

*Master Thesis*  
*Computer Science*  
*Thesis no: MCS-2009-37*  
*September 2009*



# **Quality Assurance Activities in Agile**

**- Philosophy to Practice**

**Malik Imran Ullah**  
**&**  
**Waqar Ali Zaidi**

School of Computing  
Blekinge Institute of Technology  
Soft Center  
SE-37225 RONNEBY  
SWEDEN

This thesis is submitted to the School of Computing at Blekinge Institute of Technology in partial fulfillment of the requirements for the degree of Master of Science in Computer Science. The thesis is equivalent to 20 weeks of full time studies.

**Contact Information:**

Authors:

**Malik Imran Ullah**

malikimran62@hotmail.com

**Waqar Ali Zaidi**

waqarali12@gmail.com

University advisors:

**Jeff Winter**

Jeff.winter@bth.se

**Department of Interaction and System Design**

**Kari Rönkkö**

Kari.ronkko@bth.se

**Department of Interaction and System Design**

School of Computing  
Blekinge Institute of Technology  
Soft Center  
SE-37225 RONNEBY  
SWEDEN

## ABSTRACT

Quality assurance activities, in software development, are the backbone of whole project. These activities are not only responsible of product quality, but also for process quality. In conventional software development QA is a separate group of QA experts. As the trends of software development moved towards agile development, QA activities also got changed. In agile development most of these activities are performed by developers. Close people collaboration, onsite customer and Test Driven Development are the approaches in agile development to achieve better product quality. In this thesis we have presented the philosophical as well as practical angle QA in agile development. Mindset of agile development revolves around product quality but there is much work to be done to impart quality of process in agile development to get it standardized and more organized. QA activities remain centric and focused to testing. Practices like SPI and following some standards are lacking in agile methodologies. In this thesis we have proposed to inject an extra layer of QA in agile projects. Purpose of injecting an extra layer, is to use the knowledge of QA experts to achieve quality in development process that will result in higher level of product quality.

**Keywords:** Quality Assurance (QA), Agile Development, Software Process Improvement (SPI)



## **ACKNOWLEDGEMENT**

IN THE NAME OF ALLAH THE MOST BENEFICENT, THE MOST MERCIFUL

We would like to dedicate this thesis to our parents, who are the absolute personification of love and kindness towards us. Our honest regards are for our institution and thesis supervisor who helped, supported and guided us in order to complete this work. Two shining names of Mr. Naveed Akhtar and Mr. Imran Iqbal will always reserve a handsome share of our achievements. At the end, we thank them all from the depths of our heart.

# TABLE OF CONTENTS

<b>ABSTRACT.....</b>	<b>III</b>
<b>ACKNOWLEDGEMENT.....</b>	<b>IV</b>
<b>INTRODUCTION.....</b>	<b>1</b>
<b>CHAPTER 1: BACKGROUND.....</b>	<b>2</b>
1.1 SQA UNDER CONVENTIONAL SOFTWARE DEVELOPMENT.....	2
1.2 AGILE SOFTWARE DEVELOPMENT.....	4
1.2.1 SQA under Agile Development.....	5
<b>CHAPTER 2: PROBLEM DEFINITION.....</b>	<b>7</b>
2.1 DOMAIN OF STUDY.....	7
2.2 CHALLENGES, SQA FACING IN AGILE DEVELOPMENT.....	7
2.3 GOALS.....	8
<b>CHAPTER 3: METHODOLOGY.....</b>	<b>9</b>
3.1 LITERATURE REVIEW.....	9
3.2 QUALITATIVE APPROACH.....	9
3.2.1 Informal Interviews.....	10
3.2.2 Industrial Survey.....	10
3.3 DATA VALIDITY.....	11
3.3.1 Credibility.....	11
3.3.2 Transferability.....	12
3.3.3 Dependability.....	12
3.3.4 Confirmability.....	13
<b>CHAPTER 4: AGILE DEVELOPMENT AND QUALITY.....</b>	<b>14</b>
4.1 AGILITY.....	14
4.1.1 Extreme Programming (XP).....	15
4.1.2 Scrum.....	16
4.1.3 Crystal Clear Methodology.....	17
4.2 NEW DEVELOPMENT APPROACH IN ORGANIZATIONS.....	18
4.3 SPREADING TREND.....	18
4.3.1 The Bright Side of Agile Development.....	19
4.3.2 Challenges Adopting Agile Development.....	20
4.4 PHILOSOPHICAL ANGLE OF QUALITY IN AGILE DEVELOPMENT.....	20
4.4.1 Quality of Product.....	21
4.4.2 Quality of Product and Testing.....	21
<b>CHAPTER 5: AGILE QUALITY ASSURANCE, APPLIED.....</b>	<b>23</b>
5.1 AGILE TESTING APPROACH.....	23
5.2 PAIR PROGRAMMING.....	24
5.3 REFACTORING.....	24
5.4 SPI AND AGILE METHODOLOGY.....	25
5.4.1 Problem Affiliations.....	25
5.5 PROPOSED SOLUTION.....	26
5.5.1 Extra Layer of QA.....	27
5.6 THE NEED OF EXTRA LAYER OF QA.....	27
<b>CHAPTER 6: SURVEY AND RESULTS.....</b>	<b>29</b>
6.1 QUESTIONNAIRE STRUCTURE.....	29
6.2 QUESTIONNAIRE DISTRIBUTION.....	30
6.3 DATA VALIDATION.....	30

6.4 ANALYSIS.....	34
6.5 ANALYSIS WITH RESEARCH QUESTIONS.....	35
6.6 APPLYING EXTRA LAYER OF QA.....	36
6.6.1 <i>Expected Disadvantages</i> .....	38
6.7 RESULTS.....	38
<b>CHAPTER 7: CONCLUSION.....</b>	<b>39</b>
<b>CHAPTER 8: FUTURE WORK .....</b>	<b>40</b>
<b>REFERENCES.....</b>	<b>41</b>
<b>APPENDIX.....</b>	<b>45</b>

## **TABLE OF FIGURES**

<b>FIGURE 1: SINGLE ITERATION OF AGILE DEVELOPMENT PROCESS [12] .....</b>	<b>5</b>
<b>FIGURE 2: NATURE OF AGILE METHODOLOGIES [13] .....</b>	<b>5</b>
<b>FIGURE 3: METHODOLOGICAL HIERARCHY .....</b>	<b>13</b>
<b>FIGURE 4: TEST-FIRST DEVELOPMENT .....</b>	<b>23</b>
<b>FIGURE 5: ORGANIZATIONAL VIEW OF AGILE PROJECT AFTER INJECTING EXTRA LAYER OF QA. ....</b>	<b>37</b>

# INTRODUCTION

Software Quality Assurance (SQA) stands on high level of importance in today's software industry. We can find sufficient research work carried out in this field. SQA is interpreted in different ways and words. NASA Software Quality Assurance Center describes SQA, "Software Quality Assurance (SQA) is defined as a planned and systematic approach to the evaluation of the quality of and adherence to software product standards, processes, and procedures" [1]. Ultimate purpose of quality assurance is to attain better quality in software product. Different approaches and several quality models are followed in this discipline. SQA activities are practiced during project and these activities include process control, documentation, audits and verification and validation.

Current findings seem to express that plan oriented projects or the projects following conventional development process, are influenced under SQA activities. Agile development methodology has changed the scenario of conventional software development. Manifesto for Agile Software Development [2] emphasizes on close team collaboration, quality and relationship between developer and customer. According to Miller, one of the characteristics of agile software process is "People-oriented, i.e. agile processes favour people over processes and technology" [3]. So, if agile development methodology is more people centric, then the effective role of SQA in projects that adopt agile development; needs to be more comprehended.

Sajid Ibrahim Hashmi and Jongmoon Baik carried out a comparison between XP and spiral model and the focus was on quality assurance. They claim that in agile development, developers may also be responsible for QA activities [4]. Agile development projects consist of short iterative development and release of product. And projects, following agile development, evolve around the developer and customer who are responsible to maintain product quality [5]. If responsibility for quality, in agile development, is shifted on customer and developer, then the supporting role of QA must be identified. SQA is not only responsible for a particular project but also maintain the processes and culture of organization. In [7] User Experience Design (UXD) team approach is introduced collaborating with developers, this UXD team approach seems an attempt to redefine and replace the role of SQA in agile projects. Main focus of this thesis work is to highlight some gaps in agile SQA activities and to put forward suggestions for improvement.



# CHAPTER 1: BACKGROUND

Ultimate target of producing services or products is to satisfy the customer. Quality of product or service can be directly related to customer's satisfaction. To attain the required level of quality, certain standards for product or service must be specified. Several procedures and processes are followed to meet the specified and required standards. Besides achieving customers' satisfaction organizations also target their competitive advantage in industry. To earn good-will and profitability in the industry, organization needs continuous quality improvement. Quality management (QM) is essential and vital for organization. Quality management is a process of ensuring that required level of quality is achieved not only in products but also quality in the process through which these products are produced. It involves defining some appropriate quality standards and procedures to ensure that these are followed. The aim should be to develop a 'quality culture' where achieving quality is seen as everyone's responsibility. Quality assurance, quality planning and quality control are the activities which are involved in quality management. [8]

In software industry the concept of quality is expressed and defined in many different ways, some definitions focus on error-free functionality of software product whereas some definitions are found to emphasize on customer satisfaction, but it is really a hard deal to cite an absolute definition of quality even after going through the literature. If we conceptualize quality by understanding the term "utility" from economics, it could be easier to develop meaning, clear thoughts about quality. As the utility is considered a state or measurement of satisfaction that someone attains by the use of goods, in the same manner quality of software may also be conceptualized as the state of satisfaction under particular environment or mean of use. There are numerous attributes that can be encompassed to formulate a comprehensive quality definition; therefore to look what causes the lack of quality might lead to better understanding as in [9] we find Genichi Taguchi's narrating, "*The lack of the quality is the losses a product imparts to the society from the time the product is shipped*". If, in this definition, we prefix word "software" with the word "product" and replace "shipment" with "deliver" for our understanding, it would provide us with broader and specific vision of quality regarding software development and product. But we cannot classify *quality* in different school of thoughts regarding quality; it seems all about your perception on the basis of your experience or understanding.

For continuous improvement, continuous evaluation and monitoring is required. Product evaluation and process monitoring are the core responsibilities of software quality assurance (SQA) during software development. SQA also monitors and evaluates the quality of:

- Standards
- Processes
- Procedures<sup>1</sup>

Conventional and agile are two methodologies that may be adopted for software development. Both methodologies are different in approach. They deal with SQA in different ways, accordingly.

## 1.1 SQA under Conventional Software Development

In software engineering, conventional software development is referred as the methodologies that are process oriented in nature. Structure of software development process or software development life-cycle is followed to develop a software product, strictly in conventional development. Predefined activities are practiced and these activities include: [1]

- Requirement gathering and Requirement Analysis
- Specification
- Architecture

---

<sup>1</sup> Procedures are the established criteria to which the development and control processes are compared [1]

- Designing
- Development
- Testing and Implementation
- Maintenance

There are numerous models for software development that are process oriented in approach. Each model defines its own steps, processes, procedures and activities to be followed under a software development life-cycle. While discussing about conventional software development, Waterfall model and Spiral model may be specified as more mature and practiced. In conventional software development customer interacts with requirement engineers in order to specify the requirements. Decisions policies and standard are defined on managerial level of an organization.

Quality assurance activities works as monitoring and evaluating entities under any software development model. These activities focus on the quality of product/service and process as well. Besides all philosophies of quality there are several quality models. These quality models provide us with quality characteristics, attributes and factors which are supposed to be targeted during software development. McCall's Quality Model presented in 1977 [10] and Boehm's Quality Model presented in 1978 [11], these quality models formed the foundation for today's quality models. Although to pinpoint or to count the quality characteristics seems like an infinite loop, but there are some of the common quality attributes and characteristics and those can be listed as:

- Correctness
- Reliability
- Integrity
- Usability
- Efficiency
- Maintainability
- Testability
- Interoperability
- Flexibility
- Reusability
- Portability
- Clarity
- Modifiability
- Documentation
- Resilience
- Understandability
- Validity
- Functionality
- Generality
- Economy

As software industry is growing, software products are getting more complex and users' demands are increasing. These factors are, ultimately, increasing the complexity of software development projects. To tackle mounting complexities several models and standards are being followed. Some of the most practiced models are

- ISO 9000
- ISO 9126
- CMM (Capability Maturity Model)
- CMMI (Capability Maturity Model Integration)
- SPICE (Software Process Improvement Capability dEtermination )

- Six Sigma

Besides targeting quality of product/service, these models also focus on quality of software development process on organization level. But eventually, the responsibilities of QA are being increased. QA activities have gained the importance of back bone in an organization. These activities are responsible for development process, quality of product and these activities keep the project on track. On managerial level, standards and procedures are established for software development. The role of SQA is to assure that defined standards are documented properly, procedures are followed. For this purpose, product evaluation is conducted, audits and meetings take place to monitor and evaluate that processes are following defined procedures. Solely, on product development side, verification and validation is core SQA responsibility and activity to maintain and evaluate product quality. SQA conducts formal software reviews, software inspection and documentation reviews in order to monitor software development. SQA assures that software quality is meeting the required standards of quality and requirements. Quality assurance is vital in all phases of software development process to achieve higher and required level of quality, in terms of product and process. For assurance there are numerous tools that are used. These tools include audits, inspection checklists, metrics and automated code standard analyzers. [1]

SQA activities have gained dramatic importance. To imagine a successful project without the involvement of SQA is almost impossible. Efficient quality assurance is the key to successful project. Besides, all these facts SQA activities are becoming more and more complex. Although, these activities lead an organization towards higher level of quality, but; these activities consume sufficient resources, time and effort. Heavy documentation, step by step evaluation and monitoring is a rigorous process.

## 1.2 Agile Software Development

Agile software development approach has changed the way of software development, entirely. Unlike, conventional development approach, agile prefers short iterations during software development process. This approach is short iterative, incremental and people-centric. While defining agile development, manifesto for agile software development must be the starting point. It is stated [2]:

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

***Individuals and interactions*** over processes and tools

***Working software*** over comprehensive documentation

***Customer collaboration*** over contract negotiation

***Responding to change*** over following a plan

*That is, while there is value in the items on the right, we value the items on the left more.[2]*

Agile software development encourages people collaboration through the project. As compared to conventional way of software development, it responds to change efficiently as it is incremental and iterative.



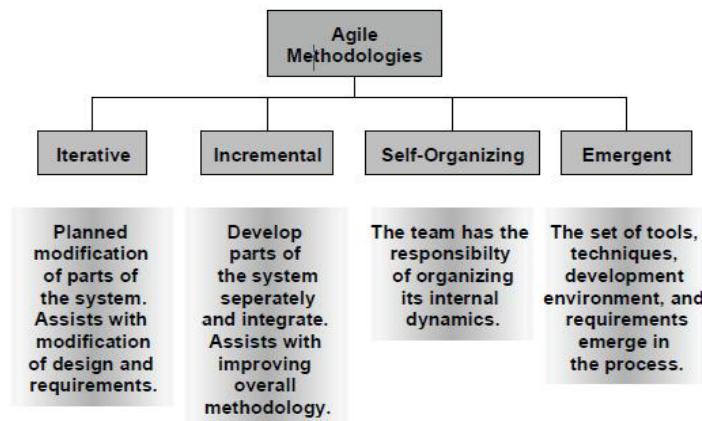
**Figure1: Single Iteration of Agile Development Process [12]**

Figure 1 available at [12] represents a single iteration of agile development approach. As we can observe that in, one short, iteration all required phases of software development (Plan, Design, Development, Test and Analysis) takes place, after one another. After the completion of, one, iteration, the software product is sent for users'/customers' review. The next iteration starts to add required functionality according to feedback of users'/customers' review.

There are many agile methodologies been introduced but commonly used agile methods are:

- Extreme Programming (XP)
- Scrum
- Crystal

In nature all agile methodologies have similarities. Figure 2 shows the nature of agile methodologies extracted from [13].



**Figure 2: nature of Agile Methodologies [13]**

Agile methodologies are known as light weight methodologies as they are not process intensive. Agile methodologies have also changed the way of SQA activities. Documentation is not too much heavy but only as customer/user requires.

### 1.2.1 SQA under Agile Development

Testing is integrated in each iteration that helps to shorten the release time and provides with better quality. Focal point of our study will remain the topic of SAQ activities in agile methodologies; so, all aspects related to the topic will be dug deeper in later chapters. We will have a look on the difference of view point between theories and practitioners.

Mindset was changed, when agile methodologies hit the software industry. In agile development, software industry observed dramatic changes in QA activities. Agile moved development from process oriented to people oriented. Testing is not the responsibility of a QA tester, developers are supposed to test their programmes and fix the bug whenever they find it. No more rigorous and time consuming testing phase is in practice. Bugs are fixed at the same stage they are found and no matter who finds them and fixes them. Close customer collaboration is merged to get effective and rapid feedback of developed system before starting a new iteration, to increase product quality.

As agile methodologies are appreciated for integrated testing approach, quick response to change and being people centric, some claims also prompted, stating that these methodologies are not mature enough to be practiced on large scale. We will discuss these issues in detail and will try to put forward improvement suggestions for few of the gaps in SAQ activities in agile.

## CHAPTER 2: PROBLEM DEFINITION

Quality software product has always been the core target of a company or a project and flawless product can be observed as the top most characteristic of quality in a software product. But to develop a software product of good quality, it is needed to develop it in a good way. Any software methodology is followed to develop software product in right way, accordingly. Quality assurance activities are driving authorities of quality in an organization. World of software engineering took a turn by moving from process oriented engineering to agile engineering. Agile development is introduced to be flexible and quick responding to change. And we have chosen the field of quality assurance in agile development. Besides discussing about how quality is achieved in agile development, we will also highlight and try to bridge some gaps in this area of interest.

### 2.1 Domain of Study

We would like to quote “The Agile methods radically alter the quality assurance landscape by moving responsibility for quality to the developers and customer and by defining a new supporting role for the QA professional. Much work remains to be done in fleshing out how this new landscape will work.” [14]. Within this field we limit our research work to given research questions:

1. What is the role of Software Quality Assurance activities in Agile Development, according to existing literature?
2. Identify gaps of agile iterative culture that can be improved regarding Quality Assurance activities and propose an approach to fill the gaps?
3. How do organizations practice SQA activities, while adopting agile development in projects?

Agile Development has changed the way of development in our projects. But this methodology cannot be considered as mature and practiced, as conventional development is. SQA activities monitor all processes in conventional development.

### 2.2 Challenges, SQA Facing in Agile Development

For true agile development quality is an inherent element and it is commonly said about agilists that they are *quality infected* and *test infected* [15]. By adopting more involvement of customer and stakeholders agile development has introduced new approach to meet the requirements. Being flexible and quick response to change also make this methodology more effective for development. Beside all of its effectiveness, yet, this methodology has been under critics and facing challenges to win the heart of practitioners following conventional development in software industry. During this limited time span we pointed out some challenges, agile development is facing in QA and in this section we discuss them, precisely.

On-site customer involvement and collaboration with developer is good approach to develop the system as it is required. According to [16], on-site customer’s availability makes evaluation of the system efficient and customer gives useful feedback when work is fresh in everyone’s mind. In this scenario it is ignored what if customer is not technically aware of quality parameters and attributes of the system and this might lead to misunderstanding about system. Moreover if customer is not available all the time on site then there must be someone fulfilling the need of customer to maintain the quality and agility throughout the project. Quick feedback is one of important element of agility and customer is supposed to provide with feedback after each iteration.

Agile software development came up with fresh approach to get rid of extensive documentation by minimizing it to least level of required documentation of product. Agility discourages extensive documentation but several practitioners have put forward their critics that documentation cannot be neglected. Agile development minimizes technical documentation but documentation is the part of

software product development [17]. *“Documentation cannot be traded off and should be factored into the cost of development as a high-level priority”* [18] documentation is needed in order to maintain software quality in future. It is the responsibility of QA to look after the documentation throughout the project. But if agile discourages extensive documentation and on the other hand documentation is thought to be essential then the role of QA is needed to be redefined, in agile.

Testing is a QA activity. In conventional development approach, testing is conducted as a separate and complete phase. Both, automated and manual testing techniques are in practice now days. Agile development changed the whole scenario of software testing. Agile shifted the responsibility of testing to developers by adopting the techniques like Pair Programming and Test Driven Programming. Agilists are considered more test infected. But does it mean that we have shorten the need QA testers, and programmers have become able to take the responsibility of testers as well? According to [19] a survey held in Motorola showed that programmers cannot deal with all kind of defects during development of software product. To overcome this problem a master thesis [21] was carried out in Blekinge Institute of Technology that proposed a Separate Testing Team in agile development. Our point of issue is to keep whole quality assurance under consideration rather than testing.

As software products are getting complex and as large manual systems are required to be automated, quality concerns and testing is getting rigorous. Although this research work is more influenced by agile approach but it is also a fact that numerous organizations are getting benefits from formal approaches while agile approaches are not widely spread. New ideas and concepts are merging in QA and all other elements discussed in this writing target to discuss QA in agile. As *software quality is beyond software testing, software testing is also beyond testing tools.*

## 2.3 Goals

Although, there are many claims that agile development is providing us with higher quality. But through comparing academic writings with practitioners', on agile SQA activities, we will suggest improvements for gaps of QA activities in agile projects. Feedbacks of questionnaire, from different organizations will be evaluated to understand, how organizations are practicing SQA activities in agile development projects, and opinions of respondents will help us to understand the limitations and benefits of agility in SQA, which is necessary to put forward the suggestions for improvement. We hope that our, this thesis, work will help to understand agile SQA activities and to put new idea for research, in this field. If research work is done continuously in our subjected topic, agile development can be proved more potential and productive regarding quality of product and work as well.

## CHAPTER 3: METHODOLOGY

Before starting any research work, according to [20] one must figure out what methodology, approach or action-plan will be followed that links methods to outcomes? According to Oxford definition (available at [22]) word *methodology* can be defined as “a system of methods used in a particular field”. To adopt any methodology for research is essential, in order to keep the study or research on track, measured, structured and organized. Most commonly used research methodologies or approaches are:

1. Qualitative Research Approach
2. Quantitative Research Approach
3. Mixed Research Approach

Research approach or methodology that engaged our study was Qualitative Approach. The purpose of using Qualitative study was to gather the available research work, literature and practitioners' opinion in the field of SQA, in agile projects. Having identified most critical issues in our field of study, we targeted to use these literature findings to construct and propose solution in order to overcome critical issues in agile SQA activities. Therefore our study can be classified as influenced by Qualitative research work.

### 3.1 Literature Review

Our research work is influenced by Qualitative approach; keeping this in view we conducted study of literature. Literature review was carried out to gather theoretical data of different researches in the area of our interest and to have insight of philosophical perspective, for better understanding in the study area.

To gather literature of our interest, we used different resources and databases available for free on internet and provided by *Blekinge Institute of Technology* for research and study purpose of its student. Mostly used resources are as follow:

- IEEE Explorer
- ACM Digital Library
- Engineering Village
- Google search engine,
- e-brary

We gathered information particularly focusing on QA activities and related activities in agile projects. Several research papers, journals and proceeding were gathered. To find relevant literature of our study area, we used different keywords in above mentioned search engines and research databases. Abstract, introduction and conclusion of articles were read for assessment of relevance. From selected literature required information was extracted and presented and presented during writing this thesis.

### 3.2 Qualitative Approach

Our research topic and problem area is dealing with theories rather than numeric and statistical data. We needed to explore the concept and role of QA in agile development in order to identify the gaps in our area of interest. Due to the nature of our study, Qualitative approach was adopted, as according to [20] in qualitative research “author will describe a research problem that can best be understood by exploring a concept or phenomenon”.

Research inquiry plays an important role in a research. It leads and helps the researchers to construct the results or theories. One of the main purposes of our research was to propose the solution for



identified gaps. To construct our solution, practitioners' opinion, who were working in related field, was required instead of mathematical results. Thus, our study can be categorized as Qualitative research because using this research approach "the researcher collects open-ended, emerging data with the primary intent of developing themes from the data [20]".

### **3.2.1 Informal Interviews**

Before designing our questionnaire, we conducted informal interviews with several SQA professionals and developers of different organizations, via telephone. Purpose of these informal interviews was to remain specific and comprehensive while designing questionnaire. These interviews helped us to understand the way we should adopt to extract concerning data through questionnaire.

### **3.2.2 Industrial Survey**

When gaps and critical issues were observed through study of literature, we worked on proposing the solution for improvements. But to propose solution we needed practitioners' opinion about SAQ activities. To get practitioners' opinion, we conducted an industrial survey in form of questionnaire.

We distributed an open-ended questionnaire. Questionnaire structured as open-ended allows retrieving respondents' opinion on behalf of their experience [23]. Although our study is focused on agile SQA activities but to make our observation and proposed solution strong and effective, we also sent our questionnaire to the practitioners working under conventional software development environment.

For distribution of our questionnaire we selected 5 different organizations working on software development, of different kinds. Moreover we also forwarded our survey questionnaire to a researcher who was researching on agile development. For convenience organizations were selected, where the people we, personally, knew were working.

Three out of 5 organizations and the research responded to our questionnaire. We denoted organizations as A, B, C and researcher as D. A is Multinational Company office located in Islamabad, Pakistan. This company is working in technology since 1987. They deliver technology based strategic solution and products around the world. Their quality oriented solutions, products and services are listed below.

#### **Solutions**

- Data Warehousing and Business Intelligence
- Custom Software Development
- Package Implementation and Modification
- Telco BI
- Rental BI

#### **Products**

- Human Resource Management System
- Document Management System
- Financial Package
- E-Governance Enabler
- Internet Banking Application

## **Health Care Services**

- Electronic Medical Record
- Medical Billing
- Medical Transcription

B is a leading multinational company branches in 7 countries. Headquarter of company is located in Stockholm, Sweden. This company is providing IT industry with Storage Management Solution since 1995. We happened to receive their feedback from headquarter, responded by their developer.

C is one of the leading IT companies of Northern Europe established in 1968. The company's focused market is Northern Europe, Germany and Russia. Company has a long list of IT expertise; some of them are in the following areas:

Forest Supply Chain

Healthcare

Banking and Insurance

Telecom and Media

Processing and Network

Information Systems Manufacturing

D is one of the members of a research group. Research group is working on agile development. Their research in originated in a university in Belgium

All the data gathered from above mentioned respondents was comparatively analyzed on different elements and some of them are:

- Development methodology adopted for software development
- Numbers of QA staff was compared to number of development teams
- Their opinion about working of developer without assistance of QA personnel
- QA Activities Performed and performed by whom?

## **3.3 Data Validity**

The validity of data collected from quantitative or qualitative research is necessary [70]. The work in this thesis is completed and compiled with the help of qualitative research. The validity of qualitative research is seen as strength to suggest whether the findings are accurate from participant's point of view [71]. The criteria to validate interview results suggested by Trochim [72] are used in this report. It consists of four different validity assessments to judge qualitative research approach. These validity assessment approaches are as follows:

- Credibility
- Transferability
- Dependability
- Confirm ability

### **3.3.1 Credibility**

The results and findings of this thesis report should be believable or credible from the perspective of the participants involved in this qualitative research [72]. A multi-phased research including qualitative research approach has been followed to achieve credibility of this thesis report. At first we started from literature review to find out agile quality assurance activities or methods presented in academia, to get information about agile quality assurance process in software organization.

After finding gaps and critical issues were observed through study of literature, we worked on proposing the solution for improvements. But to propose solution we needed practitioners' opinion about SQA activities. To get practitioners' opinion, we conducted an industrial survey in form of questionnaire.

Before designing our questionnaire, we conducted informal interviews with several SQA professionals and developers of different organizations, via telephone. Purpose of these informal interviews was to remain specific and comprehensive while designing questionnaire. These interviews helped us to understand the way we should adopt to extract concerning data through questionnaire.

For distribution of our questionnaire we selected 5 different organizations working on software development, of different kinds. Moreover we also forwarded our survey questionnaire to a researcher who was researching on agile development. For convenience, organizations were selected, where the people, we personally, knew were working.

To avoid the threats to data validity, main theme of study was not told to them, so that it might not influence their opinion. Moreover, questions directly addressed their organizations and project experience, to avoid general information and feedback. Moreover the personal whom we conducted informal interviews and the personal those who reply our questioner are not the same. Authors are confident about the credibility of the study, after pursuing this validation process.

### **3.3.2 Transferability**

Transferability is concerned to make concrete conclusions, and to generalize the results. In this thesis report the respondents' view on our questionnaire, remain very helpful to identify the appropriate results. This thesis report can be useful for software industry to apply and improve their agile development process in more effective approach to achieve higher software quality. The survey results, and the issues identified in this thesis report are concerned with achieving higher software quality by applying agile development methods.

The respondents involved in the survey process can be a threat. All persons are working on different domain, having different working experience, may have an effect on the results. E.g. Company A is Multinational Company, They deliver technology based strategic solution and products around the world. B is providing IT industry with Storage Management Solution. C company's focused market is Northern Europe. So, all the contacted individuals from their respective organization may have their own background, experience and knowledge, which can be different. But, all participants' are supporting on highlighted improvement issues in questionnaire regarding quality of products. Other possible threat can be educational system of participants. It is possible that their educational background is different, which may have an effect on the findings of the study. These practitioners' are involved in the concerning field of study regarding software quality improvement process by adopting different measures so natural settings of environment help us to generalize the finding of the study by conducting an industrial survey.

### **3.3.3 Dependability**

Dependability is: "the ever-changing context within which research occurs" [73]. Before forwarding questionnaire, we sent a general overview of the topic for which survey was going to be held. As we mentioned before that we forward open-ended questionnaire, which included questions that are flexible and easy to understand. The participants have no restrictions to answer by ticking the multiple statements. Respondents have choice that they can elaborate the answers by their own view according to their experience and observation in the concerning field of study.

We tried to select high profile and high CMM (Capability Maturity Model) level companies to generalize the results. One threat is that all companies are not of same level and high profile. Some of

them are market project driven organizations and others are providing solutions to Banking sectors. The other threat is that different companies are using different development methodologies one company is depending on the nature of project. One company is working with plan driven development, other two are following agile development process. The finding of the study may be affected by difference in number of companies. There is no validity threat regarding companies, because all companies are concerned and motivated to achieve higher quality in product development.

### 3.3.4 Confirmability

To enhance the confirmability of this thesis report and to stay focus on the topic about quality assurance activities in agile software development. We conducted semi-structured informal interviews in which we formulate and refined our open-ended questions for industrial survey. We also got interviewees' views about questions that are asked for getting required information. We also have discussed issues while analysing the results with the respondents of our open-ended questionnaire to confirm their point of view, and stay focus on our findings to purpose a better solution in our selected area of research.

Our research work from research methodological perspective can be visualized by given figure 3.

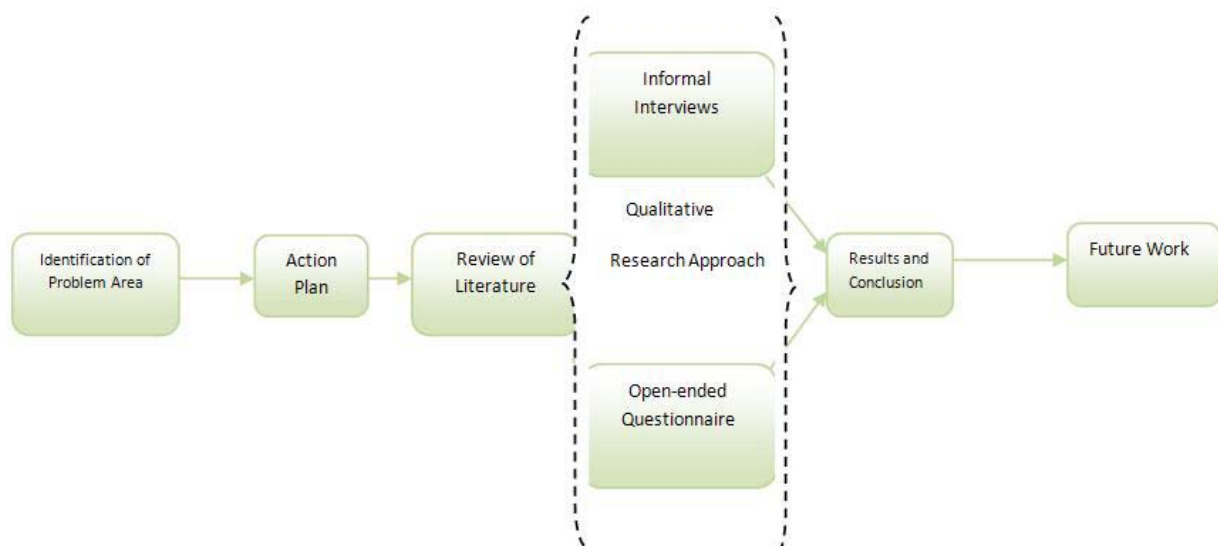


Figure 3: Methodological Hierarchy

## CHAPTER 4: AGILE DEVELOPMENT AND QUALITY

### 4.1 Agility

Agile – denotes “the quality of being agile; readiness for motion; nimbleness, activity, dexterity in motion” according to the Oxford Dictionary [24]. Agility, with regard to software development, can be expressed as the flexible, ready to change and quick-responsive nature of software development process.

Ericksson et al. [25] defined agility as “agility means to strip away as much of the heaviness, commonly associated with the traditional software-development methodologies, as possible to promote quick response to changing environments, changes in user requirements, accelerated project deadlines and the like”. Software development processes injected with agility believe in short releases and discourage heavy documentation that shortens the time wastage and clear the vision of product, to be developed. Close collaboration of customer and all people involved in project remains helpful to provide quality product. Williams and Cockburn [26] stated agile methodologies are developed to “embrace, rather than reject, higher rates of change”. In agile development, processes are divided into short workable iterations. When new iteration takes place changed requirements are merged into system as per requirement. Therefore this approach, to embrace change, also makes software development process agile.

Representation of short comparison between conventional and agile development approaches might lead us to better understating of agile philosophies.

Project Environment		Project Characteristic	
Category	Variable	Agile	Non-Agile
The Development Team	Communication Style	Regular Collaboration	Only When Necessary
	Location	Collocated	Distributed
	Size	Up to 50 People	More than 50 People
	Continuous Learning	Embraced	Discouraged
Project Management	Management Culture	Responsive	Command and Control
	Team Participation	Mandatory	Unwelcome
	Planning	Continuous	Up Front
	Feedback Mechanisms	Several	Not Available
The Customer	Involvement	Throughout the Project	During Analysis Phase
	Availability	Easily Accessible	Hard to Reach
Processes and Tools	Team Input	Team Has the Last Word	Team is Told What to Use
	Amount	Just Enough	More Than Enough
	Adaptability	May be Changed	May not be Changed
The Contract	Requirements and Dates	Flexible	Fixed
	Cost	Time and Materials	Fixed

**Table 1: Comparison between Agility and Non-Agility [13]**

In table 1, agile and conventional approaches are differentiated, with respect to variables and projects environment. We can observe that agile approach is more people-centric as it appreciates the participation of developer and customer. Moreover it embraces and responds change at each level of project, rapidly.

In the next section we will discuss different agile methodologies to understand how the concept of agility is used for software development; in different ways and approaches.

#### 4.1.1 Extreme Programming (XP)

Extreme programming is one of most popular and commonly used methodology in agile software development. The founder of this methodology is Kent Beck; he introduced and defines a number of principles and practices to maintain the productivity of development team and to raise the accuracy and quality of produced system. XP is a lightweight, predictable, efficient and flexible method. It was developed to fulfil the need of small team who are dealing with the imprecise and changing requirements to develop the software in a better way. XP contains a set of discipline and practices for software development process. To apply the XP methodology there are some practices that are needed to be follow in the development process. A summary of XP terms and practices are discussed According to [28, 33].

- **Planning** – programmers estimate required effort for implementation of user stories and the customer decides the timing of releases and scope based on estimates.
- **Metaphor** – system working is defined by a set of metaphors between the customer and the programmers
- **Small/short releases** – application development is done in a series of small, frequently updated versions.
- **Refactoring** – in refactoring system is restructured in a way of removing duplication, improving communication, simplifying and adding flexibility but the functionality of the program should not be change
- **Simple Design** – XP emphasis on designing the simplest possible solution to be implemented and to remove the unnecessary complexity and extra code.
- **40-hour week** – in XP no team member can work in over time there must be 40 hour work in the week. If work exceed then there must be problem in planning
- **Pair programming** – in development phase programmers work in pairs, code, two programmers on one computer.
- **Coding Standards** – there are certain coding rules and standards to be followed in XP that bring consistence and improve communication between the development team.
- **Collective ownership** – No individual person is responsible for code segments, anyone can change any part of the code at any time.
- **Continuous Integration** – code is integrated with the current system when it is ready. This code must pass the entire tests after or before the changes.
- **On-site customer** – Availability of customer with development team is necessary in XP.

In XP every contributor of the project has his integral part in the team. XP teams are formed around a business representative called “the Customer”. With focused on business value, XP teams use simple planning and tracking to decide and predict, what should be done next and when project will be finished. The team produces software in small releases that process the entire test defined by customer. [30]

Test Driven Development, refactoring, system metaphor, and pair programming play the main role in achieving the QA (Quality Assurance). These main practices are harmonize with each other, test driven development verify that written code is bug free. Refactoring always make sure the simplicity of code to avoid the complexity in the developing system. System metaphor provide the basic understanding of system architecture, it reduces the possibility of system failure if development work carried out according to the architecture. Pair programming is most popular practice of XP in which two programmers share their ideas and identify mistakes collectively that helps to develop the system bug free. So we may say these practices play their role to develop the better quality product with minimum risk of errors. [27]

#### 4.1.2 Scrum

Scrum is also a commonly used methodology in agile it was initially developed by Ken Schwaber. The term ‘Scrum’ is derived from a strategy in the game of rugby where it denotes “getting an out-of-play ball back into the game” with teamwork [31]. Scrum provides project management with frame work that includes development tasks like requirement gathering, design and programming are take place. It does not provide any specific method to be applied; it guides the management how their team should function to maintain the flexibility of the system, in applying the environmental changes. In a development process there are many technical and environmental variable exist that constantly change like: requirement, time, resources, and technology. Due to these variables development process becomes unpredictable and complex. There should be a process that can deal with these problems in system. Scrum has some frequent activities that can help the management to achieve better engineering [31].

Scrum contains both managerial and also development processes. Scrum involves Rapid prototyping in this practice team simply take the overview of system requirement form customer. These requirements are not only incomplete but can be changed in the development process. The main practice in scrum is daily 15 minutes meeting to coordinate and integrate the development issues [32]. Some key scrum practices are discussed below [31, 33].

- **Product Backlog** - The team writes all currently identified tasks, in a list called the Backlog, almost all actors can change the backlog, which include customers, marketing and sales and project team. The Scrum master leads the Scrum meetings, identifies the initial backlog to be completed in the sprint.
- **Sprints** –Sprints are 30-days in long. Developers are assigned with number of task to execute a sprint. During a sprint, no changes are allowed from outside the team. In a sprint main working tools of the team are Sprint Planning Meetings, Sprint Backlog and Daily Scrum meetings.
- **Sprint planning meeting** – Early Sprint planning meetings are attended by the customers, users, management, product owner and Scrum Team in which they decide the goals and functionality of the system. After that Scrum Master and the Scrum Team set their focus on product development.
- **Sprint Backlog** – A list of features are assigned to a particular Sprint. When all these features are completed a new iteration of the system is delivered.

- **Daily Scrum** – the scrum team conducts frequent meetings. These daily meetings are more or less 15 minutes long, the main purpose of these meetings is to maintain the progress and discuss the problems of team during development. These group meetings help to raise the moral of the team and improve communication between team members.

Scrum is a management methodology with some important rules and practices. It is not an engineering process that carries defined quality assurance activities. It depends on the management of the organizations to introduce and follow the activities in scrum process to get the better quality of product. Mostly organizations combine the XP and Scrum practices to improve the development process. Scrum focus on communication and feedback which is based on iterative and incremental practices, to manage the development work. In scrum practices daily sprint meetings, continuous integration and acceptance testing are supposed to be followed, in order to achieve quality. [29]

### 4.1.3 Crystal Clear Methodology

Crystal is a family of different methodologies, created by Alistair Cockburn. The name "Crystal" comes from the characterization of projects along two dimensions, size and criticality. Every crystal methodology is marked with the appropriate colour (yellow, orange, red...) that indicates the heaviness of methodology. Choosing the suitable colour of methodology mainly depends upon the Project size and criticality. In larger project we choose darker colour of methodology because it requires more resource and coordination, instead of small one [34, 35].

#### Crystal process

There are mainly three crystal methodologies constructed these are: Crystal Clear, Crystal Orange and Crystal Orange web. All of these methodologies provide tools and standard roles to be implemented in software development process [34].

Crystal Clear is designed for small project with 6 to 8 developers. Team members work in shared office or in one room to maintain the better communication during the project. Crystal orange is designed for the large project with 10 to 40 team members and project lasts for 1 to 2 years [34].

Crystal always uses the incremental development cycle; each increment length is form 1 to 3 week long. There are many features and values that are common in every methodology of crystal. Its main emphasis is on communication and cooperation with people. It does not limit any tool or practice, it also allows adopting XP and scrumming practices to maintain the productivity of the system [35].

#### Crystal Practices

The key persons who are required in crystal include: senior designer, designer programmer and user. There are also certain practices and standards that are needed to be followed or applied in development process and are discussed below described in [34]:

- Software will be delivered incrementally, in a period of 2 to3 months.
- Progress will be tracked by mile stone based on project deliveries or major decisions, instead of written documents.
- Crystal required the direct user involvement in the development process.
- There should be some amount of automated regression testing of application functionality.
- There should be two user viewings on per release.



- There should be workshops for methodology tuning in the middle or beginning of every increment

Crystal practices are mandatory to follow but can be replaced with any other practice because Crystal do not limit to adopt any practice to maintain the development process in the crystal practices automated testing, direct user involvement and side by side programming work are mainly focused by development team to achieve and maintain the quality of product [34].

## 4.2 New Development Approach in organizations

It is very common nature of adaptability, when we find something better than existing work and routine we are attracted to adopt it or replace with our existing practices. In general there are a lot of routines, procedures and processes are being followed to carry the different sort of work [36]. In achieving every task and in creating or developing every product, main factor which is being focused in whole activity is accuracy and quality to achieve these two main features we have to apply and follow the specific procedures and routines, in which some procedure and routines are standard and some are flexible. As we know, nothing is perfect in this world, but gets matured time by time so, every routine and procedure needs to be matured, for that adoption of new change is necessary. Here we will discuss the software development in agile which is new process area in this field and have been replacing the traditional development process because of its flexible methods and welcome every change in the process which is more efficient to achieve the quality and accuracy [37].

According [13], quality is fitness for use, which means the following two things: “(1) quality consists of those product features that meet the needs of the customers and thereby provide product satisfaction. (2) Quality consists of freedom from deficiencies”. Agile process have the ability of maintain and achieve quality of product due to its continuous focus on customer demands and process improvement in respect to active the defined functionality developing system. It overcomes the routine of traditional software development process in which customer collaboration is minimum, and involves high risk of customer dissatisfaction. Agile team members conduct regular meetings to discuss the customer feedback and plan for the next iteration process. Continuous feedback from customer and team members close collaboration during development process play a vital role to achieve the better quality of product. Agile set a new trend in software development process, and attract the quality conscious people to adopt it for the achievement of their quality goals. [38]

## 4.3 Spreading Trend

Whenever a process is changed or replaced, it directly depends upon the behaviour and response of that process. Software development is a complex task; to reduce the complexity of development a number of processes are being followed. But in last few years it has been observed that agile software development increasingly followed and implemented in the organizations. Agile attracts the software development organizations due to its quick response and iterative nature of work. Agile came with many way out for reduce the complexity of development process [39].

A survey conducted by Shine Technologies an Australian information technology (IT) consulting firm, to get the information about early adoption of agile in 2003[41]. They got 131 responses, the survey results indicate that XP was most popular practice in organization (80% or above). This survey indicates that by applying agile software methods they got the quality of application, ease of business and reduction of project cost. In over all response of survey respondents believe that the emphasis on people over processes is the positive feature of agile development. But agile methodologies signify the lack of structure planning and documentation are draw backs of agile. But organizations are intended to continue the use of agile process or have planning to adopt it in near future [40, 41].

An online survey was conducted by Digital Focus by an IT consulting firm in 2005, in that they got response of 137 individuals in 128 organizations from 17 different countries of the world. In total, 90% respondents have the basic knowledge of agile practices. The basic motive of organization for adopting agile development was requirement handling and to increase the speed of software development process to produce better quality products. Practitioners indicate that the ability to adopt change is a key value of agile but organizational knowledge and skills are required for agile implementation. [40, 42]

Practitioners' response and research work done on agile development is giving the clear evidence, that agile methodologies are influencing the organization to overcome the development process complexities. Organizations believed and experienced that after applying agile characteristics they develop better quality products and attain higher level of error absence. To maintain the software quality and accuracy is a main issue in organizations development goals [40]. Although implementation of agile process also carries challenges and drawbacks, but main stress on people/customer interaction and minimization of documentation process, brings the better results and higher level of customer satisfaction. This signifies that agile is helping the organizations to achieve their ultimate goal of software quality and accuracy [43].

#### 4.3.1 The Bright Side of Agile Development

Survey studies highlighted the benefits and drawbacks of agile, process proponents of agile say that agile process got considerable influence from software industry to be implemented in their working environment. A number of beneficial factors are involved in agile development that is influencing the organizations to develop software with better quality. Here we like to highlight some of these beneficial factors which are also experienced by agile practitioners. [40, 44]

- **Flexibility**-The ability to become flexible and delivery of quality software that fulfilling the requirements of customer is recognized as key benefit of agile development
- **Increased productivity**- customer's frequent feedback with changing requirements allows developer team to create better quality product with minimum risk of error.
- **Early detection of feasibility and error**- agile process execute the design, analysis and implementation in repeated iterations that give clear visibility of project progress. On behalf of progress stage customer can take decision to continue or cancel, if they find that it is not going as expected to save the extra investment.
- **High software quality**- short iterations, frequent feedback and test driven development help to improve the overall quality of software.
- **Project control**- main focus of agile is on people over process and less stress on documentation gives the opportunity to developers improve process activities like: short iteration, knowledge sharing, continuous integration and feedback with full project control.
- **Knowledge transfer**- agile team members share their knowledge and observations in regular meetings in which they mainly discuss on developing project issues. This activity increases the communication between team and they remain updated with current scenarios.

### 4.3.2 Challenges Adopting Agile Development

Although agile contains key benefits to produce better results but to adopt a new process, is also a challenging task, to accept that change organizations have to rearrange their prior setup, practitioners and opponents of agile also indicate challenges in agile adoption. Here we would like to highlight some common problems and challenges, which are becoming hurdle in agile implementation, described in [44, 45, 46].

**Serial thinking-** it is an attributed fact that from last 40 years, software development methodologies have been using the serial approach of development. Practitioners of these approaches are supposed to work serial set of task like: requirements gathering, system design and after that they start coding. These people need appropriate training, enough time and continuous monitoring to learn agile practices. Organizations who are thinking to adopt agile methodologies have to be sure first, that serial mind-set will not harm the introduction of new process.

**Individual resistance-** there are many software professionals who are not interested to learn upcoming methodologies. These people are divided in two groups, first group perceive agile as nothing, second group is anti-agile, they intentionally spread incorrect information about agile to prevent the new methodologies introduction in their organization. These group of people also required detail education about upcoming methodologies.

**Fear of change-** lack of confidence in software professionals about new methods also becomes a hurdle in the way of adopting change. Fear of change is usually associated with sense of loss. Most people have fear in their mind that they will not be able to new methods and skills to fulfil the requirements of agile use.

**Large scale organizations-** organizations working on large scale are facing problem in implementing agile process because it requires availability of onsite costumer. It may because different communities and some organization cannot afford individual translator every time when they have to communicate the costumer. They still prefer to gather the requirements at once.

**Heavy documentation mind-set-** agile focuses on code development, instead of creating UML design. But some software professionals still believe the concept that effectiveness of the software relay on the comprehensive documentation and requirements design.

**Organizational doubts-** people think that it is better to adopt suitable options and methods form new introduce methodologies. Many large organizations consider the fact that combination of agile and traditional practices is best suited for them; because adoption of overall process will require more education, skill improvement and high risk of stability.

### 4.4 Philosophical Angle of Quality in Agile Development

It would not be wrong to say that agile development approach is not as mature as conventional development approach. There has been much criticism on agile development approach from practitioners' and researchers' side. But, apart from criticism, agile approach is adopted by many organizations for software development. Being practiced in software industry is the evidence that this approach is productive and beneficial in several ways. Each of software development approaches must assure the quality of product and process itself. In next sections let us discuss about the philosophy of quality with regard to product and process, in agile approach.

#### 4.4.1 Quality of Product

*“If something is within specification<sup>2</sup>, it is of good quality” [47]*

According to above mentioned quotation, if a product/service meets the higher level of users' customers' requirements, it conceives quality. In software engineering requirements are not a static entity. They keep on changing even during the time of product development. Conventional development approaches have been criticised much that they do not respond to change more effectively. Moreover, several studies have put forward a point of being misinterpreted requirements. So, it is hard to impart quality in product/service without understanding the requirements. Lack of communication between customer and technical people related to software development may lead to misinterpretation of requirements [48][50].

In agile approach developer and customer collaboration (as on-site customer in XP) tried to fill the gap of communication. Developer keeps interacting with customer and develops the system as per the vision of customer. In such a way the organization gets closer to functionalize the requirements of customer which leads to the better quality of software product. Sixth principle of Agile Manifesto encourages even face-to-face collaboration.

Another advantage of involvement of customer is to overcome the problems of change in requirement. As is agile projects, at the end of iteration, a functioning piece of working software is released for feedback. When next iteration takes place changed requirement are merged in previous release; so, it does not affect the project in the means of development flow. Prototypes are developed in initial stages of project that help to avoid the conflicts of how a system would look or work. Unlike conventional development approach, in agile projects product is evolved rather than developed, and evolution mostly leads to enhance the quality of product/service.

#### 4.4.2 Quality of Product and Testing

Quality of the software product may also be seen in terms of flawlessness and reduced or zero number of errors and bugs. To make a software product free from errors and assess its efficiency, testing is conducted. Where software development is moving towards agile software development, testing approaches also gets changed in projects. In any software development quality assurance (QA) activities are directly related and responsible for quality of software development processes as well as of software product. Agile software development redefines activities and practices of QA.

In agile software development testing and development of software runs parallel in order to achieve higher level of quality. Testing can be valued as the backbone of QA activities and a vital step to attain quality in software product. In agile development process an error or bug is fixed as soon as it is known and no matter who fixes it [48]. Adopting this manner, bugs and errors are detected and fixed in early stages. Fixing bugs and errors sooner, implies to save time, cost, resources and maximization to quality.

##### 4.4.2.1 Quality of Process

As organization targets the quality of software product, she also tends to reduce cost, time and resources. In large and complex software projects, point of attention is time and effectiveness, so that quality may not be compromised. We can say that if a development process ensures the maximum utilization out of minimum resources, depicts quality. In this section we will discuss how agile development approach benefits an organization in the means of development process.

---

<sup>2</sup> A specification is an explicit set of requirements to be satisfied by a material, product, or service. [ASTM Definition]

Fowler and Highsmith mention:

“In an agile project, it’s particularly important to use simple approaches, because they are easier to change. It’s easier to add something to a process that’s too simple than it is to take something away from a process that’s too complicated” [47]

Agile development process are claimed to be more flexible toward change. So by nature agile approach responds to change efficiently and extra time and effort is not needed. And being iterative the resources of the organization are not locked in linear processes. Several studies have found and many practitioners have claimed that errors and bugs found after the release cost more time and resources to organization [49]. Agile practices like Pair Programming and Test Driven Development targets to detect and fix flaws of software before final release and ultimately this development nature benefits an organization in terms of time and resources.

Agile methods are not process-centric they are claimed to be people-centric. Agilists encourage and trust in expertise of people involved in project. In Agile Manifesto principles it is stated to provide people with such environment that persuades them to work freely rather than to be confined within processes. If, during the project, collaboration is enhanced and involved people are trusted to get the job done, may lead to significant increase in quality and amount of work. [47]

Although agile methods are fundamentally adaptive to change, incremental and self-organized, but a lot of work is being carried out in the field of Process Improvement, regarding agile development. Several books are written to adopt CMMI in agile projects. More over Six Sigma and ISO standards have also recognized the importance of agile development in projects and encouraging this methodology towards maturity.

## CHAPTER 5: AGILE QUALITY ASSURANCE, APPLIED

Agile development methodologies (also known as light weight methodologies) are being, increasingly, used and accepted for software development projects. These methodologies are adaptive to change, flexible and the features like refactoring attracted a large amount of practitioners to adopt agile development methodologies. Some practitioners are in favour of agile development some argue against it, where some of the practitioners recommend adopting mixed agile and plan-driven practices [50]. In this section we will discuss agile QA within development and management, more specifically. This section is more influenced by practitioners' perspective regarding agile development. Moreover, in this chapter we will discuss our proposed techniques in detail mapping it with feedbacks of our industrial survey.

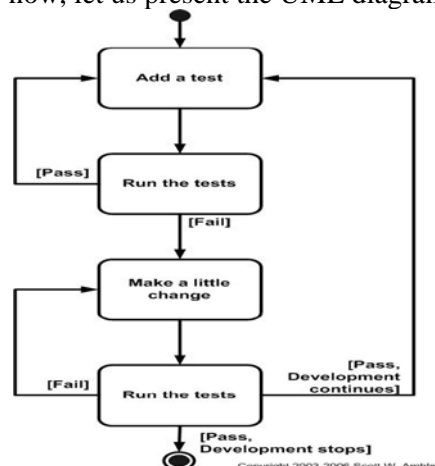
### 5.1 Agile Testing Approach

Pair programming and Test-Driven Development (TDD) are some of the key practices of agile development to achieve quality in software products. Using these approaches, agile development makes testing an integral part of project. But agile placed a dramatic change in testing practices by shifting the testing responsibility from QA tester to developer. In agile, developers are supposed to write tests and test their code or each other's while doing Pair Programming. Customer is supposed to be highly involved throughout the project in agile development. Acceptance testing is the responsibility of customer who is participating in project. Agilists claim that customer involvement in project and in testing leads to develop the software for higher conformance to requirements.

**Test-Driven Development (TDD)** is getting popular day by day, within agile community. TDD focuses on writing tests before coding and frequently integrating the new code. In TDD new code denotes to either a piece of code or a piece of code that already exists and integrated after few changes; in order to emerge changed requirements (according to customer's feedback) [51]. TDD is influenced by Test-First Development (TFD) approach where developers are to write all test cases and tests before starting actual programming [67]. *"It's one way to think through your design before your write your functional code. Another view is that TDD is a programming technique"*, Scott W. Ambler says and expresses TDD as:

**TDD = Refactoring + TFD**

To visualize the concept of TFD now, let us present the UML diagram of TFD by Scott W. Ambler.



**Figure 4: Test-First Development**

We can say that TDD is not only a testing approach but also an approach to design the software gradually and incrementally. As we can see in the above presented diagram that adopting TFD, each piece of code is supposed to pass all test cases before it is written.

TDD has changes the role of Quality Assurance in software development as testing activities are conducted by developer [51]. Agile practitioners claim that this continuous testing results in software product with high quality. [68] Defines agile quality assurance as “the development of software that can respond to change, as the customer requires it to change. This implies that the frequent delivery of tested, working, and customer-approved software at the end of each iteration is an important aspect of agile quality assurance.” In practice, QA in agile development evolves around customer’s feedback and developer who is tester at the same time. Besides achieving higher quality through continuous testing, agile opened new chapters of criticism elaborated by practitioners and researchers. But before discussing the problems and critics that agile QA is facing in industry lets peep through few more QA activities of agile development.

## 5.2 Pair Programming

According to [53] Pair Programming (PP) means, “*All production code is written by two people at one screen/keyboard/mouse.*” PP is one of the core activities of XP. Purpose of adopting PP is to monitor and learn from each other continuously, while writing the code. Monitoring of software development and process is the responsibility of QA staff but PP imposes this responsibility on developers by sticking them to work together.

Claims have been observed in industry that PP approach is adopted to achieve high quality as [54] states “Pair programming can improve design quality and reduce defects”. But the point of debate is that how to trust developers over professional QA testers for productivity of real software product, apart from philosophy? There is no doubt that agile development emphasizes on selecting good and experienced personnel; but if quality is targeted through the best personnel then where lays the productivity of a development methodology?

Participants admit that a particular level of experience that personnel possess is required to make agile projects successful. “There was some consensus that 25%-33% of the project personnel must be competent and experienced.” Here, competence and experience is meant by experience of building similar systems, prior knowledge of technology domain and having sufficient interpersonal and communication skills. The point of consideration is that for successful project, experienced people that have developed same systems are more important than the people who have worked under agile environment. [55]

## 5.3 Refactoring

Refactoring is an integral practice of agile methodologies. One of the common definitions of Refactoring the code is presented by [56] as:

*“Refactoring is a disciplined technique for restructuring an existing body of code, altering its internal structure without changing its external behaviour. Its heart is a series of small behaviour preserving transformations. Each transformation (called a 'refactoring') does little, but a sequence of transformations can produce a significant restructuring. Since each refactoring is small, it's less likely to go wrong. The system is also kept fully working after each small refactoring, reducing the chances that a system can get seriously broken during the restructuring.”*

According to [53] design of the system is evolved through transformations and it is achieved by refactoring in XP. The purpose of refactoring is to make code more understandable, reduction in complexity of code and make the flow of code simpler. Studies and practices have shown that refactoring of the code has a positive impact of software quality as it reduces the possibility of error. It also taken as approach of reviewing the code because when developer goes through the code for refactoring it also helps him to remove bugs and errors besides restructuring the code.

A study was presented in 2007 that observed that evolution of software development using XP, over more than two years. In this study comparative results are presented with respect to XP (agile

development) and plan-driven development. This study comes up with the result that using refactoring (in XP) there is noticeable absence of complexity in code while compared with plan-driven development. In plan-driven development code can be restructured when all code is written and tested. But in adopting refactoring approach code is restructured and integrated continuously that keeps it less complex and helps to identify errors in early stages of development. [57]

Refactoring has a positive and important effect on QA. It helps to reduce the errors that ultimately saves time. In agile integration and refactoring of code is done continuously thus the issues of compatibility can be overcome efficiently on early stages of software development. Practitioners also present an argument that refactoring the code ultimately reduces the need of extensive documentation. But there has been no serious debate regarding this issue.

## **5.4 SPI and Agile Methodology**

Software Process Improvement (SPI) provides an organization with a consistent method to get the tasks done, and SPI influences an organization administratively not technically [58]. In today's competent age, companies believe that by improving their software development process they may develop quality software products.

Over the era, different commercial standards and models are introduced in order to improve software development process, commonly used standards and models are ISO standards [59] and Capability Maturity Model Integration (CMMI) [60]. These commercial models and standards might differ in nature but their ultimate goal is to improve software development process, continuously. For getting certified for a particular standard or model, up to a level, an organization is supposed to full fill the standards within their process. To achieve this, detailed documentation of product development throughout the process, standardized procedures are supposed to be followed and monitored, categorically and frequent audits and evaluation procedures are conducted. Chasing these standards and models SQA plays the most active role. QA personnel are responsible to implement, monitor and guide selected model or standards for SPI on an organization level as well as on project level.

Due to the lack of time and scope of our study, we may not be able to discuss all of the standards and models for SPI. In this section we keep CMMI under our discussion as it is widely used SPI model. Out there in industry, some believe CMMI is a set of standards but, besides its name, CMMI experts claim it as a model for SPI. There is no doubt that if some organization is getting continuous improvement in her development method it will affect her products positively. But all these philosophies and practiced are welcomed in plan-driven software development. Agile methodologies are claimed not compatible for these standards and models, and the debate regarding SPI and agile continues.

### **5.4.1 Problem Affiliations**

Unlike conventional development methodologies agile development integrates QA practices in development activities, rather than practicing them independently and separately. Sometimes due to managerial or/and organizational issues, customer or project requires some standards to be followed [61]. And for standard conformance generation of documentation is essential where as agile principle is to develop working software providing only comprehensive documentation [61].

Last year Software Engineering Institute of Carnegie Mellon University published a technical report [62] that addresses the problems that make CMMI and agile methodologies incompatible. According to this report one of the important reasons which makes CMMI and agile methodologies incompatible is that the organizations adopted CMMI, were large scale with high level of management, organizational hierarchy and governance; where the organizations which adopted agile methodologies were focusing on small or single team projects. [62]



Another case study [63] was carried out in 2005 that focused on using CMMI and agile methodologies for organization maturity. This case study places a claim “*Currently, the existing agile project learning techniques seem to lack means to perceive the organizational SPI aspect. For example, they do not address the important aspects of systematically defining, validating, packaging and storing the SPI results of agile projects.*”[63]. It can be observed that all the aspects, this claim addressed, are SQA responsibilities. Any kind of improvements within the organization, project or development are to achieve higher level of software products. If agile methodologies are practiced to produce quality products without CMMI, then there is a vast space for improvement.

From 1998 to 2002 a study was conducted at Ericsson in Gothenburg. This study was carried out in order to bridge the gaps in SPI and agile development. Researchers involved in this study stated that industry in need of formulating and enhancing Agile Software Process Improvement practices. To achieve their goal they observed 18 different projects injected by two different SPI tactics. They developed two SPI tactics for SPI and called them *Supertank Tactic* and *Motorboat Tactic*. The baseline of supertank tactic was CMMI mindset and the baseline of motorboat tactic was agile mindset. After concluding their study they realized that motorboat tactic for SPI came up with better results and positive impact. But in their study researchers also admitted that detailed material guidelines were available to practice supertank tactic because it was developed on CMMI mindset. Despite from better results, researchers claim that agile development focuses on software development on project level while CMMI mindset is to focus on software development through development process on organizational level. [64]

In plan-driven development SPI encompasses all the activities related to QA and software development in a well defined, organized and standardized manner. Although people involved in project are participants of SPI but QA personnel are responsible of SPI activities. SQA personnel conduct surveys, evaluation, meetings and internal audits. They are considered as experts within an organization as they educate and implement SPI in it. Besides, merging QA activities in software development, agile methodologies cut short the organizational role of QA. Developers may be aware of testing and designing but they might be less aware of SPI on organizational level. Literature shows that there is need to redefine the role of SQA in agile development projects in order to increase organization knowledge and maturity for maximum output.

## 5.5 Proposed Solution

We gathered available literature and research data by conducting literature review, in our field of interest. In previous sections we have provided philosophical and practitioners’ perspective of SQA in agile development. We observed that literature, in our field of study, has following flavours of influence on our study:

- Claims of higher quality through agile development (*guidelines, survey results, favours and critics*)
- Integrated QA activities in agile methodologies (*guidelines, favours and critics*)
- Claims of lacking in defined standards in agile development (*critics, surveys*)
- Research done to enhance throughput over agile development projects (*research results and suggestions*)

After this literature review, we think that role of QA practices in agile projects needs to be redefined. Although, our fundamental observation depends on the literature but we also conducted empirical study. By sending questionnaire (it will be discussed in next chapter with detail) to three different companies and one researcher, we targeted to get individuals’ perspective about SQA and agile.

“Integrating quality assurance into projects to enhance their agility requires making changes”. [65] Followed statement was given by a famous agilest in 2003 on the early stages of agile development methodologies. According to this statement QA must be integrated rather than shifted, here shifting

implies as most of the QA activities in agile projects are performed by the developer or QA personnel are supposed to perform as developer. There is no doubt that agile methodologies came up with higher quality due their incremental and test-driven nature. Production of higher quality, and absence of systematic, organized and well defined procedures and standards, show that there is much space of improvement in agile SQA activities. Baseline of our suggested solution is “to redefine SQA activities rather than shifting”.

### **5.5.1 Extra Layer of QA**

Quality assurance activities in agile projects evolve around testing and feedbacks. Literature available on agile quality assurance remains focused on testing and its different approaches. Agile methodologies are incremental, adaptive to change and self-organized. But in any situation, leadership is required because self-organization cannot lead future when requirements and system get complex. As these methodologies are called people centric, so focusing on this point, agile methodologies seem to use expertise of people not for the purpose they are trained for.

Different research works in software industry and practitioners claim that agile methodologies are less supportive to develop complex software systems. Subjected methodologies require best people to compose a team and doing so experts and design analyst are, in some situations, play the role of a developer and on the other hand developer is also supposed to conduct QA activities. In plan-driven software development, QA activities have been monitoring and separate entities. Agile integrates QA activities in software development by requiring some QA personnel to play the role of developer in projects.

In convention approach SQA activities target not only quality of product but also the quality of process over organization level. SPI is the vital part to achieve in conventional practice of SQA. Industry is not accepting to avoid the importance and benefits of SPI, therefore sufficient research work has been carried out in order to get CMMI and agile, compatible for each other.

We think that agile development approach needs to redefine their SQA role and activities in development projects. By enhancing the role of SQA may result in making agile methodologies effective over organization level. We are likely to propose the approach of an extra layer of QA in agile development projects.

In our proposed solution, applying extra layer of QA means to place QA experts with development team. Adopting this approach, organization might be able to maintain quality standards. All typical QA activities like testing, requirement gathering and evaluation must be the responsibilities of QA expert rather than developer. To maintain the agility communication among development teams and within team members must be encouraged. Application of extra layer of QA in organization, provided with feedbacks, will be discussed in next chapter to make detailed understanding.

## **5.6 The Need of Extra Layer of QA**

Several research works have been carried out in order to find the way to get benefit by merging conventional and agile QA. We think that our proposed approach is a small contribution in this concern. Being virtually separated and extra layer of QA may perform conventional QA activities without disturbing agility of project.

Focus of QA is to monitor the process and product development to ensure the quality of the product, delivered. Throughout the development it is the responsibility of QA to provide the organization with process assurance and product assurance. In process assurance QA conducts meetings and audits to assure that standards, procedures and plans are being followed. In product assurance QA activities are to assure the quality of requirements, design and code. To follow a defect prevention technique and conduct testing of developed software is typically a QA activity and responsibility.

By imparting testing culture in agile projects may increase the value to product quality but process quality cannot be compromised to make good business. According to [26] agile practitioners are less attracted towards CMMI certification or other commercial standards of product and process quality. Through literature, we observed that agile focuses more on testing that may cause to neglect other QA activities. If the responsibility of requirement analysis and testing is shifted on developer, it may overload the work of developer causing lack of quality in process and product. There is no question on the role of QA professionals in software development projects.

We observe that QA professionals' role, in agile development projects, needs to be redefined in order to get higher quality rather than putting their responsibilities on developers. Through this thesis work we are likely to propose an Extra Layer of QA in agile projects. Our proposed technique is to allocate QA professionals in development team as member. And all QA activities within the team must be monitored by that QA professional and mutual collaboration of those QA professionals must be encouraged and maintained to confirm agility of project.

In agile methodologies either developer works as tester or vice versa, this makes a dramatic shift in responsibilities and may be taken as a challenge of expertise. By adopting proposed approach an organization will be able to get quality of work as all resources will be working in their expertise. According to [65] developer must not test his own written code, but in agile development most of the testing is done by developer that must be kept on minimum level in order to enhance quality.

We think that agile methodologies are lacking standards to access their capabilities and maturity unlike conventional development. This may be because all experts are involved in development and do not consider to monitor and measure the project on organizational level. By adopting proposed approach organization may develop and adopt standards and may handle the quality of complex projects efficiently.

Instead of educating the customer it is better to utilize expertise of those people who have abilities to interpret customer need to be implemented technically. But continuous interaction with customer is also a positive value to product quality and agility. In our approach only difference is rather than developer; QA personnel must interact with customer on regular basis to ensure the quality of product.

Our proposed approach is based on gaps identified in literature review. To support our proposed approach we conducted an industry survey that will be discussed in next chapter with details. We received the feedback of our survey from 4 different resources, three of them were companies related to software development and one of them was a researcher working on agile development in a university. All of the feedbacks favour the fact that in industry importance of documentation cannot be neglected even adopting agile development. During literature review we find that many practitioners claim that maintenance of detailed documentation is required. Documentation is not only required by customer, but also essential to maintain the quality of product in future. Moreover, feedback led us to believe that developer cannot follow all of the quality concerns and procedures without the assistance of QA personnel. We believe that if knowledge and experience of conventional QA activities is imparted in agile project through extra layer of QA, it may lead us to maximization of product quality and maturity in agile methodologies.

## CHAPTER 6: SURVEY AND RESULTS

We conducted survey in industry by sending questionnaire to different organizations which are working in software development. Feedback obtained in result of distributed questionnaire helped us to get to our proposed solution in effective way.

### 6.1 Questionnaire Structure

Most of our thesis work is influenced by qualitative research. Our research questions were formed to identify the gaps in Software Quality Assurance using agile development in software development projects. After identifying the gaps we structured our questionnaire to support our proposed solution with empirical data from software industry.

For in-depth study, before designing our questionnaire, we conducted informal interviews with several SQA professionals and developers, of different organizations, via telephone. Purpose of these informal interviews was to remain specific and comprehensive while designing questionnaire. In these interviews we pose general question and then deeply highlight the practical facts concerned with agile software development and quality assurance. There were no time limits in these interview sessions because we already told them that we want to discuss some general and particular aspects regarding our selected study. These connected persons strongly welcomed us and communicated with us in a very friendly environment. Although some personal were busy in their routine work so they simply excused us that they were not ready for those informal interview.

We have discussed our findings and gaps in agile quality assurance activities according to our literature studies and the interviewee came with his own views according to his experience and observation. We briefly described our findings and gaps in agile quality assurance process and we also discuss our basic idea as a solution to deal with the gaps, upon which they appreciated and supported us that we were conducting research in this area. Moreover our main motive was to formulate a specific and precise open-ended questionnaire to perform an industrial survey. While we are two group partners, both of us take part in discussion while communicating with the SQA personnel form different companies. But in the meanwhile one of us remains busy in document the views and key points in discussion. These informal interviews become very productive for us to refine our survey questions, which are to the point and easy to understand.

Our questionnaire was comprised of 10 questions. Except Question1 and Question2, each question was open-ended; therefore, overall our questionnaire was an Open-Ended Questionnaire. “Open-ended questions allow the respondent to express an opinion without being influenced by the researcher.” [69] Most of our research work was based upon literature, it was important for us to know the opinions of current practitioners. Rather than collecting statistical data, in our study, practitioners’ opinion was required and all of questions were designed keeping this requirement in view. (These selected questions were structured with assistance of literature study, group meeting, also discussion between concerned persons in informal interviews via phone in which we removed and refined those questions that are not related. At the end, as per opinion of and thesis advisor, questionnaire for industrial survey was finalized)

(Questionnaire Presented in Appendix)

## 6.2 Questionnaire Distribution

After finalization of our survey questionnaire, we started to distribute it via e-mail. Due to limitation of time we sent questionnaire to 5 different companies and to one researcher. Frequency of feedback had been satisfactory as 3 of the companies responded within 4 days and one researcher provided us with feedback the very next day, we sent him our questionnaire. We did not receive any feedback from two companies.

Companies selected for questionnaire distribution were convenience samples for us, because we have personal contacts with one employee in each company. The main reason to select those companies was, easy to access and communication with the personnel working in the companies. Otherwise it would be a time consuming and difficult task to contact with different companies and to have feedback from them, as we had to face while conducting informal interviews. Because every company has its own routine and procedures some take interest in these kinds of surveys, some simply reject by saying that they cannot manage the time for this due to their running projects and deadlines. So selecting the companies which are convenience samples, gave us the ease to contact and have feedback with less time and effort. But even then three out of five companies respond us other two companies did not reply us because of their managerial issues.

Moreover, respondents of our questionnaire are working in the concerning field and have key role in the software development team. The three companies are working with different methodologies e.g. company A following mixed methodology, company B working with plan-driven and company C along with one researcher working with agile development. So luckily we got practitioners' opinions who are working with different methodology. Although our research work is targeting agile development methodologies but opinion of plan-driven and mixed methodology practitioners also reserves importance, for our proposed solution. In below given table, further information is provided about the companies and respondents of our questionnaire.

Company	Methodology	Location	Designation of respondent
A	Depends on Project	Islamabad, Pakistan	QA Engineer
B	Plan-Driven	Stockholm, Sweden	Software Developer
C	Agile	Denmark	Software Developer
D	Agile	Belgium	Researcher

**Table 2: Respondents Information**

## 6.3 Data validation

Data validation is very important to remove confusions about answers and to be satisfied with results. After getting the feedback, we contacted with respondents and asked if they were sure about these provided answers. Then, they have mentioned us that while filling out our questionnaire, they also consulted with other team members in the company who had higher experience in the concerning field including team leader or project manager. So after getting agreed and sure about the opinion, they filled in the questionnaire. So indirectly, key personnel who have their vast vision and experience in the concerning field were also involved to answer the survey questions. So respondents were sure and confident about the answers.

Below are the tables of some questions from our industrial survey. Answers presented in tables are the opinions of our respondents. These answers helped us to propose our solution in the light of practitioners' experience about QA practices.

Company	What approach are you using for product development? <ul style="list-style-type: none"> <li>Agile</li> <li>Conventional (Plan Driven)</li> <li>Depends upon the nature of the project</li> </ul>
A	Depends on Project
B	Plan-Driven
C	Agile
D	Agile

**Table 3: Question1**

Company	What are the core quality assurances activities being followed in your organization, with respect to plan driven development?
A	Auditing, Testing and Process Improvement
B	Reviews, Audits, Testing, Verification and Validation
C	—
D	—

**Table 4: Question2**

Company	If you are following agile development, what is the role of quality assurance in your projects?
A	Continuous testing
B	—
C	The role of QA is to prevent defects, not merely find them. In order to achieve it QA should be moved up to the front of Development Cycle. It helps in overcoming many communication errors that result in delays, defects and waste. In addition test should be implemented before code is written, it will help the developers in seeing how their code is behaving and at the same time, it will ensure that high quality code is produced.
D	We do user studies; focus on good quality code, surveys, focus on user interfaces design and best practices

**Table 5: Question3**

Company	Q4: If you are using or have been using a conventional approach for development, do you think quality assurance is complex and consumes more resources during projects when compared to agile development?
A	Yes, to some extent
B	I guess it's not like that. In agile development there is no hard and fast rule to be followed but in plan driven approach, everything is documented and planed and there are less chances of complex development but planed development it always takes more resources.
C	—
D	Yes

**Table 6: Question4**

Company	How many members are there in your QA team, and how many development teams do you have in your organization?
A	3 members of QA staff 5 Development teams
B	4 QA personnel 7 Development teams
C	In our project total 25 members in which 5 of them are QA personnel
D	We are research group

**Table 7: Question5**

Company	Comparatively, agile development focuses more upon product testing, throughout the project. Is it enough to have a better testing approach in order to produce quality products? Give your opinion?
A	Agile testing approach works well with small projects.
B	Experienced testers are vital to acquire quality of product.
C	—
D	Agile development demands the best people for development teams to bridge this gap.

**Table 8: Question6**

Company	From your project experience, do you think that a developer can follow all quality procedures without the assistance of QA personnel?
A	No at all.
B	Not at all, quality assurance is a complete science.
C	No.
D	In large scale project one may need QA personal, but for prototype scale development there is no need for QA personal.

**Table 9: Question7**

Company	If a project demands more customer involvement for product development, who should interact with the customer, and why?
A	Business Analyst so that he can directly communicate with the client on ambiguities, changes and its effects on overall product
B	<b>QA Professional Developer</b>
C	QA Professional
D	QA Professional Team Lead

**Table 10: Question8**

Company	Do you think that extensive documentation is required for projects, either in plan driven or in agile development, or is it possible to limit the documentation in order to save time and resources?
A	It depends upon the nature of project.
B	Documentation is necessary it give guide lines and extra help to developer as well to user
C	By focusing on user stories time and resources can be save
D	It depends

**Table 11: Question9**

Company	Have you ever observed that your organization has had to compromise on process quality and prioritise product quality in agile development or vice versa in plan driven development, and if so, in which way?
A	In implementation phase situations are occur when u need to prioritise the issues of product and process to maintain quality assurance
B	—
C	Quality is a core issue so it depends when face with problem
D	It is always needed to maintain the process quality to maintain the product quality

**Table 12: Question10**



## 6.4 Analysis

We analysed the empirical data on critical issues faced in agile development methodologies. Fundamental motive of our proposed approach is to use the knowledge of plan-driven methodologies under agile projects. Getting to this solution we needed practitioners' point of view. After gathering the data from industrial survey, we have analysed the data from many perceptions for that we highlighted different issues and find their connection with problem domain. These different issues are interrelated each other and help us to get appropriate results. The group meeting was arranged between both thesis partners to discuss important factors from survey results and findings.

One of the critical issues was that agile development methodologies keep on shifting QA responsibilities on developer; moreover their QA practices are more focused on testing. Most practitioners and researcher claimed that developers should not be testing their own code. Here we felt that agility needs to redefine QA activities. When we analysed the feedbacks of our questionnaire, we found that organizations following agile are only conducting testing activities in QA, whereas organizations following plan-driven also focusing on other issue of quality like:

- Stability
- Security
- Reliability

Resources are also the critical issue in any organization, to map our proposed solution our **Question5** was written to estimate their resources. We observed that all companies are consuming almost same ratio of resources. Agile methodologies are shifts QA practices of developer so they are supposed to have less QA personnel but data showed the different story. This point drags us to think that even adopting agile methodologies, yet organization are in need of QA staff; therefore we propose to merge QA staff and development teams but allowing them virtually separated.

SPI is not like a single project, it is a continuous improvement of organization in terms of product quality as well as process quality. SPI also requires expertise and deep knowledge of this field. In plan-driven development QA staff reserves responsibility of SPI. But in agile methodology focus remains the product throughout the project. In **Question7** we questioned if developer may follow all QA procedures without assistance of QA staff, but all feedbacks refused to accept this. Development should not be taken as mere philosophy; it is not possible for all organizations to assemble their development team with the best of people. This emphasizes on the need of planting quality experts in agile to let them to perform their role in order to institutionalize the culture of SPI in organization.

Another critical issue was to interact with customer frequently for the conformance of quality and requirements. Agile methodology plotted really ideal scenario that developers are more technical so they must interact with customer to interpret the requirement technically. This philosophy neglected the fact that developers are technical minded people and customer might not possess technical knowledge, so in this situation when customer is not aware of complexities of software, QA personnel must interact with them because they are the people who know how to interpret customers' requirement to be implemented in code. Only feedback from company B favours that developer must interact with customer, rest of the feedback either state Business People or QA staff.

After reading feedbacks we observed organizations adopting agile methodologies, either keep on getting conventional QA services, partially, for complex system or they invest a lot many resources and time to educate customer and developer.

These were some of critical issues we kept in view while analyzing feedbacks. But due to the limitation of our study and resources, we think that there are, much vast, topics left to be discussed in agile development. We hope that if our proposed approach is adopted after further reforms, it may help

to bridge the gaps of quality assurance in agile projects and increase in quality over organizational level.

## **6.5 Analysis with Research Questions**

The data analysis is described by using industrial survey results and literature findings. The relationship between research questions and our survey questionnaire answers provides guidelines and help to analyse the data from different perceptions of agile quality assurance process. We present complete analysis that satisfy research questions and our findings as well. The followings sections showing data analysis with each research questions. The research questions are presents as RQ1, RQ2 and RQ3.

### **RQ1: What is the role of Software Quality Assurance activities in Agile Development, according to existing literature?**

The solution of this question we have presented in our chapter # 1, where we give brief introduction about agile software development. We have more discussed more about the research question issue in chapter # 4 as well. We also discuss SQA activities in agile development, that how they are being applied with different agile methods e.g. Pair programming, Scrum and Crystal to achieve software quality according to literature agile software development is productive and helpful to achieve customer satisfaction as compare to conventional methods. Agile quality assurance activities are flexible; they give importance to quality of product, instead of following restricted procedures. Later on the result of our survey **Question # 3** in which we asked about the role of QA activities in agile, also shows the flexibility in agile QA activities like focus on continuous testing, test after each design and focus on user stories and interface implementation. But all have one main focus behind that is to achieve higher quality.

### **RQ2: Try to identify any specific process area of agile iterative culture that can be improved in the means of Quality Assurance activities?**

We have presented the solution of this question in chapter # 2 about problem definition where we discuss about the SQA challenges facing by agile development. Letter on we discuss the problem affiliation in chapter #5, in which we describe the proposed solution. We came with the idea to redefine the role and the task of QA professionals in agile development team. Because there are many issues that can be handled with this solution. E.g. if QA professional also collects user stories so he can communicate in better way with customer and can give better suggestions, during requirement gathering, with reference to maintain the quality of product.

Some time due to limitation of time developers are testing their own code so we also propose here that there must be a QA member in the team to handle with these task, so there are many facts regarding our propose solution which we mention in our report. The industrial survey results also support to purpose a better solution like, In **Question7** we questioned if developer may follow all QA procedures without assistance of QA staff, but all feedbacks refused to accept this. Development should not be taken as mere philosophy; it is not possible for all organizations to assemble their development team with the best of people. This emphasizes on the need of planting quality experts in agile to let them to perform their role in order to institutionalize the culture of SPI in organization.

### **RQ3: How organizations are practicing SQA activities, while adopting, agile development projects?**

We have also discussed some facts and details about this question in our chapter 4 but our findings from industrial survey also remain helpful for us to present this research question. Selected organizations are working with different methodologies like, plan-driven and mixed approach. But they are also influenced and attracted towards agile methodologies because of flexibility in agile methods. To locate resources is a main issue in organizations, as we asked in our **Question 4** about resources consumption in agile and other development methods. While performing QA activities, they agreed the fact that resources consumption are less in agile as compare to others. Documentation is also a key issue in software development, in our **Question 9** we questioned about document procedure, we find developers support the procedure to make document should depend on the nature of project when it is required or necessary. But in plan-driven they are restricted to follow the document procedure. So according to our findings and observation in the research, practitioners' are motivated to maintain the quality of product and process and agile SQA activities are giving them ease to maintain it in better way.

## **6.6 Applying Extra Layer of QA**

As per our proposed approach, at least two of development teams must be allotted one QA expert to work as the member of each development team. This QA expert will not play the role of developer but as the monitoring authority of teams over all work and this QA expert will be called QA-node. This extra layer of QA may serve the development process in the following manner keeping agility of project maintained.

Testing is the soul of quality. Developer may carry out a large part of testing by adopting automated testing in organization. But it must be QA-nodes whom should write test cases rather than developers. More over manual testing must be carried out by these QA-nodes as they have expertise in testing and have the broader vision of this field.

Requirement Gathering / User Stories in agile methodologies are done by collaboration of user and developer. We suggest that QA-node and development team lead must carry out this task and then must explain these requirements to developers. Adopting this approach will help implementing user requirements more effectively even they are complex enough. On the other hand, presence of development team lead will allow, gathering user requirements from technical perspective as well. As in agile requirements are not extensively documented because user stories are written and developer interprets these stories to implement them as software functionality.

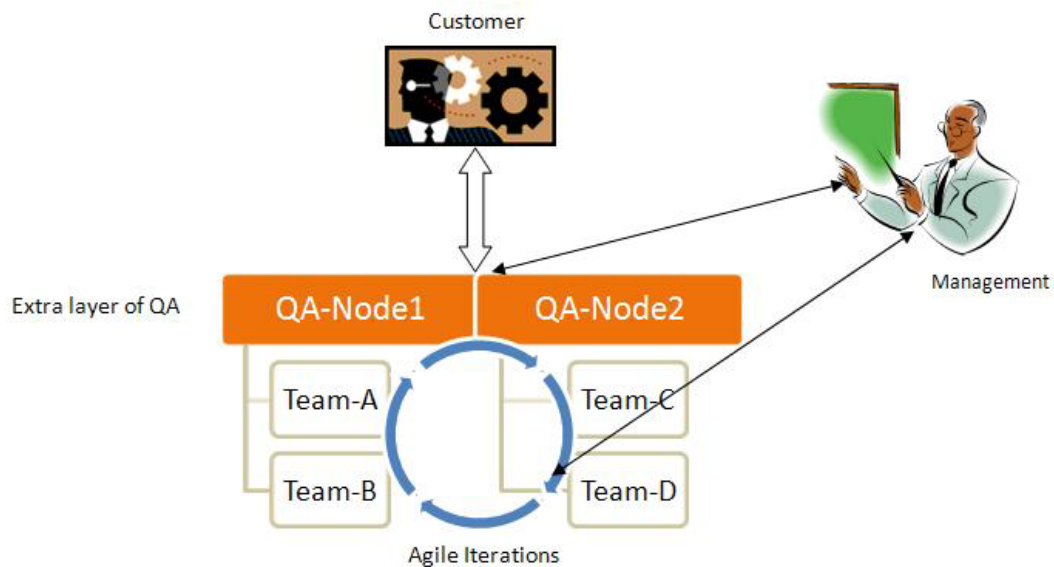
Collaboration is the fundamental element of agile projects. But when collaboration is done among the people of different approaches, it may lead to misunderstandings and wastage of time. We suggest that QA-nodes must be on up-front for customer collaboration because QA professionals are the people with technical as well as social understanding.

Documentation has been the critical issue among conventional and agile methodologies. Agile development is not against documentation but it encourages comprehensive documentation in order to save time and resources. In some situations when customer demand or software is complex, documentation needs to be extensive. As in plan-driven documentation is responsibility of QA, here in our proposed approach QA-nodes are supposed to perform this task only within their development teams. This may avoid the extra work load on developer.

Interaction within Extra Layer of QA must be one of the project routines like daily meeting. Development teams must interact with each other and with their QA-nodes. These QA-nodes must maintain their mutual interaction in order to measure the progress, discuss critical issues share knowledge and to share their resources within the project. Collectively, QA-nodes are seen as QA Extra Layer of the project.

SPI requires continuous fellowship of particular standards or models. QA personnel are the active resources to initiate and maintain SPI in an organization. They are experts and have sufficient knowledge of SPI. Besides, monitoring project QA-node may also take step to initiate SPI under agile environment.

Below we are going to present the organizational view of our proposed solution to overcome shortcoming in agile SQA.



**Figure 5: Organizational View of Agile Project after Injecting Extra Layer of QA**

In above given figure when have graphically injected extra layer of QA. Each QA-node has been assigned two development teams (A, B, C and D) respectively. Number of teams or QA-nodes may differ from organizational resources and level of experience of each node. These QA-nodes are keeping agile development alive within their teams and participating in teams as team member. Without disturbing the main mindset of development project, proposed extra layer of QA is virtually separated and should maintain collaboration with meeting of all QA-nodes. Customer is supposed to interact with QA-node and Team lead, mutually. Role of management is to keep watching and machining concerned decisions without disturbing technical decision as they are taken by the team to maintain agility.

### 6.6.1 Expected Disadvantages

During this study we have observed that proposed technique may have possible disadvantages before it gets mature by continuous research. Some of the expected disadvantages can be list as:

- **Time consumption** as in proposed technique the organization is required to change the way and procedural setup.
- **Shifting the some of the responsibilities** from developer to QA personnel may slow down the pace of projects, in the beginning.
- **Developing new tools** to follow proposed technique can be a big deal of time and cost, for small-scale organizations.

Advantages and disadvantages of any technique, in real environment, can be observed by implementing it. Due to limitations of our study this technique is not implemented to put forth results in real environment. Having implemented our proposed technique may lead to other disadvantages than listed above, which can be used to get this approach mature.

## 6.7 Results

Good business is the ultimate target of any organization and to sustain good business, organizations are focusing better quality. Fundamental approach of agile methodologies is quick response to changing requirements and rapid development of error free software product. While compared with agile, conventional development is document extensive, rigorous and slow process of software development. Despite of all facts agile methodology still needs to redefine its QA activities. In the first phase of our research work we gathered literature on agile QA from different sources. Literature and practitioners' expressions found on agile QA activities, was concentrated on testing. Through keen observation of data, sufficient space of improvement in agile QA activates was identified. There is no question on conventional or plan-driven development is more mature than agile development. But it is also a fact that agile development approach is being adopted rapidly over software development industry.

After finding bottlenecks and shortcomings in QA activities of agile development our second phase of research began. In second phase we proposed an Extra Layer of QA in agile projects. We observed organizations using agile development, target on quality of product by focusing on testing. On the other hand organizations following conventional development, concentrate on several aspects of quality by focusing on quality of product, project and process. Agile is criticised to be not productive in large organizations and complex systems. Through results we identified this critic, up to some extent, is valid as agile provides approach for close collaboration but not for organizing which need collaboration over large scale. Company A is a large scale company and practicing conventional as well as agile development, but organization is sustaining QA as a separate entity rather than integral part of development process. Going through this research work, we find that imparting expertise of conventional QA activities in agile development projects; we might achieve and sustain better quality of product and process.

## **CHAPTER 7: CONCLUSION**

Through this research work we conclude that agile development must also concentrate on quality of process to manage, organize and measure in better way. We think rather than investing resources and time on customer's technical education, organization must use the people who are qualified to bridge the gap between technicality and social demands. By integrating knowledge of conventional QA in agile projects, measured, standardized and development of complex systems can be maximized. No doubt, testing is the backbone of product quality, but quality can be attributed far more than testing. Besides utilizing a tester or designer as developer, their capabilities must be used to build organizational knowledge to increase the quality of process and product. Practices like pair programming and test-driven development are effective approaches in order to attain absence errors and refactoring is good to save time. But in agile development they must be quantified and standardized to achieve higher quality and to keep the project tracked and measured.

## **CHAPTER 8: FUTURE WORK**

In our survey, questionnaire was distributed to 6 convenience samples and only 4 feedbacks were received. To draw more effective results, such research must be conducted over large scale. Our proposed solution is not supported by statistical analysis. If this proposed solution is tested on a large scale organization, then we hope that empirical data from this test would be helpful to refine this proposed solution.

Many large scale organizations are not adopting agile development because they prefer quality of process as they prefer quality of product. But in agile development more focus in on product quality. We believe that higher quality of product and process might be attained injecting proposed extra layer of QA. But this is not a mature solution and it needs much research work done on it before it becomes effective and productive for agile development projects. In agile quality assurance most of research work is done in the area of testing. But to achieve higher quality, organized and standardized development, we must use quality experts in agile projects who have deeper knowledge of quality concerns.

## REFERENCES

- [1] “NASA Software Quality Assurance Center”, Available at <http://satc.gsfc.nasa.gov/assure/assurepage.html> , last ched: 04-01-2009
- [2] Agilemanifesto, Available at <http://www.agilemanifesto.org> , last checked: 05-01-2009
- [3] Miller, G. G. The Characteristics of Agile Software Processes. *The 39<sup>th</sup> International Conference of Object-Oriented languages and Systems* 2001.
- [4] Sajid Ibrahim Hashmi, Jongmoon Baik, “Software Quality Assurance in XP and Spiral - A Comparative Study”, *Fifth International Conference on Computational Science and Applications*.
- [5] E. Mnkandla, .B. Dwolatzky,” Defining Agile Software Quality Assurance”
- [6] Koch, Alan. “Agile Software Development : Evaluating the Methods for Your Organization”, Norwood, MA, USA: Artech House, Incorporated, 2004. p 134.
- [7] Charles Field, Elyse Sanchez, Sholom Sandalow , Alice Toth, “Agile Development with a Compass: Integrating User Experience Design”, April 2007.
- [8] W.E.Lewis. A Quality Framework for Software Testing
- [9] Bergman, B: Klefsjö, B. Quality from Customer Needs to Customer Satisfaction, Second edition, P 23
- [10] McCall, J. A., Richards, P. K., and Walters, G. F., "Factors in Software Quality", Nat'l Tech. Information Service, no.Vol.1, 2 and 3, 1977.
- [11] Boehm, B. W., Brown, J. R., Kaspar, H., Lipow, M., McLeod, G., and Merritt, M., Characteristics of Software Quality, North Holland, 1978.
- [12] Available at : [http://www.jasonunes.com/2008\\_04\\_01\\_archive.html](http://www.jasonunes.com/2008_04_01_archive.html), last check: 12-02-2009
- [13] By Ioannis G. Stamelos, Panagiotis Sfetsos, Agile software development quality assurance, Published by Information Science Reference, ISBN 1599042169, 9781599042169, 2007.
- [14] Koch, Alan. Agile Software Development: Evaluating the Methods for Your Organization. Norwood, MA, USA: Artech House, Incorporated, 2004. p134.
- [15] SCOTT AMBLER Ambyssoft, Inc. “Quality in an Agile World”, American Society for Quality, 2005.
- [16] Cockburn A. “Agile Software Development”, Publisher: Addison Wesley Longman, 2002, pages 304, ISBN-10: 0201699699.
- [17] *David Howard*, “Is Agile development a threat to the product management discipline?”, 280 Group, The Product Marketing & Product Management Experts, 2008.
- [18] “Agile Development - Opportunity or Threat?”, British Computer Society, 2006. Article available at: <http://www.bcs.org/server.php?show=conWebDoc.2766>.
- [19] T. Dyba° , T. Dingsøyr, “Empirical studies of agile software development: A systematic review”, ScienceDirect Information and Software Technology, 2008.



- [20] By John W. Creswell, "Research design: qualitative, quantitative, and mixed method approaches", Published by SAGE, ISBN 0761924426, 9780761924425, 2003.
- [21] Ahsan Nawaz, Kashif Masood Malik, "Software Testing Process in Agile Development", Computer Science Master Thesis, Blekinge Institute of Technology, 2008.
- [22] Definition available at: <http://www.askoxford.com>, last checked: 18-05-2009.
- [23] Foddy, W: Constructing Questions for Interviews and Questionnaires: Theory and Practice in Social Research. Cambridge: Cambridge University Press, 1993.
- [24] Oxford English Dictionary, <http://www.dictionary.oed.com>, Accessed 20/4/2009
- [25] J. Erickson, K. Lyytinen, K. Siau, Agile Modeling, Agile software development, and extreme programming: the state of research, Journal of Database Management 16 (4) (2005) 88–100.
- [26] L. Williams, A. Cockburn, "Agile software development: it's about feedback and change", IEEE Computer 2003
- [27] [27] Sajid Ibrahim Hashmi and Jongmoon Baik, "Software Quality Assurance in XP and Spiral - A Comparative Study", IEEE, 2007.
- [28] K. Beck, "Embracing change with Extreme Programming". IEEE Computer, Vol. 32, Issue 10 October 1999
- [29] Scrum and Quality Assurance, Posted by Michael James, 2007, available on [http://danube.com/system/files/Scrum+and+QA+blog\\_1.pdf](http://danube.com/system/files/Scrum+and+QA+blog_1.pdf), Accessed 3/5/2009.
- [30] Ron Jeffries, "What is Extreme programming" URL: <http://www.xprogramming.com/xpmag/whatisxp.htm>, (last checked: May 17, 2009).
- [31] Schwaber, K. and Beedle, M., "Agile Software Development with Scrum", 1st Edition, Publisher: Prentice Hall PTR, ISBN: 0130676349, 2001.
- [32] AvJim Highsmith, James A. Highsmith, "Agile Software Development Ecosystems", Publisher: Addison-Wesley, ISBN: 0201760436, 2006.
- [33] M. A. Awad, "A Comparison between Agile and Traditional Software Development Methodologies", IEEE, 2005.
- [34] AvDoug Rosenberg, Matt Stephens, Mark Collins-Cope, "Agile development with ICONIX process: people, process, and pragmatism", Publisher: av Apress, ISBN-1590594649, 2005.
- [35] Abrahamsson, P., Salo, O., Ronkainen, J., & Warsta, J., "Agile Software Development Methods: Review and Analysis", VTT Publications, 2002.
- [36] Saya Poyu Sone " MAPPING AGILE PROJECT MANAGEMENT PRACTICES TO PROJECT MANAGEMENT CHALLENGES FOR SOFTWARE DEVELOPMENT" pages: 191 Year: June, 2008
- [37] Martin Fowler, The New Methodology URL: <http://martinfowler.com/articles/newMethodology.html> 13 Dec 05 (last checked: May 10, 2009).

- [38] Rick Dove, Garry Turkington "Relating Agile Development to Agile Operations" Conference on Systems Engineering Research, 2008
- [39] Minna Pikkarainen, Outi Salo, and Jari Stillp "Deploying Agile Practices in Organizations: A Case Study" Springer-Verlag Berlin Heidelberg 2005
- [40] Leo R. Iyayasarathy, Dan Turk "Agile Software Development: A Survey of Early Adopters" A Publication of the Association of Management, Journal of Information Technology Management Volume XIX, Number 2, 2008 ISSN #1042-1319
- [41] Shine Technologies, "Agile Methodologies Survey Results," <http://www.shinetech.com/download/attachments/98/ShineTechAgileSurvey2003-01-17.pdf>, December 2007.
- [42] Digital Focus, "Agile 2006 Survey: Results and Analysis," Digital Focus was bought by Command Information in January of 2006.
- [43] Agile Manifesto, "Manifesto for Agile Software Development," <http://agilemanifesto.org/>, December 2007.
- [44] Aniket Mahanti, "Challenges in Enterprise Adoption of Agile Methods – A Survey" Journal of Computing and Information Technology - CIT 14, 2006, 3, 197–206
- [45] S. AMBLER, Remaining Agile, AmbySoft White Paper, 2005,
- [46] J. HIGHSMITH, Objections to Agile Development, Cutter Consortium White Paper, 2004,
- [47] Martin Fowler: The new methodology, <http://www.martinfowler.com/articles/newMethodology.html>, 26/4/2009
- [48] David Talby and Arie Keren, Orit Hazzan and Yael Dubinsky, "Agile Software Testing in a Large-Scale Project", IEEE SOFTWARE, 2006
- [49] David L. Parnas, Mark Lawford, "Inspection's Role in Software Quality Assurance", IEEE Computer Society 2003.
- [50] B. Boehm and R. Turner, "Using risk to balance agile and plan-driven methods," Computer, vol. 36, pp. 57-66, 2003.
- [51] Sridhar Nerur, Radha Kanta Mahapatra and George Mangalaraj, "Challenges of Migrating to Agile Methodologies", COMMUNICATIONS OF THE ACM May 2005.
- [52] Jonna Kalermo, Jenni Rissanen, "Agile Software Development in Theory and Practice", University of Jyväskylä, 2002.
- [53] Beck, K. "*Embracing change with extreme programming*", IEEE, 1999.
- [54] A. Cockburn and L. Williams, "The Costs and Benefits of Pair Programming," in Extreme Programming examined, G. Succi and M. Marchesi, Eds. Boston: Addison-Wesley, 2001, pp. xv, 569p.
- [55] Mikael Lindvall, Vic Basili, Barry Boehm, Patricia Costa, Kathleen Dangle, Forrest Shull, Roseanne Esoriero, Laurie Williams, and Marvin Zelkowitz, "Empirical Findings in Agile Methods", Springer-Verlag Berlin Heidelberg 2002.

- [56] Martin Fowler, Kent Beck, "Refactoring: Improving the Design of Existing Code", Published by Addison-Wesley, 1999, ISBN 0201485672, 9780201485677
- [57] Capiluppi, A., Fernandez-Ramil, J., Higman, J., Sharp, H.C., Smith, N., "An Empirical Study of the Evolution of an Agile-Developed Software System", 29th International Conference on Software Engineering, IEEE 2007
- [58] D. Jacob, Accelerating Process Improvement Using Agile Techniques, Auerbach Publications, 2006.
- [59] [www.iso.org](http://www.iso.org)
- [60] <http://www.sei.cmu.edu/cmmi/>
- [61] W.H.Morkel, Theunissen, Derrick G. Kourie and Bruce W. Watson, "Standards and Agile Software Development", Proceedings of SAICSIT 2003.
- [62] Hillel Glazer, Jeff Dalton, David Anderson, David J. Mike Konrad, Sandy Shrum, "CMMI® or Agile: Why Not Embrace Both!", Software Engineering Institute, Carnegie Mellon University, 2008.
- [63] Outi Salo & Pekka Abrahamsson, "Integrating Agile Software Development and Software Process Improvement: a Longitudinal Case Study", VTT Technical Research Centre of Finland, IEEE, 2005.
- [64] I. Aaen, A. B. Orjesson, L. Mathiassen, "SPI Agility: How to Navigate Improvement Projects", John Wiley & Sons, Ltd, 2006.
- [65] Laurie Williams, Alistair Cockburn, "Agile Software Development: It's about Feedback and Change", IEEE, 2003.
- [66] Myers, G.J. "The Art of Software Testing", New York: John Wiley & Sons, 1979.
- [67] Kent Beck, "Test-driven Development", Publisher: Addison-Wesley, ISBN-0321146530, 2003.
- [68] McBreen, P. Quality assurance and testing in agile projects. McBreen Consulting, 2003. Article available at: <http://www.mcbreen.ab.ca/talks/CAMUG.pdf>
- [69] Foddy, W: Constructing Questions for Interviews and Questionnaires: Theory and Practice in Social Research. Cambridge: Cambridge University Press, 1993.
- [70] L. Harison, "Political Research an Introduction", British Library Publication, ISBN: 0-415-22655-4.
- [71] Creswell, J. Research Design Qualitative, Quantitative and Mixed Method Approaches. Sage Publications Ltd. 2002.
- [72] William M.K. Trochim, "Qualitative Validity", Research Methods Knowledge Base, 2006, September 05, 2008, (<http://www.socialresearchmethods.net/kb/qualval.php>).
- [73] Object-Oriented Software Engineering. Ivar Jacobson, Addison Wesley Professional. ISBN 0-201-54435, 1992.

# APPENDIX

Given are the survey questions which were forwarded.

- 1: What approach are you using for product development?
- 2: What are the core quality assurance activities being followed in your organization, with respect to plan driven development?
- 3: If you are following agile development, what is the role of quality assurance in your projects?
- 4: If you are using or have been using a conventional approach for development, do you think quality assurance is complex and consumes more resources during projects when compared to agile development?
- 5: How many members are there in your QA team, and how many development teams do you have in your organization?
- 6: Comparatively, agile development focuses more upon product testing, throughout the project. Is it enough to have a better testing approach in order to produce quality products? Give your opinion.
- 7: From your project experience, do you think that a developer can follow all quality procedures without the assistance of QA personnel?
- 8: In plan driven development, if a project demands more customer involvement for product development, who should interact with the customer, and why
- 9: Do you think that extensive documentation is required for projects, either in plan driven or in agile development, or is it possible to limit the documentation in order to save time and resources?
- 10: Have you ever observed that your organization has had to compromise on process quality and prioritise product quality in agile development or vice versa in plan driven development, and if so, in which way?