Embedding Quality in Scrum: Study and Evaluation of Quality Management in Scrum in Pakistan

by

Amna Ejaz
MEM143009

A thesis submitted to the
Department of Mechanical Engineering
in partial fulfillment of the requirements for the degree of
Master of Science in Engineering Management

Faculty of Engineering
Capital University of Science and Technology
Islamabad
March, 2017

Dedicated to my Family

CAPITAL UNIVERSITY OF SCIENCE & TECHNOLOGY
ISLAMABAD
CERTIFICATE OF APPROVAL

Embedding Quality in Scrum: Study and Evaluation of Quality management in Scrum in Pakistan

by

Amna Ejaz

MEM143009

THESIS EXAMINING COMMITTEE

S No	Examiner	Name	Organization
(a)	External Examiner	Dr. Abdul Hameed	Iqra University, Islamabad
(b)	Internal Examiner	Dr. Abdul Baseer Qazi	CUST, Islamabad
(c)	Supervisor	Mr. Saad Ahmed	CUST, Islamabad

_____

Mr. Saad Ahmed

Thesis Supervisor

March, 2017

_____

Dr.Saif ur Rahman

Head of Department

Department of Mechanical Engineering

Dated : March, 2017

_____

Dr. Imtiaz Ahmed Taj

Dean

Faculty of Engineering

Dated : March, 2017

CERTIFICATE OF CHANGES INCORPORATED

This is to certify that Amna Ejaz has incorporated all the changes in the thesis suggested by internal or external supervisor under my supervision.

_____

Mr. Saad Ahmed

Thesis Supervisor

March, 2017

ACKNOWLEDGMENT

In the name of Allah the most merciful and gracious. All praises to Allah who enabled me to successfully complete my research work.

After an intensive period of effort and learning, I am finally writing this note of thanks to acknowledge the efforts of everyone who made this possible for me. I would first like to give my sincere gratitude to my supervisor Mr. Saad Ahmed for his continuous support, knowledge sharing and cooperation throughout this research.

I would also like to thank the participants of the survey and interviews conducted for the research. Thank you for your time and effort, without that it would not have been possible.

In the last, I want to thank my parents, siblings and friends from the core of my heart for the consistent motivation and encouragement throughout.

DECLARATION

It is declared that this is an original piece of my own work, except where otherwise acknowledged in text and references. This work has not been submitted in any form for another degree or diploma at any university or other institution for tertiary education and shall not be submitted by me in future for obtaining any degree from this or any other University or Institution.

Amna Ejaz
MEM143009

ABSTRACT

Software industry in Pakistan is shifting towards Agile project management to fulfill the current software development requirements of the industry. Transition from traditional methods of development to Agile methods is a challenging task and needs constant study and improvement of the process. Of all the Agile methods, Scrum is one of the most adopted method by the industry. A survey based research approach is used to identify quality related issues in Scrum and bring them to light. Qualitative and quantitative analysis is performed on the data to produce the results. The research highlighted issues in the areas of unit testing, customer collaboration, continuous integration and sprint planning. To overcome the highlighted issues, Scrum framework is merged with eXtreme Programming (XP). XP practices compliments Scrum process and provides a better framework for the development of quality products.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ACRONYMS

XP	eXtreme Programming
SM	Scrum Master
PO	Product Owner
US	User Story
CI	Continuous Integration

Chapter 1

Introduction

1.1 Overview

Ever since the advent of Agile, experiments and discussions are done to measure the effectiveness of Agile for software development. Agile was introduced to develop software in a dynamic environment where requirements are changing constantly during the development of the product. Different Agile methods and techniques are being developed to benefit with the principles of Agile in software development. The global software industry is adopting Agile because of the rapidly changing needs of the customer.

Of all the Agile methods used, Scrum is the most adopted method in the global software industry. Scrum is a complete framework consisting of different phases. Scrum development team has the flexibility to plan its own activities in cooperation with the Scrum Master who acts as a facilitator and manager for the development team. Product specifications are taken from Product Owner in the form of simple description known as User Stories. These user stories are then added to the Product Backlog. In each development iteration called Sprint, user stories from product backlog are selected on the basis of their priority and development team with the help of Scrum Master prepare a Sprint Backlog consisting of small tasks to be implemented in one sprint iteration. After the completion of a sprint cycle, a working part of a system is delivered to the product owner and next user story to be implemented is selected for next iteration. Since the product is developed in small chunks with constant collaboration and acceptance from

Product Owner, he/she has complete freedom to change the requirements during development of the product. Therefore, Scrum can produce software in a dynamic environment with high customer satisfaction.

Quality has always been an important factor in evaluating any software product. Quality can have two different perspectives, i.e. customer satisfaction and quality characteristics. Customer satisfaction can be achieved through the implementation of functional requirements and quality characteristics can be measured through non-functional attributes of the software. Both functional and non-functional aspects are important for the success of any software project/product. Scrum provides full independence in changing requirements during development, thus, user acceptance rate of product developed through Scrum is high. The Product Owner usually does not explicitly define non-functional requirements. Scrum team should infer and implement non-functional requirements according to the software specifications.

As Scrum allows fast paced development and short sprint cycles, developers are under high pressure to complete the functional requirements within limited time, they usually ignore or procrastinate non-functional requirements of the software. This results in piling up quality attributes that can create problems for developers as well as Product Owner in using the product.

1.2 Problem Statement

Scrum is most adopted Agile methodology in software industry. Scrum has been used to produce products in a dynamic environment with changing requirements. Although Scrum is a complete framework equipped with project management practices but

somehow it results in compromise on quality of the product. Many researchers have attempted to improve the framework but none of them focused on improving quality of the products produced through Scrum in Pakistan. The focus of the research is to identify the practices that result in decline of quality and provide suggestions to improve the process for betterment.

1.3 Objectives and Aims

Aim of the research is to study the quality management practices of software industry of Pakistan in the organizations that are implementing Scrum framework for development. The research with answer two research questions:

RQ1: What problems are faced by Scrum practitioners in Pakistan in achieving quality goals of the product?

RQ2: What improvements can be done to overcome the problems and improve quality of the products?

1.4 Approach of the Research

The study includes six chapters.

Chapter 1 gives the overview of the research.

Chapter 2 explains Scrum framework, problems faced in adopting Scrum and existing solutions to overcome the problems.

Chapter 3 discusses the methodology adopted to gather data for the research and data analysis technique to extract results from the data.

Chapter 4 contains the results obtained by analyzing data and analysis of the results to draw conclusions.

Chapter 5 highlights the findings of the research and provides discussion on the findings.

Chapter 6concludes the research work with recommendations for an improved framework and provides future work.

1.5 Chapter Summary

This chapter covered the overall introduction to the research. A brief summary of Scrum framework is discussed along with the problems associated with the process. The detail explanation of Scrum is included in Chapter 2. In the end, problem statement is declared and objectives are defined to achieve the research goals.

Chapter 2

Literature Review

2.1 Overview

Software development followed a defined and linear process in the beginning. Waterfall model was one of the earliest model for software development. It consists of pre-defined steps i-e Requirement gathering, Design, Development, Testing, Deployment and Maintenance [1].

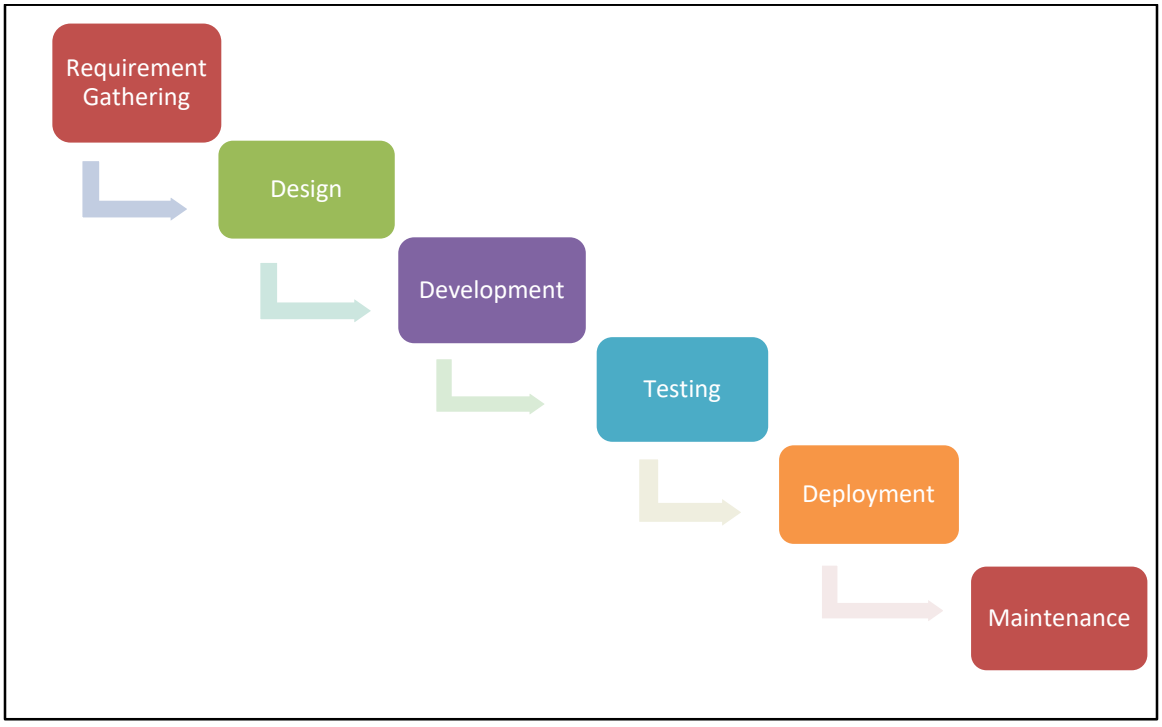

Figure 1: Waterfall Model

Each step has to be completed by the team before moving to the next step. The biggest problem with this model was the upfront extensive planning and lack of flexibility to incorporate changes in the system during development process [2].

Spiral model was introduced by Barry Boehm in 1988 [3] as an advancement in waterfall model. Spiral model is shown in the figure below:

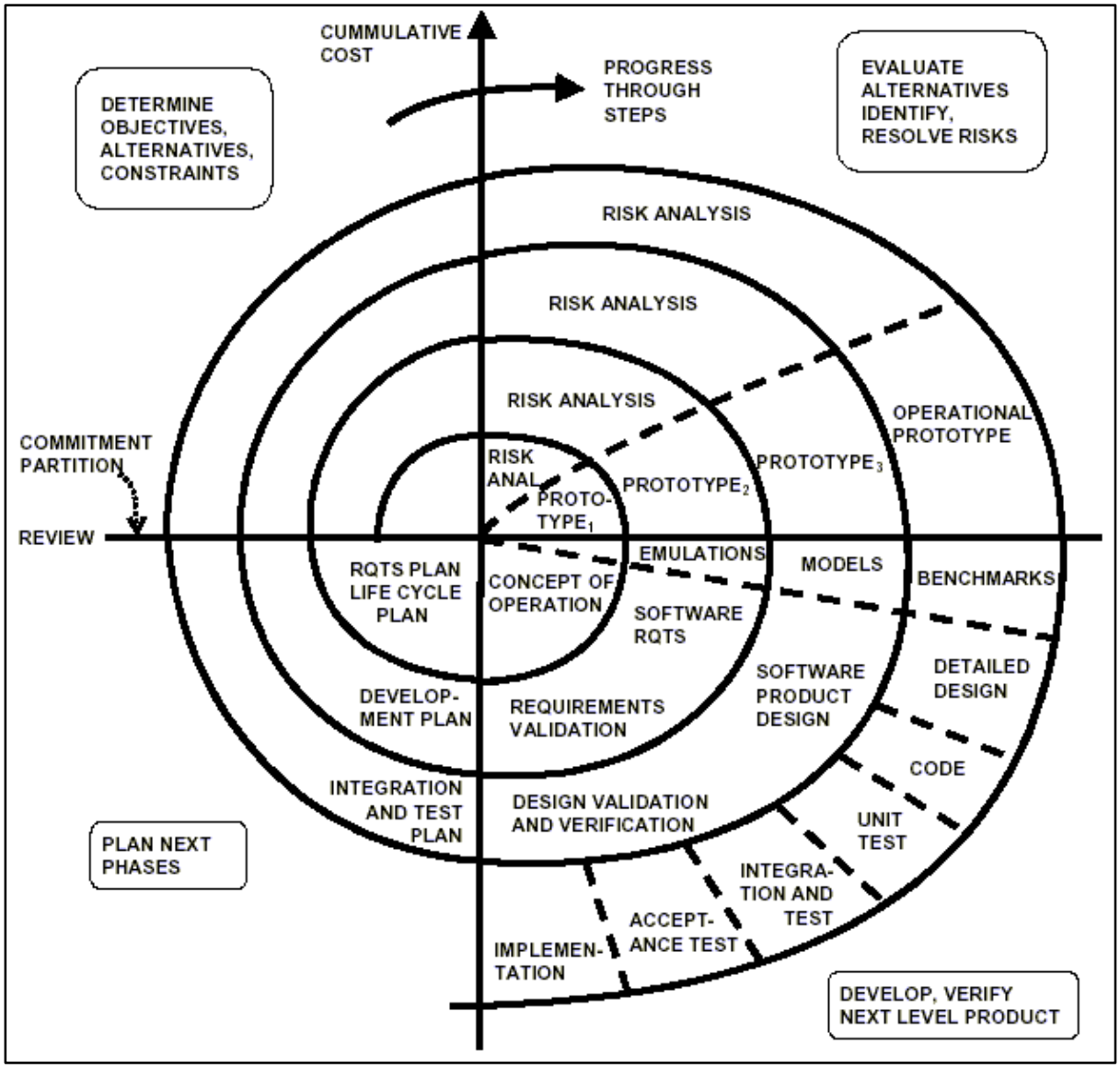

Figure 2: Spiral model [3]

A risk assessment activity was added after each phase of waterfall but the process was still linear and involved a lot of upfront planning with minimum customer involvement during the process.

Iterative Model was a major improvement in software development where a working part of the system was delivered to the customer early in the process [2]. System was divided into small modules. Each module was developed in one iteration where each iteration followed waterfall steps.

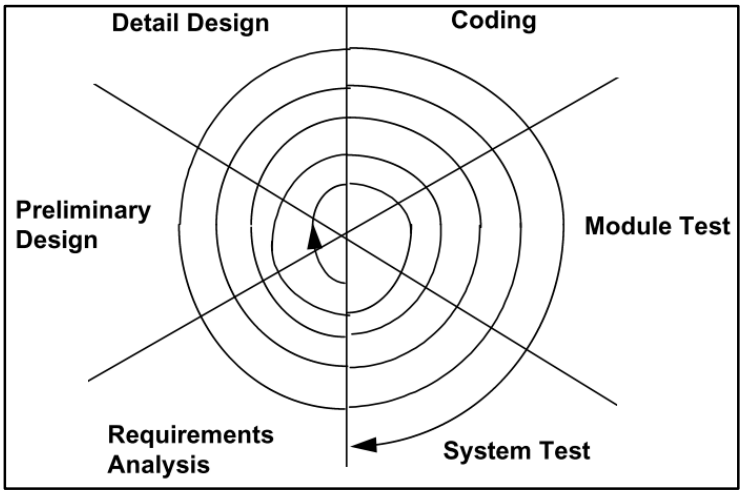

Figure 3: Iterative model [2]

All the development models still followed a predefined and linear process. In software industry, requirement change very rapidly. A main reason for project failure in software industry was unsatisfied customer because requirements were gathered in the start of the process when customer was not clear about the needs of the system. During development process, customer was completely cut off from the process resulting in a completely different final product then the one that customer had in mind.

Agile was introduced in 2002 with a completely different perspective. A group of experts from software industry highlighted the problems faced by practitioners every day and

proposed a detailed and practical approach in the form of Agile. Agile manifesto was based on four main principles [4]:

(1) Emphasis on people and interactions over processes and tools.

(2) Delivery of working software over extensive documentation.

(3) Constant customer collaboration over contract negotiation.

(4) Embracing change over following a plan.

Agile rapidly took over software industry due to its flexible nature. Different Agile methods were developed based on basic principles of Agile like eXtreme Programming (XP), Scrum, Lean Software Development, Crystal Clear methods, Adaptive Software Development (ASD), Kanban etc.

2.2 Description of Scrum

Scrum is one of the most used methodologies for Agile development. Scrum is a light weight development framework for managing software development in a changing environment [4]. Scrum consists of three main components: Roles, Processes and Artifacts [5].

2.2.1 Scrum Roles

Scrum consists of three main roles to run the process [6]:

a) Scrum Master (SM):

Scrum master plays the role of project manager in Scrum. Scrum master ensures the application of scrum practices and values by the team. SM plays the role of intermediary

between management and the team. SM is also responsible for smooth running of the process by arranging all the meetings.

b) Scrum team:

Scrum team is responsible for developing the system. Teams are self-organizing and cross functional and the roles of team members change with the situation.

c) Product Owner (PO):

Product Owner is a customer representative who works with the team as a team member. PO has the information about the product to be built and responsible for prioritizing user stories in product backlog. PO handles the business aspect of the product and act as a bridge between the stakeholders and the team.

2.2.2 Scrum Processes

Scrum process holds the following meetings for the smooth flow of work [4].

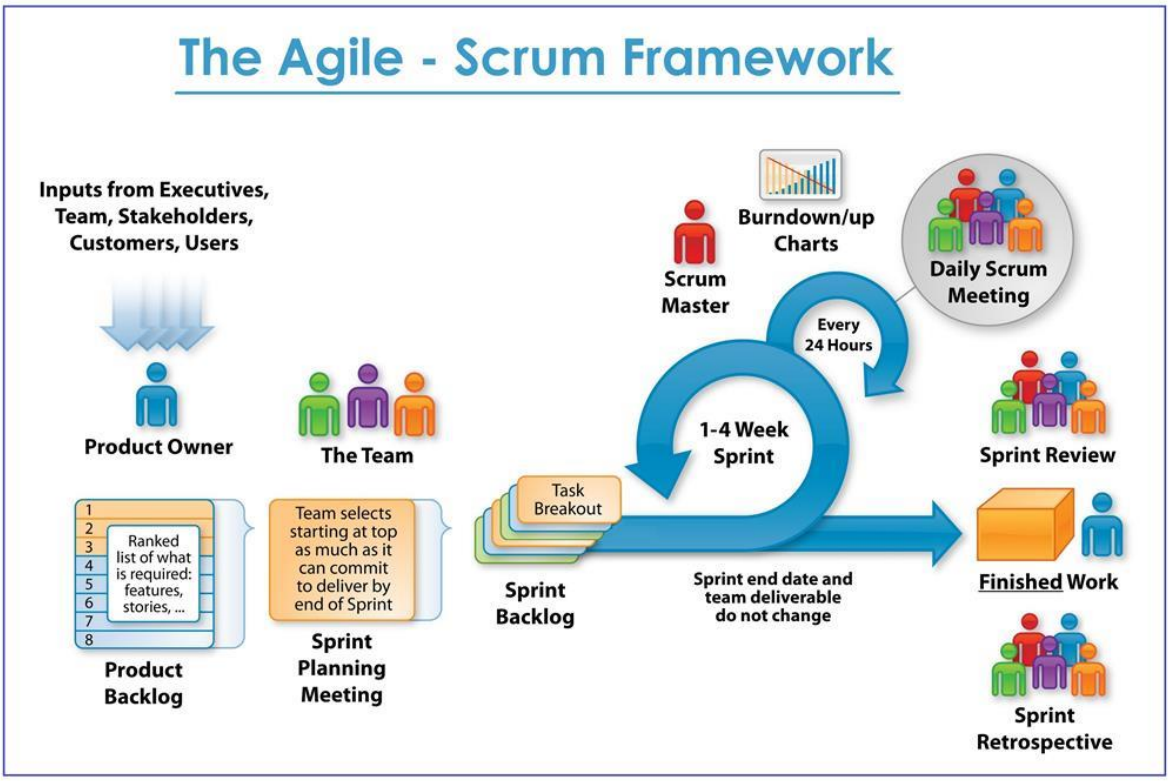

Figure 4: Scrum Process [7]

a) Kick off meeting:

Kick off meeting is held at the time of project initiation. Since the customer and teams are not sure about the details of the project, high level goals of product are set in product backlog.

b) Sprint Planning meeting:

In sprint planning meeting, SM, PO and Scrum team participates to define detailed product backlog with its prioritization. After the creation of product backlog, sprint backlog is created by picking the most important items from product backlog and breaking them into small implementable tasks.

c) Sprint:

Sprint is the development phase when scrum team implements tasks from sprint backlog. During sprint the requirements are not allowed to be changed.

d) Daily Scrum meeting:

Daily scrum meeting is a 15 min daily standup meeting between Scrum Mater and Scrum team where they share their previous day progress and current day plan with each other.

e) Sprint review meeting:

At the end of each sprint, sprint review meeting is held where Scrum team presents their work to product owner and PO gives feedback to the team on the product.

2.2.3 Scrum Artifacts

Scrum framework three main artifacts to support the process [5]:

a) Product Backlog:

Product Backlog contains prioritized list of product requirements and is owned by PO. Product backlog is created in project kickoff and is further refined in sprint planning meeting. Product Backlog cannot be changed until next sprint planning meeting

b) Sprint Backlog:

Sprint Backlog is created by Scrum team and is a subset of Product backlog. Scrum team picks the high priority items from Product backlog and breaks them down to smaller tasks

for sprint backlog. Team also provides estimate completion time of each task to identify the number of tasks that can be completed in the next sprint.

c) Burndown chart:

Burndown charts are used to track the progress of the project. Different burndown charts are created to track the progress of sprint, releases and product so that everyone in the team can evaluate their progress and work accordingly.

2.3 Software Industry and Scrum implementation

Some of the similar studies referring to the problems faced by the practitioners by implementing Scrum are discussed below:

Adopting a completely new and different development framework in a young software industry like Pakistan can raise a lot of problems. Akhtar *et al* [8] conducted interviews and used qualitative research to highlight some of the problems faced by the industry in adopting Scrum.

- Development team considers itself a sole stakeholder and does not involve other stakeholders during development. In Scrum, every stakeholder from Executives to customers, sponsors and managers have to be involved in the development process.
- Conventional decision-making style should be changed to democratic decision making.
- Teams are guided and controlled during development. They should be empowered to and facilitated for their tasks.

- Requirements gathering and analysis is an important step in the development of any product. Poor requirement elicitation phase results in rework for the development team and the team is not ready to accept change during development.
- Teams have an introvert behavior with customers due to the fear of changing requirements. Teams must understand that Scrum embrace change and that is the key to customer satisfaction.

Akif and Majeed [9] conducted a survey in two companies to report issues faced by them in implementing Scrum. Some of the issues highlighted by the research are:

- Risk management strategy is not defined in scrum framework.
- Limited development time hinders proper integration testing.
- Due to short development/sprint cycle, quality is ignored by the developers resulting in the pileup of quality related issues which will affect the overall performance of the system later.

Other identified issues belong to changing the mentality of team and management while moving from plan-driven development to Scrum. Solution was proposed for each problem but no quantitative evidence was provided to verify their usability and effectiveness in solving the problems.

Similar research was done by Ashraf and Mian [10] in identifying problems in Scrum implementation using survey and interviews. Problems identified in this research are:

- Constant interruption during development by customer and Scrum master.

- Scrum development needs an experienced team to cope up with the pace of development and dynamic environment.
- Scrum emphasize on delivering a working product in a sprint. Therefore, putting a lot of pressure on developers, which result in a compromise on quality and ignorance of most of the testing in order to deliver the product on time.

Suggestions were given to overcome the problems by improving Scrum framework.

França *et al* [11] selected 25 critical factors from the literature that are considered important for the success of projects managed through Scrum. These factors were tested against real projects to find the validity of these factors towards success. The result showed that only 8 factors actually contributed towards success of project including regular delivery, delivering most important features first and correct integration testing. However, the research cannot be generalized outside the context of the study.

A longitudinal survey was conducted by Tufail and Malik [12] in a software house to analyze the impact of scrum adoption on software quality. Software quality was measured through three parameters: severity of errors, defects and ratio of pass to fail test cases. Quantitative analysis was performed on the data gathered before and after adoption of scrum. The results showed significant reduction of serious errors and defects related to incorrect requirements and show-stopper errors that can cause a system to crash. However, errors and defects related to non-functional attributes of software were increased by 50% and 20% respectively. This can raise serious concerns about the quality of software developed through Scrum.

Cartaxo *et al* [13] conducted a survey to measure the relationship impact of Scrum on customer satisfaction. A total of 7 factors were selected to measure customer satisfaction that are: time, goals, quality, communication and transparency, agility, innovation, and benchmark. In the results, where scrum showed good results in the aspects of time, communication and transparency and agility, it failed to perform in innovation, benchmark, goals, and quality. However, the author concluded that there exists no relationship between scrum and customer satisfaction. This research has some limitations that are important to be highlighted. Firstly, the sample size for Scrum and non-Scrum projects was not the equal. And secondly, more factors can be added to measure customer satisfaction in terms of early delivery of a working system and acceptance of changing requirements during the development phase.

Chauhan [14] discussed the factors responsible for reopening a User Story (US)/bug in Scrum development. A US/bug can be reopened by Business Analyst, Developer, Quality Analyst (QA) or Environmental factors.

- The Business Analyst will reopen a US/bug in case of unclear, incomplete or unrealistic requirements.
- The Developer will reopen a US/bug due to insufficient unit testing or incorrect implementation of any requirement.
- QA will reopen a US/bug if the developed product fails acceptance testing due to incomplete implementation of US.

The author concluded that most of the time a US/bug is reopened by developer or QA. Although it was a qualitative research and no practical evidence was provided to prove

these factors. The results can question the adequacy of Scrum framework as there are no defined testing principles and quality assurance guidelines present in the framework to ensure the correct implementation.

2.4 Existing solutions to improve Scrum framework

No single development framework is ideal for all kinds of projects and environments but every framework can be generalized to provide maximum benefits for most of the projects. To fill the gaps present in scrum framework highlighted above, many researchers had experimented with scrum framework by combining it with other Agile and non-agile frameworks in search for a better framework.

Rong *et al* [15] combined a plan driven process, PSP (personal software process) with scrum to achieve the benefits of both agility and concrete practices.

PSP (personal software process):

PSP is a process designed for individual use that applies to structured personal tasks. The PSP process starts with a requirements statement, the first step in the PSP process is planning. There is a planning script that guides this work and a plan summary for recording the planning data. While the engineers are following the script to do the work, they record their time and defect data on the time and defect logs. At the end of the job, during the postmortem phase (PM), they summarize the time and defect data from the logs, measure the program size, and enter these data in the plan summary form. When done, they deliver the finished product along with the completed plan summary form [16].

The method proposed by Rong *et al* [15] is iterative. First iteration starts with requirement gathering and their prioritization and planning related to number of iterations and cycle time. Then the product is developed in different iteration. In a single iteration, task and quality plan is developed and risk strategy is defined. Iteration is ended with a postmortem, evaluating the team performance and goals achieved in the iteration. System and integration testing is optional after each iteration. The proposed model was tested on a real life project that produces good results in terms of productivity and code quality. However the drawback of the model is that customer did not get an early delivery. Product will be delivered when it is complete.

Hayata & Han [17] also attempted to combine a traditional plan driven approach i-e V Model with Scrum to help the software industry in transition towards agile methods gradually.

V Model:

V - Model is an extension of the waterfall model and is based on association of a testing phase for each corresponding development stage. This means that for every single phase in the development cycle there is a directly associated testing phase. This is a highly disciplined model and next phase starts only after completion of the previous phase. Under V-Model, the corresponding testing phase of the development phase is planned in parallel. So there are Verification phases on one side of the .V. and Validation phases on the other side. Coding phase joins the two sides of the V-Model [18].

In the hybrid model [17], requirements gathering and design phase follow waterfall approach. Once the design is complete, the development phase follows agile iterative

approach. Unit testing is done in each iteration with development. After complete development, integration and system testing is done following waterfall model. The drawback of the proposed model is that all the requirements are collected in the start and are fixed during development. This can decrease customer satisfaction since customer requirements can change during development. Also the model is not tested in any real life project to verify its applicability.

Apart from cross-framework integration, many attempts are done to integrate Scrum with different agile models. Qureshi [19] proposed a hybrid model that combines the practices of XP (eXtreme Programming) and Scrum.

eXtreme Programming (XP):

XP is a light weight software development methodology, first introduced by Kent Beck in 1999 [20]. XP compliments Agile as it enables development in vague and changing environment. XP consists of a set of practices, values and principles defined as a guideline for team members to help them develop software in a rapidly changing environment [21].

XP Practices

Some of the major practices of XP are summarized below [22]:

- Planning game:

During the planning phase, developers provide estimates for user stories. Then customer decides which user stories will be implemented in the next iteration.

- Small releases:

Rather than planning and designing for a complete system, small modules are developed and released often.

- Simple design:

Design of the system should be simple and contains minimum classes and methods covering all the requirements. Design should run all the tests at every moment.

- Testing:

Programmer writes unit test and customer write functional test for the stories. System should run all the tests at any moment.

- Refactoring:

Refactoring is the process of modifying and upgrading the design of the system with time. But it should still run all the tests.

- Pair Programming:

Development is done by programmers in pairs helping them to share ideas with each other.

- Continuous Integration:

Integration should be done on daily basis and it must run all testes or the changes should be discarded.

- On-site customer:

Customer should be a part of the team and should be present on the site during development.

XP Values

The underlying core values of XP used to guide the practices are following [23] [24]:

- Communication:

All Agile frameworks emphasize on constant and effective communication within team and with customer. XP also consider effective communication to be the most import of all values since it plays an important role in success and failure of the project.

- Simplicity:

Agile was introduced with the idea to embrace change [25]. Since the future is ambiguous and full of changes, designing for future is useless. XP emphasize on designing for today only and keep design as simple as possible. Simplicity in design helps in expansion and integration of the product.

- Feedback:

XP practices enable the team members and customer to get the feedback on the product early in the process. XP practices like Continuous integration, small releases and testing helps in getting the feedback on the system which helps both customer and developers to plan for the future accordingly.

- Courage:

  Incomplete information is a developer's greatest fear as well as a risk to the system. XP encourage programmers to be aggressive and develop enough courage to start developing the system with little information and let time unfolds the rest.

In the proposed model [19] sprint zero is introduced in the model to estimate the effort of product attributes before the creation of product backlog. Development is done in different sprint iterations. XP practices like pair programming, unit testing and continuous are practices during development phase. After an iteration, a working part of the product is released to customer. Proposed model is tested by a controlled case study which shows good results for customer satisfaction but the study lacks a comparison to prove its better results than Scrum.

Mushtaq and Qureshi [26] also integrated XP and Scrum to combine management practices from scrum and engineering practice from XP to achieve good quality and customer satisfaction. The proposed process gives complete detail about the steps to be performed in sprint zero which was left for the organization to decide in the traditional Scrum model. The process is consists of a complete life cycle steps. In planning phase, system definition is developed and high level design of the system is developed. The design phase follow keep it simple rule. Class diagram and object diagram are developed for all the requirements in sprint backlog. In development phase, XP practices like pair programming, coding standards and continuous integration are practiced to ensure the quality and productivity. The model is tested in controlled environment but it should be tested with real life projects to verify its effectiveness in complex projects.

A recent attempt was done by Ahmed *et al* [27] to integrate XP, Scrum and RUP (rational unified process). A combination of agility with a plan driven approach is proposed to produce high quality software.

RUP (rational unified process):

RUP is a software development process framework that provides a disciplined approach to assigning tasks and responsibilities within a development organization. Its goals is to ensure the production of high quality software that meets the needs of its end users within a predictable schedule and budget. The software life cycle of RUP is decomposed into four sequential phases: Inception, Elaboration, Construction and Transition. In Inception phases, business case for the project is established and its feasibility is ensured. Elaboration phase is where the software architecture is established that provides a stable foundation for design and implementation. In the construction phase remaining requirements are clarified and development of the system is completed based on the architecture build during elaboration phases. In the transition phase acceptance testing is done with user and the system is deployed in user's environment [28].

The customized framework [27] covers the complete lifecycle from inception to deployment. In the inception phase, a little upfront planning is done regarding the requirements, budget, and team and risk management. The construction phase consists of iterations which produces deliverable modules with proper testing. The last phase is the transition phase in which system testing is done and deployment procedures are followed. The proposed model combines the management and engineering practices from three

different frameworks however it should be tested in real practice environment to verify its effectiveness and prove its quality claims.

Table 1 shows the summary of all the hybrid models discussed in the literature along with their improvements they provide in Scrum framework.

Hybrid Model	Improvement in Scrum
PSP + Scrum [15]	• Manageability • Predictability
V- model + Scrum [17]	• Manageability • Risk assessment
XP + Scrum [19]	• Customer Satisfaction
XP + Scrum [26]	• Customer Satisfaction
XP + RUP + Scrum [27]	• Structured approach to software development

Table 1: Summary of hybrid models

Many structural and implementation issues are highlighted in literature and attempts are made to improve Scrum process but none of the hybrid models are focused on improving the quality of products produced through Scrum. Therefore this research is aimed to target this area. The research will identify the quality issues faced by Scrum practitioners in Pakistan and will provide the suggestions to improve the quality of the products.

2.5 Chapter Summary

This chapter discusses the Scrum framework at length; Scrum roles, processes and artifacts are explained for the understanding of Scrum. Then the application of Scrum has

been studied in software industry to highlight the concerns of practitioners and researchers about the framework. Last part of the chapter summarizes the attempts that have been made by researchers to improve Scrum framework.

Chapter 3

Methodology

3.1 Research approaches:

There are two basic approaches to research, i.e., **quantitative** approach and the **qualitative** approach.

3.1.1 Quantitative approach:

The former involves the generation of data in quantitative form which can be subjected to rigorous quantitative analysis in a formal and rigid fashion. This approach can be further sub-classified into inferential, experimental and simulation approaches to research. The purpose of **inferential** approach to research is to form a database from which to infer characteristics or relationships of population. This usually means survey research where a sample of population is studied (questioned or observed) to determine its characteristics, and it is then inferred that the population has the same characteristics. **Experimental** approach is characterized by much greater control over the research environment and in this case some variables are manipulated to observe their effect on other variables. **Simulation** approach involves the construction of an artificial environment within which relevant information and data can be generated. This permits an observation of the dynamic behavior of a system (or its sub-system) under controlled conditions [29].

3.1.2 Qualitative approach:

Qualitative approach to research is concerned with subjective assessment of attitudes, opinions and behavior. Research in such a situation is a function of researcher's insights

and impressions. Such an approach to research generates results either in non-quantitative form or in the form which are not subjected to rigorous quantitative analysis [30]. Generally, the techniques of focus group interviews, projective techniques and depth interviews are used.

3.2 Research methods:

Research methods or techniques refer to the methods the researchers use in performing research operations. In other words, all those methods which are used by the researcher during the course of studying his research problem are termed as research methods. Since the object of research, particularly the applied research, it to arrive at a solution for a given problem, the available data and the unknown aspects of the problem have to be related to each other to make a solution possible. Keeping this in view, research methods can be put into the following three groups [29]:

3.2.1 Group-I: Library research:

In the first group those methods are included which are concerned with the collection of data. These methods will be used where the data already available are not sufficient to arrive at the required solution. Methods used for library research are analysis of historical records and analysis of documents.

3.2.2 Group-II: Field research:

The second group consists of those statistical techniques which are used for establishing relationships between the data and the unknowns. Methods used for field research are mass observation, mail questionnaire, personal interview, case study and life history etc.

3.2.3 Group-III: Laboratory research:

The third group consists of those methods which are used to evaluate the accuracy of the results obtained. Methods used for this research are small group study of random behavior, play and role analysis.

The research method used to conduct this research is field research. Both survey and interviews are conducted to gather data for the research. Through this data, relationships can be established between different variables to infer results. Thus the research uses both qualitative and quantitative research approaches to answer the research questions. Inferential approach which is a type of quantitative research is used for survey analysis and descriptive approach which is a type of qualitative research is used for analysis of interviews.

3.3 Research Designs:

Different research designs can be conveniently described if we categorize them as [31]:

1. Research design in case of exploratory research studies
2. Research design in case of descriptive and diagnostic research studies
3. Research design in case of hypothesis-testing research studies

3.3.1 Research design in case of exploratory research studies:

Exploratory research studies are also termed as formulative research studies. The main purpose of such studies is that of formulating a problem for more precise investigation or of developing the working hypotheses from an operational point of view. The major emphasis in such studies is on the discovery of ideas and insights. Generally, the following three methods in the context of research design for such studies are talked

about: (a) the survey of concerning literature; (b) the experience survey and (c) the analysis of 'insight-stimulating' examples.

3.3.2 Research design in case of descriptive and diagnostic research studies:

Descriptive research studies are those studies which are concerned with describing the characteristics of a particular individual, or of a group, whereas diagnostic research studies determine the frequency with which something occur or its association with something else. In a descriptive/diagnostic study the first step is to specify the objectives with sufficient precision to ensure that the data collected are relevant. Then comes the question of selecting the methods by which the data are to be obtained. In other words, techniques for collecting the information must be devised. Several methods (i.e., observation, questionnaires, interviewing, examination of records, etc.) are available for this purpose and the researcher may use one or more of these methods for data collection. The data collected must be processed and analyzed. This includes steps like coding the interview replies, observations, etc.; tabulating the data; and performing several statistical computations. Last of all comes the question of reporting the findings. This is the task of communicating the findings to others and the researcher must do it in an efficient manner. The layout of the report needs to be well planned so that all things relating to the research study may be well presented in simple and effective style.

3.3.3 Research design in case of hypothesis-testing research studies:

Hypothesis-testing research studies (generally known as experimental studies) are those where the researcher tests the hypotheses of causal relationships between variables. Such

studies require procedures that will not only reduce bias and increase reliability, but will permit drawing inferences about causality.

For our exploratory research, experience survey is used to collect data. The goal of the survey was to address the problems faced by the practitioners/professionals who are implementing Scrum framework and relate them with the problems discussed in literature review. Questionnaire is attached in Appendix - A. Questionnaire of the survey consists of 2 sections. Section 1 contains the personal and organizational attributes of the professional. Section 2 contains questions related to testing and implementation of Scrum framework in the organization to find the problems faced by users and customers and map them with literature. The questionnaire was designed by taking help from [32][33][34][35][36]. Survey was conducted in a time period of 3 months from April 2016 to June 2016. 34 participants contributed in the survey from 21 different companies from Pakistan. 28 questionnaires were considered complete and were included in data analysis. Participants include Scrum masters, Agile coach, Software Engineers and Quality Assurance Engineers and managers.

The secondary source for data collection in our research is interviews. Interviews were conducted to validate the results of survey and to get an inside view of an organization implementing Scrum methodology. Six interviews were conducted in Elixir Technologies, an international technological organization established in 1985 and running their offices in Asia, Europe, the Middle East and headquarters in the United States [37]. One Scrum Master, two developers and two Quality Assurance Engineers participated in the interviews. Interview questions were designed to understand the

development life cycle practiced in the organization, problems faces by the team members in implementing Scrum and suggestions to improve the framework to overcome the problems.

3.4 Sample Designs:

There are different types of sample designs based on two factors viz., the representation basis and the element selection technique. On the representation basis, the sample may be probability sampling or it may be non-probability sampling. Probability sampling is based on the concept of random selection, whereas non-probability sampling is 'non-random' sampling. On element selection basis, the sample may be either unrestricted or restricted. When each sample element is drawn individually from the population at large, then the sample so drawn is known as 'unrestricted sample', whereas all other forms of sampling are covered under the term 'restricted sampling'. Figure 5 shows the sample designs as explained above.

Element selection technique ↓ / Representation basis →	Probability sampling	Non-probability sampling
Unrestricted sampling	Simple random sampling	Haphazard sampling or convenience sampling
Restricted sampling	Complex random sampling (such as cluster sampling, systematic sampling, stratified sampling etc.)	Purposive sampling (such as quota sampling, judgement sampling)

Figure 5: Basic sampling designs [29]

Non-probability sampling:

Non-probability sampling is that sampling procedure which does not afford any basis for estimating the probability that each item in the population has of being included in the sample. Under non-probability sampling the organizers of the inquiry purposively choose the particular units of the universe for constituting a sample on the basis that the small mass that they so select out of a huge one will be typical or representative of the whole. In such a design, personal element has a great chance of entering into the selection of the sample.

Probability sampling:

Probability sampling is also known as ‘random sampling’ or ‘chance sampling’. Under this sampling design, every item of the universe has an equal chance of inclusion in the sample. The results obtained from probability or random sampling can be assured in terms of probability i.e., we can measure the errors of estimation or the significance of results obtained from a random sample, and this fact brings out the superiority of random sampling design over the deliberate sampling design.

Random sampling technique is used in our research to collect data. All the participants for survey and interview are selected on random basis. After collection, significance of results can be calculated by different methods.

3.5 Research Variables

Variables have been extracted from the questionnaire. Variables’ description with their possible values is defined below:

Variable 1: Organization Size

It is used to identify the size of the organization. Possible values are:

- 1-50
- 51-100
- 101-500
- 501-1000
- >1000

Variable 2: Planning of activities for managing software quality

This variable is used to identify if the activities are planned by the team to manage software quality of the project or not. Possible values of the variable are:

- Yes
- No

Variable 3: Defining quality goals for project

At the time of project initiation, high level goals of the project are defined. This variable is used to identify who is involved in the process of defining quality goals for project. Possible values of the variable are:

- Project Manager/ Scrum Master
- Customer
- Development team
- All of them together

Variable 4: Defining quality goals for sprint

Before each iteration/sprint, sprint goals are set. This variable is used to identify who sets the quality goals for sprint. Possible values of the variable are:

- Project Manager/ Scrum Master
- Customer
- Development team
- All of them together

Variable 5: Performing testing during sprint

This variable is used to identify if testing is being performed during development iteration or not. Possible values of the variable are:

- Agree
- Disagree

Variable 6: Who is performing testing during sprint

This variable is used to identify who is performing testing during sprint. Possible values of the variable are:

- Developer
- Tester
- Not applicable

“Not applicable” value identifies that testing is not being performed during sprint.

Variable 7: Performing integration testing

This variable identifies if integration testing is performed after each iteration or not. Possible values of the variable are:

- Agree
- Disagree

Variable 8: Feeling pressure during sprint

This variable is used to identify if the team members are feeling any pressure during sprint to achieve their goals or not. Possible values are:

- Yes
- No

Variable 9: Skipping unit testing

This variable is used to find out if the team members are skipping unit testing due to lack of time or not. Possible values are:

- Yes
- No

Variable 10: Use of automated testing tools

This variable is used to find out if automated testing tools are used for testing or not. Possible values are:

- Agree

- Disagree

Variable 11: Time constraints to perform complete testing

This variable is used to identify if the team has enough time to perform all possible tests on the product or not. Possible values are:

- Yes
- No

Variable 12: Checking quality of product

This variable is used to identify who is checking the quality of the product from the team. Possible values are:

- Customer
- Quality Manager/Assurer
- Project Manager/Scrum Master
- Senior Management
- All of them together

Variable 13: Frequency of reporting quality defects in the product after delivery

This variable is used to measure the frequency of defects reported by the customers after the delivery of the product. Possible values are:

- Very often
- Sometimes
- Never

Value “very often” means that the defects frequency is high. “Sometimes” means that the defects frequency is medium and “never” means that the defects frequency is almost zero or minimum.

Variable 14: Training of personnel

This variable is used to identify if the individuals have received any training for working in Scrum environment to understand it or not. Possible values are:

- Yes
- No

Variable 15: Practicing Continuous Integration

This variable is used to identify if the team members practice continuous integration during development to integrate new modules with the existing system. Possible values are:

- Yes
- No

Variable 16: Combining Scrum with other development methodologies

This variable is used to identify if the team is merging Scrum practices with other Agile or non-Agile methodologies to get better results. Possible values are:

- Yes
- No

3.6 Techniques for data analysis:

The term analysis refers to the computation of certain measures along with searching for patterns of relationship that exist among data-groups [29]. Thus, in the process of analysis, relationships or differences supporting or conflicting with original or new hypotheses should be subjected to statistical tests of significance to determine with what validity data can be said to indicate any conclusions. Analysis, particularly in case of survey or experimental data, involves estimating the values of unknown parameters of the population and testing of hypotheses for drawing inferences. Analysis may, therefore, be categorized as descriptive analysis and inferential analysis (Inferential analysis is often known as statistical analysis) [31].

Descriptive analysis:

Descriptive analysis is largely the study of distributions of one variable. This study provides us with profiles of companies, work groups, persons and other subjects on any of a multiple of characteristics such as size. Composition, efficiency, preferences, etc. This sort of analysis may be in respect of one variable (described as unidimensional analysis), or in respect of two variables (described as bivariate analysis) or in respect of more than two variables (described as multivariate analysis).

Inferential analysis:

Inferential analysis is concerned with the various tests of significance for testing hypotheses in order to determine with what validity data can be said to indicate some conclusion or conclusions. It is also concerned with the estimation of population values.

It is mainly on the basis of inferential analysis that the task of interpretation (i.e., the task of drawing inferences and conclusions) is performed.

The important statistical measures that are used to summarize the survey/research data are: (1) measures of central tendency or statistical averages; (2) measures of dispersion; (3) measures of asymmetry (skewness); (4) measures of relationship; and (5) other measures.

This research will measure relation between variables to identify those variables that are causing the decline in quality of the products. Since all the variables in the research are nominal variables, only those tests can be performed that work with nominal data. Chi-square test of independence is used to analyze nominal data [38] but if the sample size is small (i-e less than 20), its results become invalid [39]. In this case Fisher Exact test is used for analysis.

3.6.1 Fisher Exact Test

When the sample size is small (i-e less than 20) and the expected values in a contingency table is less than 5, chi-square test cannot be used. In this case Fisher Exact test is used for data analysis [40].

Fisher exact test is used to determine dependence between two nominal variables. Hypothesis testing is used in Fisher exact test. Null hypothesis states that the variables are independent while alternate hypothesis states that dependence exist between the variables. However, it does not explain the cause and effect relationship between them.

Fisher exact test was presented by Fisher, Irwin and Yates in 1930 for 2x2 contingency table but it can be extended for nxm tables [41].

3.6.1.1 Application of Fisher Exact test

The application of Fisher Exact test is explained below with a worked example. Data is taken from the questionnaire of current research. Two variables have been taken from the questionnaire to apply the test on. First variable represents if the participants feels any pressure during the sprint cycle to achieve iteration goals within time limit. Second variable represents if the participants have ever skipped unit testing during sprint cycle due to lack of time. Hypothesis testing is used in Fisher Exact test. Following steps are performed to apply Fisher exact test on data:

STEP 1: State Null and Alternate Hypothesis

Null Hypothesis: Both variables are independent

Alternate Hypothesis: Variables are dependent on each other.

STEP 2: Set Significance level

Significance level of 0.05 is used as a standard in hypothesis testing [42].

STEP 3: Create Contingency table

A contingency table is a matrix that contains the frequency distribution of two categorical variables. One variable represent rows and second represent columns of the table. Contingency tables are used to find the correlation between two categorical variables.

Table 2 shows the contingency table for the variables discussed above. Rows represent the first variable whereas columns represent second variable.

	Skipping unit testing		Total
Feeling pressure during sprint	No	Yes	
No	10	2	12
Yes	6	10	16
	16	12	28

Table 2: Contingency table

The general form of table is given in table 3 [43].

	Column 1	Column 2	Total
Row 1	A	B	a+b
Row 2	C	D	c+d
Total	a+c	b+d	a+b+c+d

Table 3:General form of a Contingency table [43]

Where a, b, c, d are frequency count and row and column total are called marginal totals.

STEP 4: Calculate p-value

P-value represents the probability to mistakenly reject null hypothesis. Formula to calculate p-value is given in equation 1 [44].

$$p = \frac{(a+b)!(c+d)!(a+c)!(b+d)!}{(a+b+c+d)!a!b!c!d!}$$ ….. eq. (1) [44]

Inserting values from table 2 in equation 1

$$p = \frac{(12)!\,(16)!\,(16)!\,(12)!}{(28)!\;10!\;2!\;6!\;10!}$$

$p = 1.0044 \times 10^{44} / 5.7814 \times 10^{45} = 0.017$

p = 0.017 means that there is a 1.7% probability that you will mistakenly reject null hypothesis.

STEP 5: Compare p-value with significance level

According to hypothesis test, if p-value is less than significance level then null hypothesis is rejected i-e the alternate hypothesis is accepted and if p-value is greater than significance level then alternate hypothesis is rejected [42]. If alternate hypothesis is rejected, it does not mean that null hypothesis is true. It means that the data does not provide significant proof to support the acceptance of alternate hypothesis [45].

In the worked example p = 0.017 and Significance level is 0.05 since p-values is less than significance level, it means that the alternate hypothesis is true i-e there exist a dependence between the two variables. However it does not explain the cause and effect relationship between them.

3.7 Contingency tables for research

Contingency table is created between two variables to find the dependence between them. Following contingency tables are created in the research to find the relation between different variables with each other:

1. Organization size and frequency of reporting quality defects in the product after delivery
2. Define quality goals for project and frequency of reporting quality defects in the product after delivery
3. Performing testing during sprint and feeling pressure during sprint
4. Feeling pressure during sprint and skipping unit testing
5. Organization size and use of automated testing tools
6. Defining quality goals for project and checking quality of product
7. Checking quality of product and frequency of reporting quality defects in the product after delivery
8. Training of personnel and feeling pressure during sprint
9. Combining Scrum with other development methodologies and frequency of reporting quality defects in the product after delivery
10. Performing testing during sprint and frequency of reporting quality defects in the product after delivery
11. Defining quality goals for sprint and who is performing testing during sprint
12. Practicing Continuous Integration and frequency of reporting quality defects in the product after delivery

3.8 Chapter Summary

This chapter discusses the approach used to conduct the research. Survey and interviews are conducted to gather data. Then selection of data analysis technique is discussed according to the type of data gathered through the survey. Fisher Exact test is selected as an appropriate analysis method. In the end research variables are defined on which the test will be performed to acquire the results.

Chapter 4

Results and Analysis

Since manual calculation of p-value for all contingency tables is complex, Ri386 software is used to perform the calculations.

4.1 Applying Fisher Exact test on data

Since manual calculation of p-value for all contingency tables is complex, Ri386 software is used to perform the calculations. The p-value calculated through Fisher Exact test shows if the two variables depend on each other or not.

Table 1:

Table 4 shows the contingency table for organization size and frequency of reporting quality defects in the product after delivery.

	Frequency of reporting quality defects in the product after delivery		
Organization Size	Never	Sometimes	Very often
<50	1	8	3
>1000	2	3	0
101-500	2	3	2
501-1000	0	1	0
51-100	1	2	0

Table 4: Contingency table for organization size and frequency of reporting quality defects in the product after delivery

The p-value calculated through Fisher exact test for table 4 is **p-value = 0.7624**

Alternate Hypothesis	Confidence level	Alternate Hypothesis Conclusion
Dependent	50%	Reject
Dependent	80%	Reject
Dependent	90%	Reject
Dependent	95%	Reject
Dependent	99%	Reject

Table 5: Testing alternate hypothesis for Table 4 on different confidence levels

Results from Table 4 and Table 5 show that the size of organization does not depend on how often they face defect complaints from customer. Any organization, big or small can achieve quality goals and customer satisfaction by correctly implementing the defined procedures [46]. Motorola managed to win the Baldrige award for its excellent quality

practices despite being a big and complex organization. Whereas Stoner Inc. sets an example for small organizations by winning the award too for good quality practices [47].

Table 2:

Table 6 shows the contingency table for define quality goals for project and frequency of reporting quality defects in the product after delivery.

	Frequency of reporting quality defects in the product after delivery		
Define quality goals for project	Never	Sometimes	Very often
All of them together	1	11	1
Project Manager/ Scrum Master	5	6	4

Table 6: Contingency table for define quality goals for project and frequency of reporting quality defects in the product after delivery

The p-value calculated through Fisher exact test for table 6 is **p-value = 0.07161**

Alternate Hypothesis	Confidence level	Alternate Hypothesis Conclusion
Dependent	50%	Accept
Dependent	80%	Accept
Dependent	90%	Accept
Dependent	95%	Reject

Dependent	99%	Reject

Table 7: Testing alternate hypothesis for Table 6 on different confidence levels

Results from Table 6 and Table 7 shows that defect reporting from customer depends on who defines quality goals for the project. Results from the survey shows that either Scrum Master or the whole team is involved in defining quality goals for project in organizations. Customers should be involved from the beginning of the project since customer is important for competitive quality [48]. Agile emphasizes on customer involvement from the beginning till the end to ensure the success of the project [49].

R. Hoda*et al* [50] conducted a survey and discussed the adverse effects of lack of customer involvement on the team and project.

Table 3:

Table 8 shows the contingency table for performing testing during sprint and feeling pressure during sprint.

	Feeling pressure during sprint	
Performing testing during sprint	Agree	Disagree
No	10	2
Yes	14	2

Table 8: Contingency table for performing testing during sprint and feeling pressure during sprint

The p-value calculated through Fisher exact test for table 8 is **p-value = 1**

Alternate Hypothesis	Confidence level	Alternate Hypothesis Conclusion
Dependent	50%	Reject
Dependent	80%	Reject
Dependent	90%	Reject
Dependent	95%	Reject
Dependent	99%	Reject

Table 9 : Testing alternate hypothesis for Table 8 on different confidence levels

Table 9 shows that p-value calculated for table 8 rejects the alternate hypothesis proving that the variables do not depend on each other. This means that there is a constant pressure on team members during the sprint either testing is performed during sprint or not.

Table 4:

Table 10 shows the contingency table for feeling pressure during sprint and skipping unit testing.

	Skipping unit testing	
Feeling pressure during sprint	No	Yes
No	10	2
Yes	6	10

Table 10 : Contingency table for feeling pressure during sprint and Skipping unit testing

The p-value calculated through Fisher exact test for table 10 is **p-value = 0.02347**

Alternate Hypothesis	Confidence level	Alternate Hypothesis Conclusion
Dependent	50%	Accept
Dependent	80%	Accept
Dependent	90%	Accept
Dependent	95%	Accept
Dependent	99%	Reject

Table 11 : Testing alternate hypothesis for Table 10 on different confidence levels

Table 10 shows the dependence between the two variables. Frequency counts from table 9 shows that unit testing is skipped by the team members who are feeling pressure during sprint however those not feeling any pressure do not skip unit testing. M. Ashraf and N. Ali [10] highlighted the same problem in the survey where quality is compromised by the developers by skipping testing to meet sprint deadline. M. Cristal *et al* [51] suggested that sprint planning should be done properly to incorporate sufficient time for development and testing after identifying the same issue in the survey. B. Fitzgerald and K. Stol [52] suggested "Continuous *" model. According to this approach, continuous testing should be done during development since it reduces the development time by 15% and saves more time for testing. Also C. Jakobsen and J. Sutherland [53] suggested that project planning should be done before project initiation. As a result of planning, a Quality Assurance Schedule (QAS) can be defined. It contains information about the quality aspects to be checked in the final product. It can help the team to estimate time for testing activities during sprint keeping it as a reference.

Table 5:

Table 12 shows the contingency table for organization size and use of automated testing tools.

	Use of automated testing tools	
Organization size	Agree	Disagree
<50	9	3
>1000	5	0
101-500	6	1
501-1000	1	0
51-100	2	1

Table 12 : Contingency table for organization size and use of automated testing tools

The p-value calculated through Fisher exact test for table 12 is **p-value = 0.7806**

Alternate Hypothesis	Confidence level	Alternate Hypothesis Conclusion
Dependent	50%	Reject
Dependent	80%	Reject
Dependent	90%	Reject
Dependent	95%	Reject
Dependent	99%	Reject

Table 13 : Testing alternate hypothesis for Table 12 on different confidence levels

Results from table 12 and table 13 indicate that the use of automated testing tools in an organization does not depend on the size of organization. Since Scrum framework

supports short development cycle and fast development of product, there is no time for manual testing [54] thats why every organization that uses Scrum for development, use automated testing tools to complete the testing procedures during sprint regardless of its size.

Table 6:

Table 14 shows the contingency table for defining quality goals for project and checking quality of product.

	Checking quality of product			
Defining quality goals for project	All of them together	Customer	Quality Manager/Assurer	Senior Management
All of them together	7	1	4	1
Project Manager/ Scrum Master	5	3	7	0

Table 14 : Contingency table for defining quality goals for project and checking quality of product

The p-value calculated through Fisher exact test for table 14 is **p-value = 0.4459**

Alternate Hypothesis	Confidence level	Alternate Hypothesis Conclusion
Dependent	50%	Accept
Dependent	80%	Reject
Dependent	90%	Reject
Dependent	95%	Reject
Dependent	99%	Reject

Table 15 : Testing alternate hypothesis for Table 14 on different confidence levels

Results from Table 14 shows that it does not depend who defines quality goals and who checks it. Quality goals are defined during the high level goal setting in the initial planning phase [55]. Then these goals are shared with everyone on the team because agile team share collective goals and visions [56]. So it does not matter who checks the quality because everyone shares same project goals.

Table 7:

Table 16 shows the contingency table for checking quality of product and frequency of reporting quality defects in the product after delivery.

	Frequency of reporting quality defects in the product after delivery		
Checking quality of product	Never	Sometimes	Very often
All of them together	1	9	2
Customer	2	1	1
Quality Manager/Assurer	3	6	2
Senior Management	0	1	0

Table 16 : Contingency table for checking quality of product and frequency of reporting quality defects in the product after delivery

The p-value calculated through Fisher exact test for table 16 is **p-value = 0.521**

Alternate Hypothesis	Confidence level	Alternate Hypothesis Conclusion
Dependent	50%	Reject
Dependent	80%	Reject
Dependent	90%	Reject
Dependent	95%	Reject
Dependent	99%	Reject

Table 17 : Testing alternate hypothesis for Table 16 on different confidence levels

Table 16 and table 17 shows that the occurrence of defects does not depend on who is checking the quality of the product. As discussed above that the quality goals are shared

among the Scrum team so it does not matter who checks the quality. Rather it depends on whether the right quality goals are set for the project or not.

Table 8:

Table 18 shows the contingency table for training of personnel and feeling pressure during sprint.

	Feeling pressure during sprint	
Training of personnel	No	Yes
No	3	6
Yes	9	10

Table 18 : Contingency table for Training of personnel and Feeling pressure during sprint

The p-value calculated through Fisher exact test for table 18 is **p-value = 0.687**

Alternate Hypothesis	Confidence level	Alternate Hypothesis Conclusion
Dependent	50%	Reject
Dependent	80%	Reject
Dependent	90%	Reject
Dependent	95%	Reject
Dependent	99%	Reject

Table 19 : Testing alternate hypothesis for Table 18 on different confidence levels

P-value from Table 18 shows that both trained and untrained professionals are feeling pressure to achieve iteration goals. Team members should be trained properly to better understand the working of Scrum [9]. Even if few team members are trained, the other

members of the team will let them down because it is a collective effort. So eventually they will start to feel the pressure too.

Table 9:

Table 20 shows the contingency table for combining Scrum with other development methodologies and frequency of reporting quality defects in the product after delivery.

	Frequency of reporting quality defects in the product after delivery		
Combining Scrum with other development methodologies	Never	Sometimes	Very often
No	5	10	2
Yes	1	7	3

Table 20 : Contingency table for combining Scrum with other development methodologies and frequency of reporting quality defects in the product after delivery

The p-value calculated through Fisher exact test for table 20 is **p-value = 0.3614**

Alternate Hypothesis	Confidence level	Alternate Hypothesis Conclusion
Dependent	50%	Accept
Dependent	80%	Reject
Dependent	90%	Reject
Dependent	95%	Reject
Dependent	99%	Reject

Table 21 : Testing alternate hypothesis for Table 20 on different confidence levels

Results from Table 20 indicate that defect occurrence does not depend on whether Scrum is used in combination with other methodologies or not. Many experiments have been done to improve Scrum framework by combining it with other Agile and non-Agile methods as discussed in literature. But if Scrum is implemented with correct procedures, it has the capability to produce quality products. J. Li *et al*[57] conducted an experiment to calculate defect density in the product before and after adoption of Scrum. The results of defect density remained the same because it depends on team members' knowledge about the development framework and how the team follows the procedures.

Table 10:

Table 22 shows the contingency table for performing testing during sprint and frequency of reporting quality defects in the product after delivery.

	Frequency of reporting quality defects in the product after delivery		
Performing testing during sprint	Never	Sometimes	Very often
Agree	6	15	3
Disagree	0	2	2

Table 22 : Contingency table for performing testing during sprint and frequency of reporting quality defects in the product after delivery

The p-value calculated through Fisher exact test for table 22 is **p-value = 0.2195**

Alternate Hypothesis	Confidence level	Alternate Hypothesis Conclusion
Dependent	50%	Accept
Dependent	80%	Reject
Dependent	90%	Reject
Dependent	95%	Reject
Dependent	99%	Reject

Table 23 : Testing alternate hypothesis for Table 22 on different confidence levels

Table 22 and table 23 shows that reporting of defects does not depend on either testing is done during sprint or not. Refer to Table 10 results, it depends on whether testing is done or not. Defects will occur if testing is skipped because of lack of time or inadequate sprint planning.

Table 11:

Table 24 shows the contingency table for defining quality goals for sprint and who is performing testing during sprint.

	Who is performing testing during sprint	
Defining quality goals for sprint	Developer	Tester
All of them together	2	8
Customer	0	1
Development Team	2	2
Project Manager/ SM	4	9

Table 24 : Contingency table for defining quality goals for sprint and who is performing testing during sprint

The p-value calculated through Fisher exact test for table 24 is **p-value = 0.6715**

Alternate Hypothesis	Confidence level	Alternate Hypothesis Conclusion
Dependent	50%	Reject
Dependent	80%	Reject
Dependent	90%	Reject
Dependent	95%	Reject
Dependent	99%	Reject

Table 25 : Testing alternate hypothesis for Table 24 on different confidence levels

Table 24 shows that the variables do not depend on each other. Once the quality goals for the sprint are defined, they are shared with the whole team so it does not natter who does the testing to ensure them.

Table 12:

Table 26 shows the contingency table for practicing Continuous Integration and frequency of reporting quality defects in the product after delivery.

	Frequency of reporting quality defects in the product after delivery		
Practicing Continuous Integration	Never	Sometimes	Very often
No	2	2	4
Yes	4	15	1

Table 26 : Contingency table for practicing continuous integration and frequency of reporting quality defects in the product after delivery

The p-value calculated through Fisher exact test for table 26 is **p-value = 0.00829**

Alternate Hypothesis	Confidence level	Alternate Hypothesis Conclusion
Dependent	50%	Accept
Dependent	80%	Accept
Dependent	90%	Accept
Dependent	95%	Accept
Dependent	99%	Accept

Table 27 : Testing alternate hypothesis for Table 26 on different confidence levels

P-value for Table 26 shows significance dependence between the variables. Frequency counts from the contingency table show that defects have been reported by the customers even with the practice of Continuous Integration. Because Continuous Integration is

successful only if the previous phase of unit testing is done properly [58]. Refer to Table 9, when unit testing is skipped to meet sprint deadline, issues will pile up in the integration phase and have more severe consequences leading to defects.

Chapter 5

Findings and Discussion

5.1 Findings

The results from survey and interviews highlighted the following findings in Scrum process:

- Customer being the most important stakeholder of any project is ignored by the team members. He/she is not involved in the project on daily basis. Team members are performing the tasks that customer is supposed to do. This results in compromise on quality and low customer satisfaction.
- During sprint, team members feel constant pressure to meet the deadline thus compromising on quality by skipping testing activities on the product. This results in pile up of issues that showed up in the integration phase.
- Wrong application of Continuous Integration is being practiced by the organizations. New modules are integrated with rest of the system without complete and successful unit testing, causing bigger issues in integration phase.
- All the organizations are using automated tools for development and testing to cope up with the fast paced development.
- All members of Scrum team share same values and information, helping them to work as a team as well as achieving their individual goals.

5.2 Discussion

Every software development methodology has its pros and cons. Many researchers have combined different software development methodologies to minimize their deficiencies and maximize their advantages to develop a better technique for development. Scrum provides a set of management practices but lack engineering practices to develop a quality product [26]. Many efforts were made by researchers to combine Scrum with traditional and Agile methods to develop a new and improved methodology that overcome the shortcomings of Scrum. M. Qureshi [19] incorporated eXtreme Programming (XP) practice into Scrum and achieved 80% - 90% customer satisfaction level as well as a better quality product.

Chapter 6

Conclusion and Recommendations

6.1 What problems are faced by Scrum practitioners in Pakistan in achieving quality goals of the product? (RQ1)

Agile gave a new perspective of development to Software industry. Different Agile methods are being practiced in many organizations of Pakistan according to their needs and suitability. Scrum is one of the most used Agile method because of its defined management practices to handle any project. Software industry in Pakistan is still new and growing and the companies are struggling to meet the quality standards of foreign clients.

The first research question was to identify the quality issues in the organizations that are implementing Scrum in Pakistan. Results gathered through survey and interviews highlighted the following issues:

- Ignorance of customer by the team from the beginning of the project.
- Omission of Unit testing by the team due to time constraints and pressure to meet deadlines.
- Wrong application of Continuous Integration practiced by the team, causing defects in the product that are identified at later stages of testing.

6.2 What improvements can be done to overcome the problems and improve quality of the products? (RQ2)

The second research question was to find the solution for the problems highlighted through the research. So to overcome the shortcomings of Scrum, XP practices can be incorporated in Scrum process. XP provides a set of simple practices that can guide the Scrum team to overcome their existing problems and improve their development framework for better quality products and its combination with Scrum has showed good results in literature as well.

- Lack of customer involvement in Scrum can be overcome by adapting XP practices like “On-site customer” and “Planning game”. Agile emphasizes on constant customer collaboration in its manifesto [25]. XP ensures constant customer collaboration through its practices. In XP, requirement prioritization is done by customer [22]. User stories are not assigned to the iterations in the beginning of the project. Before each iteration, customer picks a user story to be implemented in the coming iteration. Also customer provides system wide test cases for the stories before each iteration to define the definition of done. This gives the team clear idea about what the customer expects from a given module [20]. These practices keep customer involved in the process till the completion of the project ensuring customer satisfaction.
- No clear practice of unit testing is defined in Scrum framework. Organizations usually improvise the process according to their suitability. But the findings of the research showed the lack of practice of unit testing by them. Unit testing is the heart of XP [22]. In XP, testing is different than conventional unit testing. Test cases

should be written before implementing a task. Programmers should write the test case before they code[22]. This practice eliminates excess pressure from the developers and team as they will know clearly what exactly is to be implemented in the iteration and when it is done.

- XP states clear guidelines for Continuous Integration as well. The research highlighted the wrong application of Continuous Integration in the organizations. New modules are integrated in the existing system without complete testing, causing defects in the delivered system. The practice of "Continuous Integration" in XP can eliminate the problem. XP suggests that a new module must not be integrated in the existing system without complete and successful testing [23]. After integration, all the tests must run or the new module must not be integrated [22].
- XP advocates for simple design [24]. Design should be done for each iteration to keep the process simple. Complex design of the system in the beginning of the project can complicate things during development. Simple and iterative design helps in fast and error free development of the system. It also helps in incorporating changing requirements of the customer during the project. Good design is easy to modify and test and can save a lot of time in development and testing.

6.3 Limitations and Future work

The research has the following limitations:

- The research uses a small sample size to address the research question.
- This research is localized to Pakistan software industry only.

- The suggestions made to overcome the highlighted problems are not tested in real environment to verify its results.

The research can be extended in the future by implementing the improved Scrum process in the organizations and studying its effectiveness on the quality of the products.

REFRENCES

[1] S. Balaji, "Waterfall vs v-model vs agile : A comparative study on SDLC," *WATEERFALL Vs V-MODEL Vs Agil. A Comp. STUDY SDLC*, vol. 2, no. 1, pp. 26–30, 2012.

[2] K. Schwaber, "Scrum development process," *Proc. Work. Bus. …*, no. April 1987, pp. 10–19, 1995.

[3] B. Boehm, "A Spiral Model of Software Development," *IEEE Comput.*, 1988.

[4] H. F. Cervone, "Understanding agile project management methods using scrum," *Int. Digit. Libr. Perspect.*, vol. 27, no. 1, pp. 18–22, 2011.

[5] J. Cho, "Issues and Challenges of Agile Software Development with Scrum," *Issues Inf. Syst.*, vol. 9, no. 2, pp. 188–195, 2008.

[6] M. Mahalakshmi and M. Sundararajan, "Traditional SDLC Vs Scrum Methodology – A Comparative Study," *Int. J. Emerg. Technol. Adv. Eng.*, vol. 3, no. 6, pp. 2–6, 2013.

[7] K. Yelkar, "The Agile - Scrum Framework." [Online]. Available: http://www.c-sharpcorner.com/UploadFile/d9c992/the-agile-scrum-framework/. [Accessed: 30-Nov-2016].

[8] M. J. Akhtar, A. Ahsan, and W. Z. Sadiq, "Scrum adoption, acceptance and implementation (A case study of Barriers in Pakistan's IT Industry and Mandatory Improvements)," in *Proceedings - 2010 IEEE 17th International Conference on Industrial Engineering and Engineering Management, IE and EM2010*, 2010, pp. 458–461.

[9] R. Akif and H. Majeed, "Issues and Challenges in Scrum Implementation," *Int. J. Sci. Eng. Res.*, vol. 3, no. 8, pp. 1–4, 2012.

[10] M. Ashraf and N. A. L. I. Mian, "Agile Project Management : Identification and Analysis of Problems in Scrum Implementation," *VAWKUM Trans. Comput. Sci.*, vol. 2, no. 2, 2013.

[11] a. C. C. França, F. Q. B. da Silva, and L. M. R. de Sousa Mariz, "An empirical study on the relationship between the use of agile practices and the success of Scrum projects," in *Proceedings of the 2010 ACM-IEEE International Symposium on Empirical Software Engineering and Measurement*, 2010, pp. 1–4.

[12] R. Tufail and A. A. Malik, "A Case Study Analyzing the Impact of Software Process Adoption on Software Quality," *10th Int. Conf. Front. Inf. Technol.*, pp. 254–256, 2012.

[13] B. Cartaxo, A. Araujo, A. S. Barreto, and S. Soares, "The impact of scrum on customer satisfaction: An empirical study," in *Proceedings - 2013 27th Brazilian Symposium on Software Engineering, SBES 2013*, 2013, pp. 129–136.

[14] V. K. Chauhan, "Why Reopen Counts are high in Scrum Development?," *Int. J. Eng. Dev. Res.*, vol. 2, no. 1, pp. 2321–9939, 2014.

[15] G. Rong, D. Shao, and H. Zhang, "SCRUM-PSP: Embracing process agility and discipline," *Proc. - Asia-Pacific Softw. Eng. Conf. APSEC*, pp. 316–325, 2010.

[16] W. Humphrey and J. Over, "The Personal Software Process (PSP) tutorial," *Proc. - 27th Annu. NASA Goddard Softw. Eng. Work. SEW 2002*, no. November, pp. 39–92, 2002.

[17] T. Hayata and J. Han, "A hybrid model for IT project with Scrum," in *Proceedings of 2011 IEEE International Conference on Service Operations, Logistics and Informatics*, 2011, pp. 285–290.

[18] PMoC, "V-model testing," *Integr. Vlsi J.*, 2003.

[19] M. R. J. Qureshi, "Empirical Evaluation of the Proposed eXSCRUM Model : Results of a Case Study," *J. Comput. Sci.*, vol. 8, no. 3, pp. 150–157, 2011.

[20] K. Beck and C. Andres, *Extreme Programming Explained, Second Edition: Embrace Change*, 2nd ed. Addison-Wesley Professional, 2004.

[21] R. C. Martin, "Agile Software Development, Principles, Patterns, and Practices," *Book*. p. 529, 2002.

[22] K. Beck, "Embracing change with extreme programming," *Computer (Long. Beach. Calif).*, vol. 32, no. 10, pp. 70–77, 1999.

[23] L. Lindstrom and R. Jeffries, "Extreme programming and agile software development methodologies," *Inf. Syst. Manag.*, vol. 21, pp. 41–52, 2004.

[24] J. Newkirk, "Introduction to agile processes and extreme programming," *Proc. 24th Int. Conf. Softw. Eng. ICSE 2002*, pp. 695–696, 2002.

[25] "Manifesto for Agile Software Development." [Online]. Available: http://agilemanifesto.org.

[26] Z. Mushtaq and M. R. J. Qureshi, "Novel Hybrid Model: Integrating Scrum and XP," *Int. J. Inf. Technol. Comput. Sci.*, vol. 4, no. 6, pp. 39–44, 2012.

[27] G. Ahmad, T. Soomro, and M. Brohi, "XSR: Novel Hybrid Software Development Model (Integrating XP, Scrum & RUP)," *Ijsce.Org*, no. 3, pp. 126–130, 2014.

[28] A. . Fallis, "An Introduction To The Rational Unified Process," *J. Chem. Inf. Model.*, vol. 53, no. 9, pp. 1689–1699, 2013.

[29] C. R. Kothari, *Research Methodology: Methods & Techniques*. 2004.

[30] C. Neville, "Introduction to Research and Research Methods," *Bradford Univ.*, pp. 1–5, 2005.

[31] P. Pandey Meenu Mishra Pandey, *Research Methodology:Tools and Techniques*. 2015.

[32] Sauce labs, "TESTING TRENDS IN 2016 : A SURVEY OF SOFTWARE," 2016.

[33] V. Garousi and J. Zhi, "The Journal of Systems and Software A survey of software testing practices in Canada," *J. Syst. Softw.*, vol. 86, no. 5, pp. 1354–1376, 2013.

[34] P. Haberl, A. Spillner, K. Vosseberg, and M. Winter, “Survey 2011 : Software Test in Practice,” 2011.

[35] R. Torkar, S. Mankefors, P. O. Box, and S.- Trollhättan, “A Survey on Testing and Reuse,” in *IEEE International Conference on Software:Science, Technology & Engineering*, 2003, pp. 164–173.

[36] D. G. David Zubrow, Will Hayes, Jane Siegel, “Maturity Questionnaire,” 1994.

[37] “http://www.elixir.com/apac/.” .

[38] C. R. Mehta and N. R. Patel, “Exact Inference for Categorical Data,” *Encyclopedia of biostatistics*, pp. 1411–1422, 1998.

[39] A. Agresti, *Categorical Data Analysis*. New York: Wiley., 1990.

[40] G. Yates, D., Moore, Moore, D., McCabe, *The Practice of Statistics*, 1st ed. New York: W.H. Freeman., 1999.

[41] C. R. M. and N. R. Patel, “A Network Algorithm for Performing Fisher’s Exact Test in r × c Contingency Tables,” *J. Am. Stat. Assoc.*, no. August 2014, pp. 37–41, 2012.

[42] W. R. Rice, “Analyzing tables of Statistical Tests,” *Soc. study Evol.*, vol. 43, no. 1, pp. 223–225, 1989.

[43] J. V Freeman and M. J. Campbell, “THE ANALYSIS OF CATEGORICAL DATA : FISHER ’ S EXACT TEST,” 2011.

[44] P. Sprent, “Fisher Exact Test,” *International Encyclopedia of Statistical Science*. Springer Berlin Heidelberg, pp. 524–525, 2014.

[45] D. M. Lane, “Interpreting Non-Significant Results.” [Online]. Available: http://onlinestatbook.com/2/logic_of_hypothesis_testing/nonsignificant.html. [Accessed: 15-Aug-2016].

[46] A. Ghobadian and D. Gallear, “TQM and organization size,” *Int. J. Oper. Prod. Manag.*, vol. 17, no. 2, pp. 121–163, 1997.

[47] J. R. Evans, *Quality & performance excellence*. Cengage Learning, 2013.

[48] C. a Lengnick-Hall, “Customer contribution to quality: A different view of the customer-oriented firm,” *Acad. Manag. Rev.*, vol. 21, no. 3, pp. 791–824, 1996.

[49] H. Saiedian and R. Dale, “Requirements engineering: Making the connection between the software developer and customer,” *Inf. Softw. Technol.*, vol. 42, no. 6, pp. 419–428, 2000.

[50] R. Hoda, J. Noble, and S. Marshall, “The impact of inadequate customer collaboration on self-organizing Agile teams,” *Inf. Softw. Technol.*, vol. 53, no. 5, pp. 521–534, 2011.

[51] M. Cristal, D. Wildt, and R. Prikladnicki, “Usage of SCRUM practices within a global company,” *Proc. - 2008 3rd IEEE Int. Conf. Glob. Softw. Eng. ICGSE 2008*, pp. 222–226, 2008.

[52] B. Fitzgerald and K.-J. Stol, “Continuous software engineering and beyond: trends and challenges,” *Proc. 1st Int. Work. Rapid Contin. Softw. Eng. - RCoSE 2014*, pp. 1–9, 2014.

[53] C. R. Jakobsen and J. Sutherland, “Mature Scrum at Systematic,” *Methods & Tools*, pp. 1–11, 2009.

[54] M. Sumrell, “From waterfall to agile - How does a QA team transition?,” *Proc. - Agil. 2007*, pp. 291–294, 2007.

[55] N. B. Moe and T. Dingsøyr, “Scrum and Team Effectiveness : Theory and Practice,” *Xp 2008*, *Lnbip 9*, no. 7465, pp. 11–20, 2008.

[56] E. Whitworth, “Experience Report: The Social Nature of Agile Teams,” *Agil. 2008*, pp. 429–435, 2008.

[57] J. Li, N. Moe, and T. Dybå, “Transition from a plan-driven process to Scrum: a longitudinal case study on software quality,” *Proc. 2010 ACM-IEEE Int. Simp. Empir. Softw. Eng. Meas. - ESEM’ 10*, pp. 1–10, 2010.

[58] D. Ståhl and J. Bosch, “Modeling continuous integration practice differences in industry software development,” *J. Syst. Softw.*, vol. 87, no. 1, pp. 48–59, 2014.

APPENDICES

Appendix – A: Survey Questionnaire

Section 1: Personal/Organizational Information

Organization Name:

Job title:

Experience (in years):

Email:

Organization Size:

- 1-50
- 51-100
- 101-500
- 501-1000
- >1000

Does your organization implement Scrum as a development framework?

- Yes
- No

Section 2: Development Framework

1. Are the activities for managing software quality planned for the project?

- Yes
- No

2. Who defines quality goals for project?
    - Project Manager/ Scrum Master
    - Customer
    - Development team
    - All of them together
3. Who defines quality goals for an iteration/Sprint?
    - Project Manager/ Scrum Master
    - Customer
    - Development team
    - All of them together
4. Testing is performed during development iteration/ Sprint.
    - Agree
    - Disagree
5. Testing during development iteration is performed by:
    - Developer
    - Tester
    - Not applicable

6. Integration testing is performed after each iteration.
    - Agree

- Disagree

7. System testing is performed after completion of project.
    - Agree
    - Disagree
8. During the iteration/Sprint cycle, do you feel pressure to achieve iteration goals within time limit?
    - Yes
    - No
9. Has unit testing ever skipped due to time constraints?
    - Yes
    - No
10. Automated test tools are used for testing.
    - Agree
    - Disagree
11. Is there enough time to perform all possible tests on the product?
    - Yes
    - No
12. Quality of the product is checked by:
    - Customer
    - Quality Manager/Assurer
    - Project Manager/Scrum Master
    - Senior Management

- All of them together

13. Do you follow same quality assurance practices for all projects?

    - Yes
    - No

14. How often does customer report quality defects in the product after delivery?

    - Very often
    - Sometimes
    - Never

15. In your opinion, what is the ideal size of iteration/Sprint Cycle?

    - 2 weeks
    - 3 weeks
    - 4 weeks
    - >1 month
    - Depends on the projects

16. Have you received training for working in Scrum environment?

    - Yes
    - No

17. How do you spent your software development time? (Note: Applicable only to developers)

    - Writing new code
    - Writing new test
    - Debugging/fixing

- Refactoring (optimizing code)
- Not applicable

18. Does your development team practice Continuous integration?
    - Yes
    - No
    - I don’t know
19. Do you perform risk analysis for the project?
    - Yes
    - No
20. Has Scrum development made the entire development team:
    - more responsible for testing
    - less responsible
    - no change in responsibility (team ownership of the test effort)
21. Do you combine Scrum with other methodologies (eXtreme Programming, Waterfall, V-model, Lean, Kanban etc) for better results?
    - Yes
    - No

Please give suggestions to improve the testing process in Scrum.