A Structured Approach to Anyalyze Scrum, Xp or Spiral Model for Qualitative Development of Software

Sana Saeed[1], Akmal Rehan[2], Iqra Mahmood

[1,2,3]Department of Computer Science, University of Agriculture Faisalabad, Pakistan

ABSTRACT: *Now a day's success and failure of Software Company depend upon the selection of an appropriate model for the development of the product. Many software methodologies were used to develop the quality software. But it is still a challenge for developers to select which methodology may be best suited for software development. In this research some methodologies such as XP, spiral and scrum were analyzed along with their strength and weaknesses and find out the best suited methodology in various situations. This analysis also helps the software practitioners in selection of model which save time and provide customer satisfaction by in time delivery of right product. Efficient utilization of' resources was also being done by the model which grows up the company.*

Keywords: *Extreme Programming Xp, spiral, scrum*

I. Introduction:

Software process is the protection of any software development. The survival of any software product in the market and organization depends on the process model. Some organizations find that the process model is more efficient and some of the continent found that another core goal of each organization is to achieve high quality products with less resources and time. Agile Process Model Is Better Than Quality Development [1].

The research exhibits, that which process model is best for the development of software. XP is one of the software development methodologies so first we will focus on Xp then on Scrum and in the last spiral process model.

A. Extreme Programming:

Kent Beck first introduced extreme programming methods. The basic principles of XP are communication, simplicity, feedback, courage and respect [2]. XP starts collecting user requirements. On the basis of these requirements, the entire development process is divided into a small number of cycles. The next stage is iterative planning, that is, determining the number of cycles, requiring prioritization, and estimating the amount of effort required to implement each cycle. Use each iteration for programming. New user requirements may be in the development phase, and the iteration plan should be adjusted accordingly. In the next step, if an error is detected for the latest development version, the error will be eliminated in the next iteration. After each acceptance test project is tracked, feedback should be obtained from the project on how much work has been done [3]**.**

The methodology is based on the following.

1) **Communication:** XP has a verbal communication culture; its practice aims to encourage interaction. Communication values are based on the observed majority of project difficulties, because someone should speak to others to clarify issues, collaborate, or get help. The problem with a project can always be traced back to the fact that one does not talk to others about important things.

2) **Simplicity** Designed to meet the needs of customers the simplest products. An important aspect of value is the design and coding of the content in the current requirement, not the expected and planned non-stated requirements.
3) **Feedback:** The development team gets feedback from customers at the end of each iteration and external release. This feedback drives the next iteration. In addition, through programming and test-driven development, a very short design and implementation feedback loops are built into the method.
4) **Courage:**

The other three values allow the team to have courage in their actions and decisions. For example, the development team may have the courage to resist the pressure to make unrealistic commitments.

B. SCRUM:

Scrum found its true origin at Easel in 1993, where it was successfully applied to a software project. In software development, the job is to publish a version. Scrum's software has developed a rapid prototyping community because prototyping people want a way to support an environment where requirements are not only incomplete at the beginning, but can also evolve rapidly during development. Unlike XP, Scrum methods include management and development processes [4]. Scrum is an iterative and incremental system, rather than programming and project development in the more common continuous waterfall structure. The Scrum system takes into account less stringent requirements and less mature responses to new data and abrupt changes. The system consists of several roles, events and artifacts.

In the Scrum process, a project management wrapper around a software development method. The method is flexible how much / how many rituals, but the Scrum philosophy will guide a team with as few rituals as possible. Usually a Scrum team works in the same location. However, there is already a Scrum team working geographically, and team members attend daily meetings via hands-free phones. The Scrum team is self-directed and self-organizing. The team is committed to a defined iteration goal and empowers power, autonomy and responsibility to determine how best to achieve it[5].

There are three main roles in scrum, Project Owner, Scrum Master, and Team. Together they were called the Scrum team. The person in charge of the product is the person responsible for the software. The individual organizes things according to the wishes of the software and the partners. Scrum Master is a person who participates in Scrum, he or she sports product leader and team according to Scrum rule. Scrum Master also shields the group from external interference, all groups together need to work as the center. The team is a collection of about seven individuals that implement an organized task and add it to the final product. Forward work is done in a round called Sprint. Each sprint can be from one to four weeks in length and repeated immediately.

C. Spiral:

Spiral model risk identification is concerned with any other incremental model. It has four phases, namely planning, risk identification, design and evaluation [1]. There are distinctly different requirements in the planning, such as business and software requirements gathering requirements gathering phase. The next stage is to identify the stage of the hazard identification phase and its alternative risk solution. In the product development phase of the project, then it. The customer outputs test and evaluation software products and starts the next spiral. Quality is always in the spiral model because, like analytics, prototyping, development, validation and validation, and testing every step to ensure that development does not go into the next phase until each stage is a satisfactory activity [6].

D. Phases in Spiral model:

1) Planning Phase:

Requirements are collected throughout the planning phase. Requirements like ‘BRS’ that is ‘Business Requirement Specifications’ and ‘SRS’ that is ‘System Requirement specifications’.

2) Risk Analysis:

In the risk analysis phase, a process is used to identify risks and alternative solutions. Prototypes are generated at the end of the risk analysis phase. If any risk is found during the risk analysis, an alternative solution is recommended and implemented.

3) Engineering Phase:

At this stage, software development, and testing at the end of the phase. Therefore, at this stage, development and testing are complete

4) Evaluation phase:

This phase allows the customer to evaluate the project's output before the project proceeds to the next spiral.

Table1.

II. Characteristics of different process models:

Characteristics	Xp	Scrum	Spiral
Approach	Iterative	Iterative	Incremental
Iteration Time	1-3 weeks	1-2 weeks	2-12 months
Size of Team	Less than 12 persons	5-9 person	15 persons
Customer involvement	User is highly involved	Though product owner	Early user involvement
Project Size	Small	All types of projects	Large projects / risky projects
Practices	Pair programming, user stories, on-site customer involvement	Product owner, sprint planning, scrum master, backlog	Risk assessment, constraint checking, plan for next phase, commitment
Documentation	Basic	Basic	Excessive

III. Evaluation of different process models.

Each of these models which are under study were analyzed in detail and 40 different practices were identified, based on these 40 practices 3 under study models were evaluated (table 2). Hence it is possible to find out the situations in which these models can be used. Below the table list practices of different models (Y) indicate the particular practices for the particular model as per literature study. For example, incremental development is the characteristics of both Xp and spiral model processes

Table 2

Sr. No.	Practices	Xp	Scrum	Spiral
1	Iterative development	Y	Y	
2	Incremental development	Y		Y
3	Object oriented development	Y		
4	Managerial Development		Y	Y
5	Oral communication with the customer	Y		
6	Simple design	Y		
7	Feedback after every external release	Y		Y
8	Courage to make a decision	Y		
9	Respect to the team members	Y		
10	Co-located team work	Y	Y	
11	Geographically distributed team participation		Y	
12	Self-organized teams		Y	
13	User story card usage	Y		
14	CRC	Y		
15	Pair programming	Y		
16	Customer Acceptance Test	Y		
17	Test first program	Y		
18	Refactoring	Y		
19	Daily Meetings	Y	Y	
20	Rapid prototyping		Y	Y
21	Product backlog		Y	
22	Sprint planning		Y	
23	Sprint backlog		Y	
24	Product owner		Y	
25	30 day's increment		Y	
26	Focus on actual rather than presentation only		Y	
27	Risk analysis			Y
28	Engineering and evaluation			Y
29	Evaluation of output by the customer			Y

30	BRS document			**Y**
31	SRS document			**Y**
32	Testing at the end			**Y**
33	UML of each phase			**Y**
34	Time management	**Y**	**Y**	**Y**
35	On site customer involvement	**Y**	**Y**	
36	Development with vague requirement	**Y**	**Y**	
37	Small scale development	**Y**	**Y**	
38	Large scale development			**Y**
39	Functionalities are prioritized	**Y**		
40	Early defect removal	**Y**		

IV. Research methodology:

All the data were gathered from literature survey were analyzed to find out the practices of different models. Software houses. The questionnaire was then prepared on the basis of these practices. Online survey method was used for data collection. In the survey respondents were asked to select the best suited practices as mentioned above. The results of the mentioned techniques were analyzed by applying statistical tool.

V. Results of the Study:

In order to obtain the results of the research study, the survey was conducted. This survey was used for compression purpose to check which model should be used for the development of small to medium size software. During the experiment it has been observed that in some cases spiral gave better result in large projects due to its risk identification nature and when budget the budget is constraint for the customer and development team. When the requirements are complex and need valuation for clearness, on the other hand some situations shown that scrum gave better performance in terms of small projects on the basis of sprint planning and daily meetings. In scrum development customer is not directly involve with the development team product owner is the representative of the customer, scrum focus on the big picture rather than the small component development. One advantage of scrum is that scrum teams are geographically distributed and can participate in meetings. On the other side in spite of spiral and scrum in most of the cases Xp gave good results because of acceptance test driven development and pair programming. Xp involved more customers throughout the development and also focus on peer review of the code on basis of pair programming. User stories are means of conversation between user and development team. BRS document and SRS document is prepared. Among all these models Xp gives better results for the development of small to medium scale software development.

VIII. Acknowledgement:

This research was supported by university of agriculture, Faisalabad.

VI. Conclusion:

Software processes are important because they provide guidance on the order in which projects should perform their main tasks (phases, increments, prototypes, verification tasks, etc.). Many software projects have been grieving because they seek a variety of development and evolution stages in the wrong order. The purpose of this research was to test which process model is best for the development of the of software in software houses. A case study was conducted in different software houses, where developers use different process models for the development of the software. It was set out to be an experiment to conclude that which process models best among Xp, spiral, and scrum. The experiment is lasted for 5 months. The deployment of the Xp model for that particular case study can be seen as a success, in spite of not satisfying some criteria and breaking some of its standards. This experiment will help the software practitioners in selection of model which will save time and provide customer satisfaction by in time delivery of right product. Efficient utilization of resources will also be done by the model which will grow up the company.

VII. Future work:

There are other agile software development methodologies as well. These include adaptive software development, agile modeling, dynamic systems development method, Lean development. Xp should also be tested with these methodologies so that it can be observed that it is best among them.

References:

[1] Hashmi, S. I., and Baik, J. 2007. Software Quality Assurance in XP and Spiral - A Comparative Study. 2007 International

Conference on Computational Science and Its Applications ICCSA 2007).

[2] Abrahamsson. 2013. Extreme programming: first results from controlled case study. In Proceedings of the 20th IEEE Instrumentation Technology Conference, EURMIC-03:pp. 259–266.

[3] Matharu, G. S., Mishra, A., Singh, H., and Upadhyay, P. (2015). Empirical Study of Agile Software Development Methodologies. SIGSOFT Softw. Eng. Notes ACM SIGSOFT Software Engineering Notes, 40(1), 1-6.

[4] Scharf, A. and A. Koch. 2013. Scrum in a software engineering course: An in-depth praxis report. Software Engineering Education Conference, Proceedings, 159–168.

[5] Chhavi. M., Anuradha Chug.2013. Agile Testing with Scrum- A Survey , International Journal of Advanced Research in Computer Science and Software Engineering 3(4), pp. 452-459.

[6] Boehm, B. and B. Boehm. 2016. The Spiral Model as a Tool for Evolutionary Acquisition.International Journal of Advance Research in Computer Science and Software Engineering, 2(8): 2277-2279.