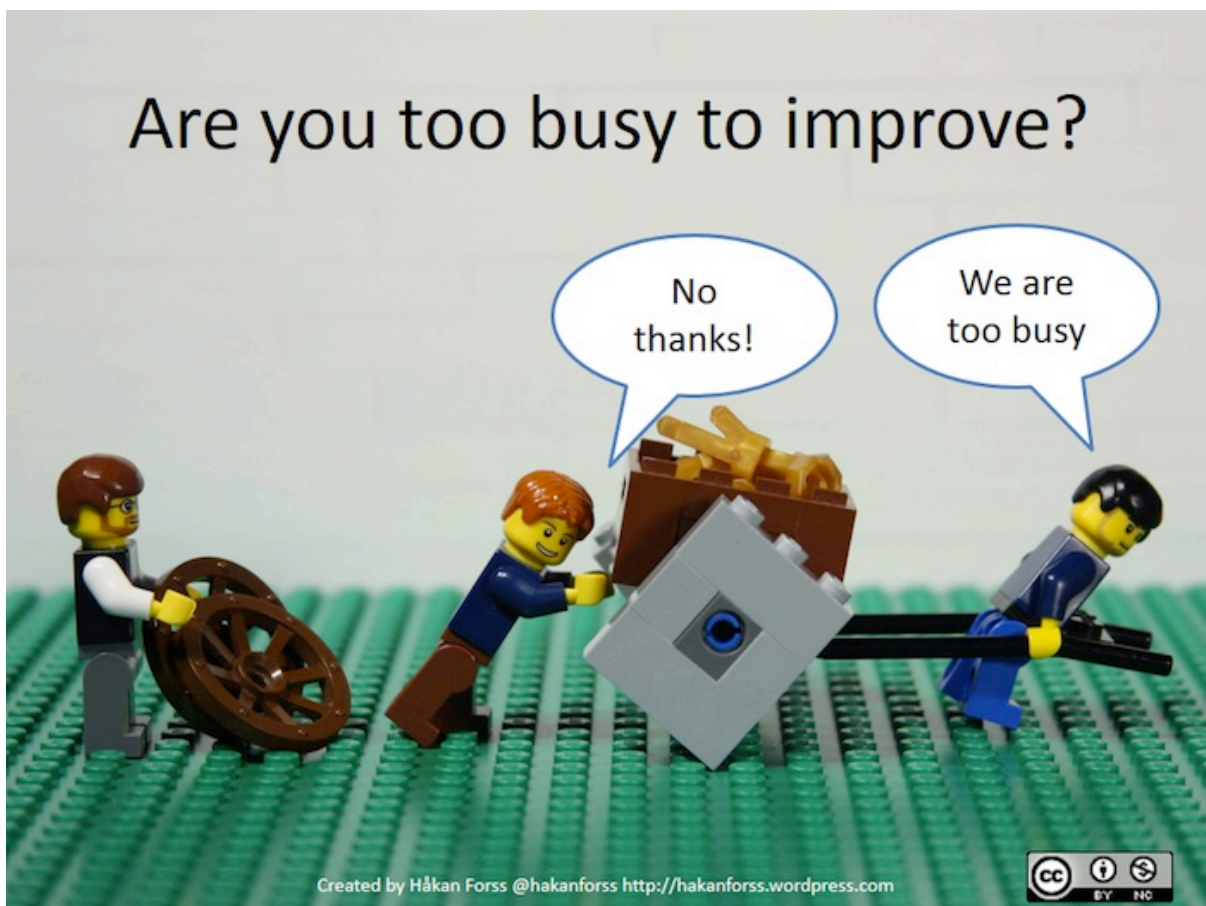
Oui parce qu'au final on ne sait pas de qui vient exactement la motivation dans l'entreprise.

Croyant c'est une chose, souvent ça vient de problème de communication on doit implémenter Agile, on met Agile ou Scrum on met une barrière, une raison de dire non pour ne plus venir. C'est : on fait Agile oui ou non ? alors je peux te dire non, le but de l'IT, du développeur, leur but ce n'est pas pour le client. Ça, c'est clair, ce n'est pas leur souci.

Agile fonctionne très bien avec de nouveaux produits, quand ce sont des nouveaux produits, tu as toute l'équipe sur le produit. Et aussi un nouveau produit ça veut dire on est pas, on ne sait pas, on ne connaît pas assez bien, on doit faire des changements, des ajustements très souvent. Donc ça fonctionne très bien avec les nouveaux produits.

Avec les produits qui restent déjà longtemps, Agile ce n'est pas évident du tout. Parce que ça marche beaucoup mieux avec les start-ups.

#### Appendix 17 : Are you too busy to improve ?



Source : <https://hakanforss.wordpress.com/2014/03/10/are-you-too-busy-to-improve/>

