



## AGILE METHODOLOGIES, THEIR IMPACT ON SOFTWARE DEVELOPMENT AND IMPLEMENTATION: AN EVIDENCE FROM PAKISTAN

\*Faisal Shafique Butt, Zeeshan Ahmed, Rabia Mukhtar, Daud Abdullah, Khalid Ibrahim and Riaz Ahmed  
Department of Computer Science, COMSATS Institute of Information Technology, Wah Cantt, Pakistan

### ABSTRACT

This paper examines that Agile Methodologies such as Scrum, Feature Driven Development, Extreme Programming, Lean Software Development, Hybrid and Adaptive Software Development have an impact on software quality, team productivity and cost of software development in Pakistan. In this study, questionnaire was developed and distributed to different organizations working in Pakistan in order to know their familiarity and experience with Agile Methodologies and the reason behind adoption or rejection. Data analysis was done through Microsoft Excel. The results were astonishing and all the methodologies have positive impact on productivity of employees, software quality and cost of development in software organizations of Pakistan.

**Keywords:** Agile, agile methodologies, scrum, extreme programming, hybrid, feature driven development, adaptive software development, lean software development, crystal methodologies.

### INTRODUCTION

The professional aim of each software developer and a software company is to bring the maximum likely outcomes to their customers with-in time and budget. There are many reasons due to which our project fails like inflation of the process, though well intentioned, is guilty for at least some of this failure. As the company or developers can only survive if their products or projects are successful otherwise the company will bear a big loss of revenue and also a loss of customers.

The competition in software industry is on peak now a days and everyone is trying their best to reach the market before its competitor and make a good place for its better future and survival in the market. But there are many hurdles as the successful software development is not an easy task.

As the fundamental intention of software engineering is to give the cost effective and proficient solution within the diminutive time, which has been discussed in software engineering communities for many years. Many solutions for enhancement have been recommended, from the consistency and the measurement of the software procedure to a mass of concrete techniques, tools and practices.

Recently, experienced software practitioners have

suggested several remedies for enhancement and they labeled their methods as “Agile Software Development”. Agile methods were introduced as a way to help developers and focus on simple techniques to achieve their goals efficiently (Leffingwell, 2010).

Agile software development (ASD) isn't a set of activities, tools or practices but a philosophy introduced in 1990's and put to paper in 2001 with an initial 17 signatories.

There are many agile process models like Extreme Programming (XP), Adaptive Software Development (ADP), Crystal Model, Lean Software Development, Feature-Driven Development (FDD) and SCRUM (Dingsøyr *et al.*, 2012). It was a significant departure from the heavyweight document-driven software development practices such as waterfall—in general use at the time. ASD is followed by number of software houses in all over the world. As this is quite a new approach and not applicable for all projects, especially in the case of Pakistan where the shortages of knowledge and the complexity of software are the major blockages for the implementation of agile practices (Ali, 2012). So in this study, we conducted a survey based research to create awareness among Pakistan's software houses about the ASD benefits, requirements, and its challenges during implementation. The overall goal of this paper is to improve the understanding of the agile software development in the Pakistan so that they can get ASD advantages and better outcomes.

\*Corresponding author e-mail: faisalshafique@yahoo.com

The management of the development of software is major issue in software organizations, where the other factors are volatile requirements, low budget and short time to market. To manage development, it is important to note down the major issues which have large impact on time or budget and try to cope with them as early as possible. There were number of development approaches proposed in the last couple of decades of which only few have survived and are in use today's software development.

The earlier software development practices like waterfall and others was in use of every organizations they were free and organizations try to get benefit from the latest technologies or practices because those old methodologies were not effective in some scenarios. They learned different tips and techniques to successfully manage a software development and they also can predict the cost of software development by their experience. The most common methodology used before agile is waterfall that was in use for decades and still some organization using the same waterfall model for managing the development of software. A waterfall is a process where water flows in the form of vertical drop in the course of a stream or river. There are 5 stages involved in waterfall model (Larman and Basili, 2003) which executes in describe a serial method shown in figure 1.

Figures show the life cycle of water fall in which the output of one stage will become the input of next stage mean we can't jump to the next stage until we complete the precious one.

### Agile methodologies

In 2001, the "agile manifesto" was written by the software practitioners who anticipate many of the agile development methods (Fowler and Highsmith, 2001). In the proposed manifesto, software practitioners recommended that agile development should focus on the four core values:

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

Agile methods are a subset of evolutionary and iterative methods (Larman, 2004) and are based on iterative enhancement and opportunistic development processes. The advantage of this iterative development is that final product needs less changes and it saves both the time to market and the cost. Figure 2 illustrates a generic agile development process.

Agile Development has many agile process models like Extreme Programming (XP), adaptive software development (ADP), Crystal Model, Lean Software development, feature-driven development (FDD) and

SCRUM that are illustrated in figure 3. Each model has its own advantages and place of use so we will discuss each of them separately.

### Extreme programming (xp, xp2)

Extreme programming focuses on twelve best practices of the development which are "the planning game, small releases, metaphor, simple design, testing, refactoring, pair programming, collective ownership, continuous integration, 40-h week, on-site customers, and coding standards". The revised extreme programming consist these best practices: "sit together, whole team, informative workspace, energized work, pair programming, stories, weekly cycle, quarterly cycle, slack, 10-minute build, continuous integration, test-first programming, and incremental design" (Rao *et al.*, 2011). Figure 3 shows simplified version of XP.

### Adaptive software development

Adaptive software development centers mostly on the troubles in developing difficult, huge systems. The method strongly encourages incremental, iterative development, with constant prototyping (Abrahamsson *et al.*, 2010). An Adaptive Software Development (ASD) process carried out in three-phase cycle which is Speculate, Collaborate, and Lean. ASD is explicitly component-oriented. In practice, this means that the focal point is more on outcomes and their superiority rather than the activities or the procedure used for producing the outcomes.

### Crystal methodology

A family of methods is available in crystal family of methodology for project of different size and different technicality. Each method is marker with a color like Clear, Yellow, Orange, Red, Blue which indicating the heaviness of the methodology on the basis of darkness of the color (Abrahamsson *et al.*, 2003). The most agile process are crystal clear (Cockburn, 2004), centers on communication in tiny teams developing small scale software which is not decisive. Crystal Clear has seven characteristics: "frequent delivery, reflective improvement, osmotic communication, and personal safety, and focus, easy access to expert users, and requirements for the technical environment".

### Lean software development

Lean software development model was introduced in 2002, a new approach to perform modeling activities. It consists of seven principles: "eliminate waste, amplify learning, decide as late as possible, deliver as fast as possible, empower the team, and build integrity" (Poppendieck and Poppendieck, 2003).

### Feature-driven development

The Feature Driven Development (FDD) is an agile approach which only covers the design and development

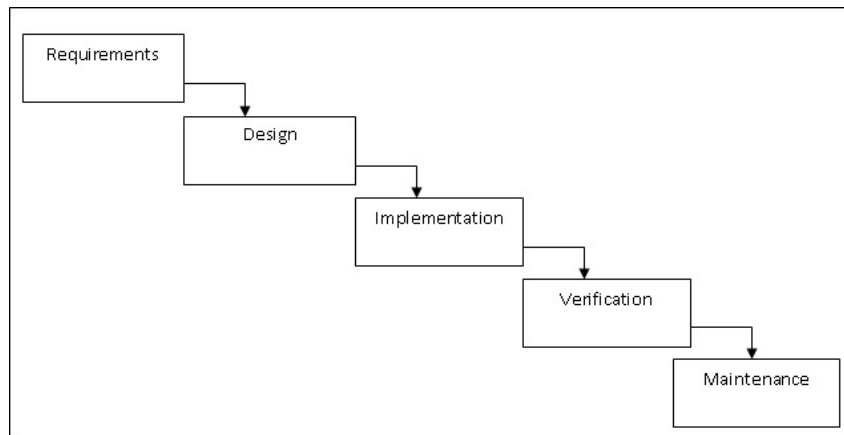


Fig. 1. Stages of waterfall model.

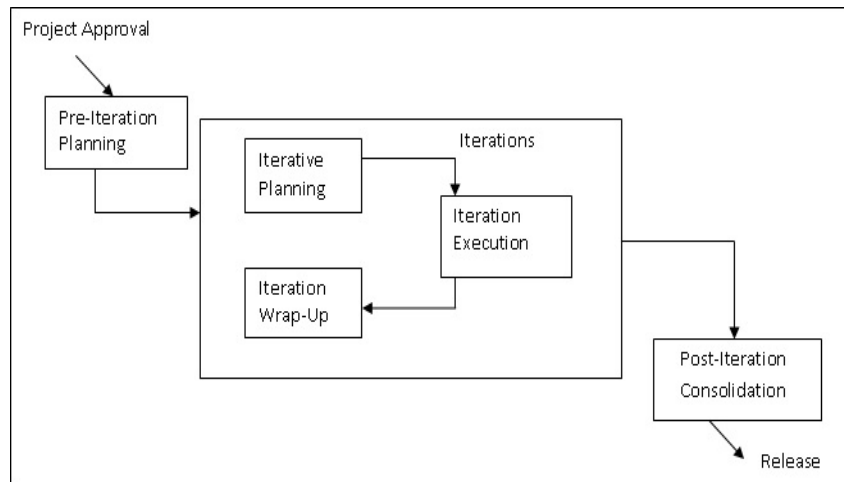


Fig. 2. General agile development processes.

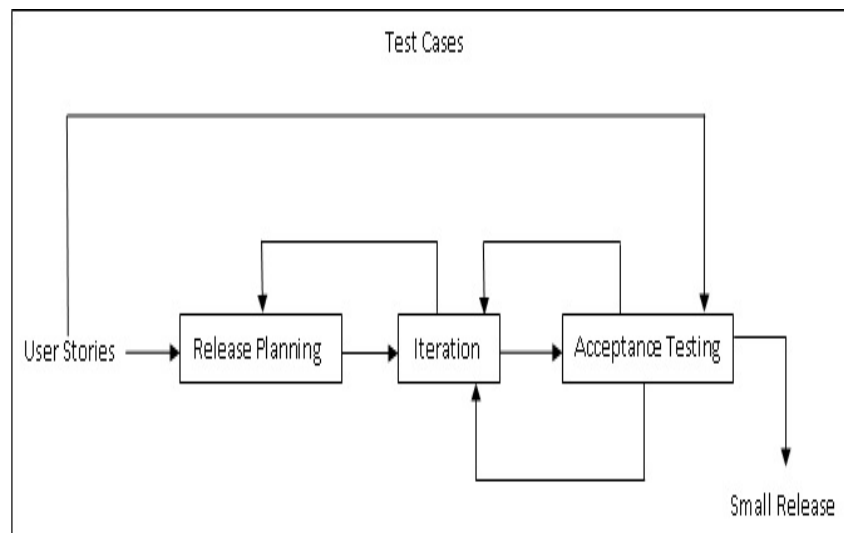


Fig. 3. XP process.

process. FDD is best suitable for the critical system rather than other agile approaches which include the methods, techniques, artifacts and goals needed in the project.

As figure 4 shows FDD consists of five sequential processes in which the design and development of the system carried out (Felsing and Palmer, 2002). When development begins, the domain experts are previously

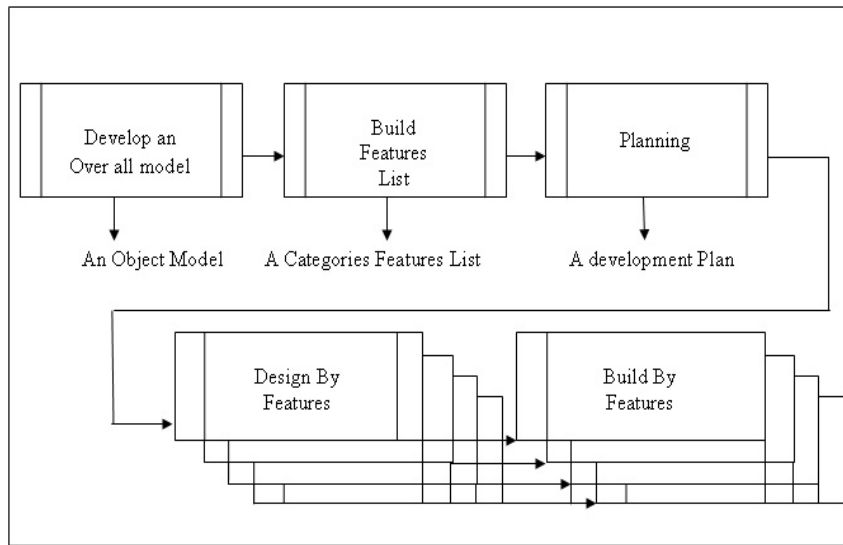


Fig. 4. XP Feature-driven development.

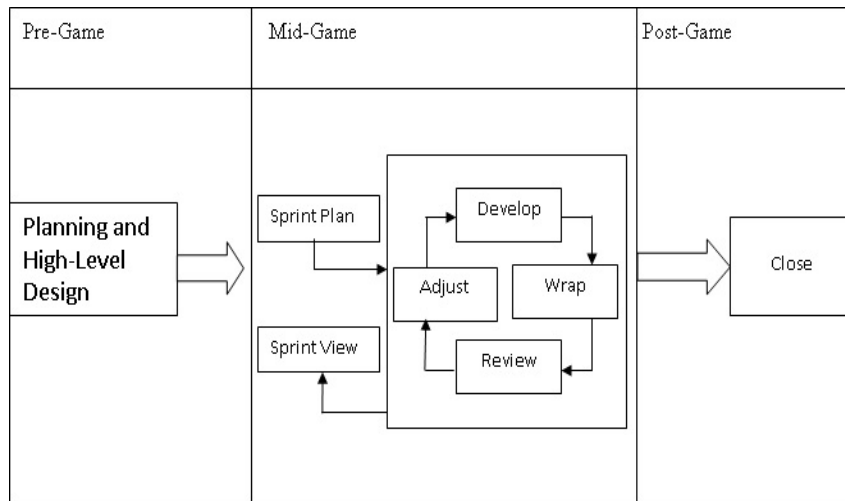


Fig. 5. Scrum Process.

conscious of the scope of system, context and initial requirements. So a detailed featured list is developed at the start and a plan is made according to the priority of features. Then the team starts selecting the features form the requirements or the features list and develop. Every successful sprint will add a new feature to the software as per the featured list. In FDD, domain object modeling and the individual class ownership concept is used (Felsing and Palmer, 2002). Although this approach is relatively new however it is still evolving in the software development.

**Scrum**

Scrum approach is used to manage the system development process. If mainly focus on flexibility, productivity and adaptability. Scrum itself does not define any specific technique for software development. It concentrates on how team member should work to

produce the efficient system in constantly changing environment (Schwaber and Beedle, 2002).

Software is developed by a self-organizing team in increments which is start from the planning phase and ends with a review. As for as the scrum is concerned, it comprises of three phases: “pre-game, development and the post-game.” Pre-game phase further subdivided into two different phase which are planning and high level design. Pre-game phase includes the system definition and the list of all the currently known requirements which will be further updated time to time when the new requirements came through the above mentioned requirements list, effort, time and cost can be estimated for the implementation. When the development starts in development phase (also called the game phase and taken as a black box where the impulsive is expected), system will developed in sprints (Schwaber and Beedle, 2002).

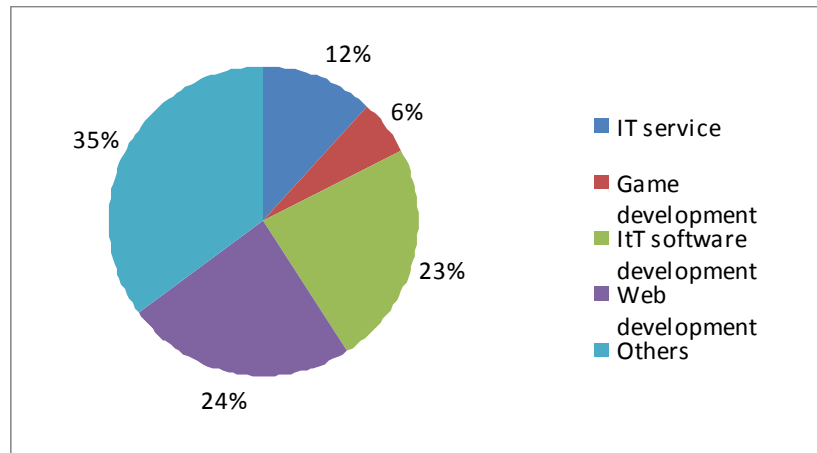


Fig. 6. Application Area.

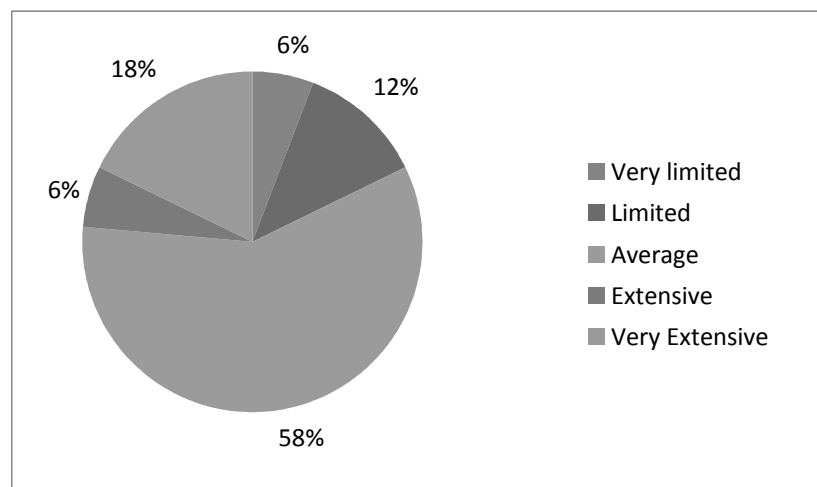


Fig. 7. Agile Methodology.

Sprints are iterative cycle in which the whole functionality will be achieved to develop a new system. Each Sprint includes the traditional SDLC phases which add a new feature to the software development team coordinates their work on daily basis which will be tested or examined by the team of scrum so as to expand their motivation and commitment for the next iteration.

Last phase is post-game phases which will start when all the requirements are completed as per the agreement. In this phase no more features will be added nor invented. This is the final phase in which the system integration will done and after final testing system will be released. Scrum process is shown in the figure 5.

## MATERIALS AND METHODS

### Methodology

The research questions were examined from the different software houses of Islamabad, Pakistan. Questionnaires were used as a secondary source of data collection. As

with the avail of Questionnaires, it is very facile to gain data efficiently for research purpose. In this paper questionnaire were administered personally and distributed among the software houses to collect data. The survey is confined to local area and results were compiled in a short period of time.

### Materials

The MS Word 2010 was employed to develop a survey with thirteen questions which was utilized for the study. The survey was developed online on “KWIK” survey which is a fine & limitless survey tool. A thorough drafted letter sent by e-mail along with a copy of the questionnaire to different participants. Results were compiled by using MS Excel and graphical analysis was completed.

### Sampling size

Sample size tenaciousness denotes culling the number of observations and to utilize a statistical sample. It is a consequential feature of any empirical study in which the

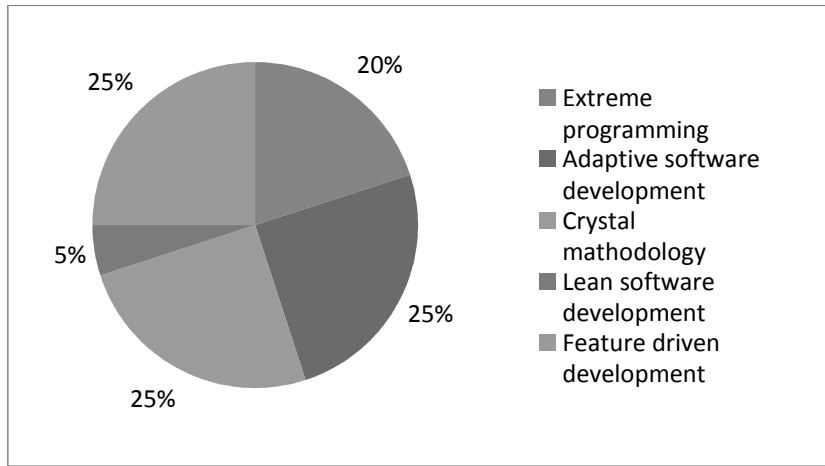


Fig. 8. Agile Process.

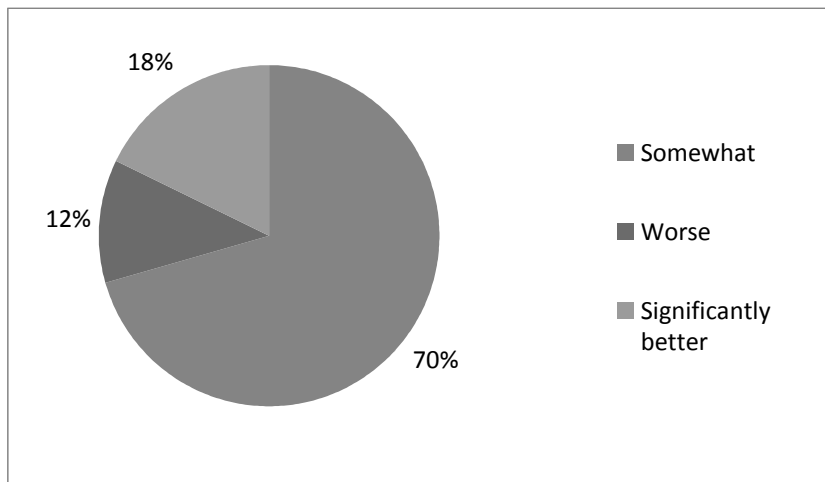


Fig. 9. Team productivity.

goal is to make inferences about a population from a sample. This paper has only one subject of sample i.e. Software houses of Pakistan and the number of companies surveyed is n=17.

**Sampling procedure**

In this research Islamabad, Pakistan was taken as a sample area where questionnaires are administered to make the sample size more appropriate in understanding awareness of agile methodologies and their practical implications. Simple random sampling technique is used in this paper. Precision and confidence are important issues in sampling because when sample data is used to draw inferences about the population, it was expected to be fairly “on target”, and have some idea of the extent of possible error. Because a point estimate provides no measure of possible error, interval estimation is done to ensure a relatively accurate estimation of the population parameter.

**Data analysis**

Seventeen responses were received from the different software houses of Islamabad, Pakistan like Design Webbs, GOL Technologies, Zeeteck, Web-Hive, Knowledge Corporation, 360 Technologies and others. As shown in figure 6, the respondent’s organizations are from different industries i.e. 11.76% from the IT Services and only few are form the game development i.e. 5.88% as shown in figure 6. Most of them are from the IT software development and Web development 23.53% and remaining ones are considered belonging to the companies having hybrid status 35.29% as shown in figure 6. The responses were given from the companies having various sizes. Most responses were from the companies whose size is smaller than 50.

**Understanding of agile methodologies**

As described in figure 7, around 58.82% respondents said that they have average understanding and experience of the agile methodologies. While, 11.76% respondents have

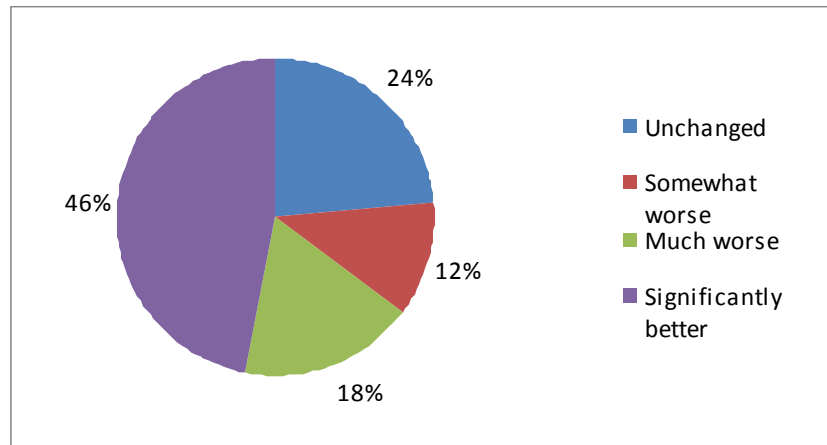


Fig. 10. Quality of applications.

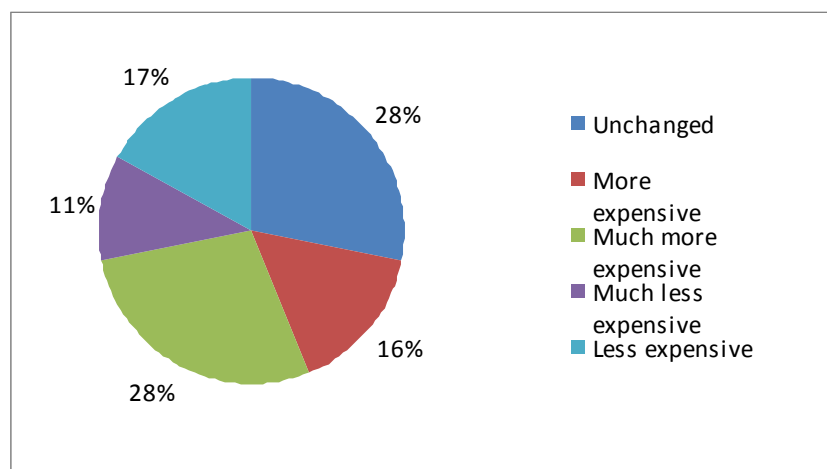


Fig. 11. Development Cost.

limited knowledge of the agile and 5.88% responded that they have very limited understanding of agile methodologies. 5.88% said they have extensive knowledge and 17.65% rate that they have very extensive understanding of agile methodologies. All these facts are clearly demonstrated in the figure 7.

#### Agile processes

The Feature Driven Development Model (FDDM), Crystal methodologies and adaptive software development are the favorite models for software development in Pakistan as 75% people responded that they use these methods in their organizations as shown in Figure 8. And surprisingly only 20% organizations said that they use extreme programming methodology of agile software development. The least practice is in use is lean software development that is only 5% organization using this in their organization.

#### Team productivity

From the respondents of the survey as shown in figure 9, 70.59% believed that adoption of agile process has

somewhat altered their team productivity. While, 11.76% respondents said that their productivity gets worse due to the adoption of agile and 17.65% believed that productivity became significantly better with the adoption of agile. Effect of agile process is demonstrated in the figure 9 to clearly understand the impact.

#### Quality of applications

Around 47.06% participants claimed that agile has a good impact on the overall quality of the applications as shown in figure 10. Other 23.53% said that there is no impact of agile on quality. 11.76% respondents believed that their quality got somewhat worse and other 17.65% felt that their applications got much worse due to adoption of agile methodologies; figure 10 clearly demonstrates the above discussed facts.

#### Effect on development cost

According to the results as shown in figure 11, only 28% respondents feel no change in development cost but others have an effect on the development cost like 16% believed, it is more expensive, 28% felt that it is much more

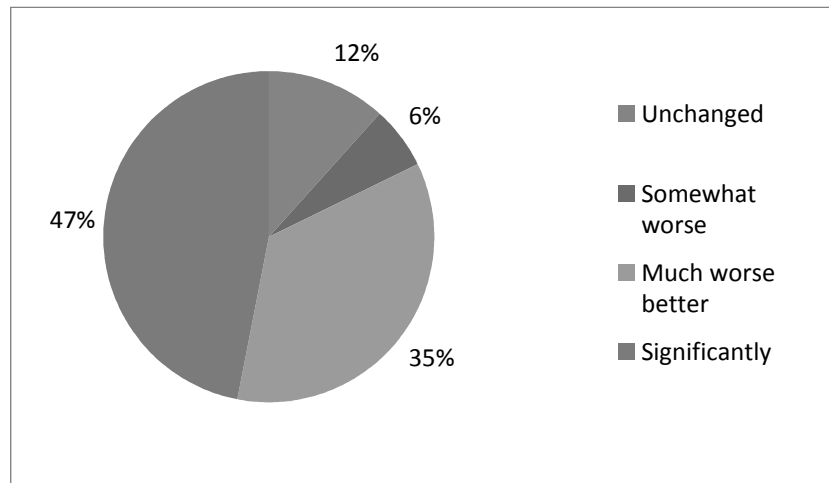


Fig. 12. Level of business satisfaction.

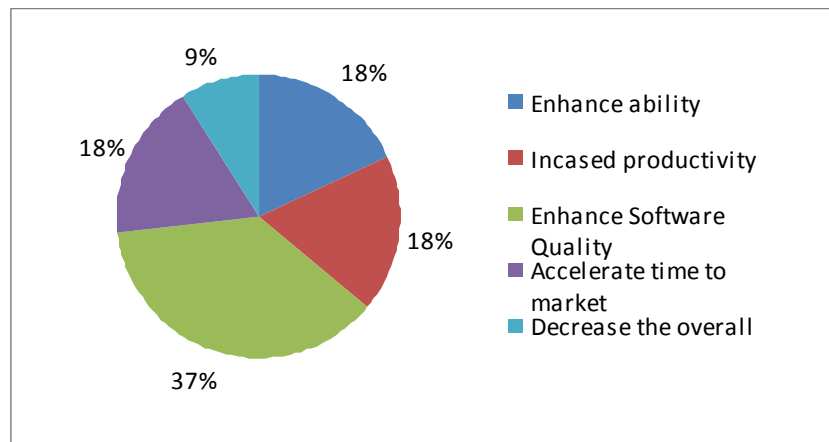


Fig. 13. Features of Agile Process.

expensive, 11% respondents felt that it is much less expensive and 17% participants said that it is less expensive as given in the figure 11.

#### Level of business satisfaction with the software

As expected, most of the people who adopted agile methodologies in their firm said that business satisfaction level of software increase as survey results clearly shows that agile has a good lasting impact on it. 47.1% respond that business satisfaction level significantly better than before. All other statistics of the responders are the demonstrated in figure 12.

#### Feature of agile process

According to the respondents there are different features of agile process as shown in figure 13, 18% respondents believed that it enhances the ability, 18% respondents said that it increases productivity, 37% said that it enhances the software quality, 18% feels that it accelerates time to market and only 9% are using it because it decreases the overall development cost as shown in the figure 13.

#### Intended to use agile next year

64.7% respondents are willing to use or continue the agile methodologies as shown in figure 10, in the next year as they are satisfied and happy with it and 35.3% said they don't want to use agile next year. Figure 14 below clearly demonstrating the percentage.

#### Size of application project

Only 17.65% respondent believes that agile is applicable to all projects. 29.41% respondents believe that agile is good for only some projects and the same percentage is for the respondents who said agile is applicable for most of the projects. While, 11.76% of all respondents said agile is not applicable for any project. All these percentages are demonstrated in the figure 15.

#### Reasons for using agile

According to the respondents reasons to use agile are different. 27% respondents believes that it enhances the ability to manage changing priorities, 19% respondents said the reason is increased productivity, 27% are giving



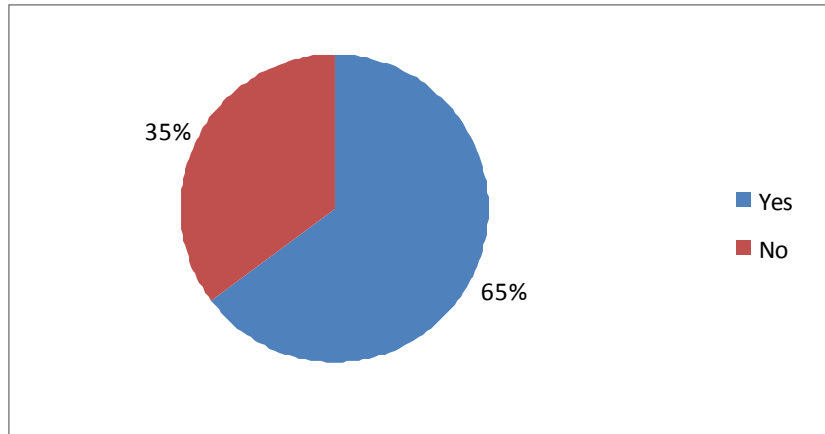


Fig. 14. Adopt agile processes in the next year.

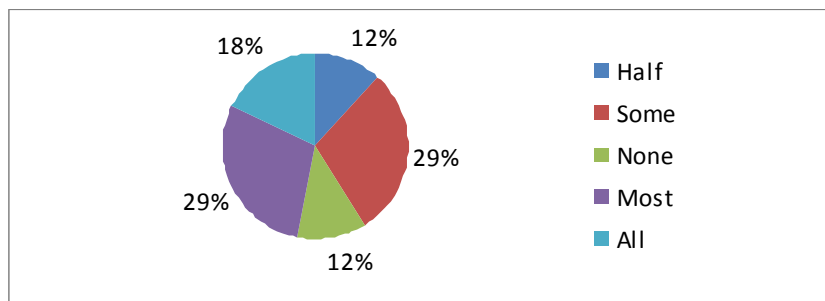


Fig. 15. Proportion of projects.

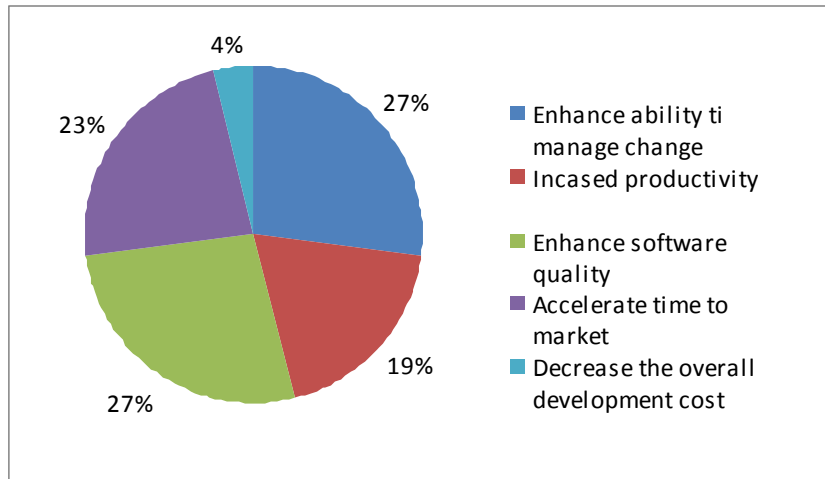


Fig. 16. Reasons for using agile.

the reason of using agile that it enhances the software quality, 23% are using agile to accelerate time to market and only 4% use it to decrease the overall development cost as shown in the figure 16.

#### Main reason for not using agile

When respondents were asked that why they do not use agile methodologies in their firms, the result was really the same as expected because 35.29% respondents said

that lack of experience is the major hurdle for not adoption the agile methodologies and this is the main issue because of which Pakistani companies are not adopting the agile. There are certain other reasons stated i.e. customer collaboration, size of complexity of the project and management support etc. All these reason with the percentage of the responses are demonstrated in the figure 17.

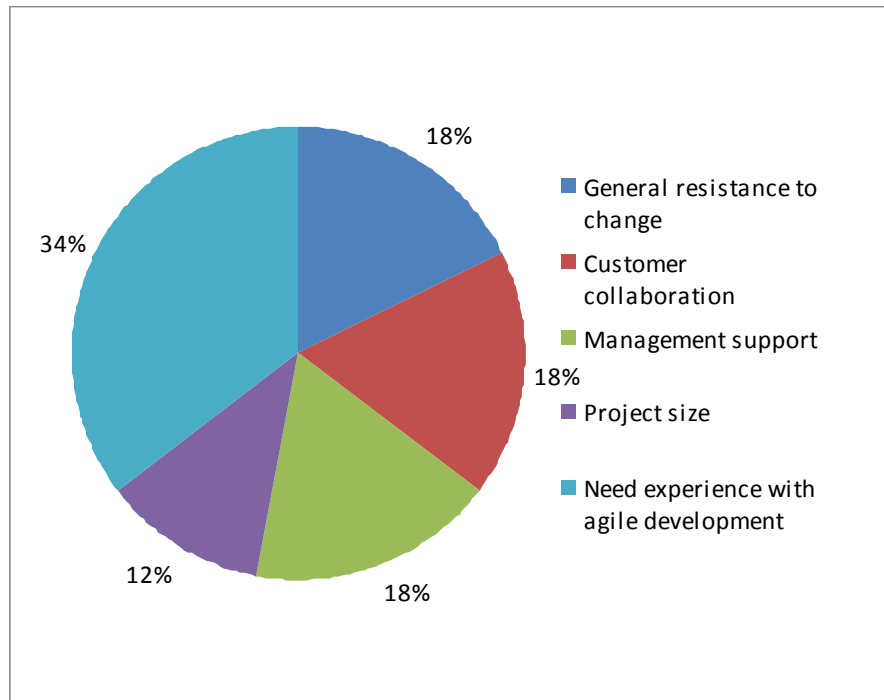


Fig. 17. Non usage of agile methodology.

## CONCLUSION

The objective of the present study was to discuss about agile methodologies, its implementation and acceptance in Pakistan and which features of agile methods altered the team productivity, quality of applications, cost of development and business satisfaction in Pakistani software companies. A survey was conducted to see the different features of Agile, their impact on software quality and productivity of employees and adoption of agile methodologies in Pakistani software houses with a sample size of  $n=17$ . As per the survey results, it is revealed that the main reason for not adopting the agile is the lack of experience and unfamiliarity of the managers and team leads with some designations such as the Agile Coach, Chief Agilest and Scrum Master. Although there are many research papers and articles were published but only few are cognate to the Pakistani software industry therefore there is a strong need of more empirical studies in this domain. It has been seen that agile methodologies has a positive impact on both the software and team which is necessary for the industries of growing economic countries like Pakistan. So the results of this research show that there is vigorous need to give proper training to employees (especially project managers and team leads) of agile methodologies and if agile methodologies are properly applied in Pakistani organizations the quality of software, cost of development and team productivity can be incremented.

## LIMITATIONS

The limitations of this study are that we are only covering organizations in Pakistan. The sample size may be on the lower side, it is possible that if there is large sample size, results would be clearer and specified.

## REFERENCES

- Abrahamsson, P., Oza, N. and Siponen, MT. 2010. Agile Software Development Methods: A Comparative Review1 Agile Software Developmentpp. 31-59: Springer.
- Abrahamsson, P., Warsta, J., Siponen, M. T. and Ronkainen, J. 2003. New directions on agile methods: a comparative analysis.
- Ali, MA. 2012. Survey on the State of Agile Practices Implementation in Pakistan.
- Cockburn, A. 2004. Crystal clear: a human-powered methodology for small teams: Pearson Education.
- Dingsøyr, T., Nerur, S., Balijepally, V. and Moe, NB. 2012. A decade of agile methodologies: Towards explaining agile software development. *Journal of Systems and Software*. 85(6):1213-1221.
- Felsing, JM. and Palmer, SR. 2002. A Practical Guide to Feature-Driven Development. *IEEE Software*. 7:67-72.

---

Fowler, M. and Highsmith, J. 2001. The agile manifesto. *Software Development*. 9(8):28-35.

Larman, C. 2004. *Agile and iterative development: a manager's guide*: Addison-Wesley Professional.

Larman, C. and Basili, VR. 2003. Iterative and incremental development: A brief history. *Computer*. 36(6):47-56.

Leffingwell, D. 2010. *Agile software requirements: lean requirements practices for teams, programs, and the enterprise*: Addison-Wesley Professional.

Poppendieck, M. and Poppendieck, T. 2003. *Lean software development: an agile toolkit*: Addison-Wesley Professional.

Rao, KN., Naidu, GK. and Chakka, P. 2011. A study of the agile software development methods, applicability and implications in industry. *International Journal of Software Engineering and its applications*. 5(2):35-45.

Schwaber, K. and Beedle, M. 2002. *Scrum: The Art of Doing More with Less*. Addison-Wesley Professional.

Received: April 1, 2015; Revised: May 11, 2015;  
Accepted: June 24, 2015