Study and analysis of the challenges, and guidelines of transitioning from waterfall development model to Scrum
This thesis is submitted to the School of Engineering at Blekinge Institute of Technology in partial fulfillment of the requirements for the degree of Master of Science in Software Engineering. The thesis is equivalent to 20 weeks of full time studies.

Contact Information:
Author(s):
Junaid Naseem
Wasim Tahir
E-mail: juna06@student.bth.se
E-mail: wata06@student.bth.se

University advisor(s):
Dr. Bengt Carlsson
School of Engineering

School of Engineering
Blekinge Institute of Technology
Box 520
SE - 372 25 Ronneby
Sweden

Internet: www.bth.se/tek
Phone: +46 457 38 50 00
Fax: +46 457 271 25
ABSTRACT

Software engineering practices have experienced significant changes over the period of past two decades. Keeping in view the competitive market trends, now is the high time for many organizations to shift from traditional waterfall models to more agile technologies like Scrum [22][23]. A change of this magnitude is often not easy to undertake. The reason that both software engineering techniques are different in many respects, organizations require considerable amount of analysis of the whole transitioning process and possible scenarios that may occur along the way. Small and medium organizations are normally very skeptical to the change of this magnitude. The scale of change is not limited to only software processes, in fact, difficult part is to deal with old attitudes and thinking processes and mold them for the new agile based Scrum development.

The process of change therefore need to be understood in the first place and then carefully forwarded to the implementation phase

Keywords: migration/switching from traditional to agile software development, Scrum, Agile, waterfall, hybrid research methodology, agile adoption challenges.
ACKNOWLEDGEMENTS

We are thankful to Allah for giving us the courage and ability to finish this thesis. Without His blessings we would not have been able to finish this research work.

We are extremely thankful to our advisor Mr. Bengt Carlsson who has been a motivating factor for us and his expert opinion throughout our research were guiding force for us. He has been extremely patient during all this time and provided invaluable feedback and support during every step of this thesis work. We would also like to thank Mr. Shahid Mujtaba and Mr. Nadeem Mughal of Ericsson who gave us their precious time and helped us in getting responses from Ericsson and also arranged an interview meeting for our research.

Lastly we would thank our families who were with us during our studies and were the key source of motivation.
Table of contents

1.1. BACKGROUND ............................................................................................................ 6
1.2. RESEARCH QUESTIONS AND METHODOLOGY ......................................................... 7
1.3. SUITABILITY OF HYBRID METHODOLOGY FOR OUR THESIS ............................... 8

2. SOFTWARE DEVELOPMENT MODELS ........................................................................ 10
   2.1. INCREMENTAL DEVELOPMENT ........................................................................... 11
   2.2. XP (EXTREME PROGRAMMING) ......................................................................... 11
   2.3. SPIRAL MODEL .................................................................................................. 11
   2.4. RAPID APPLICATION DEVELOPMENT ............................................................ 11
   2.5. WATERFALL MODEL ......................................................................................... 12
   2.6. SCRUM ............................................................................................................. 13

3. SCRUM COMPARISON WITH WATERFALL MODEL .................................................. 18

4. CHALLENGES FOR SCRUM ....................................................................................... 21

5. QUESTIONNAIRE ....................................................................................................... 23

6. ANALYSIS .................................................................................................................. 25
   6.1. ARITHMETIC MEAN AND STANDARD DEVIATION ............................................. 26
   6.2. ANALYSIS OF TABLE 6.1 .................................................................................. 26
   6.3. ANALYSIS ON LIKERT SCALE DISTRIBUTION OF DATA ............................... 28
   6.4. SUBGROUP CREATION ...................................................................................... 33
   6.5. DESIGNER ....................................................................................................... 34
   6.6. DEVELOPER ..................................................................................................... 35
   6.7. OTHER JOB FUNCTIONS .................................................................................. 36
   6.8. SCRUM DEVELOPMENT MODEL .................................................................... 37
   6.9. OTHER THEN SCRUM DEVELOPMENT MODEL ............................................. 38
   6.10. LESS THEN 2 YEAR OF TRADITIONAL DEVELOPMENT MODEL EXPERIENCE ... 39
   6.11. MORE THEN OR EQUAL TO 2 YEAR OF TRADITIONAL DEVELOPMENT MODEL EXPERIENCE ........................................................................ 40
   6.12. LESS THEN OR EQUAL TO 1 YEAR OF AGILE DEVELOPMENT MODEL EXPERIENCE ......................................................................................... 41
   6.13. MORE THEN 1 YEAR OF AGILE DEVELOPMENT MODEL EXPERIENCE ............ 42
   6.14. GREATER THEN OR EQUAL TO OF 2 YEARS TRADITIONAL AND AGILE EXPERIENCE ............................................................................ 43

7. GUIDELINES FOR AGILE ADOPTION ..................................................................... 44

8. DISCUSSION .............................................................................................................. 48
   8.2.1. SAMPLE SIZE / NUMBER OF RESPONSES ............................................... 48
   8.2.2. CONTENT/ DATA VALIDITY ......................................................................... 48
   8.2.3. SUBJECT’S PROFILE .................................................................................... 48
   8.2.4. SUBJECT’S BACKGROUND ......................................................................... 49

9. CONCLUSION .............................................................................................................. 50

10. REFERENCES ............................................................................................................ 51

11. APPENDIX 1: QUESTIONNAIRE ............................................................................. 53
The history of Software Engineering has experienced many changes and evolved gradually to develop and enhance the software technology and its productivity. Over the years number of models for software development are introduced and applied. The same way as other engineering disciplines require discrete steps to develop an engineering artifact, software engineering models have detailed phases and defined tasks to carry out the development process. These development models are of various types that are suitable for achieving good results for various types of projects and products.

In today’s market changes are very frequent. Often due to change of business domain or need of current market leads to change of software development methodology. In case the change is a big shift of curve then organizations may face many challenges because employees working with a particular software methodology for some time may not accept or adopt the new changes very smoothly [7].

Our study is to investigate organizational impact of change in software development model. The theme of our thesis is to focus on shift from traditional waterfall software development model to Agile based SCRUM development model and highlight the difference between two approaches, challenges faced during the change and to suggest ways to minimize those challenges.

In first section of this chapter we briefly describe software development models following elaboration of classic life-cycle model, waterfall and SCRUM development models in next section which will be followed by a concise comparison of both development models.

In the late sections, discussion is carried out about the outcomes of an online questionnaire which are received from various professionals working in the related development environments of software industry. In last section of this document, detailed discussion, suggestions and conclusions are drawn.

1.1. Background

The traditional Software Development Model has served software industry for more then two decades and still it is popular in many software organizations around the globe where requirements change is not an issue.

Many software specialists feel comfortable while following Traditional model because of its step by step approach toward development life cycle. On the other hand, companies going for Agile Base developments are also increasing.
Adaptation of agile based methods has a number of reasons including requirements change, tight time deadlines and fear of over budgeting.

An Agile based methodology named Scrum has recently become the favorite choice of organizations for their software development. People at software organizations find it really hard to switch from conventional Traditional methods to Scrum because of the diverse change approaches of both the methodologies.

1.2. Research Questions and Methodology

Following are the three main questions which we will try to answer in our thesis report.

- What are the major differences in approaches between Waterfall and SCRUM development models?

- What are the challenges faced while transitioning from Waterfall development model to Scrum?

- How the transition of Waterfall model to SCRUM based development can be achieved? What are the guidelines?

Research methods are important in the sense that they minimize the validity threats faced by research outcome. There are three methods of research carried in the field of science and technology

- Qualitative
- Quantitative
- Mixed/Hybrid

Qualitative research relies upon in depth study of documents and material and interviews for information gathering and verification. In this method, a comparatively smaller sample can be chosen but its relativity (relation/specific to the topic) to the research topic in important. Case studies play a big part in qualitative research.

Quantitative research differs from qualitative in the sense that it deals with comparatively bigger sample and measurements are obtained by using statistics. It is widely used in social sciences.

Mixed or hybrid methodology is an approach which uses both the qualitative and quantitative approaches acquisition and analysis of data.

For this report we will follow hybrid methodology.
1.3. Suitability of hybrid methodology for our thesis

The process of investigating challenges of transition from waterfall to scrum is quite a demanding task and hence requires a very comprehensive research methodology. For a balanced approach to capture and analyze data for our thesis work, we opted for a hybrid research methodology. We can find lot of work in literature that is somehow similar to this very report but all of that work is based on literature review and assumptions. There is no comprehensive empirical study on ‘analysis and guidelines regarding scrum adoption’. This was biggest motivation for us to go for hybrid methodology because we want to fill this gap by conducting an empirical study. Our empirical study is based on a questionnaire. This questionnaire is distributed to targeted subjects who have exposure of both waterfall and Scrum development models.

A comprehensive study of literature related to waterfall and Scrum based development is conducted to get in depth understanding of both approaches. We have selected peer-reviewed research papers, books, and arguably credible websites for literature review. The documents that we had selected for our study have focus over the nature of two methodologies, their advantages and disadvantages, suitability, differences between them and/or issues regarding transitional phases from waterfall to scrum.

On the basis of this literature study we have composed an online questionnaire and sent it to targeted subjects. The questionnaire has been designed in a way that it gives us good ground for analysis and also we have tried to minimize data validity threat by randomizing positive and negative choices in answers,
throughout, our questionnaire. Along with an online questionnaire, we also conducted in depth interview with IT professionals to strengthen analysis part of this report. Interview has been conducted with professionals in IT sector whose field will be closely linked to our research area. In the final stages of our thesis work, we have analyzed the outcomes of online survey and interviews to formulate guidelines for industry.

Question number 1 is answered in chapter 3, question number 2 is answered in chapter 4 while question number 4 is answered in chapter 7.

Figure 1.2 The research roadmap
2. Software development models

It is very important to know that why we need software development model at first place. About three decades back software products were relatively simpler with very low complexity and much lesser Line of Code (LOC). At that time there was not as such need of a structured software models to write an in-time, quality software [9]. But since then software product has evolved a lot and now software is a very complex product which is highly complex, with interdependent functions, and Line of Code in hundreds of thousands. Like other engineering products software are also customer focus so they must be engineered carefully under guidelines of suitable models to achieve customer satisfaction at minimum as customer satisfaction was motto of software companies in 80’s, in this competitive world now is the time of customer delight [8][9].

A Software development model is an arrangement of different activities in development of a software product. Some of the activities that can be part of typical software model are Domain Analysis, Software Elements Analysis, Scope Analysis, Specification, Architecture, Implementation, testing, Deployment and Documentation [8].

In a broader sense we can distinguish software methodologies in two categories, the traditional plan-driven methodologies and modern agile methodologies [5]. Example of traditional methodologies can be waterfall and spiral models while some of the agile based methodologies are extreme programming also known as XP and Scrum. Traditional software development approach is more leaning towards the process but at the other end modern agile methodologies have focused on people and their creative talent [10] [11].

Due to high degree of difference between both types of methodologies it is not easy for organizations to migrate from traditional methods to agile as we have to scan multiple characteristics of an organization like organizational structure, its culture, and communication channels in order to implement the new changed practices. If all those factors are not analyzed properly transition from traditional to agile can pose serious challenges [5]. To make it more specific, we have focused our attention over two development models. Waterfall model is taken from traditional development models while Scrum is chosen from the agile methods. Both these models have great significance and are widely used in the industry. In coming section, various models from traditional and agile based developments will be discussed in general while Waterfall and Scrum will be explored in details before proceeding with potential challenges that an organization can face while migrating from Waterfall to Scrum methodologies.
Some important development models are discussed below,

2.1. Incremental Development

As pretty much clear from its name, the software is development in increments. In this method, enhancements are made to the system in increments that is basic functionality is achieved first and then additions are made periodically. Normally iterative and incremental development is used simultaneously. This method is also borrowed in many agile based development models such as XP and Scrum. [8][14]

2.2. XP (Extreme Programming)

XP is another agile based development model which has been introduced by Kent Beck in 1996 and matured as times passed by. Extreme Programming has gain significant acceptance over the years in software industry. In this methodology, changes are expected to happen to requirements and development teams are ready to cope with those changes and that makes it agile. Coding and unit testing are done by the programmers. [8][14][16]

2.3. Spiral Model

It is risk oriented model where emphases is made over risk assessments and steps are taken to minimize those risks to the projects. In spiral model, a project is broken into many sub parts where changes can be easily made to those segments of the project. This breaking of project into small segments helps in assessing those risks and to figure them out before their occurrence in the life cycle.

Its four basic principles help in analyzing and then elimination of any risks that can cause damage to the system during software development process. [8][16]

2.4. Rapid Application Development

Rapid application development is also an iterative type of development methodology. It is a mergence of different techniques particularly prototyping techniques and data driven information engineering. Rapid application development uses these techniques to characterize user’s requirements and final system design. [8][15]

Rapid application development was developed in a need, to respond, to the fast delivery requirement. Switching of development from traditional session based client-server to session-less open development (like Web 2.0) has increased the need for faster iterations throughout software development life-cycle. [15][17]
2.5. Waterfall model

Waterfall model is process and document centric with a belief that all the necessary requirements needed to complete a software product are known and no major change is expected during software development process. Customer is involved during analysis phase of project but after that there is no such role of customer in rest of development process [5] [8].

Majority in software engineering area will agree that best known and oldest software development process is waterfall development model. Waterfall is sequential software development where development process flows downwards like a waterfall in different phases mainly starting from requirement elicitation, analysis and then design, implementation, testing, and deployment maintenance [8].

Requirements consist of such statements that describe overall deeds of system required by customer and/or users. Analysis provides such quantifiable system constraints that help in approximation of projects progress throughout software development process. Analysis also provides acceptance testing criteria. Implementation is conversion of design into program code using a programming language. Testing is verification and validation of software product.

In waterfall model one phase must be completed first before going on next phase and a review is conducted after every phase to ensure that everything is complete before proceeding to next phase. Like starting from requirement analysis, requirements for whole project must be well-defined and testable before going to next phase, which is design. This also means that all the requirements must be clear before finalizing requirements analysis and starting design phase. In design phase a sketch for implementation is devised following implementation, if the components are developed by different teams or individuals then those modules are integrated, tested and software product is deployed. A classical waterfall model discourages backward and onward jump neither an overlap between phases [8].

Waterfall model also encourages documentation and because of that knowledge, in organizations, using waterfall of other such traditional life-cycle models becomes explicit. Also waterfall model gives more stress on specialized roles rather frequent rotation of roles as it increases expertise of employees in particular area, indirectly resulting in better quality of product. [14][15]

Waterfall model in practice

Different surveys have shown that agile methodologies are gaining popularity and by every coming day more and more organizations are adapting agile methodologies [5] [11] but no matter how much popular those agile
methodologies are going to be, there will be still such organizations that will enjoy working with traditional methodologies like waterfall or others customized forms of it.

Waterfall model is a widely used software development model, preferred in most of the big software development companies and sensitive organizations like U.S. Department of defense and NASA. In critical and real-time systems there is need of discipline in development which is imposed by waterfall model. In complex systems improved quality is expected by using waterfall model because of its stress on complete analysis and design along with through documentation before writing a single line of code. [8][12][14]

In current time if still there is a project where requirements are well defined and no major modification is expected then waterfall model can be the best choice. As in such case a decent estimation can be made for all the required resources of project and everything can be planned for the project with a road-map to high quality product. [14][15]

Waterfall model can be ultimate choice if there is need to build maintenance release of a well-defined product or while porting an existing product to another platform [12].

2.6. SCRUM

In these days, development teams face enormous difficulties in coping with change in customer demands. Besides requirements change, the adoption of new technologies also hurdles when it comes to traditional ways of developments. Higher level of competition in the software market has increased the desire for development models that can adopt and adjust with changes easily. Waterfall model lack this feature due to its subtle nature regarding requirements freezing during the whole development process. Keeping the changing nature of market, organizations have started to adapt to the new dynamic, changing and evolving nature of development methodologies commonly known as Agile based development. [1][3][4]

Agile does not represent any specific technique rather it’s the name of a concept of development software systems. As name agile suggest, moving quick and lightly, is the overall idea. The emphasis in this concept is to cope with changing market demands and requirements and tackle them with efficiency.[1][6]

Scrum is an agile based development technique. The term is taken from rugby which describes that ‘to bring back an out of play into the game’. This first mention of Scrum can be traced to 1986 by Takeuchi and Nonaka where they presented a fast, adaptive and self-organizing software development process.
Later Schwaber and Beedle worked on it and published their combined work in 2002 over Scrum. [3][4][6]

The basic theme of this methodology is that system development relies upon many variables. These variables can be environmental or technical such as unstable requirements, time deadlines, resources and developing technologies. In spite of all precautions, these variables change. Scrum is aimed to provide such flexibility in development process that an efficient system is delivered to the customers on time. [1][3]

Scrum development consists of three phases, [1][2][3][4][6]

**Pre-game phase**

This phase involves planning and most of the design and architecture. A Product Backlog list is maintained which contains all the requirements for the project. Requirements can be from customer, a marketing department or from developer’s side. Requirements prioritizations are done and estimates are taken which are then added to the product backlog list. This list is updated after each iteration. People are chosen in this phase which is to be dedicated for the project. Planning is done for the selection of tools, resources, risk management, and controlling tasks are carried out. [2][4][6]

High level system architecture design is obtained based upon product backlog list. At the end of this phase, a high level meeting is called to analyze the design and review it for the next phase of implementations.

**Development phase**

This phase is also termed as the Game Phase. In this phase changes are expected and it is tried to manage all those unexpected with utmost efficiency. This phase is carried out in Sprints. Sprints can be understood as iterative cycles in which increments to the systems are developed. A sprint normally consists of 14 to 28 days of cycle. A sprint contains phases of planning to implementation and then delivery phases. At the end of a sprint some shippable functionality of the system must be achieved. Different teams can work over different increments in various sprints. In a system there can be different amount of sprints depending upon the nature of the project that is being developed. [1][4][6]

**Post-game phase**

This phase indicate the end point before the system can be released. This phase starts when all requirements are implemented. There are no issues to address in this phase. Post game task include tasks like QA of the system, integrating different modules and documenting. [2][4]
Roles in Scrum

There are different roles defined in scrum. Few of them are explained below

**ScrumMaster**

This is a management role whose responsibility is to facilitate the development team and remove all impediments which can cause any delays during the sprint. Scrum Manager is not the boss but in fact behaves like a facilitator between the project team and customers. These are normally the people with decent industry experience to carry the tasks smoothly. This role is also responsible to play a vital role in maintaining a frank atmosphere to encourage communication and cooperation between scrum team members. The responsibilities of ScrumMaster also include taking care of the rules of Scrum to be followed. In daily scrum meetings, team member describe any problem that prevent them to do their jobs, ScrumMaster solve those issues and facilitate the members. This role can be compared with a coach of a team who is always looking for ways to improve the performance of his/her team members. [1][2][3][4][6]

**Scrum Owner**

This role is associated with customer needs. Scrum owner prioritize customer needs and make sure that project is developed according to customer need and the business needs are properly being addressed. Scrum owner strictly keeps an eye over customer requirements and make sure that all efforts go in the right direction. Product Backlog list is maintained and updated by scrum owner. No one else is allowed to modify the product backlog. During meetings, Scrum Owner explains the requirements and then discussion upon the feasibility and completion is carried out in systematic way. Removal or additions of requirements are tackled by Scrum Owner. [1][2][4]

**Scrum Team**

It’s the actual team responsible for all other activities like analysis, design, implementation, and testing. Number of the members of the team may vary and can be as minimum as 3 or can reach to the number 8. Scrum team is self organizing and point out any impediments to the ScrumMaster. They work in mutually interacting atmosphere and participate in daily meetings to update other members about progress. [3][6]

**Customers**

They deal with the tasks which are related to requirements. Customers are linked with Product Backlog lists. Customer satisfaction is among the basic
goals of Scrum development and their views are given much importance. Customer can also participate in scrum meetings taking place at development sites. [1][3][4]

**Chickens**

People those are interested in the project but they do not have any official scrum responsibilities. They participate in scrum meetings but are not allowed to speak or interrupt the proceedings. [4][6]

**Pigs**

It is opposite to the chicken. Any person who has official responsibilities in the scrum team is termed as pigs. These can be a ScrumMaster, Product Owner or any one from the team with a commitment to fulfill a job. [1][2][3][4][6]

**Scrum Meetings**

There are various types of meetings involved in Scrum to carry out the process. [1][2][3][4][6]

**Sprint Planning Meeting**

This duration of this is scheduled for 8 hours and is held in two parts each of 4 hours duration. In the middle a break of 1 hour is given. [1][2][4][6]

The meeting is attended by ScrumMaster, Product Owner and the Team. There can be additional people related to the Sprint but they later they have no part to play in the actual work. Chickens are not entertained in this meeting [14]. In this meeting, the Product Owner presents his Product Backlog, which is prepared prior to the meeting, in front of all participants. The team asks different questions from the Scrum Owner about Product Backlog and then it is decided in this meeting that which items of the Product Backlog can really be implemented during the Sprint. The Team prepares strategy for implementing those items which will be in shippable form at the end of Sprint. [1][2][3][4]

**Daily Scrum Meeting**

As explained from its name, these meetings are held in Scrum development process on daily basis. Meeting duration is about 15 minutes. These meetings provide guidance that how much work is done and what next to do. [2][4]

Each participant is expected to answer three basic questions,
- How much work has been done since yesterday?
- What is the target till tomorrow meeting time?
- What are the issues which are causing delay in achieving the goals?
Meeting is conducted by ScrumMaster. Besides ScrumMaster and the Team, customer may also participate in the meeting depending upon the situation at hand. [3][6]

This meeting is held earlier in the day, at the same time and at the same location.

**Sprint Review Meeting**

The duration of Sprint Review Meetings is of 4 hours. Another one hour is given for the preparation of this review meeting. This meeting is arranged for the Product Owner and other stakeholders to see the outcome of the Sprint. They are shown the practical presentation of the shippable functionality achieved during the Sprint. No discussion is made on what has not be achieved during this meeting. [1][3][4][6]

During the meeting, a member starts with the items of the Backlog completed successfully. Also good and bad experiences are shared during the meeting. Questions of the Product Owner and stakeholders are entertained by the Team.

**Scrum Retrospective Meeting**

The duration of this meeting is kept up to 3 hours. Attendees of this meeting are the Team, ScrumMaster and/or Product Owner. [3][6]

Following questions are answered by all Team members [2][4]
- What went fine in the recent Sprint?
- What needs improvement in the coming up Sprint?
ScrumMaster notes down the answers and only listens to Teams answers.
3. SCRUM Comparison with Waterfall Model

SCRUM, being an agile method has a significantly different approach towards software development than the traditional waterfall model. [5]

Traditional Waterfall model is more detail oriented in its nature. For example documentation has been given tremendous importance. These writings are used and kept so that processes can be well planned and repeated in future. These documents are used for implementation later. Different team members rely solely over documents during development life cycle and mutual communication is less considered important. [2][5][8]

SCRUM relies less over documentation in comparison with Waterfall. Lists and charts that are used in SCRUM are prone to changes at most of the time so mostly new documents are not added but the old ones are modified. Frequent meetings among team members are used as an alternative of those specification documentations. [5][8]

Waterfall model is also termed as heavy weight model because of its planned and detailed nature. All phases of the development life cycle are planned. Requirements are once finalised at the start of the project and are supposed to remain unchanged during the whole period of development. More emphasis is putted on the analysis and design. SCRUM is more adaptive and flexible. Instead of concrete requirements list at the beginning, customers are kept closely intact and requirements changes are obvious. Less amount of time is spent during the analysis and design. As incremental development takes place in SCRUM so there is no need for a complete analysis and design. Small functionalities are development in each sprint. And then integration of those modules is achieved at the post-game phase. [1][4][5][8]

Traditional model assigns role to members for each tasks. These roles are system analyst, designers, requirements engineer, developers and quality engineer. Tasks are controlled by a Manager or CTO. There is no strict command and control structure applied in SCRUM. SCRUM manager’s main task is to remove impediments that are preventing the development team from achieving their set goals. Teams are made self organizing. [1][3]

Customer involvement is a MUST in SCRUM. They are kept in tact all the time and their feedback is taken in to consideration. SCRUM Manager acts on behalf of customers. Customers can be allowed in company regular meetings and their participation is valued. On the other hand in Waterfall, interactions with customer is maximum at the start of the projects and after requirements finalization, there job is considered done owing to the reason that the requirements will not change. [4][8]
A team works over the complete projects in waterfall so the team size may vary from few members to up to a hundred members. In SCRUM team size is kept from 3 to 8. If the size increases then a new team is formed. [2][6]

Waterfall is step by step approach toward software development, tasks are done in their respective orders which means that one group wait for the other one to complete their task than the next team can initiate. This also results in consumption of time and resources. SCRUM works in an iterative way hence less time and resources are wasted. [5][8]

Heavyweight development method like waterfall strictly follow the planning everything, use of tools for the development process is of great importance too. Deviation from such tools and technique is not appreciated. While SCRUM is more people oriented. Strict roles are not as such specified. Decisions can be left to the developers and are not imposed by the management. [1][5][8]

Unlike other new evolving methodologies, SCRUM has been applied to thousands of projects around the globe and they proved extremely successful [1] [3] [8].

On next page the comparison in shown in tabular form.
Table 3.1 gives a brief comparison of waterfall and agile methodologies with respect to a diverse scope. [1][2][3][4][5][6][8]

<table>
<thead>
<tr>
<th>Waterfall (Traditional)</th>
<th>Scrum (Agile)</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Explicit knowledge, as traditional models encourages generation of artifacts in almost every phase and work is done usually in a sequence.</strong></td>
<td><strong>Tacit Knowledge, most of the knowledge remains in the brain of developer.</strong></td>
</tr>
<tr>
<td><strong>Formal but limited communication</strong></td>
<td><strong>Informal but more communication in SCRUM</strong></td>
</tr>
<tr>
<td><strong>Requirements are clear, conventional, change of large magnitude is not expected and system is build-able through proper planning</strong></td>
<td><strong>Basic Supposition</strong></td>
</tr>
<tr>
<td><strong>Process (the life-cycle model)</strong></td>
<td><strong>In-charge</strong></td>
</tr>
<tr>
<td><strong>Management Style</strong></td>
<td><strong>People (in a SCRUM team)</strong></td>
</tr>
<tr>
<td><strong>Project Cycle</strong></td>
<td><strong>Object oriented technology is ideal</strong></td>
</tr>
<tr>
<td><strong>Technology</strong></td>
<td><strong>Involvement of customer</strong></td>
</tr>
<tr>
<td><strong>Favorable Structure</strong></td>
<td><strong>Organizational Structure</strong></td>
</tr>
<tr>
<td><strong>Involvement of customer is most important thing for SCRUM and other agile methodologies. SCRUM owner takes care of customer involvement and satisfaction using feedback</strong></td>
<td></td>
</tr>
<tr>
<td><strong>As there is need of out of the channel communication in SCRUM so it favors flexible organizations where informal communication is encouraged.</strong></td>
<td></td>
</tr>
</tbody>
</table>
4. Challenges for Scrum

It has always been a challenging task for the management to switch from a development methodology that is followed for years to another having no practical exposure to the development team about. As discussed in previous section transition from waterfall to scrum development model is never easy due to differences among both development models at different dimensions. It is a continuous process of learning and implementation. This can be easily seen as discovering new methods and techniques toward software development when an organization wants to switch from waterfall to scrum based development.

It is in human nature that most of the time change of any sort is resisted. Same is the case with the employees working in software industry. They resist changes made to the processes in different ways [7]. Change of software development model is critical one because it modifies process-flow, tools, technologies, structure of teams and/or organizations and last but not the least management practices [4] [5] [6].

Another trend observed after shifting to SCRUM is that people with less experience feel uncomfortable with fast pace scrum methods. Scrum has a structure that focuses over work done in limited time. This requires experts in their domains to carry out the job done. In waterfall, team leads behave like managers and supervise team members where they stuck with any technical matter. Team lead helps them to solve technical problems. On the other hand, Scrum Master does not behave like a manager rather he/she focuses over removing impediments to achieve daily tasks given. People with less experience find difficulties in breaking down big modules in smaller parts for incremental developments. Self managerial skills are needed to carry out scrum procedures. Water fall is process oriented while scrum is dependant upon individual persons to deliver the tasks. [3][4][5]

Communications at all levels are quite important in scrum. Scrum team members with less communication skills find it harder to express the developments and problems that they face during daily scrum meetings. In this situation, members are unable to understand the importance of new proposed system for development.

Most people resist scrum because majority of the people lack knowledge about this methodology. Most of the stakeholders involve in software development may have some idea about general agile based development techniques, but knowledge about scrum is not common. Comprehensive training in this regard may provide us with the solution. [6] [13]

Organizational structures based on waterfall undergo changes and improvements from top to bottom. Scrum grows in bottom to top fashion which
can be time demanding in producing significant results hence implementation require a total support from all stakeholders especially from the executives. [8][13]

Another trouble faced with earlier scrum transition phases is that many releases contain large number of bugs. Development teams focus over sprint completion while less attention is paid to hidden potential bugs. This practice may cause dedicated sprints for those bugs to be fixed which result in over budget and time delays. [5] [13].
5. Questionnaire

As discussed in previous section of ‘Research Methodologies’ that our report is based on a hybrid methodology i.e. it includes both qualitative and quantitative research methodologies. Quantitative part of our research is based on a questionnaire that had helped us to better understand different parameters regarding adoption of agile development models in various organizations.

We have designed questionnaire carefully in a way that it had helped us in doing vigilant analysis. Questionnaire is designed in 3 parts. First part makes ground for multidimensional analysis, second part provides data for quantitative analysis, and third part is for a supportive role i.e. it contains few open-ended questions.

<table>
<thead>
<tr>
<th>Design of Questionnaire</th>
</tr>
</thead>
<tbody>
<tr>
<td>1: Ground for analysis</td>
</tr>
<tr>
<td>Questions for sub-group division information</td>
</tr>
<tr>
<td>2: Core questionnaire</td>
</tr>
<tr>
<td>i. Likert scale</td>
</tr>
<tr>
<td>ii. Non-Likert scale</td>
</tr>
<tr>
<td>3: Supportive optional part</td>
</tr>
<tr>
<td>Open ended questions</td>
</tr>
</tbody>
</table>

Table 5.1

There is always need of a ground to perform analysis, for this reason we have divided responses in various sub groups, to apply quantitative analysis. We have considered various approaches like role of respondent (Designer, Developer, tester, and Project Manager), experience with agile methodologies, experience with traditional methodologies, location etc while making these sub-groups. Such classification of respondents had helped us to understand point-of-view from a particular perspective.

We have focused on issues like pre-adoption training, quality of software product, productivity, communication, testing, changing requirements, customer satisfaction, and self-opinion of respondent from each perspective. To further clarify and validate our analysis we have added three open ended questions at the end where respondent can add something which might have left in questionnaire.
Above is a bubble diagram which explains questionnaire design in an abstract view. Biggest bubble shows core questionnaire, medium bubbles are showing sub-group base, and smaller bubbles are sub-groups. Most of the questions are based on Likert scale that had helped us later during extraction of results from responses.

To compose questionnaire we have used an online tool ‘esurveypro’ which is free and user-friendly. After completing questionnaire we have distributed questionnaire online through target e-mails. We have send questionnaire to our contacts in multiple IT organizations in Sweden and Denmark.
6. Analysis

Analysis of data is most interesting part of this report. We almost have decent number of responses of the questionnaire and along with questionnaire responses we also have couple of face-to-face interviews that had also helped us to perform a better analysis on the available data. As discussed in previous section that our questionnaire is roughly based on three parts i.e. ground for analysis, core research questions, and supportive optional questions. With the help of ‘Ground for analysis’ part we have distributed questions and their respective responses.

We will present analysis of the data that can be found on table below. Each column’s title is representing question number in accordance with questionnaire i.e. column with title ‘1’ means it is linked with question number 1 of questionnaire and in the same way column with title 16 means it is linked with question ‘16’ of questionnaire. For more clarity please view questionnaire used in this study, in appendix 1. First column shows number of respondent. We have calculated Arithmetic Mean and Standard Deviation to know average of values for a question and to better know dispersion of data. All of the abbreviations can be found in appendix 2.

| No | 1   | 2   | 3   | 4   | 5   | 6   | 7   | 8   | 9   | 10  | 11  | 12  | 13  | 14  | 15  | 16  | 17  | 18  | 19  | 21  |
|----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| 1  | S   | OTH | 0.5 | ON  | TD  | 0.5 | 4   | 3   | 4   | 22.5| 3   | 3   | 0   | 30  | 3   | 4   | 4   | 3   | 4   | 4   |
| 2  | S   | DEV | 0.5 | ON  | NS  | 2   | 4   | 1   | 4   | 60  | 4   | 4   | 4   | 70  | 50  | 2   | 4   | 4   | 4   | 4   | 4   |
| 3  | S   | DEV | 2   | ON  | S   | 2   | 4   | 2   | 4   | 30  | 3   | 4   | 70  | 90  | 3   | 4   | 4   | 4   | 4   | 4   |
| 4  | S   | PM  | 2   | ON  | S   | 0   | 4   | 2   | 2   | 25  | 3   | 3   | 25  | 10  | 3   | 2   | 3   | 3   | 4   | 3   |
| 5  | S   | DEV | 1   | ON  | S   | 3   | 3   | 1   | 3   | 25  | 3   | 3   | 40  | 50  | 3   | 3   | 2   | 4   | 3   | 3   |
| 6  | S   | DEV | 3   | ON  | NS  | 0.5 | 4   | 2   | 3   | 10  | 3   | 3   | 3   | 3   | 4   | 4   | 4   | 4   | 4   | 3   |
| 7  | S   | DEV | 1   | ON  | TD  | 2   | 4   | 2   | 3   | 15  | 4   | 4   | 30  | 20  | 3   | 4   | 4   | 4   | 3   | 3   |
| 8  | S   | OTH | 2   | OFF | S   | 1   | 3   | 2   | 2   | 4  | 4   | 1  | 60  | 40  | 3   | 1   | 3   | 3   | 4   | 3   |
| 9  | D   | OTH | 0   | ON  | TD  | 1   | 4   | 1   | 1   | 5   | 1   | 3   | 30  | 40  | 3   | 2   | 3   | 2   | 3   | 2   |
| 10 | S   | DEV | 2   | OFF | TD  | 1   | 4   | 1   | 4   | 45  | 4   | 4   | 80  | 30  | 4   | 4   | 4   | 4   | 4   |
| 11 | D   | DEV | 1   | ON  | S   | 2   | 3   | 2   | 3   | 12.5| 4   | 3   | 60  | 80  | 3   | 2   | 3   | 3   | 3   | 3   |
| 12 | S   | DES | 1   | ON  | NS  | 4   | 3   | 2   | 4   | 15  | 3   | 3   | 50  | 55  | 3   | 3   | 3   | 4   | 4   | 3   |
| 13 | S   | DES | 3   | ON  | NS  | 2   | 3   | 1   | 4   | 17  | 5   | 4   | 3   | 20  | 25  | 3   | 4   | 3   | 3   | 4   | 3   |
| 14 | S   | DEV | 1   | ON  | S   | 3   | 4   | 2   | 4   | 15  | 4   | 4   | 40  | 60  | 2   | 4   | 4   | 4   | 4   |
| 15 | S   | OTH | 1   | ON  | TD  | 1   | 4   | 3   | 4   | 90  | 4   | 4   | 60  | 70  | 4   | 4   | 4   | 4   | 4   |
| 16 | D   | TST | 1   | ON  | S   | 2   | 3   | 2   | 3   | 15  | 3   | 3   | 10  | 45  | 2   | 2   | 4   | 4   | 4   |
| 17 | D   | TST | 5   | ON  | S   | 4   | 4   | 3   | 4   | 20  | 3   | 3   | 15  | 30  | 2   | 4   | 3   | 4   | 4   |
| 18 | S   | TST | 3   | ON  | S   | 4   | 4   | 3   | 4   | 15  | 3   | 4   | 0   | 0   | 2   | 4   | 3   | 4   | 4   |
| 19 | S   | DES | 2   | ON  | S   | 5   | 4   | 2   | 3   | 15  | 3   | 3   | 10  | 20  | 2   | 4   | 3   | 4   | 4   |
| 20 | S   | DEV | 0   | ON  | S   | 1   | 3   | 2   | 3   | 15  | 3   | 3   | 40  | 60  | 1   | 3   | 3   | 3   | 3   |

| Mean | 2.05 | 3.65 | 1.95 | 3.3 | 3.25 | 3.4 | 2.75 | 3.3 | 3.25 | 3.6 | 3.65 | 3.4 |
| Sdev | 1.39 | 0.49 | 0.69 | 0.86| 0.72 | 0.5 | 0.79 | 0.98| 0.72 | 0.5 | 0.59 | 0.6 |

Table: 6.1
6.1. Arithmetic mean and standard deviation

We have used some statistical calculations like arithmetic Mean and Standard Deviation to better understand data. Arithmetic mean is used to calculate average [18] and we will use arithmetic mean throughout our analysis section to compare an average value with other values in a particular column. Other important statistical calculation that we will be interesting in is standard deviation. Standard deviation is a measure of dispersion of data [18]. We will calculate standard deviation to analyze that there is how much variability in the data. If there is a relatively high value for standard deviation then we have to analyze that what can be the possible reasons behind greater dispersion of data.

6.2. Analysis of table 6.1

While looking at table 6.1 we can see a very strong positive response of employees who have switched from a traditional to agile development model. It can be seen on column labeled 8 and also through figure 6.1. This is the first thing a manager wants to see that whether employees will like this change or not, once manager knew that employees do enjoy working with new model then its easy to take steps which can take transition to the next level. All of the respondents of our questionnaire have positive response about migration to agile based methodology. We also have observed that responses of this question are least dispersed and almost all of the respondents have sort of mutual consensus.

While going to bring a major change in software development there is need of extensive coaching and training. Good mentoring could help in assuring a smooth transition and least adoption problems. In column labeled 9, we have asked whether enough training was given to employees or not before going for a transition in development model, it is interesting thing to know that most of the respondents were not satisfied with the level of training given. It could also be seen from figure 6.2 that 25% of respondents said that training was never sufficient for them and 55% said that more could have done to make things better. Standard deviation value for this question is 0.69; we will examine this number later, in sub groups, to see that which subgroup has more harmony when it comes to the question asked.

Agile is more about breaking walls in an organization. In traditional models every one from requirement engineer to tester was creating documents and agile philosophy is that number of documents being created can be reduced to a great extent hence making software development more pacing. But is it that simple? At one of the organization where we were conducting our research there is a joke that, “make a copy of document before destroying it”. The point is that they are not willing to get rid of documents also they don’t know that which document is important and which is not. Problem is worse when
there is no effective communication between employees. Fifteen percent of employees think that there is no effective communication between employees and thirty-five percent thinks that there is need of better communication. Standard deviation of responses against this question is 0.86. We will have a look at this number later in subgroups to see that in which groups’ dispersion of data is relatively low.

Then comes a question related to amount of time spent on daily meeting. We’ll discuss responses of this question later to understand a possible connectivity of an out of place response to daily spent time on meeting. Like if there is a negative trend of most of the answers from a respondent then is it because of less time spend on daily meeting, or if a positive set of answers from a respondent is relating to time spend on daily meeting. May be daily meeting time duration has something very interesting to deal with, may be not. We’ll see ‘daily meeting duration’ later in subgroup and there will be discussion about this question in discussion section of the report.

Then comes a question about productivity. Column 12 in table 6.1 is showing responses about this question. Almost all of the respondents are agreed of the fact that productivity of developers is increased after switching to agile based methodology. Interesting thing is that only one respondent is against this point-of-view and according to this respondent productivity id decreased. We have observed this respondent in detail and we have noticed two things which might have lead to dissatisfaction regarding productivity for this subject. First is that this respondent have no experience with any traditional software development methodology and second that duration of daily meeting is five minutes. These are quite interesting things which we have noticed during analysis of data and will discuss about it in discussion part. But it is apparent that experience and daily meeting duration has something to do with question we have asked.

All of our respondents also agreed that switching to agile does not bring negative impact on quality of software. As shown in column 13 of table 6.1 and also figure 6.5 there is not a single respondent with negative response about impact on quality question. So once again there is a mutual consensus among all of the respondents our questionnaire. But we’ll see later in coming sections that if there is any slight disparity among various subgroups regarding this question. While responding about difficulty or simplicity of testing in agile based testing methodology, our respondents are not confidant enough to decide that how is it, exactly, to manage testing in agile based methodology? 45% thinks that it’s partially easy and 35% said that it’s partially difficult. Having quite dispersed feedback (with ‘0.79’ standard deviation) from respondents we have also marked this column and will see how does it looks in various subgroups.

Requirement engineering related questions also have most of the positive responses. 75% of the respondent thinks that it is somehow convenient to
handle changing requirements in agile based development methodology while 85% thinks that requirements testability and traceability is easier in agile based development methodology. All of respondents agreed that customer’s satisfaction has increased after adopting agile based methodology. 95% thinks that it was a good move to switch from traditional development methodology to agile based methodology and once again 95% also think that agile based development methodology can be a good option for them in log run.

There will be some charts based on likert scale data collected during our study and after that we’ll present analysis of data in subgroups.

6.3. Analysis on Likert Scale Distribution of Data

We have tried to design questionnaire in a way that most of the potential answers could be mapped on likert scale for ease of analysis and conclusion. Below you can find some important questions from the questionnaire and their responses in percentage.

**Employee’s Response**

In a previous section we have discussed that change is something that’s not welcomed by employees, especially when there is a change in workflow or other basic practices. In our research we were interested to know that what employee’s response was, while organization had migrated from traditional to agile development model.

![Figure 6.1](image)

**Training**

Training is a crucial thing to do while bringing any change in an organization. Change could be a change/update in tools, technologies, or model used by employees in an organization. It is quite interesting thing to know that a big
percentage of employees do think that enough training was not given to them while switching from traditional to agile development model.

![Training Given? Graph](image)

**Figure 6.2**

**Communication**

There is need of effective communication in agile based development models. In most of the agile based methodologies there is daily meeting and other activities like pair programming so if employees lack fluent communication then they should be mentored to polish their communication skill.

![Effective Communication? Graph](image)

**Figure 6.3**

**Productivity**

Switching to agile based methodology must have some impact on productivity of developers. Desired impact is increased in productivity and if it is not the case then there is need to bring whole process under scanner to check issues with the current model/work-flow.
Quality of product

It is interesting to observe that none of the questionnaire respondents have said that switching to agile have resulted in decreased quality of software product.

Software Testing

Testing is not the same in agile development models as compare to traditional development models. There is more iteration hence more integration. To cope with changing requirement and iterative development there is need of automated regression and integration testing.
Changing Requirements

One of the basic motivations to switch on agile based software development models is that it supports changing requirements. We have tried to examine in our research that what does practitioner of agile methodologies thinks about agile development’s ability to handle changing requirements. It comes as no surprise that seventy five percent of them are positive about this characteristic of agile based software development models.

![Figure 6.7](image)

Requirements Engineering

Though it seems from our previous results that it is convenient to accommodate changing requirements in project, but what about other requirements engineering issues like tractability and testability?

![Figure 6.8](image)

Customer Satisfaction

Customer satisfaction is perhaps the most important thing for any business. In software houses there is stress to meet deadline and to assure maximum quality of a software product. Why they do that? The reason is to achieve customer satisfaction. So what ever software development model is being used
by an organization for its software products they do keep customer satisfaction aspect in mind while adopting that model.

![Customer Satisfaction Increased?](image)

**Figure 6.9**

**Overall Experience**

Last question of likert scale nature in our questionnaire was to judge that so far, by and large, what agile practitioner feels about their experience with agile based models. Companies around the globe are switching to agile based methodologies and by considering that it came with no surprise to us that Ninety five percent of the questionnaire respondents have some sort of positive feeling about their agile based models experience.

![Was Migration a Good Move?](image)

**Figure 6.10**
6.4. Subgroup Creation

To better analyze data we have created sub groups from our main table 7.2. These subgroups are quite diverse in nature and they helped us to better analyze data in multiple perspectives.

<table>
<thead>
<tr>
<th>Subgroup</th>
<th>No. of Participants</th>
<th>Q. No</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Designer</td>
<td>3</td>
<td>2</td>
<td>Respondents with job function as a Designer</td>
</tr>
<tr>
<td>Developer</td>
<td>9</td>
<td>2</td>
<td>Respondents with job function as a Developer</td>
</tr>
<tr>
<td>Other Job</td>
<td>8</td>
<td>2</td>
<td>Respondents with other job functions</td>
</tr>
<tr>
<td>Scrum</td>
<td>11</td>
<td>5</td>
<td>Respondents with Scrum as development model</td>
</tr>
<tr>
<td>Other Agile</td>
<td>9</td>
<td>5</td>
<td>Respondents with other then Scrum model</td>
</tr>
<tr>
<td>&lt;2 Traditional</td>
<td>8</td>
<td>7</td>
<td>Less then 2 years of any traditional model experience</td>
</tr>
<tr>
<td>&gt;=2 Traditional</td>
<td>12</td>
<td>7</td>
<td>More then or equal to 2 years of any traditional model experience</td>
</tr>
<tr>
<td>&lt;=1 Agile</td>
<td>11</td>
<td>3</td>
<td>Respondents with less then or equal to 1 year agile exp</td>
</tr>
<tr>
<td>&gt; Agile</td>
<td>9</td>
<td>3</td>
<td>Respondents with more then 1 year agile experience</td>
</tr>
<tr>
<td>&gt;=2 Agile &amp; Traditional</td>
<td>5</td>
<td>3 &amp; 7</td>
<td>More then or equal to 2 years of any traditional and agile model experience</td>
</tr>
</tbody>
</table>

Table: 6.2

Now we will elaborate each subgroup and will perform analysis using data from corresponding subgroup table.
6.5. Designer

In this subgroup we have put together responses of all the designers and not surprisingly we have found lots of similarities in their responses. Please find the table showing responses from developers below.

<table>
<thead>
<tr>
<th>No.</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
<th>11</th>
<th>12</th>
<th>13</th>
<th>14</th>
<th>15</th>
<th>16</th>
<th>17</th>
<th>18</th>
<th>19</th>
<th>20</th>
<th>21</th>
</tr>
</thead>
<tbody>
<tr>
<td>12</td>
<td>S</td>
<td>DES</td>
<td>1</td>
<td>ON</td>
<td>NS</td>
<td>4</td>
<td>3</td>
<td>2</td>
<td>4</td>
<td>15</td>
<td>3</td>
<td>3</td>
<td>30</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>4</td>
</tr>
<tr>
<td>13</td>
<td>S</td>
<td>DES</td>
<td>3</td>
<td>ON</td>
<td>NS</td>
<td>2</td>
<td>3</td>
<td>1</td>
<td>4</td>
<td>17.5</td>
<td>4</td>
<td>3</td>
<td>20</td>
<td>3</td>
<td>4</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>4</td>
<td>3</td>
</tr>
<tr>
<td>19</td>
<td>S</td>
<td>DES</td>
<td>2</td>
<td>ON</td>
<td>S</td>
<td>5</td>
<td>4</td>
<td>2</td>
<td>3</td>
<td>15</td>
<td>3</td>
<td>3</td>
<td>10</td>
<td>2</td>
<td>2</td>
<td>4</td>
<td>3</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>Mean</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>3.3</td>
<td>1.7</td>
<td>3.7</td>
<td>3.3</td>
<td>3</td>
<td>2.67</td>
<td>3.67</td>
<td>3</td>
<td>3.33</td>
<td>4</td>
<td>3.67</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SDEV</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0.6</td>
<td>0.6</td>
<td>0.6</td>
<td>0.6</td>
<td>0</td>
<td>0.58</td>
<td>0.58</td>
<td>0</td>
<td>0.58</td>
<td>0</td>
<td>0.58</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table: 6.3

We got three responses from software designers in our study and there is a certain trend in their responses and we can see that in column thirteen, seventeen, and nineteen there is zero standard deviation. So according to software designers agile software development has positive impact on software quality and also it is convenient for requirement engineering related activities. They also strongly agree that overall it was a good move to switch from traditional to agile based development model.

Designer group thinks that things could have even more improved by giving some additional training. Designer sub group don’t have any negative remark in whole questionnaire, as it could be seen that on scale from 1 to 4 on likert scale they are seldom below 3. One other interesting thing we have seen is that designers are not quite sure about a question that testing has become easier in agile or not. There is partially yes and partially no sort of response from designers against that question. It would be interesting thing to see that what developers and respondents with other job function replied about this question.
6.6. Developer

Below you can see table with all of the respondents who are working as a developer.

<table>
<thead>
<tr>
<th>No</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
<th>11</th>
<th>12</th>
<th>13</th>
<th>14</th>
<th>15</th>
<th>16</th>
<th>17</th>
<th>18</th>
<th>19</th>
<th>21</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>S</td>
<td>DEV</td>
<td>0.5</td>
<td>ON</td>
<td>NS</td>
<td>2</td>
<td>4</td>
<td>1</td>
<td>4</td>
<td>60</td>
<td>4</td>
<td>4</td>
<td>70</td>
<td>50</td>
<td>2</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>3</td>
<td>S</td>
<td>DEV</td>
<td>2</td>
<td>ON</td>
<td>S</td>
<td>2</td>
<td>4</td>
<td>2</td>
<td>4</td>
<td>30</td>
<td>3</td>
<td>4</td>
<td>70</td>
<td>90</td>
<td>3</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>5</td>
<td>S</td>
<td>DEV</td>
<td>1</td>
<td>ON</td>
<td>S</td>
<td>3</td>
<td>3</td>
<td>1</td>
<td>3</td>
<td>25</td>
<td>3</td>
<td>3</td>
<td>40</td>
<td>50</td>
<td>3</td>
<td>3</td>
<td>2</td>
<td>4</td>
<td>3</td>
</tr>
<tr>
<td>6</td>
<td>S</td>
<td>DEV</td>
<td>3</td>
<td>ON</td>
<td>NS</td>
<td>0.5</td>
<td>4</td>
<td>2</td>
<td>3</td>
<td>10</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>3</td>
</tr>
<tr>
<td>7</td>
<td>S</td>
<td>DEV</td>
<td>1</td>
<td>ON</td>
<td>TD</td>
<td>2</td>
<td>4</td>
<td>2</td>
<td>3</td>
<td>15</td>
<td>4</td>
<td>4</td>
<td>30</td>
<td>20</td>
<td>3</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>3</td>
</tr>
<tr>
<td>10</td>
<td>S</td>
<td>DEV</td>
<td>2</td>
<td>OFF</td>
<td>TD</td>
<td>1</td>
<td>4</td>
<td>1</td>
<td>4</td>
<td>45</td>
<td>4</td>
<td>4</td>
<td>80</td>
<td>30</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>11</td>
<td>D</td>
<td>DEV</td>
<td>1</td>
<td>ON</td>
<td>S</td>
<td>2</td>
<td>3</td>
<td>2</td>
<td>3</td>
<td>12.5</td>
<td>4</td>
<td>3</td>
<td>60</td>
<td>80</td>
<td>3</td>
<td>2</td>
<td>3</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>14</td>
<td>S</td>
<td>DEV</td>
<td>1</td>
<td>ON</td>
<td>S</td>
<td>3</td>
<td>4</td>
<td>2</td>
<td>4</td>
<td>15</td>
<td>4</td>
<td>4</td>
<td>40</td>
<td>60</td>
<td>2</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>20</td>
<td>S</td>
<td>DEV</td>
<td>0</td>
<td>ON</td>
<td>S</td>
<td>1</td>
<td>3</td>
<td>2</td>
<td>3</td>
<td>15</td>
<td>3</td>
<td>3</td>
<td>40</td>
<td>60</td>
<td>1</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
</tr>
</tbody>
</table>

Mean: 3.7 1.7 3.44 3.6 3.6 2.78 3.6 3.6 3.8 3.4 3.44
SDEV: 0.5 0.5 0.53 0.5 0.5 0.97 0.7 0.7 0.4 0.5 0.53

Table: 6.4

Respondents of this group are agreed that customers’ satisfaction has increased after switching to agile based methodology. While average daily meeting time is about twenty-five minutes, according to a developer where development is offshore the daily meeting time is forty-five minutes, though it is understandable while considering challenges of offshore development. In this group an interesting thing is that respondent with no traditional development experience does not look that happy. That respondent have never selected a maximum positive option presented on likert scale and they think that more training could have improved results and testing is difficult to manage in agile based development models.

We can see that most of the respondents have their concern about testing in agile based development, whether it is because of increased automated testing? There is need to scrutinize that why developers are thinking that testing is not easy to manage in agile based development models.
6.7. Other Job Functions

We have analyzed data from respondents who have job function of a developer or a designer and now we’ll analyze data from rest of respondents. These respondents are working as tester, project manager, and some other job function in software development lifecycle. Below you can find a table representing data from all those respondents.

<table>
<thead>
<tr>
<th>No</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
<th>11</th>
<th>12</th>
<th>13</th>
<th>14</th>
<th>15</th>
<th>16</th>
<th>17</th>
<th>18</th>
<th>19</th>
<th>21</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>S</td>
<td>OTH</td>
<td>0.5</td>
<td>ON</td>
<td>TD</td>
<td>0.5</td>
<td>4</td>
<td>3</td>
<td>4</td>
<td>22.5</td>
<td>3</td>
<td>3</td>
<td>0</td>
<td>30</td>
<td>3</td>
<td>4</td>
<td>4</td>
<td>3</td>
<td>4</td>
</tr>
<tr>
<td>8</td>
<td>S</td>
<td>OTH</td>
<td>2</td>
<td>OFF</td>
<td>S</td>
<td>1</td>
<td>3</td>
<td>2</td>
<td>2</td>
<td>45</td>
<td>3</td>
<td>4</td>
<td>60</td>
<td>40</td>
<td>3</td>
<td>1</td>
<td>3</td>
<td>3</td>
<td>4</td>
</tr>
<tr>
<td>9</td>
<td>D</td>
<td>OTH</td>
<td>0</td>
<td>ON</td>
<td>TD</td>
<td>1</td>
<td>4</td>
<td>1</td>
<td>1</td>
<td>5</td>
<td>1</td>
<td>3</td>
<td>30</td>
<td>40</td>
<td>3</td>
<td>2</td>
<td>2</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>15</td>
<td>S</td>
<td>OTH</td>
<td>1</td>
<td>ON</td>
<td>TD</td>
<td>1</td>
<td>4</td>
<td>3</td>
<td>4</td>
<td>90</td>
<td>4</td>
<td>4</td>
<td>60</td>
<td>70</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>16</td>
<td>D</td>
<td>TST</td>
<td>1</td>
<td>ON</td>
<td>S</td>
<td>2</td>
<td>3</td>
<td>2</td>
<td>3</td>
<td>15</td>
<td>3</td>
<td>3</td>
<td>10</td>
<td>45</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>17</td>
<td>D</td>
<td>TST</td>
<td>5</td>
<td>ON</td>
<td>S</td>
<td>4</td>
<td>4</td>
<td>3</td>
<td>4</td>
<td>20</td>
<td>3</td>
<td>3</td>
<td>15</td>
<td>30</td>
<td>2</td>
<td>4</td>
<td>3</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>18</td>
<td>S</td>
<td>TST</td>
<td>3</td>
<td>ON</td>
<td>S</td>
<td>4</td>
<td>4</td>
<td>3</td>
<td>4</td>
<td>15</td>
<td>3</td>
<td>4</td>
<td>0</td>
<td>0</td>
<td>2</td>
<td>4</td>
<td>3</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>4</td>
<td>S</td>
<td>PM</td>
<td>2</td>
<td>ON</td>
<td>S</td>
<td>0</td>
<td>4</td>
<td>2</td>
<td>2</td>
<td>25</td>
<td>3</td>
<td>3</td>
<td>25</td>
<td>10</td>
<td>3</td>
<td>2</td>
<td>3</td>
<td>3</td>
<td>4</td>
</tr>
<tr>
<td>Mean</td>
<td></td>
<td></td>
<td>3.8</td>
<td>2.4</td>
<td>3</td>
<td>2.9</td>
<td>3.4</td>
<td></td>
<td></td>
<td>2.8</td>
<td>2.9</td>
<td>3</td>
<td>3.5</td>
<td>3.8</td>
<td></td>
<td></td>
<td></td>
<td>3.3</td>
<td></td>
</tr>
<tr>
<td>SDEV</td>
<td></td>
<td></td>
<td>0.5</td>
<td>0.7</td>
<td>1.2</td>
<td>0.8</td>
<td>0.5</td>
<td></td>
<td></td>
<td>0.7</td>
<td>1.2</td>
<td>0.8</td>
<td>0.5</td>
<td>0.7</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0.7</td>
</tr>
</tbody>
</table>

Table: 6.5

As we have feedback of respondents with relatively diverse background in terms of job description hence there is considerable amount of variance in columns of this table. As seen from other groups that employees’ response regarding switching to agile methodology was good, this group all suggest that in general employees were happy to switch to this new method of software development. There is also a degree of consensus in some other questions like, when asked about impact on quality after switching on agile; they said that over all impact of agile methodology on software quality was good. There was also positive feedback when asked about level of customer satisfaction achieved after adopting an agile methodology.

There is some divergence when asked about effective communication between employees. It is quite different from as compare to analysis for developers and designers; we have seen that they think that there is effective communication taking place among them. In agile based methodologies effective communication in team members is of high significance as it paces overall progress and eliminates unnecessary documentation.
6.8. Scrum Development Model

Now we will present data from another facet. After presenting data from aspect of ‘job description’ now we will come up with group of respondents who are currently working with Scrum development model.

| No | 1   | 2   | 3   | 4   | 5   | 6   | 7   | 8   | 9   | 10  | 11  | 12  | 13  | 14  | 15  | 16  | 17  | 18  | 19  | 20  |
|----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| 3  | S   | DEV | 2   | ON  | S   | 2   | 4   | 2   | 4   | 38  | 3   | 4   | 78  | 3   | 4   | 4   | 4   | 4   | 4   | 4   |
| 4  | S   | PM  | 2   | ON  | S   | 0   | 4   | 2   | 2   | 25  | 3   | 3   | 25  | 3   | 2   | 3   | 4   | 3   | 3   | 3   |
| 5  | S   | DEV | 1   | ON  | S   | 3   | 3   | 1   | 3   | 25  | 3   | 3   | 40  | 5   | 3   | 2   | 4   | 3   | 3   | 3   |
| 8  | S   | OTH | 2   | OFF | S   | 1   | 3   | 2   | 2   | 45  | 3   | 4   | 60  | 4   | 3   | 1   | 3   | 3   | 3   | 4   | 3   |
| 11 | D   | DEV | 1   | ON  | S   | 2   | 3   | 2   | 3   | 13.5| 4   | 3   | 60  | 8   | 3   | 2   | 3   | 3   | 3   | 3   |
| 14 | S   | DEV | 1   | ON  | S   | 3   | 4   | 2   | 4   | 15  | 4   | 4   | 40  | 6   | 2   | 4   | 4   | 4   | 4   | 4   | 4   |
| 16 | D   | TST | 1   | ON  | S   | 2   | 3   | 2   | 3   | 15  | 3   | 3   | 10  | 4   | 2   | 2   | 2   | 4   | 4   | 4   | 4   |
| 17 | D   | TST | 5   | ON  | S   | 4   | 4   | 3   | 4   | 20  | 3   | 3   | 15  | 30  | 2   | 3   | 3   | 3   | 3   | 3   | 4   |
| 18 | S   | TST | 3   | ON  | S   | 4   | 4   | 3   | 4   | 15  | 3   | 3   | 4   | 0   | 2   | 4   | 3   | 4   | 4   | 4   | 3   |
| 19 | S   | DES | 2   | ON  | S   | 5   | 4   | 2   | 3   | 15  | 3   | 3   | 10  | 20  | 2   | 4   | 3   | 4   | 4   | 4   | 4   |
| 20 | S   | DEV | 0   | ON  | S   | 1   | 3   | 2   | 3   | 15  | 3   | 3   | 40  | 60  | 1   | 3   | 3   | 3   | 3   | 3   |

| Mean |      |      |      |      |      |      |      |      |      |      |      |      |      |      |      |      |      |      |      |      |
| SDEV | 3.55 | 2.09 | 3.2  | 3.2  | 3.4  | 2.4  | 3   | 3   | 3.6  | 3.7  | 3.3  | 0.52 | 0.54 | 0.8  | 0.4  | 0.5  | 0.7  | 1.1  | 0.6  | 0.5  | 0.5  | 0.6  |

Table: 6.6

While looking at responses of respondents who are currently working with scrum development model, we can notice that they have given positive feedback for most of the questions. There is some accord among respondents. We will rate this subgroup highly in discussion section as respondents for this section are ones’ who are having hands-on experience in agile methodology are they are currently working with Scrum development model.

We have noticed that respondents with less experience in any agile based software development model thinks that it is difficult to manage testing in agile based methodology while respondents who are relatively more experienced in this methodology are saying that in agile it is not difficult to handle software testing. One thing is quite clear that experience does matters in agile software development methodologies.
6.9. Other then Scrum Development Model

In this group we have presented data from all those respondents who are currently working with an agile software development model other than Scrum. Below you can find a table with data received from such respondents.

<table>
<thead>
<tr>
<th>No</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
<th>11</th>
<th>12</th>
<th>13</th>
<th>14</th>
<th>15</th>
<th>16</th>
<th>17</th>
<th>18</th>
<th>19</th>
<th>21</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>S</td>
<td>OTH</td>
<td>0.5</td>
<td>ON</td>
<td>TD</td>
<td>0.5</td>
<td>4</td>
<td>3</td>
<td>4</td>
<td>22.5</td>
<td>3</td>
<td>3</td>
<td>0</td>
<td>30</td>
<td>3</td>
<td>4</td>
<td>4</td>
<td>3</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>2</td>
<td>S</td>
<td>DEV</td>
<td>0.5</td>
<td>ON</td>
<td>NS</td>
<td>2</td>
<td>4</td>
<td>1</td>
<td>4</td>
<td>60</td>
<td>4</td>
<td>4</td>
<td>70</td>
<td>50</td>
<td>2</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>6</td>
<td>S</td>
<td>DEV</td>
<td>3</td>
<td>ON</td>
<td>NS</td>
<td>0.5</td>
<td>4</td>
<td>2</td>
<td>3</td>
<td>10</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>3</td>
</tr>
<tr>
<td>7</td>
<td>S</td>
<td>DEV</td>
<td>1</td>
<td>ON</td>
<td>TD</td>
<td>2</td>
<td>4</td>
<td>2</td>
<td>3</td>
<td>15</td>
<td>4</td>
<td>4</td>
<td>30</td>
<td>20</td>
<td>3</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>3</td>
</tr>
<tr>
<td>9</td>
<td>D</td>
<td>OTH</td>
<td>0</td>
<td>ON</td>
<td>TD</td>
<td>1</td>
<td>4</td>
<td>1</td>
<td>1</td>
<td>5</td>
<td>1</td>
<td>3</td>
<td>30</td>
<td>40</td>
<td>3</td>
<td>2</td>
<td>2</td>
<td>3</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>10</td>
<td>S</td>
<td>DEV</td>
<td>2</td>
<td>OFF</td>
<td>TD</td>
<td>1</td>
<td>4</td>
<td>1</td>
<td>4</td>
<td>45</td>
<td>4</td>
<td>4</td>
<td>80</td>
<td>30</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>12</td>
<td>S</td>
<td>DES</td>
<td>1</td>
<td>ON</td>
<td>NS</td>
<td>4</td>
<td>3</td>
<td>2</td>
<td>4</td>
<td>15</td>
<td>3</td>
<td>3</td>
<td>50</td>
<td>55</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>13</td>
<td>S</td>
<td>DES</td>
<td>3</td>
<td>ON</td>
<td>NS</td>
<td>2</td>
<td>3</td>
<td>1</td>
<td>4</td>
<td>17.5</td>
<td>4</td>
<td>3</td>
<td>20</td>
<td>25</td>
<td>3</td>
<td>4</td>
<td>3</td>
<td>4</td>
<td>3</td>
<td>4</td>
</tr>
<tr>
<td>15</td>
<td>S</td>
<td>OTH</td>
<td>1</td>
<td>ON</td>
<td>TD</td>
<td>1</td>
<td>4</td>
<td>3</td>
<td>4</td>
<td>90</td>
<td>4</td>
<td>4</td>
<td>60</td>
<td>70</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>Mean</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>3.8</td>
<td>1.8</td>
<td>3.4</td>
<td>3.3</td>
<td>3.4</td>
<td>3.2</td>
<td>3.7</td>
<td>3.6</td>
<td>3.6</td>
<td>3.6</td>
<td>3.6</td>
<td>3.6</td>
<td>3.6</td>
<td></td>
</tr>
<tr>
<td>SDEV</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0.4</td>
<td>0.8</td>
<td>1</td>
<td>1</td>
<td>0.5</td>
<td>0.7</td>
<td>0.7</td>
<td>0.7</td>
<td>0.5</td>
<td>0.7</td>
<td>0.5</td>
<td>0.5</td>
<td>0.5</td>
<td></td>
</tr>
</tbody>
</table>

Table: 6.7

Some of the respondents in above table are working with an agile software development model which is not Scrum but somewhat close to it while other are working with a totally different agile software development model. According to [2] some of the famous agile development models other than Scrum are Extreme Programming, Feature Driven Development, The Rational Unified Process, Dynamic Systems Development Method, and Adaptive Software Development.

Though respondents of this group are working on different agile software development models but still they have degree of consensus among them regarding employees’ response in adopting an agile methodology. Mean value for this response is 3.8 from a maximum of 4.0. One interesting fact about this subgroup is that a good number of respondents in this group think that training was never enough for them and there was a serious need of more training for them.

There is some variance in responses when asked about effective communication and impact on software quality. Respondents who are working with an agile software development model think that there is effective communication among team members but ones who are working with other agile development methodologies have some doubts about that.
6.10. Less Then 2 Year of Traditional Development Model Experience

This is group with respondents who have less then two years of experience with any traditional software development model. Please find table 6.8 below to find out responses for this group.

| No | 1   | 2   | 3   | 4   | 5   | 7   | 8   | 9   | 10  | 11  | 12  | 13  | 14  | 15  | 16  | 17  | 18  | 19  | 21  |
|----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| 1  | S   | OTH | 0.5 | ON  | TD  | 0.5 | 4   | 3   | 4   | 22.5| 3   | 3   | 0   | 30  | 3   | 4   | 4   | 3   | 4   |
| 4  | S   | PM  | 2   | ON  | S   | 0   | 4   | 2   | 2   | 25  | 3   | 3   | 25  | 10  | 3   | 2   | 3   | 3   | 4   |
| 6  | S   | DEV | 3   | ON  | NS  | 0.5 | 4   | 2   | 3   | 10  | 3   | 3   | 3   | 4   | 4   | 4   | 3   | 3   |
| 8  | S   | OTH | 2   | OFF | S   | 1   | 3   | 2   | 2   | 45  | 3   | 4   | 60  | 40  | 3   | 1   | 3   | 3   | 4   |
| 9  | D   | OTH | 0   | ON  | TD  | 1   | 4   | 1   | 1   | 5   | 1   | 3   | 30  | 40  | 3   | 2   | 2   | 3   | 2   |
| 10 | S   | DEV | 2   | OFF | TD  | 1   | 4   | 1   | 1   | 4   | 4   | 4   | 80  | 30  | 4   | 4   | 4   | 4   | 4   |
| 15 | S   | OTH | 1   | ON  | TD  | 1   | 4   | 1   | 1   | 4   | 1   | 60  | 3   | 4   | 4   | 4   | 4   | 4   |
| 20 | S   | DEV | 0   | ON  | S   | 1   | 3   | 2   | 3   | 15  | 3   | 3   | 40  | 60  | 1   | 3   | 3   | 3   | 3   |

**Table: 6.8**

As we know that subjects of this group are relatively less experienced. They have less then two years of experience with traditional software development methodologies. It is a good opportunity for us to analyze how less experienced employees see adoption of agile methodology.

It is quite clear that initial response of employees was quite good and we haven’t observed any negative response but they were in favor of little more training. One respondent disagrees that there is effective communication among team members. When we observed that employee closely we analyzed that respondent’s negative answer could be for one reason as in this group this particular respondent was having no experience from agile software development experience. So may be it is necessary to have at least some experience before getting comfortable with agile software development.

There was another column with high standard deviation and we also analyzed that column more closely and observed that a respondent working with off-shore development is not happy with the way changing requirements are being handled. In that case off-shore software development has provoked a negative response from our respondent.
6.11. More Then or Equal to 2 Year of Traditional Development Model Experience

This table 6.9 shows response with a group of respondents which have more then 2 years of traditional software development experience. For more clarity please take help from questionnaire, which could be found enclosed, in appendix.

<table>
<thead>
<tr>
<th>No</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
<th>11</th>
<th>12</th>
<th>13</th>
<th>14</th>
<th>15</th>
<th>16</th>
<th>17</th>
<th>18</th>
<th>19</th>
<th>21</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>S</td>
<td>DEV</td>
<td>0.5</td>
<td>ON</td>
<td>NS</td>
<td>2</td>
<td>4</td>
<td>1</td>
<td>4</td>
<td>60</td>
<td>4</td>
<td>4</td>
<td>70</td>
<td>50</td>
<td>2</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>3</td>
<td>S</td>
<td>DEV</td>
<td>2</td>
<td>ON</td>
<td>S</td>
<td>2</td>
<td>4</td>
<td>2</td>
<td>4</td>
<td>30</td>
<td>3</td>
<td>4</td>
<td>70</td>
<td>90</td>
<td>3</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>5</td>
<td>S</td>
<td>DEV</td>
<td>1</td>
<td>ON</td>
<td>S</td>
<td>3</td>
<td>3</td>
<td>1</td>
<td>3</td>
<td>25</td>
<td>3</td>
<td>3</td>
<td>40</td>
<td>50</td>
<td>3</td>
<td>3</td>
<td>2</td>
<td>4</td>
<td>3</td>
</tr>
<tr>
<td>7</td>
<td>S</td>
<td>DEV</td>
<td>1</td>
<td>ON</td>
<td>TD</td>
<td>2</td>
<td>4</td>
<td>2</td>
<td>3</td>
<td>15</td>
<td>4</td>
<td>4</td>
<td>30</td>
<td>20</td>
<td>3</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>3</td>
</tr>
<tr>
<td>11</td>
<td>D</td>
<td>DEV</td>
<td>1</td>
<td>ON</td>
<td>S</td>
<td>2</td>
<td>3</td>
<td>2</td>
<td>3</td>
<td>12.5</td>
<td>4</td>
<td>3</td>
<td>60</td>
<td>80</td>
<td>3</td>
<td>2</td>
<td>3</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>12</td>
<td>S</td>
<td>DES</td>
<td>1</td>
<td>ON</td>
<td>NS</td>
<td>4</td>
<td>3</td>
<td>2</td>
<td>4</td>
<td>15</td>
<td>3</td>
<td>3</td>
<td>50</td>
<td>55</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>4</td>
</tr>
<tr>
<td>13</td>
<td>S</td>
<td>DES</td>
<td>3</td>
<td>ON</td>
<td>NS</td>
<td>2</td>
<td>3</td>
<td>1</td>
<td>4</td>
<td>17.5</td>
<td>4</td>
<td>3</td>
<td>20</td>
<td>25</td>
<td>3</td>
<td>4</td>
<td>3</td>
<td>3</td>
<td>4</td>
</tr>
<tr>
<td>14</td>
<td>S</td>
<td>DEV</td>
<td>1</td>
<td>ON</td>
<td>S</td>
<td>3</td>
<td>4</td>
<td>2</td>
<td>4</td>
<td>15</td>
<td>4</td>
<td>4</td>
<td>40</td>
<td>60</td>
<td>2</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>16</td>
<td>D</td>
<td>TST</td>
<td>1</td>
<td>ON</td>
<td>S</td>
<td>2</td>
<td>3</td>
<td>2</td>
<td>3</td>
<td>15</td>
<td>3</td>
<td>3</td>
<td>10</td>
<td>45</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>17</td>
<td>D</td>
<td>TST</td>
<td>5</td>
<td>ON</td>
<td>S</td>
<td>4</td>
<td>4</td>
<td>3</td>
<td>4</td>
<td>20</td>
<td>3</td>
<td>3</td>
<td>15</td>
<td>30</td>
<td>2</td>
<td>4</td>
<td>3</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>18</td>
<td>S</td>
<td>TST</td>
<td>3</td>
<td>ON</td>
<td>S</td>
<td>4</td>
<td>4</td>
<td>3</td>
<td>4</td>
<td>15</td>
<td>3</td>
<td>4</td>
<td>0</td>
<td>0</td>
<td>2</td>
<td>4</td>
<td>3</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>19</td>
<td>S</td>
<td>DES</td>
<td>2</td>
<td>ON</td>
<td>S</td>
<td>5</td>
<td>4</td>
<td>2</td>
<td>3</td>
<td>15</td>
<td>3</td>
<td>3</td>
<td>10</td>
<td>20</td>
<td>2</td>
<td>4</td>
<td>3</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>Mean</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>3.6</td>
<td>1.9</td>
<td>3.6</td>
<td>3.4</td>
<td>3.4</td>
<td>2.5</td>
<td>3.5</td>
<td>3.2</td>
<td>3.8</td>
<td>3.8</td>
</tr>
<tr>
<td>SDEV</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0.5</td>
<td>0.7</td>
<td>0.5</td>
<td>0.5</td>
<td>0.5</td>
<td>0.5</td>
<td>0.8</td>
<td>0.7</td>
<td>0.5</td>
<td>0.5</td>
</tr>
</tbody>
</table>

Table 6.9

We could see that responses in this group are less diverse as compare to few previous groups. There are a few slightly negative responses but only one of them is with value ‘1’ that is maximum negative value in our likert-scale model of questionnaire. According to that respondent there is exceedingly need of more training. Now we will take this training issue in our next section as most of the respondents seem to be unhappy of training given to them. May be training/mentoring is something that needs more attention to get better results in agile software development model.

There is no such conspicuous figure in the table other then one we have already explained above.
6.12. Less Then or Equal to 1 Year of Agile Development Model Experience

In this table we have grouped together respondents with respect to their experience in agile based software development model. All the respondents with less then one year of experience in any agile based software development are grouped in table 6.10, which could be found below.

<table>
<thead>
<tr>
<th>No</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
<th>11</th>
<th>12</th>
<th>13</th>
<th>14</th>
<th>15</th>
<th>16</th>
<th>17</th>
<th>18</th>
<th>19</th>
<th>21</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>S</td>
<td>OTH</td>
<td>0.5</td>
<td>ON</td>
<td>TD</td>
<td>0.5</td>
<td>4</td>
<td>3</td>
<td>4</td>
<td>22.5</td>
<td>3</td>
<td>0</td>
<td>30</td>
<td>3</td>
<td>4</td>
<td>4</td>
<td>3</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>2</td>
<td>S</td>
<td>DEV</td>
<td>0.5</td>
<td>ON</td>
<td>NS</td>
<td>2</td>
<td>4</td>
<td>1</td>
<td>4</td>
<td>60</td>
<td>4</td>
<td>4</td>
<td>70</td>
<td>50</td>
<td>2</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>5</td>
<td>S</td>
<td>DEV</td>
<td>1</td>
<td>ON</td>
<td>S</td>
<td>3</td>
<td>3</td>
<td>1</td>
<td>3</td>
<td>25</td>
<td>3</td>
<td>3</td>
<td>40</td>
<td>50</td>
<td>3</td>
<td>3</td>
<td>2</td>
<td>4</td>
<td>3</td>
</tr>
<tr>
<td>7</td>
<td>S</td>
<td>DEV</td>
<td>1</td>
<td>ON</td>
<td>TD</td>
<td>2</td>
<td>4</td>
<td>2</td>
<td>3</td>
<td>15</td>
<td>4</td>
<td>4</td>
<td>30</td>
<td>20</td>
<td>3</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>3</td>
</tr>
<tr>
<td>9</td>
<td>D</td>
<td>OTH</td>
<td>0</td>
<td>ON</td>
<td>TD</td>
<td>1</td>
<td>4</td>
<td>1</td>
<td>1</td>
<td>5</td>
<td>1</td>
<td>3</td>
<td>30</td>
<td>40</td>
<td>3</td>
<td>2</td>
<td>2</td>
<td>3</td>
<td>2</td>
</tr>
<tr>
<td>11</td>
<td>D</td>
<td>DEV</td>
<td>1</td>
<td>ON</td>
<td>S</td>
<td>2</td>
<td>3</td>
<td>2</td>
<td>3</td>
<td>12.5</td>
<td>4</td>
<td>3</td>
<td>60</td>
<td>80</td>
<td>3</td>
<td>2</td>
<td>3</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>12</td>
<td>S</td>
<td>DES</td>
<td>1</td>
<td>ON</td>
<td>NS</td>
<td>4</td>
<td>3</td>
<td>2</td>
<td>4</td>
<td>15</td>
<td>3</td>
<td>3</td>
<td>50</td>
<td>55</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>4</td>
</tr>
<tr>
<td>14</td>
<td>S</td>
<td>DEV</td>
<td>1</td>
<td>ON</td>
<td>S</td>
<td>3</td>
<td>4</td>
<td>2</td>
<td>4</td>
<td>15</td>
<td>4</td>
<td>4</td>
<td>40</td>
<td>60</td>
<td>2</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>15</td>
<td>S</td>
<td>OTH</td>
<td>1</td>
<td>ON</td>
<td>TD</td>
<td>1</td>
<td>4</td>
<td>3</td>
<td>4</td>
<td>90</td>
<td>4</td>
<td>4</td>
<td>60</td>
<td>70</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>16</td>
<td>D</td>
<td>TST</td>
<td>1</td>
<td>ON</td>
<td>S</td>
<td>2</td>
<td>3</td>
<td>2</td>
<td>3</td>
<td>15</td>
<td>3</td>
<td>3</td>
<td>10</td>
<td>45</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>20</td>
<td>S</td>
<td>DEV</td>
<td>0</td>
<td>ON</td>
<td>S</td>
<td>1</td>
<td>3</td>
<td>2</td>
<td>3</td>
<td>15</td>
<td>3</td>
<td>3</td>
<td>40</td>
<td>60</td>
<td>1</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>Mean</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3.5</td>
<td>1.9</td>
<td>3.3</td>
<td>3.27</td>
<td>3.36</td>
<td>2.6</td>
<td>3.2</td>
<td>3.18</td>
<td>3.5</td>
<td>3.5</td>
<td>3.5</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SDEV</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.5</td>
<td>0.7</td>
<td>0.9</td>
<td>0.9</td>
<td>0.5</td>
<td>0.8</td>
<td>0.9</td>
<td>0.87</td>
<td>0.5</td>
<td>0.7</td>
<td>0.5</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 6.10

We can see that standard deviation is little high in this group, which is something we were expecting as these respondents are relatively less experienced with agile based software development methodology.

Respondent number ‘9’ is with almost no agile software development experience and they are may be new to this methodology so we could expect that they are not comfy with a methodology which is relatively new to them. Another point about respondent ‘9’ is that they are having five minutes of daily meeting in the morning and we could easily imagine that’s why respondent ‘9’ is responding with negative responses to various question. Firstly they are having very less experience and secondly only five minutes of daily meeting is literally too less.
6.13. More Then 1 Year of Agile Development Model Experience

In this group we have grouped together respondents with more than one year of experience in agile software development model. Please find table 6.11 with all respondents who have more than one year of agile development model experience.

<table>
<thead>
<tr>
<th>No</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
<th>11</th>
<th>12</th>
<th>13</th>
<th>14</th>
<th>15</th>
<th>16</th>
<th>17</th>
<th>18</th>
<th>19</th>
<th>21</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>S</td>
<td>DEV</td>
<td>2</td>
<td>ON</td>
<td>S</td>
<td>2</td>
<td>4</td>
<td>2</td>
<td>4</td>
<td>30</td>
<td>3</td>
<td>4</td>
<td>70</td>
<td>90</td>
<td>3</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>4</td>
<td>S</td>
<td>PM</td>
<td>2</td>
<td>ON</td>
<td>S</td>
<td>0</td>
<td>4</td>
<td>2</td>
<td>2</td>
<td>25</td>
<td>3</td>
<td>3</td>
<td>25</td>
<td>10</td>
<td>3</td>
<td>2</td>
<td>3</td>
<td>3</td>
<td>4</td>
</tr>
<tr>
<td>6</td>
<td>S</td>
<td>DEV</td>
<td>3</td>
<td>ON</td>
<td>NS</td>
<td>0.5</td>
<td>4</td>
<td>2</td>
<td>3</td>
<td>10</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>8</td>
<td>S</td>
<td>OTH</td>
<td>2</td>
<td>OFF</td>
<td>S</td>
<td>1</td>
<td>3</td>
<td>2</td>
<td>2</td>
<td>45</td>
<td>3</td>
<td>4</td>
<td>60</td>
<td>40</td>
<td>3</td>
<td>1</td>
<td>3</td>
<td>3</td>
<td>4</td>
</tr>
<tr>
<td>10</td>
<td>S</td>
<td>DEV</td>
<td>2</td>
<td>OFF</td>
<td>TD</td>
<td>1</td>
<td>4</td>
<td>1</td>
<td>4</td>
<td>45</td>
<td>4</td>
<td>4</td>
<td>80</td>
<td>30</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>13</td>
<td>S</td>
<td>DES</td>
<td>3</td>
<td>ON</td>
<td>NS</td>
<td>2</td>
<td>3</td>
<td>1</td>
<td>4</td>
<td>17.5</td>
<td>4</td>
<td>3</td>
<td>20</td>
<td>25</td>
<td>3</td>
<td>4</td>
<td>3</td>
<td>3</td>
<td>4</td>
</tr>
<tr>
<td>17</td>
<td>D</td>
<td>TST</td>
<td>5</td>
<td>ON</td>
<td>S</td>
<td>4</td>
<td>4</td>
<td>3</td>
<td>4</td>
<td>20</td>
<td>3</td>
<td>3</td>
<td>15</td>
<td>30</td>
<td>2</td>
<td>4</td>
<td>3</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>18</td>
<td>S</td>
<td>TST</td>
<td>3</td>
<td>ON</td>
<td>S</td>
<td>4</td>
<td>4</td>
<td>3</td>
<td>4</td>
<td>15</td>
<td>3</td>
<td>3</td>
<td>4</td>
<td>0</td>
<td>0</td>
<td>2</td>
<td>4</td>
<td>3</td>
<td>4</td>
</tr>
<tr>
<td>19</td>
<td>S</td>
<td>DES</td>
<td>2</td>
<td>ON</td>
<td>S</td>
<td>5</td>
<td>4</td>
<td>2</td>
<td>3</td>
<td>15</td>
<td>3</td>
<td>3</td>
<td>10</td>
<td>20</td>
<td>2</td>
<td>4</td>
<td>3</td>
<td>4</td>
<td>4</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>No</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
<th>11</th>
<th>12</th>
<th>13</th>
<th>14</th>
<th>15</th>
<th>16</th>
<th>17</th>
<th>18</th>
<th>19</th>
<th>21</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>3.8</td>
<td>2</td>
<td>3.3</td>
<td>3.2</td>
<td>3.4</td>
<td>2.9</td>
<td>3.4</td>
<td>3.33</td>
<td>3.7</td>
<td>3.9</td>
<td>3.2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>0.4</td>
<td>0.7</td>
<td>0.9</td>
<td>0.4</td>
<td>0.5</td>
<td>0.8</td>
<td>1.1</td>
<td>0.5</td>
<td>0.5</td>
<td>0.3</td>
<td>0.7</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 6.11

Once again we have respondents with average standard deviation as these respondents have developed some skill while working with agile based software development methodology for more than a year.

Couple of negative responses came again from respondents who are working with off-shore software development. Though we are not investigating agile software development challenges with respect to on-shore/off-shore software development but still we’ll include this discussion in our next section.
6.14. Greater then or equal to of 2 years traditional and agile experience

This is a hybrid group which we have created at the end of data analysis section to analyze data in more detail. Here we have grouped all the respondents with fair bit of experience. Table 6.12 have all those respondents who have more then or equal to two years of experience in both agile and traditional software development experience. Please find table below.

<table>
<thead>
<tr>
<th>No</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
<th>11</th>
<th>12</th>
<th>13</th>
<th>14</th>
<th>15</th>
<th>16</th>
<th>17</th>
<th>18</th>
<th>19</th>
<th>21</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>S</td>
<td>DEV</td>
<td>2</td>
<td>ON</td>
<td>S</td>
<td>2</td>
<td>4</td>
<td>2</td>
<td>4</td>
<td>30</td>
<td>3</td>
<td>4</td>
<td>70</td>
<td>90</td>
<td>3</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>13</td>
<td>S</td>
<td>DES</td>
<td>3</td>
<td>ON</td>
<td>NS</td>
<td>2</td>
<td>3</td>
<td>1</td>
<td>4</td>
<td>18</td>
<td>4</td>
<td>3</td>
<td>20</td>
<td>25</td>
<td>3</td>
<td>4</td>
<td>3</td>
<td>3</td>
<td>4</td>
</tr>
<tr>
<td>17</td>
<td>D</td>
<td>TST</td>
<td>5</td>
<td>ON</td>
<td>S</td>
<td>4</td>
<td>4</td>
<td>3</td>
<td>4</td>
<td>20</td>
<td>3</td>
<td>3</td>
<td>15</td>
<td>30</td>
<td>2</td>
<td>4</td>
<td>3</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>18</td>
<td>S</td>
<td>TST</td>
<td>3</td>
<td>ON</td>
<td>S</td>
<td>4</td>
<td>4</td>
<td>3</td>
<td>4</td>
<td>15</td>
<td>3</td>
<td>4</td>
<td>0</td>
<td>0</td>
<td>2</td>
<td>4</td>
<td>3</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>19</td>
<td>S</td>
<td>DES</td>
<td>2</td>
<td>ON</td>
<td>S</td>
<td>5</td>
<td>4</td>
<td>2</td>
<td>3</td>
<td>15</td>
<td>3</td>
<td>3</td>
<td>10</td>
<td>20</td>
<td>2</td>
<td>4</td>
<td>3</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>Mean</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SDEV</td>
<td>0</td>
<td>0.84</td>
<td>0.45</td>
<td>0.45</td>
<td>0.55</td>
<td>0.55</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 6.12

Probably we could see least variance in responses in table 6.12, thanks to experience of respondents. There is 100% concurrence among respondents when asked about requirements engineering and overall impact related question. All of the respondents agreed that it is easy to handle requirements in agile based software development and also they agreed that it was a good move to switch to agile based software development model.

Training/mentoring has been a controversial question in our question in a sense that we haven’t seen harmony of respondents on this question and once again we can see that standard deviation in 0.84 in table 6.11. Respondents in above given table are also not mutually concurred on long-term effect of agile based software development.
7. Guidelines for agile adoption

After going through all the processes during our thesis work, we believe following guidelines would benefit organizations intending to perform a transition from traditional waterfall methodology to an agile based Scrum development.

1. First and most important point for transitioning to Scrum is the training of employees. A comprehensive training session should be conducted for all concerning employees and other important stakeholders who are going to participate in Scrum based projects. There are certified Scrum experts available for this purpose. Training sessions can be interactive so that people can ask and clear any confusion after the future development model. This way, employees are prepared mentally for the change and they can really see the benefits of new development model. As seen in analysis section of this report subjects with lesser experience think that there should be more training.

2. ScrumMaster is a very important position in Scrum development. ScrumMaster is the real person that coach and guide the rest of development team. This position should be given to a person well familiar with the methodology and has a good record with it. So if a project manager in waterfall method going to take over the position of ScrumMaster then he/she need to go through a lot of changes and should understand the underline differences between two positions. There are certified ScrumMaster classes and training sessions available and it should be attended by those prospective ScrumMasters. In interviews we got the impression that managers are becoming ScrumMasters but they are not pleased with their new role. Mentoring for ScrumMasters can certainly improve the situation.

3. Scrum is about self organizations. Every one participating in Scrum development should be given enough space and opportunities to self organize themselves. There are certain indications from industry that inexperienced team members are still not confident enough when it comes to self-organization. There would be no use if same old methods are followed rather each member help each other to self organize instead following defined criterion.

4. More meetings and more communication play a key part in Scrum development. Organizations should create an environment where more and more interaction is possible among the members. One example can be like arranging a place for combined lunch for all employees and it is made obligatory for all to participate. This way communication and interaction would improve which is essential for Scrum based development.
Subjects with job title other than developer and designer think that there is no effective communication taking place among team members. Subjects with less experience in traditional software development model are also not satisfied with level of effective communication. During an interview we have been told by an industrial researcher that after spending quite some time with Scrum, now, their organization is going to have big rooms to accommodate whole team members for a project. Earlier most of team members were not in a single room. Such practice will be helpful to achieve desired results.

5. Old methods and practices should be discouraged once transition is made from waterfall to Scrum. During our research we came to know the fact that organizations that have shifted to Scrum did not benefit completely because its employees still continue to practise old methods of detailed documentation and practically did not follow the Scrum practises. This check should be maintained by ScrumMaster that all members actually implementing new methods and discarding old techniques. We have been told by an industrial researcher that there is a joke in their organization, “Make a copy of document before wasting it”. Un-necessary documentation must always be discouraged as it is against basic Scrum principle.

6. Asking questions at any time and at any place should be facilitated and answering them should be obligatory for experienced and senior members in Scrum teams. Scrum greatly encourages knowledge sharing among the members and this is only possible when questions of the employees will be entertained in the best way. In waterfall, because of its mechanical nature employees find it harder to interrupt senior members for queries and technical problems they face.

7. During a sprint, all team members are obliged to focus over the tasks only involved with the sprint. A member of a sprint team should never do any task outside sprint goals. In organizations, because of bureaucratic environment senior management may ask for other tasks to be completed on priority bases. This attitude is against the basic Scrum principals and should be avoided in order to get benefit from a full fledge Scrum implementation for an organization.
8. Discussion

The rise in demand for Scrum implementation in software organizations around the globe was a motivating factor for us to start our work in this direction. Many organizations who would like to do the transition are familiar with traditional sequential models like waterfall. There are enough written material available on this matter but it was noted that organization struggle to go the change. We went through documents study on the topic of comparison between waterfall and Scrum development, challenges faced during transitioning and guidelines for the smooth transition.

In our literature study we analysed the basic underline differences between the two methods. It is of great importance for organizations to first know that how these two methods are so diversifying and mostly contradicting each other. Going for Scrum implementation from waterfall is not meant a light change in development practices but indeed it is a total change of attitude and management structure which causes a considerable change in the development methods. These were thoroughly highlighted with the help of proper references to research articles.

Scrum structure and mechanism has been highlighted in a manner it would be easier for readers to comprehend basic concepts of it. Careful comparison between the two models is made with the help of research material. Pros and cons of both development models are also highlighted so that readers get a clear picture of the two.

This was later followed by an online questionnaire and at the end a special interview was organized with the field related person from the industry.

Responses received from our online questionnaire give valuable information about differences between waterfall and Scrum based development methodologies. Most respondent considered the change towards agile based development a very healthy and productive keeping in view the current market trends. This shows the competitiveness of today’s software market and its impact over organizations. Majority of the respondents to the online questionnaire have also emphasized the need for proper training before implementing the new development methodology. No one denies the importance of training prior the transitioning phase but taking the training lightly may lessen the number of expected advantages of proposed new methodology.

In the comments section of questionnaire, respondents have highlighted the advantages and disadvantages of Scrum methodology. The fact that understanding of agile method is very crucial as some time concepts of agile methods are not fully understood by all in the organization while
implementation is started. It results in confusion and full advantages of the new agile methodology are not felt in organization.

In the online questionnaire, another issue raised by correspondent was the task division among team members while following Scrum. As the methodology is very much about self organization so it is harder to assign tasks to all members according to their skills and ability which they can handle easily. Task assignments in a team need a very high level of competence so if this competence is lacking then employees may find it harder to carry out the assigned jobs.

As mentioned in the analysis section of this report there are issues with adoption of agile based methodologies in industry. The same point has been highlighted in the responses to the online questionnaire. These issues can be eliminated or reduced if we can somehow locate causes of problem. In our research we have analyzed data in various perspective i.e. view of respondents on pre-adoption training, quality of software, productivity, communication, testing, customer satisfaction and ability to handle changing requirements.

What our investigation revealed is that pre-adoption training and mentoring is most common problem. Most of respondents do think that there is need of more comprehensive training and mentoring. This argument can also be strengthened by another piece of information mentioned in analysis section of report that more the experience of agile is, more they are satisfied by agile methodology. To some extent alternative of experience can be training in agile methodology. Another issue that came across quite often during our analysis was effective communication, though it could be managed through effective training and mentoring but still scrum master has to play a role here and should provide an environment which encourages communication. At that point we will propose that agile methodology should not be put into practice at once, instead, one must carefully pick few most relevant agile principles first and let team/organization get used to of it.

We were lucky enough to have got the opportunity of interviewing a researcher, who is working for a reputed international organization, about Scrum implementation in the organization. He was also working on an investigation that deals with overall impact of agile adoption in his organization. He was working on the same lines as to see the impacts of Scrum implementation hence we didn’t design any structural questions. Straight away we asked him to share his experiences and study with us. The interview lasted for one hour duration where the transitioning of the organization to Scrum was explained and all the after effects were elaborated by him.

He explained that initial trainings were given to the employees prior to making the transitioning from sequential model to agile based Scrum. He also explained how employees were still using all those documentations and
methods of old technique even after the transitioning. This was a reason that Scrum was not fully visible in the organization despite of few months after the transitioning period. The organization was seeking further ways to train and implement Scrum in its full entirety. They were indented to move the process more gradually and hence evaluation of the new process was going on and the impacts of those new practices were noted down.

8.2. Validity Threats

Despite of all the hard work done by researchers, there are always factors which have some sort of influence on the outcome of a research report and same goes with our thesis. [19] [20] [21] have also discussed such factors which have ability to interfere with a research study. There are different categories of validity threats like concurrent validity, predictive validity, convergent validity, criterion-related validity, internal/external validity [21].

We have tried to minimize validity threats to our research by designing a hybrid research methodology. Our conclusion is not solely based on quantitative research i.e. questionnaire, statistical but also fair bit of qualitative research is conducted i.e. literature review. We have selected well-known sources for our literature review. Still, in our case there are a few validity threats which should be considered while reading this thesis report.

8.2.1. Sample size/ number of responses

We received twenty responses of the questionnaire which could be lower limit of acceptance level. This could pose a generalizability threat but most of the time standard deviation value for data was not that big, so we are quite confident that our sample is not scrawny enough to be easily overlooked.

8.2.2. Content/ data validity

We have taken responses of our questionnaire from various professionals. Some of them were busy enough that we are not sure of the legitimacy of their responses as they might have filled questionnaire in a rush.

8.2.3. Subject’s profile

Ideal subject (respondent) for our research was one who has practical exposure of both traditional and agile software development methodologies. But there were couple of respondents who were not having any experience in traditional/agile based software development methodology and their response is based on academic knowledge.
8.2.4. Subject’s background

There is lot of variation in experience of our respondents ranging from 0 years to 5 years. This could be the reason behind high standard deviation value in some columns i.e. standard deviation value of 0.98 and 0.86 for column 16 and 10 respectively in table 6.1.
9. Conclusion

Software based organizations around the globe uses different methodologies for their software development. Every organization has their own set of priorities and ambitions. Waterfall development model, after ruling the industry for so long has now seen a diminution and people have started shifted to more agile based development due to competitive market and requirements instability.

In this thesis document, we have focused over the transitioning from Waterfall model to Scrum based agile model. To make this process easy and understandable, we went through the differences between the two software development methods. On the bases of those differences, challenges are highlighted which come when the transitioning process starts. As both techniques vary significantly in approaches so we have applied various means for gathering information. Earlier in our thesis work we studied a number of research articles and books on relevant topic and then we followed by an online questionnaire sent to different companies working in the same fields. We were lucky enough to get observations from industry. By following this procedure, we were able to draw our guidelines for a smooth transitioning and to avoid any impediment along the way.

This empirical study has strengthened some of the literature work, in the same time it also provided new impression that designers and developers are more confident with Scrum as compare to other job functions. Software designers have a strong feeling that Scrum has positive impact on software quality. Most of the subjects do suggest that better training could have positive impact on their work. We have seen a link between experience and confidence level for subjects, they are directly proportional to each other. Respondents with lesser experience in agile methodology think that it is difficult to manage testing in agile based software development model.

We have concluded that organization should not adopt Scrum or any other agile methodology in rush. Gradually implementation of Scrum principles along with proper training, and Scrum friendly corporate culture will increase success rate.

This document can prove very helpful for management and team leads who want to undergo a transitioning of development model. This work is not such research which only becomes part of a bookshelf; it could serve as white paper in industry as it directly deals with an issue faced in many organizations concerning the adaptation of Scrum model.

We do recommend a further study on this topic by following the same hybrid research methodology to substantiate outcome of this report.
10. References


11. Appendix 1: Questionnaire

Online Questionnaire
Answers marked with a * are required.

1. A study in agile development methodology

This questionnaire is designed to identify challenges in agile based development methodology, especially after having a recent transition from traditional (waterfall etc) methodology. Your contribution in this study will be highly appreciated. This survey is for people who are currently working with agile methodologies.

Your identity will be kept confidential and all the information gathered by you will be kept completely confidential and data will be presented in a way that you identity will not be connected with published data in any sense.

This questionnaire will take only approximately 5 minutes so please look at your schedule according to it before starting with questionnaire. Please fill this questionnaire asap and in-case of any ambiguity please contact any of following persons.

Wasim Tahir (Student)
wata06@student.bth.se
0046-739-300160

Junaid Naseem (Student)
juna06@student.bth.se
0046-739-795485

Bengt Carlsson (Supervisor)
bengt.carlsson@bth.se
0046-708-385813

1. Where is your site situated? *
☐ Sweden  ☐ China  ☐ India  ☐ Denmark

2. What is your role? *
☐ Project manager  ☐ Developer  ☐ Designer  ☐ Tester  ☐ Other

3. How many years have you worked with agile methodologies? *

☐

4. Type of software development at your site is? *
☐ Onshore development  ☐ Offshore development
5. Current development methodology in your organization is *
   - SCRUM
   - Not SCRUM but close to it
   - Totally different

6. If in question 5 your answer is totally different please specify your development model a bit more

7. How many years have you worked with traditional methodologies like Waterfall development model? *

8. While switching from traditional to agile development methodology, employees response at your organization was!
   - Good
   - Partly Good
   - Partly Bad
   - Bad

9. Was enough training given to employees before switching to agile development methodology? *
   - Yes
   - To some extent
   - No

10. Do you think there is effective communication between team members in agile based development methodology?
    - Disagree
    - Partly disagree
    - Partly agree
    - Agree

11. In agile based development methodology of your organization, daily meeting lasts for? Answer is in minutes *

12. Do you think that productivity of developers has increased /decreased after switching to agile based methodology? *
   - Increased
   - Partly increased
13. Agile development method has very good/bad impact on software quality? *
- Good
- Partly Good
- Partly Bad
- Bad

14. How much automated testing was performed before and after switching to agile based methodology? Please write in percentage (approximately) *

<table>
<thead>
<tr>
<th>Before</th>
<th>After</th>
</tr>
</thead>
</table>

15. Testing in agile based development methodology is very easy/difficult to manage *
- Easy
- Partly easy
- Partly difficult
- Difficult

16. Its more convenient to handle changing requirements using agile development methodology. *
- Disagree
- Partly disagree
- Partly agree
- Agree

17. In your new agile method requirements testability and traceability have become more easier? *
- Disagree
- Partly disagree
- Partly agree
- Agree

18. Customer satisfaction level has increased after switching to agile development methodology? *
- Disagree
- Partly disagree
- Partly agree
- Agree

19. After carefully analyzing whole impact of switching to agile development methodology do you think that it was a good move to switch from traditional development model to agile? *
- Disagree
- Partly disagree
- Partly agree
- Agree
20. Do you think that it is fine to apply SCRUM or other agile development methodologies on the following systems? Please mark multiple alternative if necessary *

- Real-time Systems
- Security critical systems
- Safety critical systems

21. You think that switching of roles in agile based development team can be a good option in long run? *

- Disagree
- Partly disagree
- Partly agree
- Agree

22. Please specify the three most important advantages of agile development model for your organization. Your response for this question is highly appreciated.

23. Please specify the three most important disadvantages of agile development model for your organization. Your response for this question is highly appreciated.

24. Any other think which you might want to add here