An-Najah National University
Faculty of Graduate Studies

Introducing Agile Software Development Methodology (Scrum) into a Software Development Project at a Local Firm

By
Adham "Mohammad Wasfe" Hannoun

Supervisor
Dr. Baker Abdalhaq

This Thesis is Submitted in Partial Fulfillment of the Requirements for the Degree of Master of Engineering Management, Faculty of Graduate Studies, An-Najah National University, Nablus, Palestine
2013

Introducing Agile Software Development Methodology (Scrum) into a Software Development Project at a Local Firm

By
Adham "Mohammad Wasfe" Hannoun

This thesis was defend successfully on 04/11/2013 and approved by:

Defense Committee Members	**Signature**
Dr. Baker Abdelhaq (Supervisor).	.....
Dr. Muath Sabha (External Examiner)	.....
Dr. Muath Asmar (Internal Examiner)	.....

Acknowledgement

The author wishes to thank several people. Foremost, I would like to thank my parents, Father and Mother, for their love, kindness and support they have shown during the past two years; it has taken me to finalize this thesis. Furthermore I would also like to thank my soul mate, Thawq, for her endless love and support, during crises after crises, and yet her support stayed until the last moment. I also would like to thank my brother, sisters, my mother-in-law and father-in-law for their understanding, for their prayers and for the continuous support they showed me a lot.

I would like to express the deepest appreciation to my advisor Dr. Baker Abdalhaq, who has the attitude and the substance of a great mentor: he continually and convincingly conveyed a spirit of patience, motivation, enthusiasm, and immense knowledge. His guidance helped me in all the time of research and writing of this thesis. I could not have imagined having a better advisor and mentor for my master thesis.

Last but not least, I would also like to acknowledge with much appreciation the crucial role of the staff and company of Hulul business solutions, who gave the permission to use all required equipments and the necessary materials to complete the paper and provide me with any available support. Whoever helped me by gathering my data, making my steps or conquering obstacles, please honor me by receiving my forever gratitude.

الإقرار

أنا الموقع أدناه مقدم الرسالة التي تحمل العنوان:

Introducing Agile Software Development Methodology (Scrum) into a Software Development Project at a Local Firm

إدخال طريقة تطوير البرمجيات المسماة (أجايل سكرم) إلى مشروع تطوير برمجيات في شركة محلية

اقر بأن ما اشتملت عليه هذه الرسالة إنما هي نتاج جهدي الخاص، باستثناء مـــا تمـــت الإشارة إليه حيثما ورد، وأن هذه الرسالة ككل، أو أي جزء منها لم يقدم من قبل لنيل أية درجة علمية أو بحث علمي أو بحثي لدى أية مؤسسة تعليمية أو بحثية أخرى.

Declaration

The work provided in this thesis, unless otherwise referenced, is the researcher's own work, and has not been submitted elsewhere for any other degree or qualification.

Student's Name:	.....	اسم الطالب:
Signature:	.....	التوقيع:
Date:	.....	التاريخ:

Table of Contents

List of Tables

List of Figures

Abbreviations

ASD	Agile software development
CRM	Customer Relationship System
DS	Daily Scrum
DSDM	Dynamic Systems Development Method
HR	Human Resources
IID	Incremental and Iterative Development
ISP	Internet Service Provider
IT	Information Technology
MRT	Media Richness Theory
OT	Over Time
PB	Product Backlog
PE	Poker Estimation
PM	Project Manager
PO	Product Owner
PP	Pair Programming
QA	Quality Assurance
RB	Release Backlog
SB	Sprint Backlog
SM	Scrum Master
SP	Sprint Planning
SR	Sprint Review
ST	Scrum Team
SW	Software
US	User Stories
WB	West Bank
XP	Extreme Programming

Introducing Agile Software Development Methodology (Scrum) into a Software evelopment Project At a Local Firm

By
Adham "Mohammad Wasfe" Hannoun
Supervisor
Dr. Baker Abdalhaq

Abstract

The researcher tried to trade-off two different methodologies in a practical way through a case study done on a local firm, and supported it with a lot of interviews to analyze more results utilizing in both cases the qualitative approach. This research aims to investigate whether it is possible to achieve a notable progress in the process of the developing a SW project in many perspectives. By interviewing project managers, developers and SW engineers, and observing how Scrum was adopted in a local firm, the use of Scrum will be inspected. At the same time, the measures of progress within the project must be examined through literature studies. The adoption of Scrum will be compared to another project that has the same characteristics but was previously developed using the old traditional methodology.

The research had many results and findings, like there are some minor differences between Scrum in general and Scrum applied in this research, and there are differences after applying Scrum in many firms in Palestine in comparison to the traditional way of development methodologies used for years also in Palestine. There was also a difference

between how Scrum is explained in the literature and how the researcher has implemented it in the local firm after taking the feedback from other local firms for both traditional and agile methodologies.

Some changes and additions need to be made to the usage of the Scrum as it is now, and a quality policy must be in place, used and continually improved based on feedback. It is also found that the customer has a big role if he is to commit to what his role dictates him to do.

The researcher thinks that this research has a good contribution for the local market and for the local firm who adopted the research. This research is a trial to ask questions of why problems in this local market happen and how can we solve these problems. The researcher thinks that this research offers something new and hadn't been searched a lot on the local market, and if really a formal result can be conducted from this research, then at least part of the problems should be solved, and the suffering of the IT customers and firms developing the system can be got rid of, which opens the door for more creativity in the field. After the case study finished, the researcher explored the views of the Scrum members involved in the project; they reported that a lot of the problems found in previous projects were solved and felt much better about their work and progress besides the new things that they have learned through this new way of work which they were aware of. This research also gave a great contribution to the local firm that adopted the research; it gave a clear assessment for the current way of current working procedures, a clear

understanding of the concepts behind upgrading customer communications, testing acceptance and adapting changes within the work.

Based on the research findings, local firms should perform studies and researches to determine the best methodologies of SW development that fits their business environment and fit the Palestinian market. Furthermore, firms should focus more on the customers. Moreover, there should be studies and researches to differentiate between huge, middle and small projects, to make sure that the new way is right and fit the firm business value. Eventually, a new better way of doing things can be arranged till this market –or at least part of it- reaches the state of art that is supposed to be.

Chapter One
INTRODUCTION

Chapter One
INTRODUCTION

1 Introduction:

1.1 Overview:

SW development is a complicated process; such a complication needs some ways to manage them, and this is the point where the SW development methodologies emerged in the trial to take control over the SW development process and set it on the shore of safety with the least bearable costs. The SW development witnessed two main methodology approaches in their march of progress in the field of SW development; the traditional methodologies [i.e. the so called Waterfall] and the modern technologies [i.e. Agile].

1.2 The Case:

So far, a lot of studies and researches had been conducted on the failure and success of the IT projects, yet no study resulted that the success rates are high enough to convince the IT firms to adopt and follow, which lead a lot of companies to search for success factors in their development, and started to differentiate between SW methodologies and categorizing them into traditional and modern SW development methodologies. Palestine is no different from any other market in the IT industry regarding

the development and progress in this field; its local market deserves to have better SW project development and better customer satisfaction.

One local SW firm who was from the start of its foundation working on the traditional methodologies represented by “Waterfall” -where the steps of the project are usually sequential and can’t have intended progress unless previous steps are to be done- agreed to make a transfer in its methodology into newer methodology [i.e. Agile Scrum] after seeking for long time for factors to upgrade its way of work and in the search for new paths to success. The idea of the research was suggested on the firm, and the firm found it as an opportunity to start with a change and showed a large interest in the transition to agile development, like Scrum, while its other projects are still being developed under the formal old way of development. The firm agreed to implement agile Scrum on one of its projects that was still about to begin, in order to check the progress comparing to other projects that have the same characteristics but had been developed on the old traditional ways.

1.3 Aim:

The aim of this thesis is to investigate if it is possible to achieve a notable progress in the process of the developing a SW project in many perspectives in the local market when using the agile development framework Scrum. By interviewing project managers, developers and SW engineers are observing how Scrum was adopted in this local firm, the use

of Scrum will be inspected. At the same time, the measures of progress within the project must be examined through literature studies. The adoption of Scrum will be compared to another project that has the same characteristics but was previously developed using the old traditional methodology.

1.4 Problem Definition:

Software projects have a huge and frightening percentage of failure relative to other projects in the other aspects; it is a fact that a high percentage of software projects are doomed to failure, and our locale is no exception for this statistic and many local projects were failed due to many reasons. What this paper will try to do is to address some of the related reasons to these failures, and to adopt the necessary steps that weigh the balance toward the software projects success over their failures through following the agility rules. It is important to be critical on why agile-based processes should be recommended to organizations. Changes are not always acceptable or adaptable, and customer requirements are not always fully understood and it might take some time for the team to understand them, therefore, it is important to be able to change a project plan and incorporate this change in a project. This research is about illustrating the use of an agile-based software development process and if agile-based process can be recommended to organizations for software development process.

This master thesis aims to answer the following questions:

- How scrum is implemented within local software project to provide agility?
- What are the changes caused by the implementation of scrum inside some SW company?
- Is it possible to make additions to Scrum so that it more fits the local environment?
- How can the success or failure of the implementation be measured?

1.5 Research Objectives:

This research aims to achieve the following objectives:

1. To introduce the agile SW development into local SW projects and to adopt them as a formal procedure inside local institutions of SW development.
2. The simplest objective is to find a way to mitigate the percentage of failures among the SW projects.
3. To pinpoint the importance of agile software development in the IT projects for a local IT company.
4. To introduce a faster and more easy-going procedures for testing and getting results of any process created or developed throughout the project.

5. To establish a means to accelerate the time it takes to deliver business value.
6. To reach a way that resembles simplicity and low overhead to manage and track the SW development.
7. To create an environment for development skills that are associated with the successful construction of a SW solution with the commercial client.
8. To seek equilibrium point among the business elevated requirements throughout the project versus the trilogy of cost, time and effort.
9. To facilitate the work of the team working on the project items using the concepts of agility gathered in the “agile manifesto”.
10. To clarify as possible the roles of each party involved in the SW development project.
11. To conclude if scrum is most compatible with the local market of SW industry and to offer more improvements and refinements that can be introduced into agile within the local market.

1.6 Limitation of Scope:

The first limitation was the secrecy of the materials provided by the local firm since a master thesis is published to the public, classified information must not be used carelessly. The other obstacle was the management commitment toward the new project methodology since the new methodology crosses the borders of the organizational structures and deals with all members of the team as

peers, which was sometimes difficult at the start to the management.

Structure of the Thesis:

Chapter 1: INTRODUCTION

In this chapter, an overview about SW development methodologies and their needs were mentioned at the first. Then the case was identified along with the aim of this thesis. A brief problem definition was also given in this part along with questions that this research will try to answer them. Introducing agile SW development methodologies into local environment and mitigating the percentage of failures among the SW projects are some of the objectives that were also mentioned in this section along with many other objectives. In this section, the limitations of the research and its structure were also explained.

Chapter 2: LITERATURE REVIEW

In this section, the research talks about a historical background about SW development and its methodologies**,** and about the related studies that had been conducted on the same subject previously and their results. It also talks about agile SW development and its relation with the local SW development projects in the light of the agile manifesto; it starts this by exploring the local IT SW development projects, then mentioning the need for agile project management and the benefits of introducing agile software development into local software development and implementation projects.

Then an overview about SCRUM was briefed along with its practices, behaviors, artifacts and roles. Then a brief description for SCRUM usage in general was concluded mentioning advantages and disadvantages.

Chapter 3: RESEACRH METHODOLOGY

In this section, the research methodology is explained starting by the introduction about used methodologies in this research, mentioning the research population and sample type. The research also talks about the used triangulation and the doability of it under the current circumstances. Then explaining the differences between the different applied strategies was also explained over the case study used and its validation methods. The last thing this section explained is the data gathering techniques used in this research and mentioning their advantages and disadvantages.

Chapter 4: ANALYSIS AND DISCUSION

An introduction about project description and purpose was started in this section, then appended by the explaining of the interviews and their followed procedures. Then the practical behavior and procedures followed in applying the case study were being explained in details.

Chapter 5: FINDINGS AND RECOMMENDATIONS

In this section, a discussion was arranged for both the interviews and case study, and the differences between the different frameworks applied in Palestine. Then the findings from this research were listed along with its

contribution to the community, mentioning the conclusions from this research and the future studies expected to be built on this research.

Chapter Two

LITERATURE REVIEW

Chapter Two

LITERATURE REVIEW

2.1 Introduction:

Software plays an important factor in the march of individuals and in the competition between companies; software can be used as an application in personal computers or as a part of an industry, or it can be used for enabling services and automating processes. Since the 1950's, software has been developed, and different methods, paradigms and process models have been invented to handle the complex efforts of development. During the time, some of the development methods have become heavily documentation oriented or expect the developers to rigorously follow certain processes. Those can be called heavy or traditional methods, for example structured analysis and design (Boehm, 2003). In the turn of the millennium, new development method ideas were presented in the form of Agile Manifesto as a counteraction to rigorous, plan-driven software development (Boehm, 2003). Agile Manifesto gives an ideological background for agile software development more than procedural advices.

So far, a lot of studies and researches had been conducted on the failure and success of the IT projects, yet no study resulted that the success rates are better than 50% either by time or cost, which lead a lot of companies to search for success factors in their development, and started to differentiate between SW methodologies and categorizing them into

traditional and modern SW development methodologies. It is claimed that the most significant differences between agile and traditional, heavy methods can be found in the emphasis on methods, agile methods are less document oriented than traditional methods; instead, agile methods are more code-oriented and they emphasize working code over documentation (Fowler, 2001). He also claims that traditional methods would resist change but agile methods would be more open to meet changes. Besides, agile methods are more people-oriented that process-oriented (Fowler, 2001).

It is claimed that the most significant differences between agile and traditional, heavy methods can be found in the emphasis on methods. Agile methods are less document-oriented than traditional methods. Instead, agile methods are more code-oriented and they emphasis working code over documentation. Those traditional methods would resist change but agile methods would be more open to meet changes. Besides, agile methods are more people-oriented that process-oriented (Fowler, 2005).

2.2 Agile software development and local software development projects:

Agile development can be described as a flexible way of working (Shore & Shane, 2008). Agile development is not a process; it is a philosophy, a way of thinking about software development (Cockburn, 2007). This refers to the Agile Manifesto, written in 2001 by the Agile Alliance consisting of 17 people with large experience of agile

development. After discussing common features among agile development, the participants in the meeting signed a Manifesto of values. This is known as the Manifesto for Agile Software Development, but is often referred to as the Agile Manifesto (Softhouse, 2006).

The manifesto for Agile Software Development consists of four addresses that all other issues pour into them; they are values being viewed as better ways of developing software by doing it and helping others do it. These values are:

- **Individuals and interactions** over processes and tools
- **Working software** over comprehensive documentation
- **Customer collaboration** over contract negotiation
- **Responding to change** over following a plan

So while there is value in the items on the right, the manifesto values the items on the left more, and all these values are written by practitionars who use their experiences on the fields (Softhouse, 2006).

There are several kinds of agile development, the most well known are XP (Extreme Programming) and Scrum, but other types, such as Lean Development, are used as well (Szalvay, 2004). In general, traditional software development methods are not always feasible in the rapidly changing business environment. However, the agile development approach has been claimed to be such.

2.2.1 Agile methodologies effectiveness:

Chow and Cao (2008) suggested a model to determine the success factors of projects that use agile methodologies, where the following factors were proposed to affect the success or the failure of an agile project:

- **Organizational factors**, such as management commitment.
- **People factors**, such as team capability and customer involvement.
- **Process factors**, such as the project management.
- **Technical factors**, such as agile software development techniques and procedures.
- **Project factors**, such as project environment.

2.2.2 The need for agile project management:

Dealing with the increase volatile organizational environment is an enormous challenge for any manager of any software development project (Coldewey, Eckstein, McBreen & Schwanninger, 2000). The traditional methodologies can be characterized as linear, sequential processes, and the related management approaches can be effective in developing software with stable, known, consistent requirements, yet most real-world development efforts are much more likely to be conducted in more unstable and volatile environments, as organizations adapt to changing technology, markets, and social conditions. Requirements for systems must be able to change with those organization and environments with a high speed (Truex, Baskerville & Klein, 1999).

Besides, we live in an uncertain world where it is difficult to predict what will happen in the future. Not only the governments and organizations face unprecedented budget cuts, but also are still under pressure to perform and deliver, mean while, the project managers face tough challenges like unforeseen situations, budget cuts and organizational changes which can have a dramatic impact on project outcomes. The key to successfully managing projects is to break them into stages, plan the current stage in detail and be flexible regarding the subsequent stages. Providing a flexible yet controlled process that can be used to deliver solutions; the agile project management combines effective use of people's knowledge together with techniques such as iterative development to achieve tight project delivery timescales. It offers flexibility while still recognizing the processes that give project managers the confidence to run their projects effectively.

2.2.3 The benefits of introducing Agile Software Development into local software development and implementation projects:

After interviewing many firms and forming the literature review, the researcher found that many local organizations seek to adopt an agile approach, and they are sometimes concerned about some of the messages and myths of agile, and it can be very difficult to separate rumors from facts. For example, some agile approaches suggest project management is not very crucial, which may lead some to the thought that agile can only be applied for simple pieces of work and that would organizations need either to follow agile only or to follow the formal project management. One

advantage that might be an indication in favor of agile project management is that organizations can adopt an agile approach that has a track record of successful management and delivery in the corporate environment, and an approach that complements and works with existing corporate processes. This doesn't mean that an organization needs to specify agile management processes for itself; rather, the organization can simply adopt a tried and tested approach. Besides, the organizations can benefit the fact that that agile has a formal recognized certification process for individuals, which can be used to develop professionalism in employees, and as part of staff professional development. The certification can also be used as part of the recruitment process, to identify individuals who have already been independently assessed and certified as possessing accurate knowledge on agile and having proven track records on experiencing agile.

2.3 SCRUM:

In this research, Scrum was used in the case study performed at a local firm comparing it to the traditional way of working. In this part of the research, an overview of agile development is given and the agile development framework Scrum is presented in details.

2.3.1 Agile development:

Agile development can be described as a flexible way of working (Gustavsson, 2007). According to Shore and Warden (2008), agile development is not a process; it is a philosophy, a way of thinking about

software development. This refers to the Agile Manifesto, written in 2001 by the Agile Alliance consisting of 17 people with large experience of agile development. After discussing common features among agile development, the participants in the meeting signed a Manifesto of values. This is known as the Manifesto for Agile Software Development, but is often referred to as the Agile Manifesto (see below Figure 2.1) (Cockburn, 2000).

Manifesto for Agile Software Development

We are uncovering better ways of developing software by doing it and helping others do it. Through this work we have come to value:

· Individuals and interactions over processes and tools

· Working software over comprehensive documentation

· Customer collaboration over contract negotiation

· Responding to change over following a plan

That is, while there is value in the items on the right, we value the items on the left more.

Figure 2.1 Manifesto for Agile Software Development from Agile Alliance (Agile Alliance, 2001)

2.3.2 Introduction to Scrum:

There are several kinds of agile development, the most well known are XP (eXtreme Programming) and Scrum, but other types, such as Lean Development, are used as well (Softhouse, 2006).

The researcher used the (American Heritage, 2013) to define the Scrum from a language perspective, before he goes into the details of what

really meant by the term. The (American Heritage, 2013) defines the Scrum as a: A play in Rugby in which the two sets of forwards mass together around the ball and, with their heads down, struggle to gain possession of the ball. In the same time, it also defines other concepts:

- Method: A means or manner of procedure, especially a regular and

systematic way of accomplishing something (American Heritage, 2013).

- Methodology: A body of practices, procedures, and rules used by those who work in a discipline or engage in an inquiry; a set of working methods (American Heritage, 2013).

- Praxis: Practical application or exercise of a branch of learning (American Heritage, 2013).

- Framework: A structure for supporting or enclosing something else, especially a skeletal support used as the basis for something being constructed (American Heritage, 2013).

Schwaber (2004) mentions Scrum as a “management and control process” in his first book about agile development. More recently, he has explicitly defined Scrum, “Scrum is a tool, a framework” (Schwaber, 2008). The definition of a framework suits well together with what Scrum is. Scrum can provide support for software development but it will not define how you shall do it, just like a framework. Scrum is a skeleton that

needs to be built upon, or a framework that needs to be filled (Schwaber, 2008).

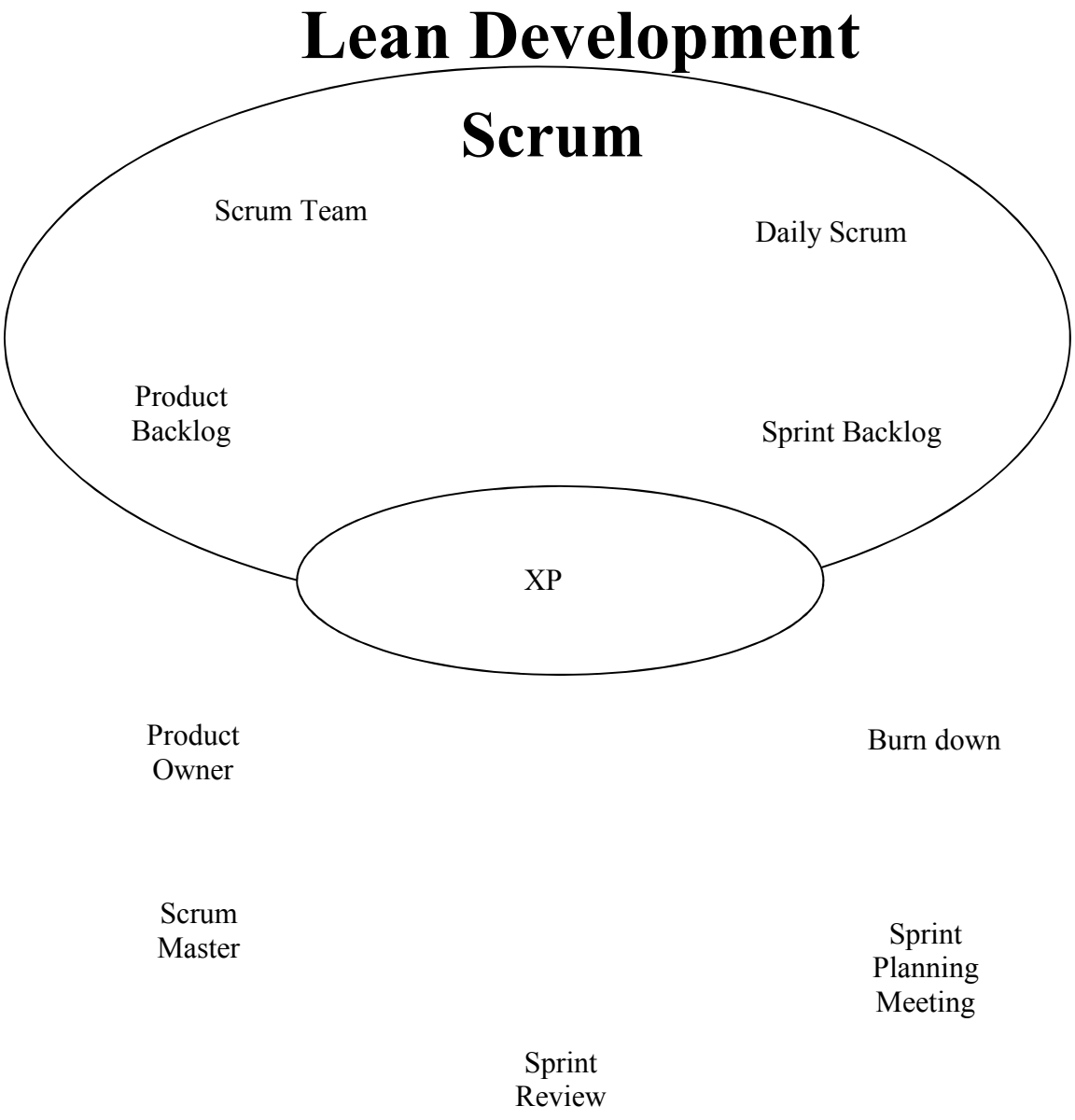


Figure 2.2 Scrum as a framework (Adopted from Kniberg (2007))

The researcher in this research chose to follow Schwaber's definition of Scrum as a framework, since this definition was used by the founder of Scrum. The definition of Scrum as a framework will be adopted in this Master thesis. This definition may not be fully correct in all situations, but it is believed to be the most appropriate definition in this context.

Scrum is a management and control process that cuts through complexity to focus on building software that meets business needs (Schwaber & Beedle, 2002). Scrum deals primarily at the level of the team, it enables people to work together effectively, and by doing so, it enables them to produce complex, sophisticated products (Schwaber & Beedle, 2002).

Scrum is a term that describes a product development process (Schwaber, 2004). Scrum was first described in 1986 in the article "The new product development game" in Harvard Business Review by Hirotaka Takeuchi and Ikujiro Nonaka (1986). The name Scrum was taken from a term in the sport Rugby (Schwaber, 2004).

Scrum is based on short iterations of about 2-4 weeks, small project teams, and daily status meetings. The idea is to have a working product at the end of every iteration, possibly with only limited functionality. A release is made after each Sprint. This could possibly be delivered as a final product. The project team adjusts the delivered functionality according to the complexity and time available, and uses feedback from the customer to

focus their work on valuable functionality. Not every detail and task needs to be known at the start of the project, instead Scrum helps adapting to constantly changing requirements (Schwaber, 2004).

2.3.3 Scrum roles:

According to Schwaber (2004), there are only three roles in Scrum: the Scrum Master, the Product Owner and the Scrum Team. Working together, the different roles complement and balance each other. The Scrum Master makes sure that the rules and values of Scrum is present within the organization, the Product Owner represents the customer and communicates the business goals, while the Scrum Team is a self-organizing unit that takes care of the actual work (Schwaber, 2004).

2.3.3.1 The Scrum Team:

The Scrum Team is a small, cross-functional team that performs all development. At the start of each Sprint, the team commits to a number of items from the Product Backlog that the team thinks they will be able to complete. The team then has the full authority to do whatever it finds necessary to fulfill the commitment within the Sprint (Schwaber & Beedle, 2002).

The fact that the Scrum Team is self-organizing is a key principle in Scrum. Even if it is tempting for a Scrum Master to resolve the team's internal problems, helping them figuring out how to do it themselves by

removing unnecessary responsibility and problems is usually the best practice (Schwaber & Beedle, 2002). The best size for a Scrum Team is seven people, according to Schwaber and Beedle (2002). Larger teams increases complexity, decreases productivity and make the Scrum practices hard to use. Smaller teams limit the amount of interaction and reduce the possible productivity gains (Schwaber & Beedle, 2002).

2.3.3.2 The Scrum Master:

Schwaber and Beedle (2002) describe the Scrum Master as a fundamental part of Scrum, a guardian of its rules, values, and practices against external forces. The Scrum Master should also be a driving force behind the Scrum implementation and teach relevant lessons to team members. He or she also works closely together with the customer and management to select a Product Owner, and with management to build the necessary Scrum Teams, since the right type of people need to have these roles. When this is done, the Scrum Master helps the Product Owner and the Scrum Teams to create a list of outstanding work, known as the Product Backlog (Schwaber& Beedle, 2002). The Scrum Master teaches the Product Owner how to use the Product Backlog and other parts of Scrum as effectively as possible. If someone does not fulfill his or her role, the Scrum Master is held accountable since he or she has not taught how to do the job well enough (Schwaber, 2007).

Unlike a traditional project manager, the Scrum Master does not assign tasks to different team members and have no authority over the development teams. Instead, he or she focuses on making sure that all the details of Scrum are fully functional within the team and organization. The success of the project is one of the responsibilities of the role. The probability of success increases by helping the team turns the Backlog into functionality, as well as helping the Product Owner create and maintain the Backlog (Schwaber, 2004).

During everyday work, the Scrum Master helps the Scrum Team by removing impediments that hinders their progress as well as making decisions, leading the Daily Scrum and improving productivity. Decision-making should be done instantly if necessary, without waiting for all information to be available. This helps the team focus on the task and know what to do at all times, even if more information becomes available at a later point and decisions may be changed (Schwaber & Beedle, 2002).

2.3.3.3 The Product Owner:

In a Scrum project, the customer is represented by the Product Owner. The main responsibility is to define what currently constitutes the highest priority business value and make sure the Scrum Team is working on that during every Sprint. This is done through adding and prioritizing different items of work to be done in the Product Backlog. After each Sprint, the Product Owner uses the result and feedback to asses if the

priorities were right. If circumstances or business opportunities have changed, the Product Backlog could be updated accordingly before the next Sprint, dropping, adding, or reprioritizing items (Schwaber, 2004).

It is important that only a single well-respected individual is Product Owner and responsible for updating the Product Backlog. Otherwise, Schwaber and Beedle argues, there is a risk that different versions of the list will appear depending on the different opinions on what is important, leading to confusion and frustration. The Product Owner also works together with the Scrum Team to estimate how long each item would take to complete (Schwaber & Beedle, 2002).

The Scrum Team is not managed by the Product Owner in a traditional sense. He or she suggests what needs to be done at the start and evaluates the progress at the end of each Sprint, focusing on the customer business values (Schwaber, 2004).

2.3.4 Practices:

Compared to traditional project management, Scrum introduces some new principles and practices that need further explanation.

2.3.4.1 Sprint:

In Scrum, all work is done in iterations known as Sprints. Schwaber suggest a Sprint to last for 30 calendar days, but it could possibly be shorter. Every Sprint begins with a Sprint Planning meeting. The Product

Owner meets with the team, describes the top priority items from the Product Backlog and gives them the opportunity to pick as many as they think they will be able to complete during the sprint (Schwaber, 2004). After a number of items have been chosen the team creates a Sprint Backlog, the plan during the Sprint, and set the Sprint Goal. Each day a Daily Scrum is held to communicate the progress and efforts made in the team. At the end of the sprint a Sprint Review session is held where the functionality from the completed items are demonstrated to everyone that is interested. The Sprint ends with a Sprint Retrospective where the performance and lessons learned from the recent Sprint is discussed (Schwaber, 2004).

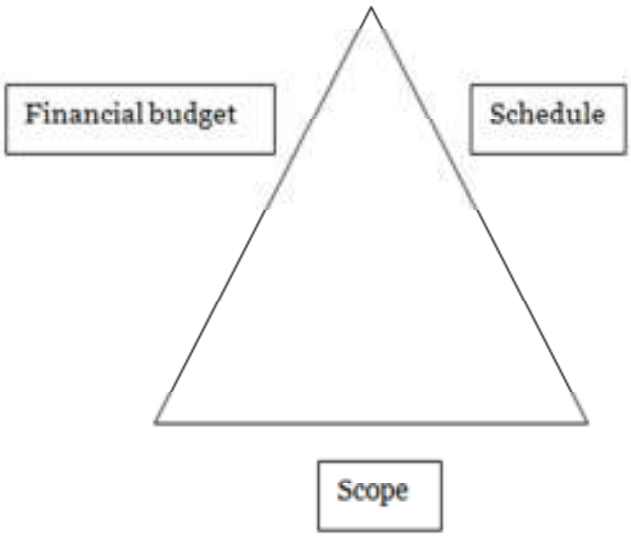


Figure 2.3 Project trade-off triangle (Dalcher & Brodie (2007), p. 37)

According to Dalcher and Brodie, a project is defined and limited by its financial budget, schedule, and the scope of the project. Figure 2.3

shows the trade-off triangle, which shows that a change to one side of the triangle has to result in a change of the other two correspondingly (Dalcher & Brodie, 2007).

Schwaber and Beedle further divide the scope into two variables. That is, according to them four variables constrain all development projects: available time, cost, delivered quality and delivered functionality. During a sprint, the first three are fixed, since its duration and hence the cost of salaries is known together with the organizational quality standard. The only thing that may be changed is the functionality, as long as it meets the sprint goal. By increasing or decreasing the scope of the functionality, the team is able to complete their assigned items. Any unfinished functionality that is deemed necessary is re-entered onto the Product Backlog at the end of the sprint (Schwaber & Beedle, 2002).

It is possible to cancel a sprint in some cases. The sprint goal may become obsolete, due to changes in the market, business decisions, or other reasons. The team may realize mid-way that it cannot achieve the sprint goal, or it feels it needs more direction from management before proceeding with the implementation of more functionality. If someone outside the team tries to change the goal or nature of the sprint, the team also has the power to cancel the Sprint, and call for a new SP meeting. At that point, the reason for the cancellation will be explained, making people careful not to make any such changes (Schwaber & Beedle, 2002).

2.3.4.2 Self-organization:

One of the core ideas in Scrum is that the team is self-organizing. No one outside of the team is allowed to tell the team what to do once it has committed to a number of items from the Production Backlog at the start of the Sprint. Any resources, such as external consultants or new equipment, that the team thinks would increase their productivity, should be available within budgetary constraints (Schwaber, 2004).

Self-organization is, according to Schwaber, something that is easy to learn on the intellectual level, but hard to use in practice. It takes some experience with Scrum before a team will understand how to take the responsibility and authority traditionally reserved for the management. The SM should help the team figure out how to organize itself, while at the same time avoiding the risk of taking their self-management away through explicit decisions (Schwaber, 2004).

2.3.4.3 Sprint Planning Meeting:

As mentioned earlier a sprint starts with a SP. It is time-boxed 5 to eight hours since the whole idea is to get work; not to think of working. The SP meeting consists of two parts. In the first part the PO and the ST gets together to decide what will be done for the coming Sprint. The PO presents the most prioritized items from the Product Backlog to the team and tells them what is most desired. The ST questions the PO about content, intentions, purpose, and meaning with the items. This is for the ST

to understand the range and functionality of the item. When the ST knows enough to estimate how much they can turn into functionality during the next sprint, they select that number of top priority items from the PB. The PO is then informed about their decision and that the team will do their best to complete the assigned items within the duration of the Sprint (Schwaber, 2004).

During the second part, the team plans the sprint and breaks down the items into tasks. When the second part has started, the sprint is started as well, and the clock is ticking towards the end of the time-boxed sprint. The team is now responsible to manage themselves through the sprint and they start by creating a Sprint Backlog, plan the sprint, and set the sprint goal. It is important for the team to have a sprint goal to aim at in order to complete the tasks (Schwaber, 2004).

2.3.4.3.1 Poker Estimation:

The estimation was done in the Scrum planning meeting using the Poker estimation process using the cards like Poker players. With this estimation process, each task must be weighed and estimated to distribute the load among all the members. K.Molokken-Ostvold and N.C.Haugen (2007) found that estimates obtained through the Planning poker process were less optimistic and more accurate than estimates obtained through mechanical combination of individual estimates for the same tasks.

Poker Estimation is a consensus-based technique for estimating, mostly used to estimate effort or relative size of user stories in software development and most commonly used in agile software development (Cohn, 2005). The reason to use poker estimation is to avoid the influence of the other participants. If a number is spoken, it can sound like a suggestion and influence the other participants' sizing (Cohn, 2005). Planning poker should force people to think independently and propose their numbers simultaneously. This is accomplished by requiring that all participants show their card at the same time (Cohn, 2005). It is based on a list of features or tasks to be delivered and several copies of a deck of numbered cards (Cohn, 2005). The feature list, often a list of user stories, describes some software that needs to be developed or worked on (Cohn, 2005).

The cards in the deck have numbers on them that differs based on the team wants to work on. A typical deck has cards showing the Fibonacci sequence including a zero: 0, 1, 2, 3, 5, 8, 13, 21, 34, 55, and 89. The below picture gives a glance about what is meant by the PE (Cohn, 2005).

Figure 2.4 Poker card deck (Wikipedia)

The above deck uses the sequence: 0, ½, 1, 2, 3, 5, 8, 13, 20, 40, 100, and optionally a ? (unsure) and a coffee cup (I need a break). Some organizations use standard playing cards of Ace, 2, 3, 5, 8 and King. Where King means: "this item is too complicated or too big to estimate." "Throwing a King" ends discussion of the item for the current sprint (Cohn, 2005). PE encourages the pessimistic conditions, which reflects the inherent uncertainty in estimating larger items. This feature is represented by the Fibonacci sequence usually; a three minutes time can be used to limit time spent in discussion of each item. At the estimation meeting, each estimator is given one deck of the cards. All decks have identical sets of cards in them (Cohn, 2005).

2.3.4.4 Daily Scrum:

One of the most fundamental parts of Scrum is the daily status meeting, known as Daily Scrum. During approximately 15 minutes every working day, the Scrum Team meets in a fixed location. The only thing that should be mentioned in the meeting is the team member's answer to three important questions: what have you been doing since the last Daily Scrum, what will you be doing until the next Daily Scrum, and is anything hindering you from doing your work? (Schwaber & Beedle, 2002) The location and time for the Daily Scrum should be fixed so everyone will know when and where to go every day. All members of the team are required to attend, but if it is not possible, alternatives include telephone attendance or representation by another team member. The meeting should

start on time and latecomers are usually required to, for example, donate a small amount to charity (Schwaber, 2004).

The Daily Scrum gives the whole group an update on what is going on in the project, and an opportunity to ask for help, input and feedback on the work. If any impediments are reported, it is the Scrum Master's most important responsibility to remove them as soon as possible. The Daily Scrum should not turn into a working session or a place for long discussions between a few people. If this happens, a separate meeting should be held right after the Daily Scrum to address the issue (Schwaber & Beedle, 2002).

2.3.4.5 Sprint Review:

When a Sprint is finished, a SR meeting is held. At the SR meeting, the Scrum team presents its completed items in a demonstration for the Product Owner and other Stakeholders that wish to attend. In order to present the item it must be defined as done. The Product Owner reviews if the items have the intended functionality. The result of the Sprint is inspected concerning the project goals and adaptations are made to maximize the chance of reaching the goals. This meeting is an informal meeting and it is time-boxed to four hours together with the Sprint Retrospective (Schwaber, 2007).

This way others can embrace these experiences. The Scrum Master is responsible for setting up and coordinating the meeting, deciding how and

by whom the result will be presented. The Scrum Master also notifies all attendants a week before the Sprint Review. No one should prepare extensively for the demonstration, and therefore are PowerPoint and similar presentations forbidden (Schwaber, 2004).

2.3.4.6 Sprint Retrospective:

After the Sprint Review, before a new sprint starts, a Sprint Retrospective meeting is held. This meeting is used to evaluate the finished sprint and is, similar to the Sprint Review, time-boxed to three hours. The Scrum Master here encourages the team to revise and improve the work process for future Sprints. This hopefully results in a more effective and enjoyable work-process for the next Sprint.

2.3.4.7 Working environment:

To make it easier for people to communicate and facilitate the self-organization, open working environments is highly recommended. Rolling desks and walls covered with whiteboards helps creativity and let the team rearrange the environment depending on whom they are working together with currently. In the same way, the team should be able to set its own working hours, as long as they are appropriate to the rest of the organization. Since the employees normally cost a lot more than their equipment, the best possible tools and resources should be provided if it increases their productivity (Schwaber &Beedle, 2002).

2.3.4.7.1 Pair Programming:

Many researchers like Denning (2004), Kautz and Nielsen (2004) and Slappendel (1996) indicated that communication is important for innovation. The Agile Alliance does not see comprehensive documentation as important as working software, however the dissemination of knowledge through documentation or a knowledge management system is also a requirement of innovation (Denning, 2004). Communication between everybody is vital for an organization to be innovative so that everybody understands what the other is doing (Allen and Henn, 2007).

An important feature of the work on Scrum for the team was the Pair Programming (PP) and Coaching. Pair Programming (PP) can be dealt with as an effective means for knowledge transfer when new members join a team (Benedicenti and Paranjape, 2001). According to Coman, Sillitti and Succi (2008), the occurrence of PP in a team can be explored as: novices are shown to do PP more during their first month in the team than veterans. In the second month, novices drastically reduce their time spent working in pairs, while experts do PP for about the same percentage of their total time.

One of main agile Scrum practices is the Pair Programming, and the research tried to instill this concept on the developers and keep it all the period of the development in their minds in the trial to commit to agile Scrum as possible.

2.3.4.7.2 Team Autonomy:

Team autonomy is defined by Hoegl and Parboteeah as the influence of management and other individuals (outside the team) on the team's activities. Such influence can be deliberate actions from management to limit autonomy, such as requiring the team to make certain decisions regarding work strategies or processes, project goals and resource allocation (Hoegl and Parboteeah, 2006). The autonomy of teams also can be described as the team's ability to regulate their boundary conditions (Emery and Thorsrud, 1976).

Software development requires the integration of the knowledge of intellectual capital of skilled professionals for both as individuals and as collaborators in a team. Such teamwork has become crucial for software firms, with team autonomy as a critical factor for work group effectiveness (Langfred, 2000). However, while individual work is becoming more autonomous, there is less interaction between group members, which may be a threat to teamwork effectiveness (Langfred, 2000). Kirkman and Rosen (1999) emphasized the importance of taking both individual and group effects into account, stating that "what is needed most in the team effectiveness literature is research that examines empowerment at the individual and team levels simultaneously".

The researcher had the necessary approvals from the firm's management to keep some authority to the team away from it the firm's management control for some time during the project.

2.3.4.7.3 Communication Channels:

A research made by Daft and Lengel (1986) proposed Media Richness Theory (MRT) which is one of the most notable theories of communication. MRT proposes that different media are more efficient conveying different information, and the selection of the medium should be aligned with the needs of the task; information can be equivocal, thus offered to misunderstandings due to the possibility of multiple conflicting interpretations of the same content or uncertainty due to a lack of information (Daft and Lengel, 1987). Uncertainty of the information can be high or low, communication media capable of clarifying ambiguous subjects are considered rich, while communications requiring a long time to achieve a common understanding or that cannot clarify different perspectives are classified as lean communications (Daft and Lengel, 1986). Rich communication channels should be used while managing ambiguous information, while leaner channels are suitable for processing well understood messages and standard data (Daft and Lengel, 1986).

Different communication channels were analyzed based on the categorization of Cockburn (Cockburn, 2000); he has described his perceptions of the effectiveness of different communication media. The

below figure represents in simple mind what are the categorization suggested by the MRT and Cockburn, and as noticed, the face to face communication is seen as the most efficient communication media.

Efficient communication is one of the most essential factors in SW development projects, and according to Komi and Tihinen (2005) is even more important in agile or distributed environments.

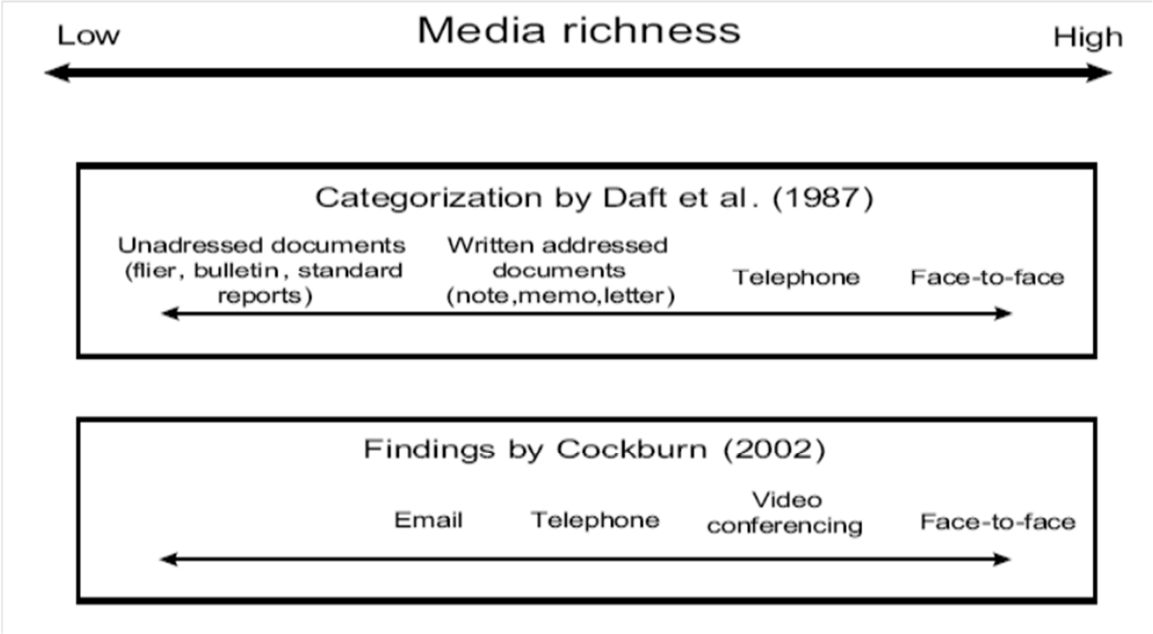


Figure 2.5 Communication Effectiveness off Different Communication Media

Thus, the researcher suggested to the local firm in its trial to follow the distributed agile development, to focus on establishing an efficient customer relationship, enabling more meaningful communication. However, not everything related to the communication channel worked perfectly since the customers have used to the traditional way of work and it seemed to them as weird to be involved in the process from the start of it,

which gave the researcher the idea to search in the communication with the customers to be addressed in a different research.

System requirements are being widely represented by User Stories (US); which is a key aspect in agile to make a fast communication between customers and developers (Cohn, 2004). In agile methodologies, the list of business requirements does not need to be complete at the beginning of the project and some of them come out during the development and to update the list of requirements (Cohn, 2004).

2.3.4.7.4 Leadership and Team Orientation:

Decision authority and leadership need to be shared (Morgan, 2006). Pearce (2004) argues that leadership should be rotated to the person with the key knowledge, skills, and abilities for the particular issues facing the team at any given moment. While the project manager should maintain the leadership for project management duties, team members should be allowed to lead when they possess the knowledge that needs to be shared or utilized during different phases of the project (Pearce, 2004).

All team members should also jointly share decision authority, rather than a centralized decision structure where one person makes all the decisions or a decentralized decision structure where all team members make decisions regarding their work individually and independently of other team members (Hoegl and Parboteeah, 2006).

2.3.4.7.5 Redundancy

Redundancy is any kind of excess capacity that can create room for innovation and development to occur (Morgan, 2006). Any system with an ability to self-manage must have a degree of redundancy, where the members in a team need multiple skills so that they are able to perform each other's jobs and substitute each other as circumstances demand (Morgan, 2006).

2.3.5 Artifacts:

Scrum introduces a number of new artifacts: Product Backlog, Sprint Backlog, Release Backlog, and Burn-down.

2.3.5.1 Product Backlog:

The Product Backlog is a to-do-list considering the product and each team should not have more than one Product Backlog. It is a result of the work done by the Product Owner and contains a number of items with different priorities. Items are requirements and desired actions that will lead to improvement of the product, for example, new features or bugs needed to be fixed. The Product Owner gathers all items in the Product Backlog and the items are the basis for modifications of the product. The priorities are set by the Product Owner and the priorities decide in which order the items must be done. The Product Owner bases the ordering of priorities on the customer's needs and the demands from the market. The Product

Owner is responsible for keeping the estimates up-to-date and as reliable as possible (Schwaber, 2004).

Since it is always possible to do some improvements to the product, the Product Backlog never will be empty or complete. The Product Backlog evolves when the environment of the product changes. When the customer's needs and requirements change, the Product Backlog must also be changed. The estimates of items in the Product Backlog are more precise the higher priority they have (Schwaber, 2007).

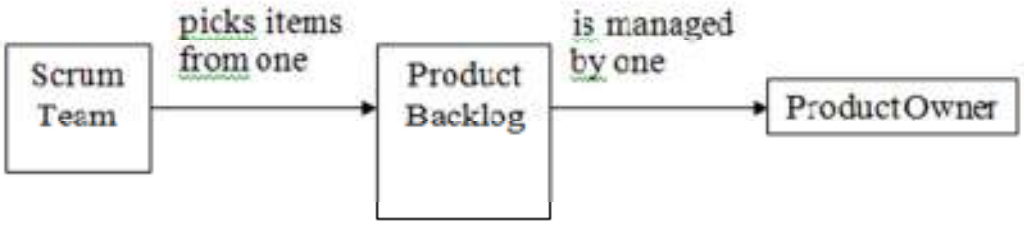


Figure 2.6. Relationship to the Product Backlog

Figure 2.6 shows how the relationship is between the Product Backlog, the Scrum Team, and the Product Owner. Each Scrum Team must have exactly one Product Backlog to pick items from. Each Product Backlog must also be managed by exactly one Product Owner.

2.3.5.2 Sprint Backlog:

A Sprint Backlog is a Backlog just like the Product Backlog, but the Sprint Backlog is limited in time to a Sprint and consists of all the items that are supposed to be done during the Sprint. The items for the current Sprint are the items from the Product Backlog with the highest priority. The

team breaks items down into smaller tasks. Tasks should have enough detail so that it is estimated to take 4-16 hours to finish, if not, the item should be divided to fit the time interval. Once the Sprint Backlog is set, only the Scrum Team can change it (Schwaber, 2007).

2.3.5.3 Release Backlog:

A Release Backlog is limited by the time before a release. It is a subset of the Product Backlog and consists of selected items for a release. For example when a new version of a product is released the Release Backlog contains items to be done before the release. The Release Backlog shall be estimated in days (Schwaber & Beedle, 2002).

2.3.5.4 Burn-down and Burn-down chart:

For each Sprint, a Burn-down is set up. It consists of all items that is planned to do for the current Sprint. When an item is completed, it is removed from the Burn-down, and the Burn-down decreases. When the Burn-down is empty, all the items in the Backlog are completed.

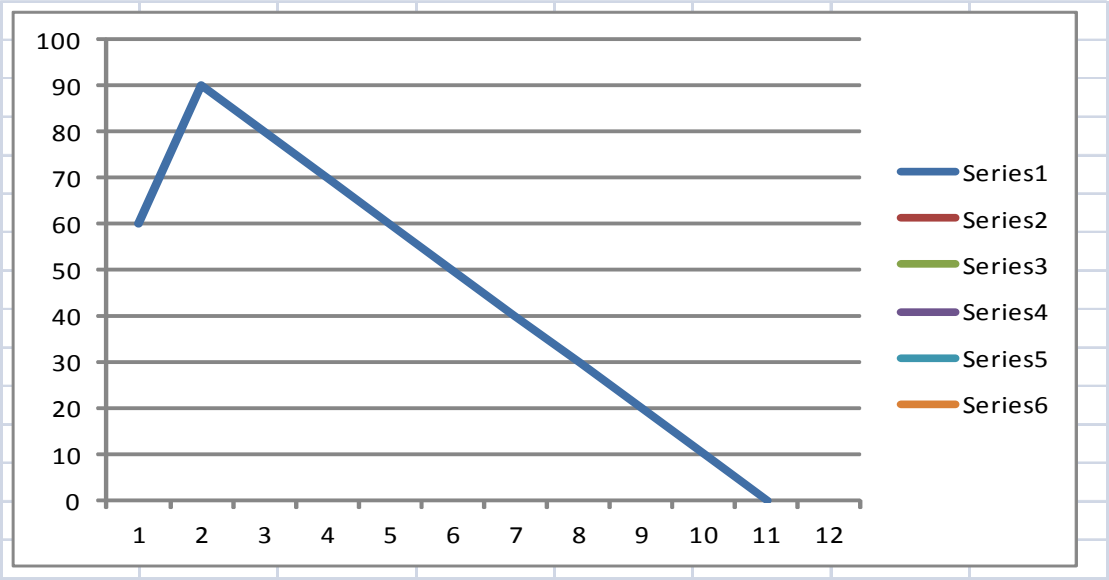


Figure 2.7. A typical Burn-down-chart.

A Burn-down is illustrated in a Burn-down chart, a graph starting at the sum of all estimated hours in the Backlog and when time is passing and items are completed the graph is declining. The Burn-down illustrates the estimated hours remaining in the Product Backlog, Sprint Backlog, or Release Backlog at any point in time (Softhouse, 2006). This shows how long time the project will last and a typical Burn-down chart is illustrated in Figure 2.7.

2.3.6 The Usage for Scrum in general:

The flow of the Scrum framework used in practice is indicated by Figure 2.8. Top priority items are taken from the Product Backlog to the Sprint Backlog, where they are being worked on during the entire Sprint. Each sprint is in turn divided into days, with Daily Sprint being held at the start of every one. After a number of Sprints, the customer accepts the

release, and a Final Product is delivered. The below graphical summary (figure 2.8) is provided by Schwaber and Sutherland (2007).

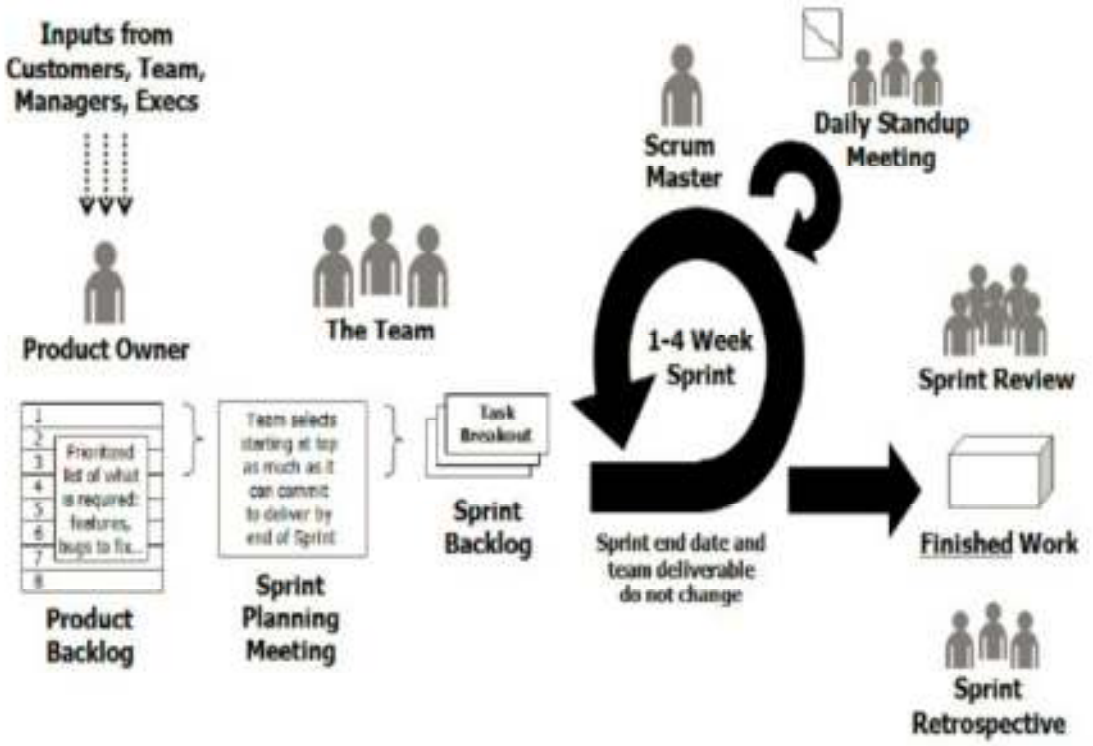


Figure 2.8. How Scrum works, based on case study

2.3.6.1 Scaling Scrum:

Schwaber (2004) refers to the requirement of more than one Scrum Team working simultaneously to get the desired functionality delivered on time for many projects as a scaled project, where the basic Scrum principles are used in a larger setting. Even if 800 persons are working on a single project, using Scrum they will still be divided into groups of about seven people each (Schwaber, 2004).

Before all the Scrum Teams starts working in parallel, the facilities, and methods for communication and technology use must be worked out.

This is done by one single Scrum Team working from a Backlog that combines non-functional items related to the scaling with items that provide real business value. Schwaber recommends the first team to be divided, with each member forming a team together with new team members. Each of the teams will then start with a Sprint Planning meeting and pick items from the Product Backlog (Schwaber, 2004).

According to Cohn (2007), one commonly used solution to scale the use of Scrum is to conduct a Scrum of Scrums meeting, where one member per Scrum Team participates. The Scrum of Scrums meeting is similar to the Daily Scrum meeting and the questions that all needs to answer is almost the same. Instead of answering the three questions as a person, the participants represent their own Scrum Team as a whole. This meeting especially focuses on areas of overlap and integration involving the different Scrum Teams. If the organization is big, the Scrum of Scrums meeting can be scaled up in recursive manner, which would result in meetings called Scrum of Scrum of Scrums, even though that name is rarely used (Cohn, 2007).

2.3.6.2 Suitable circumstances:

Depending on the projects and situations, Scrum could be more or less appropriate to use. According to Williams and Cockburn (2003), Scrum is most suited for “non safety-critical projects with volatile requirements, built by relatively small and skilled collocated teams”.

Although projects with several hundred people could have been successfully performed using agile development, it is best suited for collocated teams to be limited with number of people in total. It is also suggested that Scrum should not be used for life critical projects (Williams & Cockburn 2003).

2.3.6.3 Effectiveness of Scrum:

A lot of researches that focus on the effectiveness of Scrum had been conducted, like the research made by Rising and Janoff (2002) where they found that small teams can be flexible and adoptable in defining and applying an appropriate variant of Scrum, which lead them to suggest that Scrum is not an appropriate approach for large, complex teams. However, they found that small, isolated teams on a large project (like Scrum of Scrums) could be of use in some elements of Scrum. On the other hand, in many large companies the opposite can frequently be the case, with several teams in each department supporting several projects simultaneously (Cho, Kim and Olsen, 2006).

Another study by Marchenko and Abrahamsson was conducted about how Scrum can be used in multi-project environment and what challenges to be aware of during the adoption of Scrum in such environments. According to Marchenko and Abrahamsson (2008), there are many challenges identified:

- Placing an overemphasis on the Scrum process and practices.

- The SM cares only about the individuals and interactions.
- A lack of clear management expectations and actions.
- Too much maintenance and bug fixing undermining the team productivity.
- Fitting Scrum and short iterations into research intensive teamwork.
- Difficulty in tracking progress and in using the results of the tracking.
- Management interfering too much.

Cho, Kim and Olsen (2006) have conducted a case study on the applicability and the effectiveness of Scrum in large-scale and mission-critical projects, They find that it is best to decide the Scrum method in dependence of the nature of applications and the development environment because there is no single software development process that is the best approach for every project (Cho, Kim and Olsen, 2006).

Mann and Maurer (2005) have carried out a case study showing that after Scrum was introduced, customer satisfaction increased, while at the same time overtime for the developers decreased (Mann and Maurer, 2005). Many studies to illustrate the use and performance of fully distributed Scrum had been conducted; one was done by Sutherland (Sutherland, Schoonheim, Kumar, Pandey and Vishal, 2009) and found that Co-located teams are more productive than distributed teams and often

doubling productivity over teams distributed within the same building (Sutherland, Schoonheim, Kumar, Pandey and Vishal, 2009).

2.4 Related Studies:

Software development projects attract a great attention because of the great risk and failure percentages of such projects. Many old and recent studies knocked out the door the utilizing time, cost and mitigation of the risks of SW development projects, some of them had been addressed with a huge failure and some of them moved the process into new levels of success.

Victor Szalvay (2004) tried to demonstrate the short-comings of the waterfall approach while providing a solution in iterative, and more specifically, agile methods. He mentioned and explained the short comings of the traditional waterfall approach. Victor claims that under the waterfall approach, traditional IT managers have made valiant efforts to craft and adhere to large-scale development plans. However, studies of past software projects show that only 9% to 16% are considered on-time and on-budget. Then he turned into iterative and agile methods; the simple ability to revisit the “phases” of development dramatically improves project efficiency. The idea of revisiting phases over and over is called “incremental and iterative development” (IID). The development lifecycle is cut up into “iterations” and each iteration touches on each of the traditional “phases” of development. IID allows for multiple iterations, over a project lifecycle to

properly address complexities and risk factors. He addressed the agile methods as embracing the change, and spoke about the agile project management as an empirical process in terms of cost, schedule, requirements and quality. One of the biggest advantages to IID is that work can begin before all of the requirements are known. Quite the contrary, in the gained experience often the bottleneck in the development process has been the lack of availability of customer domain experts for detailed requirements analysis. IID is ideally suited then to take on bite-sized chunks of requirements that the customer can easily digest.

Victor concluded that the notion that Agile is a radical deviation from the long history of waterfall software development. Although traditional methodologies are still dominating, yet agile is simply the latest theory that is widely replacing the waterfall approach that itself will change and evolve well into the future.

Kristofer Gustafsson and Johan Jacobsson (2009) took Motorola as a case to make quality assurance with agile software development. Motorola have previously been using a project model based on a traditional waterfall model. However, the Research and Development department adopted the Scrum framework in their software development process. This was due to the perceived flexibility and adaptively Scrum offered compared to the previously used model.

Gustafsson and Jacobsson mentioned the limitation faced his research which was the secrecy of material belonging to Motorola. They then moved to speak about their research strategies for the research, and compared among the different strategies; the methodical orientation, the qualitative or quantitative research strategy, triangulation or case study, and their choice was to follow the case study since it resembles the empirical thinking. Then they determined the data gathering techniques which were through literature study and interviews.

The interviews were performed with developers, managers and consultants in a semi-structured form, with a list of questions forming the basis for the interview and additional questions added to highlight interesting topics. Notes were taken continuously by interviewers regarding the answers given by the interviewee.

Motorola realized they were in need of more efficient ways of developing software, the solution was a transition to agile development with the framework Scrum. A certain level of quality must also be assured of the delivered goods and services. His research questioned if it is possible to use the different agile software development and still meet the quality requirements. The conclusion is that it is possible to meet the quality requirements when using agile, under the condition that some additional processes are performed and that other parts of the organization also fulfills the remaining requirements. This is needed since there are requirements that are out of scope for the agile framework.

Zishan H. Mirza (2010) tried to illustrate the use of agile software development process in his paper. There are lots of factors to consider when evaluating a software process that fits an organization's needs. His research tried to illustrate how two organizations are using an agile based software development process in their projects and if agile software development process can be recommended to organizations to use in their software development projects. The data in his research have been collected from published articles and journals, as well as from interviews from two organizations.

He included a Theoretical Background section that contains the related theories that are available in published articles, journals, and conference papers. In the Research Methodology, his paper states how the research was initiated and done. The Data Collection section contains the answers to the interview questions; this section is supplemented by an appendix, where all of the interview questions that were asked are stated. An analysis of the data collected is made in the Result and Analysis section. This is where the data collected is related back to the theoretical background.

In his research, Zishan conducted many interviews at two organizations in order to understand how organizations are using agile based software development process. Both organizations that have been studied in this research are using an agile methodology in their software projects and are successful. There are a lot of benefits of using agile in

comparison to a traditional plan-driven development process. The main benefits of using an agile software development process are that it is a flexible method that can incorporate various changes in a project. These changes can be that the customers want to change the requirements. Another possible reason can be that it is not always possible that all of the customer's requirements are fully understood and it might take some time for the team to understand it fully. Therefore, it is important to be able to change a project plan and incorporate this change in a project. One of the interviewees mentioned that a plan-driven development ("Waterfall") process can be slow and not as effective as the Dynamic Systems Development Method (DSDM), which is an agile methodology. A plan-driven development process can be successful when the requirements are fully understood. Therefore, it is possible to conclude that agile can be positive for software projects and it can be recommended to use in software projects.

Jonna Kalermo and Jenni Rissanen (2002) tried to analyze Agile Manifesto and its applicability through conducting a literature review and an empirical case study. They used the qualitative approach to gain understanding and a thorough insight of the undefined processes of their case organization and find out the weaknesses and strengths of their case.

The case study was conducted to investigate the product development process of a corporate venture. The case study consisted of a current state analysis to gather information about the software product

development process, about the involved parties and their relations, as well as their working methods and working environment. The employees of the venture as well as other people relevant to the project were interviewed on structured interviews theme, and observation was not used.

Their conclusion was that none of the ideas presented in Agile Manifesto were truly innovative. The most obvious heritages of traditional methods in agile software development are iterative and incremental development. However, Agile Manifesto gives a quite holistic view on software development, in which not only technical issues are of importance but also customer satisfaction and human well-being are strongly emphasized.

Nichamon Chantachaimongkol and Puangpetch Sincharoenpanich (2013) conducted a study that entailed the critical factors for implementing the Scrum SW development methodology, their basic question was about the ability of an organization to succeed in managing Scrum implementation and about the critical factors in such an implementation, in the purpose of identifying and analyzing of the factors that are critical to the success in Scrum implementation. Their methodology based on a deductive approach using qualitative analysis where information was acquired from both secondary and primary sources. Based on a literature review of previous research, empirical data were collected from respondents in companies using the Scrum methodology both in Sweden and in Thailand. They concluded that the critical factors were divided into

three factors: organizational, people, and technical factor, whereas seven critical factors of Scrum implementation success were found to be crucial both through the literature review and the empirical study: Management Support, Customer Commitment, Work Place, Tools and Technology Support, Communication, Learning and Training, and Plan-driven Project.

Kokko (2013) also conducted a study about improving requirements management practices in agile software development environment. By utilizing the action research strategy, qualitative data was gathered using a comprehensive questionnaire. Altogether 36 professionals took part in the interviews, results were analyzed and an internal development project roadmap was developed. One of the target projects played an important pilot role during the internal development project. In this particular pilot project, changes were implemented into practice with a help of a project manager, leading architect and with a product manager who was also the mentor of this thesis work. A key finding of this study is that requirement's continuous, systematic and collaborative management improves requirements quality. Better requirements quality with improved project practices are key factors in successful software development projects.

Vohra Surabhi (2013) attempted to combine traditional software development approaches with agile methods. Agile testing practices have been suggested to be incorporated inside traditional software development approaches with the focus on V-model. Four hypotheses emphasizing benefits of agile testing methods inside V-model have been formulated.

Three case studies i.e. three projects using agile testing methods inside V-model have been presented and compared for weaknesses and strengths. The results of his research and case studies analysis indicate that Scrum can be used as a framework inside V-model, within which other agile testing methods like Feature Driven Development (FDD), FDD was found to be least agile and best candidate for adoption inside V-model, and that the hybrid approach combining agile testing and traditional methods provides a balance between agility and stability.

Chapter Three

RESEARCH METHODOLOGY

Chapter Three

RESEARCH METHODOLOGY

3.1 Introduction:

The researcher used two methodologies in this research, the first one was the interviews and the other one was the case study. The interviews were done by many respondents who met predetermined conditions set by the researcher to ensure the validity and reliability of the interviews. On the other hand, the case study was on a project inside a local firm that wished to implement the agile Scrum methodology for the first time instead of the traditional waterfall methodology that was being used until then.

Case study: The research conducted in this paper builds on a case study where an agile methodology –Scrum- was implemented on a SW project at a local firm instead of a traditional methodology that was common in this firm, and observations, notes and conclusions were made out of this research.

Interviews: In order to support the findings and provide more strong foundations, interviews have been conducted at many other local organizations in the SW development field. The interviews questions that had been asked can be found in the interviews part in this thesis. The data that were collected from the interviews and analyzed based on the theoretical background on how organizations are using agile software

development process, and used in the case study as a supplementary knowledge provided to the Scrum team.

This chapter shows an overview of the methodological approach the researcher used for constructing his research on it; empowered by the literature review that will help the researcher in selecting the way to conduct the analysis. The wide interviews that been made, the targeted population and the case study that been managed, all of them are to establish a base to research upon it and to conclude from it in order to make a small trial in the road of change and upgrade of the SW development industry that is being suffering a lot around the world, and especially in the local markets.

3.2 Research Population:

Population is "the whole groups of individuals, phenomenon, or things that we aim to generalize our study's results on" (Alhamdani et. al., 2006). This research has two main methodologies to depend on, on for the interviews and the other for a case study in a local firm.

Population for the interviews: the population is all the Palestinians SW development firms that are listed in PITA website in West Bank area, which consists of 80 firms, distributed in different IT development fields, where the firms in other Palestinian lands weren't available in this research; the researcher here focused on the firms that have a SW development as one of their main business lines. The researcher discriminated between

firms that have SW development as a business line and those who don't by emailing them, calling them or asking people.

Population for the case study: the researcher chose a local firm that usually implements the traditional methodology, and wishes to transfer for a new development methodology and implement it on a SW project of its own. The firm was already trying to migrate from one methodology into another, but couldn't trigger the change fearing from shortage of readiness to do so. The researcher knows this firm well enough from the market; so when he knew of their idea of change, he encouraged the firm to start it, and offered ideas about agile Scrum and volunteered to provide the knowledge he gained from the literature and to transfer it in training sessions.

The researcher took the observations for this project on all over phases of the project besides the reporting and information got from certain team members.

3.3 Research Sample:

Elder (2009) states that there are seven types of sampling methodologies that any research should follow at least one of them depending on the nature of the research. According to Elder, the types are:

3.3.1 Probability Samples:

Probability samples are the most accurate of the sample selection methods, they are sometimes known as random samples. Any survey aimed at generalizing results drawn from a sample to the whole population of interest must be based on probability sampling. With a probability sample, the first step is usually to try to find a sampling frame. Using this frame, individuals or households are numbered, and some numbers are chosen at random to determine who is surveyed.

3.3.2 Quota Sampling:

Quota sampling refers to selection with controls, ensuring that specified numbers (quotas) are obtained from each specified population subgroup (e.g. households or persons classified by relevant characteristics), but with essentially no randomization of unit selection within the subgroups. No population list is used, but a quota, usually based on census data, is drawn up.

3.3.3 Purposive Samples:

A purposive sample refers to selection of units based on personal judgment rather than randomization. This judgmental sampling is in some way “representative” of the population of interest without sampling at random. One of the commonest uses of purposive sampling is in studies based on very small numbers of areas or sites. In these studies, variability

with random selection is expected to be excessively large and, hence, potentially more damaging than the bias inherent in selection by judgment. The areas included may be determined on the basis of judgment although, within each area included, the selection of ultimate units may be randomized. Generally speaking, if the budget is small and only a small number of towns and cities can be included, we may choose these in a purposive way, perhaps ensuring that different types of town were included.

3.3.4 Snowball Samples:

In some communities (especially those in developing countries), the only feasible way to find its members is by asking other members. The first step in this procedure is to find a few members of the population using any method. This step is denoted as the first round. Then you ask each of these first-round members if they know of any others. The names given will form the second round. Then you go to each of those second-round people, and ask them for more names. This process is repeated for several more rounds. The process is stopped when you start hearing about the same people over and over again.

3.3.5 Volunteer Samples:

A volunteer sampling procedure might be used when the above procedures are not possible. In general, samples of volunteers should be treated with caution. However, since all survey research involves some

degree of volunteering, there is no fixed line between a volunteer sample and a probability sample. The main difference between a pure volunteer sample and a probability sample of volunteers is that, in the former case, volunteers make all the effort; means there is no sampling frame is used.

3.3.6 Stratification:

A more representative sample can be selected using the stratification procedure. The basic idea here is to divide the target population into strata (groups) based on characteristics that you think are important. Stratification leads to reduced sampling error because it can ensure that all relevant portions of the population are included in the sample. Stratification is easy to do, and it should be used whenever possible for optimal coverage purposes.

3.3.7 Combined Sampling Methods (Multi-stage sampling):

In national-based surveys, sampling is done in several steps. The first step is usually to choose a purposive sample of governorates from the total number of governorates in the country. In the second stage, a stratified sample of districts within each governorate is selected. This procedure continues until the sample of targeted individuals is determined.

Since this research has two methodologies, the researcher had to apart the sampling for the interviews from the case study. For the case study, the only sample to take was the case study that the researcher took

his observations during the project and received many reports about the march of the project for the research purposes, while for the research sample for the interviews the story is different. This research part depends on a relatively small number of areas, no random choices of interviewees and depends on personal judgment rather than randomization, which means the sample method the research found best to assign and follow is the purposive way of sampling. The personal judgments of the individuals in this research only depend on those judgments that had been taken out from experts who have particular experience that is most likely useful in the advancement the researcher's interests and potentially open new doors; the theme is an expert sampling one more than any other kind of purposive sampling.

This research sample consists of a part of the research population which consists of 80 Palestinians firms that listed in the PITA website from within the area of WB. The following table displays the sample classified by firm type:

Table 3-1: Research Sample classified by Area

Firm IT Field	No. of Firms	Percentage
Bethlehem	3	3.75%
Ramallah and Al-Bireh	63	78.75%
Hebron	3	3.75%
Nablus	10	12.5%
Qalqilya	1	1.25%
Total Companies That Has SW Industry	80	100%

3.4 Triangulation:

By using several different and independent sources and/or methods to verify the same phenomenon, the validity is improved. This technique is known as triangulation and results in a more complete answer. The results from different methods or sources may either coincide or differ. If the result differs, additional investigations are needed. If the results coincide it is most likely a true picture of the situation, and it is these results that are the most interesting in the study (Gustafsson & Jacobsson, 2009).

The interviews at the local organizations were the form of semi-structured interviews. During these interviews, notes were taken on a recorder by the interviewer; these notes had been taken on a recording machine after asking the permission from the interviewee and then after the interview, all of the notes were drafted and rewritten neatly. The interviews were made with many parties; including developers and managers. Two interviews had been done with developers from different firms and they had plenty of knowledge about tools and work environments, two with team leaders who also working as developers and were part of technical and operational teams, two with SW engineers and two interviews with project managers.

However, many of the interviews weren't enough for one session, so the researcher was forced to conduct more through email and in form of structured interview. The historical data were available from the projects

developed inside the firm; another part of the historical data could be got from questions within interviews where the interviewees were getting back into their projects archives and get the researcher some explanations for the questions been asked during the interviews; usually these reviews were being done through emails.

When creating the interview questions, books about interview techniques were used in order to make the questions as clear and obvious and as understandable as possible (see chapter 3.7.2 Primary sources); there are a lot of useful books to get useful information from, like “Using Semi-Structured Interviews in Small-Scale Researcher: A Teacher’s Guide” (Drever, 1995) and “Qualitative Researching” (Mason, 2002). There were basic questions that had been created within the sight of the research objectives and questions like questions about testing procedures, customer relations and satisfactions, development methodologies used and other related questions about the interviewee expertise, however, since the interviews are semi-structured, a lot of questions were emerging during the interviews and then used in the later interviews and so on.

For the case analysis, a local firm has accepted to implement agile Scrum on one of its SW development project, where the researcher distributed educational agile Scrum material and utilized the knowledge gained from the former interviews. Then after the development was set up, the researcher started taking periodical reports about the development status, and kept on taking observations during the while period about the

team members, work environment, management behavior and commitment and other observations that utilize the research purposes. The researcher then had an access over a SW project that had been developed under the traditional methodologies and had the same characteristics like the case study project in terms of nature, purpose, team, time and budget. Through this access, the researcher tried to have a comparison between the old and new projects in terms of customer satisfaction, change requests, support agreement, internal environment for developers, team work and self-organizing, leadership, management behavior, customer involvement and budget and time limits.

3.5 Applied Research Strategy:

Usually there are three main steps when conducting a research, starting with the understanding and definition of the problem, the continuing with how it shall be studied and what approaches will be used, ending with preparation and execution of the research (Malekzadeh, 2010). The following figure provides a visual explanation of the research approach applied in this research.

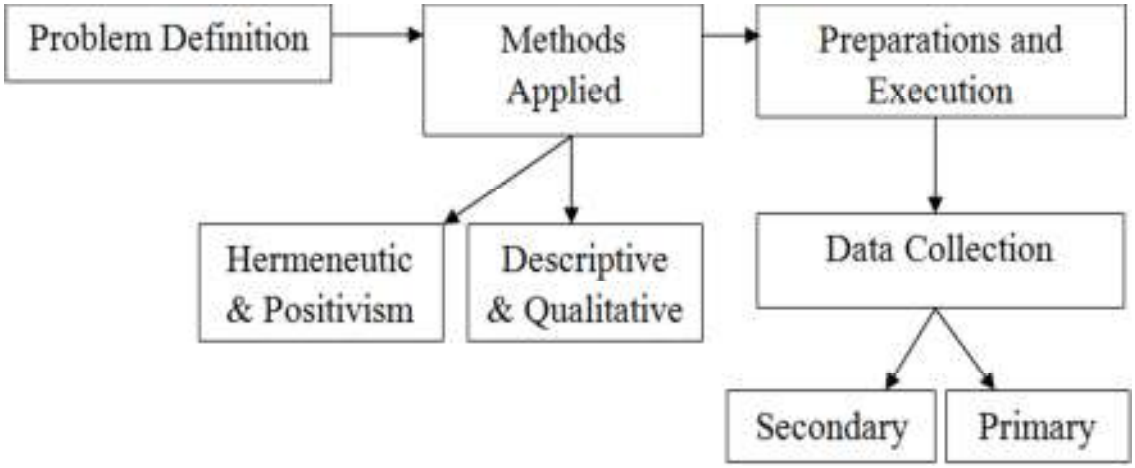


Figure 3.1 Research Process

3.5.1 Positivism and hermeneutic:

Methodology is the style or method researchers follow in conducting their research. Often researchers select the research methodology according to the nature of the research itself. Each research has its properties and uniqueness (Alhamdani et. al. 2006).

As a researcher, it is crucial to determine the scientific approach to follow. According to Bryman (2004), there are several ways to relate to knowledge and research, like Positivism and Hermeneutic. Positivism is usually thought of as an attitude towards the theory of knowledge where only a phenomenon observed and verified through the senses is considered knowledge. It is usually applied to scientific research in natural science and aims to test a hypothesis based on previous knowledge that can be proved or disproved objectively from the surrounding research environment. The test should give the same result every time it is performed, even if the

scientist is replaced, and the result could therefore be extended to a universal explanation (Bryman, 2004).

On the other hand, Hermeneutic theory stands in the opposite of Positivism, since it claims that the findings needs to, and will be, interpreted before any conclusions can be drawn. The Hermeneutic theory is used in several different scientific disciplines, but foremost in human, culture and social science (Gustafsson & Jacobsson, 2009).

A Hermeneutist argues that human reality can be interpreted through the language. The Hermeneutic approach of gaining knowledge is through observation and interpretation of human actions (Gustafsson & Jacobsson, 2009).

Personal opinions and human interactions are the main tie for the scope of this research, so a hermeneutic orientation is most suited. This research includes interviews, observations, and literature studies, and all of these are affected and influenced by persons understanding and personal opinions about the subject. It is impossible for a person to read literature and not be affected by personal opinions and previously knowledge in relation to the subject, since reading demands a certain amount of interpretation.

3.5.2 Quantitative or Qualitative Research Strategy:

For any study, it is common to categorize it as either qualitative or quantitative. A qualitative study is usually used to achieve a greater comprehension on a specific subject, a specific issue, or a specific situation (Gustafsson & Jacobsson, 2009). The qualitative study can for example be interviews and observations and it is used to answer questions that are more complex and when a deeper understanding for a specific subject is desired and a greater amount of interpretation is demanded, quantitative research strategies are however mostly used when the studied information is numerically measurable; it seeks to understand the research problem from the local population where the researchers often try to get specific information about the phenomenon, variables, and values by exploring opinions, behaviors and perceptions of a particular population (Gustafsson & Jacobsson, 2009).

Examples of quantitative research methods are interviews, questionnaires and surveys (Gustafsson & Jacobsson, 2009).

The research usually has to follow either qualitative research or quantitative research or both of them, and to help the researcher to determine which to use or follow, the problem definition can be used. If the problem is defined as: Where, How, or what are the differences, a quantitative research method is appropriate to use. If the problem is about interpreting and understanding, for example peoples' opinions, then it is

better to use the qualitative research method. However, it is possible to perform research founded somewhere in-between the two methods (Gustafsson & Jacobsson, 2009). According to Creswell (1994), “the qualitative study approach is considered an appropriate method when little is known about the phenomenon under investigation and the concepts are immature due to lack of theory and previous research and a need exists to explore and describe the phenomena”.

Based on the researcher’s interviews with a vast variety of local IT firms, he found that agile software development is not widely practiced across our country because of the less of practical knowledge about the subject in the local market besides the relatively new attention given to the subject. Agile software development of short iterative cycles offers an opportunity for rapid, visible and motivating software process improvement. The agile principles suggest the regular reflections of agile project teams for improving the efficiency and adaptation of the process. However, current literature provides little support or empirical evidence for conducting such improvement efficiently, systematically and in a validated manner. Thus, this thesis proposes an Iterative Improvement Process for conducting software process improvement within individual agile project teams, which aims at increasing the ability of software developers to improve the development process based on their experiences and context knowledge. Qualitative approach involves analysis of data such as words, pictures, or other objects which usually are gathered through interviews,

literature or other artifacts. The quantitative approach involves analysis of numerical data and the aim is to classify features, count them, and construct statistical models in an attempt to explain what is observed, however, the opinion that these two approaches are in most cases combined and researchers are using both in parallel is also presented strong (Patel and Davidson, 2003).

This thesis uses the qualitative research methodologies since it is like finding why and how more than a quantifying theoretical hypothesis due to the nature and content of the topic, which dictates to follow the qualitative methodologies over the quantitative ones specially that no quantitative data have been collected or analyzed. The methods for collecting data are described in the following sections.

3.6 Case study:

The research in this paper depends on two data sources, interviews and a case study. A case study is an empirical inquiry that investigates a contemporary phenomenon within its real-life context, especially when the boundaries between phenomenon and context are not clearly evident (Yin, 2007).

Case study as a research strategy embodies all aspects of research, including test design, data gathering techniques, and approaches to data analysis (Yin, 2007). A case study design should be considered when the focus of the study is to answer “how” and “why” questions, or you cannot

manipulate the behavior of those involved in the study, or you want to cover contextual conditions because you believe they are relevant to the phenomenon under study, or when the boundaries are not clear between the phenomenon and context (Yin, 2007).

Yin (2007) categorizes case studies as explanatory, exploratory, or descriptive. The Explanatory is a type of case study that would be used if you were seeking to answer a question that sought to explain the presumed causal links in real-life interventions that are too complex for the survey or experimental strategies. In evaluation language, the explanations would link program implementation with program effects (Yin, 2007). While the Exploratory is used to explore those situations in which the intervention being evaluated has no clear, single set of outcomes (Yin, 2007). The Descriptive is used to describe an intervention or phenomenon and the real-life context in which it occurred (Yin, 2007).

On the other hand, there are another three categorizations to describe case studies types: intrinsic, instrumental and collective proposed by Stake (Stake, 1995). According to Stake (1995), an intrinsic case study is conducted when a researcher is interested in a unique situation; means that he has an intrinsic interest in the subject and he is aware that the results have limited transferability. If the intent of the case study if to gain insight and understanding of a particular situation or phenomenon, then the researcher is advised to use an instrumental case study to gain

understanding. This author also uses the term collective case study when more than one case is being examined.

For this research, the exploratory type of case studies was used; since the researcher took the observations all along the project was running, and didn't interfere with the track of work that was planned originally. However, the researcher offered the advice and training sessions about agile Scrum before the project starts in order to share his knowledge with the team and gives a promotional and motivational factor for the team to adopt the new so-called Scrum introduced to them. In general, the research was observational over the project inside a local firm, in addition to interviews made with a lot of other firms about the project.

When designing a case study it is possible to take two different approaches: single- or multiple-case study. Multi-case designs combine different case studies into a single analysis, where single-case design concentrates on a single case. Multi-case designs are usually preferred because of the added analytical benefit the comparison of two or more cases can give. However more resources, especially time, and knowledge, how to choose and analyze the different cases, are generally required (Yin, 2007).

When preparing a case study, something called a case study protocol is written. It contains the procedures and general rules to be followed by the researcher, together with more specific questions that is the overall aim

of the research. These questions are not used during interviews but are a reminder to the researcher about the important topics in the study. Through the protocol the reliability of the study is increased since it documents the way the study was performed, enabling other researchers to replicate it later. It is also especially useful in multi-case designs where it ensures that the same questions are asked in the different cases (Yin, 2007).

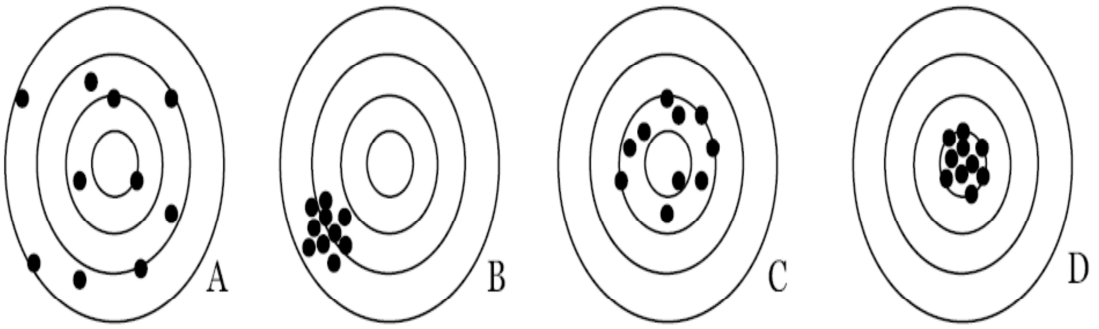
Six different types of sources can be used in a case study: documents, archival records, interviews, direct observation, participant-observation, and physical artifacts. They are all useful in different situations depending on the context and aim of the study. More than one source of information should generally be used to give a more balanced view of the subject (Yin, 2007).

Documents and archival records are persistent and contain a lot of information, but might also be hard to find or gain access to and must be interpreted by the researcher. Interviews might give answers to specific questions related to the study, but the answers might be biased because of the interaction between interviewer and interviewee. Direct observation and participant-observation describes real-life events in its context, but takes time and events might be altered because of the presence of observers. Physical artifacts describe cultural or technical aspects, but might not be accessible or relevant enough (Yin, 2007).

The question about anonymity is common in case studies. Should the name and location for the entire study be revealed, and should individuals participating in the case study be identified. The most desirable option is to disclose all available information. This enables the reader to use any previous knowledge about the case, and to verify and check the details of the study. However, sometimes anonymity is necessary (Yin, 2007).

3.6.1 Reliability and Validity:

Case study designs need to maximize their quality through four critical conditions related to design quality: construct validity, internal validity, external validity and reliability (Yin, 2007). The difference between reliability and validity will be explained and illustrated in Figures 3.2 and 3.3.



A: Neither reliable nor valid. B: Reliable but not valid. C: Valid but not reliable. D: Both reliable and valid.

Figure 3.2 Reliability and validity (Björklund and Paulsson, 2003)

TESTS	Case Study Tactic	Phase of research in which tactic occurs
Construct validity	♦ use multiple sources of evidence	data collection
	♦ establish chain of evidence	data collection
	♦ have key informants review draft case study report	composition
Internal validity	♦ do pattern matching	data analysis
	♦ do explanation building	data analysis
	♦ address rival explanations	data analysis
	♦ use logic models	data analysis
External validity	♦ use theory in single-case studies	research design
	♦ use replication logic in multiple-case studies	research design
Reliability	♦ use case study protocol	data collection
	♦ develop case study database	data collection

Figure 3.3 Case Study Tactics for Four Design Tests (Yin, 2007)

Reliability is used in testing and measurements, and is a measure of the authenticity of the test or measurement. For example is it not especially reliable to measure long distances by pacing with your feet. Thus pacing is a measurement with low reliability. In this case, a measuring-tape would have given a result with higher reliability (Gustafsson & Jacobsson, 2009).

Reliability ensures that another researcher should be able to do the same research and end up with the same result. This requires the methodology to be well documented and the research conducted in a way that could be repeated (Yin, 2007).

Validity states whether a measurement really measures what it is intended to (Bryman, 2004). For example if an intelligence test measures a

person's memory capacity, it measures an important aspect of intelligence but not all parts. Thus, this intelligence test has low validity since it does not present a complete measure of a person's intelligence (Gustafsson & Jacobsson, 2009).

Construct validity tests if the definition of the subject of the research is clearly defined. Otherwise, it is hard to know if the result actually presents any objective evidence or if it is just the subjective opinion of the researcher that has been given a scientific look. Why a certain aspect has been studied and why the selected measures are valid are two important questions the researcher has to answer in the report. Using multiple sources of evidence, results in a chain of evidence that can be followed and understood by the reader. Key informants review the draft to ensure that the results are in line with their description (Yin, 2007).

If the research infers that an event is based on an earlier occurrence, internal validity has to be considered, like if there could be another theory that leads to the same result or a rival explanation (Yin, 2007).

External validity is related to the generalization of the result. Unlike for example a survey, the result from a case study is not statistically generalizable, with the particular case seen as one sample, to all populations, instead, the result should be expanded and generalized into a theory, which can later be proved or disproved by replicated cases in

different environments, which is similar to the way experiments are used in science (Yin, 2007).

In general, data collection and data gathering is something we are doing on daily basis, in some cases we are getting information from sources that are already validated and approved by others, most often these sources are used broadly and generally by many collectors, however when we construct the tools to collect data we are not sure if we get the right information (Malekzadeh, 2010).

Any challenge must be dealt with in two ways, firstly we must be sure that we are investigating what we aim to (validity) and secondly we must be sure that we are doing the research in a reliable way (reliability); reliability and validity are not excluding each other and must both be considered separately (Malekzadeh, 2010). Validity can be achieved in two ways; the first one is validation of the content by letting an external partner review the content and analyze the validity of the topics and the content (Malekzadeh, 2010). The researcher used this validation method within this research where the validation has been performed by the supervisor of the researcher, and both the topics and content had been analyzed and discussed.

The second approach when verifying the validity is by conducting a parallel validation where the tool developed is used in other but similar circumstances. In most occasions it is about using different techniques

when studying the same topic (Malekzadeh, 2010). This approach has also been implemented within this research by studying different architecture frameworks. After the interviews, the notes were combined and summarized for each interview. In order to increase the validity, all interviewees were given the opportunity to read a summary of their own interview to help minimize misunderstandings and give the interviewees a chance to adjust their statements. If requested the notes were corrected and the updated versions used for the summary instead. However, only a few of the interviewees actually had any comments on the notes other than an agreement.

The different theoretical frameworks used have been a very good base for validating the context and content of this thesis. Many tools had been used to ensure validity in this research according to this approach, during the case study or during the interviews. When the researcher made the interviews, he used to record all the interviews on a recorder, and then he used to draft the interviews and review them again to make sure of the questions and dictation, and based on this, many questions had been re-asked about some related issues. During the case study, many tools were used by the Scrum development team; they used the Google docs at first to track the tasks of the project, and then they converted into JIRA tracking tool provided by Atlassian especially designed for agile Scrum. During the development, no product or feature to go on live without being tested and reviewed by the Product Owner and the development team.

Reliability is focusing on two aspects of a research where the first is the level of consistency of a measure of a concept (internal) and if the result of the research is repeatable (external) (Bryman and Bell 2007). In this research, the researcher is dealing with traditional vs. modern SW development methodologies, which is a topic that had –and still having- received a lot of attention both from the academic world (a lot of studies) and the industry. As a result the measures and frameworks used have been aligned and standardized which makes them more generic than a set of specific procedures.

Another important issue when discussing reliability is concerned with the sources; the researcher has done a lot of efforts to get the reliable resources and collect the relevant data in which the researcher was dealing with well-known topics from within the industry that simplifies the process of getting the relevant information, and it is expected that if any other research with the same topic uses the same resources, then there is a big probability that it will come up with the same results.

As a measure of validity and reliability, a comparison had been made between the project in this research and some projects in the history of the local firm mentioned in this research. The researcher asked for statistics about the support queries asked by customer to fix a bug or to enhance a feature compared to the support queries previously asked by the customer on similar projects worked on with the traditional way of development within the same local firm. The researcher used the number of queries

ordered by the customer, customer satisfaction, management satisfaction, team satisfaction, over cost, time schedule commitment, speed of delivery, acceptance to change and the morale of work within team. The researcher made some interviews with the team and had access to historical projects within the firm in order to get the background and answers about his comparison measurements.

3.6.2 Case Study Protocol:

This Case Study Protocol is outlining the approach to the Case Study the researcher will perform. The Protocol is used to guide and control our efforts to be in line with the assumptions and expectations set at the start of the study. The purpose, scope and problem definition of the Case Study is the same as the Master thesis as a whole.

3.6.2.1 Data Collection Procedures:

The Case Study was performed at a local SW firm during 8 weeks from June 2012 to August 2012. Information was gathered through literature studies, observations, and interviews. The interviewees were from within the firm and from outside the firm in different companies. Observations can be used as a feed back to review the interviewees and getting better results during the development process. For those interviewees from outside the firm, their interviews were part of the knowledge base that the researcher used for the case study in this research.

The researcher used literature study and interviews to gain his knowledge base and to transfer this knowledge into the team members during two training sessions; one theoretical and one practical session, in order to facilitate the transfer as much as possible and implement the new methodology in a flexible way.

3.6.2.2 Case Study Questions:

These are the Case Study questions and the ideas where we might find relevant information. The Case Study questions are brought from the problem definition.

- **For the local area, how can we enhance the probability of SW project failure into more suitable probability? And by suitable we mean less failure for the local market, more customer satisfaction and less time consumption.**

  This can be determined by making conclusions from the answers in the four questions below.

  - Is the current SW procedures followed in this current local firm fruitful enough? [i.e does the project satisfy all the parties with the current procedures?] What are the current procedures followed in the firm?
  - Would changing the SW development methodology from a traditional one into Agile (Scrum) make the results better? [i.e

detecting the reasons of failure and raise the possibilities of successful projects,…]

- **How is Scrum explained in existing literature? And what is the applicability of Scrum in Palestine?**

3.7 Data Gathering Techniques:

Data collection is an important element in every study. In this section different approaches of collecting data are presented. Basically there are two different sources of data: secondary and primary data sources. Already existing data is referred to as secondary data, such as books, magazines, internet sources, and publications. Secondary source of information is the second-hand information about the happenings. Primary data is data observed and collected from first-hand sources in various ways, such as interviews, surveys, or questionnaires (Bryman and Bell 2007).

3.7.1 Secondary Sources:

The data used in this research has depended a lot on the secondary sources of data. Secondary sources have been used in defining the theoretical background and the framework. The research was initiated by the process of reviewing literature from previous research. The aim of the theoretical background was to understand and describe the topic field. In this category of data, the researcher searched for information in Al-Najah University library, literature, academic publications, internet sites and news articles. A lot of the e-books were found published through internet with

the help of Google books; and the other books were found through the different books sites.

3.7.2 Primary Sources:

The data used in this research were mainly originated from the primary sources of data. This research is about enhancing and more controlling the SW development in the local environment of SW development industry; such a mission can't be done without being fully involved in the process of development for a SW project within a firm that work totally in the local environment. That local firm was taken as an example, and a SW project within it was followed and observed in the full details in order to gain the main difficulties and obstacles to make the SW development process easier to do. On the other hand, interviews with a lot of other local firms had been made in order to support the results and inferences that the researcher had from the local case, and to make things more understandable and clearer for the researcher and the reader of the research, and to make the inference less biased by the opinions the researcher already have.

In the case of interviews they can be carried out in several ways: structured, semi-structured, or unstructured (Bryman and Bell 2007). The main method used to collect primary data in this research has been semi-structured interviews. Applying semi-structure interviews allowed the researcher to ask specific questions though leaving some space for open

discussions and analysis to the respondent. The semi-structured interviews did also allow the researcher to collect much more data from the interviews.

When collecting the primary data with the help of the selected respondents, a qualitative approach was adopted. According to Patel and Davidson (2003) a qualitative method enables a deeper and more complete understanding of the research area and its complex nature in contrast to a quantitative method. Since personal interviews allow a very high level of interaction, this form of qualitative method was strived after. In this research, seven interviews were conducted. According to Davidson and Patel (2003), the reliability of interviews is increased by having trained respondents. When selecting the respondents the researcher has been trying to identify trained respondents according to two selection criteria.

- The interviewee (respondent) should have long experience that ranges between 5 to 6 years experience or more, in the field of SW development either practical or theoretical; no interviewees had less than this experience.

- The interviewee should have earned his experience in the local market of SW development, and is preferred to be having additional knowledge from the outside markets, since the outside markets may have added some values to the interviewee expertise and makes him more open minded about the local market shortages.

When contacting the firm that the researcher intends to make the interview with it, he asks for someone that meets the criteria mentioned above and asks the firm to nominate someone has them, or if the researcher already knows the interviewee personally, he asks him for the interview after taking permission from the firm. In total, there were seven interviews were done face to face, and to ask more questions emails and telephones were used. All the interviews were semi-structured, although the interviewees had been told about the subject that they were interviewed about.

The following respondents were selected after contacting them directly or after contacting their companies requesting for interviews within the determined criteria for interviewees:

1- Mr. Ala Thiab, a SW project manager and a team leader at a local company. He worked for many years in SW development in traditional and agile –scrum- methodologies.

2- Mr. Hani Abughazaleh, a SW project manager and a team leader at a local company. He worked for years in SW development in agile scrum.

3- Mr. Rami Shahwan, a SW engineer who worked on traditional methodologies for many SW projects and then moved into working on SW development in agile scrum.

4- Mr. Mohammad Nassar, a SW developer who worked many projects on traditional methodologies then worked on SW development using agile scrum methodology.

5- Mr. Diya Awad, a SW development team leader who worked for years in SW development using the traditional methodologies.

6- Mr. Ahmad Othman, a SW developer who worked for years on SW development using the traditional methodology.

7- Mr. Waseem Azzam, a SW developer who worked for years on SW development using the traditional methodology.

It is important to mention that the researcher have contacted a lot more contacts than the previous contacts, but the fact that a lot of them didn't respond to the researcher made him to select only the previous ones. Emails was sent to the whole population, and phone calls had been made with many contacts in the population; many of them refused regarding the confidential information they have, others didn't reply at all, but the previous contacts were the ones who responded and gave a great assistance in both information and knowledge relating to the subject field.

Chapter Four

ANALYSIS AND DISCUSSION

Chapter Four

ANALYSIS AND DISCUSSION

4.1 Introduction:

This research was about applying agile SW development methodology to a SW project within a company that is used to work only with the traditional SW methodologies like the waterfall model to design and develop all of their products. The researcher convinced the company to try the transfer from old methodologies into agile scrum on only one medium project that is limited to two or three-month period. It was decided that Scrum should be deployed for a small SW development project, and the team was about seven employees and the first Sprint was started.

For this research, the researcher used two main methods to build the research upon them; the first one is the case study in a local SW firm and the second one is the interviews. The case study took about two months of work, and a month before it preparing for it. The local firm was using a waterfall project model to design and develop all of their products. A project team is lead and directed by a project manager. When the researcher of this research asked for the necessary approvals from the firm management, he didn't find much objections since the management already familiar with this notion from before because of some employees from within who had some knowledge about the subject from general references; the new thing the researcher introduced here was getting the full

commitment from the management toward agile scrum and that it can be applied to a certain SW project that the management interpreted it as a small size project as a phase two for a previous project. The same team members were gathered, but the idea of agile scrum was introduced to the team, and each member of the team started to read about agile Scrum himself in addition to the presentations that the researcher made to the team based on the knowledge he gained from the Scrum Handbook (Sutherland, 2010) and from the interviews he had made with many local SW organizations.

According to Lawrence and Yslas (2006), changing the development model from traditional waterfall towards agile methods is a challenge to any organization (Lawrence and Yslas, 2006). Successful transformations from traditional to agile software development have been reported, which encourages organizations to take agile methods into use (Schatz and Abdelshafi, 2005).

This research takes two parts into considerations regarding its data sources; the interviewees made with many SW firms located in Palestine, and the other part is the case study project that was implemented inside a local firm and during it the observation and notes had been taken during it. The interviews were used to support the knowledge base got from the literature that the researcher used to build the case study project on. Sometimes the interviews were being used to support the knowledge base, but they were also a strong confirmation for the results attained from the

case study, and in many occasions, the interviews had the role to be the idol for practical situations and problems that faced the team during the project.

4.1.1 Project Description:

The case study was a development project for a local internet service provider (ISP) that wishes to make a new ADSL provisioning system for its services and switches. Previously, all applied orders related to this ISP were executed on the switches form the back office, now all those orders became provisioned from a third party that is integrated into customer relationship system (CRM). When any subscriber orders ADSL service, weather a new service or upgrade or downgrade speed, disconnect, add new service, all will be added after a fulfillment is done from the CRM side and an automatic provisioning will be done for them. This provisioning system is done within a .Net environment since this is the most suitable for the requested system. The new way of work can be started by the orders that are being inserted into the CRM, and then the integration system sends all those orders into the back end (switches) after they took all the approval cycles. Sometimes there would be some problems like technical and business issues like moving telephone lines from place to place. It was supposed for this system to take three months for development, and it did.

4.1.2 Project Purpose:

The purpose of this project is that no one executes orders from switches directly; all must be done through the system automatically unless

something needs approval. The system has an administration board, for the technical data related into switches, boards, free boards, technical roles, reporting –for revenue assurance purposes, like comparisons among CRM, the third party and switches in order to remove differences that might appear and ensure that no more problems emerge

4.2 Interviews:

There are two main parts to analyze in research based on the data sources the researcher used to gather his information. The first part is the interviews that the researcher made with a lot of firms and is covered in this section, while the second part is the case study part. The interviews had been made with the interviewees who replied to the researcher after contacting them either by phone call or using the emails. After reading many books and sources about how to conduct an interview and how to deal with interviewees, about how to construct the questions and how to deliver them, the researcher started to construct the questions and contact the firms in the population by sending the emails or making the phone calls. The researcher was either contacting the interviewees individually or sending requests to the HR department in each firm asking for permission to do the interviews and asking them to nominate some persons to do the interviews with them based on conditions stated in the methodology, and then the researcher tried to choose the most suitable one for him to make the interview with him. All the interviews were made inside isolated rooms inside the firms, and the researcher had taken the notes for each interview,

although all of the interviews had been recorded by the researcher in order to be drafted and reviewed again by him and all of the notes were rewritten neatly. The interviews were semi-structured interviews, and there were some basic questions for each interview, yet the interviewer were asking more questions based in the context of the interview with each interviewee; those questions were necessarily on the sight of the objectives of the research. During each interview, if the interviewee didn't understand the question or he couldn't find an answer for a certain question, the researcher had to rephrase the question in more understandable way or to give examples explaining what he means by his question; the questions were being asked with a quite caution not to be giving answers to the interviewee. The questions asked in general on the interviews were about the subject of the research and covering the research objectives and questions, and they were intensive since the researcher used them in addition to the research literature alone with the agile Scrum handbook and other resources; he used them as a knowledge base and guidelines for his research and for the case study implemented in the local firm that the researcher observed its first implementation of agile Scrum instead of the traditional methodologies used inside it.

After the interviews, the notes were combined and summarized for each interview. In order to increase the validity, all interviewees were given the opportunity to read a summary of their own interview to help minimize misunderstandings and give the interviewees a chance to adjust their

statements. If requested the notes were corrected and the updated versions used for the summary instead. However, only a few of the interviewees actually had any comments on the notes other than an agreement.

After that, the researcher made a justification for each question in compliance to the objectives and questions of the research; each question been asked in the interviews were justified within the sight of the research questions and the objectives.

Then the interviews were neatly analyzed in order to gain as much data and knowledge as possible in order to have the necessary indications for the research about the local market in Palestine and for the case study to be done as part of the project. The justification was a necessary part since it shows why the questions were asked and how they benefit the research. The interviews questions and justifications are sectioned in the questions appendix.

Many questions had been asked in the interviews, those questions are related to how the current IT firms implement their projects and what are their problems during, during and after a project implementation, to how the management behaves towards its developers, why they are having problems, why they didn't transfer into other ways of work, and so on. Most of the firms indicated that they still working on the same old way of development in their SW project within the sight of the so-called "traditional methodologies", yet there are few who tried to change their

way of work although not all who did so was able to success. More details about questions and answers are stated on the findings and results chapter.

4.3 Case Study:

Before the project case study started, the researcher made two sessions for the Scrum team formed by the local firm to implement a new project based on Scrum instead of the traditional methodologies; one theoretical session and one practical session. The theoretical session was held to transfer the knowledge that the researcher gained from the different scientific sources and some practical tips got from the interviews into the Scrum team formed by the local firm. The practical session was about trialing fake trainings to simulate Scrum starting by short meetings in which the developers tried to shorten a list of tasks to be used as a backlog, then to simulate the Scrum roles and discussions, and to explain any features that are not clear to the team. As mentioned before, the training was done by the researcher and assistance of two other members depending on the different scientific materials about Scrum, the Scrum handbook besides the interviews knowledge that also was used during the development. After the training sessions were held, the work on the project started with short period and the team divided its members and managed by the team itself, and the team members kept on sending reports periodically about their progress, problems and queries if any to the researcher who gathered their info and itemized them into sub addresses in the research.

4.3.1 Scrum roles:

The expected roles from the Scrum literature are all used: the Scrum Team, the Scrum Master, and the Product Owner. The Project Manager was already assigned by the firm to manage the project.

4.3.1.1 The Scrum Team:

A Scrum Team is comprised of employees from different fields of expertise, such as testing, hardware, platform, developers and software engineers. It is a well known fact that the personality of the individuals in the team affects the well-being in the group. One member of the team is appointed Scrum Master and had been given the associated responsibilities in addition to his normal work. There was a problem that certain members of the team members already have other responsibilities like supporting customers and cooperating external development team, leaving less time to work towards the actual Sprint goal. This makes it hard for the team to estimate the number of items to commit to during a Sprint, but not a high degree and the problem was acceptable.

There are also other related problems, such as when a team member promises an external customers –other than the project customer- to support them for a shift, but the Product Owner does not prioritize the task high enough to be at the top of the Product Backlog, then the member now have to choose between picking a low priority item instead of a high priority item, or breaking the promise to the external part. On the other

hand, some members of the team appeared as fundamentalists concerning Scrum and refuse to perform tasks that do not come from the Backlog, which lead to frustration for Project Managers for a certain time. This researcher itemized the former problem as non-enough understanding of agile Scrum.

It has been decided to have a team dedicated to respond to urgent tasks only; this way other team members would not be interrupted during their Sprint. And after the project deployed, a maintenance team was set to support the already released products that may use Scrum.

A few team members admit that they could follow the Scrum framework more strictly, even though they tried their best to do so. Generally, the Scrum team members were satisfied with the new working process, which is also reflected in the expressions and reflections during and after the project is done.

The team had at least one tester responsible for making sure that the items and features are completed concerning their functionality. One risk was noted with having this tester as a part of the team is that the feature that is not completed may be marked as done because the tester feels the pressure from the rest of the Scrum team to do so, which dictated on the Scrum master sometimes to make other team members to make sure that the feature is really done.

4.3.1.2 The Scrum Master:

Each Scrum team must have a Scrum Master (SM). The SM was one of the employees that already heard of agile Scrum and had a background about it. Besides, he was given some proper documentation and discussed the subject with the researcher more intensively to make sure that all issues are understood correctly. One interviewee argued that the SM role demands hard work and big effort to be used correctly, and that more effort should be put into this. More problem solving and more coaching were also needed from the Scrum master; for our case, the Scrum master was the team leader (technical leader) who worked a facilitator, coordinator and as a developer. The researcher benefited the interviews he made to gain some necessary knowledge to avoid many problems other had been fallen to, and to use the maximum usefulness of some followed procedures in other firms even if not agile procedures. That knowledge had been also shared with the Scrum Team in addition to the articles provided about the subject.

4.3.1.3 The Product Owner:

The Product Owner (PO) in the team supervised and updated the Product Backlog; he was prioritizing the items in the Product Backlog and asking the Scrum team to estimate the amount of time it will take to complete a task. The main important thing the PO had to do is to organize the features and tasks to be inserted into the back log. The PO can be represented by one of the team who understands the customer's needs the

most or by the customer himself if he was dedicated to the project. The PO was telling his needs and requirements with as a story; means that each required feature had being told to the developers and scrum master as a user story, and each user story is assigned an address to use through the backlog. Requirements were broken up into short user stories, which were written on small cards, and each developers was either choosing (volunteered) or being assigned to one or more of the story cards to work on, where an estimated and actual time spent is also tracked on the story card. It is noticed that the introduction of the story cards is increasing the group autonomy with story cards being selected and prioritized by the whole team during their planning meetings.

The PO also was answering questions from the Scrum teams about priorities, while the questions that were more detailed had been handled by the Project Manager or other stakeholders. The PO did not always have a complete view of the item when they are created, instead, he can receive input from the customer, the sales department, the project manager, and the Scrum team. One of the interviewees said that there are problems with the team members to cooperate with the PO; it can sometimes be hard for the PO to verify whether an item has full functionality or not on the demonstration. Another problem is the confusing fact that there are both stakeholders and the PO, and the developers do not know whom to go to. Usually, the stakeholders are in most cases a Project Manager. The PO

participates in the Retrospective meeting with the Scrum team to help them evaluate the performed work in the Sprint.

4.3.1.4 Project Manager:

The role as Project Manager (PM) originates from the Project model that was in use in the company and not from Scrum, which lead to confusion about where the PM fits into the Scrum framework. It is also sometimes hard for a PM to get questions answered from the PO and the Scrum members. One a future research, suggestions to improve the implementation of Scrum can be studied to avoid this problem. The firm can let the PO to be responsible for team with a Backlog.

4.3.2 Practices in Scrum:

The team has tried his best to follow Scrum as it is followed in Scrum guide and as followed as semi-Scrum in some of the other local firms in the territory.

4.3.2.1 Sprint:

The sprint was a two-week period, during which development activities are performed. The duration had been chosen to balance between responsiveness to changes and current development of the work. Each sprint starts with a sprint planning meeting and ends with a Sprint Review. Retrospectives are usually held after each sprint is finished. Some

interviewees mentioned that they sometimes have to use the retrospective day to finish tasks that remain from the previous sprint.

4.3.2.2 Sprint Planning Meeting:

The sprint planning meeting marks the start of a sprint when all the Scrum team members, PO and other stakeholders meet during approximately two - four hours. It usually opens with information from management and the PO regarding the latest development, customer feedback, and plans for the future. The meeting was usually held in a meeting room to ensure the enough space for the attendants.

All items from the Backlog are printed on paper notes that have been sorted according to their priority and attached to a large whiteboard in the meeting room.

Included in the notes is a short description of the item, its priority and its estimated amount of work needed (based on the description). A rough presentation of the top-priority items is usually done, but the PO was available to provide a more detailed description if any team should need it.

Since the Backlog is common to all the teams, any team may in theory pick any item voluntarily, as long as the most important items will be picked. The team members usually know what the top priority items will be prior to the sprint planning meeting and have discussed which ones they are interested in to take. The volunteer choices will be based on his

knowledge about the technical aspects of the tasks. The team members also would know the speed that they will have to consume through the previous knowledge they have gained during previous sprints and use that as a guideline for their commitment, but still it might be hard for the members to know how much work he will be able to commit to, since he might be occupied with other work and travel arrangements during the sprint. The team members in general were avoiding large items that are not time boxed as they include too much uncertainty about the time estimation; only the ones they worked on such items before might pick them or might be assigned to them by the Scrum Master.

For two months, each two weeks the team was holding the sprint planning meeting where the team selects the requirements and features that are going to be listed on during this sprint. Each sprint is two weeks long; where the team selects the features and assign them to the SCRUM members; each one of the members selects his features to work on during the next two weeks; some features are assigned automatically for certain ones because they all know those features or those features might be sequenced or depends on previous features that were also done by the same members. In the same meeting, the team also discusses each feature and their usage in the project. The estimation is also done in this meeting using the Poker estimation process using the cards like Poker players. Using this estimation process, each task must be weighed and estimated to distribute the load among all the members.

A distinct thing for the team members when they were estimating the features was that they used the estimation points for each task in the list of features. The team members estimate each task by the experience of every one related to the task in the team members, so if the team has a feature, the team members estimate its worth-of-work by points not by days; the team followed the standard of having a total load of 150 points per sprint (depending on the technology used in the project, this project used .Net technology) for the suggested tasks, noticing that the sprint days are 14 after deducting the holidays, and if one of the members needs a vacation, he must have planned for the vacation before the sprint (the 2-weeks) and for such reasons it was preferred to use the points-based in the sprints. So, in brief, the team's work in the planning meeting is to determine how many points a member needs to accomplish this task, and this process is to be done using the Poker Estimation.

From the interviews that the researcher made, one interviewee said that when they do the Poker Estimation, they proceed in it as follows, and the team also did the same in his trial to transfer into agile Scrum:

- A Moderator: who usually doesn't play, only organizes the meeting.
- The Product Owner who provides a short overview. The team discusses the situation and project through asking questions and queries to clarify assumptions and risks. The Project Manager then summarizes the discussion.

- Each member must lay a card face down to represent the estimate he puts. The member can use days, hours or story points as an estimate, and after he uncovers his card, the members interested in this feature discuss the time without mentioning numbers in relation to feature size in relation to feature size to avoid affecting each others' estimation without even beginning the discussion.
- Everyone calls their cards simultaneously by turning them over.
- People with high estimates and low estimates are given the chance to offer their justification for their estimate and then discussion continues until a consensus about the feature's estimation is reached.
- Repeat the estimation process until a consensus is reached.

The longer the estimate means the more uncertainty is there, so if a developer wants to play a 6 he is forced to reconsider and either work through that some of the perceived uncertainty does not exist and play a 5, or accept a conservative estimate accounting for the uncertainty and play an 8.

When the team was holding the Scrum planning meeting, each one of the members must have a card deck that contains cards with numbers; each task can be assigned a number of points that appears on the card. Each feature is assessed by the members who most knowledgeable about the task even if the whole team assessed it. Each member has his own deck of

cards, and each card has a number that is matched with a Fibonacci number; he raises the card with the number on it to indicate his estimation over the task, and he must keep the number on the card hidden from the other members so that he guarantee the independency of his estimation from them; this step must be done by all members, and when all members finish their estimation, they reveal their cards to each other per task. If the estimation points from the different members for the same feature have the same or close value, then its estimation is correct and taken, while if the value are different for the different members, then they have to convince each other, lest, the average among all card points can be taken for that feature. This step (the consensus for one feature) is demanded because SCRUM is the responsibility of all members, not of particular one even if the task or feature is to be done by only one.

4.3.2.3 Daily Scrum:

In the first day of sprint, the team was holding the sprint planning, and from the next day, the members started their working on the tasks assigned (voluntarily) to them, and for every day the team was holding the stand up (SCRUM) meeting that takes from 10-15 minutes in which every member answers three questions: what did I work yesterday, what we are going to work for today and what problems we are facing; if I had a problem I must brief it in this meeting so that everyone else has to be aware of it in case of repetition or to get some feedback from others. The team tried its best to follow SCRUM, so at the second week of sprint (for 2-

weeks sprint) the team was making the SCRUM refinement meeting (grooming meeting) in which it prepares for the next sprint, like the requirements, problems, features, recommendations, etc… and a preliminary distribution of the tasks over the members, so that when the next sprint comes, the team wouldn't be bothered a lot with the features and roles of members and tasks.

4.3.2.4 Demonstration:

At the end of each Sprint, a Demo is held. All the team members, the PO, stakeholders and other interested parties gather in the same place as during the sprint planning meeting. The teams then take turns in presenting their items marked as done. Some interviewees indicated that items are demonstrated even when they are not completely done, which might result embarrassing moments or broken functionality. Features that do not involve any directly visible functionality, such as bug fixes, are also hard to demonstrate. Even if the functionality has been implemented as required, the test code showing the changes might not be written until a later Sprint.

4.3.2.5 Retrospective:

At the end of each sprint, the team members meet to share their experiences and opinions from the last Sprint. The sprint planning meeting and the retrospective meeting are held together when discussing those details. Meant by that, the team determines the tasks, assigns them and estimates an expected points per task, the team gathers the lessons learned

from previous sprints, improvements' suggestions, answer the why(s) that will appear during our sprints i.e. why did we estimated 5 points for this task although it took us 15 points? Or why did we only finish 120 points in a whole sprint from 150 points? (Each sprint must maintain a 150-points standard). Now as a PO (sometimes one of the team members take the role of the product owner on behalf of the client), he must explain the task in details, what is expected to do, its nature and effects on the system, and on their turn, the members who are supposed to work on this feature has to analyze the functionality, validation, expected procedures, and as a final act comes the estimation. It is needed to notice that the team followed a rule in which if the task consumes more than 15 points, then it must have to be broken down by the team members into smaller tasks; this is only a recommendation, and not an obligation. At the end of each sprint, the team must deliver a working release if possible, or if not possible, then something touchable and can be felt, meant by that; I have to deliver a complete (done) feature. The feature can't be agreed to be done unless the following conditions were achieved:

1- Coded completely.
2- The feature must be unit tested.
3- Reviewed and test by the QA team.
4- Committed on a certain configuration management i.e. CVS (concurrent versioning system) that serves as a store for the project and can be reused whenever needed.

If a task wasn't finished as planned, then the team members have to compromise it with the customer and take the lessons learned to prevent it from happening again, and prevent the causes of such action, beside more testing and QA to the task if needed.

4.3.2.6 Working Environment:

The team members have their own separate room with whiteboard and their workstations. The usage of whiteboard varied between using it for planning and discussion.

The team was deeply working in pairs, seniors with juniors, experts with fewer experts; the ones with the more knowledge about a feature transfer his knowledge and experience the other one; and it proved to be very useful and the progress of the team was better in whole. It is most likely that the process of team integration depends mainly on the pair programming, especially that the tasks were being distributed voluntarily over the team members, and the PP distribution could be done by the Scrum master or by the members themselves. Moreover, many of the interviews stated that the PP was a prime insurance to gain in a SW project to ensure a steady rate of progress throughout the project, not to mention that agile Scrum considers the PP to be one of the main features to follow it. It is important to notice that the while team holds the responsibility of the success of failure within SCRUM; all responsibility are shared among all members, and if any member found any problem and couldn't solve it,

then the whole team is responsible to be with him and solve his problems. If one of the members was fresh and doesn't have a lot of knowledge and experience, then one of the members of the team (usually the experienced members) sits with him and discusses the task(s) that was assigned to the fresh one. Besides, one of the recommendations of SCRUM when any member is having a problem is to notify the other members so that he'll be able to save time and progress more rapidly; it is all for the sole purpose of finishing the sprint successfully which is the shared goal among all members. The commitment that must be presented during a sprint among the team members is the commitment toward finishing the sprint successfully and delivering the expected features without failures, since if no release for that sprint, then the whole team would have a problem with the customer even before we say that a problem between the team and the project manager; the end of each sprint represent a deadline for our release, and if the team didn't meet the deadline, then the customer would have his points against the team unless the team had an extraordinary situation that prevents it from going live the release. For each SCRUM, what the team cares about is how much still left, not how much finished, since the main concern is to finish the sprint within the allowed 2 weeks.

Agile teams value individuals and interactions over processes and tools[1]; however, agile teams still seek for tools since they know the importance and need for good tools to support developers in their tasks.

[1] Agile Manifesto http://agilemanifesto.org

Examples of tools that directly support agile practices are automated testing frameworks such as JUnit [2] or test coverage tools such as NCover[3]. Many tools are currently available to assist developers with their regular tasks; many more tools appear constantly, trying to better address developers' needs or to improve their effectiveness by integrating multiple functionalities or offering various services (Gross, Melideo and Sillitti, 2005). Regarding this point, it was one of the problems faced in team for applying Scrum is the task progress. The team were using Google docs tool to monitor the progress of the sprint; as stated previously, the team was meant by what is left not what has been done, so this consideration must be reviewed and monitored through a tool (Google docs) that was initially used; it involves all the team members and obligates the daily contact with the tool. Each team members has to log into this tool and insert his progress points. Now the problem that appeared afterwards is that not all members insert their progresses, or at least not on daily basis, which causes the SCRUM master and the project manager to have difficulties dealing with the tasks progress and with the sprint as a result. The causes of such a problem were mainly end users (members) problems; like Oblivion and busyness, or wrong estimation, mistaking with the inserted actual time. The team was then moved into another tool from JIRA; it was the JIRA agile tool (Atlassian JIRA: Green Hopper) which presents a good presentation of the project tracking specially for agile projects. One of the problems that

[2] JUnit http://www.junit.org
[3] NCover http://www.ncover.com

the team faced when the members were working on JIRA agile tool is that when they create a task that contains many subtasks, you can monitor and manage task by task, but the problem is that when you finish the subtasks, and close them, the main task just doesn't close; it is expected that the main task to automatically finalized when the team finishes its subtasks but this doesn't happen, which causes some confusion to the SCRUM master and the project manager. Although it has some problems with the sprint management, the team wished for this tool to be more flexible with dealing with sprints and SCRUM members.

A lot of time wasted on the non-coding related activities during the development of the project, which can be considered normal, since according to (LaToza, Venolia and DeLine, 2006), the developers spend large shares (up to 50%) of their time on non-coding related activities, however, there is not a clear consensus over the shares of various non-coding activities.

Agile software development (ASD) insists on the customer taking control and being constantly involved and stresses collaborative partnership based on daily interaction between developers and the customer (Highsmith, 2002). The project manager followed the test acceptance to be that between two iterations there is always a test phase, which is a post activity of the preceding iteration and there is a concept phase, which is the preceding activity of the next phase. The project manager confirmed that

the acceptance tests were run by the customer, meaning that the customer had the responsibility and decision power in these tests.

4.3.2.7 Self-organizing:

Some of the interviewees indicated many opinions about the extent the Scrum team can be self-organized. Some of them argued that the agreed commitment to items helps the teams to decline additional work during a sprint. They explained that by arguing that this only works in theory since the Project Manager bypasses the Backlog and directly controls the team during the Sprint when it comes to tasks has to be done outside of Scrum. This is because of the fact that the importance of a feature might be different from the Product Owner, priority of the backlog and the promising with some functionality to the customer. The alternative would be to leave the external part –out of scope- waiting, which is considered worse than the overhead incurred by the switching of task prioritization. They also indicated that the support tasks are also hard to estimate with a time and plan and may appear at any time during a sprint, which may cause some malfunction of the idea of self-organizing.

In relating to the team autonomy, according to the interviewees and the observations from the project in the local firm, it is noticed that the Scrum master did not manage to protect all the team the whole time, which in turns might decreases the level of autonomy of the team. The Scrum

master was noticed to be talking to so many customers, which might changes his perspective constantly.

On the other hand, it is also noticed that everyone in the team was involved when making decisions, and that the ones who have to do the work are most involved when making decisions.

4.3.3 Artifacts Used in Scrum:

In the process of imitating the Scrum handbook and literature, the backlog was divided into three different Backlogs; the Product Backlog, Sprint Backlogs and Release Backlog; Burn-down chart was also used.

4.3.3.1 Backlog:

The product backlog is kept up to date by the PO with cooperation of the Scrum master and it was written and updated with the JIRA Atlasian tracking tool, and other employees have their access on the project tools. The PO prioritized the product backlog together with Scrum master with the PO as an owner, and ranges it thereafter. The employees generally believed that the product backlog is one of the biggest benefits when using Scrum; since when prioritizing a task or a feature, the developers do not have to make the prioritization themselves, which gives the developers a better understanding of what is important and not, and if a little explain is accompanied with the items, then PO would not have to answer so many questions. When a feature has not been completed during a sprint, the PO

marks it as “started but not completed” which means that this item wasn’t paid much effort as required to complete it as the estimate indicates. The product backlog can give many advantages such as the increased knowledge distribution within the company. Not all things that are being done are added to the product backlog, like the mentioned urgent customer support that is not always registered, which is usually followed by the re-prioritization or switching priority of these unplanned tasks, instead of the PO doing it through the backlog. The sprint backlog consisted of all the items that the team has already committed to do during the current sprint; they were usually displayed on the whiteboard in the team room.

4.3.3.2 Burn-down Chart:

The Burn-down chart is drawn on the whiteboard in the team room during the sprint. All employees can have the look and monitor the progress of the Scrum team. During the Sprint, the Burn-down chart is made available on the Atlassian JIRA tracking system.

Chapter Five

Findings and Recommendations

Chapter Five

Findings and Recommendations

5.1 Interviews Discussion: The Traditional SW Methodology in Palestine:

Many local IT firms tried to transfer their way of work from the current traditional methodology of SW development into newer versions, yet most of those trials weren't successful because of many reasons. Many interviewees indicated that the management in their firms didn't give the full commitment to newer SW methodologies either because the idea of the change proposed by some of its employees were convincing enough, or the management found that any transfer into newer methodologies other than the known way of work is to be considered as a threat to their business and market.

Other interviewees indicated that their customers didn't agree with the new process and were dissatisfied about it because they didn't want to hold any responsibility more than what they hold at the moment; they thought that being involved in the process would only increase their responsibility and cost them more time and efforts than the current situation; this is surely because of the miscommunications between the firm and the customer which lead to the misunderstanding of the customer involvement in the process.

While three of the interviewees claimed that they have the necessary knowledge about new SW development methodologies like agile, others said that there are still no clearance about the value of those methodologies knowing that the traditional ones are still living on decades since the past century which forces them to question themselves of why having to follow something new that yet not proved to be of a better value. Many interviewees also mentioned that there is no need to the use of something new in their methodology of development since the customers are already satisfied with what they are working till now although the percentage of success rate among the SW projects are significantly low in comparison to other projects types. The fear of historic failures prevents a lot of firms to move into something new in their development way-of-work; many firms had a lot of trouble ensuring the stability in their development ways, and many of them reach the level of stability where they look for, which enforce a mentality that thinking in disturbing themselves with something new other than their way-longed stability will only move matters ups and down and make things worst at least for short periods. Many interviewees said that their management and customers are satisfied with the current way of work and production, yet no body of the interviewees indicated that all – or at least most- of their projects went successfully within the ranges of success for IT projects mentioned in the literature. Many interviewees also indicated that the traditional way of development can't be simply dispensed; since even if new methodologies have emerged, still the old usual way is easier to understand according to them.

Four of the interviewees mentioned that their firms still working on the traditional ways of SW development, they also mentioned a lot of the problems and shortcomings emerge from using those methods and wished they can get rid of them. The most dominating one is the miscommunications between the firms represented by the management, architects and developers from one side with the customers on the other side; starting from the requirements gathering phase until the deployment phase. Another shortcoming mentioned by the interviewees is the acceptance of the changes requested by the customers; the firms usually have many problems regarding the change including the time to change, its effects and cost. Besides, some of the interviewees said that they are not comfortable with the environment of the work they do in the traditional way; they prefer to work together with other developers and like to have more privileges than the ones they already had.

Most of the IT companies in Palestine are most likely using the traditional ways of SW development, said one of the interviewees, yet their projects are not above the mentioned success average, and the customers are not really satisfied with what they have, but still only very few companies followed something other than the traditional methodologies, and to be specific, agile Scrum. Those few interviewee from the firms who already adopted Scrum, indicated that their success rate was higher, they didn’t find a lot of complains from the customers about system bugs or functionalities, they reported that their communications with the customers

are better, they were satisfied with the privileges and responsibilities they have and also indicated that their management are –almost- fully committed to the new methodology. They also said that they were able to accept changes during the project easily after they adapted the new methodology which made the changes easier for them. They, in general, preferred the new SW development methodologies represented by agile Scrum over the traditional ways, even they indicated that it became harder for them to abandon agile into traditional again.

5.2 Case Study Discussion: Scrum at The Local Firm:

There had been a lot of trials made by firms from inside Palestine to transfer from an old SW methodology into newer ones like agile, and this research in a local firm is one of those trials. The market demands in Palestine dictate the intention of giving faster response upon the local firms to meet those demands, faster than the traditional methods of SW development, while giving developers the necessary tools to increase their performance. The below points are some of the discussion points related to the subject:

- **Communications with customers:** Through the continuous dialogue and communications between developers and customers, facilitated by the PO, Scrum compresses the wants and needs of the customer into requirements that can be converted into functionality, along with the continuous feedback from customers through the PO,

which also helps keeping the product requirements visible, updated, and well adapted to the market needs.

- **Customer involvement:** The increased involvement of the customers into the project given by Scrum is likely to give more responsibilities to the team members to decide how things should be done in practice. It was noticed that it increases the overall quality of the work. This workflow is supported by Agile Manifesto, where collaboration, interaction and working software are identified as more important than documentation, contracts and processes. The customer is put in the center of Scrum process through the facilitation of the PO who in some cases can be the customer itself. The PO usually acts as the link between the development team and the customer, and constantly works to ensure that the requirements are met in the product developed by the team. Many interviewees indicated that short-term sprints are better to make the decisions making as late as possible; these decisions should be made by the most knowledgeable persons about the situation as much as possible, for example, the Scrum team is using the knowledge it gained about its own performance and progress to make estimations to the work and planning about what to do. Scrum is not a defined set of processes and steps that need to be followed to achieve it; it is a framework with guidelines that are necessary to walk within their sight, means that the framework implemented here may appear as a

number of interconnected processes, that both the customer and supplier inputs are highly valued in the framework.

- **Quality improvements**: Quality improvements are a key component of the Scrum framework through the continuous testing and adjustments of the methods in use. The high level of communication between team members at all times is also crucial in this improvement process. At the end of every sprint, the evaluation done in the sprint retrospective also forms a structure for the improvements of the approach.
- **Employee involvement:** One of the interviewees said that the conditions for employees involvement is created in Scrum through self managed teams, focusing on the individuals, and adjustments to the work based on feedback, this will help increased motivation and improve the results from the employees.
- **Dedicated leadership:** In addition to that, although dedicated leadership is something that is not in the core of the Scrum framework, yet the inspiring and motivation of the co-workers is however very present in Scrum. It was noticed that the team expressed that everyone in the team was involved when making decisions; the ones who have to do the work are most involved when making decisions.
- **Professional coaching:** As deducted from the interviews the researcher had made, professional coaching is being defined as

helping people use the knowledge they already have. For a new team to enter into the field of SCRUM agile, the main problem is the misunderstanding of the process; SCRUM seems easy on the paper, but it is difficult to follow and apply since it concerns all the team in terms of communications, leadership, responsibility, coordination with the customer, and not just one single member. So the first step to move into agile is coaching the team for the principles; which had been done by the researcher with the team of the case study. The researcher gained a lot of knowledge about agile and agile Scrum through the huge amount of articles in the literature that he read talking about the same subject plus interviews he made. In addition to that, many of the team members have already heard of agile and its concepts; what the researcher did was only to put all the stories part together to form one complete story about the Scrum agile after taking the required approval from the firm management and confirmed their commitment toward this transfer in the project.

5.3 Differences between the Scrum framework and Scrum in Palestine:

When looking at how Scrum is being applied in Palestine and comparing it with how Scrum is explained in the literature, the reader will find a few differences, which is a natural consequence of using a framework in such a complex reality, where a lot of factors play in the scene, and business value and uncertain market have to be considered.

5.3.1 Common Backlog:

One of the most obvious features in the use of Scrum in the literature and the one applied in Palestine in general is the common backlog for the team. However, the literature does not explicitly specify how the Backlog should be handled when working on such a very uncertain environment like Palestine. The researcher couldn't find any reason why this cannot be done with a common Backlog, as long as the PO is fulfilling the responsibilities associated with this role.

5.3.2 Product Owner:

The Scrum literature indicates that the PO is the link between the developers and the customers, although that more than one PO can exit to break down the huge user stories and share the big quantum of the project information in addition to discuss the different questions from the Scrum team. The case in this research was only for one small project, but if the case was for a huge project, or for instance for multi-project, then the results definitely will be different and the indicators will refer to something else, since such a case needs a team of POs that holds the meetings with long discussions about the approaches to follow. In such cases it is very important to make sure that the POs take the full advantage of the roles of the team members are to ensure their roles are well defined and obvious to the rest of the organization.

5.3.3 Sprint Planning:

In the Scrum literature, at the start of a sprint planning meeting, the top priority items should be explained in details by the PO. In the meetings applied within this research as a reflection of the Scrum in Palestine, the same as literature is being done, but with the addition that after the planning meeting, the estimation using the PE and the writing of the user stories had been introduced so that the team knows the expectations for each item. Also the PO was available to clarify and answer questions about the items. The items are also available for inspection the day before the start of a sprint and at the estimation-planning meeting, which gives more time for the Scrum team to discuss and ask for clarifications.

5.3.4 During the Sprint:

The interviews and the case study show that not all firms in Palestine that are trying to follow agile Scrum are following the Scrum framework strictly at all times; like not every team has daily Scrum every day, something that contradicts Scrum literature and may cause a decrease in the exchange of information within the Scrum team. Skipping the daily Scrum will probably not have a major impact on the project, but it may lead to negative results, therefore, it is recommended by Scrum handbook to always have daily Scrum meetings.

During the daily Scrum meetings, it is rarely to have answers about the question related to the obstacles and problems faced by the team.

Stating the problems faced during the sprint is one of the main questions mentioned earlier during the literature, but the fact is different; most of the team members do not update their burn-down chart at the Scrum board every day, something that the Scrum framework suggests. The burn-down chart is used to visualize the project advance and give the team a hint of the remaining work for a certain sprint. Hence, this should not be a problem; especially that Scrum handbook recommends following Scrum as strict as possible, this is something the Scrum master can assure among the team.

A lot work was being done out of Scrum scope for different reasons and never added to the backlog at all. Tasks as supporting customers on the spot or work that the project managers need to be done immediately outside the sprint commitments, such work help in keeping things running smoothly from the management perspective, but it may also contradict the intentions of Scrum sprint by increasing the pressure on team members to choose between Sprint items and other work. This may also affect negatively the level of traceability when work is done without the specified requirements and verification. This needs to be addressed if the quality could be affected negatively, while keeping in mind that it is not possible to add every single task in the organization to the backlog. This is also the responsibility of the Scrum master to guard the team from tasks that are not originating from the team's sprint backlog.

5.3.5 Roles and Responsibilities:

The Scrum master in the research didn't put as much effort in being a Scrum Master as the Scrum framework recommends, this is also mentioned during the interviews. It is not demanded from the Scrum master to be fully involved in the programming and development activities, yet levels of Scrum master shortening had been found during the case study, like he wasn't being able to support and cover for the team when the project manager wants to assign tasks out of scope to one of the members, or when it is related to the customer collaboration, the Scrum master wasn't expert enough to handle such an issue; this can be – in general -attributed into the low knowledge that the team have about agile Scrum.

The testers had been involved directly into the development, which is what recommended in the literature. It is better for the testers to be part of the Scrum team since that may help tasks pass the test rapidly especially that the testers know exactly what the definition of the "done" is, which results in a more progress and higher velocity for their own team . Yet, this may cause the code to be less quality and ends with a less good product. This can be overcome by specifying exactly what the definition of the "done" to be adopted is.

5.3.6 At The End of The Sprint:

When a Sprint has come to its end, a demonstration is held where all members demonstrate their results and experiences from the sprint; there

was no specific time, but it was preferred for the meeting to be from 2 to 4 hours. The literature recommends the demonstration of an item after the definition of “done” is totally met by the team for this item, but within this research, this demonstration was done based on the progress of the item and project; this obviously contradicted the Scrum recommendation, but suited best for the team in this case.

Usually to make sure that the definition of done has been met for an item, the PO should take time and discusses it with the Scrum team before it stamped as being approved, and then the demonstration would be held, and that what was happening here.

When a sprint is finished, one of the team members writes a sprint report. It includes, for example, the sprint goal, the description of the work done during the sprint, and the most important changes from the retrospective. A sprint report increases traceability and facilitates the Scrum improvement process. This is also a good way to document experiences from the sprint. On the other hand, after the item is demonstrated, the PO might think that this item is completely done, and also for the developers whether individuals or team, might believe the item is done and everything is working fine while it is still not even if a small percentage of the work is not done, or they might easily think that it can be fixed later, but if the problem was found after the item had been demonstrated and marked as done, it was inconvenient to tell the PO that it is no longer done and put it back in the loop with the product backlog,

which forces the team trying to fix it or completing it outside the sprint time box, or just ignoring it although it is not completed. As a result, the requirements wouldn't be traced and verified; this could be solved from the start by giving a high priority to the definition of the done and not demonstrating it unless the task had been check, verified and approved by the PO.

The retrospective is held to evaluate the sprint and to summarize the experiences earned from the sprint. Without the retrospective being held, a lot of important experiences will not be gained and discussed, and the same problems may have to be solved again later.

5.3.7 Customer Collaboration and Release Management:

Scrum recommends a high level of customer collaboration, which means a high interactivity structure between the team and the customer must be established, like letting the customer to participate in the sprint review is a good practice to follow. In Palestine, the customers usually is not being involved from the primary phases of the project in the traditional way, which is deducted from the interviews, but it is also noticed that their involvement is crucial for the project development, which is what agile and Scrum came to solve. A higher level of customer collaboration leads to customer satisfaction, and this was obviously seen during the project.

Scrum demands releases after each sprint; having releases after every sprint facilitates customer feedback since the customer has the possibility

to give feedback and see the result of the feedback already after next sprint. Besides, this gives a quick feedback for the developers themselves about how things are going to be when deployment.

The direct participation of the customer in this research was represented by the onsite customers who themselves worked with the future system to a certain extent during the development phase, they were also intermediaries for other users like operative staff in the customers' divisions; these employees indirectly participated when they commented during presentations or when they provided their viewpoints or descriptions of their work processes to the onsite customers. They have however also participated directly when they tested the results of the iterations or when they observed the developers, who were in contact with them directly; there were two staff was assigned by the ISP to follow these tasks, but they weren't considered as the PO since the PO assigned was one of the team. The PO was one of the team because this project is a new trial to follow not like the traditional old way of development where all roles and responsibilities are known.

Since this is a new trial, the team and management preferred the whole Scrum roles to be from within the organization, while the team can welcome the introduction of the ISP's staff into it with specific involvements; although it was a new thing to do with Scrum, but it didn't exist at first in the traditional way, which caused the customer to feel satisfied into an extent even before the project was started.

If the team–for any reason-didn't finish the sprint, then they have to explain why to the customer and management; and since they were new to the subject, most of the time they weren't able to deliver the sprint successfully to the customer; not all the tasks for the most of sprints had been finished during its sprint. But since what the customer really care for are the results, he didn't much complain about the project's progress, which can be considered as one of the benefits of agile. In fact, the team didn't have to lie to the customer just to show him that the project is progressing or to convince him about the teams' work and ideas, not to mention that if the team had any bug or problem during development, they didn't hesitate to show it to the customer since they are working on agile. But of course the team was only committing the parts that are working even if some parts had some bugs that needed fixes. Convincing the customer wasn't a big issue since it implies letting him involved in the process from the start till the end. The customer (the ISP) in general, had two of the employees to follow the project from their side and they were in consistent contact with the team; through the Scrum master into the team members directly for a certain feature, and through the project manager to the total progress of the project. Meanwhile, they were sending all the information into their business and technical side to make sure that they are on the right path of the project progress from their side. At the end of each sprint, the customers users at the site come and check the new features added through a demo presented by one of the developers, and the users return with their feedback or comments either momentarily or after taking some time upon

agreement with the team, during this time they take their notes into consideration and send their feedback also to their divisions inside the ISP, and even they might let someone from inside the ISP to come and check over the new features added that sprint.

5.3.8 Contract Management:

In the traditional way of SW development, many interviewees talked about the contract management. It is indicated that sometimes limitations are imposed by customers, like contracts or technology; they just want to give the scope and requirements, and expect the organization to deliver the development exactly as planned, or perhaps they look for a fixed price contract. The problem is that they ask for a fixed price along with a fixed deadline and a fixed scope, and when –mostly- the project fails to meet those limitations, most of cases legal conflicts are being emerged in front of courts and a lost confidence in the IT industry appears in the market.

One of the interviewees mentioned that agile refuses to go with a fixed price projects; agile talks about embracing change, since fixed price projects with changes will come. Yet in Palestine, it is still early for agile practitioners to introduce a new way of contracting into the market, since this same market isn't responding to the fast-change SW world.

Many ways to introduce the new ways of contracting that are attached with agile methodologies with the local market of Palestine. Agile contracting comprehends the fast tracking changes of the market so it gave a lot of

ways to keep up with its new ways being introduced. One of the ways that can be followed with customers is by allowing the customers to use agile on a trial basis, agile practitioners are able to build confidence among customers and provide them with risk coverage. Once the customers have tried a few iterations, then they are offered the option to buy more iterations or features as needed. Some customers are still hard to convince about that, which forces agile practitioners to compromise with fixed bid contracts. In such situations, many agile practitioners keep the customer unaware of the agile practices being followed internally at the agile organization.

Another way of contracting in agile that could be used is to allow customers to change feature priorities for free so long as the total contract work remains same, it also enables the customers to add new features if low priority items of equal work are removed from the contract.

The firm and customer in this research used something similar to the second way of contracting in which they kept the contract as it was in the traditional way, and also kept the customer in loop of Scrum process; although this contract may not be fair for one of these both parties, yet the relatively small size of this project compromised that problem into nothing.

5.4 Findings:

A lot of interviews had been made from within all districts of Palestine (west bank), in order to make a step forward in the SW

development field in an uncertain market with the least costs of changes. The below addresses are the findings after taking into consideration the Scrum literature, interviews and the practical experience about agile gained throughout the project development.

Initial Adoption of Agile: The agile initiative in the firm was undertaken by a researcher who was once one of its employees; the researcher offered the research idea on the organization management, and found that there were already some employees with the same idea but didn't succeed with turning that idea into reality.

The management supported the idea by allowing it to be applied on a medium project that was still under development. The researcher made a lot of interviews to support his idea about the traditional methodology of development and the new methodologies followed in Palestine (WB).

The Agile Team and Training: the team consisted of 7 people with a development background including architects, developers, testers and project manager. The researcher and two of the team had some knowledge about agile Scrum; the researcher gained a lot of knowledge using the research literature, and made many interviews that empower his knowledge in theory and practice about SW methodologies in Palestine; this knowledge helped in many cases where non-interpreted issues were being faced. For the other two in the team, they already knew a few things about agile through IT magazines. So the researcher distributed the Scrum

handbook over the team and explained the concepts for them in two sessions; one for concept and one for practice.

The Environment: The team was given freedom to establish their own work environment in a meeting room inside the building at start, but then they moved into separate room that has enough space for all the team. This led to increase the communication between those in the office. From a negative view it was reported that the environment was very noisy and no room left for people to think on their own especially the overhearing of irrelevant discussions.

Communication: Within the office room, face to face communication was easy. Externally, information was conveyed via meetings, emails and verbally. Internally, a communication rift has been noticed between the project manager who were external to the office and the developers who were in the office, but through the repeated actions and visits, that rift kept hidden, besides, the Scrum master was the one responsible for the link between the members and the project manager.

Project Planning: The management was used to the traditional model project plans, but since the management wasn't always able to determine the percentage of work completed what stages they were at, it didn't have confidence with that way. Delivery dates, costs analysis and statistics on the progress being presented were also part of the planning the management used to it. Although this wasn't the main concern of the

Scrum master, yet the tracking tool used within the project helped the project manager to a great extent to get some indications like the previous report plans. Besides, having one release after each sprint was another great indication about the progress of the report.

The Customer: The PO was the customer interface, but what was reported that the PO could not prioritize the features and tasks properly to determine which should be developed first. However, two staff had been assigned by the ISP to periodically pass by the Scrum team and offer the help in this work, which made the communication very easy with the customer.

Agile Methods and Practices: The team tried its best to follow Scrum as written in the Scrum handbook and applied by some firms in the local market. The more popular practices included user stories, stand-up meetings, pair programming, continuous integration of code, continuous testing and adaptability to change. Some time wasted on non-coding activities like learning, chatting and other stuff. Some tools were used to help in the work such as the JIRA tracking tool and MS office.

Throughout and after the case study project implementation, the researcher had access on a similar previous project that has the same budget and features but was developed with traditional methodologies (Waterfall) from within the same local firm in order to get some light on how progress was made using the agile Scrum on this case study in comparison to the previous ways. The old project was also about

developing a provision system for a telecomm company who wanted to update their provisioning system to contain more switches and to add more services related to the customers usage of their services. Although it is not for an ISP system, yet it has the same features and functionalities, so has for the budget and the set time to complete, even more, some of the developers worked on the old project worked also on the case study project, which makes it fine to make some analytical measurements to compare between these two projects.

The below addresses are the main point in light to measure the progress upon the new SW methodology followed by the local firm with another similar project previously developed inside the firm with the old methodology (Waterfall).

Customer Satisfaction: As reported by the developers involved in the case study, the customer was very satisfied with the project march from the starting phase like the way the team gathered the requirements, how he recited his requirements, how he was able to sort his priorities and change them according, how it was easy to change his needs in the middle of development and he was also satisfied with his involvement in the process from the start, he was also satisfied with product that he had and didn't send his complains about bugs or wrong functionalities; his only requests were for new features to add because of the change in his business way. On the other hand, according to the developers who participated on the old project, the customer kept complaining about the low quality of the product

and kept asking for amendments in the functionalities which causes him dissatisfied in general, especially that he wasn't involved totally in the process other than the requirements gathering, yet it was only a short period where he gave his needs for that period and wasn't updated with any information until the acceptance stage where a lot of things needed to be changed and a lot of latency happened. In general, the customer satisfaction on the new project was a lot better than the old one.

Change Requests: In the case study project, the change requests were controlled by the Scrum master, PO and the project manager. Since the customer was involved in the process from the start, so his changes weren't too late to be adopted, and the priorities of the tasks could be easily according without the need to panic or to add any more additional time or cost to the project, and the customer's acceptance on the change would be easily done at each sprint without waiting for the end of the project. On the other hand, on the old project, the change requests weren't easy to handle since it was difficult for the project manager to change features without making sure that previous steps had been made, furthermore, the customer simply wasn't able to ask for changes until the acceptance begins, where the work had already been completed for a large percentage, and at the best case, if the project manager let the customer to ask for a change in the requirements, some contractual issues may appear sooner or later. And for that, the project manager insisted to gather all requirements before the start

of the development and asked the customer not to ask for any change for an agreed limited period until the team further progresses in the project.

Support Agreement: In both projects, there is a support agreement in which the firm provides the necessary maintenance of the product like update or add a new feature, change a current feature or to solve any problems or bugs that may appear. The contract states that the system problems or bugs would be fixed for free, and for adding or updating any feature, the prices differ depending on the time. The project manager for the old project reported that at the time of deployment and production, there are many bugs appeared because of lacking of testing, not being part of the requirement or simply wasn't calculated for it by the customer. He said that for the first three months after deployment, there happened to be 11 system bugs appeared while the customer was working on, and the customer asked for 6 changes and updates to be done on the system. Then after that, it is around one update per two months. On the other hand, after the case study project finished, the project manager reported that after the deployment and production of the project, only three system bugs appeared with the customer; one of them was intentionally postponed after agreement with the customer. The customer asked for two updates because of the change in the business procedures of the customer himself and not because of a misunderstanding in the requirements.

Internal Environment for Developers: Many developers stated that they felt more freely and responsible within the case study project, more than

they felt during the old project, mentioning that some of them participated in the both projects. They liked working environment with agile Scrum more than the traditional one; as an example they liked the idea of volunteering for their own tasks although some tasks were being assigned to them. They also liked the principle of PP much more than each one works for himself; at least not for all the time. The juniors were comfortable that someone more expert than them is helping and coordinating, and although the Scrum deals with all team members as equal, yet the seniors didn't find problems dealing with the juniors as peers since this distribute the work load among all members without one members works a lot more than other. This same idea helped the juniors to develop more rapidly than in the ordinary environments; when they got the knowledge directly from their seniors, it saved them a lot of time and efforts and gave them the opportunity to be more creative.

Teamwork and Self-Organizing: In both projects, the team members reported that team working was good in general, although there was a team leader in the old project but there was no explicit team leader in the case study project, the Scrum master was doing the coordination matters when needed. For the self-organizing, the team felt with more autonomy when working under the Scrum agile since it increases the responsibility and privileges for members on the contrary of the traditional methodology where the project manager or team leader determine exactly for each developer what to do and how to do although the degree of this

determination differs from firm to firm and from person to person. For the redundancy, where the team members need multiple skills to be able to perform each other's jobs and substitute each other as their circumstances demand, this could be helped when the more self organizing exists since the PP and internal understandings facilitates the carbon copying for team members to each others. Redundancy can also be found in the old project since the team members were concerned, as one developer said, to back up each other so that they be able to be free when they had to do something out of scope.

Leadership: As stated before, in the old project there were one project manager and team leaders, and the team leader monitors and may be the one who distributes the tasks into the members of the team, and the team members do what is assigned to them; although this process differs from time to time and may depend on personal opinions, yet it still has the same scope for all differences, and there is always the leader who is responsible over the employees and members. On the other hand, when the team members worked on the Scrum agile project, the responsibilities were somehow distributed among them, and felt that they didn't need a team leader although they felt the at most need for someone to defend them and coordinate organizational matters for them. In both projects, there must be at least someone not only does the coordination and managerial stuff, but also inspire and motivate the employees, which in the old project was the

team leader, and in one way or another was the Scrum master in the new project.

Management Behavior: Usually the commitment of the management for agile in general is difficult and can't be seen since it depends on the convictions of the management about the feasibility of agile. The local firm's management was convinced that old traditional methodologies have many shortcomings, and found the chance to implement Scrum within it since it was also convinced that a change must happen to the SW development in the firm. Although the management tried its best to show the commitment to the research objectives and gave a huge freedom to the Scrum team to work, yet the management tried to intercept the work through introducing some tasks out of the sprint's scope, and sometimes the Scrum master was able to cover for the team, although in other times, he didn't do that and the tasks had been introduced which dictates to sometimes changing in the priorities in the backlog, but that in general didn't affect the overall progress.

Customer Involvement: The customer involvement wasn't a big concern in the old project, at least not for all phases of the project. The project manager of the old project stated that the customer was mainly involved in the requirements gathering phase, but wasn't mostly involved in the later stages until the user acceptance began where he had periodical visits to the site and watch the demonstrations done by the team. The customer sometimes felt that many features weren't supposed to work like that or

weren't supposed to be like that, and sometimes he rejected the feature and didn't accept it at all, he wasn't being involved in the development, but he was allowed to ask for changes only for a limited time after the development began when the project manager closed the door of the change until the acceptance starts. This made the customer someone strange on his project and made him confused about what he really wants. On the other hand, on the Scrum agile project the customer was allowed to be involved in the project from the start of the project, and his opinions were being taken into consideration at all stages so he didn't have to ask for formal change at the last of the project and didn't feel strange in front of his project. He was also allowed to ask for change, although that many changes were postponed into the next sprints. In general, the customer was satisfied with his involvement more than being involved in certain phases of the project, and saved a lot of efforts for the team and the customer to overload themselves with the late changes.

Budget and Time Limits: Both projects almost had the same budget and time limitations, but the old project went a little over budget since it was going to pass the allowed time which forced the firm to make its developers to work overtime. Besides, the team was exhausted by the fixes asked by the customer after the delivery based on the support agreement until the system became stabilized. The newer project was within the budget and time limitation, although the customer had asked a few changes that were out of scope and the management agreed to do.

5.5 Future Studies:

1) Optimizing Agile Scrum framework to more suit the local environment.
2) Many possible conflicts between the local IT organizations and customers emerge during the development, a detailed research about the contract management in agile versus the traditional contract management can conducted and it will help a lot in lessening the legal conflicts.
3) The customer communications with the development team are extremely important; a research can be conducted explaining in details the difference between the communications between the traditional way and agile.
4) The role of project manager in agile can be thoroughly studied in comparison with the team autonomy recommended by agile; this research fits most to be conducted areas that have not been studied before like here in Palestine.
5) The same research can be conducted on a local organization that have multiple projects, where a Scrum pool can be created and researches can be conducted upon it.
6) A research about risk management in DSDM (Dynamic SW Development Methodologies) in Palestine.
7) The results that had been concluded from this research might be applicable to SW projects with similar cases and not with all SW

projects, and only applicable for local environment. They can't be confirmed for other huge SW projects or SW projects that might not be in Palestine since others may believe that traditional methodologies might be even better for other projects. A research about the effect of agile Scrum over huge projects or multiple projects would be a great addition to this research.

5.6 Research Contribution:

A lot of interviews had been made with many local IT firms; most of them were following the traditional methodologies of SW development, and only few is following the new methodologies like agile Scrum, yet most of them reported that they are not developing successful projects from the first time and that the support agreements are costing them. This research tried to ask questions of why problems in this local market happen and how can we solve these problems. The researcher thinks that this research offers something new and hadn't been searched a lot on the local market, and if really a formal result can be conducted from this research, then at least part of the problems should be solved, and the suffering of the IT customers and firms developing the system can be get rid of, which opens the door for more creativity in the field. After the case study finished, the researcher explored the views of the Scrum members involved in the project; they reported that a lot of the problems found in previous projects were solved and felt much better about their work and progress

besides the new things that they have learned through this new way of work which they were aware of.

This research was conducted with the sponsor ship of a local IT firm in Palestine called “Hulul for business solutions” that offered the place, project and staff for this research. Hulul was searching for a new way of SW development that helps to keep it in the vast and uncertain market of IT. Although currently Hulul stopped taking SW projects from any customer because of internal issues, but during the research, the results were great and the management was very happy to adapt the new way introduced by this research.

This research gives a great contribution for Hulul business solutions:

1) It is a practical trial to develop old traditional way of work into a new modern one; that could make Hulul a pioneer in the IT local market if adopted in the correct way.
2) It gives a clear assessment for the current way of work followed within Hulul in the project management.
3) Gives a clear understanding of the concepts behind agile, upgrading customer communications, testing acceptance and adapting changes within the work.
4) Introducing a new way of thinking into Hulul that worked on the same way of development over years.

5.7 Conclusions:

It is important to mention that what worked for this research isn't general for all other cases; although the result of this research encourages to follow agile Scrum in our local environment in Palestine, yet this needs to be searched more to ensure the results of this research and generalize it in order to make procedural rules to follow when different cases happen.

The problem definition will be stated in italic, and below it there will be the notes and indications:

How scrum is implemented within local software project to provide agility?

The conclusion is that there are some minor differences between Scrum in general and Scrum applied in this research. The following differences and suggestions for improvements have been identified:

- The roles of the Product Owner should be well defined so that all responsibilities are clearly covered.
- Work should be specified and included in the Backlog, if omission of requirements specifications and verification, there will be a risk to decrease the quality of the expected outcome.
- The Scrum framework is recommended to be followed as strictly as possible by the Scrum Team.

- Scrum team should not be disturbed during a sprint with tasks that does not originate from the sprint backlog.
- The Scrum master role could be less involved with software development and more involved with organizational issues.
- Adoption of a more specific definition of done is recommended.
- A sprint review, where the PO and Scrum team discuss the implementation, should be held before an item is considered done and allowed to be demonstrated.
- Sprint reports may help to share the experiences between teams.
- An increase of the level of customer collaboration is recommended.
- Trying to make new releases after every sprint.

If the above recommendations are followed the difference will be without significance.

What are the changes caused by the implementation of scrum inside some SW company?

There are a lot of sensed changes that can be caused by the implementation of a new agile methodology rather than following Scrum in such a local environment. After the implementation of this research inside the local firm, the results resembled the indications and deductions from the interviews: the customers started to change their mentality and seeking for being involved more and more into the project details, they started to feel the difference between being a knowledgeable customer who holds his

responsibility and an ignorant customer who only reads the contract and follow it items literally. On the other hand, the firm started to search for a real way to adapt with the changes asked by the customers, and making these changes deployed as soon as possible without affecting the whole progress of the project. Besides, the management was totally convinced that changing is inevitable, it is only a matter of who would start the change and lead it.

Is it possible to make additions to Scrum so that it more fits the local environment?

- Some changes and additions need to be made to the usage of the Product Backlog:
  - Work should be specified and included in the backlog, if omission of requirements specifications and verification then there will be risks.
  - The Product Backlog should be handled in a more suitable way than through a shared Excel-file.
  - Creation of any documentation not written in a sprint report or as a result of the definition of done should be handled as an item in the product backlog.
- A quality policy must be in place and improved based on feedback.
- Putting more strict limits about the sprints and removing the idea of "finishing the sprint within the expected time is something normal"

- With the practice over Scrum, some members may have a lack and loss-of-dedication to what they do, which leads into a loose of commitment to finish the sprint holding in mind that there will be no problems if they didn't finish the sprint on time. The whole methodology must be questioned on how to enforce preventing such a culture among the Scrum members and by whom.
- A sort of motivations and punishments can be attached into the process other than the rules set by the HR of an organization, this sort of actions make the members give attention to what really the important is.

How can the success or failure of the implementation be measured?

According to the interviews, the customer is the one responsible for determining the success or failure of any project, and if it is agile project then the customer has bigger role to decide that. The customer in the case of the implementation conducted in this research, the customer finished the project very satisfied and gave a very positive feedback to the management about the track of process and its results. He was very happy with keeping him in the loop of the project progress, and didn't have to ask for a lot of changes after a release had been deployed.

References

Agile Alliance (2001). **Manifesto for Agile Software Development**.

Alhamdani, M., Aljaderi, A., Qandelji, A., Bani Hani, A., & Abu zeneh, F (2006). **Research Methodology: First Book: The basics of scientific Research,** 1st Edition, Amman, Amman Arab University for Graduate Studies.

America Heritage website, URL is http://www.ahdictionary.com/, last visited 20-12-2013.

Baskerville, R., Ramesh, B., Levine, L., Pries-Heje, J., and Slaughter, S. (2003). **"Is internet-speed software development different?"**. IEEE Software 20, 6.

Benedicenti, L., Paranjape, R. (2001). "Using Extreme Programming for Knowledge Transfer". **Proceedings of 2nd International Conference on Agile Processes and eXtreme Programming in Software Engineering – XP**.

Bjorklund, Mary, Paulsson, Ulf (2003). **Seminar and book-writing, presentand oppose**, Lunds: University Press.

Boehm, B.W., Turner, R. (2003). **Balancing Agility and Discipline: A Guide for the Perplexed**, Addison-Wesley, Reading.

Bryman, A, Bell, E. (2007). **Business Research Methods**, 2nd ed.,Oxford University Press.

Bryman, Alan (2004). **Social Research Methods**, Second edition, Oxford: Oxford University Press.

Chantachaimongkol, Nichamon, Sincharoenpanich, Puangpetch (2013). **Critical factors for implementing the Scrum software development methodology**, University of Malardalen, Sweden.

Cho, J., Kim, Y., Olsen, D. (2006). "A case study on the applicability and effectiveness of Scrum software development in mission-critical and large-scale projects". **Americas Conference on Information Systems (AMCIS), 2006.**

Chow, T. ,Cao, D. B. (2008). "***A survey study of critical success factors in Agile software projects"***, **Journal of System software**, vol. 81, no.6.

Cockburn, Alistair (2000). **Agile Software Development**, Upper Saddle River: Addison-Wesley.

Cockburn, Alistair (2007). **Agile Software Development –The Cooperative Game**, Upper Saddle River: Addison-Wesley.

Cohn, John (2007). **Advice on Conducting the Scrum of Scrums Meeting**, Upper Saddle River: Addison-Wesley.

Cohn, M. (2004). **User Stories Applied: For Agile Software Development, the Addison- Wesley Signature Series**, Addison-Wesley Professional, Reading.

Cohn, Mike (2005**). Agile Estimating and Planning**, Prentice Hall.

Coldewey, J., Eckstein, J., McBreen, P. & Schwanninger, C. (2000). "Deploying Lightweight Processes". **Conference on Object-Oriented Programming, Systems, Languages, and Applications on Addendum to the 2000 proceedings.**

Coman, I.D., Sillitti, A., Succi, G. (2008). "Investigating the Usefulness of Pair-Programming in a Mature Agile Team". **Proceedings of 9th International Conference on Agile Processes and eXtreme Programming in Software Engineering – XP.**

Creswell, J. W. 1994. **Research design: Qualitative and Quantitative Approaches**, Thousand Oaks: Sage Publications.

Daft, R.L., Lengel, R.H. (1986). "***Organizational information requirements, media richness and structural design***". **Management Science 32**, 1986.

Dalcher, Darren, Brodie, Lindsey. (2007). **Successful IT projects**, London: Thomson Learning.

Denning, P. (2004). "***The Social Life of Innovation***". **ACM Magazines and Online Publications.**

Drever, E (1995). **Using Semi-Structured Interviews in Small-Scale Research: A Teacher's Guide**, Edinburgh, Scottish Council for Research in Education.

Elder, S. (2009). **ILO School-to-work transition survey: A methodological guide,** Geneva, ILO.

Emery, F., Thorsrud, E. (1976). **Democracy at work: the report of the Norwegian industrial democracy program**, Martinus Nijhoff Social Sciences Division, Leiden.

Fowler, M. 2005. **"The New Methodology**", http://www.martinfowler.com/articles/newMethodology.html.

Gross, H.-G., Melideo, M., Sillitti, A. (2005). ***"Self-certification and Trust in Component Procurement"***. **Journal Science of Computer Programming**, Volume 56, Issue 1-2.

Gustafsson, Kristofer, Jacobsson, Johan (2009). **Quality assurance with TL 9000 in agile software development of set-top boxes**, University of Linkoping, Sweden.

Gustavsson, Tomas (2007). **Agile art to complete the project**, Karlstad: TUK Publishers AB.

Highsmith, J. (2002). **Agile Software Development Ecosystems**, Addison-Wesley, Boston.

Hoegl, M. & Parboteeah, K. P. 2006. "***Autonomy and teamwork in innovative projects"***. **Human Resource Management**, 45(1); 67-79.

Kalermo, Jonna, Rissanen, J. (2002). **Agile software development in theory and practice**, University of Jyovaskola, Finland.

Kautz, K., Nielsen, P. (2004). "***Understanding the implementation of Software process improvement innovations in software organizations"***. **Information System Journal 14**; 3-22.

Kirkman, B.L., Rosen, B. (1999). "***Beyond self-management: Antecedents and consequences of team empowerment***". **Academy of Management Journal**, 42; 58-74.

Kniberg, Henrik (2007)**. Certified Scrum Product Owner course**. PowerPoint-presentation. Stockholm.

Kokko, Antti (2013). **Improving requirements management practices in agile software development environment**, University of Applied Sciences.

Kolawa, Adam, Huizinga, Dorota (2007). ***"Automated Defect Prevention: Best Practices in Software Management***". **Wiley-IEEE Computer Society Press.**

Komi-Sirvio, S., Tihinen, M. (2005). "***Lessons Learned by Participants of Distributed Software Development***". **Knowledge and Process Management**, 12(2).

Langfred, C.W. (2000). ***"The paradox of self-management: Individual and group autonomy in work groups"***. **Journal of Organizational Behavior, 21**; 563-585.

LaToza, T.D., Venolia, G., DeLine, R. (2006). "***Maintaining Mental Models: A Study of Developer Work Habits***". **International Conference on Software Engineering.**

Lawrence, R., Yslas, B. (2006). "***Three-way cultural change: Introducing agile with two non-agile companies and a non-agile methodology***". **Proceedings of AGILE Conference**.

llen, T., Henn, G. (2007). **The Organization and Architecture of Innovation: Managing the Flow of Technology**, Elsevier, Oxford.

Malekzadeh, Behrooz (2010). **Event-Driven Architecture and SOA in collaboration**. University of Gothenburg, Department of Applied Information Technology, Gothenburg, Sweden.

Mann, C., Maurer, F. (2005). ***"A case study on the impact of scrum on overtime and customer satisfaction"***. **Proceedings of the agile development Conference**, 2005.

Marchenko, A., Abrahamsson, P. (2008). "**Scrum in a multi project environment: An ethnographically-inspired case study on the adoption challenges**", http://dx.doi.org/10.1109/Agile.2008.77

Mason, J (2002). **Qualitative Researching**, London, Sage.

Mirza, H.Z. (2010). I**llustrating the use of agile software development process,** Sweden: University of Gothenburg.

Molokken-Ostvold, K. Haugen, N.C. (2007). "***Combining Estimates with Planning Poker-An Empirical Study***". **Software Engineering Conference.**

Morgan, G. (2006). **Images of Organizations**, SAGE publications, Thousand Oaks.

Patel, Runa, Davidson, Bo (2003). **Research methodology bases to plan, conduct and report an investigation**, Lunds: University Press.

Pearce, C.L. (2004). "***The future of leadership: Combining vertical and shared leadership to transform knowledge work***". **The Academy of Management Executive** 18(1); 13.

Rising, L., Janoff, N. (2002). "***The Scrum software development process for small teams***", **Software, IEEE**, 2002.

Schatz, B., Abdelshafi, I. (2005). "***Primavera Gets Agile: A Successful Transition to Agile Development***". **IEEE Software, 22(3);** 36-42.

Schwaber, Ken (2004). **Agile Project Management with Scrum**. Redmond: Microsoft Press.

Schwaber, Ken (2007). **The Enterprise and Scrum**. Redmond: Microsoft Press.

Schwaber, Ken (2008). **"Scrum",** http://www.scrumalliance.org/resource_download/441.

Schwaber, Ken, Beedle, Mike (2002). **Agile Software Development with Scrum**. Upper Saddle River: Prentice Hall.

Shore, James, Warden, Shane (2008). **The Art of Agile Development**, Sebastopol: O'Reilly Media Inc.

Slappendel, C. (1996). "***Perspectives on innovation in organizations***". **Organization Studies** 17(1); 107-129.

Softhouse (2006). "**Scrum in five minutes**", http://www.softhouse.se/Uploades/Scrum_eng_webb.pdf. Retrieved: 2012-11-08.

Stake, R. E. (1995), **The art of Case Study Research,** Thousand Oaks, CA: Sage Publications, ISBN 9780803957671.

Surabhi, Vohra (2013), **Agile Testing in a Plan-Driven Process**. University of Erlangen-Nuremberg, Germany.

Sutherland, J., Schoonheim, G., Kumar, N., Pandey, V., Vishal, S. (2009). "***Fully Distributed Scrum: Linear Scalability of Production between San Fransisco and India***". **Agile Conference, 2009**; 277-282.

Sutherland, J., Schwaber, Ken (2007). "**The Scrum Papers: Nuts, Bolts and Origins of an Agile Process**", http://www.jeffsutherland.com/scrum/ScrumPapers.pdf

Sutherland, Jeff (2010). **Scrum Handbook**, Scrum Training Institute.

Szalvay, Victor. 2004. **An Introduction to Agile Software Development**, Danube Technologies, Inc.

Takeuchi, Hirotaka, Nonaka, Ikujiro (1986). "***The new product development Game***". **Harvard Business Review**, vol 64; 1.

Truex, D., Baskerville, R., and Klein, H. 1999. "***Growing systems in an emergent Organization***". **Magazine Communications of the ACM**, vol 42; 117–123.

Wikipedia website, URL is http://en.wikipedia.org/wiki/Planning_poker, last visited 30.05.2012.

Wikipedia website, URL is http://en.wikipedia.org/wiki/Staff_augmentation, last visited 27.06.2012.

Williams, Laurie, Cockburn, Alistair (2003). "***Agile software development: it's about feedback and change***". **IEEE Computer Society**, vol 36 (6); 39-43.

Yin, R. K. (2007). **Case study research: Design and methods**, Thousand Oaks, CA: Sage.

Appendix A
Interviews Questions

Interview 1:

Mr. Ala Thiab, Project Manager at a local company

Question1: The first question is about the automation testing. How do you relate between automation testing and the instability of environment?

Answer1: We have to remember two things: the first thing is, when we do Scrum, we just can't introduce any new task into the sprint after it starts, unless we swap among the already existing tasks with new ones suggested by the client (product owner) and on a limited scale. The other thing to say is that we don't use Scrum (or full Scrum) on all our projects; so the automation testing isn't something that we use on all projects. In general, other than meeting the requirements against the changing and instable environment, we also seek the efficiency of the testing when using the automation. It saves us a lot of time, like if we are working on an application that needs continuous testing or regression testing. While if the project we're working on is small, it is better to manually do the testing than to automate the test. We usually use the automation to minimize the efforts and costs.

Justification1: One of the benefits of using SCRUM is the automation testing in order to adapt with the new requirements. Using the automation testing in which we have a tool to automate our tests or we can use manual scripts, we can respond to the changes that are to be made during the different sprints of the project, and we in our turn must check and analyze

the results emerged from those automated tests and make our decisions based on them. This question complies with the objectives: 1-6.

*Automation Testing : test automation is the use of special software (separate from the software being tested) to control the execution of tests, the comparison of actual outcomes to predicted outcomes, the setting up of test preconditions, and other test control and test reporting functions. Commonly, test automation involves automating a manual process already in place that uses a formalized testing process (Kolawa and Huizinga, 2007).

*Regression testing is any type of software testing that seeks to uncover new software bugs, or regressions, in existing functional and non-functional areas of a system after changes, for the purpose that a change such as those mentioned above has not introduced new faults (Wikipedia).

Question2: To what degree do you evaluate your usage of automation testing on your projects?

Answer2: Before I answer this question, I have to tell you that we are outsourcing company that deals with staff augmentation. With staff augmentation, it might differ from outsource to outsource depending on what the outsource working on; means by that we might be working with an outsourcing company that doesn't work totally on agile, or might be working on agile but the system is small, where in both cases we use the manual testing, as the same as we might work on projects with a completely automation testing. But in general, for my working experience within this company, the automation testing is the main testing currently existing.

Justification2: This question to indicate the level of change response the interviewed company reached. This question complies with the objectives 1-6.

Staff augmentation is an outsourcing strategy which is used to staff a project and respond to business objectives by evaluating the existing staff and then determining which additional skills are required. (Wikipedia).

Question3: What are the tools usually used in the automation testing in your projects?

Answer3: There are a lot of tools used; some are commercial tools like "test complete", we use the scripting in general using: Python, Perl and some shell scripting depending on the environment. For other projects, we might use internal tool (built-in tool); in which the client itself has a built in tool that is used for his sake.

Justification3: This question to see the variance in the usage of the used tools among the different companies apply agile in order to measure which are the best tools to be used, which leads us to advise them for other companies who wants or needs to follow SCRUM. This question complies with the objectives 1-6.

Question4: How do you reconcile between the roles of project manager vs. the scrum master vs. the team leader? And how it is complied with the local area culture?

Answer4: Ok, let's be honest. We don't follow literally the Scrum recommendations written in the Scrum Book, since we don't have the full commitment from all the customers to follow Scrum with us. Some customers don't follow agile, others may follow agile but not with Scrum, it depends on the teams working with those customers, but in general the

main software methodology followed here in this company is agile Scrum. So in case we apply a project with Scrum, we try to follow Scrum as much as we can and benefit from all and every recommendation written in the Scrum standards. If we decided to follow Scrum, then at least the Scrum basics must be followed in order to gain the maximum benefits obtained by Scum, but I still can't say that we apply Scrum 100%, I can tell you that we reach to 90% of Scrum through our projects that we decide to use Scrum in. Now for the project manager (PM) role, the PM doesn't get involved in the Scrum meetings nor in Scrum application; he usually attend the Scrum meetings only as listener for his own information because he is the one responsible to the communications with the customer. On the other hand, for each Scrum team we shall have the Scrum master who maintains the Scrum process only as a process with no responsibility over the team. On the organization level, the company gives roles to its employees like project manager, team leader, seniors, etc. but while we're working on Scrum, those titles became only for organizational needs and our work becomes peer to peer with no one better than the other; we all work together and all are equal and all are responsible equally.

Justification4: We are searching for the applicability to follow SCRUM in our local environment; in which there is a huge conflict among the roles of project manager and team leader, so what will happen if we add a new role to the chain of commands to be followed in such an environment. This question complies with the objective 2.

Question5: What is the degree to commit agile, from within the team and from the organization management?

Answer5: To be honest, I don't think that the higher management is mature enough to adopt the agile in its full formation, but to mitigate this

immaturity, but usually the higher management stays out of the project even in the final and financial decisions, it is taken by the project manager himself. So, on the project level, yes we have the commitment of the management, but in the organization level, we still lack the concept of agile culture, not because the organization doesn't want to absorb the concept, but the business environment is what obligates the company to be not immature to follow agile, since as said before, we can't follow agile on all projects since it depends on the client methodology which reminds us that we works under the concept of staff augmentation.

For the team's internal commitment to Scrum, I can say that in general, the Scrum team members lack the knowledge about Scrum, but as a general case, you will find two types of people within the team, one who likes to follow Scrum and the one who don't, the type of members who don't like Scrum usually are from the senior members who don't like being treated as peers with the juniors or fresh members especially that the juniors have independency just like the seniors, and which eventually leads some of the senior to the fear that the junior will not do the tasks as expected. In addition to the matter of seniors concerns, we found that some of the juniors are not responsible enough to be part of the Scrum team not to mention the lack of the Scrum knowledge.

Justification5: Some of the main obstacles for applying agile usually are the lack of commitment from inside the organizations i.e. management, lack of commitment from within the team, so this question is to take indications about those commitments toward SCRUM within the local companies that is trying whether totally or partially moving into agile. This question complies with the objective 2.

Question6: Did you work in a traditional software methodology i.e. waterfall? In your opinion, which is better, agile or waterfall for the normal projects that you work on and why?

Answer6: Well, not exactly. I worked on projects that follow the traditional ways of software development methodologies, but they didn't apply the waterfall exactly, you know in the local market when we start to work on a project, we usually take our time with the documentations and requirements (if the case was a formal traditional way of development), but when we find that we took much time than expected during the first phase (documentation), we start working in unorganized way of development due to the pressure of time and cost; you can say that this situation is one of the shortcomings of waterfall methodology.

Now for the second section of the question, of course I will much prefer the agile to work on. Although that the local culture still doesn't have what it takes to work under agile, and doesn't assist our attitude with agile, still we need to build such a culture, so we try as much as we can to make a solid base following Scrum. On the other hand, although theoretically Scrum is good, but this still in theory, and may be a kind of advertisement to agile, yet we can see a lot of benefits to Scrum. One of the benefits is the communications with the clients; in the waterfall, the communications with the clients usually happened from some of the seniors and management to contact the clients, while in agile, there is a chance that each one of the time may contact and communicate the client, which in turns builds the capacity and skill of all the members and increase the speed of obtaining knowledge and understanding of the system, which also creates more independency throughout the team, more seniors and more team leaders.

Another benefit we get from Scrum is the adaptability to change, since previously in the waterfall projects, any change to be introduced into the

project, it must go through a long process of formal procedures and approvals that is called “change requests”; which also may lead to change the expected end date, but with Scrum, the changes is different, and the customers are more satisfied when they see our dynamic response with the value needed and expected. Another thing in Scrum is the incremental process; where you as a developer gives a value to the customer through the delivered products and awaits his feedback, which makes the customer a quick learner since as a fact, we’ve never faced a customer who knows what he wants exactly, but with incremental, now it is possible; we can reach along with the customer to the best to think of. The customer can’t give a real feedback unless there are touchable ingredients that he wants to check.

Justification6: We need to know if the interviewee has an experience in waterfall also in order to know if he can give us some difference between SCRUM and waterfall, or the benefits of one methodology over another. This question complies with the objectives 1, 2 and 6.

Question7: How do you measure the quality, results, time to perform tasks? And what about the cost of performing a task?

Answer7: Quality is the most important thing, and when we talk about quality, I prefer the Scrum; I mean the better applying the Scrum the better quality we get. But I think it is honest to say, it varies from case to case; some case we find that the quality is going better and for other we see no difference in which depends on the team members since they are the ones who are going to do the retrospectives, discuss their problems and how to solve them. Now for our case as a local market or in this organization, there isn’t anyone who has experience from out of the country; in the outside world there are Scrum coaches and trainers, and as far as I know till now,

in this country there are only 3 Scrum masters including me, and when I say Scrum master I mean certified masters. You have to know that scrum is very simple and attractive when you read it in books, but it is very hard to implement; since you deal with a lot of people very closely, so it is around people more than anything else. We simply don't have scrum coaches in the local country, because we didn't invest in it and that is because we are not a product company; we are just following clients' methodologies. In general, the teams that are currently working within the organization gain their knowledge from the experience that the certified people here are transferring.

Justification7: If the interviewee is a project manager in a project with SCRUM, we need to know his ways in management, and if they were different from the ones used in waterfall. This question complies with the objectives 1, 3 - 6.

Question8: How much time have you been working on Scrum?

Answer8: About three years. In general all the teams in the company work on Scrum, but each time apparently has its own definition of Scrum.

Justification8: We need to know how much experience the interviewee has before proceeding in our questions with more in depth about SCRUM. This question complies with the objective 1.

Question9: How much commitment do you find from the customers? And how about the feedback you get from them in comparison to projects developed with other traditional ways i.e. waterfall?

Answer9: I faced the two cases where customers accepted the Scrum and where they didn't. In general, the customers are flexible about accepting Scrum; they don't have problems as an idea to work with Scrum, but only

as much as they can but they don't promise us anything; means that if they consider any new features as urgent features that must be introduced directly into the system, they directly try to push them into the current sprint without giving a big deal to the Scrum principles, so what we can do from our side is only to switch those urgent features with the ones that are currently within the sprint to keep our agility. So, my answer is that the customers had some intervention over the process, but I can confirm it is limited and we make the agility goes through as smoothly as we want. For the feedback, they like the fact that they know or expect to receive a release of the project once and awhile, like the releases provided at the end of two or three weeks as agreed with them from the beginning, and usually the releases come on time. They also like the coordination with the team and that they are being involved from the start of the project in the system mood.

Justification9: We are in local environment where there is still no final acceptance from the local market, and it may still prefer the old traditional way. This question complies with the objective 1, 2, 7 and 8.

Question10: For the clients that you work with, what is the degree of agility in scrum do you see them follow?

Answer10: I can say that they are just like us; they have the same engineers as we do, and when they find themselves under the pressure of something, they try to push things in the expense of other things, Meant by, they also don't follow Scrum 100% as suggested in the Scrum Bok.

Justification10: More questions to gain knowledge about the interviewee experience about SCRUM and waterfall. This question complies with the objective 1, 2, 7 and 8.

Question11: what is your evaluation for the success of Scrum over the projects that you worked for or managed?

Answer11: before I give you a clear answer, I have to know in which context you mean as successful, do you mean the success for Scrum itself or for the whole project, since the project might be successful with Scrum or without Scrum. **The interviewer said:** I mean if I am a customer who wants to implement a software project, on your practical experience, do you advise me to direct myself into Scrum or to keep the current local culture that we all know, not to mention that we already knew that we might face a failure based on known percentage of failures for software projects?

The interviewee said: I personally had some cases like that, but to be professional in my answer I will tell you one of these cases. In a certain software project of local market, for ministry of justice to be specific; the first phase of the project was made on the traditional ways of developing the software in the local market. The result of the first phase was low quality software that went totally out of scope with more time than expected and over-budgeted in cost.

In the end, there was no satisfaction; not from the ministry nor from the company, since from one side, the ministry didn't have what it expected, and on the other hand, the company suffered a lot of losses due to this phase, and the project was left out with a lot of problems. When the involved parties started the second phase of the project (the developing company was also us because we were the ones who already finished the first phase due to the contract); it was managed by Scrum and the good news was that we never faced any problem. I had the personal believe that this project wasn't going to success unless we followed Scrum, and thanks to Scrum, we really managed to overcome all the problems existed in the

first phase of the project and were able to launch the project with the maximum expected feedback from customer and the minimum expected bugs. We completed it on time, and every drop was delivered every two weeks into the ministry and they were reviewing those releases and give their direct feed backs. Those feedbacks were inserted into the next sprint of Scrum and fixed in the next iteration, and the result that they were satisfied and everything was verified, all the requirements that were either ambiguous or not mentioned from the start in the documents were clarified and implemented through Scrum, and at the end, the maintenance contract that was already signed with tem has been never used till now; they got back to us only once to help them to migrate their databases into the newer system although this step was out of contract. Now for other projects, we usually start the Scrum from scratch, sure we face bugs, but definitely they are not serious bugs that block the release and can be handled.

Justification11: We need to know the feedback that our interviewee had from working or managing project using SCRUM. This question complies with the objective 1, 2, 7 and 8.

Question12: What are the size of the projects that you use Scrum in?

Answer12: Currently, the biggest project size I am supervising is 6 people; 4 developers and 2 QAs, including the Scrum master who is also a part of the working team. We have a client that has up to 23 people assigned to it, but not in one team, they are distributed over 4 to 5 teams working for the same client.

Justification12: some opinions claim that it is recommended through a lot of studies to use SCRUM with small and middle projects and to keep it from huge projects. From the previous information, we need to know what

our interviewee knowledge and recommendations about that information. This question complies with the objectives 1 and 2.

Question13: Since you have many teams working for the same client, how do you integrate them with each other in the whole system?

Answer13: We don't integrate them; they all fit in the same product, but each team has its own work and they all at the end fit without the need to integrate them, but I don't gave the case where I have two teams like example, and each team is working on Scrum for the same product and in the end I have to integrate them, I don't have such thing.

Justification13: We need to check how the integration process is done; if needed. This question complies with the objectives 1 and 2.

Question14: Can you determine the size of the project with another classification; I mean can you use the cost for the size? Like if you have a project with a huge cost within a time limit?

Answer14: Well, I can say that the majority of software projects that exist in the local market as a whole, and in our organization, don't exceed the 9 engineers working on it. Now based on the number of engineers you have on the project you can determine the cost size and classify the project that you are working on based on the cost.

Justification14: We need to know if the cost has a huge effect within SCRUM projects over traditional methodologies, and to see the best way to compare with the traditional projects using the cost. This question complies with the objectives 3 and 4.

Justification15: We need to compare the cost –in money- when following SCRUM. This question complies with the objectives 3 and 4.

Question15: Do you apply the overtime when you work on Scrum?

Answer15: No, we don't like to use the overtime; we try to reach the perfect situation where all the employees are out of their offices at 05:00 PM, but as I said, it is a perfect situation, sometimes –although a few- we use the over time if we had an urgent case that needs to be done urgently, or this case might happen because of some dereliction from the employees themselves.

Justification15: We need to compare the cost –in money- when following SCRUM. This question complies with the objectives 3 and 4.

Question16: As we know, the tasks are voluntarily assigned, so each volunteer must put estimation for his time consumption, so how such a thing like the latency happened in Scrum?

Answer16: Yes indeed it is voluntary, but sometimes the knowledge only exists with only one person, so it is assigned automatically. In general, we have two types of estimation; the one that is determined in the product backlog which is done at the first sprint planning, and the other estimation is the one that is happened during the second sprint planning. The estimation that happens in sprint planning one usually is done by the team in which they discuss the project with each other through the so-called broker planning. The second sprint planning is the actual planning that includes the real time with hours, and usually done by the engineer that volunteered for the task; we don't have the condition that it has to be done by all the team, we may be able to tell him that he can do it faster or something like this, but the final estimation is his estimation.

Justification16: As we know, in SCRUM, each one volunteers for the tasks in the backlog; he doesn't volunteer unless he knows exactly what and how to do, or at least the tasks should be understandable to him. Yet,

we found that many times latency happens even with SCRUM, and we are searching for the causes that might cause that latency to happen. This question complies with the objectives 1 - 4.

Question17: In your opinion, what are the risks attached to use Scrum on our local environment?

Answer17:

-No commitment from the customer side to Scrum, since we are working under the concept of staff augmentation which ties our environment to the client environment and methodologies.

-The immature commitment from higher management; i.e. when a human force is to be shared among two projects, then the conflict of being agile or not emerges, which leads the management to abandon agile

-The lack of commitment to Scrum from within the team and the experience inequality of Scrum peers.

- The lack of the independent engineers. One of the Scrum recommendations is that "Scrum team is independent engineers" in which we mean by responsible and knowledgeable persons.

Justification17: We are asking the interviewee's personal opinion -based on his own experience- about the risks of SCRUM to assess the usage of it within the local environment. This question complies with the objectives 1 and 4.

Question18: In your opinion, what are the Benefits attached with using Scrum on our local environment?

Answer18:

- The iteration and incremental process in scrum is a good way to keep the engineers focusing on the development.

- The Scrum gives the engineers high confidence and forms a better work environment
- A lot better communications with the clients; which leads to more satisfied customers and more eligible staff.
- The adaptability to change

Justification18: We are asking the interviewee's personal opinion -based on his own experience- about the benefits of SCRUM to assess the usage of it within the local environment. This question complies with the objectives 3, 5 and 6.

Question19: If you were allowed to add something into SCRUM, what would it be? Based on your experience, what would you advise the SCRUM experts to look more deeply into? And what to remove from SCRUM?

Answer19: Scrum is an open framework in which any tool, methodology can be applied. Grooming meeting in Scrum does not get the same attention that Sprint Planning 1 has. From experience working with Scrum, Grooming meeting is much more important and must be defined and conducted in an organized form.

Justification19: We are looking for any improvements that might enhance the readiness of agile Scrum to handle SW projects especially for the local market. This complies with the objective 10.

Question20: What is the traditional way of developing software in the local market?

Answer20: Traditional SW development methodology (Waterfall) and in some cases revolutionary SW methodology (iterative). Needless to say, many SW providers tries to adopt SCRUM but they fail to benefit much

from it, due to the prematurity of the SW providers and lack of commitment from management.

Justification19: To gain a deeper insight about the environments used within the local market. This question complies with objectives 2,3,4 and 5.

Question21: How do you deal with the contracts? And how do you define contract management?

Answer21: In our type of work, we don't. As we work Staff Augmentation (we sign long term contracts for services, not for projects). In SCRUM, working with contracts is a real challenge. Since SCRUM is dynamic framework that adopts new changes easily and may take the product into different one in comparison to what was initially in mind.

Interview 2:

Mr. Hani Abu-Ghazaleh, Project Manager at a local company.

Question1: Is there any automated testing made within your agile? What are the tools used for the automated testing? What are the tools used in agile?

Answer1: I am not sure I can answer this question because I don't have enough experience from technical side with Scrum.

Justification1: one of the benefits of using SCRUM is the automation testing in order to adapt with the new requirements. Using the automation testing in which we have a tool to automate our tests or we can use manual scripts, we can respond to the changes that are to be made during the different sprints of the project, and we in our turn must check and analyze the results emerged from those automated tests and make our decisions based on them. This question complies with the objectives: 1-6.

Question2: What is the degree to commit agile, from within the team and from the organization management?

Answer2: In general, the business of today follows agile, and your clients are also following agile, so you have to direct your team into agile environment and make them adopt it. On the same line, the top management is aware of agile and its benefits, and so it is also part of the support and commitment to this pace.

Justification2: this question to indicate the level of change response the interviewed company reached, and to know if it really has the Constituents to follow agile. This question complies with the objectives 1-6.

Question3: Why did you follow Agile? Is it because you have to follow what the clients dictate you?

Answer3: If you look at it in a way, we follow what clients work on and want us to follow them on the same track, but on the other hand, agile currently is vastly spread and among most of our clients in the world it is the followed methodology, and for the customers who don't use agile we offer them to use it and try to train them to follow us as easiest as possible.

The interviewer said: do they find problems in adopting agile if they don't know it? **The interviewee said**: personally I didn't find such an issue among our customers who don't know agile.

Justification3: We need to know what reasons caused the interviewee to follow agile and if he followed agile from his convince or because it is a dictation; since agile needs personal commitment before the clients' commitment. This question complies with the objective 9.

Question4: How do you reconcile between the roles of project manager vs. the scrum master vs. the team leader? And how it is complied with the local area culture?

Answer4: usually, the project manager is defined from day one, and he is responsible for the communication with the customer, and conveys the demands and feelings of the team into the top management, in addition to the follow up of the progress of the project. The same is also for the team leader who already has his role defined by the organization, although he may act only as a team member within the sprint.**Justification4**: This question to see the variance in the usage of the used tools among the different companies apply agile in order to measure which are the best tools to be used, which leads us to advise them for other companies who wants or needs to follow SCRUM. This question complies with the objectives 1-6.

Question5: As a project manager, are you the one responsible for the financial stuff related to the project?

Answer5: Yes and no. originally, in this company we have two kinds of projects; the fixed project with the known start and end, and the other type is opened projects like staff augmentation. The financial issues is related to the project manager directly since we are working based on assigned budget, while in staff augmentation we are lending our human forces into our clients and they pay for their contracts based in the agreements. And for the over time issues, it is assessed case by case.

Justification5: We need to verify the degree of financial freedom the interviewee has in case he works on SCRUM, since we also need to measure the difference in SCRUM over traditional methods. This question complies with the objective 3.

Question6: size issues

Answer6: don't know

Question7: How do you evaluate the agile as a process in your company? And how do you evaluate the success in your projects?

Answer7: Actually I think we work about 80% to 90% of Scrum on our projects that we're supposed to follow agile in them; since even if we want to use agile especially Scrum-, the customers may still want to work under traditional ways like waterfall because they are familiar with those methods other than agile. And for my evaluation of the successful projects, you have to remember that it is also important to finish the projects with the wanted quality. For me, the customers were most of the time satisfied with the product although we may have differences in some points, but in general the customers were satisfied with our products; and these differences usually because of the ad-hoc features that are being done without planning throughout the project and under the desires of the customers. But also to be honest, I have never been in a project were it worked 100% agile; there must be some outlines of agile.

Justification7: We are trying to see how this company adopt agile and commit agile. If the interviewee's evaluation to SCRUM is doubtable, then our measure also is doubtable. This question complies with the objective 1.

Question8: how do you face the closure in Scrum? I mean usually the most crucial time in a software project is the time of deployment and delivery, in other means it is the closure, so do you face real problems during this phase?

Answer8: sometimes yes we have problems and sometimes we don't; it depends on the new features and updates that we face at the last of the

project. Means by that, the customers may ask at the end of the project for new features that take a lot of time and effort and out of plan.

Justification8: We are trying to compare the problems faced in SCRUM with those being faced in traditional methodology, and one pace of search is the closure. This question complies with the objective 1.

Question9: In your opinion, based on your experience, what do you prefer to follow, Agile (Scrum) or traditional methods (i.e. waterfall)?

Answer9: to be honest with you, I didn't work a lot in waterfall environment, but on the other hand, I like Scrum and I prefer it over waterfall because I feel it is better organized.

Justification9: We need to know the interviewee's own convince about the usage of SCRUM. This question complies with the objective 1.

Question10: How do you measure the quality in your Scrum projects? And how do you measure it in terms of time, cost and effort?

Answer10: The most important measurement here is the accomplishment of the needed requirements; the more accomplished the more quality we have, the customer asked so and he gained so. Another thing is the cost itself; we have a budgeted project and we must not over limit it, and so also the time needed to finish within the due dates. Of course above all, the feedback we get from the client is a main measurement for our work. It is important to say here, that if during the sprints, if we thought that any feature can be done in a better way we expose our opinion to the customer and it is to him either to accept or reject our advice.

Justification10: It is assumed that the output with SCRUM will be good enough to be compared with the traditional methodologies of development, so we are trying to see how the interviewee makes his perceiving about the

quality in terms of time, cost and effort. This question complies with the objectives 3, 4 and 5.

Question11: How do you bear your responsibility as a project manager in agile projects?

Answer11: Well, from the technical side the team leader gives me the daily update of the work, and for the project updates, I involve myself in the daily sprints only as a listener and how the load of the project and overall project progress is. In addition to that, I usually communicate with the customer periodically or when needed. So, as you see, my main functionality here is Communications.

Justification11: One of the problems of agile (SCRUM) in our local environment is that there might be a conflict among the roles of the project manager vs. the team leaders vs. the SCRUM master at the organization level; so we are asking this question to know if there is such a thing inside their organization and the degree of it if there is. This question complies with the objective 1.

Question12: How do you deal with the instable requirements? And what is the effect on the cost?

Answer12: well, it happened with me before in a previous project; we gave requirements as a project input, but they weren't clear for all parties, the given requirements didn't give the whole picture for the needed product. In such a case, we were forced to take an educated guess, where yes we have to guess, but in the correct way. In that case, we informed the customer that we didn't have the whole picture of the system, but we also told him our expectation for the missed or unclear requirements, and asked his about our expectations and if it is ok to proceed (Sanity Check), and the customer in

his turn asked for time to review our expectations and returned with feedback if yes or no along with his notes.

Justification12: One of the benefits of using SCRUM is to face the instable requirement much easier than in the traditional way. We need to know the way this organization deals with the instability of requirements, means if they really are gaining the benefits of using SCRUM within their organization. This question complies with the objectives 3, 4 and 5.

Question13: Do you usually face problems in the local culture against agile methodology (SCRUM)?

Answer13: Yes indeed. The main problem is that a lot of people in the current organizations just don't say no, means if the client ask for anything, the organization simply doesn't say no for his requests; it is important to say no when we can't handle some requests.

Justification13: Our research is about applying SCRUM within the local community, so the merge of this industry with the local market must reached to a limit of adoption and acceptance that is worthy of trying. This question complies with the objectives 2, 4, 6, 7 and 8.

Question14: Do you have problems in clarifying the meaning of SCRUM to the customers?

Answer14: yes, most of them don't know what the meaning of SCRUM, so we have to make some introduction sessions to this environment before we start the development, but I can't say that these are problems; it is just they don't know what SCRUM is and we do that role for them.

Justification14: We need to know if we can consider this point as one of the problems that we might face during our work on SCRUM. This question complies with the objective 8.

Question15: In your opinion as a project manager, what are the risks that you face when you follow SCRUM?

Answer15: I think that the quality assurance is the biggest challenge here; like if the test plans don't have the right loop, or like if the developers always follow the rule of saying "NO" to the client.

Justification15: A direct question to gain some knowledge about the risks faced against following SCRUM within the local community. This question complies with the objectives 3, 4 and 5.

Question16: In general, how do you define the "Done" for your statuses in SCRUM?

Answer16: As you know we have the backlog, which within we already defined the done as the product owner visions it and lock the sprint, some sprints take 1 week, other take 2 weeks.

Justification16: Each SCRUM team must have a clear definition of the "Done" status for any task. This question complies with the objective 1.

Question17: Some people claim that the 15-minutes meetings are visionary meetings, since in our culture it is so difficult to follow every day meetings with only 15 minutes without being distracted with other conversations. Do you believe that these 15-minutes meetings are really applicable in our local environment?

Answer17: will, sometimes we consume less, and sometimes we consume more, and this happens for a lot of reasons; usually the organizer must have already made an agenda for this meeting and have control over this meeting to overcome the scope creep that might happen in such meetings, and he, as an organizer, he will estimate the needed time which might exceed the 15 minutes.

Risks:

- Desires of the customers with new features.
- The culture of not saying "NO" to the customer

Justification17: We need to assure the applicability of SCRUM within the local community. This question complies with the objective 1.

Interview 3:

Mr. Rami Shahwan, SW Engineer at a local company

Question1: Did you work on Agile? If yes, for how long?

Answer1: I worked on agile for more than year and a half, and expect to work on agile for more than one year to come.

Justification1: We need to confirm the experience of the interviewee against us in order to ensure the integrity of the answers. This question complies with the objective 1.

Question2: Did you worked on something other than agile? How did you start your transfer into Agile from the old way of development that you were working on?

Answer2: We was working on classical methodologies; we used to sit together with the customer and get the requirements, and after that we write the scope and vision document which we and the customer sign them off, and after this document we start our development and the customer starts seeing the system at the final stages of the project. But since this held a lot of risks towards our business, we started to introduce prototypes in which we show the system presentations over the papers to the customers along with some explanation to enrich the vision obtained by the customer during the different stages of the project, after that we started to let the customer

see the system for twice or thrice before we reach the last stages of the project. I mean that in here we were using the delivery-in-chunks for the system presentation to the customer, until we moved totally into SCRUM in our newer projects.

Justification2: We need to enrich our knowledge about the interviewee against us and to go in more deep searching for the ways transforming into agile. This question complies with the objectives 1, 2 and 10.

Question3: When do you think it is better to use agile?

Answer3: Using agile (SCRUM) makes sense in the large projects, not on the small ones. In small projects you can have plenty of control over them without the need to agile unlike the large projects where you have a lot of stake holders, a lot of risk, a lot of requirements and changes then the best methodology to use is SCRUM, especially when you have a constant change of requirements; those changes that lead into other changes that affect the whole project, so you just can afford those changes to happen, which mean we better use SCRUM.

Justification3: To gain a more understanding for agile. This question complies with the objectives 1, 2 and 10.

Question4: Do you sign the customer feedback?

Answer4: No, not always. We take their feedback as comments over the release that we give them after each sprint. They might take day or two to reply us with some tactical question about the current release that they have, lack of how-to-use knowledge about the features and other recommendations that they might give us for a better use. Although that these recommendations might breed more requirements, but it won’t make

a big change since the targeted actions are already clarified, and these small changes are only tiny changes that with them the system will be better.

Justification4: To gain more knowledge about the way of work followed inside the interviewee firm. This question complies with the objectives 1, 2, 7 and 8.

Question5: What do you think the risks that come with using agile (SCRUM)?

Answer5: In my opinion, the constant contact with the customer, makes the customer asks for more requirements, and creates a greedy customer who thinks that whatever he asks we would do, especially that we don't have the culture of saying no to the customer, which in a way or another is called "Scope Creep". Another risk I can think about is the availability of the team; meant by that, when you already have a team that is working on SCRUM then this is good, but the problem lies when some of this team members get out of the project (like job quitting) then you have to replace them and educate the replacements the way of doing SCRUM, which may be difficult for some newbie in this field.

Justification5: We need to know the different opinions related to risks attached to agile in the minds of the local firms who adapt agile as a SW methodology. This question complies with the objectives 1, 2 and 4.

Question6: What do you think about the usage of SCRUM?

Answer6: I think that SCRUM isn't the easy that one thinks it is, but also isn't the difficult that one may imagine; it needs a lot of practice from the whole SCRUM members, and with time the team will get used to it and it will become an easy way of development. But in case you lack the

experience, you can use only parts of SCRUM until you had the necessary control over those parts, then you can start using the full SCRUM.

Justification6: We need to know the different opinions related to risks attached to agile in the minds of the local firms who adapt agile as a SW methodology. This question complies with the objectives 1 2 and 5.

Question7: What do you think SCRUM needs more?

Answer7: I told you that we don't finish the sprints most of the time, and all those times, we really don't have serious problems to finish our work or dealing with the customers, but my point is that I don't want the culture of "not finishing the sprint within the expected time is something normal" to be dominant, I personally feel that the SCRUM teams indulge with that culture since they always get out of this intact. The aspect for the improvement here is to question the methodology onto how to enforce preventing such a culture among the SCRUM team and by whom; if this aspect wasn't ensured then it will be repeated over and over in each sprint. I think if some type of motivations and punishment were inserted into the process then that could be fine.

Justification7: We need to know the different opinions related to risks attached to agile in the minds of the local firms who adapt agile as a SW methodology. This question complies with the objectives 1 and 2.

Question8: what do you think of agile benefits?

Answer8: it important to build an environment where everyone within the team knows exactly what load he has to bear which leads into a relaxed environment and more focus on the productivity and the quality of the product. Another benefit from my sight is the accountability; where everyone on daily basis will be accountable for what he did in the day

before and what problems you faced and what you are going to face in this day. **Justification8**: We need to know the different opinions related to risks attached to agile in the minds of the local firms who adapt agile as a SW methodology. This question complies with the objectives 1, 2 and 3.

Question9: Do you follow the agile standard literally?

Answer9: We usually work on SCRUM environment where we try to follow the SCRUM standards. I can't say that we literally stick into agile and SCRUM standards, but we follow what we think it is better to be applied and complied with our local environment and doesn't oppose SCRUM recommendations; in other way I can say that we follow SCRUM till 80% of the whole project.

Justification9: We need to analyze current state of agile inside this firm to gain more strength in our research base. This question complies with the objectives 1 and 2.

Interview 4:

Mr. Diya Awad, SW Development Team Leader at a local company

Question1: How many IT projects did you work on or was involved in?

Answer1: They were about nine projects.

Justification1: We need to know the development back ground that the interviewee against us has until now so that we ensure that we are building our research on a strong surroundings. This question complies with the objectives 9 and 10.

Question2: Did you hear about SW methodologies?

Answer2: Yes I heard about SW traditional methodologies like waterfall, and also heard about the newer ones like incremental methodologies and agile.

Justification2: We need to know the development back ground that the interviewee against us has until now so that we ensure that we are building our research on a strong surroundings. This question complies with the objectives 1 and 6.

Question3: And what methodologies you have used through your career life in SW projects?

Answer3: I have worked in many projects, and you can say I worked most of them in traditional ways of so SW methodologies, and through those projects, we –me and other staff- reached to point that some additions into the methodologies must be introduced into the old ways.

Justification3: We need to know the development back ground that the interviewee against us has until now so that we ensure that we are building our research on a strong surroundings. This question complies with the objectives 1 and 10.

Question4: Can you speak a little about what you mean by the additions needed?

Answer4: I worked on many projects as I said. The last two projects I worked on, we used to gather requirements and divide the phases and determine our priorities in our requirements. Based on that we used to divide the work into many parts over groups, and each group has a weekly schedule to follow, and the tasks is distributed over the members within daily basis; these tasks were reviewed at the end of the day, and other tasks were being reviewed within the team on weekly periods, based on the task

nature and time needed. After finishing each phase, we were moving into the next phase and so on until we finish a certain phase or after certain number of months or periods. During these periods and phases, there will be a lot of meetings to discuss the progress within the project where all notes and observations are taken into consideration.

Justification4: We need to know the development back ground that the interviewee against us has until now so that we ensure that we are building our research on a strong surroundings. This question complies with the objectives 1 and 10.

Question5: Can you say about your working methodologies in the SW development that you followed, can you say that you were following Agile? I.e. did you use the sprint concept in your work?

Answer5: We never called it a sprint although it resembles the sprint a lot; we were holding a meeting at weekly basis where at each start of the week a brief discussion about the week before it, and within the same week were closing the issues of the previous week and determine the issues to be discussed in the next week.

Justification5: We need to know the development back ground that the interviewee against us has until now so that we ensure that we are building our research on a strong surroundings. This question complies with the objectives 1 and 10.

Question6: How were the tasks been distributed over the team members? Were they by the leader of the group or by volunteering?

Answer6: The tasks were being discussed and the team leader was distributing them among the members based on the knowledge that the team leader having on each member experience.

Justification6: We need to know the development back ground that the interviewee against us has until now so that we ensure that we are building our research on a strong surroundings. We also need to know degree of agile followed within this organization so that we can use the data in our research. This question complies with the objectives 1 - 5.

Question7: Was there a big difference in the levels and positions among the members that affect the pace of the development process? I.e. being senior or supervisor or team leader or being fresh employee.

Answer7: There was a technical team leader that has experience more than everyone else in the team, and he was distributing the tasks based on his practical knowledge with the other members and their experience, but in general, we worked like a team rather than individuals.

Justification7: We need to know the development back ground that the interviewee against us has until now so that we ensure that we are building our research on a strong surroundings. We also need to know degree and steps of agile followed within this organization so that we can use the data in our research. This question complies with the objectives 1 - 5.

Question8: What were the problems faced during the development of the project?

Answer8: Well, the first problem we mainly face is the shortage and amendments of requirements; we had many hard times dealing with the business when you work for a long time on a certain track and you suddenly find that the business is thinking in another track. Later way of working included some involving for the customers into the process of development, but that involvement was not regular and was made based on

the needs of the projects and development, and based on the management adaption for the process.

Justification8: We need to know the development back ground that the interviewee against us has until now so that we ensure that we are building our research on a strong surroundings. We also need to know degree and steps of agile followed within this organization so that we can use the data in our research. This question complies with the objectives 1 - 5.

Question9: Did you have problems related to the human interactions; either with the customer or within the team?

Answer9: Really we didn't face problems from management side, neither from the customer side adapting with our trial to modernize our way of work, on the contrary, the business side were rested for being involved in the process of development on shorter time intervals; especially being shown the progress of the project as some working samples. We tried to follow the progress with the customer based on groups of tasks or iterations.

Justification9: We need to know relationship with the customer or within the team during the whole followed set of procedures. This question complies with the objectives 1 - 5.

Question10: Did you have problems with the closure phases of the projects that you tried to introduce a new ways of development into them?

Answer10: Really we had problems at the last stages of closure, especially from the business side during the UAT; they had changed the way of doing their business in a certain aspect, which reflected the new system we were developing, where we had to change a lot of things in order to cope with the new changes introduced at the last stages of the project which had a lot

of effect on the closure especially that the business side needs everything momentarily.

Justification10: We need to know the details of problems faced during the transition; in this question we are focusing on the closure phase. This question complies with the objectives 1, 2, 6 and 10.

Question11: You are saying that they keep changing their mind especially at the late stages. Why do they ask for changes although you had involved them in the different processes of the project as you say on certain periods?

Answer11: For our case, the business changes the way they doing their work at the late stages; means that the system we are developing are no more suitable for their work, which lead into changing a lot of parts to the system; their way of work changed forcibly and not optionally, so we only had the option to change the system based on that.

Justification11: We need to know the details of problems faced during the transition; in this question we are focusing on the closure phase. This question complies with the objectives 1, 2, 6 and 10.

Question12: What other problems did you have in this project?

Answer12: In the requirement gathering phase, the business stake holders weren't there to present their side of information and knowledge related to the requirements, so the loop missed some parts that had a substantial effect on the system, which leads into more changes into the system being developed.

Justification12: We need to know the details of problems faced during the transition. This question complies with the objectives 1, 2, 6 and 10.

Question13: Who did introduced the idea of agile into the team that you were working within? I mean how did the team started working on it?

Answer13: First, it was the work experience that was gained through the years of working. Besides, some guys had some knowledge about agile and they suggested to start trying to use it in the development pace within the company; their knowledge was gained either from academic path or knowledge from other companies. Also before we started using agile, we made a search on the work development methodologies currently in the world, and agile just seemed the good one for us. We followed agile in this project since it wasn't a huge system that may complicates the process, so the fear of following agile wasn't so great that prevents us from using it in this situation.

Justification13: We need to know the details of problems faced during the transition. This question complies with the objectives 1, 2, 3, 6 and 10.

Question14: How do you evaluate using agile in your projects?

Answer14: I can say it is a successful story to introduce agile into projects, even better than the old ways of development that we were following; it saved us time, reached the target faster than before, can start working even before you gather all the requirements and also you can work on parallel with business and in development and coding.

Justification14: We need to know the details of problems faced during the transition. This question complies with the objectives 1, 2, 6 and 10.

Question15: If you were asked to enhance agile, what would you do?

Answer15: As I told you already, the problems that we have faced during the development with agile weren't really related into the way of methodology, but they were related into the business and clients. May be if

we thought of a better way of information gathering and merged it with agile can raise agile into the first line of methodologies among all others.

Justification15: We need a methodology fit as possible in the local market, even if we had to criticize agile, the method that we are trying to implement and adapt. This question complies with the objectives 1, 2, 6 and 10.

Interview 5:

Mr. Mohammad Nassar, SW Developer at a local company

Question1: did you work on traditional (i.e. waterfall) and newer software methodologies (i.e. Agile)?

Answer1: Yes I really did. The local software community depends on the requirements collection, analysis, design, documentation and testing, then comes the deployment. So if you want to make a system for a hospital, you need to see what and how the hospital works, then to analyze what to do for about another 6 months, and if a mistake was found then another time will be wasted, while when you deploy the project after you finish the development then another time will also be wasted. On the contrary, when I worked on agile on a same project, the work was divided into modules where we can focus on those modules, and those modules also divided into sub-modules and so on; those modules are being developed per 2-weeks sprints. In those sprints you have a clear vision of what you have to do and how to do it, besides you have more control over those stages of development in terms of time and resource utilization. I just wanted to say that it is really truly beautiful to follow agile, but I don't think that applying agile on the teams of support that work based on daily operation is not suitable in this case.

Justification1: We need to know the development back ground that the interviewee against us has until now so that we ensure that we are building our research on a strong surroundings. This question complies with the objectives 1, 9 and 10.

Question2: what do you prefer to work on, agile or waterfall?

Answer2: I think the project style is what determines what method to follow in our development; if the system being developed is a new feature or addition or a new system then agile scrum would be more appropriate, while if you have a team that have a good knowledge of team work, a clear product owner and the project manager is available within the available hierarchy and utilization you have, then we have to follow agile scrum, else there is no need for scrum.

Justification2: We are interested in the personal opinion for the interviewee against us so that we can get as much information as possible in the way he has been working on for some time. This question complies with the objectives 1, 2, 4 and 5.

Question3: Based on your experience, what do you think it is good to add into scrum? And what to remove? what is best to exist in scrum and what is not to be there?

Answer3: Well yes, I have a note about the scrum master; when we work on a development then I prefer the scrum master to be technical leader instead of only being a scrum master; he needs to be a technical leader to solve all the problems that he might face during the process of development in addition to be like a site managing. This one thing that I think it is not applied in scrum and felt that it is a must in the development process.

Justification3: We need a methodology fit as possible in the local market, even if we had to criticize agile, the method that we are trying to implement and adapt. This question complies with the objectives 1, 2, 6 and 10.

Question4: How much –in percentage- do you estimate your application for agile?

Answer4: I have work on fife project in my work life, and the last three projects were on agile; I guess I can rate them on the 80% and above were based on agile scrum.

Justification4: We need a methodology fit as possible in the local market, even if we had to criticize agile, the method that we are trying to implement and adapt. This question complies with the objectives 1, 2, 6 and 10.

Question5: In your opinion, how can you trade-off between agile and traditional methodologies?

Answer5: well, let's begin from the time management. I am convinced that scrum is better in this term, since we usually finish the planned system within allowed time limits. On the other hand, the risks tied with agile are less than when faced in traditional methodologies. So we are taking about a better time management that a better risk management comes with it; i.e. like efforts and costs are better handling, but although I didn't work a lot in the traditional methodologies, but I have never faced a project that ran smoothly without crossing the limits of resources into high limits.

Justification5: We are trying to compare between traditional and agile so that we get a feedback about what is best for the local market SW projects. This question complies with the objectives 1, 2, 6 and 10.

Question6: If I took you as an example and asked you if you ever worked on a waterfall project and didn't get the success you need, so you switched into agile and worked successfully, have you ever faced such a situation?
Answer6: No not really, but to be honest we have faced situation where we stuck at a failed project and didn't managed to get out of it, but since agile is something very new into this locale, we didn't have to use it unlike now.
Justification6: We are trying to compare between traditional and agile so that we get a feedback about what is best for the local market SW projects. This question complies with the objectives 1, 2, 6 and 10.

Question7: On the waterfall projects that you'd worked on, can you tell me a brief idea about their success or failures and why?
Answer7: In general, I have worked on about two projects with agile, and they were all successful, but we had a lot of problems during development that didn't kill the project or make it fail; and for the worst scenario we had was that to load the half of sprint into the next sprint. On the other hand, on the waterfall I have worked on three projects that mainly were using the waterfall development, and we had failure stories where we used to work through all the five life cycle phases, and we had a lot of trouble during each phase but in the end we've been forced to change the development from the start which caused the company a lot of costs and losses.
Justification7: We are trying to compare between traditional and agile so that we get a feedback about what is best for the local market SW projects. This question complies with the objectives 1, 2, 6 and 10.

Question8: Did you have to work over time through agile scrum?

Answer8: Yes indeed we had a lot of over time even more than the waterfall since you have a focused work that has to be finished during the sprint that you already committed to finish your work within it.

Justification8: One of the purposes of this research is to seek the ability to mitigate the time and cost of development in the SW projects for the local market. This question complies with the objective 3.

Question9: How did you find the acceptance of the customers to the idea of using agile scrum instead of the well known traditional methodologies in the SW project you've implementing in the local market?

Answer9: Well, mainly it depends on the customers themselves and their wish to be involved in the project cycle; since we have faced customers that truly wished to be involved in the project from the first day, while some other customers weren't mean a lot to be involved in the development cycle.

Justification9: It is very important to address the customer acceptance as a primary factor for our reasoning in the research, so we need to know all we can in order we can move forward in the methodologies followed within our SW projects. This question complies with the objectives 7, 8 and 9.

Question10: In general, what do you prefer to work on? I mean if you were in charge of developing a middle SW project, what would you prefer to follow in the development methodology, agile scrum or traditional methodologies?

Answer10: Definitely I will use scrum.

Justification10: We are trying to compare between traditional and agile so that we get a feedback about what is best for the local market SW projects. This question complies with the objectives 1, 2, 6 and 10.

Question11: How was the company accepting the idea of following a new methodology in SW development? And how were the team members doing within scrum with each other and with the company?

Answer11: I think the company prefers and urges the teams to follow scrum, but for the team members it became like something routine or something we have to do everyday only as a procedure, and that where our point of complain resides. You see that every day we have to hold a meeting the first thing in the morning and that meeting may extend into a bigger meeting and so on. On the other hand, the pairs like senior and junior didn't have a trouble since one is working as a guide for the other and so on.

Justification11: Every change has its own supporters and objectors, and in our case if we focused on the objections that we would face against changing the methodology, the team and the firm would be the main protesters –or supporters- and we want to validate their reactions. This question complies with the objective 4.

Question12: Can you tell us about the quality resulted from scrum in the project you been involved in?

Answer12: The quality and quantity were really better. I mean by that, the productivity, performance and efficiency were all at better levels compared to the traditional methods.

Justification12: One of our goals is to get more quality if we followed something other than the current followed methodology. This question complies with the objective 5.

Interview 6:

Mr. Ahmad Othman, SW Developer at a local company

Question1: How many years have you been working in the field of IT programming?

Answer1: It has been about 4 years.

Justification1: We need to know the development back ground that the interviewee against us has until now so that we ensure that we are building our research on a strong surroundings. This question complies with the objectives 9 and 10.

Question2: And can you tell me what SW methodology you have using in your work?

Answer2: I am not sure what you mean exactly by methodology; is it the business way with development?

Justification2: We need to know the development back ground that the interviewee against us has until now so that we ensure that we are building our research on a strong surroundings. This question complies with the objectives 9 and 10.

Question3: I mean by the SW methodology is that the way you fit in your procedures that you use during your development for the software project; like a methodology called the traditional one where as known for SW industry it starts with the documentation analysis then the design then the development then the testing then deployment and so on.

Answer3: Yes I understand what you mean by the traditional one.

Justification3: This question is a clarification for the later one and has and the same goals as before.

Question4: And also we have other methodologies like agile where the way of development differs from the traditional one.

Answer4: Yes I understand. We actually only using the traditional SW methodology and we had many issues relating to requirements gathering and business understanding.

Question5: Can you please brief the issue that you face during the requirement gathering?

Answer5: Honestly we used to face a lot of change requests regarding the development, this happens since we don't work on clear modules, so sometimes you suddenly find that many changes have happened on the DB level or on the GUI level; these changes wouldn't be expected in the requirements or the client asks more to do after the start of the development which causes us a lot of changes in requests. In addition to this, a lot of requirements usually are not clear to the client neither to us, so in that case we usually start with what the client understands until now, and we will be in a case where we have to build some of the system based on what we have, but the problem is that when the customer comes saying that he wants to change what we already built, which also causes our time to be wasted.

Justification4: We need to know the development back ground that the interviewee against us has until now so that we ensure that we are building our research on a strong surroundings. This question complies with the objectives 9 and 10.

Question6: So I understand that the clients don't understand their problem or what they need for their system?

Answer6: Yes. From what I see, the requirements are always –or mostly- are not enough or not understood enough.

Justification5: The clients are one primary aspect of interest that our research focuses on; we are interested in the level of understanding that the customers have toward what they need vs. what they want, and in the level of understanding of what the IT firms can do and accomplish for what they ask. This question complies with the objective 7.

Question7: Ok, then I need to ask you a question about your methodology, are you satisfied with the way you are working with or you still searching for something better suits what you are thinking in?

Answer7: In my opinion, if the projects size were big, then our methodology is good and suitable, but if they were middle or small then no need to follow such a methodology.

Justification6: We need to know the development back ground that the interviewee against us has until now so that we ensure that we are building our research on a strong surroundings. This question complies with the objectives 6 and 10.

Question8: Ok, then if you have many issues regarding to the customers understanding and requirements, have you ever tried to search for a new methodology to follow, i.e. letting the customer in the process from the beginnings into advanced phases? or like to start the development process in parallel with the requirements gathering?

Answer8: We usually tried to do something like that; we tried to make the customer involved in the process and keep him in, but usually the conditions prevent us from doing so, or we want him to be in but we can't

find the necessary commitment from his side, or a lot of latency usually being found from his side.

Justification7: We need to know the development back ground that the interviewee against us has until now so that we ensure that we are building our research on a strong surroundings. This question complies with the objectives 1, 6, 7, 8 and10.

Question9: Then in your opinion, how do you think you can make some progress and enhancements on the methodology that you are working on?

Answer9: In my opinion I think the better way to make everything easier is to divide it into controllable modules; like dividing the DB into modules, GUI into modules, programming objects into modules, following the SW engineering practices since the programmer or develop can't do anything right without following those practices.

Justification8: We need to know the development back ground that the interviewee against us has until now so that we ensure that we are building our research on a strong surroundings. This question complies with the objectives 1, 2, 3, 5 and10.

Question10: The projects that you had worked on or participated in – knowing that your company follows the traditional methodology in their development-, can you please tell me the success rates in those projects?

Answer10: To be honest, not all projects were success, few of them were total failure and most of them were success but not as expected or as we wish. The main obstacle we were stocked by was in the testing phase, were a lot of the testing results were refused so the system will be back into development again and so on, which literally we can say it is a time wastage. On the other hand, sometimes the problems appear in the

production –or deployment on the client's side- but it usually being a development problems rather than application methodology problem.

Justification9: We are trying to evaluate the current methodologies that the local firms follow in order to put them in the track of judging them after comparison with other methodologies like agile. This question complies with the objectives 6 and 9.

Question11: Can you tell me how many projects have you ever worked in or participated in during your work life? And how do you evaluate the resulting products?

Answer11: I was involved totally in three projects in my four years work. And for the evaluation, I can say that 80% of them were a success, and to be honest, it is not system bugs as much as misunderstanding of the system functionalities or requirements.

Justification10: We need to know the development back ground that the interviewee against us has until now so that we ensure that we are building our research on a strong surroundings. Besides, we also want to know how the interviewee's firm is doing their testing results in order to check its compatibility with agile. This question complies with the objectives 2, 5, 9 and 10.

Question12: Have you ever used the automation testing into your projects?

Answer12: I really don't know much about the testing, but as far as I know, that there are rules for testing that are being used by the testing team, like retrieving data speed, system performance, data tuning, processing requests and so on.

Justification11: We also want to know how the interviewee's firm is doing their testing results in order to check its compatibility with agile. This question complies with the objectives 1, 2 and 5.

Interview 7:

Mr. Waseem Azzam, SW Developer at a local company

Question1: As a start, this interview is about the SW development methodologies applied in the local market, and when I say SW development methodologies, I mean the ways you fit in your procedures that you use during your development for the software project; like a methodology called the traditional one where as known for SW industry it starts with the documentation analysis then the design then the development then the testing then deployment and so on. And there are other methodologies like agile. Now after this explanation, can you please determine what kind of methodologies you usually follow in your company?

Answer1: Ok. We usually start with the requirements gathering from the customer side, and after this requirements, we start with the analysis phase; some of this analysis usually being with the customer directly contacting them, and some of it with our team internally.

Justification1: We need to know the development back ground that the interviewee against us has until now so that we ensure that we are building our research on a strong surroundings. This question complies with the objectives 9 and 10.

Question2: In your work life, in how many project have you participated or was involved in?

Answer2: I have participated in my work life for total of about 10 projects, whether partially participating or fully involved.

Justification2: We need to know the development back ground that the interviewee against us has until now so that we ensure that we are building our research on a strong surroundings. This question complies with the objectives 9 and 10.

Question3: And all the development methodologies were the same?

Answer3: Actually no. Some of the projects you are not being involved from the start, so you have to work with the teams as they currently are, some other projects you find yourself in the development phase while you still working on the analysis phase, so it depends, but I think that the best thing is for the analysis to be done completely, so that you give the developer side what to work in the best way, but as you know a lot of times you can't find time to completely analyze the situation, so you would be forced to pass it into the development phase before you finish the analysis, and when you got deep involved in the development, you will found new things that you don't know about, and so on.

Justification3: We need to know the development back ground that the interviewee against us has until now so that we ensure that we are building our research on a strong surroundings. This question complies with the objectives 9 and 10.

Question4: In your opinion, what is the success rate of these projects that you've worked or participated in?

Answer4: At the end, the customer must be satisfied. Sometimes, within a rate not more than 20% of whole projects, we finish the project within the time limit with the required functionality, but most of the time, we don't

catch the available time limits, which is reflected by an overhead cost because of that latency.

Justification4: We are trying to compare between traditional and agile so that we get a feedback about what is best for the local market SW projects. This question complies with the objectives 2, 6 and 10.

Question5: How can you define the customer satisfaction? And what is the degree of customer involvement in the process of development?

Answer5: Usually the customer determines the requirement he needs through the first phase of the project within the UAT (User Acceptance Test), and after that phase, the customer may review those requirements and accept them, and after some period, even after the development, if the customer found something new to add, then we have to respond to his queries and add them into requirements and append them into the development. **Interviewer:** And do you think that when you finished the requirements and agree with the customer about them, and after the whole work that you have done, and then the customer says that it is not good and still missing, do you think that this is an efficient way of working? **Interviewee**: It can't be helped; the customer doesn't know what he wants, and sometimes you don't know what the customer means until you start the development. Besides, some customer requests are chargeable within the contract. In any case, most of the time the customer doesn't like what had been done until now.

Justification5: We are trying to get the different definitions for the customer satisfaction within some local firms in order to determine the aspects of enhancement regarding the customers within the methodology being followed. This question complies with the objectives 7 and 8.

Question6: So as you said, the successful projects finished within time and budget are about 20% rate, what reasons do you think behind this success?

Answer6: May be because the requirements are less or the customer is more cooperative and involved in the process which make him more able to understand the way of work we are going through. I can say that many projects that fail to finish during the time limits are because the customer keeps asking for more changes during the process or because the customer doesn't understand exactly what to ask or how to ask. Another cause for failure is that because the resource is tied with more than one project which causes him to be distracted.

Justification6: We are trying to introduce new ways of work to ensure the higher success rate within the local territories, so we try to gather the good trials and tries by the local firms in order to unify it as a procedure. This question complies with the objectives 5, 6, 9 and 10.

Question7: Did you think with changing the way of development you are following? Like making the customers more involved? Or changing the way of programming? Or changing the way of dealing with contracts? Or changing the management behavior toward the developers or something like that?

Answer7: Well in general, if the righteous management was found on a project, adding the righteous distribution of tasks over the staff, if those two facts exist in a project, then the project would eventually be a success story. **Interviewer**: so you are not thinking in changing the way of development that you follow? **Interviewee**: no I don't think there is a need for that, since I guess the previous two facts are enough for the project to succeed; you just need the follow the requirements, and make sure that the analysis is correct.

Justification7: We need to know what the obstacles that are preventing the local firms from fixing or changing the way if development when it finds that there are some issues regarding its development procedures. This question complies with the objectives 5, 6, 9 and 10.

Question8: In your opinion, what can be added and done in order to step up with the SW development methodologies, at least for the one that you follow?

Answer8: No not really.

Justification8: We need to know what the obstacles that are preventing the local firms from fixing or changing the way if development when it finds that there are some issues regarding its development procedures. Also we are trying to find what we can do to upgrade any procedure that might have any shortcomings regarding the development process. This question complies with the objectives 5, 6, 9 and 10.

جامعة النجاح الوطنية
كلية الدراسات العليا

إدخال طريقة تطوير البرمجيات المسماة (أجايل سكرم) إلى مشروع تطوير برمجيات في شركة محلية

إعداد
أدهم "محمد وصفي" حنون

إشراف
د. بكر عبدالحق

قدمت هذه الأطروحة استكمالا لمتطلبات درجة الماجستير في الإدارة الهندسية بكلية الدراسات العليا في جامعة النجاح الوطنية في نابلس، فلسطين.
2013

إدخال طريقة تطوير البرمجيات المسماة (أجايل سكرم) إلى
مشروع تطوير برمجيات في شركة محلية

إعداد
أدهم "محمد وصفي" حنون
إشراف
د. بكر عبدالحق

الملخص

الأجايل (Agile) هي طريقة لتطوير البرمجيات وهي وسيلة جديدة بالمقارنة مع الطرق التقليدية مثل الشلال (Waterfall) المستخدمة في السوق المحلية في عملية إدارة البرمجيات بشكل عام. الأجايل لديها العديد من الأطر المستخدمة ضمن رؤيتها مثل ال XP و سكروم (Scrum). لقد استخدم الباحث في هذه الدراسة سكروم لبناء بحثه، و قد حاول المفاضلة بين اثنتين من الطرق المختلفة بطريقة عملية من خلال دراسة حالة أجريت على شركة محلية، وأيد ذلك بالعديد من المعلومات المستقاة من المقابلات و الملاحظات لجعل الدراسة متوفرة أكثر للتحليل.

يهدف هذا البحث إلى تحقيق تقدم ملحوظ في عملية تطوير مشاريع البرمجيات من أكثر من وجهة نظر، من خلال المقابلات مع مديري المشاريع والمطورين ومهندسي المشاريع، و عمل مراقبة لكيفية اعتماد و تطبيق السكروم في هذه الشركة المحلية. لقد استخدم الباحث المنهج النوعي لإتمام هذه العملية البحثية، حيث تمت دراسة التقدم في المشروع من خلال المعلومات التي تم تجميعها من خلال المصادر المختلفة، وعمل مقارنة تطبيق السكروم في هذا المشروع مع مشروع آخر له نفس الخصائص ولكن تم تطويره سابقا باستخدام الطرق التقليدية القديمة.

توصل البحث إلى العديد من النتائج والاستنتاجات، مثل أن هناك بعض الاختلافات الطفيفة بين سكروم بشكل عام وسكروم المطبقة في هذا البحث، وأن هناك بعض الاختلافات في تطبيق سكروم في العديد من الشركات في فلسطين بالمقارنة مع الطريقة التقليدية لطرق تطوير استخدمت لسنوات أيضا في فلسطين. وكان هناك أيضا فرق بين كيف يفسر سكروم في

المصادر العلمية الختلفة وكيف نفذ البحث في الشركة المحلية بعد أخذ التغذية الراجعة من الشركات المحلية الأخرى لطرق تطبيقها للمنهجيات التقليدية والحديثة.

توصل الباحث إلي إلزامية القيام ببعض التغييرات والإضافات لاستخدام السكروم كما هو عليه الآن، كما يجب أن تكون سياسة تطوير الجودة حاضرة بقوة ويجب تحسينها باستمرار على أساس التغذية الراجعة. كما أن هذه الدراسة وجدت أن العميل أو الزبون له دور كبير في نجاح المشروع إذا التزم بما يمليه عليه دوره أن يفعل.

يعتقد الباحث أن هذا البحث يقدم مساهمة و إضافة جيدة للسوق المحلية وللشركة المحلية الذي اعتمدت البحث. هذا البحث هو محاولة لطرح أسئلة عن سبب المشاكل في هذا السوق المحلية و كيفية حدوثها وكيفية حل هذه المشاكل. يعتقد الباحث أن هذا البحث يقدم شيئا جديدا لم يبحث كثيرا في السوق المحلية، و للحصول على أفضل النتائج يجب الحصول على نتيجة تعمم من هذا البحث و تدمج مع غيرها من الأبحاث، ثم يجب أن تحل على الأقل جزء من المشاكل باستنتاجات هذا البحث، و بالتالي التقليل من معاناة زبائن تكنولوجيا المعلومات وشركات تطوير البرمجيات، والذي يفتح الباب لمزيد من الإبداع في هذا المجال. بعد الانتهاء من دراسة الحالة، استكشف الباحث وجهات نظر أعضاء السكروم المشاركين في المشروع، الذين ذكروا أن الكثير من المشاكل التي وجدت في المشاريع السابقة تم حلها وشعروا بتحسن كبير في عملهم، إلى جانب الأشياء الجديدة التي تعلموها من خلال هذه الطريقة الجديدة في العمل الذي لم يكونوا على علم بها. كما قدم هذا البحث مساهمة كبيرة للشركة المحلية التي اعتمدت البحوث، و أعطى تقييما واضحا للطريقة الحالية لإجراءات العمل الحالية، وفهم واضح لمبادئ تطوير الاتصالات مع العملاء، وقبول الفحوص، والتكيف مع التغييرات الطلوبة خلال مدة المشروع.

واستنادا إلى نتائج البحوث، ينبغي أن تقوم الشركات المحلية بتنفيذ المزيد من الدراسات والأبحاث لتحديد أفضل منهجيات تطوير البرمجيات التي تناسب بيئة أعمالهم وتناسب السوق الفلسطيني. وعلاوة على ذلك، ينبغي أن تقوم الشركات بالتركيز أكثر على العملاء، كما ينبغي أن تكون هناك دراسات وبحوث للتمييز بين حالات المشاريع الضخمة والمتوسطة والصغيرة والتأكد من ماهية الطريقة الأفضل والتي تعود بأكبر قيمة تجارية.