

St. Cloud State University theRepository at St. Cloud State

Culminating Projects in Mechanical and
Manufacturing Engineering

Department of Mechanical and Manufacturing
Engineering

10-2015

Implementing an Iterative Approach to Develop a Windows Application

Shadab Rahil Mohammed

Follow this and additional works at: https://repository.stcloudstate.edu/mme_etds

Recommended Citation

Mohammed, Shadab Rahil, "Implementing an Iterative Approach to Develop a Windows Application" (2015). *Culminating Projects in Mechanical and Manufacturing Engineering*. 1.
https://repository.stcloudstate.edu/mme_etds/1

This Starred Paper is brought to you for free and open access by the Department of Mechanical and Manufacturing Engineering at theRepository at St. Cloud State. It has been accepted for inclusion in Culminating Projects in Mechanical and Manufacturing Engineering by an authorized administrator of theRepository at St. Cloud State. For more information, please contact rswexelbaum@stcloudstate.edu.

**Implementing an Iterative Approach to Develop a
Windows Application**

by

Mohammed, Shadab Rahil

A Starred Paper

Submitted to the Graduate Faculty of

St. Cloud State University

in Partial Fulfillment of the Requirements

for the Degree

Master of Engineering Management

October, 2015

Committee:

Ben Baliga, Chairperson

Gary Nierengarten

Balasubramanian Kasi

ACKNOWLEDGEMENTS

I would like to take this opportunity to express my profound gratitude and appreciation to my advisor Dr. Ben Baliga (Director of Engineering Management) for his support and guidance throughout my study here at St. Cloud State University. Despite his busy schedule, he was very understanding and supportive and provided me with different insights and guidance throughout my learning.

I am equally grateful to my professor Dr. Hiral Shah (Professor of Engineering Management), Prof. Gary Nierengarten (Assistant Professor of Engineering Management) and Dr. Balsy Kasi (Professor of Engineering Management) who provided me with different ideas and suggestion to complete my research and also throughout my learning at St. Cloud State University. I also cannot forget my friends and colleagues who has been integral part of my learning.

Last but not the least, I would like to thank my parents for their guidance and blessing throughout my education. The direction and inspiration that they have provided will carry me a long way in the journey of life that I am about to embark.

ABSTRACT

Organizations that are creating programming arrangement are confronted with the troublesome decision of picking the right programming advancement life cycle (SDLC). The waterfall model is a successive outline process, frequently utilized as a part of programming advancement forms, in which advance is seen as streaming relentlessly downwards (like a waterfall) through the stages. It would not coincide with the changes required in software development process. Whereas, Agile is simple, flexible and mostly customer oriented and yield better results compared to waterfall model. The main intention of the project was to implement a more iterative approach that was more flexible and suitable for the changing environment and find the see the effectiveness, compatibility and also find the customer satisfaction level after the project. As an outcome the organization experienced flexibility in the development, great customer satisfaction and more improved product overall.

Table of Contents

	Page
List of Tables	6
List of Figures	7
Chapter	
I. Introduction	8
Introduction	8
Problem Statement	9
Project Objective	10
Nature and Significance of the Problem	10
Questions to be Answered	11
Limitations of the Project	11
Definition of Terms	11
Summary	14
II. Background and Review of Literature	15
Introduction	15
Background Related to the Problem	15
Literature Related to the Problem	16
Literature Related to the Methodology	20
Summary	25
III. Methodology	26
Introduction	26

Chapter	Page
Design of the Study	26
Collection of the Data for Analysis	30
Cost of Project	31
Project Timeline	32
Summary	32
IV. Data Analysis	33
Introduction	33
Data Management	33
Data Analysis	36
Summary	48
V. Results, Conclusion, and Recommendations	49
Introduction	49
Results	49
Project Questions	49
Conclusion	54
Recommendations	55
References	57

List of Tables

Table	Page
1. Data for Waterfall Model	35
2. Data Related to the Agile Methodology	35

List of Figures

Figures	Page
1. Waterfall Approach	17
2. Agile Methodology	21
3. Scrum Process in Software Development	22
4. Bug Rate during the Sprint	28
5. Burn Down Chart Displayed in Conference Room	29
6. Chart Showing the Velocity of the Tasks in PMDC	36
7. The Chart Showing the Velocity of the Tasks in PremierOne	37
8. The Chart Showing Success Rate of the Tasks in PMDC	38
9. The Chart Showing Success Rate of the Tasks in PremierOne	39
10. Chart Showing the Difference in Estimated and Implemented Factors in PMDC	40
11. Chart Showing the Difference in Estimated and Implemented Factors in PremierOne	41
12. Comparing the Facts for PMDC and PremierOne	41
13. Chart after Data Analysis on Feedback of Both Approaches	45
14. Pie Chart Comparing Two Methodologies	46

Chapter I: Introduction

Introduction

Software development today requires lots of flexibility in the course of the project timeline. That is mainly because of more rapid growth of customer oriented approach. Earlier most of the companies relied on the basic, more orthodox kind of project management. For instance, Waterfall model was the most commonly used methodology for project management. There are so many companies which use waterfall model still as each project has its own structure and waterfall model fits perfectly in order to accomplish the project on time and with desired outcomes. But, what went wrong is the approach to be used by the commercial companies which are solely dependent upon the customer satisfaction. There are several cases where the companies might have lost their customers due to the product not being delivered on time, cost surpassing the expected expenditure till the end of the project, customers being unsatisfied with the final product. Most number of the times customers are not sure of what they really need.

This lead to the implementation of a new approach towards how a project should be handled in order to deliver the product on time, within the budget, leaving the customers satisfied and make them come back. Because of the reason that customers being unclear of what they exactly want at the beginning of the project, the projects should be managed in such a way that they can accommodate change, in other words they should be flexible enough to be changed whenever the customer comes up with a new requirement.

Agile is one of the methodologies where this can be achieved, as it is an iterative approach unlike the waterfall model. This approach is best suited for the projects where the customers are unpredictable. Agile is simple, flexible and mostly customer oriented and yield better results compared to waterfall model. Agile has six different methods, out of which Scrum is the most lightweight, simple and more flexible approaches among all the other agile methods.

This capstone project mainly focuses on how Agile methodology yields better results compared to the traditional waterfall model. How do we approach the project using the agile methodology and can we satisfy the customers using this methodology.

Problem Statement

The waterfall model that was used by the company to implement the projects was not best suited to the type of the projects undertaken. This resulted in the loss of market value in the form of unsatisfied customers as it was unable to deliver the product on time and within the budget limit, which is basically very difficult in the traditional method. Keeping all these in focus, agile methodology was proposed to deal with the problems as it is an iterative process and every one will be involved, more importantly the customer. As it has iterations the results will be expected and we can have a track on what we need and where the project is going.

Project Objective

The main objective of the project is to find whether the methodology adopted can overcome the problems face by the traditional method. The objectives can be specified as follows,

- 1) Evaluate how effective is the Agile/Scrum methodology in software development when compared to Waterfall and
- 2) How well it suits the project in keeping up with the customer requirements by delivering the project on time and within the budget limit.

Nature and Significance of the Problem

As the company is concentrating more on customer satisfaction, there is jeopardy in the type of methodology to be selected in order to meet the expectations and requirements of the customers. As the customer changes his mind during the course of the project timeline it will be very difficult to implement those changes if the design and estimation have already been made. This results in the delay of the project completion, and inability to deliver at the specified time. As the customer changes his requirements there will be a difference in the cost too as the it will require more working hours and also there will be change in the resources required for the project completion. Keeping everything in mind it shows us the software development needs a methodology which is iterative and also which involves customers along the timeline of the project in order to get the requirements time to time so as to satisfy the customer by the end of the project, as there would be no expectations in the beginning.

Questions to be Answered

The following are the questions that would be answered by the end of the project,

- 1) Was the Agile/Scrum methodology compatible for the project undertaken?
- 2) How was the Agile/Scrum Methodology more effective than the traditional waterfall model?
- 3) How was the involvement of the customers during the project?
- 4) Did the project end on time and within the budget?
- 5) What is the customer feedback at the end of the project?

Limitations of the Project

The project was proposed to adopt an approach that is more iterative and responsive to the changes required during application development for a specific environment and how well it suits the project in keeping up with the customer requirements. This project is not intended to cover the roles of the developed application during the course of the time and may not result into similar results if used in different environment.

Definition of Terms

- SDLC: SDLC (Systems Development Life-Cycle) is utilized as a part of data frameworks, systems engineering, and programming designing as a methodology of making new or changing existing frameworks. The SDLC can be considered an idea that lies underneath various programming advancement strategies right now utilized throughout the industries. SDLC

give a schema to make, arrange and control any information system to be produced.

- **Waterfall:** Waterfall model is the oldest SDLC methodology to be introduced. It is also called as the 'Traditional Approach'. It is a linear development procedure, in which the sequence is seen flowing from top to the bottom. It is a top-down approach having all the phases one below the other Requirement Analysis, System Design, Implementation, Testing, Deployment and Maintenance (Massey & Satao, 2012).
- **Agile Methodology:** Agile is one of the SDLC methodologies. It is an iterative process unlike the other traditional approaches. It is a methodology that is very iterative and flexible to the changing customer needs and demands. The key solutions are extracted through the collaboration between self-organizing cross-functional teams through constant communication. It divides the project in to number of incremental builds.
- **Scrum:** Scrum methodology is one of the most widely used software development methodology today. It is an iterative and incremental software development structure for managing projects. It consists of product backlog, teams, sprints, burn down charts and more.
- **Product backlog:** The product backlog has the list of features in a prioritized order that the customer wants in a product. It also contains short description about the desired product.

- **Sprint backlog:** The sprint backlog includes the list of tasks and activities that the development team believes that they could complete during a specified sprint. It is very important to list out the tasks that the Scrum team needs to complete during the specific sprint so that they can produce a deliverable within an agreed time. Unlike product backlog, sprint backlog have specified time.
- **Scrum Roles:** Scrum roles can be differentiated according to the amount of commitment they have towards the project. They are Pig Roles and Chicken Roles. Pig roles are those who are totally committed towards the project and they will be the Scrum Master, Product Owner and Delivery team. Chicken roles are the people who are less committed and whose views are taken into consideration in delivering a project.
- **Scrum master:** Scrum master is the one who is responsible for supporting and enforcing the agile values and practices throughout the team. He is often referred as a business leader.
- **Product Owner:** Product owner is the one who represents the customer and he maximizes value of the work done. The product owner determines the requirements of the customers and builds up the product backlog prioritizing the user stories. The Product owner is responsible for bringing a clear picture of what a customer needs, failure is doing so will affect the team as they will not be able to deliver according to the customer needs

- **Sprint:** Sprint is one of the most important phases of software development where the specified time is set for specified work. It usually consists of 2-4 weeks.
- **Sprint Retrospective:** Sprint retrospective is the meeting that takes place at the end of the sprint to review the features worked on by the team in that sprint. In the review the customers, product owner, scrum master and delivery team are present. This gives a chance to the customer to have a look at the project and propose if any changes are needed.

Summary

This chapter briefly covers many aspects of this project prominently to determine the actual problem that exist and how it affects in real time, main motive of the project, list of questions that are going to be answered at the end of the study, basic limitations of the project and some of the key terms that are important to understand the research are also explained in this chapter. The next chapter briefs about the literature background knowledge associated with this project.

Chapter II: Background and Review of Literature

Introduction

This chapter briefly describes the background details and literature about the problem. It also focuses on the methodology that is implemented in this project and explains the type of methodology that is used with the help of graphical representation.

Background Related to the Problem

This project is conducted in an IT firm which is developing a windows application for the customers. It will provide a UI to enter and perform different tasks when they receive a call regarding any incident. This company has more than 50 customers across the United States. The product is designed in order to connect to the State database and work along with the database. It also designs the product for the mobile users, who patrol and cannot connect to the network onsite. This application also has GPS networking to keep track on the mobile users and find out there where about. This project was conducted during the development of the application which is used on site. As the company was using the SDLC approach which makes the developing team isolated from the customers. This isolation makes it difficult to interact with them and often results in customer dissatisfaction as they find needs and modules to be included after the designing phase. For this problem to be resolved, the firm needed a new approach that was more iterative and responsive to the changing needs.

Literature Related to the Problem

SDLC (Systems Development Life-Cycle) is utilized as a part of data frameworks, systems engineering, and programming designing as a methodology of making new or changing existing frameworks. The SDLC can be considered an idea that lies underneath various programming advancement strategies right now utilized throughout the industries. SDLC give a schema to make, arrange and control any information system to be produced.

SDLC portrays a methodology utilized by designers and investigators to make and convey all parts of a data framework. These incorporate characterizing necessities, approval, preparing, and underscoring responsibility for framework. At whatever point SDLC is utilized, the objective is to make a framework that meets the essential stakeholder (or owner's) desires. These incorporate however are not constrained to conveying the task inside time and expense demands. Components considered are deployment, convenience, and minimization of flaws when uniting with legacy framework parts likely made by diverse programming sellers. To help deal with the inborn intricacy when outlining Enterprise programming, there have been a few SDLC based models made, for example, the Waterfall, Spiral, and Agile methodologies (SDLC, 2014). The most common and which is also the methodology used by this firm is the basic Waterfall model. Let us discuss in brief about the Waterfall model in the following part of this discussion.

Waterfall. Waterfall model is the oldest SDLC methodology to be introduced. It is also called as the 'Traditional Approach'. The Waterfall approach to systems

analysis and design was the first established modern approach to building a system. This method was originally defined by Winston W. Royce in 1970. It is a linear development procedure, in which the sequence is seen flowing from top to the bottom. It is a top-down approach having all the phases one below the other Requirement Analysis, System Design, Implementation, Testing, Deployment and Maintenance (Massey & Satao, 2012). These steps are described briefly below and also shown in the following figure.

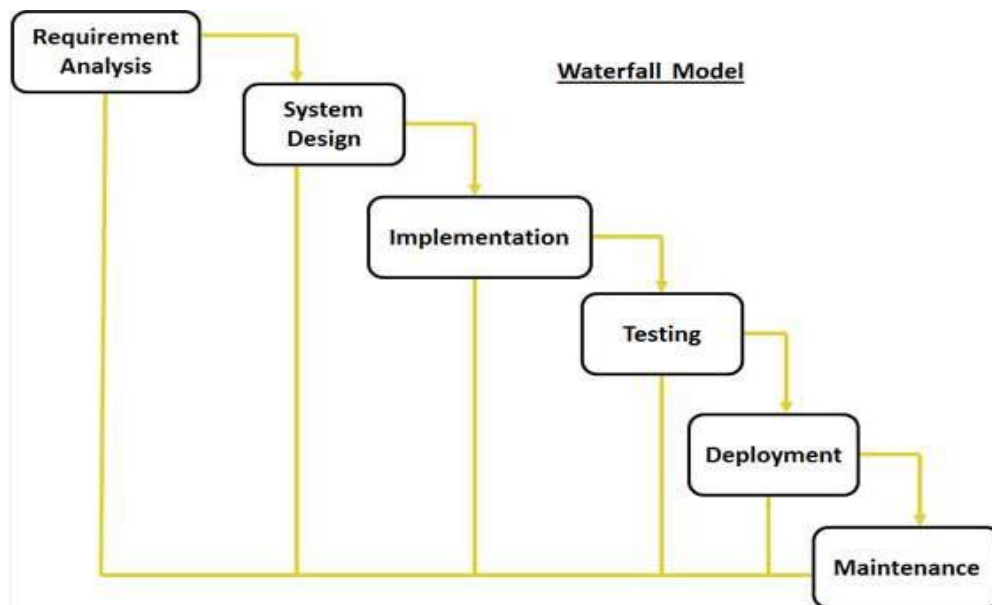


Figure 1: Waterfall Approach

Requirement analysis. Requirement analysis is the most vital and essential stage in SDLC. It is performed by the senior individuals from the group with inputs from the client, the business division, business reviews and area specialists in the business. This data is then used to plan the fundamental task approach and to lead

product feasibility study in the economical, operational, and technical areas (Software Development, 2007).

Making arrangements for the quality confirmation prerequisites and recognizable proof of the dangers connected with the venture is likewise done in this stage. The result of the specialized possibility study is to characterize the different specialized methodologies that can be taken to implement the project with lesser risks (Software Development, 2007).

System design. The requirement analysis from the first stage is mulled over in this stage and framework outline is readied. System design helps in determining equipment and framework necessities furthermore helps in characterizing general framework structural planning. System design is classified in to two design stages, the logical design and physical design. In the logical design they analyze all the requirements information collected and then using them to implement he design physically in next step (Software Development, 2007).

Implementation. Implementation can also be called as development stage where all the developing work is done. In this software developing field the developers write the code to build a working application depending on the requirement analysis and design specified in implementation stage (Software Development, 2007).

Testing. In this phase the developed product is tested by the testing team to verify whether the product is behaving in the way the customer required. This is the only stage where the product is tested for the defects, and if the defects exist, they

are reported to get them fixed and retest to ensure that the product reaches the quality standards (Software Development, 2007).

Deployment. In this phase the product is actually deployed or delivered to the customer. Usually the deployment takes place in phases where the product is first tested in the business environment then depending on the feedback it is rolled over totally with the suggested improvements if there are any (Software Development, 2007).

Maintenance. During this phase when the customer is using the product, the support is extended by the firm to ensure the proper functionality of the product. If the application do not behave in the intended manner then the issue is reported to the firm and the issue will be resolved by implementing the changes requested (Hughey, 2009).

This methodology has been criticized for being unresponsive and unable to adopt to the changes in the projects. It is also said that as it leaves no room for improvement, if the customer comes up with any of the changes then it will create a lot of confusion and have to be redesigned according to the changes specified. The only interaction with the customer will be at the starting phase of requirement analysis where they specify their needs. The customers might come up with other requirements in the middle of the project and it will be difficult to implement the desired changes (Cairns, 2012).

Literature Related to the Methodology

Today's industry is growing rapidly with innovative ideas and that can be achieved by allowing the changes and being responsive to the requirements whenever required. For this reason most of the firms now-a-days are shifting towards the Agile methodology which is more responsive and iterative approach for software development where innovative ideas keep flowing regardless of the phase of the project. The main objective is more customer interaction, responsiveness and flexibility in changing the product in the course of the project timeline.

Agile. Agile is one of the SDLC methodologies. It is an iterative process unlike the other traditional approaches. It divides the project in to number of incremental builds. These are obtained in iterations. Each iteration has teams working on different aspects like planning, design, analysis, coding, testing and deployment simultaneously. At the end of each iteration a working product is displayed to the customer. Agile is flexible, it handles each project differently to best suit the requirements. At the end of all the iterations a working project with all the features desired by the customer is delivered (Software Development, 2007).

The definition of Agile can be explained by the Agile manifesto brought up in a meeting between number of Software luminaries in 2001. The agile manifesto is:

Individuals and Iterations over processes and tools

Working Software over comprehensive documentation

Customer Collaboration over contract negotiation

Responding to Change over following a plan (Blankenship, Bussa, & Millett).

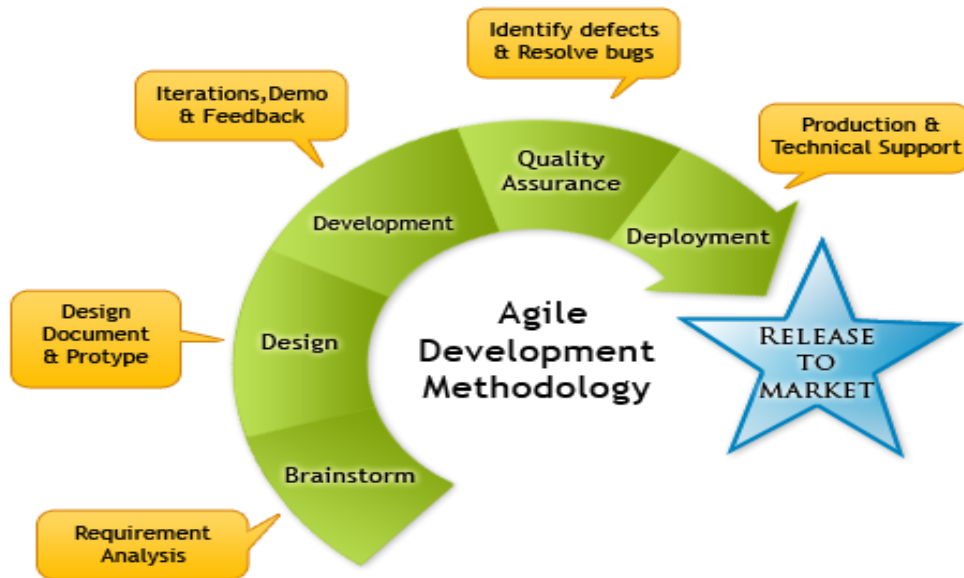


Figure 2: Agile Methodology (Agile software, n.d.)

The above figure describes how the iterations take place and after each development phase before the product is released into the market.

Agile has different methods, one among them is Scrum. It is an iterative approach which is mainly dependent upon agile principles and manifesto. It is divided into iterations which are mainly divided based on time and they are called sprints. Each sprint has a particular objective to be achieved at the end of the sprint. In the each sprint a feature is selected to work on and it is delivered at the end of that particular sprint. It is managed by Scrum Master who makes sure that there are no barriers in delivering the product on time (Blankenship et al., 2011).

Scrum. The methodology adopted to implement the project is developing an application using Agile/Scrum. The Scrum framework has three main aspects; they are artifacts, roles and activities.

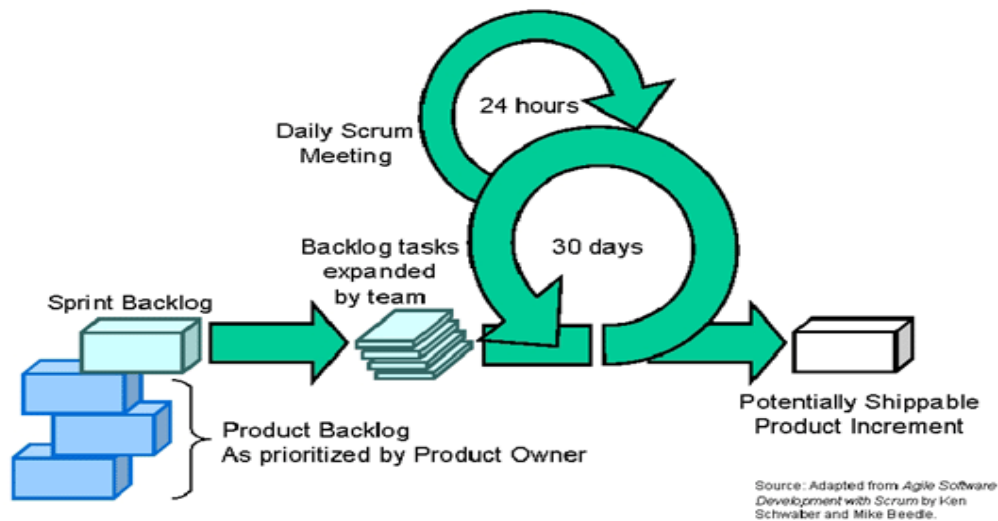


Figure 3: Scrum Process in Software Development

Scrum artifacts. There are three main artifacts in scrum. They are, product backlog, sprint backlog and burn-down chart.

Product backlog is the list of all the requirements mentioned by the customer. Product owner manages the product backlog and he adds or removes user stories from the list. The product backlog is built according to the priorities of customers and product owners. The priorities change as the customers come up with new requirement and the user stories which are most important to the customers are moved to the top in the list.

Sprint backlog is the list of all the items to be completed in that particular sprint. It can be treated as a subset of the product backlog as the product backlog

constitutes of user stories and each sprint works on a particular user story dividing it into small tasks and each task is assigned to the team members. At the end of the sprint the sprint backlog is emptied and the left over tasks are moved back to the product backlog and prioritized accordingly. This supplies data to burn-down chart which displays the actual progress of the sprint. Usually it is displayed in the area where team members and also the customers can view the progress of the sprint (Blankenship, 2011).

Scrum roles. Scrum roles can be differentiated according to the amount of commitment they have towards the project. They are Pig Roles and Chicken Roles. Pig roles are those who are totally committed towards the project and they will be the Scrum Master, Product Owner and Delivery team. Chicken roles are the people who are less committed and whose views are taken into consideration in delivering a project (Blankenship, 2011).

Scrum Master is responsible for making the team understand the process and follow it. A Scrum Master is liable to remove all the barriers which may block the work of the team. The main characteristic of a Scrum Master is to be “servant leader”. A Scum Master should have good communication skills as he have to manage if any issue rises among a team (Blankenship, 2011).

Product owner is the one who represents the customer and he maximizes value of the work done. The product owner determines the requirements of the customers and builds up the product backlog prioritizing the user stories. The Product owner is responsible for bringing a clear picture of what a customer needs, failure is

doing so will affect the team as they will not be able to deliver according to the customer needs (Blankenship et al., 2011).

Delivery team is a group which is responsible to deliver the required product at the end of the each sprint. They are a group of 2-10 and they can be programmers, designers and testers. They work as a team, they self-organize and they are cross-functional as no one will be an expert in everything they tend to transfer knowledge from one another. As the Scrum Master looks at what the team needs and the Product owner communicates with the customer they can solely concentrate on delivering the product (Blankenship et al., 2011).

Scrum activities. Scrum activities mainly focus upon project planning, review and setting up meetings. A planning meeting takes place before the start of each sprint to determine the user stories the team will be working on in that sprint. These features are obtained from the product backlog created by the Product Owner. Then, the opinion is been collected from all the team members to know whether they are satisfied with the proposal and discuss if they do not agree with the proposal (Blankenship et al., 2011).

Standup meetings are held every day to see where each team member stands on his task. The meeting usually takes place for 15 minutes and they discuss what have been done after the previous meeting and what can be done after this meeting before the next meeting. If the team is facing any issue it is discussed in the meeting and the Scrum Master takes proper measure to ensure the free flow of the project (Blankenship et al., 2011).

At the end of the sprint a review is held to present the features worked on by the team in that sprint. In the review the customers, product owner, scrum master and delivery team are present. This gives a chance to the customer to have a look at the project and propose if any changes are needed. This review is to satisfy the customer by early and continuous delivery of the product (Blankenship et al., 2011).

By adopting the above methodology, it will be easier to know the customer's requirement and the team will be able to deliver the product on time. At the end of the project it will be possible to evaluate how Agile/Scrum Methodology is more effective when compared to the traditional model to accomplish the task on time and within the budget limits.

Summary

This chapter discussed the literature related to the problem and the core concepts of the agile methodology. The major concepts required to comprehend the Scrum methodology are explained in detail. The next chapter will discuss the methodology used to research the project.

Chapter III: Methodology

Introduction

This chapter discusses the methodology that has been selected to conduct the project in detail. The data collection process and the framework are also explained. The tools and techniques used in the research as well as the budget and timeline are described in this chapter.

Design of the Study

The main intention of the project was to implement a more iterative and responsive methodology which is flexible and customer oriented. Till now the firm was using the SDLC (Software Development Life Cycle) for developing the application. Now for their new and enhanced product the Agile/Scrum methodology is adapted for better results than the traditional waterfall model. The approach selected was quantitative as it helps in generating a feedback from the customers and the members of the team on the new methodology that was implemented. As this methodology insists for regular meetings and reviews of the work done which included the team members, customers and product owner, the survey was conducted collecting the data that compared both the methodologies which could help in coming to the conclusion on deciding the best suited methodology for these ever changing requirements. The steps involved for implementing this methodology are as follows:

Passing the bureaucratic barrier. It is necessary and obvious that the support is required from the organization to implement something new that is out of the box for that organization till this time. For an idea to be implemented it should pass through all the levels of management before it gets approved. The decision to implement was not under the control of the developer or the team leads. This was the mandatory barrier to pass as any changes or enhancements made to the product require the opinion and approval from the stake holders.

Organizing a team and environment. A team was needed for implementing this methodology that included developers, testers along with the scrum master. A team was formed with seven members overall. Meetings were scheduled every day for the daily scrum and were limited to the essential team members.

Training. As the approach was new to the team the training was required for the team members from those in the team who were already involved in a project that used this methodology. Most of them had been working under the traditional waterfall methodology for years and had no idea of the roles they have to perform in the team. This can be done by the scrum master or any other agile experts that can briefly explain the process of undertaking a project in this methodology.

Bug rates. The bugs are the defects or the small objects that are part of a task. Each task consists of different number of bugs depending on the area of module the task is related to. Example of the chart that shows the bug rates related to a sprint is shown below.

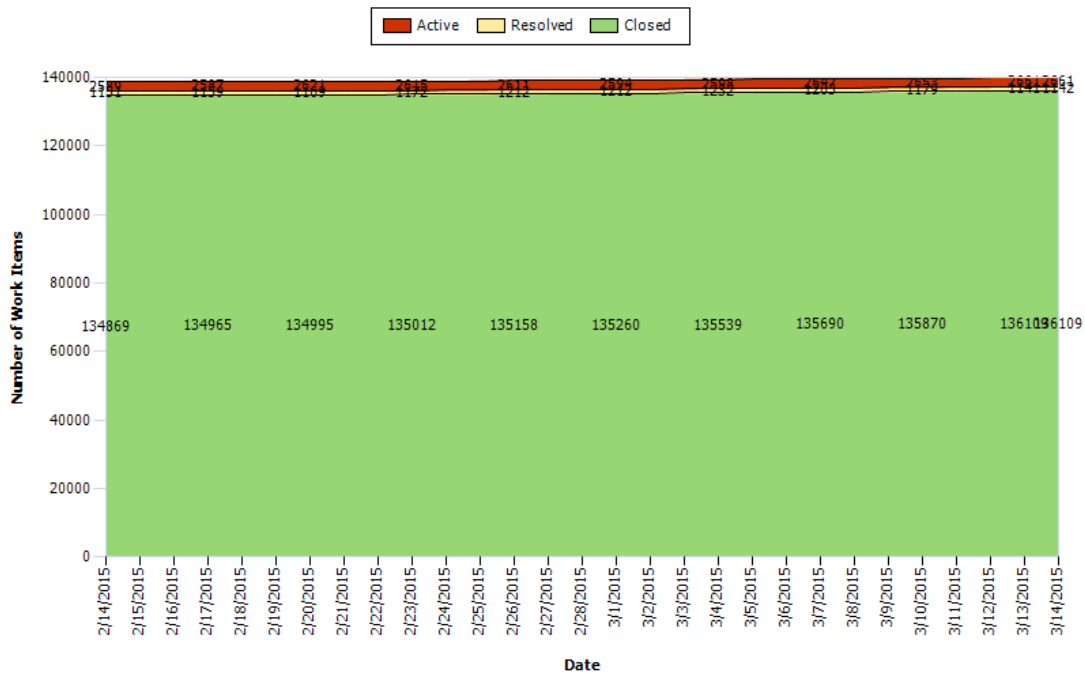


Figure 4: Bug Rate during the Sprint

This report can be generated by TFS (Team Foundation Server) that is been used to keep track of the work items along the sprint that are added in between and whether they are taken care of depending on the priorities. This chart can be used to answer the questions,

- 1) Is the team responding to the new work items?
- 2) The time taken to resolve the bug.
- 3) Number of work items that are needed to be completed before the end of the Sprint.

Burn down chart.



Figure 5: Burn Down Chart Displayed in Conference Room

Charts were displayed in meeting areas and near the conference rooms, so that the team can keep track on the progress of the user stories undertaken, completed and all remaining job that has to be undertaken before the end of the sprint.

Sprint iterations. Like clockwork there were cycles where we chose the sprint that we were going to deal with and if the business condition transformed, we would respond to that change inside the sprint of two weeks and reprioritizes the sprint overabundance. In light of these emphases, the group was not perplexed about the change asks for that would originate from the partners and the partners were extremely thankful as their suppositions were highlighted. Since the client gets to see

the item unfolding as it's being made versus seeing them at the very end, it permitted the item to have the accurate peculiarities that was sought by the client. In view of these emphases, there are no sudden astonishments at the clients end, and also the designers end. These emphases gave the group a capacity to change speedier to the obliged business conditions.

Collection of the Data for Analysis

The data was collected that was related to the tasks that were undertaken in the project that was implemented using Waterfall and also from the project that was implemented using the Agile methodology. This data was then tabulated and was used to generate the graphs and charts that helped in determining the effective methodology for that particular project.

The data was collected by two surveys that were conducted with the developers and the customers who are involved in the both the projects where in one of them they used the waterfall model and in the other agile methodology was used. They were requested to compare and contrast the new methodology with the Software development approach. They were asked for to investigate the new procedure with the Software advancement approach. One extraordinary focal point of scaled inquiries is that the information recovered from the review can be effectively evaluated. These sorts of inquiries permit the subjective information to be exhibited in more factual configuration that is less demanding for the buyer to understand. The information recovered from the review permitted the creator to arrange the respondents into the gatherings in light of the answers that they have chosen. A lot of

information was surveyed in a brief time of time because of the way of the study. Since the organized overview was more shut finished and did not permit the respondent to completely show their assessment, a semi-organized review was additionally led.

The survey helped in gathering the information regarding the satisfaction, compatibility, responsiveness and transparency of the customers towards the developers that helped in getting the product delivered on time and also based on the requirements.

Apart from the survey that compared the two methodologies a survey was also done among the developers in order to get their opinion on the methodology introduced and to get suggestions on any improvements that need to be considered while implementing this methodology.

Cost of the Project

There was no cost involved as the project was conducted using the surveys and questioners posted to the stack holders that include the developers, product owners and the customers.

Project Timeline

Task	Timeline
Literature Review Proposal	November 2014
Requirements Specification	November 2014
Analyzing the Requirements	December 2014
Design and Development	December 2014 – January 2015
Testing the developed code	February 2015
Deployment	February 2015
Review and Customer feedback	March 2015
Final Defense Write-up	March 2015
Final Defense Presentation	April 2015

Summary

This chapter explained the background of the data collection process, how the data was collected and the tools and techniques that were used to analyze the collected data. The next chapter will present the data collected and analysis of the data. It also included the important steps to implement the methodology.

Chapter IV: Data Analysis

Introduction

This chapter will include the representation of the data collected from the surveys done at the firm. It also includes the data analysis using the graphs and other charts that are more suitable to analyze.

Data Management

The data was collected from the project that was undertaken in the waterfall methodology and also from the project that was undertaken in the agile methodology. Both the data are compared mostly related to the tasks and the enhancements that are undertaken in the course of the project. The data is then used to generate the tables and charts that give us the clear picture of the better methodology that needs to be undertaken for the software development.

The data was also collected by conducting the surveys among the team and also the customers at the time of the sprint retrospective. This survey was to compare the old waterfall model to the newly implemented agile methodology. Two columns were provided in the handout with one representing the Waterfall model and the other representing the agile methodology.

The following is the pattern of the survey that includes 5 aspects to be rated from 1 to 10. Where 1 being the least favored and 5 being the most. The 5 aspects on which the survey was done are

- i) Reached expectations
- ii) Communication

- iii) Timely delivery
- iv) Flexibility
- v) Recommendable?

The model of the handout that was distributed among the team and customers for the survey is been shown below.

Waterfall Vs Agile Methodology. Please fill in your feedback on the experience you had with two different methodologies used to develop the product. Please rate from the numbers 1-5, where 1 being the least favorable and 5 being the most.

	<u>Waterfall</u>	<u>Agile</u>
Reached expectations		
Communication		
Timely delivery		
Flexibility		
Recommendable?		

Suggestions

After collecting the data, the values were categorized into the two groups, one for the waterfall and other for the agile methodology. The table below represents the values given for the waterfall model in the survey.

Table 1

Data for Waterfall Model

	Expectations	Communication	Flexibility	Timely Delivery	Recommended?
Participant1	3	1	1	2	2
Participant2	2	2	1	2	1
Participant3	2	1	1	1	3
Participant4	3	1	2	2	1
Participant5	2	2	1	2	1

The table below represents the data obtained during the survey regarding the agile methodology.

Table 2

Data Related to the Agile Methodology

	Expectations	Communication	Flexibility	Timely Delivery	Recommended?
Participant1	5	5	5	4	5
Participant2	5	5	5	3	5
Participant3	4	5	5	4	4
Participant4	5	4	5	4	4
Participant5	5	5	4	4	5

Data Analysis

Efficiency. The data received for the waterfall approach is related to the tasks that were undertaken in each quarter and how many number of tasks were completed in respective quarter that gives us the percentage of work done in that particular quarter.

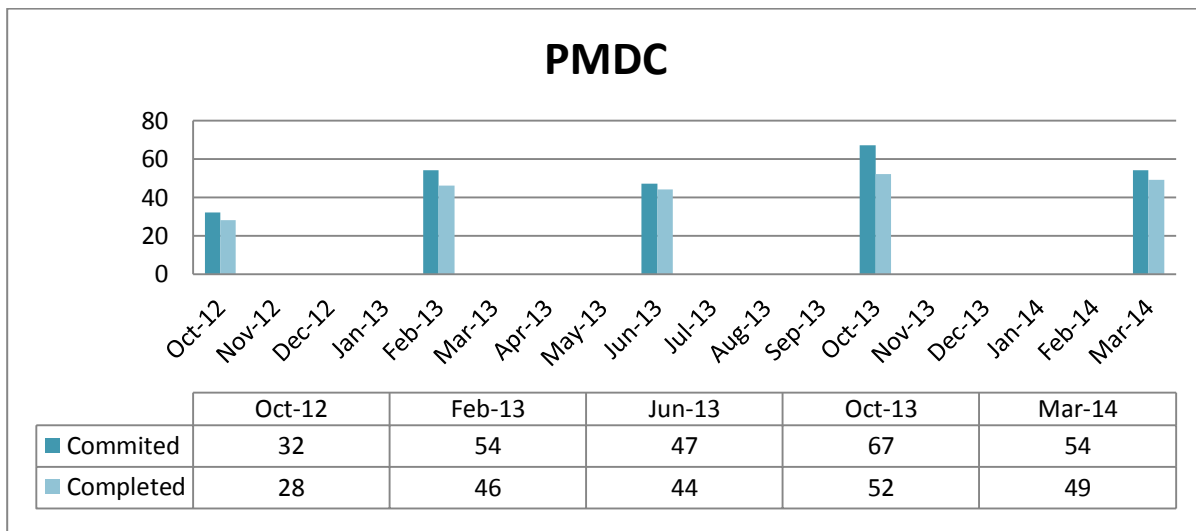


Figure 6: Chart Showing the Velocity of the Tasks in PMDC

From the above chart we can see how many tasks were committed to be completed in each quarter and how many were completed.

Total number of tasks committed throughout the project = 254.

Total number of tasks completed at the end of the project = 219.

From the numbers collected the efficiency of the methodology in completing the task can be calculated for this project using waterfall methodology as

$$\text{Efficiency} = (219 * 100)/254 = 86\%$$

The data received for the Agile approach is related to the tasks that were undertaken in each Sprint and how many number of tasks were completed in respective Sprint, that gives us the percentage of work done in that particular sprint.

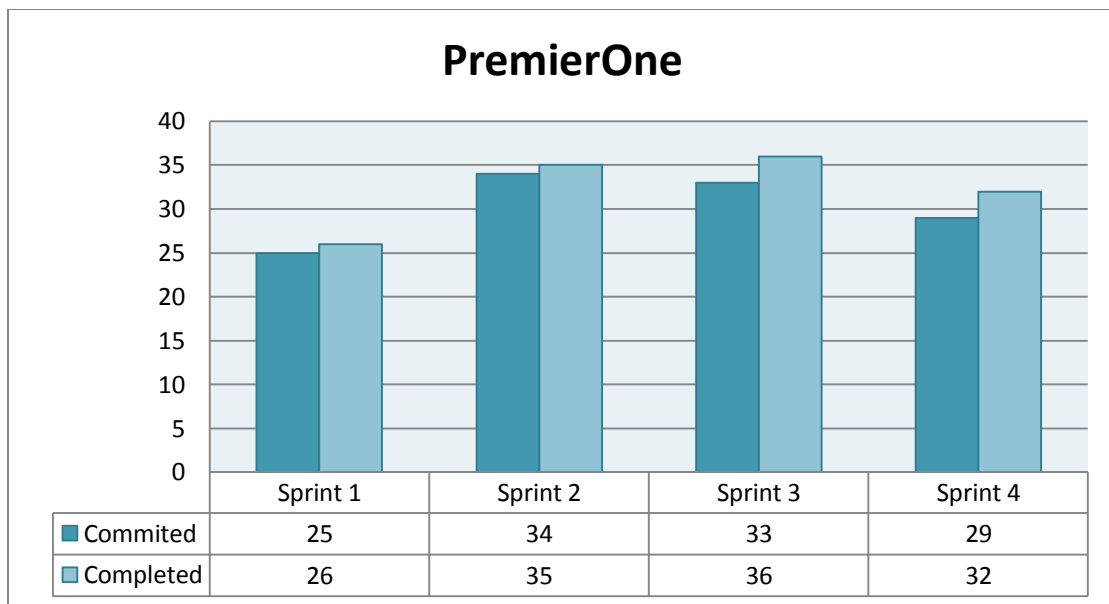


Figure 7: The Chart Showing the Velocity of the Tasks in PremierOne

From the above chart we can see how many tasks were committed to be completed in each quarter and how many were completed.

Total number of tasks committed throughout the project = 121.

Total number of tasks completed at the end of the project = 129.

From the numbers collected the efficiency of the methodology in completing the task can be calculated for this project using waterfall methodology as

$$\text{Efficiency} = (129 * 100)/121 = 106.61\%.$$

Comparing the efficiency of the two projects we can see that there is an increase in the efficiency by nearly 20% in completing the tasks in the Agile

implemented project when compared to Waterfall. This efficiency is purely dependent on the particular project and cannot be generalized.

Success rate. Apart from the data related to the tasks undertaken, the data related to the success rate of tasks has also been collected to see how efficiently the blockings were removed in both the methodologies. Considering the data related to the waterfall

Total number of tasks = 254

Number of successful tasks that were completed in single attempt without any difficulties = 43

Total of 17% were successful.

Number of tasks that were been challenges due to the waterfall model = 137

Total of 54% were challenged.

Number of tasks that were failed and had to be re-done = 74

And total of 29% were failed to be completed on time.

Taking all this data in to consideration generating a pie chart gives a clear picture.

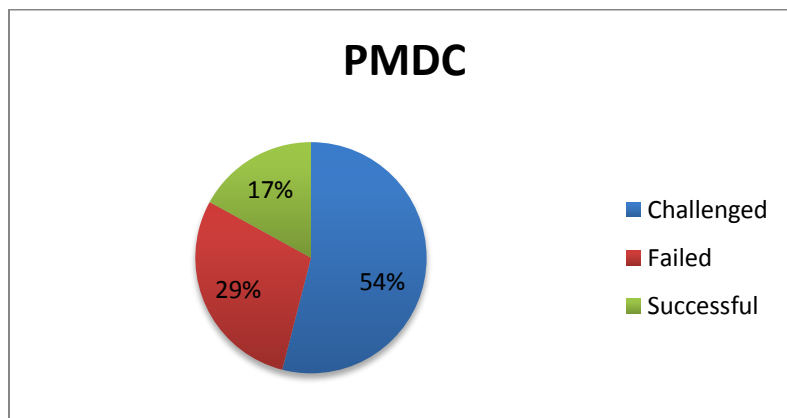


Figure 8: The Chart Showing Success Rate of the Tasks in PMDC

In addition to the data related to the project PMDC that was implemented in the waterfall methodology the data was collected related to the PremierOne project that was implemented with the Agile Methodology. This data comprises of the tasks that were Successful, Challenged and failed throughout the project.

Total number of tasks = 129

Number of successful tasks that were completed in single attempt without any difficulties = 64

Total of 49% were successful.

Number of tasks that were been challenges due to the waterfall model = 54

Total of 42% were challenged.

Number of tasks that were failed and had to be re-done = 11

And total of 9% were failed to be completed on time.

Taking all this data in to consideration generating a pie chart gives a clear picture.

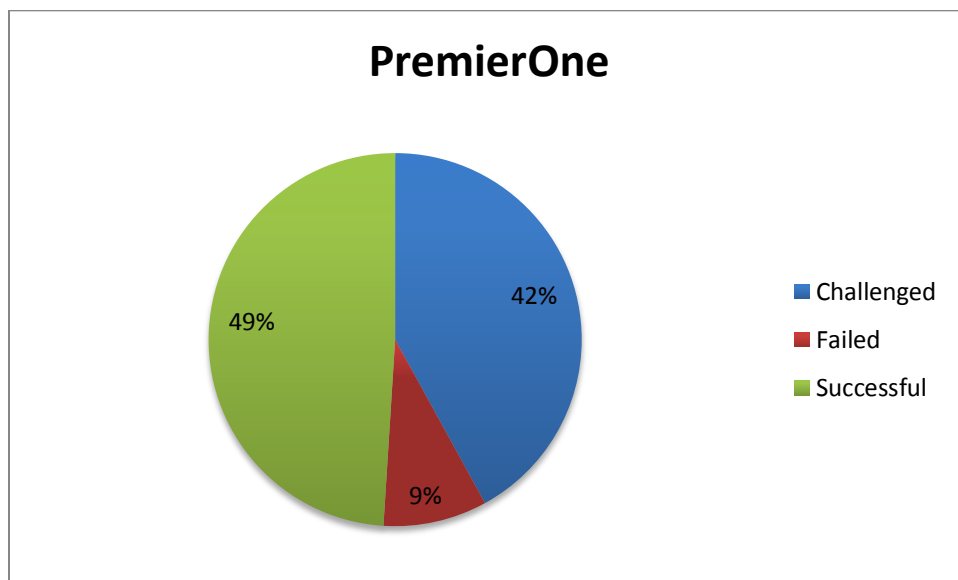


Figure 9: The Chart Showing Success Rate of the Tasks in PremierOne

The data related to cost was not disclosed in numbers but it was given in relevance to the scale of 100 and the other factors like features or tasks and time is also being converted in to the scale of 100 and the data has been used to generate the chart below.

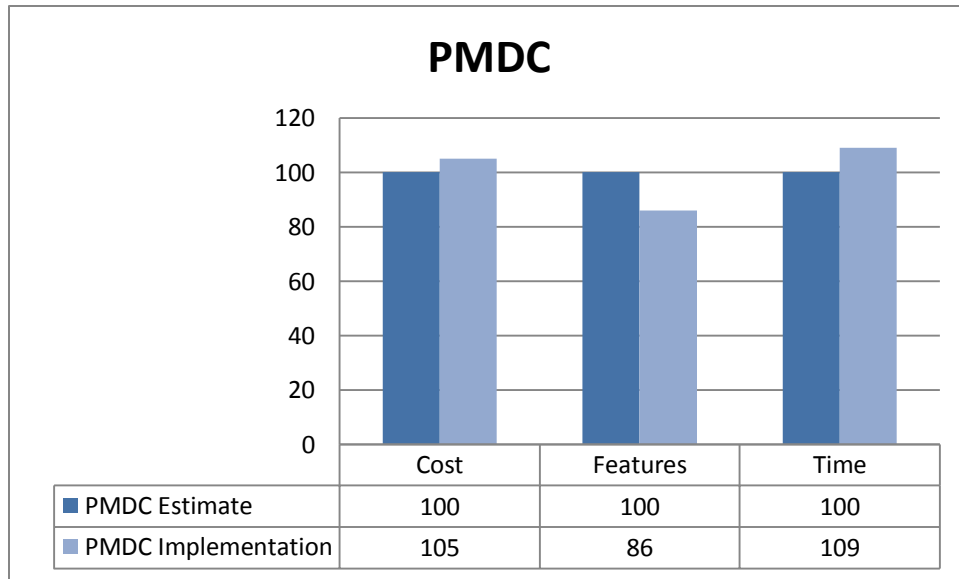


Figure 10: Chart Showing the Difference in Estimated and Implemented Factors in PMDC

The same data has been collected for PremierOne project that was implemented using the agile methodology. The cost and the time has been scaled down to the scale of hundred and is been compared to the estimated and the results are populated in the form a chart in the following page.

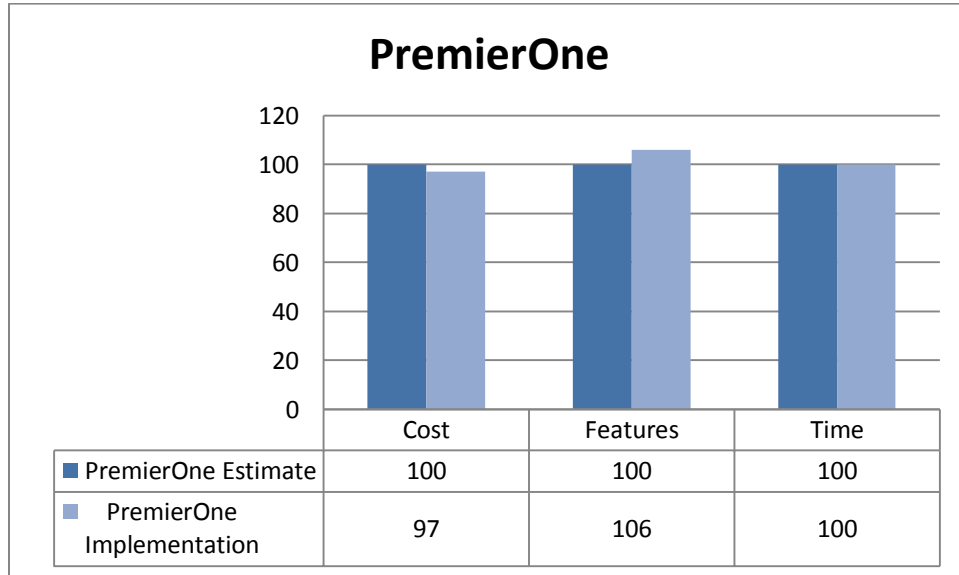


Figure 11: Chart Showing the Difference in Estimated and Implemented Factors in PremierOne

Combining both the charts we can get the clear picture of the better approach for the project that costs less and completes in given span of time with more efficient task completing capability.

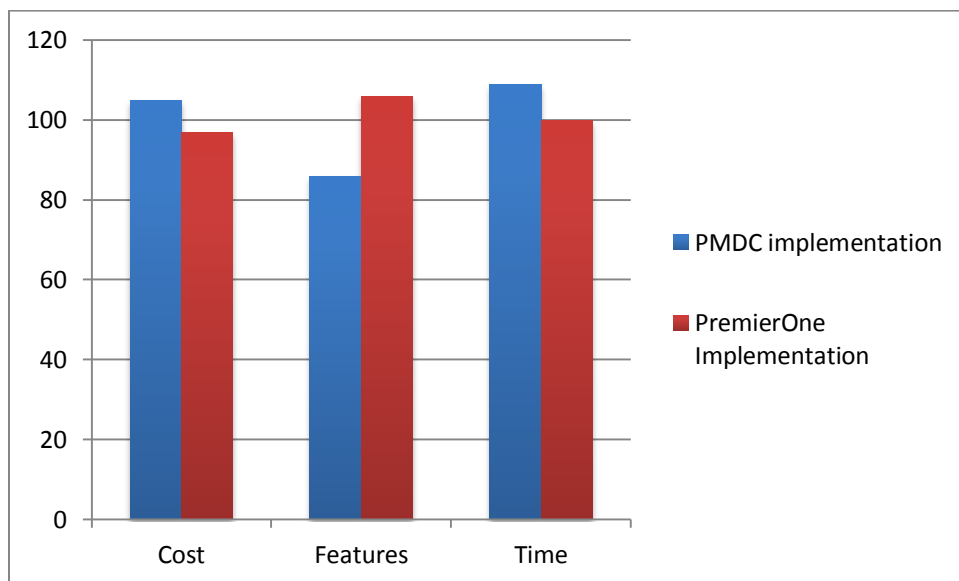


Figure 12: Comparing the Factors for PMDC and PremierOne

The data shows how the cost and the time to complete the project have been reduced by implementing the Agile Methodology and also there is the increase in percentage of features or tasks that has been worked on during the project.

After the project the survey was also conducted on the thoughts and views of the team and also the customers to find what their perspectives would be and also to get some suggestions in order to make the Agile more efficient to increase the productivity and also to complete the project in much lesser time and avoid any blockings that were experienced in the PremierOne.

Survey analysis. The responses received were nicely tabulated so that it will be easier for the analysis and as they are used to compare with two methodologies. These are imported in to the charts for comparison to generate the clear picture of the difference in the feedbacks for both the approaches.

If we look at the table with the data specified in the table of the waterfall model, it is clearly evident that neither the developers nor the customers were satisfied with the waterfall model looking at the results it yield in the previous projects. It mostly consists of 1 and 2, which shows that the change in methodology was necessary. As per data, it was very clear that the necessary communication was lacking among the team members. The most concerning statistics was that scores that “late changes and the customer’s involvement” received. All the developers agreed that the late change was not acceptable in Software Development Approach as the methodology was not reciprocal to the change. The lack of customer’s

involvement in development means that the possibility of the product not meeting the specification was very high.

Whereas in the table related to agile methodology the scenario changes completely. We can only see 5's and 4's for all the aspects for which the survey was conducted. We can understand that the level of communication was very high among the team members. The continuous iterations mean that the team was happy to accept the late change request from the customers. Since the Stakeholders were involved in the iterations, they were able to get access to the product early and provide their feedback. Due to this factor, the product specification and the involvement of the stakeholders was very high.

From the tables an average of the response can be calculated from all the individuals.

Calculation. Summing up the value for the factor 'Expectations reached' in waterfall model and then getting the average of all the responses.

$$\begin{aligned} \text{Total} &= \text{Participant1} + \text{Participant2} + \text{Participant3} + \text{Participant4} + \text{Participant5} \\ &= 3 + 2 + 2 + 3 + 2 = 12 \end{aligned}$$

Average for Expectations reached factor in waterfall model would be

$$\begin{aligned} \text{Average} &= \text{Total}/\text{No. of participants} \\ &= 12/5 = 2.4 \end{aligned}$$

Applying the same logic for the Agile methodology

$$\begin{aligned} \text{Total} &= \text{Participant1} + \text{Participant2} + \text{Participant3} + \text{Participant4} + \text{Participant5} \\ &= 5 + 5 + 4 + 5 + 5 = 24 \end{aligned}$$

Average = $24/5 = 4.8$

For the factor 'Communication' applying the same logic as above we get the calculated value as

Average for Water fall = $7/5 = 1.4$

Average for Agile = $24/5 = 4.8$

For the factor 'Flexibility'

Average for Waterfall = $6/5 = 1.2$

Average for Agile = $24/5 = 4.8$

For the factor 'Timely Delivery'

Average for Waterfall = $9/5 = 1.8$

Average for Agile = $19/5 = 3.8$

For the factor 'Recommended'

Average for the Waterfall = $8/5 = 1.6$

Average for Agile = $23/5 = 4.6$

The data calculated above was imported into Excel sheet and when compared gave the clear picture of the difference between the level of satisfaction that differs between two methodologies.

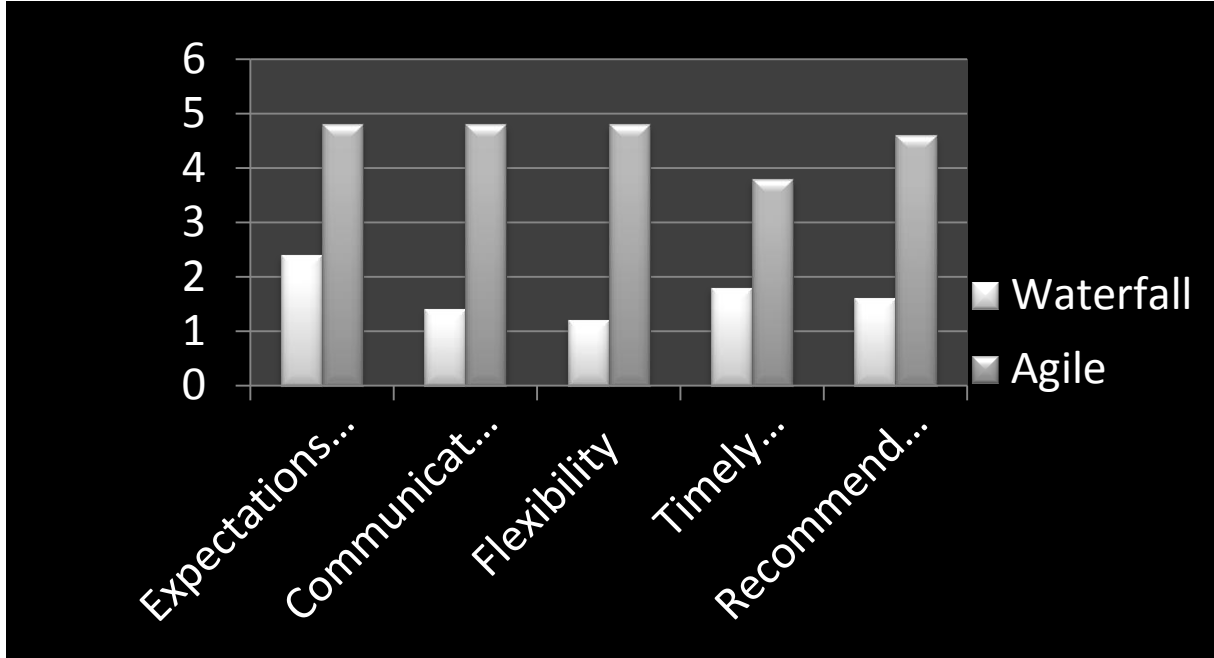


Figure 13: Chart after Data Analysis on Feedback of Both Approaches

The data obtained from the survey can also be represented by implementing a pie chart. It gives us the most favorable among the two methodologies. In order to create a pie chart we need to add the total responses of all the aspects obtained from the participants. Then taking an average of all the responses gives us the rating of each methodology.

Adding the average values obtained for waterfall model:

Total = Avg. of the factor 'Expectations Reached' + Avg. of factor 'Communication' + Avg. of factor 'Flexibility' + Avg. of factor 'Timely Delivery' + Avg. of factor 'Recommended'

$$\text{Total} = 2.4 + 1.4 + 1.2 + 1.8 + 1.6 = 8.4$$

Adding the average values obtained for Agile

Total = Avg. of the factor 'Expectations Reached' + Avg. of factor 'Communication' + Avg. of factor 'Flexibility' + Avg. of factor 'Timely Delivery' + Avg. of factor 'Recommended'

Total = 4.8 + 4.8 + 4.8 + 3.8 + 4.6 = 22.8

Adding those two totals we get a Grand total of 31.2

Here in the Grand total the amount favorable for the agile methodology is comprised of 73.1 %

Whereas for Waterfall model it falls down to mere 26.9%.

Using these results we can represent the data in the picture that shows the most favorable methodology.

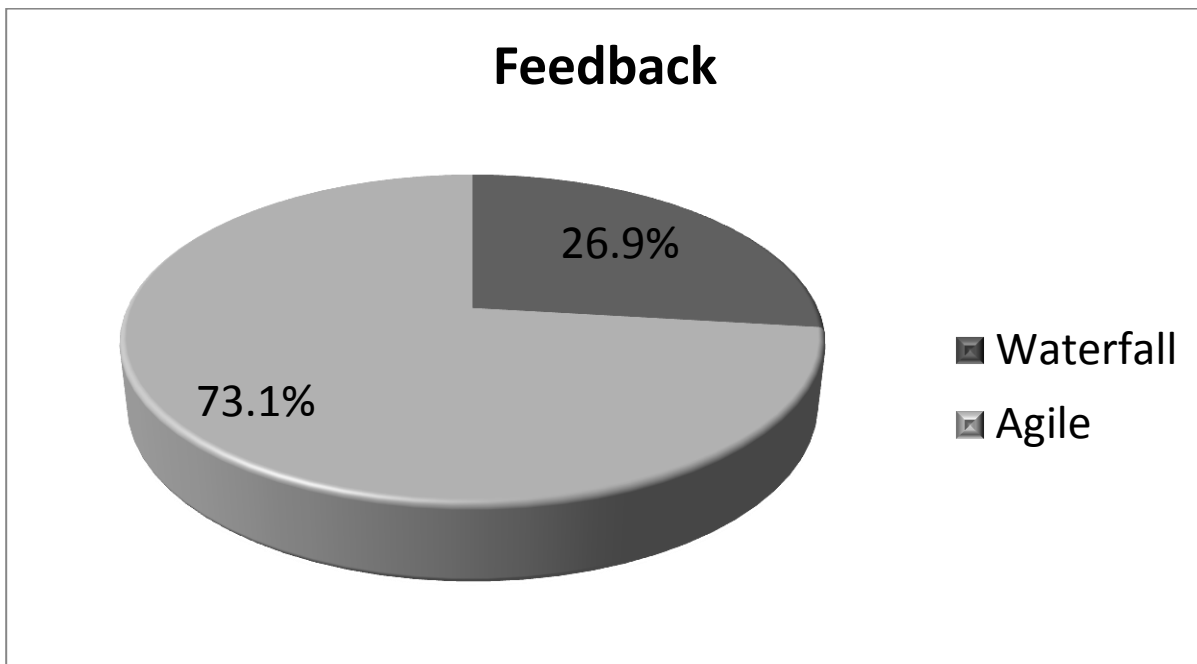


Figure 14: Pie Chart Comparing Two Methodologies

In this context it is clearly evident that Agile methodology is more favorable than the old traditional Waterfall model.

As the handout also had requested for their valuable suggestions. Different responses were obtained from the customers and also the developers.

Suggestions from the team and customers. Summing up the suggestions obtained, there are few that were common among them. They are listed below.

Documenting the work. In this fast moving methodology there is no time allocated for the documentation as the sprint is mainly assigned for developing and testing. This makes it difficult to the developers that are new, who join in the course of the project timeline and the customers to understand exactly the functioning of the product.

Impact of non-Agile teams. As a member of the agile team is completely involved in the tasks assigned to them and have occupied schedules mostly. As there are other teams that do not work on this methodology may not understand the process and may interfere with any other issues asking for help on any tickets raised. That might be little but is surely a distraction and interference when it comes to the methodology. Apart from requesting for help, if the developer who is involved in the project needs any access in order to complete the task and if he/she is not authorized to do so, may have to wait until the ticket passes through the management to grant access.

Incomplete product. As the whole project is divided into the number of sprints and the customer usually expects the product to have the full functionality and

could not see what is expected then it would be a drawback. As it is not necessary that each and every task in the sprint should be completed, there might be unexpected barriers that may act as hurdles for the task to be completed.

Scrum Master. Mostly the scrum master performs a role of a manager who can remove the barriers in order to let the developers perform the task given. He will be responsible for creating an environment favorable for the developers to complete the task. In real time scenario usually the scrum master performs two roles, apart from being a scrum master he might be a developer or a BA. This would result in lack of proper composure towards the task and might create confusion and delays if the scrum master fails in any of the two roles that are played.

Proper training. As this is the new methodology implemented in the firm not many would be aware of the methodology. So, if even one does not go through proper training then it might develop a confusion among the other team members and the project might be side tracked and may experience delays in delivering the product.

Summary

This chapter presented that the data that were collected during the research process. Those collected data were analyzed into various tabular and graphical formats to provide the reader with a better understating of the results. The suggestions that were retrieved during the survey are also discussed in detail. The next chapter will cover the results, recommendation and conclusion of this research project.

Chapter V: Results, Conclusion, and Recommendations

Introduction

This chapter will cover the results and conclusion that has been obtained from the research. The project questions, which were discussed at the beginning of the project, will be answered. The limitations of the methodology will also be addressed and the recommendation to improve the approach will be discussed.

Results

The results can be defined from the feedback received at the end of the project and also with the project questions that were posted at the beginning of the project.

Project Questions

1) Was the Agile/Scrum methodology compatible for the project undertaken?

Agile methodology is more iterative, more flexible and mostly customer oriented, which responds well to the changes made even after the initial design of the product. In this project of the firm the methodology that should be used must be customer oriented and should accept changes in the course of the project timeline. We talk about the customer oriented approach because of the present day situation where the firm's success is mainly depended upon the customer satisfaction.

In previous projects undertaken by the company, the approach used was the traditional waterfall model that didn't encourage any changes that came

after the design phase. There are some factors that need to be considered before we implement this methodology on the project. They are,

The agile methodology usually encourages a small team to be involved in the project which is well suited in this case as the team working on this project was of just six members.

In this methodology there must be only one team that can handle almost all the aspects related to the work and as mentioned the team includes the developers the testers and also the scrum master who acts as an executive that can handle all the management related matters so that the team do not have any obstacles in the course of the project timeline.

As the idea was to prioritize the customer needs and deliver them first, the agile methodology helps in achieving that kind of commitment where the customer is provided with the features which are requested first and that makes the agile methodology to be compatible with the project undertaken.

- 2) How was the Agile/Scrum Methodology more effective than the traditional waterfall model?

It can be seen from the review collected at the end of the project, how the feedback differs for both the methodologies. There are some aspects where the Agile methodology seem to be more effective compared to traditional waterfall model.

Responsive/Flexible. Before applying the agile methodology the company was implementing Waterfall model in developing the product where there was

no scope of change in the middle of the project. Customers come up with new requirements that need to be included in the product but, as the design is already done and estimates are already made, it was difficult for the firm to include the changes; even if the changes are included they influence the budget and estimated time of delivery. But, in case of agile methodology though the customer comes up with the requirement that needs to be included in that sprint, it is prioritized and can be included any time during the project timeline making it more flexible compared to the waterfall model.

Communication. In waterfall model there was no communication between the customer and the developers that work on the product. This made the developers to be isolated from the customer needs that were growing. It impacted at the end of the project when customer showed dissatisfaction on the product delivered. But, this was not in case of Agile methodology. The customers and developers along with the product owners met at the sprint review and discussed the requirements the customer came up and also the product that was built is also demonstrated and feedback was received at the end of each sprint which increased the scope of improvement before the final product is delivered.

3) How was the involvement of the customers during the project?

When this methodology was implemented, it was a big question among the team that whether the customers will be willing to spend time in meeting with the agile team to discuss about the project. The team went on to discuss with

the customer about the approach that is been implemented to deliver the product on time and to include the changes required even in the middle of the project, the customer representatives were comfortable to extend their support and time for the project to be completed promptly with the desired results. They were also excited to be included in the meetings as they will be observing the product unfolded as the time passes on.

A group of representatives from the customer's side used to attend the sprint retrospectives along with the product owners and provided useful suggestions and ideas that could make the product more efficient and user friendly. They also appreciated the team for their efforts and that boosted the morals up and helped in motivating the team members to work hard and satisfy the customer. At the end of the project the customers also provided the feedback which helped a lot in gathering the data for the analysis to come to a conclusion on implementing the agile methodology.

From the observation it is clearly evident that in order to make the project successful, the customer involvement is a major factor, as we can keep track of their views and their expectations on the product.

4) Did the project end on time and within the budget?

As per the previous experiences the factor that was mainly on cards was the project to be completed on time and within the budget, which was not met by implementing the traditional waterfall model. The project was behind the schedule and it was not under the estimated budget.

After implementing the agile methodology, all that the team estimates is the time and budget related to that sprint and constant communication with the customer so that, if the customer comes up with the new requirements then the customer can be made aware of the influence they have on the budget and also the estimated time. Which leaves the customer in the awareness of the completion time so that at the end of the project there would be no complains on project being not completed on time.

As the customer was made aware of the increase in time and also the difference in the budget, the customer was satisfied with the time taken for completion of the project and the total budget. The estimates were not shared numerically but, it was mentioned that the project met the expectations within the allocated budget and expected span of time. It can also be inferred from the feedback that the product was delivered on time.

5) What is the customer feedback at the end of the project?

Implementing the new methodology was mainly intended to satisfy the customer. So, the customer's review was very much required to conclude on the effectiveness of the methodology when compared to the old traditional waterfall model. Hence the survey was conducted including the customers to obtain a feedback so as to come to a conclusion.

The survey was done comparing two methodologies, using five different factors. Does the product reach their expectations, how well was the communication, how flexible was the project in order to include the changes,

was the product delivered on time and also is it recommended to use the methodology in future.

Considering the feedback received from the customers it can be inferred that they were very much satisfied with the agile methodology and also gave higher ratings when compared to the traditional waterfall model. The data analysis made from the feedbacks received gives the clear picture of the level of satisfaction of the customers on implementing the agile methodology in the place of traditional waterfall model. The customers gave a positive feedback on all the factors and also recommended the implementation of the agile methodology in future projects.

Conclusion

The main intention of the project was to evaluate how effective was the implementation of the agile methodology in software development by implementing it in a project as well as determining how well it satisfies the customers. The traditional approach was very much a linear model where the phases are considered to be implemented one after the other. The next phase is considered only after the completion of the previous phase. It also lacks the communication and flexibility of including the changes proposed by the customers between the ongoing projects.

The agile methodology was compatible in many ways in the software development and the project undertaken. It does not affect any software development rules and also it was well suited to the project where the customer involvement is required, mostly to involve them in the project to know their views and

perspectives in order to deliver the most desired product by the customer. The customer's involvement was necessary so as to keep track of their growing requirement which was not possible using the traditional waterfall model. The only way to implement that was by using the agile methodology.

The implementation was successful and the methodology proved to be compatible with the project and also satisfied the customer. These conclusions are made from the survey conducted at the end of the project, where the customers and developers were asked to rate the methodology with the common questions that are related to the project and can be answered by both the groups. In conclusion the agile methodology is recommended in future projects and it's time to leave behind the traditional waterfall approach, looking at the growing business that is mainly focused towards the customer satisfaction.

Recommendations

Throughout the project and after completion of the project we have seen the implementation of more iterative approach and came up with some recommendations.

- i) Increase the no. of team members in the team so that the team may include other specialists in order to restrict the project going off-track due to the non-agile teams.
- ii) Let scrum master be only responsible for the job he is called scrum master for. Assigning other tasks may affect the performance of the team.

- iii) Looking at the results it can be recommended to implement agile methodology in future software development projects.
- iv) Proper training in this methodology is necessary before starting the project in order to avoid confusion among the team members.
- v) Code possession ought to be supported all through the group. This will support all the colleagues to learn about the framework and minimize probability when managing defects.
- vi) Proper communication should be established between the agile and non-agile teams so as to minimize the delay caused by the hierarchal process of non-Agile teams.

References

- Agile software development methodologies*. (n.d.) QAIT DevLabs. Retrieved from <https://www.pinterest.com/qaitdevlabs/agile-software-development-methodologies/>.
- Blankenship, J., Bussa, M., & Millett, S. (2011). *Pro Agile .NET development with Scrum*. Apress.
- Cairns, A. (2012, October 11). pmhut. Retrieved January 25, 2015, from The Project Management Hut: <http://www.pmhut.com/comparing-agile-and-waterfall-methods-of-project-management>.
- Hughey, D. (2009). Comparing traditional systems analysis and design with agile methodologies. Retrieved from <http://www.umsl.edu/~hugheyd/is6840/waterfall.html>.
- Introduction to agile software development. (2007, June 5). *Serena*. Retrieved October 24, 2014, from <http://www.serena.com/docs/repository/solutions/intro-to-agile-devel.pdf>.
- Massey, V. (2002). Evolving a new software development life cycle model (SDLC) incorporated with release management. *International Journal of Engineering and Advanced Technology*, 1(4), 1.
- SDLC. (2014, October 6). *Waterfall model*. Retrieved October 24, 2014, from <http://www.waterfall-model.com/sdlc/>.
- Software development life cycle (SDLC)*. (2007). New York: Tutorialspoint.com.