



LAHDEN AMMATTIKORKEAKOULU
Lahti University of Applied Sciences

SCRUM'S SUCCESS FACTORS AND SCRUM'S BENEFIT LEVELS TO DIFFERENT PROJECTS AND PROCESSES

Case study: IT Quality team, HyperElectronics

LAHTI UNIVERSITY OF APPLIED
SCIENCES
Faculty of Business Studies
Thesis
Spring 2013
Dao Huong Thuy
Nguyen Ngoc Tran Khanh

Lahti University of Applied Sciences
Degree Programme in International Business

DAO, HUONG THUY

NGUYEN, NGOC TRAN KHANH

SCRUM'S SUCCESS FACTORS AND
SCRUM'S BENEFIT LEVELS TO
DIFFERENT PROJECTS AND
PROCESSES

Case study: IT Quality team,
HyperElectronics

Bachelor's Thesis in Business Studies 66 pages, 8 pages of appendices

Spring 2013

ABSTRACT

The objective of this Bachelor's thesis is to find out the success factors of Scrum and their assisting actions, and the projects and process types that can benefit from the application of Scrum. The target is to provide criteria for assessing the possibility of using Scrum for a project, as well as requirements for running a successful Scrum project.

This research is designed as a qualitative case study. The case study focuses on a project team who utilized Scrum in their project processes. The phenomenon observed here prompted the research problem and the research questions.

The theoretical framework provides background knowledge about Project Management standards, the traditional Waterfall approach, and Scrum. Scrum's principles and practices are described and then compared with the traditional approach. A number of debates about Scrum's application in different projects and processes are also presented.

The research employs an inductive qualitative approach. The data is collected by means of observation and interviews. The first step of the research was to observe the team's application of Scrum. The second step was to interview the team members and Scrum and/or Project Management experts from different organizations. The empirical evidence was analyzed thoroughly to deliver findings in the form of generalizations for multiple situations.

The research is concluded with findings encompassing Scrum's success factors, their assisting actions, and projects and processes types where Scrum can be beneficial or not.

Key words: Project Management, framework, Waterfall, Scrum, success factors

CONTENTS

1	INTRODUCTION	1
1.1	Background	1
1.2	Thesis objectives, research questions and theoretical framework	2
1.3	Research strategy and methods	3
1.4	Thesis scope and study design	6
2	PROJECT MANAGEMENT, WATERFALL AND SCRUM	8
2.1	Project Management	8
2.1.1	Definition of Project and Project Management	8
2.1.2	Project Management Life Cycle phases and processes	10
2.1.3	Project Management approaches and Waterfall	13
2.2	Scrum framework	18
2.2.1	Scrum roles	18
2.2.2	Scrum flow	20
2.2.3	Scrum science	23
2.3	Scrum versus Project Management	26
2.3.1	Project management success and failure	26
2.3.2	Project Management Life Cycle phases and processes in Scrum	28
2.3.3	Common Project Management problems	29
2.4	Scrum and debates about its application in Project Management	32
3	RESEARCH APPROACH AND METHODS	35
3.1	Research context	35
3.2	Acquisition of research data	36
3.3	Data processing and analysis	37
4	CASE STUDY: THE SCRUM FRAMEWORK	39
4.1	Research execution	40
4.2	Research results, findings and their analysis	42
5	CONCLUSIONS AND SUMMARY	59
	REFERENCES	67
	APPENDICES	73

LIST OF FIGURES

FIGURE 1 Thesis scope	6
FIGURE 2: Study design	7
FIGURE 3: Project Management within its constraints	9
FIGURE 4: Project Management Life Cycle (PMLC)	11
FIGURE 5: Waterfall method	14
FIGURE 6: Three main factors of agile process	16
FIGURE 7: Waterfall versus Agile Development Model	17
FIGURE 8: Roles defined in Scrum	20
FIGURE 9: A Product backlog	21
FIGURE 10: How Scrum works	22
FIGURE 11: Research procedure	35
FIGURE 12: Scrum success factors	43

LIST OF TABLES

TABLE 1: Evidence collection design	4
TABLE 2: Scrum success factors and assisting actions	60
TABLE 3: The benefit levels of Scrum to projects and processes	63

APPENDICES

APPENDIX A: Scrum definition

APPENDIX B: Scrum Cheat Sheet

APPENDIX C - List of Case study questions

APPENDIX D - Interview questions

APPENDIX E - The Stacey matrix

APPENDIX F - A Brief History of Project Management

LIST OF ABBREVIATIONS AND DEFINITIONS

PRINCE 2	Projects in controlled environments, version 2
PMI	Project Management Institute
PMP	Project Management Professional
PMBOK	Project Management Body of Knowledge
PMLC	Project Management Life Cycle
CPM	Critical Path Method
WBS	Work Breakdown Structure
AUP	Agile Unified Process
DSDM	Dynamic Systems Development Method
Open UP	Open Unified Process
ROI	Return on Investment
PM	Project Management

1 INTRODUCTION

This first chapter gives an overview of the thesis as it describes how the research is built. It contains the background, objectives, research questions, theoretical framework and scope of the thesis. It also provides the strategy and methods under which the research is conducted.

Over the past years, project management has evolved from a mere project management process into a business process. More and more companies have realized project management is now mandatory for the survival of the firm. (Kerzner 2009, xxi.).

Project management is nowadays required in every profession, including information systems, health care, consulting, pharmaceutical, banks and government agencies (Kerzner 2009, xxii); thus it has given birth to many project management frameworks and methodologies, such as Agile, Scrum, Waterfall, Spiral and PRINCE2 to name a few. There are many factors to be considered in deciding which frameworks and methodologies should be applied for one particular project. Chavrat (2003, 168) emphasized: “Selecting the wrong methodology for your project could be disastrous.”

Consequently, research about the applications of project management frameworks and methodologies is increasingly important. This thesis is written for this need. The next sub-chapter explains why the Scrum framework is chosen as the main subject.

1.1 Background

During winter 2012, one of the authors was working at HyperElectronics, as an intern in the IT Quality team. In October and November 2012, this team started adopting the Scrum framework in their project management processes, in place of the Waterfall framework.

Scrum is currently the leading agile development methodology, used widely by companies around the world, including Fortune 500's (Scrum Alliance 2013a). A quick Google search for “Scrum community” results in many different websites.

Among those is the Scrum Alliance, whose membership base is over 140,000 and growing as of 2012 (Scrum Alliance 2012). Many courses for expertise and certificates in Scrum are offered (Scrum Alliance 2013b). It can be seen that knowledge in Scrum is an advantage that many professionals seek.

Scrum was originally developed for software development, due to the growing complexity of these processes (Schwaber 2004, 13). Nonetheless, the great success of the method has made it increasingly popular in project and process management. Various case studies available from Scrum professionals have proved Scrum beneficial in different types of process and project. From observing and discussing what Scrum was able to deliver within the IT Quality team, the authors however were convinced that Scrum is not a foolproof solution in Project Management. It has its benefits and drawbacks. Thus, these questions emerged: “What are required for the implementation of Scrum? What projects and processes can benefit from Scrum?” The authors were strongly encouraged to find the answers.

The authors (referred to as “researchers” in a research context) decided to conduct research to assess the success factors in implementing the Scrum framework, and afterwards find out to which types of project and process Scrum is best suited.

1.2 Thesis objectives, research questions and theoretical framework

The primary goal of the thesis consists of two objectives. The first one is to find out the success factors of Scrum and the actions that can be taken to obtain them. The second is to reveal what types of projects and project management processes can benefit from the implementation of Scrum. The results of the thesis can help answer the following questions: when and when not to implement Scrum in a project? when doing one what need to be done?

In order to achieve this goal, four research questions are established as the backbone of the research, as listed below:

1. How is the Scrum framework compatible with the standardized project management processes (as defined by the Project Management Institute)?
2. What are the success factors of Scrum?

3. How can a project team obtain these factors?
4. Which project types and project management processes benefit from the application of Scrum?

Conducting a research can be related to building a house. A base must first be constructed for a house to be erected on, just as a theoretical framework must be employed for a research. As it is no exception, this thesis is built on the base of the theoretical framework involving project management and the particular Scrum framework and principals.

The main source for project management's definitions and processes is the ninth edition of the book "Project Management: A Systems Approach to Planning, Scheduling and Controlling" by Dr. Harold Kerzner. This book is widely used in PMP courses worldwide, and is among the best-selling books about project management (International Institute for Learning 2013). The book contains many important excerpts from the Project Management Body of Knowledge (PMBOK) provided by the PMI. The fact that the PMI is currently one of the world's largest non-profit membership associations for the project management profession (Project Management Institute 2013a) makes them highly credible as an information source. Moreover, their standards are globally recognized (Project Management Institute 2013b).

In addition, the Scrum framework's definitions and practices are extracted from the materials provided or recommended by Jeff Sutherland and co. (the creators of Scrum). This ensures the precision and trustworthiness of the information presented in the text. An important part of the literature reviews is the excerpts from the book titled "Agile Project Management with Scrum", by Ken Schwaber, a co-creator of Scrum. This book presents, among other essentials of Scrum, successes and failures of past Scrum projects (Schwaber 2004, ix). It contains fundamental information for studies about Scrum, hence chosen for this thesis.

1.3 Research strategy and methods

The research employs a qualitative case study. The empirical part looks into the implementation of Scrum within the IT Quality Function and discusses the

particular situation of the team, how specific factors supported or hindered the attempt to apply Scrum. Analyzing these factors creates the basis to find out the success factors of Scrum. As it is not included in the goal of the thesis, no solution or proposal for improvement is provided for the team.

Due to the interpretative characteristic of the research questions, the qualitative research methodology is straightforwardly chosen for this research. The reason lays in the in-depth nature of the qualitative research, as the purpose of it is to acquire depth of information. Hennink et al. (2011, 16) suggested that it is most suitable to explain and understand issues or to describe processes or behavior. As those are precisely the objectives of the thesis, it is clear that the qualitative research methodology can and should be employed herein.

It can be noticed that the research questions require an overall inductive reasoning, in which the theory is finally reached through analysis of data and case studies (Remenyi et al. 1998, 76). The mentioned data is collected from observation and interviews. The information collected for the case study is confidential. A summary of the evidence collection design employed in this thesis can be found in Table 1.

TABLE 1: Evidence collection design

Evidence Collection	Targets	Conduct location
Observation	the adoption of Scrum in project management processes in the IT Quality team	Eindhoven, the Netherlands
Semi-structured Interviews	Group 1: Four team members of the IT Quality team	Eindhoven, the Netherlands
	Group 2: Eight selected project management and Scrum experts	Online
Result: Success factors and their assisting actions, and most suitable project and process types for Scrum's implementation (generalizations)		

Observation

Observation is chosen as the primary source of data due to the direct involvement of one of the researchers (herein referred to as the “field researcher”) in the case company. Daily contact with the project team enables the richness and variety of information. The field researcher observes the adoption of Scrum of the case team. Its daily routine, the way it applies Scrum, the way its members behave and

their attitude are noted. The observed information discussed in this research is selected specifically to support the researchers in answering the research questions.

Semi-structured interviews

Semi-structured interviews allow the researchers to collect not only information but also opinions from different perspectives. Two distinctive advantages of interviews are the flexibility and the interactivity that can bring out underlying and unexpected matters. Kerzner (2009, xxii) believes that project management has a strong behavioral basis as projects are managed by people rather than tools. Therefore, it is very important to communicate with people in order to understand fully the subject.

The researchers interview informants from two groups. The informants in Group 1 include four members of the case organization. Among them are two professional project managers and two Scrum Masters. These group members are asked about their experience with Scrum, and their opinions towards the application of Scrum in different projects and processes.

Group 2's informants consist of eight selected experienced project managers who have been using Scrum in their work. These experts are chosen based on the following criteria:

1. They have had at least 2 years' experience in project management and/or using Scrum
2. They have good knowledge of Scrum and have had already a number of different successful Scrum projects

The aim of these interviews is to find out the success factors of Scrum. This group is asked to provide their own list of Scrum's success factors, the types of projects and processes that should be managed with Scrum, as well as the explanation for their opinions. Their answers come from the experience in their past work with Scrum projects; therefore ensure the practical aspect of the data.

1.4 Thesis scope and study design

When the research strategy and methods are determined, the next step is to define the scope and structure of the thesis. This sub-chapter provides an overview of the intended discussions and how these discussions are organized within this thesis.

The thesis aims to find out the success factors of Scrum, their assisting actions, and the projects and processes that can benefit from using the Scrum framework. Due to the limitation of a Bachelor thesis, no other framework or methodologies but Scrum and traditional approach (Waterfall) are covered. The thesis scope is shown in figure 1.

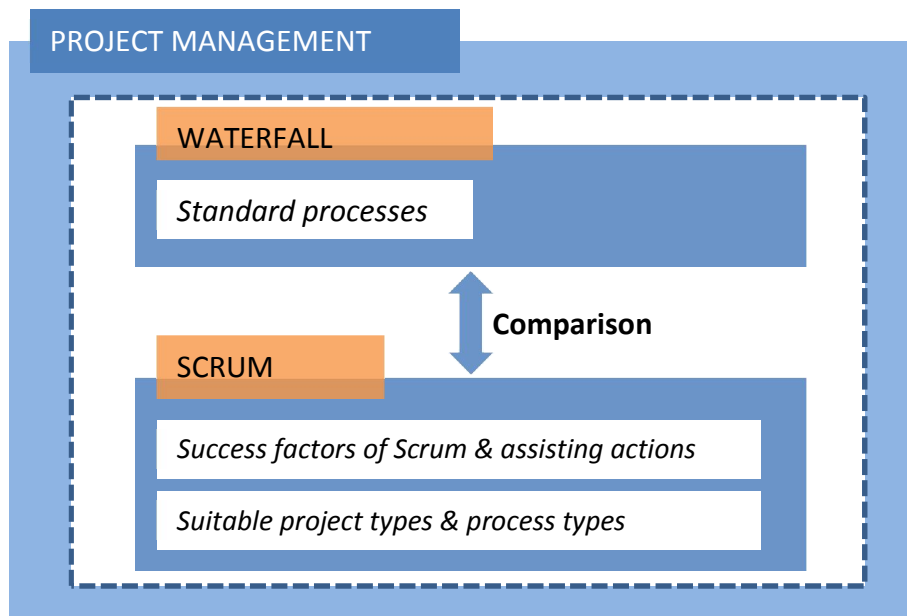


FIGURE 1: Thesis scope

The text first introduces briefly the Project Management standards, the traditional approach and the Scrum framework. In order to paint the picture of how Scrum looks next to Project Management, Scrum elements are compared and mapped to the project management standards. The empirical part analyzes and discusses the collected data. This part provides a case study with the IT Quality team in HyperElectronics. The results are the success factors of the implementation of Scrum in a project, their assisting actions, and the proposal of the suitable projects and processes for Scrum.

Figure 2 shows the study design graphically. The research structure is embedded in a house with the theoretical framework being the base, the research methods being a supporting structure for the content that answers the research questions. The findings are concluded at the end, representing the roof, which is strongly underpinned by the construction underneath.

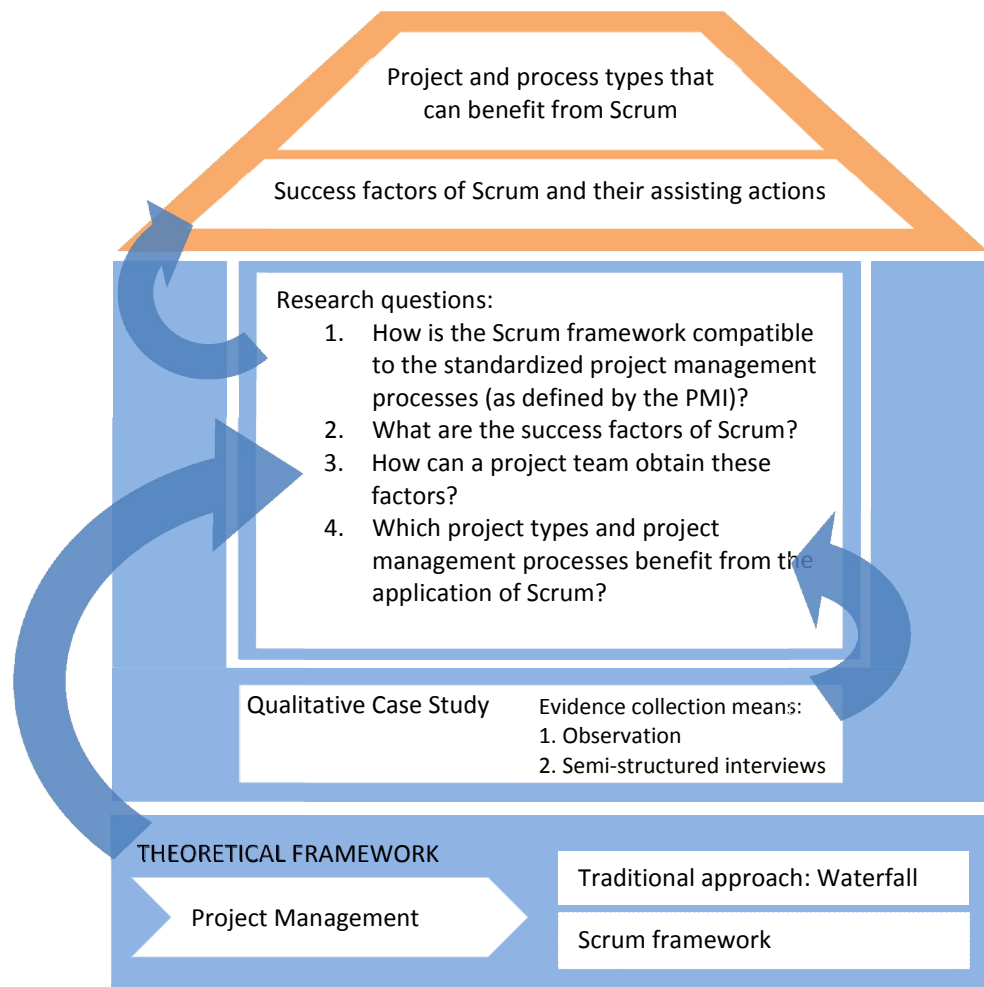


FIGURE 2: Study design

The thesis is divided into 5 chapters. Chapter 1 introduces the research. Chapter 2 describes the project management standards, the traditional approach and the Agile Scrum framework, therewith demonstrates how Scrum is compatible and complementing to Project Management. Chapter 3 explains the research approach and methods. Chapter 4 analyzes the collected data and reveals the research findings. Chapter 5 concludes the research with a summary, an evaluation and further recommendations.

2 PROJECT MANAGEMENT, WATERFALL AND SCRUM

This chapter aims to acquaint readers with the project management standards, the traditional Waterfall approach and the Scrum principles and practices. It presents brief introductions to several essential aspects of those mentioned above, in order to realize the relation between them, and to create a basis for assessing the application of Scrum. It must be noted that the discussed facets are only part of the vast expertise. They are chosen based on the assumption that Scrum is not separated but complementary to project management. Other aspects and evidence not supporting this supposition are not considered therefore.

2.1 Project Management

2.1.1 Definition of Project and Project Management

First, in order to understand the concept of project management, definition of what a project is should be established.

Project

A project can be defined simply as “a temporary group activity designed to produce a unique product, service or result” (Project Management Institute 2013c).

There are many other different definitions for a project; however, the PMBOK states: “A project can be considered to be any series of activities and tasks”, including “a specific objective to be completed within certain specifications”, “defined start and end dates”, “consume human and nonhuman resources”, “funding limits” and “multifunctional” (Kerzner 2009, 2).

Project Management

Project Management is one of the techniques created to respond to the request for improving productivity, efficiency in business while not increasing cost and labor (Kerzner 2009, 2).

Kerzner (2009, 2.) points out that changing and restructuring are in the nature of project management. Traditional management cannot adapt to changing

environment while project management helps executives to use existing resources effectively and control unexpected situations better.

According to Kerzner (2009, 4.), the PMBOK defines project management as “the planning, organizing, and controlling of company’s resources for a relatively short-term objective that has been established to complete specific goals and objectives”. Project management can be regarded as a strategy of companies to be competitive in the market. It includes the skills, knowledge and techniques required to achieve a specific goal efficiently and effectively in limited time and resources. (Project Management Institute 2013c.)

Project management success and failure

Modern project management defines successful project management as “having achieved the project objectives within time, cost, at the desired performance/technology level, while utilizing the assigned resources effectively and efficiently, and is accepted by the customers” (Kerzner 2009, 3).

As it can be inferred from the definition of successful project management, the constraints of a project are requirement factors that the project has to fulfill, no matter what. This constraints, according to Kerzner (2009, 5), are time, cost and performance. In addition, if the project is run for external customers, a fourth constrain is also included: good customer relations. Figure 3 depicts this relation.

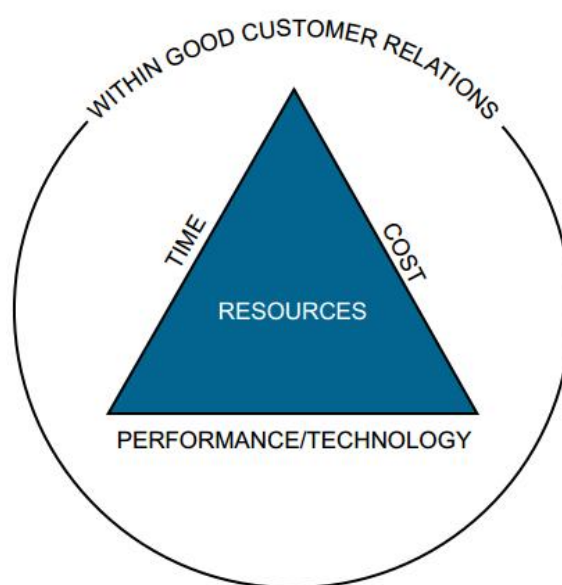


FIGURE 3: Project Management within its constraints (Kerzner 2009, 6)

Project Management constraints

Time

The Time constraint refers to the amount of time that the project needs to be accomplished (SQA Project management for IT 2007). Time is an important factor, which is hard to control. Sometimes, the project needs more time to complete the objectives because of many unexpected reasons. (TutorialsPoint 2013.)

Scope

A Scope shows what must be done during a process (SQA Project management for IT 2007). Scope is related to the end result of the project and focuses on the output of the project, such as product/service or profit.

Cost

The Cost constraint refers to the budget amount available for the project (SQA Project management for IT 2007). Cost management is necessary in a project. Under limited resources, the project manager has to estimate costs when executing the project to ensure the project will not be overbudgeted.

2.1.2 Project Management Life Cycle phases and processes

The most important goal that the project manager and project team share together is to achieve the project's objectives. With five main phases (Figure 4), the Project Management Life Cycle (PMLC) represents the path of the project from the beginning to the end. The life-cycle phases, as suggested by the PMBOK, include initiation, planning, execution, monitoring and control, and closure (Kerzner 2009, 3.)

Project Initiation & Project Planning

The initiation phase and planning phase are the first two phases of any project.

The project initiation phase, according to the PMBOK Guide, involves four processes: Selection of the best given resource limits, recognizing the benefits of the project, preparing the documents to sanction the project and finally assigning the project manager.

In the initiation phase, the project manager will determine the most proper solution and an accurate scope of the project after reaching an agreement with the sponsors (Barron & R. Barron 2009). In this first step, the sponsor takes a more important role than the project manager does. The purpose of the initiation phase is to develop a high-level plan for a proposed project and to shape a professional project team. (Project Management Methodology Guidebook 2003, 17.)

The planning phase is the step to develop all factors needed to accomplish the project's objectives such as time, schedule, cost, and performance. An accurate and detailed plan is to be created in order to start the project. (Project Management Methodology Guidebook 2003, 24.)

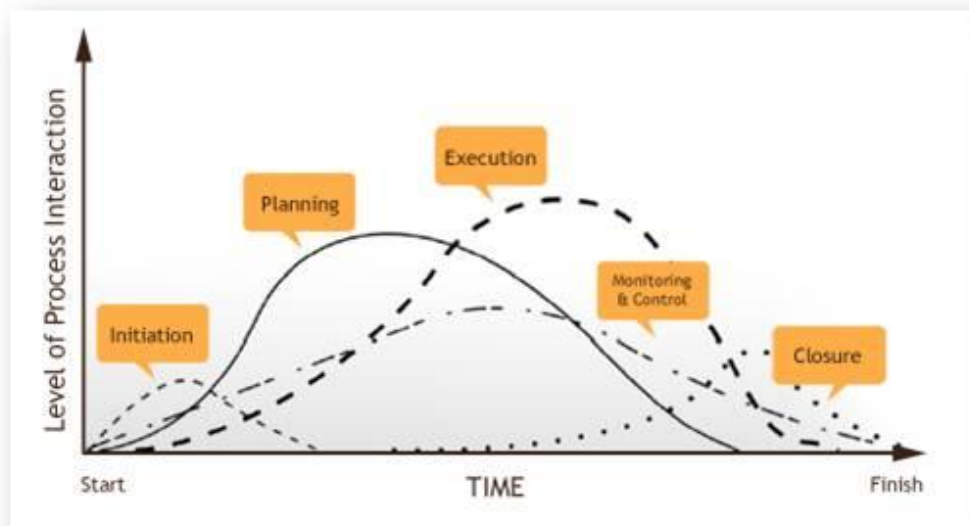


FIGURE 4: Project Management Life Cycle (PMLC) (GNSEgroup 2012)

In accordance with the traditional approach, after the work is identified, the project manager and project team ponder risk management by identifying and dealing with the high potential threats that can defeat the project. The main purpose of risk management is to avoid problems in the beginning, reduce the probability of occurring problems and their possible impact on the project. Finally yet importantly, the communication plan, which describes all details that should be informed to all the stakeholders, needs to be prepared. (Barron & R. Barron 2009.)

Project Execution & Project Monitoring and Control

The execution phase and the monitoring and control phase are two action phases in the PMLC. A project can be divided as 20% planning and 80% executing and controlling (Project Management Methodology Guidebook 2003, 59). During these phases, it is important for the team to maintain good communication and have the plan under control, aiming at its original objectives.

Team meetings are encouraged as often as possible during this step. The project team must update and report progress information to the project manager regularly. (Project Management Methodology Guidebook 2003, 61.) There are three types of review meeting: Project team review meetings, Executive management review meetings and Customer project review meetings (Kerzner 2009, 242).

Through team meetings, the project manager can decide the appropriate actions and at the same time measure the results, as well as build trust and understanding between the team members, managers, sponsors and other key stakeholders.

Project Closure

Closure is the final phase of the PMLC. At this point, all project activities must be completed. In this step, the sponsors review the project and final testing is done at the same time. The customer accepts the final project deliverable. (Project Management Methodology Guidebook 2003, 75.)

Through closing phase, the project is evaluated in all aspects. It has an important impact on starting a new project or other ongoing projects. (Kerzner 2009, 71.) During the final step, the final deliverables to the customers are emphasized. The project manager is responsible for such final work as handing over project documentation to stakeholders, assessing customer satisfaction, evaluating the project team's performance and capturing lessons learned. (Project Management Methodology Guidebook 2003, 75.) By conducting lessons learned studies and examining what the project team did well and what they did not, the project manager will have more experiences for future projects. (Barron & R. Barron 2009).

2.1.3 Project Management approaches and Waterfall

During the first years of 1950s, “Project Management” was brought into the scene “as a distinct contribution arising from the management discipline” (Cleland & Gareis 2006, 1-4) by Bernard Schriever, a U.S Air Force general. At that time, it was used only in the engineering field, as the US Air Force needed to look at how to further develop its missiles. Since then, several methods have been developed to improve Project Management itself. For example, Critical Path Method (CPM) in 1957 and Work Breakdown Structure (WBS) in 1962 are the two important methods which are still widely used. (Miles 2012.)

The traditional approach: Waterfall

Almost 20 years later, Dr. Winston W. Royce – an American computer scientist and one of the leaders in software development, popularized the Waterfall model. In the paper “Managing the Development of Large Software Systems: Concepts and Techniques”, although he used the term “a flawed, non-working model” to present this model, he is still regarded as “Father of Waterfall” (Stephen On Software 2010.)

Two basic characteristics of this model are linear and sequential. “Imagine a waterfall on the cliff of a steep mountain. Once the water has flown over the edge of the cliff and has begun its journey down the side of the mountain, it cannot turn back. It is exactly what the Waterfall model is”. That means for each phase of development, Waterfall model has definite goals to be achieved. Moreover, one phase of development must be completed before going to the next phase, and there is no turning back to review the process. (Search Software Quality 2007.)

Waterfall is regarded as the traditional approach to software development. Since a phase cannot be revisited if it is not completed, first try should be right.

Otherwise, it cannot be fixed at the end of the process. (The Agilista PM 2013.)

Figure 5 is a diagram originally developed by Royce in 1970. Other models of Waterfall are slightly different. For example, the Requirements phase can be changed into the Idea phase or broken into Planning phase and Analysis phase. (Douglas 2009.)

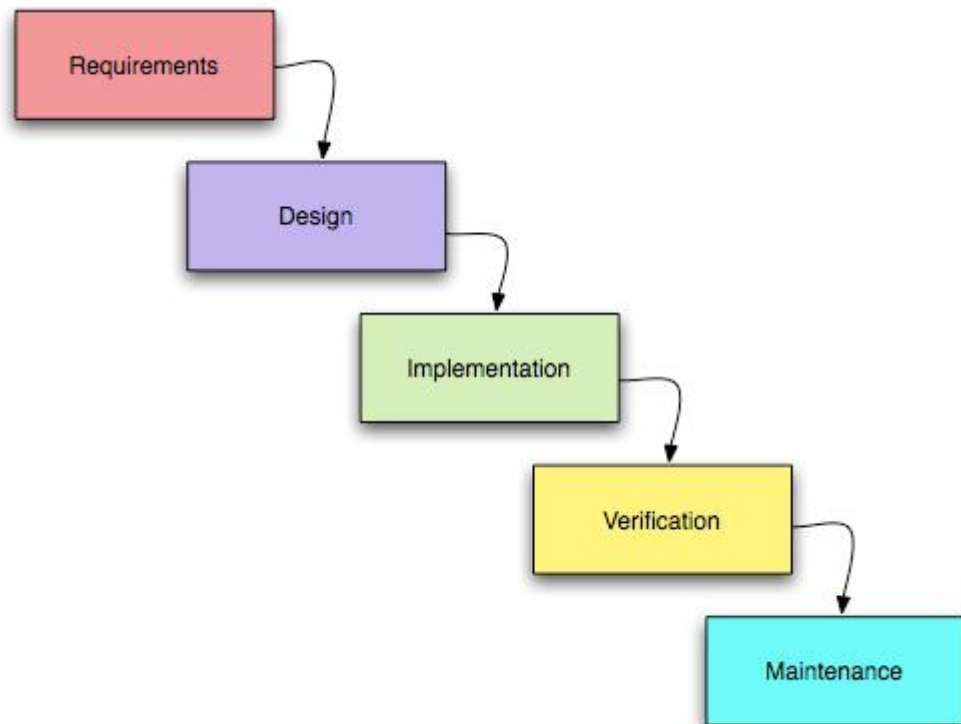


FIGURE 5: Waterfall method (Douglas 2009)

In the requirement phase, the requirements of the system are defined and set clearly. Therefore, this phase is considered very important. The project team gathers all the requirements according to customers' needs. Communication with users is the main activity in this phase. (Douglas 2009.)

After the design phase, one or more design specifications are completed as the output that would be used for the next step implementation (Tech Republic 2006).

Implementation takes place after system design and software design are done. The coders and programmers play a crucial role in this phase. They review the project requirements and code the applications (Douglas 2009).

The system is tested in a verification (testing) phase in order to remove any faults in earlier phases. Originally, the verification phase was to ensure that the project meets customers' expectation. (Douglas 2009).

Maintenance is the last phase of the Waterfall model. The customers actually use the applications during this phase. Changes can be made in this phase if improper deliverables or other mistakes are found. (Douglas 2009).

Waterfall has its own advantages. This model has a structural approach, making it easy to measure progress with clear milestones. In Waterfall, a schedule can be set with deadlines for each stage of development, therefore help delivery product on time. As there are clear phases, the project team can avoid overlapping or iterating problems by proceeding in strict order the action phases. (Search Software Quality 2013.)

Nowadays, projects are getting more and more complicated. It is not easy to decide all the steps, and plan everything at the beginning. When Dr. Royce first described this model in his paper, he also admitted this approach is too risky and fragile (Project Manager 2011). The biggest disadvantage is the poor flexibility. A complex project involves many stakeholders, making it very difficult to determine their expectations at the beginning. During the project, some requirements need to be changed in order to maintain the competitive level of the product. It is difficult and expensive to go back to the previous stage. Waterfall model is neither a practical nor an economical method in these cases. Furthermore, some project team members might stay inactive since only certain members are responsible for each action phase, creating wasteful use of resources. (Select Business Solutions 2013.)

Agile project management approach and Scrum

Emerged in mid-1990's, Agile methodologies and practices seek to embrace the increasing rate of change in software requirements, which has exceeded the capabilities of the traditional development methodologies. (Williams 2007, 209.) Alleman (PMI-ACP Guide 2013) defines Agility as something that “describes the behavior of the participants and their ability to move or adjust in new and possibly unforeseen situations”.

Agile project management emphasizes focusing on continuous alignment and deliverables in accordance with customer requirements through agile management software (Boruff 2011). Moreover, the agile project management approach is highly attentive to customer feedback in order to increase the satisfaction and company value. Due to its characteristics, agile approach can be applied to many types of projects, even under the high risk and continuous changing project structure.

The agile processes include three main factors: Incremental and evolutionary, Modular and lean, Time-based. Figure 6 below describes more precisely how those factors work in the Agile process.

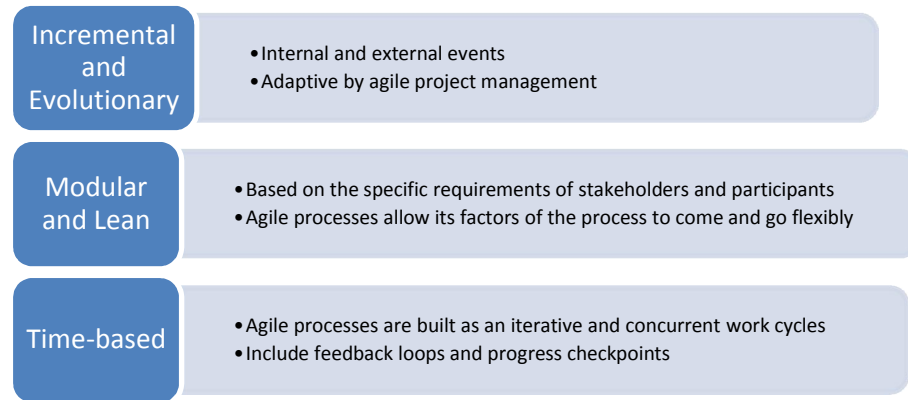


FIGURE 6: Three main factors of agile process (PMI-ACP Guide 2013.)

Figure 7 depicts the difference in Waterfall and Agile Project Life Cycle in software development. In Agile, each one of the incremental iterations is structured in a way that includes four out of five main classical stages: Planning, Execution, Monitoring & Control (Test), Closure (Deliver). There is a continuous improvement process, as lessons learned are reviewed after each iteration. It will be demonstrated more clearly in the later part of this thesis, how these iterations are structured via Scrum.

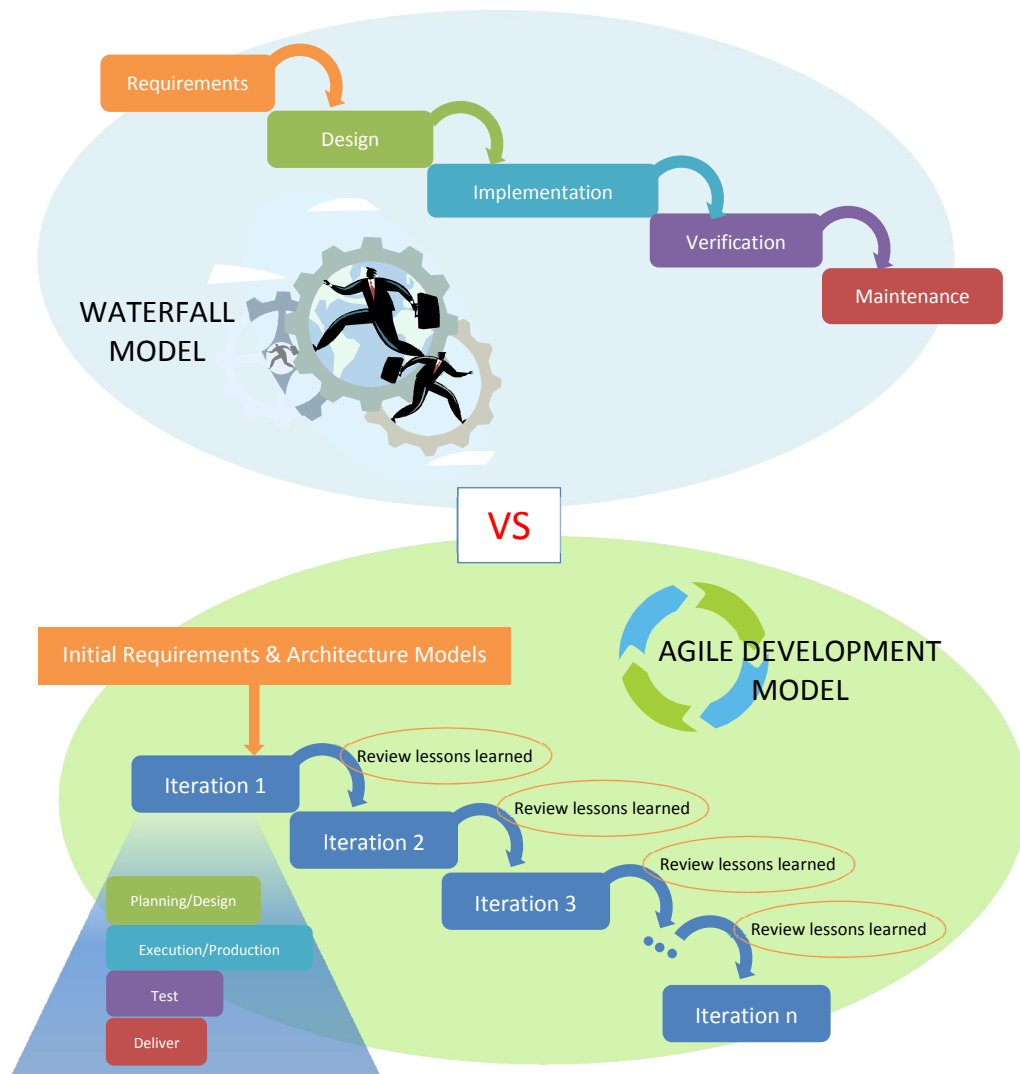


FIGURE 7: Waterfall versus Agile Development Model (modified from Hass 2010; Mc Gevna 2012)

Scrum is one of the well-known agile software development methods, besides Agile Unified Process (AUP), Dynamic Systems Development Method (DSDM) or Open Unified Process (Open UP). Cprime (2012) defines Scrum as “*a lightweight process framework for agile development, and the most widely-used one. A "process framework" is a particular set of practices that must be followed in order for a process to be consistent with the framework. "Lightweight" means that the overhead of the process is kept as small as possible, to maximize the amount of productive time available for getting useful work done*”.

The next sub-chapter introduces shortly Scrum’s practices and principals. It answers questions such as “What is Scrum? From where does Scrum get its

power? Why does Scrum insist on such practices? What benefits can it bring to businesses?” and so on.

2.2 Scrum framework

Scrum is defined by Jeff Sutherland – the original creator of Scrum – as an “iterative, incremental framework for projects and products or application development” (Sutherland 2010, 6). The term “Scrum” comes from Rugby, “relating successful development to the game of Rugby in which a self-organizing (self-managing) team moves together down the field of product development” (Sutherland 2010, 7).

The significant benefits Scrum can deliver to business are reflected in the numbers and sizes of companies who have been adopting the framework. This list includes, but not limited to, Microsoft, Google, John Hopkins APL, Siemens, Nokia, and the US Federal Reserve. The subsequent sub-chapters reveal the principals and practices underpinning the power of Scrum. Scrum employs its own set of terminologies, which can cause difficulty in understanding. Short definitions for these terms are provided in Appendix A.

The purpose of this sub-chapter is to give an overview of how Scrum works. It is not a guide for Scrum’s application in a project. The text covers the basics of Scrum that is necessary to understand it.

2.2.1 Scrum roles

The Scrum roles are different and cannot simply be translated to those of traditional approaches. It is hence necessary to, first of all, understand these unique roles.

In Scrum, there are three primary roles: The Product Owner, The Team and the Scrum Master. These roles are illustrated in Figure 8.

The Product Owner

The Product Owner is responsible for maximizing Return on Investment (ROI) for the project. The Product Owner identifies product features (requirements) and

then translates these into a list with differently prioritized items. This list is called the Product Backlog (Schwaber 2004, 19). The Product Owner decides which should be done first, placing these requirements on top of the list for the next iteration. The Product Owner can be the same person as the customer in some cases, which is common for internal applications. In other cases, the role is comparable to the Product Manager or Product Marketing Manager position, though with certain differences. (Sutherland 2010, 14.)

The Team

The Team is responsible for building the product that the customer is going to use. The Scrum Team is cross-functional, as it includes all the expertise necessary to deliver the potentially shippable product after each iteration. The amount of members is seven minus or plus two. The Team develops the product and provides ideas to the Product Owner about how to optimize it. (Sutherland 2010, 15.)

It is strongly emphasized in Scrum that the Team is self-organizing (self-managing), with a high degree of autonomy and accountability. There is, therefore, no team manager or project manager in Scrum. The team members decide by themselves what to commit to, and how to accomplish that commitment in the best possible manner. (Sutherland 2010, 15.) This fundamental principal of Scrum lays an important restraint in implementing Scrum, which will be discussed in the next chapters.

The Scrum Master

The Scrum Master serves the Team by protecting them from outside interference, ensuring the self-organizing principle of Scrum. The Scrum Master is responsible for educating and guiding the Product Owner and the Team “in the skillful use of Scrum”. (Sutherland 2010, 16.). In other words, the Scrum Master makes sure the principles and practices of Scrum are understood and correctly employed by the Scrum team.

The Scrum Master is sometimes compared to the Project Manager role in traditional project management. Sutherland (2010, 16), however, states that the Scrum Master is not the manager of the team or a project manager. Schwaber

(2004) repeatedly emphasizes that the Scrum Master's role is only a facilitator or enabler, who has no authority over the team. His job is not to assign tasks, but to provide the team with wholehearted support and to remove any impediment that occurs.

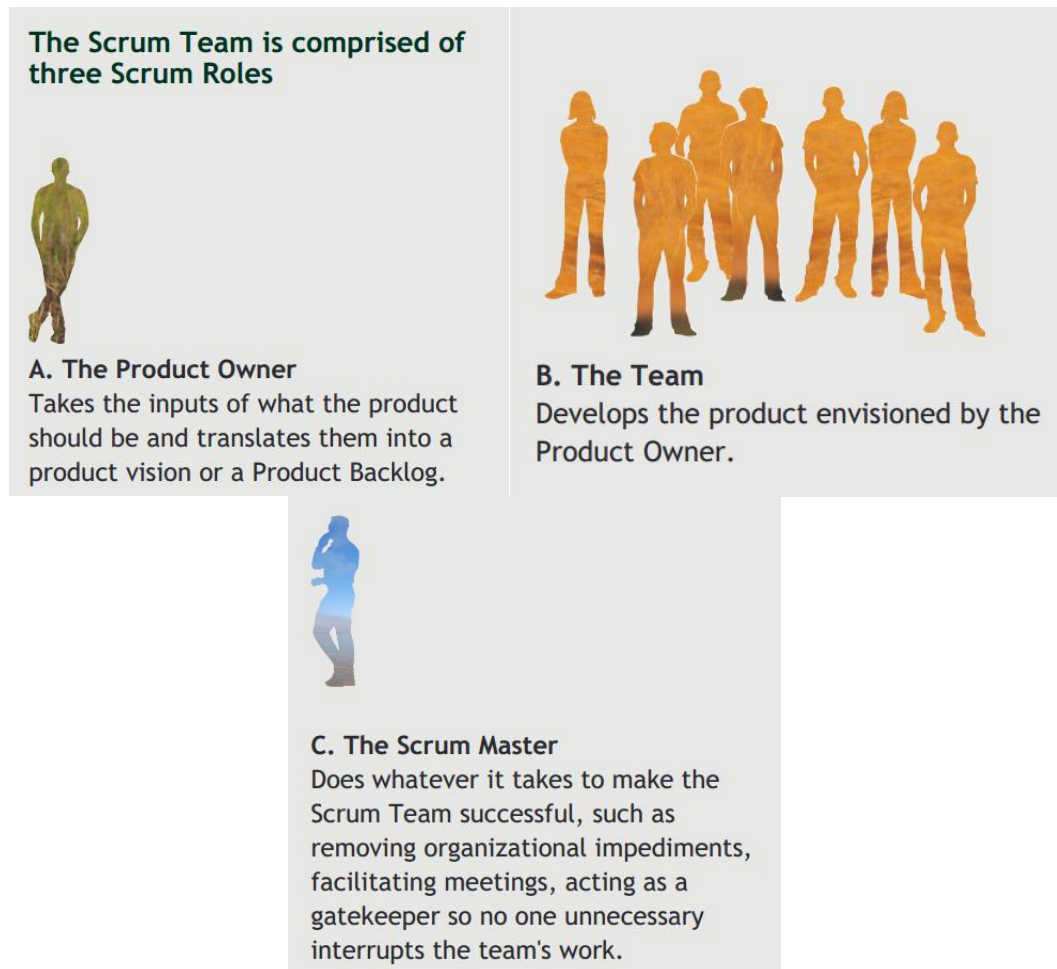


FIGURE 8: Roles defined in Scrum (modified from Sutherland 2010, 10)

2.2.2 Scrum flow

The Product Backlog

A Scrum project starts with the Product Owner delivering a vision of the product, which must maximize the ROI of those funding the project. This vision then evolves into a refined and prioritized list of features (*the Product Backlog*), ranked in order of value to the customer or business. (Sutherland 2010, 10-18). This backlog exists and constantly evolves over the lifetime of the product, reflecting “changes in the needs of the customer, new ideas or insights, moves by the

competition, technical hurdles that appear, and so forth”. It can be considered as the roadmap of the product. (Sutherland 2010, 18-19.) Figure 9 presents an example of the Product Backlog.

Item	Details (wiki URL)	Priority	Estimate of Value	Initial Estimate of Effort	New Estimates of Effort Remaining as of Sprint...			
					1	2	3	4
As a buyer, I want to place a book in a shopping cart (see UI sketches on wiki page)	...	1	7	5				
As a buyer, I want to remove a book in a shopping cart	...	2	6	2				
Improve transaction processing performance (see target performance metrics on wiki)	...	3	6	13				
investigate solutions for speeding up credit card validation (see target performance metrics on wiki)	...	4	6	20				
Upgrade all servers to Apache 2.2.3	...	5	5	13				
Diagnose and fix the order processing script errors (bugzilla ID 14823)	...	6	2	3				
As a shopper, I want to create and save a wish list	...	7	7	40				
As a shopper, I want to add or delete items on my wish list	...	8	4	20				

FIGURE 9: A Product backlog (Sutherland 2010, 18)

The Team is responsible for providing the estimates of the effort required for each item to the Product Owner. Additionally, the Product Owner assigns a business value estimate to each item. Based on these estimates and the capability of the team, the possible date for a release or the possible released features for a specific date are projected. The estimates of the items in Scrum are usually expressed in form of “points”. The progress of the team is measured by indicators such as the completed “points” each iteration, the initial estimated, the real effort needed and so on. (Sutherland 2010, 19.)

The Sprint

Scrum operates in cycles of iterations. Each iteration (called *Sprint*) lasts from one to four weeks. The Sprints are of fixed duration, which is never changed. Each Sprint ends on a specific date, whether the work is complete or not. (Sutherland 2010, 10.)

Sprint Planning

The Sprint planning meeting takes place at the beginning of each Sprint. The Product Owner and the Team (with the Scrum Master as the facilitator) review and discuss the Product Backlog items. The Team then collectively selects the items it wants to commit to and complete by the end of the Sprint, starting at the top of the Product Backlog. Each item is broken down into a set of individual tasks, each task takes roughly four to sixteen hours to finish. The list of tasks is

called the Sprint Backlog. (Sutherland 2010, 10; Schwaber 2004, 24.) The Sprint Backlog must be updated daily and kept visible to every team member (Schwaber 2004, 125).

One vital principal of Scrum is that: “once the Team makes its commitment, any additions or changes must be deferred until the next Sprint”. That means no one can interrupt the work of the Team to change, subtract or add an item during the current Sprint. Under extreme circumstances, the Sprint can be terminated, giving place for a new Sprint planning meeting; and a new Sprint starts. (Sutherland 2010, 23.)

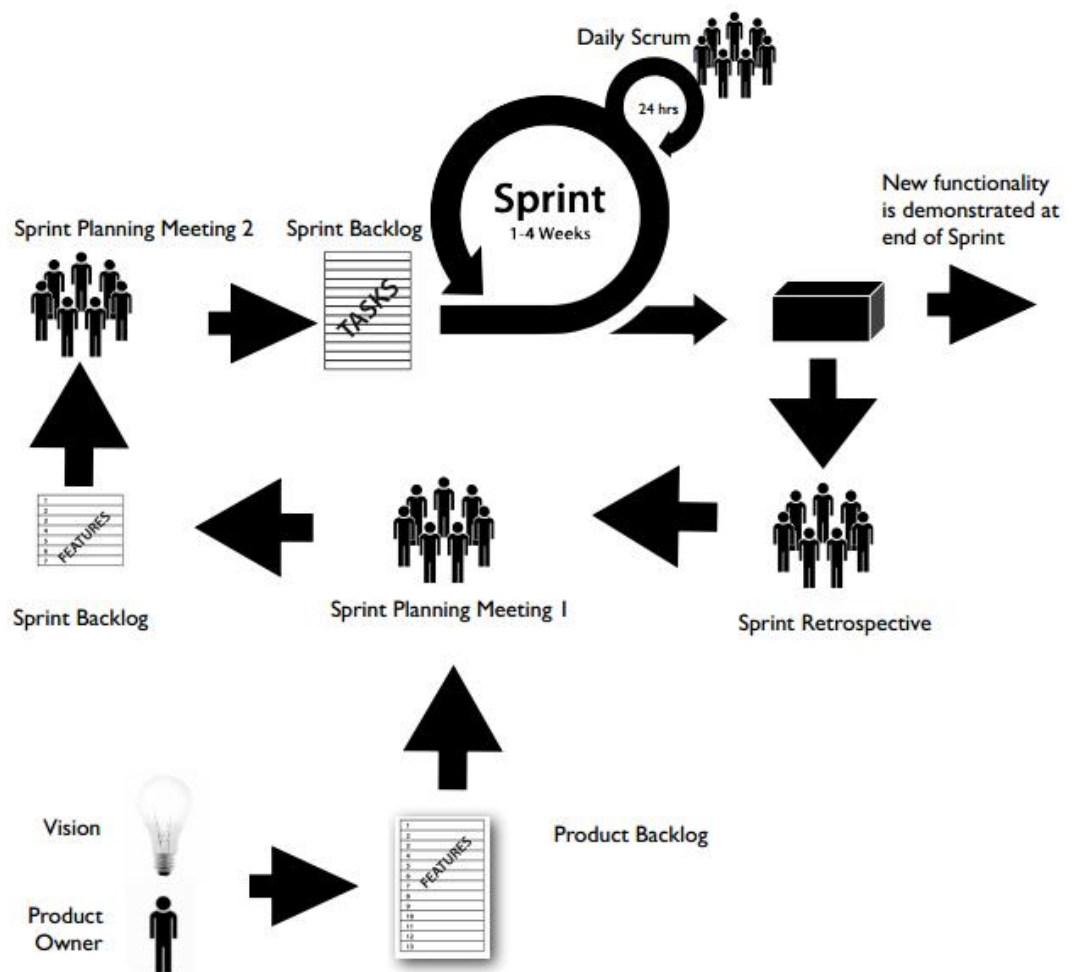


FIGURE 10: How Scrum works (Sutherland 2010, 11)

Daily Scrum Meeting

The Daily Scrum Meeting is defined by Sutherland (2010, 10) as “a short (15 minutes) meeting that happens every workday at an appointed time”. Everyone has to attend and report his or her progress to one another, ensuring transparency within the team. The information presented in the meeting may result in re-planning and further discussions or actions immediately afterwards. (Sutherland 2010, 10.)

Sprint Review and Sprint Retrospective

When the Sprint ends, the team conducts a review session (*Sprint Review*), with the Product Owner, the Team, the Scrum Master, plus customers, stakeholders, experts, executives, and anyone else interested. The Scrum team and the stakeholders inspect and discuss what was done during the Sprint, then figure out together what to do next. The Review is followed by the *Sprint Retrospective*, which provides an opportunity for the team to discuss what is working and what is not. The team discusses solutions and agrees on what to change. (Sutherland 2010, 10.)

The sections above just summarize the workflow of Scrum. Figure 10 illustrates how Scrum works in graphics. It can be seen that Scrum has developed several elements unique to it. What are the reasons behind these creations? What benefits can they bring? In order to answer these questions, the next sub-chapter continues to explore the underlying science that Scrum employ in its practices.

2.2.3 Scrum science

This section explains certain pillar practices of Scrum, as well as the advantages of Scrum over the traditional approach in Project Management. Due to the focus of the research, only aspects deemed relevant to revealing the success factors of Scrum are discussed.

Empirical process control

The empirical process control is embedded in the nature of Scrum. Schwaber (2004, 13.) suggests that the using of empirical process control emerged from the increasing complexity and unpredictability of development processes.

“Laying out a process that repeatedly will produce acceptable quality output is called defined process control. When defined process control cannot be achieved because of the complexity of the intermediate activities, empirical process control has to be employed.”(Schwaber 2004, 13.)

Schwaber (2004, 13) points out that there are three pillars underpinning every implementation of empirical process control: visibility, inspection and adaptation.

Visibility

Visibility means that those aspects of the process that affect the outcome must not only be visible, but also be true, to those controlling the process. For example, it must be agreed and clear to everyone in the Scrum team what is the “definition of done” of each task and functionality. The project cannot run productively if different individuals expect different results from one certain task. (Schwaber 2004, 13.) Scrum holds on to this principal constantly through every process. Scrum includes the Definition of Done in its practices, which is agreed by the whole team during each Sprint Planning meeting (Sutherland 2010, 20).

Furthermore, Scrum makes a clear distinction between two groups of stakeholders in each project: It must always be visible who is committed and who is merely involved. It does so to “ensure those who are responsible for the project have the authority to do what is necessary for its success”, and “those who aren’t responsible can’t interfere unnecessarily”. (Schwaber 2004, 19.)

The structure and operation of meetings in Scrum also allow high level of transparency within the team. The size of a Scrum team (seven minus or plus two) allows voicing of opinions from every member in a timely manner. The daily Scrum meetings ensure any impediment to be updated quickly, and known to the whole team. Schwaber (2004, 99) states: “Scrum relies on high-bandwidth, face-to-face communication and teamwork; cubicles and unneeded artifacts promote isolation and misunderstandings”. Scrum encourages multi-skilled workers, in the sense that team members must “go to where work is” and help one another with their tasks (Sutherland 2010, 22). This mindset enables members to work together and learn from one another. Consequently, sharing knowledge and collaboration

are strong advantages of the Scrum framework, essential to creating phenomenal productivity.

Inspection and Adaptation

Inspection and Adaptation are the means for Scrum to implement the empirical approach. Scrum is able to respond to the volatile nature of product development, as well as to protect the high level of creativity and innovation, thanks to its insistence on continuous inspection and adaptation. Sutherland (2010, 6) explains the reason as Scrum takes short step of development, and repeats inspecting both the result product and the efficacy of current practices, and then adapt the product goals and process practices.

“All of the stakeholders are brought together every month to inspect progress on the system and to determine whether it meets their perceived needs, addressing their highest priority first. To the extent that the process of translating the requirements into demonstrated increment of functionality doesn’t meet their needs, the process, people, technology, or requirements are adapted to be more effective.” (Schwaber 2004, 104.)

According to Schwaber (2004, 125), the length of a Sprint is limited within one to four weeks with the following considerations taken into account. First, it is the amount of time required for the Team to build something of significant interest to the Product Owner and stakeholders, and bring it to a potentially shippable state. Second, it is also the maximum time that most stakeholders would wait without losing interest, and without losing the belief that the Team is doing something meaningful for them.

Scrum employs a practice called “sashimi”. The term is derived from the Japanese thin slices of raw fish: each slice is complete itself, and similar to the others. The sashimi technique in Scrum requires that every slice of functionality created be complete. At the beginning of each Sprint (which must be kept short), the stakeholders can see that they have the opportunity to redirect the project and optimize the value they can acquire. Then at the end of each Sprint, they see new complete functionality. (Schwaber 2004, 60.)

Self-organization

A Scrum team has to be self-organized. Schwaber (2004, 54) insists that Scrum rely on individual and team commitments, not top-down control through planning (common with the traditional approach). Through his work, he solidly asserted that self-organization and human commitment, though difficult to implement, far exceed the imposed controls, plans and even loyalty. These factors constitute the Scrum mechanism that fosters creativity and worker satisfaction. (Schwaber 2004, 48-98.)

In summary, the science of Scrum depends greatly on the principles of visibility, the inspection/adaptation cycle and self-organization. Scrum is however, as Schwaber describes, “the art of possible”. It means “to focus on what can be done rather than be frustrated by what can’t be done”. (Schwaber 2004, 46.) How does this science fit into the big picture of project management? Sub-chapter 2.2 provides a mapping of Scrum to the Project Management standards presented previously, thus answers this question.

2.3 Scrum versus Project Management

Scrum may employ different principles than the traditional approach in “planning, organizing and controlling”. However, this section will later prove that Scrum embraces all these processes in its practices. Scrum complements each process with its own unique elements, therefore solves certain problems, which the traditional approach could not achieve.

This section demonstrates how Scrum manages the processes and the constraints in Project Management, as well as how it addresses the common management problems. In other words, the text below draws a picture of how Scrum looks like when placed next to the traditional Project Management.

2.3.1 Project management success and failure

Understandably, it is not an easy job to deliver a project successfully. It is evidenced that only 16,2% out of the total number of projects can be considered completed on time and within budget while fulfilling all specified functions and

features (Frese and Sauter 2003). According to Attarzadeh and Ow (2008, 236), the top two reasons contributing to projects' failure are incomplete requirements and lack of user involvement.

Scrum clearly sees through these problems, as it focuses on involving the users and discussing the requirements. The developing product and requirements are checked with the customers at the beginning of each Sprint, to ensure the team is producing what the users want. If there is any mismatch, or new realization from the customers, new requirements will be added to the backlog. This mechanism allows a complete alignment between the project team and the customers.

Moreover, by constantly involving the customers in the development of the product, Scrum automatically fulfills the last objective "accepted by the customer". Other characteristics of the Scrum team are the cross-functionality and the encouragement of multi-skilled team members. Sharing knowledge and helping one another is a focus of the Scrum teamwork. The relatively small size of the Scrum Team and its high level of transparency in every operation prevent free riders and non-participants. Thus, Scrum possesses a strong basis for "utilizing the assigned resources effectively and efficiently".

Time, Cost and Performance constraints

The time and cost perspectives of a project receive attention in Scrum. The Product Owner has the responsibility to optimize ROI; therefore supports the cost control of the project. The Product Backlog contains estimates of effort needed to complete every task. These estimates usually consider the hours each task require (expressed as "points"), therefore are able to reflect the timing and costs of the project. Sprints are time-boxed, and each Sprint is to deliver a complete functional product, which enables the team to perform in a timely manner.

Regarding the classic time, cost and performance constraints, Scrum however approaches them differently rather than trying to keep them under control. In fact, Scrum, as an empirical process control, accepts and embraces the inevitable volatile nature of project scopes. In other words, in Scrum, projects may not finish on time, within budget and intended performance. This can be still considered

acceptable and a successful project if it satisfies other requirements and brings benefits to the organization.

2.3.2 Project Management Life Cycle phases and processes in Scrum

Project Initiation & Project Planning

Scrum's initiation and planning phases include the Product Owner getting together with the customers and other stakeholders to recognize the main objectives and benefits of the project. The Product Owner analyzes what he has in hands to find out the requirements for the products and the priorities of these requirements, so that the project can be executed with the highest ROI possible. The adoption of these two phases in Scrum is reflected partly in the Product Backlog. As mentioned before, the Product Backlog acts as a dynamic overall roadmap or plan of the project. It is not proclaimed in Scrum how detailed a Product Backlog should be, therefore it depends on the Product Owner and the team to decide what should be therein.

Risk Management & Communication Planning

Scrum does not clearly identify these processes in its practices. It can be inferred from the flexibility principle of Scrum that risk management is not a focus in planning. Since Scrum accepts that the changes during the project are neither predictable nor avoidable, it has developed a structure that can adapt to instability, rather than control it. Scrum however tries to prevent changes during one Sprint - which can cause problems and disrupt the work process of the team.

Nonetheless, the making and execution of the communication plan are not addressed in Scrum. Due to the intangible characteristics of communication, applying the Scrum practices must be considered thoroughly. The case study with the IT Quality team will give an example of this subject.

Project Monitoring & Control

The focus of Scrum on the Sprints exhibits Scrum's adaptation of this phase. Traditional planning and controlling are displaced by human commitment and embedded inspection and adaptation. The team keeps its visibility to the highest level possible; constantly reports work progress and problems to one another. By

being committed to be true to its members, the team “controls” itself instead of being controlled.

Project Closure

Scrum does not openly define any practice particularly aimed for the closure phase. Nevertheless, the sashimi rule requires that all the deliverables at the end of each Sprint be completely “done”, tested, error-free and shippable, just as a final product usable by the users. It therefore can be assumed that the final closure phase of a Scrum project is similar to an end of a Sprint, much less heavy than that of the traditional approach.

This section just presented how Scrum appears next to Project Management, from the processes’ perspective. It showed that Scrum practices also embrace the traditional processes, though with Scrum’s own modifications and complements. A summary of this analysis can be found in Appendix B. The next part continues to explore how Scrum addresses the common problems in project management.

2.3.3 Common Project Management problems

Time

Time is limited in projects; therefore, project managers should know how to use it effectively (Time Management). Unlike other routine operation, a project has specific beginning and ending days. There are numerous reasons for losing time during the project process, such as a lack of job description or technical skill, too many meetings or much travelling or even too many people involved in making decisions. (Kerzner 2009, 287.)

Scrum answers to the timing problem with its time-boxed routine and its efficient and effective utilization of resources, as well as the sashimi rule, as previously discussed. Schwaber (2004, 104) however admits that imprecision in estimation is inevitable, and that “any estimate of a release date becomes suspect”. As the word “estimate” suggests, when the Scrum team estimates their required effort and time for tasks and requirements, imprecision is unavoidable. Scrum does not provide any solution for this lack of certainty. On the other hand, Schwaber (2004, 104) considers such imprecision is, though worrisome, inherent problem of software

development. He is positive that after one or a first few Sprints, the Team will improve and be able to provide estimation that is much more precise nonetheless.

Kerzner (2009, 289.) believes that project managers must practice as many techniques as possible to keep the project finished on time. These techniques should be used flexibly and based on specific situations. A good project manager must be able to decide quickly, and must learn how to separate unimportant matters and follow the project's objectives in order to manage the time.

In Scrum, this responsibility however falls into the hands of the Product Owner. The Product Owner must decide what is more important, what brings more value to the stakeholders and the project, and what should be completed first. The Product Owner must learn to say "No" to deliverables that do not contribute to or even hinder the success of the project. (Kniberg 2012.)

Stress

Stress always exists in projects, affecting from the project team to the project manager. Due to the constraints of the project, all people involved in the project are under stress of deadlines, roles and responsibilities, conflicts between project team, communication with project manager and so forth. Understandably, roles that have more responsibilities will impose more stress. Project managers are usually tired, depressed or unhappy if something is wrong in the project. (Kerzner 2009, 291.)

It can be seen that Scrum methods efficiently reduce stress imposed on the project team. First, the responsibility is shared among all team members, Product Owner, Scrum Master and the Team. No one is the manager and everyone is equally responsible for the project as the others. The Product Owner and the Scrum Master do not work on their own, but supported by the whole team to fulfill their roles and responsibilities. The Scrum Master, whose role is the closest to that of a Project Manager, does not manage anyone but facilitates and enables others instead.

Furthermore, Scrum requires frequent contact between team members. Human interaction and face-to-face communication are strongly encouraged. A workplace without any physical obstacles, which allows employees to walk around freely

and interact with others, is very beneficial to a Scrum team. (Schwaber 2004, 106.) It is widely known that human interaction is an effective means to reduce stress, as well as to provide help within the community. Strong bonds between the team members also create a pleasant atmosphere and allow everyone to have fun even at work. Employee satisfaction increases, consequently improves productivity.

Conflicts

Project teams are formed across different departments or organizations. Therefore, the members might not have had a chance to work with others before. Especially when the economy becomes more international, the differences of culture, language, religious, political view are getting bigger. (Kerzner 2009, Chapter 7.)

This indeed can be a tremendous obstacle in the implementation of Scrum. Scrum requires a fundamental and concrete commitment between team members and between the team and the project. If there is a disparity between the levels of commitment of the team members, the Scrum project cannot achieve its requisite visibility. Since Scrum relies heavily on visibility, the project is very likely to fail.

Communication

Communication plays a crucial role in a project success. Effective communication can be defined with 4R: Right information, Right people, Right time and Right cost (Kerzner 2009, 233).

However, it is very difficult to ensure that the communication will go smoothly during the project. Deciding how many people will join the project team is very useful in the initiation phase to avoid a bad communication plan. In order to keep the right information go from the sponsors to the end user, the project manager can use team meeting as an effective measure. Team meeting can include daily meeting, weekly/monthly meeting or irregular meeting according to the characteristic, length and size of the project. (Kerzner 2009, 238.)

Continuous communication and visibility are strongly stressed in Scrum. It employs daily meeting within the team, and Sprint-based meetings with other stakeholders. At the beginning each Sprint, the stakeholders are asked if the requirements fulfilled last Sprint match with their demand. At the end of the same

Sprint, the results and overall progress are announced and discussed with the stakeholders to detect problems and find out the solutions.

This however can be a pitfall of Scrum in certain projects. When the project in question has a large population of stakeholders who have to be involved, Scrum's too frequent communication can be overbearing and become cumbersome.

Overall, Scrum provides great benefits to stakeholders in a project. Scrum has become increasingly popular in the past years. Many people use Scrum as a general-purpose framework, which is not only suitable for software development as originally aimed for. It has been proved effective not just in software development, but in other processes as well.

Scrum however might not be a "silver bullet" in Project Management. That is to say, it has its own drawbacks along with its advantages: nothing is perfect, not even Scrum. Searches on Google and professional networks such as LinkedIn or Stack Overflow result in many discussions on "when or when not" Scrum. There are many different views on this subject, as it is a rather complicated question. The next sub-chapter presents a short summary of the discussions collected by the researchers. These discussions are chosen as they do not rely on one-sided assumptions and the arguments are based on real-life experience.

2.4 Scrum and debates about its application in Project Management

Scrum was dissected in sub-chapters 2.2 and 2.3, therefrom proved to be complementing to Project Management, as it provides solutions to several common problems in traditional Project Management. Nonetheless, it can be seen that Scrum's unique elements might not fit in every project or organization. Though many organizations might argue that they have been able to apply Scrum in most cases, they often have to modify one or more elements of Scrum in fact. Thus, these modified versions of Scrum are not actually "Scrum", but "Scrum variants".

CPrime, a well-known Agile coaching site, provides a case study with the company Wiredrive. This company is a software developer who succeeded in adopting Scrum in their software development process. The company then tried to

displace traditional Waterfall with Scrum in their sales process, and failed. They found out that “Standard Scrum metrics, such as tracking “Stories done” and “work remaining,” were not very useful for a sales process”. They opted to use different strategy, and got more success with this change. (CPrime 2012)

The flexibility and uncertainty in Scrum also pose a question: How does Scrum deal with a project with fixed end date and budget? Scrum insists the project scope will change, therefore accepts imprecision in estimates. When the project requires very exact estimation and planning to end at the planned date and within budget, Scrum might be inferior to Waterfall, or another solution in Project/Process Management.

The Scrum frequent meetings also can raise the costs and burden the operation. In certain situations, when the project is small (in size, length), or purely operational work, and so on, Scrum might be too “heavy” to use. Additionally, it has been mentioned before that the estimates in Scrum need time to develop their precision. Therefore, projects with too short length might not be suitable for Scrum either. Berteig (2010) argues that teams less than five people or projects shorter than two months are not suited for Scrum.

Cohn (2011), Berteig (2010) and Thompson (2009), among others, also believe Scrum is much less useful in projects without enough complexity or uniqueness (e.g. purely operational work, when the customers know exactly what they want, when the work is thoroughly understood in advance and so forth). In these cases, Waterfall might be a better choice as the work can be planned before execution and there are not so many changes or creativity requirement for an Agile approach such as Scrum.

Thompson (2009) adds that Scrum might as well be too much for projects with work that is not cyclic. The heart of Scrum lies in the incremental iterations that deliver functional products after each Sprint. It is therefore reasonable to question the compatibility of Scrum in such projects, when the delivery of complete work is not possible in time-boxed cycles of iterations.

Regarding the human factors in the implementation of Scrum, it is important to realize that Scrum relies heavily on people and their commitment in the processes.

The Agile Scrum mindset is innovative and rather new to the established management profession. In addition, it requires specific work culture and environment. Many changes need to be imposed in the old traditional way of working and mindset for Agile Scrum to thrive. Moreover, making people entirely committed in all processes can be a very difficult task. The case study with the IT Quality team will depict this point more clearly.

Berteig (2011) mentioned the cases of management teams, pointing out that they have significant difficulty in using Scrum for their processes. The reasons include them having “operational responsibilities (non-creative, non-problem-solving work), urgent and legitimate interruptions (e.g. escalated customer issues), real commitment that are calendar based (e.g. management off-site)” and ego.

Overall, the application of Scrum in different types of processes and projects is rather controversial. The abovementioned debates create a base for the researchers to investigate further to answer the research questions. The literature review ends here and empirical study starts in the next chapter.

3 RESEARCH APPROACH AND METHODS

The preceding sections laid out a solid theoretical background for the research. This chapter delves into the case of the IT Quality team. It examines the way of implementing Scrum in the team's project management processes. By doing this, the researchers aim to find out the success factors of Scrum, coupled with the types of projects and project management processes favored by Scrum. The next section describes how the case study was carried out. This process is depicted in Figure 11.

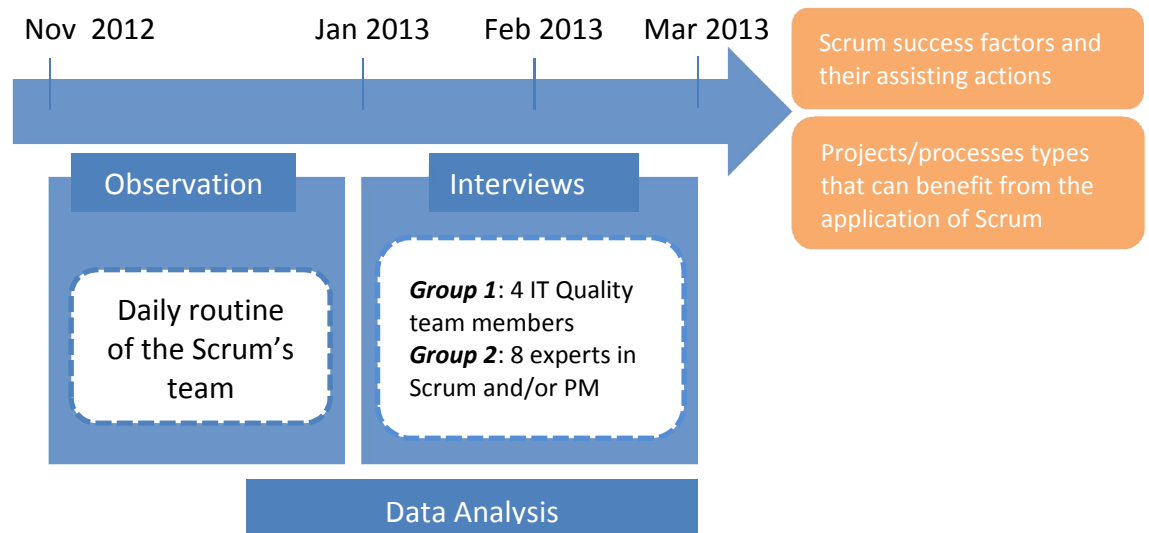


FIGURE 11: Research procedure

3.1 Research context

HyperElectronics is a large corporation operating on an international scale. It has a bulky and heavy operation. It was in need of a standardized quality management framework for the whole organization. For this reason, the IT Quality team was established. Their main mission was to implement an industry-best-practice quality management framework in HyperElectronics.

The IT Quality team consisted of eight core members. Five members were located in Netherlands, three in India. As the team's products would be used for different IT processes, the team members included representatives from several processes

within IT. These members were working in different projects from their own departments. The work in the IT Quality team became a part-time project for them.

In November 2011, HyperElectronics was undergoing a major Agile transformation in their business processes. The Agile transformation started in 2009 in several software development teams. The Scrum approach was used here for a number of marketing projects. After one year, it was assessed as a success and the decision was taken to apply it through the whole of HyperElectronics. By the end of 2012, more than 80% of the projects in the organization had been categorized as “Agile”. The aim is to have all projects run in Agile frameworks, including Scrum, before the end of 2013.

As a part of this transformation, the IT Quality team decided to change their management methodologies. The Waterfall framework was then replaced by the Scrum framework.

3.2 Acquisition of research data

Before conducting the case study, it is important to outline the study protocol. The researchers must define clearly the objectives, the procedures, the expected outcome of the case study and so forth. These issues are identified below.

The main objective of the case study was to obtain empirical evidence of how the Scrum framework is used in project management processes. It first started with the case of one specific organization. To create empirical generalizations, the researchers then opened the discussions to external experts in the field.

The evidence was collected by means of passive observation and semi-structured interviews. The informants were encouraged to express their opinions freely. The researchers insistently promoted open discussions and sharing of knowledge between two parties.

Two focal points of the case study were the success factors of the Scrum framework and the projects and processes suitable for it. The evidence was

analyzed and discussed in such manner that explained clearly why and how the researchers came to their conclusions.

At the beginning of the study, passive observation offered basis data to the researchers. Taking advantage of the direct involvement in the project processes, the field researcher observed the daily routine of the case team. The comportment of the Scrum roles and the entailed artifacts (e.g. Product Backlog) were noted and studied during the process. The observation was carried out for ten weeks. The analysis of the evidence collected during this phase created a foothold in answering the research questions.

Afterwards, semi-structured interviews with two groups took place. The first group consisted of four observed IT Quality team members. These informants included a senior manager and the team members who held different Scrum roles, i.e. Product Owner, Scrum Master and Team Member. The objective of these interviews was to reveal their honest opinions and feedback on the application of Scrum in the project and in general. The second group consisted of eight experts in Scrum and/or Project Management. This group was selected from inside and outside of the HyperElectronics organization, via recommendation of the first group and via LinkedIn professional network. The researchers were concerned that Agile/Scrum proponents can overlook the pitfalls of their favored framework. Consequently, to avoid bias, neutralists were also interviewed. These interviews aimed to collect evidence from different organizations. The evidence, in turn, enabled the researchers to form empirical generalizations that realize the research objectives.

A set of questions for the case study was established beforehand as a guideline to collect the right evidence. These questions, listed in Appendix C, are in alignment with the research questions to ensure the attainment of the research objectives.

3.3 Data processing and analysis

As this is a qualitative case study, the evidence collected was analyzed under a qualitative approach. The researchers focused on examining in-depth data from different perspectives. The evidence engendered from observation and interviews

were looked at holistically; great care was taken to identify biases. The researchers also investigated opposite views to each argument.

The data was first recorded in various forms, such as written texts, printed materials, audio records, and electronic materials. Afterwards, all of them were transferred to electronic format. Oral interviews were recorded in digital audio records with a laptop computer. These records were then translated to transcription and summarized in spreadsheets. Email conversations were also transferred to spreadsheets in the same structure with interview answers. Non-digital materials were studied thoroughly so that the relevant information can be collected and summarized in spreadsheets.

The responses were categorized according to the research questions that they answer and the perspectives represented. Responses from the two informant groups were analyzed separately and then compared. There were elaborated conversations, which contained information not included in the research scope. This information was treated as knowledge support and not discussed in the thesis. The interviews and their analysis were conducted in parallel; any interesting finding occurred during the analysis were discussed in subsequent interviews. A second round of the analysis was run after all the data had been collected to ensure the thoroughness of the research.

The interviews and analysis lasted roughly one month. The findings were deemed satisfying in relation to the research objectives. During the process, the researchers discovered that the success factors and their associated actions are numerous and should be placed under different categories.

During the interviews, the informants were specifically asked for comments in general situations based on their personal experience and knowledge. The researchers were thus able to make generalizations applicable in multiple situations other than the case study. These generalizations will be presented in table form in Chapter 5.

As the protocol for the case study has been laid out, the next sections examine in-depth the study itself. The evidence collected will be presented and evaluated in detail.

4 CASE STUDY: THE SCRUM FRAMEWORK

The preceding chapter presented the approach and methods of the research. This chapter describes the actual execution of the research, the resulting evidence, its analysis and findings.

As mentioned in the previous chapter, the IT Quality team wanted to change their management framework from Waterfall to Scrum. However, within the IT Quality team, except for one member who had been working with the Agile Delivery Center, the rest were not familiar with Scrum. Scrum was introduced to the team by using the book “The Power of Scrum”, which describes the essentials of the Scrum principles and practices. The Project Manager and his assistant first spent some sessions with a professional Agile Coach to gain insights into the application of Scrum. Believing in the feasibility of the transformation and that “Scrum is better learned in action”, the team started to apply Scrum two weeks after the proposal was taken into consideration. This change was implemented in the hope of higher level of productivity.

“Productivity can be improved by Scrum, when the method is properly implemented” (IT Quality team 2013).

The results, nonetheless, did not live up to the initial expectations. In fact, after the first ten Sprints, when asked about the improvement and diminution in the project, the informants from the team did notice that productivity had not increased at a visible level. Though some aspects of the project (visibility, achievement etc.) improved, some have counter-intuitively diminished. The application of Scrum did not help the team reach the intended goal. Two possible causes for this were that the project team and other stakeholders did not provide necessary success factors for Scrum, and that the nature of the project and its processes did not lend itself to the Scrum principles. These causes were not mutually exclusive. They can together contribute to the failure of the application of Scrum in any project in reality.

The researchers considered these possibilities most likely and conducted the research under this assumption. No matter what, the thesis ultimately aimed to find out the generalizable success factors of Scrum and the suitable project and

process types for it, as well as to explain the underlying reasons. The research therefore did not examine other possible causes. This evidence and its analysis are discussed in detail under the next subheading.

4.1 Research execution

The research data was collected over a course of five months, from November 2012 to March 2013. The field researcher worked as a member and observed the team actions for ten weeks, from November 2012 until January 2013. The team took up two-week Sprints in the project, hence this period accounts for five Sprints. It must be noted that the project has not ended and its success cannot be determined. Only the application of Scrum during these ten weeks was examined. Five Sprints, however, is a fair amount of time to observe the operation of Scrum. It was mentioned in the literature review, and during the interviews, that three Sprints is usually the time needed for a team to adapt to Scrum, and for Scrum's engine to take off and get running.

The data emerged from the observation came in various forms and served as the basis of the research. The evidence included notes about the daily comportments of the team members, their arrangement and adaptation of Scrum practices. Arguments, disagreements and any plan miscarrying were paid special attention. Scrum artifacts, namely the Product Backlog, Sprint Backlog, and Burndown chart, were examined and stored for further study. These artifacts, however, were not shown to anyone outside of the team to protect confidential information. During this period, the research questions were developed along with the researchers' understanding of Scrum.

After the observation phase ended, the researchers were able to study the theoretical materials more thoroughly, with the help of real-life experience. The researchers reserved two weeks to complete the literature review and the research questions. The researchers then proceeded with further investigation into the subject. More specifically, the interviews took place from the third week of February 2013.

The researchers spent roughly one month on the interviews. First, the interview structure was formed and revised (see Appendix D). Afterwards, the researchers contacted the informants to schedule interviews. For informants who were not able to give face-to-face interviews, the researchers prepared Google forms with the same questions and asked them to fill in. If needed, the informants were contacted again and more discussion took place via emails to extract elaborated feedback.

Regarding Group 1, the procedure went smoothly, as all the informants were aware of the research and knew the field researcher in person. The good personal relationship between two parties was considered beneficial, as it encouraged the informants to give detailed and honest answers, not superficial ones. The four informants include all different roles in the Scrum processes (Scrum Master, Product Owner, Team members), as well as different involvement levels in the project (100% committed, part-time 50% committed, high-level manager/supporter). All the interviews in this group were finished in the third week of February 2013.

Regarding Group 2, the researchers had difficulty scheduling the interviews, as most informants were external consultants and had busy schedules. One Scrum expert inside of HyperElectronics, who was referred to for the researchers, replied immediately. Invitations were sent out continuously on LinkedIn to different Project Management/Scrum experts until the planned number of informants (eight) was reached. All the informants' profiles fulfilled the criteria stated in Chapter 1. Opened to sharing knowledge, these informants were motivated to help the researchers and contribute to the research. The informants were also interested in the findings of the research and wished to exchange more ideas in the future. After obtaining their permission, the researchers sent out the interview questions. The informants were asked to fill it in with their answers and to give a short interview afterwards. All informants filled in the form in detail and agreed to elaborate their answers via emails if asked. Among them, three agreed to be interviewed via Skype. These conversations were recorded to digital audio files with a laptop computer. The audio was then transferred to transcription in electronic spreadsheets. The informants were then asked if they wanted to have

their names and titles referred to in the thesis. The interviews in Group 2 and their analysis were completed in parallel and finished after one month.

When all the data collection and analysis had been done, the researchers came up with the findings fulfilling the initial objectives of the research. These findings are described in the subsequent sub-heading.

4.2 Research results, findings and their analysis

This section encompasses the actual evidence collected and the findings derived therefrom. The process of thoughts, how the evidence was analyzed, will be described in detail. The remaining research questions will be finally answered.

What are the success factors of the Scrum framework?

How can a project team obtain them?

It is reasonable to infer that there are many factors required to apply successfully Scrum in project management. Due to the limitation of the research, it is not possible to find out all of them. The success factors mentioned herein are deemed intrinsic to Scrum. Their critical levels may vary; some are vital, while some are recommended. Nevertheless, they are all highly important in most situations, and must always be taken care of for Scrum to thrive in any circumstances.

The factors are grouped into the following categories: People factors, Organization factors, Application factors and External factors. Each category is further divided into sub-divisions. In many cases, the factors are interdependent, which means they should be looked at holistically. The research also introduces the actions to help fulfill the requirements of these factors. Figure 12 illustrates the factors and their interdependency.

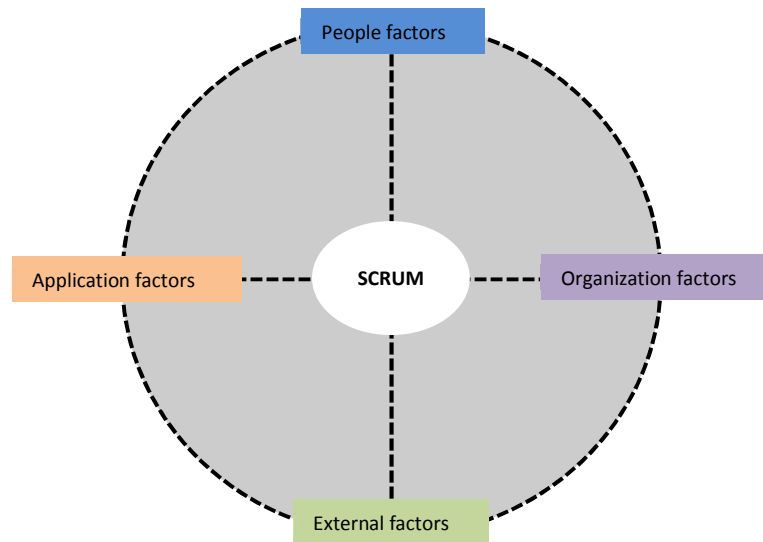


FIGURE 12: Scrum success factors

People factors

The people factors are the most important factors in Scrum (Lander 2013). They entail requirements for all the roles in Scrum, i.e. the Product Owner, the Scrum Master and the Team.

First and foremost, the whole Scrum team must be totally dedicated and committed to the project. Not only stated theoretically in the Scrum guide, this requirement was also mentioned by all informants. The observation and interviews with the IT Quality team indicated that this requirement was unfulfilled in its case. Since most team members were engaged in different part-time projects, the level of commitment was in fact greatly reduced. Without fully committing to their work, the team could not fulfill their respective responsibilities, nor follow the Scrum principles.

Equally important to their dedication and commitment is the openness to changes and willing to go “fully Agile” from the people.

“One of the key factors for success in Scrum is that team members not only want to do Scrum in name, but really want to be “Agile”, and are open to change” (Lander 2013)

For many people, using Agile is more of a transformation than a mere change.

The Agile mindset and principles are very different from those of the traditional

management, which has been used for decades. It is therefore very important that anyone who wants to do Agile, is able to embrace wholeheartedly all the changes.

The abovementioned success factors are the required elements from all the roles involved in any Scrum project. Each role in Scrum, in turn, has its own responsibilities, which must be fulfilled for the success of Scrum. Therefore, the Product Owner, the Scrum Master and the Team roles have their specific requirements, examined below.

The Product Owner

The Product Owner must understand his heavy responsibilities, have sufficient mandate and have a clear vision. A good Product Owner is vital to the success of a Scrum project. A 2012 Agile survey in the Netherlands, conducted by Xebia (a Dutch consultant company), reveals that the top reasons for failing an Agile project are associated with the Product Owner role (Xebia 2012). These include Product Owner not being available, making no choices, having no mandate, and so on. Indeed, Borselaer (2013) stated in his interview that the situation where “Product Owner hasn’t enough mandate” is default in corporate culture in the Netherlands. Thus, solving this type of problem is rather complicated and requires cooperation between all parties involved. Whether this situation is unique to the Netherlands or not, it demonstrates the great importance of the Product Owner to Scrum processes.

Product Owner must be able to make decisions when it comes to the products and act quickly to respond to changes, since his responsibilities mainly rest on the products. In many cases, he needs to discuss and negotiate with the stakeholders and the customers to protect the feasibility of the project, as well as make them understand the Scrum principles and practices. He acts as the bridge between the actual production and the expected output. Without the proper guidance of the Product Owner, the Team does not know exactly what the product should be like and easily deviate from the expectations from the stakeholders. The Product Owner must therefore be available for the Team and spend sufficient effort and time in making the requirements as clear as crystal to everyone. His role has heavy burden and should not be shared with other roles (Scrum Master or the Team).

In support to this argument, a member of the IT Quality team noted that what could be changed to improve their application of Scrum was the Product Owner role. In particular, this role was given to the members of the Team, whereas it should have been a customer from the business side. Additionally, most informants in Group 2 chose the Product Owner role as one influential success factor. When asked specifically, Lander (2013) stated that the role of the Product Owner, though can be shared between teams, should not be shared with other roles. In other words, a Product Owner can work for several teams at the same time, however in any case he must be the Product Owner only.

On top of understanding his responsibilities, the Product Owner must also have a very clear vision of the product. This requirement came up either explicitly or implicitly during conversations with the informants. A clear vision means a very thorough knowledge of the product, its requirements and the value of each function, both from the functionality and from the business value perspectives. The Product Owner is the one who decides which requirements add more value and the order in which they should be done. In order to do so, he is required to have a certain level of experience and good technical and business knowledge. Additionally, there must be ample communication, preferably face-to-face conversations between the Product Owner and the stakeholders, the Product Owner and the Team. How much communication is enough may vary greatly depending on different situations. Communication is discussed in detail later under its own section.

An interesting note during the research is that the Product Owner role was much more frequently discussed than the Scrum Master role. As the Scrum Master is supposed to be the replacement of the Project Manager in traditional management, one can assume that the Scrum Master role usually receives more attention. This, however, does not suggest either role is more important. One possible explanation is that the Product Owner role is more complicated, consequently more often misinterpreted or underestimated and underperformed.

The Scrum Master

The Scrum Master is the facilitator with knowledge and understanding, not the manager. This is one of the key elements in Scrum and must not be forgotten. The

long-used Project Manager role, who manages people, has been embedded in project management. If the Scrum Master falls into this trap and becomes a manager, not letting people manage themselves, Scrum's fundamentals are broken and the framework would not work anymore. This requirement depends also on the organizational factors, which are discussed later under its own section. Being the protector of the Scrum processes, the Scrum Master must understand the underlying science of Scrum, as the name already suggests. Without full insights into Scrum, a Scrum Master may misinterpret and apply it in the wrong way. For instance, he may break a Scrum principle without knowing so; or he may blindly follow the documented practices without adopting it in his particular situation. In addition, it is ideal if the Scrum Master also knows about the organizational change. The Scrum Master is responsible for removing impediments in the project. In cases where the impediments come from outside of the team, namely from the organization, knowledge in this aspect becomes highly useful.

The Scrum Master of the IT Quality team had never been in a Scrum project before. He had many difficulties in dealing with the communication in the team. Nothing from the procedures of Scrum was missing, the Planning Meeting, the Daily Standup, the Retrospective, etc. However, communication became worse than when Waterfall was used.

“Communication in the true sense went wrong” (IT Quality team 2013)

The situation may have been improved if he had found a best practice for handling communication that fitted in the team.

In reality, there are Scrum projects that follow the guidance of a dedicated Scrum Coach, who has extensive knowledge and experience with Scrum. Organizations may establish an Agile team/center to support Agile projects. Additionally, organizations can also hire consultants from outside to help them run Scrum projects. Bringing outsiders inside can have certain benefits, including different perspectives and independency from the old organizational culture (which can block out any change). With decent support from Scrum professionals, the importance of the knowledge requisite of Scrum Master may lessen.

Nonetheless, the facilitator role of the Scrum Master cannot be underestimated. Whether a project can run smoothly depends a lot on his facilitation. The Scrum Master must be able to provide help to the Team when necessary. Ideally, a Scrum Master should possess some coaching skills and the ability to motivate others.

The Team

A Scrum Team needs a certain maturity level and sufficient capacity to adopt Scrum. This requisite came up in half of the interviews. Team maturity is very important for the reason that Scrum employs the self-management concept. In Scrum, unlike Waterfall, work is not imposed on the Team members. Instead, they are encouraged to pick the tasks they want and request help from others if needed. If the Team members, for any reason, choose not to approach work proactively, or not to support others, it is very likely that Scrum will fail.

“In order to be successful, a Scrum team needs a certain level of maturity and self-containment. As there is no hard time management or fixed mile stones defined, deliverables may shift in time more easily than in a strictly top-down managed project”. (IT Quality team 2013)

Maturity is also necessary to embrace in the most honest way the concept of visibility. Lander suggests that in Waterfall, it is easier to hide; however not in Scrum. Scrum insists that everyone must be open, showing what he or she is working at to others, ask for and provide help whenever needed. For this to be possible there must be a sense of trust within the Team. Team building activities, coaching sessions and such are common means of improving team maturity. Among other factors, time is needed to achieve this. The time factor is studied later in its own section.

“...there is a great dependency on the team to embrace it [Scrum] to the core. The team needs to realize that it as a group is responsible to deliver the product.” (IT Quality team 2013)

Scrum promotes the attainment of multiple skills by Team members. This requires a certain capacity of the Team members. Scrum teams are set up with members from different functions with different sets of skills. Team members can pick the tasks requiring different skills not part of their expertise. For example, during a Sprint, software developers can work on either development or testing. One can

question the feasibility of meeting this aspiration, since learning new skills may take time and require effort from both the learner and the helper. In scenarios where Team members are highly specialized, or there is a significant disparity between the experience and skill levels of the team members, it may not be possible to follow the collaboration principle of Scrum.

“...success factor in Scrum: open and collaborative atmosphere in the Scrum team and a strong team of equals” (IT Quality team 2013)

This requirement suggests that, in any case, the Scrum Team members should be open to learning new knowledge and skills aside from their already acquired expertise, as well as teaching what they know to others. All these aspects must be taken into account when forming a project team.

“I do not only look at the skills, but also to their Scrum values... if the person is willing to learn and feel accountable for the whole product as a team, or if he only looks at his area of responsibilities”

Lander mentioned in his interview the tactic of using certain criteria in interviewing candidates for setting up a Scrum team, such as openness, willing to learn and the sense of responsibility for others' work rather than just his own. In addition, people with “single job description” (one skill) should be motivated to “do more for the team”, that is to learn more and take up responsibilities from other team members to help enhance the overall performance.

Application factors

Application factors refer to the exhaustive understanding of Scrum that can lead to a proper application. In other words, any team using Scrum must be certain to understand why and how it works, to the extent that they can make appropriate adaptation without breaking the unchangeable principles of Scrum.

Scrum is not “the one that can solve everything”. Scrum practices must be adjusted to specific situations. However, a modification that denies the heart of Scrum is no longer Scrum. On the other hand, blindly adopting Scrum without taking into consideration the unique circumstances is as likely to lead to failure. To avoid these situations, Scrum practitioners must know what should be changed and what not.

“One success factor of Scrum is the level of understanding of the "why" of Agile and Scrum, which enables people to get an agile mindset, instead of just doing Scrum without really understanding it” (Linders 2013)

The application factors presented herein entail the most problematic aspects, as either observed in the IT Quality team, or discussed during the interviews. They are the Product Backlog, Time, Sprint length, Team structure and Adaptation, namely.

Product Backlog

A Product Backlog should contain clearly defined items, following the Scrum guideline. Many teams make the mistake of filling in the Product Backlog in a Waterfall way, e.g. the Product Owner defines all the requirements without face-to-face conversations with the Team (Lander 2013). It can be easily seen that utilizing this approach is much pointless, as it completely blocks the science of Scrum, which employs a strong collaboration between the Product Owner and the Team to decide what to do and how to do it. A Product Backlog deprived of the contribution from either party may very likely end up impractical. If a Product Backlog comes only from the Product Owner, the product might look nice on paper but not feasible technically. On the other hand, missing the evaluation from the Product Owner, it might guide the Team to producing a different product from the stakeholders' expectations.

The IT Quality team had to struggle to create the Product Backlog and consequently the Sprint Backlogs. It happened very often that the tasks planned in each Sprint were not finished. In an extreme case, there was no item done in a Sprint. An informant from the team stated: “...it [Scrum] requires clearly defined product backlog items”. Additionally, when asked for comments on such cases, most informants believed that the Product Backlog items were probably defined wrongly. In any case, the importance of a well-defined Product Backlog in Scrum is undeniable, since it serves as the roadmap for the whole project.

Time

The Time factor is extremely important in any project. Time is also a classic constraint in management and must be consumed efficiently and effectively. Rushing, in case of lacking time, is unwise. When a person is new to Scrum, he

needs sufficient time to understand and adapt to its practices. Linders also includes “the time that a team takes to reflect upon how they use Scrum, and how they adapt their way of working” in his list of Scrum success factors.

Furthermore, estimating the effort needed for the tasks contains a certain level of imprecision. Therefore, the Team should be allowed roughly three Sprints to improve their estimates. Especially in cases of organizations new to Scrum, it is important to reserve time for organizational changes to take place. One common practice is to add extra time for contingency in Sprint planning, depending on the situation. The team may reserve time for extra communication, coaching and other non-value-added activities. (Lander 2013.) Understandably, this can reduce the risk of overrunning a project.

Sprint length

The Sprint length is dependent on the characteristics of each project. Nonetheless, it should be kept unchanged throughout the project. The total time and the complexity of the project usually decide the right length of each Sprint. Already mentioned in Chapter 2, a Sprint can last from one to four weeks. Too short a Sprint might be too demanding and not allow the completion of the requirements. Too long a Sprint prevents deviations when needed and takes away space for feedback loops. Short feedback loops that embrace flexible scopes is the heart of Scrum and must be embedded in any project. No matter what, failing in timing Sprints properly may increase overheads significantly or even introduce failure.

According to Lander, the purpose of the unchangeable fixed-time boxes is to keep a stable rhythm for Scrum. “The rhythm is key to Scrum” – it is very important to let the Team get used to doing their work in cycles and in similar setups every time. For instance, things must always be done in four weeks; repeatedly it creates an engine for the Team.

“Four weeks is the maximum time to freeze changes (to prevent external changes to the work inside the Team) and it also creates a really good rhythm” (Lander 2013)

A conversation with a veteran Project Manager in engineering product development revealed the difficulty in keeping the fixed time boxes. This Project Manager has always been using an approach combining Agile principles and

Waterfall, not Scrum. He runs projects in iterations with different lengths, e.g. one month and two months. The iterations do not have fixed end date, instead is run until the desired functionality is completed. This approach is in fact very different from Scrum. Nevertheless, his experience creates a question about the scenarios wherein Scrum teams employ different iteration lengths for their Sprint cycles. Some informants were consulted about this problem and suggested using other frameworks in engineering projects, such as Lean Management or Agile Kanban whose iterations do not have fixed length. No matter what, all agreed that fixed time boxes in Scrum is absolute and cannot be changed.

Communication

Communication is a pivotal pillar of Scrum. There are many requirements in communication, yet the most frequently mentioned during the interviews is that of regular communication including a high level of face-to-face interaction. Scrum encourages team members to interact in person and help each other. The office facility should be arranged in such way that supports this aspiration. A project open work area with stand-up meeting space instead of unnecessary cubicles helps promote spontaneous personal interaction and increases visibility in the office. International corporations need sufficient telecommunication facility. Teams such as the IT Quality team wherein the team members are scattered in different geographical locations should spend additional effort and time on communication to compensate the deprivation of face-to-face daily conversations. The informants from the team all addressed the problem of lacking face-to-face spontaneous contacts due to the dispersion.

“...the different locations is not improving this process [Scrum]”; “the dispersion of the team over two locations... hampered productivity a lot”; “critical success factors for Scrum are full dedication of project resources and full location in one place” (IT Quality team 2013)

A possible solution is to have a virtual environment mimicking real-life office features, or high-quality video meeting rooms with optimal realism. It must be noted that these means may increase costs significantly. Telecommunication however is inevitable nowadays, since outsourcing and international trades are increasingly common. Lander suggested that there are cases wherein the organizations must evaluate the trade-off between cost reduction from outsourcing

and cost increase from lower productivity. It is important not to forget the opportunity costs, which are not measurable in monetary term.

Team structure

There must be no sub-division within the Team. As stated in preceding sections, members from different functions with different sets of skills often constitute a Scrum team. Scrum teams wherein having highly specialized team members is unavoidable, may adopt best practices to promote learning from one another. Without this continuous improvement process at individual levels, Scrum loses much of its power, which must be avoided consistently.

“I have resources who are mostly specialists. Each does a job based on his or her expertise, and as the architecture is laid out, much of the work of the individuals is also laid out” (M.G. 2013)

The quotation above describes a situation where the project management framework is effectively Waterfall, as the tasks fall upon individuals instead of being left to their own choices.

When the Team is sub-divided, or when it is not self-contained, work is passed between teams, introducing wasteful use of resources (Lander 2013). For example, when testing cannot be done in parallel with development, instead only after the product is completed, more time is wasted until necessary changes take place.

In summary, a Scrum team must enforce strong collaboration and establish teamwork culture, as the team is more than a combination of different individuals. Scrum is about one team, dedicated for the job and knowing from each other what to expect.

Adaptation

A team must adapt Scrum practices appropriately without violating its core principles. When necessary, a Scrum project can employ best practices from other frameworks to match specific situations.

“Every method has limitations and the key is to know what they are and how to overcome them. Every method needs to be adapted to circumstances rather than blindly adopted, with the

caveat that you do not negate the core values in your adaptation” (Cooper 2013).

Cooper’s argument is effectively supported by Linders suggesting that the freedom that a team has to adopt it to their needs is also a success factor of Scrum.

There are many common processes in project management the Scrum guide does not cover. These include Stakeholder Management, Financial Management, Release Planning, Business Case Management, Financial Management, Project Start up, Project Close Down, Resource Management and such. Some practitioners mistakenly exclude these processes from the projects, assuming Scrum projects do not need them, whereas others struggle to incorporate Scrum into the projects. Scrum practitioners must follow the Scrum guide as well as common sense. Scrum can and should be combined with common "best practices" as long as these are not conflicting. (Lander 2013.) On the other hand, Cooper offered an interesting solution to “Scrumize” other processes with the example of Risk Management.

“Traditional risk management has the steps of Identify, Assess and Analyze, Plan Action, Monitor and Implement, and Measure and Control. In Scrum, once we create the backlog we can then do a Risk Adjusted Backlog, Risk Spikes, and Risk Burndown Charts”. (Cooper 2013.)

Cooper’s argument revealed another way to look at frameworks, projects and processes than trying to find out what is required for the application of one framework to be successful. Due to the limitation of the research, this point of view, though intriguing, will not be discussed further.

Organization factors

These factors concern the organization on a larger scale than the Scrum team itself. In his book, Schwaber (2004, 40-41) describes a case where he failed at running a successful Scrum project by forgetting the importance of organizational culture. In fact, the Scrum team is not operating by itself; it also needs to interact with other parts of the organization. Consequently, attention must be paid to the organizational environment encompassing the project.

An organization new to Agile/Scrum needs a certain level of coaching, mentoring, and sufficient time to immerse itself in this mindset.

Scrum is something that looks simple, but can be hard to do. Supporting people in “why” and “how” can be crucial (Linders 2013).

The organization needs to be open to change and willing to go Agile. For this to be possible, the organization requires experienced coaches who understand and can direct the organization changes properly. The changes should be taken step-by-step in a steady pace. (Lander 2013.)

As Scrum blocks any changes during a Sprint, it is essential to avoid managerial interference in the work of the Team. This is not easy if the management level does not understand and accept the Agile/Scrum principles. The Scrum team needs support from the management level to operate Scrum seamlessly.

According to the Seventh Annual “State of Agile Development” survey (2012, 7), conducted by VersionOne Inc., the inability to change the organization’s culture was the number-one barrier to further adoption of Agile in the surveyed firms. Number two was the general resistance to change. Furthermore, among the top-five leading causes of failed Agile projects, three of them come from the organizational aspect. These reasons include company philosophy or culture at odds with core Agile values, failure to integrate people and to teach team-based culture, and lack of cultural transition. These results suggest the magnitude of organizational changes to the success of Scrum application, as well as the difficulty in solving this problem.

External factors

The external factors encompass the customers and other external stakeholders. These actors need to comprehend the power of Scrum and provide cooperation to the project team. Scrum insists on constant feedback loops between these parties; without collaboration, Scrum is meaningless. In many cases, educating the stakeholders about the science of Agile/Scrum can strengthen the relationship between two parties and bring significant benefits. Otherwise, external pressure to follow traditional Waterfall processes is likely to destroy the project. This is, unfortunately, the third common reason for failing an Agile project (VersionOne 2012, 6).

It can be seen that to have a successful Scrum project is rather challenging. Many elements from different aspects must be in place for a project team to secure success. Scrum is indeed not easy, yet its benefits can be invaluable. The research findings do not indicate that all of the abovementioned factors absolutely must be obtained. There is always the possibility to make trade-off, or to improvise and adjust to particular circumstances in reality. Notwithstanding this, the significance of these requirements should not be overlooked, as the evidence already suggested.

The second objective of the research is to reveal the forms of project and process whose nature is appropriate for Scrum values, and those whose is not. The data collected through the research provided certain results satisfying this aim. The next section will discuss these findings in detail.

What project types benefit from the application of Scrum?

Fascinatingly, seven out of twelve informants believed that Scrum is beneficial to any kind of projects. Among these, four are Scrum experts from Group 2. In reality, the informants have been involved in or have observed the application of Scrum to a vastly wide range of projects. These include Sales, Marketing, Engineering, Research projects. The other opinion was that Scrum is in some cases less beneficial than other frameworks. Three informants from Group 2, when asked specifically, pointed out situations wherein Scrum might not work best for various reasons. Only one criticized certain aspects of Scrum that do not fit in his field. These perspectives will all be presented in this following section to provide a complete view of the subject and avoid bias.

Projects with high level of complexity

These projects are most favored by Scrum. As Scrum (and other Agile approaches) were developed owing to the increase of these projects, they are naturally the best match for each other. Most Scrum practices are to address the uncertainty that obstruct advance planning and increase productivity and nurture creativity. It was already revealed in Chapter 2 that this is also a widely supported belief in the Scrum community.

When is Scrum less beneficial?

Projects with low level of complexity

When everything is known and can be planned, Scrum might not thrive. The reason lays in Scrum's possible-to-be large overhead. Since Scrum introduces short feedback loops and extensive communication between all stakeholders, it consumes substantial resources. Compared to other solutions, it might be more costly. Simple projects, whose processes are well known and understood, might work much better with Waterfall in fact. This argument was explicitly supported by two informants.

However, it is difficult to gauge the complexity of a project for Scrum. One possible solution is to position the project in the Stacey diagram. Stacey diagram is a map of complexity zones based on the degree of certainty and level of agreement on the issue in question. It can help select the appropriate management actions. Due to the limitation of the thesis, this tool is not discussed, yet can be found in Appendix E.

Projects with high architectural requirements and/or low margins for change

These projects might not be able to incorporate Scrum principles. This, again, points to the conversation with the engineering Project Manager.

For complex engineering development, much of the design and documentation must be done in the early stages of the project (M.G. 2013)

Lander, when asked for comment on this, also agrees that engineering projects usually require more planning than those of IT software development do. When a project has much advance planning, it might change the lightweight nature of Scrum. Too much planning effectively falls back to traditional management, even when Scrum properties are still integrated in the processes.

On the other hand, projects in which it is very costly to make any change may also need a detailed design in advance. To give an example, constructing buildings seems to deny changes in the procedure. The physical structure must be precisely erected to ensure its safety, hence is not easily changed.

In these cases, it might be a better solution to consider another approach than Scrum, not excluding variants of Scrum (Scrum combined with other frameworks).

An interesting fact is that Scrum has been used by NASA, FBI, the US Department of Defense and in hospital systems (Lander 2013). It is, however, unknown to the researchers as to how Scrum was applied, in what projects. Nevertheless, as these organizations usually demand highly advanced products with large expenses, their examples might indicate that the constraints of high architectural requirements and low margin for change can be overcome with a Scrum approach.

Processes with purely operational work

Purely operational work might find better fit than Scrum. This is an argument emerged in the literature reviews under subheading 2.4. The assumption is that operational work does not require high level of creativity and problem-solving activities, therefore Scrum's benefits are not fully exploited. By indicating her preference for Kanban over Scrum in Operations, Warmolts (2013) effectively supported this argument.

Common project management processes without clearly defined deliverables

In these cases, combining best practices from different frameworks might be a better solution than only using Scrum. Examples of these types of process include Risk Management, Stakeholder Analysis and Management, and Change Management. When asked for comments on this argument, an informant from Group 1 explained the reason as these processes are “strongly driven by interactions with project environment and events in it (corporate strategy, stakeholders, customers, regulations etc.)”. In fact, as discussed in the preceding sub-chapter, Scrum does not provide guidelines for these processes. Therefore, there is no standardized Scrum approach here. This does not necessarily indicate that Scrum is not beneficial for them; it only suggests that Scrum practitioners may choose what works best for their particular situation.

Chapter 4 completely fulfills the initial objectives of the research, as all the research questions have been answered at this point. The findings encompass the

factors required for the success of Scrum in a project, as well as the projects and processes wherein Scrum is beneficial in different levels. These findings are generalizable, meaning they are applicable for multiple situations. The research therefore can be considered successful.

5 CONCLUSIONS AND SUMMARY

This chapter closes the thesis. It provides a summary of the findings and an evaluation of the research. Further recommendations for future studies are also presented.

The research problem emerged from a working situation. The field researcher was working for a project team when this team decided to change their project management framework. The new framework is Scrum, adopted in place of the traditional Waterfall approach. Yet the results of this change did not live up to expectations. The researchers assumed the reasons to be critical success factors missing from the project or the nature of the project not lending itself to Scrum principles. Then what is required for Scrum to thrive? What projects and processes can benefit from Scrum?

Project Management is an important and interesting field within business, whereas Scrum as a Project Management framework is getting increasingly popular. Scrum was first developed for software development. Due to its phenomenal success, Scrum is now being used in a wide variety of projects. The research problem is therefore very intriguing. The research aimed to find out the success factors of Scrum, their assisting actions and the situations (projects and processes) where Scrum is beneficial. The findings were expected to be relevant to other situations besides the case study.

After the research questions were established, the research studied a theoretical framework surrounding Project Management standards (as defined by the PMI) and its approaches: the traditional one (Waterfall) and Scrum. Their practices and their core values were presented and compared in Chapter 2. The literature review of the thesis covered most areas of the research. Some unexpected parts emerged during the research had to be left out to prevent a scope creep.

The research was designed as a qualitative case study, which lasted five months. The empirical evidence was collected by means of observation and interviews. The approach and methods were described in-depth in Chapter 3. Data came from the case organization and from external experts with broad knowledge and experience in Project Management and Scrum. The purpose of this inclusion was

to generate generalizable findings and increase the reliability of the research. Interviewing and evidence analysis were done in parallel to maintain a continuous learning process and make it possible to elaborate interesting issues when needed. The analysis and findings were presented in detail in Chapter 4. The key findings of the research are summarized in Table 2 and Table 3.

Table 2 shows the suggested Scrum success factors and their assisting actions, based on the empirical research. These findings represent the most important aspects of the existing factors in reality. The categories are color-coded for easy reading.

TABLE 2: Scrum success factors and assisting actions


SUCCESS FACTORS	ASSISTING ACTIONS
PEOPLE FACTOR	
General	
Dedication and commitment	All resources need to be fully engaged and committed to Scrum.
Openness and willing to change	Team members need to follow the Scrum principles not only in name, but really want to embrace the Scrum mindset and change the work culture (from Waterfall). They can be educated and motivated to do so.
Product Owner	
A clear vision (to understand thoroughly the product, its requirements and the value of each requirement)	Sufficient communication must be enforced, including face-to-face conversations between Product Owner and the stakeholders, Product Owner and the Team.
	Product Owner needs enough experience and knowledge in order to evaluate the values of each requirement.
Full understanding and fulfilling of the responsibilities, sufficient mandate to do so	The roles of Product Owner should not be shared with another role.
	Product Owner must be able to make decisions, act quickly in response to changes and be available for the team members when needed.
	Cooperation between all involved parties is required.

Scrum Master	
The ability to facilitate the project and prevent external interference to the Team	Scrum Master has enough knowledge and understanding of Scrum and organizational behavior.
	Scrum Master should possess some coaching skills and the ability to motivate others.
	A team new to Scrum should have a dedicated Scrum Coach to support them when necessary.
Team	
Multiple skills and/or learning ability	These criteria need to be taken into account when establishing a Scrum team.
	When forming a team, the members can be interviewed to evaluate if they are fit to Scrum.
Sufficient maturity and capacity	Team members are educated to understand and able to follow the self-management concept.
APPLICATION	
General	
Adjustment of Scrum to specific situations without breaking its core values	There must be a thorough understanding Scrum, what should be changed and what not, "why" and "how".
Product backlog	
Clearly defined items following the Scrum guideline	Product Owner has to transfer his knowledge to the Team, through face-to-face conversation and regular communication, in order to make them understand clearly what the product needs.
Time	
Sufficient time to adapt to Scrum and to understand thoroughly the principles	First three Sprints are needed for the Team to increase precision in estimating. There should be no rushing or forcing.
	Extra time should be reserved for extra communication, planning, coaching (contingency)
Sprint length	
A suitable sprint length	The Sprint length is between one to four weeks. It must be decided based upon the particular situation, should not be too short or too long.
Unchanging Sprint length to maintain the Scrum rhythm	A Sprint needs to be ended as planned. Changing of Sprint length needs good reason. In any case, it should be kept stable.

Communication	
Regular communication includes high level of face-to-face interaction	Team members are encouraged to interact in person and help out each other
	The office should have open work areas that encourages face-to-face interaction with stand-up meeting spaces; or telecommunication mimicking the same features
	Closed cubicles should be eliminated when not necessary.
	Dispersed teams may need extra communication and sufficient telecommunication facility. Meetings with videos or virtual simulation of office room over distance can be used in these cases.
Team structure	
No sub-divisions within the team	Scrum favors cross-functional teams, ideally self-contained. Sub-dividing the team and passing work between teams must be avoided.
Cultivation of teamwork culture	A Scrum team must enforce strong collaboration and establish teamwork culture, as the team is more than a combination of different individuals.
Adaptation	
Combination of Scrum practices with best practices without violation of its core principles	In the Scrum guides, there are processes not covered by Scrum, it is therefore necessary to manage such processes with suitable practices and follow "common sense".
ORGANIZATION	
Organization	
Openness and willing to change	Sufficient time should be reserve for the organizational change to take place.
	Organizations new to Scrum need experienced coaches who understand and can direct properly the organizational change.
	Organizational change should be taken step-by-step.
Sufficient executive sponsorship	Management level has to understand and agree with the Scrum principles and give the Team enough support.
EXTERNAL	
Stakeholders/ Customers	
Cooperation and common understanding of the Scrum principles	Stakeholders need to be educated about the Scrum principles.

The second set of findings is the project and process types that can benefit from using Scrum. The research evidence suggested that it is not possible to draw a line and define clearly when Scrum is beneficial and when not. The researchers therefore arranged the projects and processes on a beneficial scale. The green head of the scale indicates project types where Scrum's benefits are best exploited. The orange tail indicates areas where Scrum is less beneficial. However, it does not mean Scrum cannot be applied here. This scale is depicted in Table 3.

TABLE 3: The benefit levels of Scrum to projects and processes

Possibly, almost any kind of project can use Scrum. A way to look at it is to see how a project or a process can adapt Scrum principles. Successful examples include Sales, Marketing, Engineering, Research projects, etc.	
SCALE	PROJECTS AND PROCESSES
 <p>Scrum is most beneficial</p> <p>Scrum is less beneficial</p>	Projects with high level of complexity and/or uncertainty (a typical example is Software development projects)
	Projects with high architectural requirements and/or low margins for change, such as Engineering projects, Construction projects
	Processes not covered in Scrum guide, including Stakeholder Management, Financial Management, Release Planning, Business Case Management, Resource Management etc.
	Processes with purely operational work
	Projects with low level of complexity, everything is known and can be planned

Evaluation of the research

The researchers self-evaluated the research and gave it an overall good rate. The research questions were answered with high level of satisfactory. The planned research scope was completed and the objectives were achieved. The researchers found the findings highly interesting and useful. The knowledge gained from the research was invaluable.

The validity of the research was assessed by judging the following aspects: construct validity, internal validity and external validity. Construct validity was obtained, as the research methods and tools used were completely suitable to attain the research objectives. The findings required in-depth data acquisition and analysis, as project management is a complicated field with behavioral nature. Any assessment taken at face value can be deceptive. Therefore, the qualitative approach was the right choice as it allows in-depth examination of the subjects in

question. The data collection happening in parallel with data analysis made the research a systematic flow. There was a continuous learning and improvement process. The research plan was constantly evaluated and changes were made in alignment with new realizations. The case study enabled the researcher to gain insights into a situation that happened in real life, which is essential for a business study. The interviews were extremely helpful as the informants possessed broad knowledge and extensive experience in a wide variety of circumstances. The evidence contained interesting elements and was analyzed thoroughly.

Triangulation was taken care of as evidence from different sources was compared and studied with a holistic point of view. A number of comments were surprising to the researchers, as they shared the unique experience of some informants. On the other hand, other comments confirmed the initial assumptions of the researchers regarding the findings. Consequently, the analysis was done carefully to include different perspectives and avoid bias. Overall, the approach employed in this research did measure what it was meant to measure.

The internal validity requirement was also satisfied. The theoretical framework was the basis for the empirical research. The knowledge about Project Management, Waterfall and Scrum was fully utilized in analyzing the empirical evidence and explaining the findings. The literature review also included debates about Scrum application on the internet, which was a rather important part. This part offered basic assumptions of what the findings should look like. The generalizations were inferred from evidence provided by different sources. Information from the interviews was linked to the observations whenever possible, as shown in Chapter 4. The researchers discussed their assumptions and other opinions with the informants to understand the subjects better, and to avoid premature conclusions.

The external validity was considered obtained. This research carries the nature of a social study; therefore testing its validity in multiple contexts is very difficult. However, by including the informants from different backgrounds and organizations, the researchers were able to create appropriate generalizations for multiple situations in reality. The topic itself is rather controversial; it is not possible to include all possibilities. The researchers therefore chose to support the propositions in line with the theoretical framework. One's opinion should be

respected and should not be disregarded, as it might never be possible to label an idea right or wrong in social studies.

The reliability of the research was acquired at an appropriate level. The findings hold a good level of consistency. Each of the generalizations was generated only when it had supporting evidence from more than one source. In their interviews, informants were asked to share their experience in past situations to ensure the practical feature of the data. The findings should remain similar when a similar research is conducted with similar protocol. The researchers looked at the results from different angles and realized they can be true with a tolerable level of uncertainty.

Limiting factors in the research

The first limiting factor concerns the small number of interviews conducted. Although the interviews gave good insights into the subjects, the findings might still contain biases. There exists certain information the researchers could not confirm with the chosen informants. If more time and resources had been available, the researchers could have reached a larger sample population and examined more areas of the subjects.

The second limiting factor emerged from the short observation period. The field researcher wished to spend more time in the Scrum project to gain more experience. The limited knowledge and experience in Project Management and Scrum hampered the escalation of the analysis and the findings. In addition, the researchers encountered other points of view, suggesting divergences from the findings, during the research. Yet these perspectives were not investigated due to the short time available.

Recommendations for future research

The researchers realized that more knowledge about Agile methodologies and other project management frameworks should have been acquired. Agile methods are commonly referred to as a whole, while Scrum is currently the most popular representative. Scrum is very often combined with other Agile practices, as it should be. Sole knowledge of Scrum restrained the researchers from relating the findings to these variants.

The recommendations include further studies of other methodologies and frameworks, and other ways of looking at project management. A study of Kanban or PRINCE 2, for example, can bring interesting solutions for the constraints presented amid the results. Some informants also noted their success in doing this; therefore, this area of study is very promising.

As mentioned before, there exist other ways of dealing with Scrum and Project Management than was discussed in the research. For instance, one informant suggested that instead of deciding which project processes Scrum can or cannot be used for, Scrum practitioners can decide how these processes can adapt to the new framework over Waterfall. The researchers found this idea very intriguing. It is an attractive area and should be explored further.

REFERENCES

Published references

Attarzadeh, I., Ow, S. 2008. The Project Management Practices: The criteria for success or failure. Kuala Lumpur: University of Malaysia.

Chavrat, J. 2003. Project Management Methodologies. New Jersey: John Willey & Sons, Inc.

Cleland, D., Gareis, R. 2006. Global Project Management Handbook. New York: McGraw-Hill Professional.

Hennink, M. Hutter, I. & Bailey, A. Qualitative Research Methods. London: SAGE Publications Ltd.

Kerzner, H. 2009. Project Management: a system approach to planning, scheduling and controlling. 10th edition. New Jersey: John Willey & Sons, Inc.

Remenyi, D., Williams, A., Money, A. & Swartz, E. 1998. Doing Research in Business and Management: An Introduction to Process and Method. London: SAGE Publications Ltd.

Royce, W. 1970. Managing the Development of Large Software Systems: Concepts and Techniques. Technical Papers of Western Electronic Show and Convention (WesCon) August 25-28. Los Angeles.

Schwaber, K. 2004. Agile Project Management with Scrum. Washington: Microsoft Press.

Stacey, R. 2002. Strategic management and organisational dynamics: the challenge of complexity. 3rd ed. Harlow: Prentice Hall.

Sutherland, J. 2010. Jeff Sutherland's Scrum Handbook. Somerville: Scrum Training Institute.

Electronic references

Barron, M., Barron, A. 2009. The Project Life Cycle [referenced 16 January 2013]. Available in Connexions database: <http://cnx.org/content/m31913/latest/>

Berteig, M. 2010. What is Scrum good for? Agile Advice [referenced 18 March 2013]. Available on <http://www.agileadvice.com/2011/10/25/scrumxplean/what-is-scrum-good-for/comment-page-1/#comment-11442>

Boruff, G. 2011. Traditional Project Management vs. Agile Project Management [referenced 10 February 2013]. Available in Management Solution, LLC database <http://managementsolutionsllc.com/blogs/projectmanagement/Lists/Posts/Post.aspx?List=aff26cdf-3c42-436f-95fb-32d5ece0a703&ID=25>

Cohn, M. 2011. Deciding What Kind of Projects are Most Suited for Agile. Mountain Goat Software [referenced 12 March 2013]. Available on <http://www.mountaingoatsoftware.com/blog/pdf/deciding-what-kind-of-projects-are-most-suited-for-agile/comments>

CPrime. 2012. An Agile Case Study - Wiredrive and Agile Development [referenced 20 March 2013]. Available on <http://www.cprime.com/articles/agile-case-study-wiredrive.html>

Cprime. 2012. What is Agile? What is scrum? Agile and scrum definitions [referenced 10 February 2013]. Available on <http://www.cprime.com/help/what-is-agile-scrum-definitions.html>

Douglas, H. 2009. The Traditional Waterfall Approach [referenced 25 February 2013]. Available in <http://www.umsl.edu/~hugheyd/is6840/waterfall.html>

Frese, R. & Sauter, V. 2003. Project success and failure: What is success, what is failure, and how can you improve your odds for success? [referenced 10 January 2013]. Available in http://www.umsl.edu/~sauterv/analysis/6840_f03_papers/frese/

GNSE Group. 2012. Development Applications [referenced 08 February 2013]. Available on <http://www.gnsegroup.com/profile/methodologies.aspx>

Hass, K.B. 2010. Evolving from Traditional Project Management to Agile. Executive Brief [referenced 09 January 2013]. Available on <http://www.executivebrief.com/agile/traditional-project-management-agile-evolving/>

International Institute for Learning. 2013. About Dr. Harold Kerzner [referenced 20 February 2013]. Available on <http://www.iil.com/Kerzner/about.asp>

Kniberg, H. 2012. Agile Product Ownership in a Nutshell [referenced 15 January 2013]. Available on <http://youtu.be/502ILHjX9EE>

Mc Gevna, V. 2012. A More Agile Approach to Waterfall [referenced 15 January 2013]. Available on <http://youtu.be/eV3Xq-ocCYY>

Miles, T. 2012. A Brief History of Project Management. [referenced 21 March 2013]. Available in Mavenlink database: <https://www.mavenlink.com/community/blogs/1465-a-brief-history-of-project-management>

PMI-ACP Guide. 2013. Available on <http://acpguide.org/2011/09/07/traditional-management-vs-agile-project-management/>

Project Management Institute. 2013. About Us [referenced 11 February 2013]. Available on <http://www.pmi.org/About-Us.aspx>

Project Management Institute. 2013. PMBOK Guide and Standards [referenced 10 March 2013]. Available on <http://www.pmi.org/PMBOK-Guide-and-Standards.aspx>

Project Management Institute. 2013. What is Project Management? [referenced 15 January 2013]. Available on <http://www.pmi.org/About-Us/About-Us-What-is-Project-Management.aspx>

Project Management Methodology Guidebook. 2003. City of Chandler, Arizona [referenced 14 February 2013]. Available on <http://www.chandleraz.gov/Content/PM000PMMethodologyGDE.pdf>

Project Manager. 2011. From Waterfall to Agile [referenced 10 March 2013]. Available in <http://projectmanager.com.au/education/methodologies/waterfall-agile/>

Scrum Alliance. 2012. Strategic Plan 2012. Scrum Alliance Inc [referenced 15 January 2013]. Available on http://www.scrumalliance.org/resource_download/2865

Scrum Alliance. 2013. Courses [referenced 15 February 2013]. Available on http://www.scrumalliance.org/events/results?search%5Bevent_type%5D=course

Scrum Alliance. 2013. Scrum Is an Innovative Approach to Getting Work Done [referenced 15 February 2013]. Available on http://www.scrumalliance.org/pages/what_is_scrum

ScrumLogic. Available on <http://scrumlogicinc.com/images/Scrum-Cheat-Sheet.pdf>

Search Software Quality.2007. [referenced 20 February 2013]. Available in <http://searchsoftwarequality.techtarget.com/definition/waterfall-model>

Select Business Solutions. 2013. What is the Waterfall Model? [referenced 10 March 2013]. Available in <http://www.selectbs.com/analysis-and-design/what-is-the-waterfall-model>

SQA Project management for IT. 2007. The project triangle [referenced 15 January 2013]. Available on http://www.sqa.org.uk/e-learning/ProjMan03CD/page_13.htm

Stephen On Software.2010. [referenced 20 February 2013]. Available in <http://www.stephenonsoftware.com/2010/11/winston-w-royce-father-of-waterfall.html>

Tech republic. 2006. Understanding the pros and cons of the Waterfall Model of software development [referenced 15 March 2013]. Available in <http://www.techrepublic.com/article/understanding-the-pros-and-cons-of-the-waterfall-model-of-software-development/6118423>

The Agilista PM. 2013. [referenced 25 February 2013]. Available in <http://www.agilistapm.com/differences-between-waterfall-iterative-waterfall-scrum-and-lean-software-development-in-pictures/>

Thompson, K. 2009. When Scrum is too much. Deep Scrum [referenced 15 March 2013]. Available on <http://deepsrum.wordpress.com/>

TutorialsPoint. 2013. Project Management Triangle [referenced 14 January 2013]. Available on http://www.tutorialspoint.com/management_concepts/project_management_triangle.htm

VersionOne. 2012. 7th Annual State of Agile Development Survey [referenced 25 March 2013]. Available on <http://www.versionone.com/state-of-agile-survey-results/>

Williams, L. 2007. A Survey of Agile Development Methodologies [referenced 10 February 2013]. Available on <http://agile.csc.ncsu.edu/SEMaterials/AgileMethods.pdf>

Xebia. 2012. Agile survey Nederland [referenced 18 March 2013]. Available on <http://agile.xebia.nl/nl/content/agile-0>

List of interviews*Group 1- IT Quality team 2013*

Belgudri, V. 2013. Business Analyst. . HyperElectronics, IT Quality team.

Interview 26 February 2013

Dr. Koch, W. 2013. Manager IT Quality. HyperElectronics, IT Quality team.

Interview Feb 26, 2013

Jakubiuk, P. 2013. Performance Manager in IT Quality. HyperElectronics, IT

Quality team. Interview 26 February 2013

Keukelaar, P. 2013. Quality Management Expert. HyperElectronics, IT Quality

team. Interview 26 February 2013

Group 2

Borselaer, M. van 2013. Project Manager, Coach. Interview 17 March 2013

Cooper, L. 2013. Project Management Mentor. PMI-OVOC. Interview 3 March 2013

Dridi, M. 2013. IT Consultant. Global Accounting and Consulting firm. Interview 28 February 2013

Lander, A. 2013. Senior Enterprise Agile Scrum Transformation Coach. Interview 16 March 2013

Linders, B. 2013. Senior Consultant Quality, Agile, Lean & Process Improvement. <http://www.benlinders.com/>. Interview 3 March 2013.

M.G., V. 2013. Engineering Project Manager. Interview 12 March 2013

Warmolts, S. 2013. Agile Project Leader, Independent Consultant. Interview 12 March 2013

Willemsen, B. 2013. Agile Coach. HyperElectronics. Interview 27 February 2013

APPENDICES


Appendix A - Scrum definitions

Item	Definition
Burndown graph	The trend of work remaining across time in a Sprint, a release, or a product. The source of the raw data is the Sprint Backlog and the Product Backlog, with work remaining tracked on the vertical axis and the time periods (days of a Sprint or Sprints) tracked on the horizontal axis.
Chicken	Someone who is interested in the project but does not have formal Scrum responsibilities and accountabilities (is not a Team member, Product Owner, ScrumMaster, or other stakeholder).
Daily Scrum meeting	A short status meeting held daily by each Team during which the Team members synchronize their work and progress and report any impediments to the ScrumMaster for removal.
Done	Complete as mutually agreed to by all parties and conforming to an organization's standards, conventions, and guidelines. When something is reported as "done" at the Daily Scrum or demonstrated as "done" at the Sprint review meeting, it must conform to this agreed definition.
Estimated work remaining	The number of hours that a Team member estimates remain to be worked on any task. This estimate is updated at the end of every day the Sprint Backlog task is worked on. The estimate is the total estimated hours remaining, regardless of the number of people that perform the work.
Increment	Product functionality that is developed by the Team during each Sprint.
Increment of potentially shippable product functionality	functionality A completely developed increment that contains all of the parts of a completed product, except for the Product Backlog items that the Team selected for this Sprint.
Iteration	One cycle within a project. In Scrum, this cycle is 30 sequential calendar days, or a Sprint.
Pig	Someone occupying one of the three Scrum roles (Team, Product Owner, ScrumMaster) who has made a commitment and has the authority to fulfill it.
Product Backlog	A prioritized list of project requirements with estimated times to turn them into completed product functionality. Estimates are in days and are more precise the higher an item is in the Product Backlog priority. The list evolves, changing as business conditions or technology changes.
Product Backlog items	Functional requirements, nonfunctional requirements, and issues, which are prioritized in order of importance to the business and dependencies and then estimated. The precision of the estimate depends on the priority and granularity of the Product Backlog item, with the highest priority items that can be selected in the next Sprint being very granular and precise.
Product Owner	The person responsible for managing the Product Backlog so as to

	maximize the value of the project. The Product Owner represents all stakeholders in the project.
Scrum	Not an acronym, but mechanisms in the game of rugby for getting an out-of-play ball back into play.
ScrumMaster	The person responsible for the Scrum process, its correct implementation, and the maximization of its benefits.
Sprint	A time-box of 30 sequential calendar days during which a Team works to turn the Product Backlog it has selected into an increment of potentially shippable product functionality.
Sprint Backlog	A list of tasks that defines a Team's work for a Sprint. The list emerges during the Sprint. Each task identifies those responsible for doing the work and the estimated amount of work remaining on the task on any given day during the Sprint.
Sprint Backlog task	One of the tasks that the Team or a Team member defines as required to turn committed Product Backlog items into system functionality.
Sprint planning meeting	A one-day meeting time-boxed to 8 hours that initiates every Sprint. The meeting is divided into two 4-hour segments, each also time-boxed. During the first segment, the Product Owner presents the highest priority Product Backlog to the Team. The Team and the Product Owner collaborate to help the Team determine how much Product Backlog it can turn into functionality during the upcoming Sprint. The Team commits to this Product Backlog at the end of the first segment. During the second segment of the meeting, the Team plans how it will meet this commitment by detailing its work as a plan in the Sprint Backlog.
Sprint retrospective meeting	A meeting time-boxed to 3 hours and facilitated by the ScrumMaster at which the Team discusses the just-concluded Sprint and determines what could be changed that might make the next Sprint more enjoyable or productive.
Sprint review meeting	A meeting time-boxed to 4 hours at the end of every Sprint at which the Team demonstrates to the Product Owner and any other interested parties what it was able to accomplish during the Sprint. Only completed product functionality can be demonstrated.
Stakeholder	Someone with an interest in the outcome of a project, either because he or she has funded it, will use it, or will be affected by it.
Team	A cross-functional group of people that is responsible for managing itself to develop software every Sprint.
Time-box	A period of time that cannot be exceeded and within which an event or meeting occurs. For example, a Daily Scrum meeting is time-boxed to 15 minutes and terminates at the end of those 15 minutes, regardless.

(Schwaber 2004, 129-130)

Appendix B - Scrum Cheat Sheet

<h3>Roles</h3> <h4>Scrum Team</h4> <ul style="list-style-type: none"> Team is cross-functional and consists of 5-9 people There are no set project roles within the team Team defines tasks and assignments Team is self-organizing and self-managing Maintains the Sprint Backlog Conducts the Sprint Review <h4>Product Owner (PO)</h4> <ul style="list-style-type: none"> Accountable for product success Defines all product features Responsible for prioritizing product features Maintains the Product Backlog Insures team working on highest valued features <h4>Scrum Master (SM)</h4> <ul style="list-style-type: none"> Holds daily 15 minute team meeting (Daily Scrum) Removes obstacles Shields the team from external interference Maintains the Sprint Burndown Chart Conducts Sprint Retrospective at the end of a Sprint Is a facilitator not a manager 	<h3>Artifacts</h3> <h4>Product Backlog - (PB)</h4> <ul style="list-style-type: none"> List of all desired product features List can contain bugs, and non-functional items Product Owner responsible for prioritizing Items can be added by anyone at anytime Each item should have a business value assigned Maintained by the Product Owner <h4>Sprint Backlog - (SB)</h4> <ul style="list-style-type: none"> To-do list (also known as Backlog item) for the Sprint Created by the Scrum Team Product Owner has defined as highest priority <h4>Burndown Chart - (BC)</h4> <ul style="list-style-type: none"> Chart showing how much work remaining in a Sprint Calculated in hours remaining Maintained by the Scrum Master daily <h4>Release Backlog - (RB)</h4> <ul style="list-style-type: none"> Same as the Product Backlog. May involve one or more sprints dependent on determined Release date <p>"DONE" = Potentially Shippable!</p>	<h3>Meetings</h3> <h4>Sprint Planning - Day 1 / First Half</h4> <ul style="list-style-type: none"> Product Backlog prepared prior to meeting First half - Team selects items committing to complete Additional discussion of PB occurs during actual Sprint <h4>Sprint Planning - Day 1 / Second Half</h4> <ul style="list-style-type: none"> Occurs after first half done - PO available for questions Team solely responsible for deciding how to build Tasks created / assigned - Sprint Backlog produced <h4>Daily Scrum</h4> <ul style="list-style-type: none"> Held every day during a Sprint Lasts 15 minutes Team members report to each other not Scrum Master Asks 3 questions during meeting <ul style="list-style-type: none"> "What have you done since last daily scrum?" "What will you do before the next daily scrum?" "What obstacles are impeding your work?" Opportunity for team members to synchronize their work <h4>Sprint Review</h4> <ul style="list-style-type: none"> Team presents "done" code to PO and stakeholders Functionality not "done" is not shown Feedback generated - PB maybe reprioritized Scrum Master sets next Sprint Review <h4>Sprint Retrospective</h4> <ul style="list-style-type: none"> Attendees - SM and Team. PO is optional Questions - What went well and what can be improved? SM helps team in discovery - not provide answers 	<h2>SCRUM CHEAT SHEET</h2> <h3>Estimating</h3> <h4>User Stories</h4> <ul style="list-style-type: none"> A very high level definition of what the customer wants the system to do. Each story is captured as a separate item on the Product Backlog User stories are NOT dependent on other stories Story Template: <ul style="list-style-type: none"> "As a <User> I want <function>. So that <desired result>" Story Example: <ul style="list-style-type: none"> As a user, I want to print a recipe so that I can cook it. <h4>Story Points</h4> <ul style="list-style-type: none"> A simple way to initially estimate level of effort expected to develop Story points are a relative measure of feature difficulty Usually scored on a scale of 1-10. 1=very easy through 10=very difficult Example: <ul style="list-style-type: none"> "Send to a Friend" Story Points = 2 "Shopping Cart" Story Points = 9 <h4>Business Value</h4> <ul style="list-style-type: none"> Each User Story in the Product Backlog should have a corresponding business value assigned. Typically assign (L,M,H) Low, Medium, High PO prioritizes Backlog items by highest value <h4>Estimate Team Capacity</h4> <ul style="list-style-type: none"> Capacity = # Teammates (Productive hrs x Sprint Days) Example - Team size is 4, Productive hrs are 5, Sprint length is 30 days. Capacity = 4 (5 x30) = 600 hours NOTE: Account for vacation time during the Sprint! <h4>Velocity</h4> <ul style="list-style-type: none"> The rate at which team converts items to "DONE" in a single Sprint - Usually calculated in Story Points.
<h3>Process</h3>  <p>The diagram illustrates the Scrum process flow: Product Backlog (a stack of items) feeds into a Sprint (a cycle of Daily Scrum, Scrum Meeting, and Sprint Review). The Sprint produces a Shippable Product, which is then added to the Release Backlog (a stack of items).</p>	<h3>FAQ</h3> <ul style="list-style-type: none"> Who decides when a Release happens? At the end of any given Sprint the PO can initiate a Release Who is responsible for managing the teams? The teams are responsible for managing themselves. What is the length of a sprint? Tasks should take no longer than 16 hours. If longer then the task should be broken down further. Who manages obstacles? Primary responsibility is on the Scrum Master. However, teams must learn to resolve their own issues. If not able then escalated to SM. What are two of the biggest challenges in Scrum? Teams not self-managing, Scrum Master managing not leading. 	<h3>Glossary of Terms</h3> <ul style="list-style-type: none"> Time Box - A period of time to finish a task. The end date is set and can not be changed Chickens - People that are not committed to the project and are not accountable for deliverables Pigs - People who are accountable for the project's success Single Wringable Neck - This is the Product Owner! <p>Visibility + Flexibility = Scrum</p>	<h3>Tools</h3> <h4>Task Board</h4> <ul style="list-style-type: none"> While Board containing teams Sprint goals, backlog items, tasks, tasks in progress, "DONE" items and the daily Sprint Burndown chart. Scrum meeting best held around task board Visible to everyone

(ScrumLogic)

APPENDIX C - List of Case study questions

1. How is Scrum introduced to the IT Quality team?
2. How is the organization's culture in terms of openness and awareness to Agile/Scrum principles and practices?
3. How well are the team members coached about Scrum?
4. How are the roles played in the team?
5. How does the Scrum flow run in the IT Quality team? (Scrum planning, daily meeting, review and retrospective meetings, Scrum artifacts and such, how are they dealt with?)
6. What was wrong or missing in this flow, if any?
7. How is the Scrum framework received by the team members? Do they support or reject it?
8. Does Scrum improve or diminish any aspect of the project?
9. How is Scrum in comparison to Waterfall? What are the pros and cons?
10. What are the success factors of Scrum?
11. What are the actions can be taken to obtain these factors?
12. What types of project and processes that lend themselves to Scrum?
13. What types of project and processes that do not lend themselves to Scrum?

APPENDIX D - Interview questions

Group 1

1. What do you like about the Scrum approach?-

2. What do you not like about the Scrum approach?

3. What do you think is better in Scrum than in Waterfall and vice versa?

4. Do you think the productivity and/or other aspects of the project was improved or diminished by using Scrum? Please specify which aspect(s)

5. What do you think was wrong, or can be improved, in the application of Scrum in the IT Quality team?

6. In your opinion, what are the success factors for the application of Scrum in a project?

7. Which processes in the project do you think Scrum is not suitable for?
Please explain shortly why

8. Which project types (in general) do you think best suited for the Scrum approach, which are not? Please explain shortly why

Group 2

1. What do you think are the success factors of the Scrum approach in a project? Please explain shortly why.

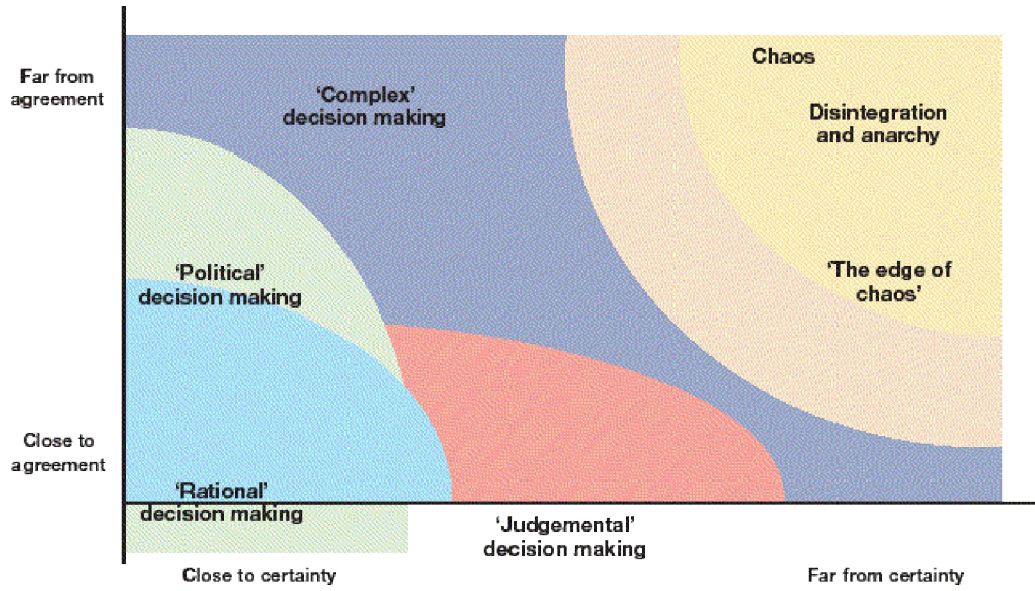
2. In your opinion, what types of process in a project Scrum is most suitable for? What not? (Or what is missing in Scrum practices?) Please explain shortly why.

3. In your opinion, what types of projects Scrum is best suited for?

4. Regarding the processes/projects mentioned above, if Scrum is not the best approach, is Waterfall a better option?

5. Do you have other comments regarding the application of Scrum in managing projects?

APPENDIX E – The Stacey matrix (Stacey 2002)



APPENDIX F - A Brief History of Project Management (Miles 2012)

A Brief History OF PROJECT MANAGEMENT

If project management is defined as the act of assembling people to systematically achieve a shared goal, then it has existed since ancient history. Think of the Egyptians erecting the Great Pyramid of Giza or the first Emperor of China commanding the manual labor of millions to build the Great Wall of China. Despite new method developments and technology changes over time, project management has kept its core goal intact: to deliver successful projects in a clear and effective way.

CREATIVE ARCHITECTS AND ENGINEERS AS PROJECT MANAGERS

The Great Pyramid of Giza 2550 BC

The man-made pyramid requires 20 years of labor and architects dedicated to each of the structure's four sides. The pyramid would become the first and only one of the Seven Wonders of the Ancient World to remain mostly intact.

206 BC The Great Wall of China

Emperor Qin Shi Huang orders the project's labor force to be organized into three groups: soldiers, civilians, and criminals.

Industrial Revolution 1750-1850

U.S. business leaders balance the management of both manual labor and mechanized manufacturing.

1896 The Transcontinental Railroad

One of America's greatest technological feats, the railroad that crosses California to the western U.S. becomes one of the first government projects that requires large-scale project management by railroad pioneers.

THE INTRODUCTION OF MODERN PROJECT MANAGEMENT

1896

Polish economist Karol Adamiecki creates the "harmonogram," a floating bar chart used to show tasks or resources laterality over time. It is limited to Polish speakers.

1912 Gantt Chart

Engineer Henry Gantt develops the "Gantt Chart," a diagram that breaks down tasks into a clearly designed schedule structure. It was first used in World War I to help build ships; it was used again in 1941 for the Hoover Dam; it is still being applied in organizations today.

1954

U.S. Air Force General Bernard Schriever coins the term "project management."

1956

The American Association of Cost Engineers (CACE International) is formed.

1957 Critical Path Method (CPM)

DuPont and Remington Rand design this method, which calculates the activities required to complete a project and the duration of each phase, to manage chemical plant maintenance. The project saves the company \$1 million in its first year of implementation.

1958 Program Evaluation Review Technique (PERT)

Developed as part of the U.S. Navy's Polaris missile submarine program, this approach analyzes the task type and the minimum amount of time needed for each task.

1962 Work Breakdown Structure (WBS)

The U.S. Department of Defense creates the method to break down projects into smaller components and organize them into hierarchical tree structures. The model focuses on what needs to be done—not the timeline for it.

1965

The International Management Systems Association (known today as the International Project Management Association) is founded in Europe.

1969

The Project Management Institute (PMI) is founded to promote project management as its own discipline and profession. Professionals can receive specific certifications through its program, including Certified Associate in Project Management and Project Management Professional.

1970 Waterfall Method

Winston Boyce creates a model whereby each phase must be completed and perfected before moving onto the next.

THE GROWTH OF SOFTWARE & APPS

Project Management Software Emerges

Several project management software companies emerge, including Artemis (1977), Oracle (1977), and Sceptor Corporation (1979).

1977

Several project management software companies emerge, including Artemis (1977), Oracle (1977), and Sceptor Corporation (1979).

1988

The "Earned Value Management" approach, combining measurements of scope, schedule, and cost—is integrated into project management.

1986 Capability Maturity Software

The Software Engineering Institute at Carnegie Mellon University starts developing "Capability Maturity Software," a five-level evolutionary path of increasingly mature processes.

2001 Agile Project Management

Created to manage complex software projects, this approach stresses adaptive planning and flexible response to change.

1996 PRINCE2

The seven-process PM system is widely used in the UK.

2008 Growth of SaaS-Based Project Management

A new class of SaaS-based tools emerges, providing unprecedented flexibility for project managers and their teams.

2006 Total Cost Management

Although the concept of Total Cost Management has been around for a long time, the AACE develops a framework to control, identify, and reduce costs. It's used in all parts of the organization, including project management.

2009

U.S. News names "project management" one of the top five in-demand skills that can lead to better job security.

2012 Mobile Applications

Mobile apps advance remote project management even further, allowing people to manage their projects any time and anywhere.

2010 Cloud Helps Spur Growth of a Virtualized Workforce

As cloud solutions allow access, virtual project teams will increase. CNNMoney reports that 30% of the U.S. job market is composed of independent virtualized workers. By 2019, it will grow to 40%.

2008 Project Management as an In-Demand Skill

U.S. News names "project management" one of the top five in-demand skills that can lead to better job security.

SOURCES: SPMU.EDU, SOURCEWAVE.COM, LEAGONS-PROJECTS.ORG, PROJECTSMART.CO.UK, PPTIPS.NET, IPMA.CH, USNEWSUNIVERSITYDIRECTORY.COM

mavenlink