

IXSCRUM-A Framework Combining Scrum and XP

Chhavi Malhotra, Anuradha Chug

Abstract— Abstract--Scrum approach being developed for managing the software development process gives no idea about how to engineer the software product. In order to adapt to the agile environment , it is desirable to construct a framework that boasts of the engineering aspect of individual approaches of software project development. XP is an iterative , lightweight , adaptive approach of agile methodology which helps to engineer a software project . In order to build quality in a product and to satisfy customer requirement on timely and efficient basis it is desirable to have a solution which uses the advantages of both the approaches of XP and Scrum. In our research we have proposed a framework that combines this integration. This model has the engineering approach of XP along with the management approach of scrum. The suitability of this approach has been verified and validated by a case study.

Index Terms – XP, Scrum, Quick Test Professional, Burn down, Artifacts

1 INTRODUCTION

In old days we had project teams where every individual played every possible role. For e.g. Tester played the role of designer, developer played the role of system analyst. With the introduction of large and complex projects, it was important that the quality should also be maintained. Much later in the year 2001 came a word “Agile” which was suited to rapidly changing needs and functionalities of users. Since then agile grew into practice which was adopted and is fully functional since then. Agile manifesto [1] highlights some proponents in which it works. Agile movement emphasizes the relationship and commonality of software developers and human role reflected in contracts, as opposed to processes and tools.

Agile manifesto is close team relationships. The main objective of software team is to continuously turn out tested software. New releases are produced at frequent time interval, in some approaches even hourly or daily. The developers are asked to keep the code simple and straightforward and thus less amount of effort spent on documentation. The relationship between client and developers is given preference over strict contracts. Agile values are focused on delivering business values immediately as the project commences, thus reducing the risk of non-fulfillment of the contract. The development group comprising software developers and customers should be competent

enough in order to consider possible adjustment needs emerging during the development process lifecycle. This means that participants are open to changes and also the existing contracts are formed with tools that allow these changes and enhancements to be made.

A. Extreme Programming (XP)

Extreme programming also called XP in short was coined by Kent Beck, is very popular and intuitive approach in agile. It promotes customer involvement as proposed in agile where customer is of utmost priority [11]. It promotes uninterrupted testing, constant feedback, planning and close teamwork to deliver working software at regular intervals .Xp coordinated with customer and thus “User Stories” are formed. The development team delivers highest priority tested working software on iteration by iteration basis.

The phases in XP are: Exploration phase, Planning, Iteration to Release, Productionizing, Maintenance and Death phase [2] XP outlines 12 rules which are systematic and implemented in the way they are built. XP begin with planning game where the customer, developers and testers meet in order to discuss about the requirements in terms of user stories captured on story card. An initial version of system is put into production after few iterations. Developers keep the design as simple as possible gradually progress to a more detailed design later on but again here the design is made as simple as possible. Developers write the acceptance test first before writing the code. Pair programming is followed as a technique for code progression. Developers add new code very often in the system by the process of continuous integration. The code is made accessible to all the developers and any changes in the code can be incorporated at any

- Chhavi Malhotra is currently pursuing masters of technology degree program in Information Technology from University School of Information & Communication Technology, Guru Gobind Singh IndraPrastha University, Delhi, India. E-mail: chhavi16in@gmail.com
- Anuradha is currently at the post of Assistant Professor in University School of Information & Communication Technology, Guru Gobind Singh IndraPrastha University, Delhi, India. E-mail: a_chug@yahoo.co.in

time. Customers work with developers at all time to answer any query or changes that may take place. For every iteration requirements are added.

B. Scrum

Scrum approach has been developed for managing the systems development process. It's an approach applying the ideas of industrial process control theory to systems development resulting in an approach that reintroduces the idea of flexibility, adaptability and productivity. Scrum works on the way in which the team members function in order to produce the system flexibly in constantly changing environment [5 and 6].

The main idea of scrum is that system variables involve several technical and environmental variables (eg. Requirements, time frame, and resources) that is likely to change during a process.

Scrum is a set of guidelines that govern the development process of a project, starting from its design stage to its final completion. It helps to explain some common failures of the development processes such as:

- Chaos due to changing requirements – As we know in agile the requirements keep on changing. The real requirement of project keep on changing over time from when the product is designed to when the product is released. Under most development methods the designing is done at the beginning of the project and no changes take place in the product when requirements change.
- Unrealistic estimates of time, cost, and quality of the product - The project management and the developers tend to underestimate how much time and resources a project takes, and how much functionality can be produced within these constraints. Actually, this usually cannot be accurately predicted at the beginning of the development cycle.
- Developers are forced to lie about the way the project is progressing - When management underestimates the time and cost needed to reach a certain level of quality, the developers must either lie about how much progress has been made on the product, or face the indignation of the management.

i. Roles and responsibilities

Mentioned below is the list of roles of people involved in scrum methodology

- Project Owner

The product/project owner is the voice of stakeholders. They deliver value to business and write a requirement in form of user stories prioritize them and

adds them to product backlog. They are the ones who manage, control the product backlog list.

- Scrum Master

A scrum master ensures that scrum process is used as intended. He/ She enforce the rules and also ensure that product is done smoothly. He /she ensure that the project is carried out according to the practices, values and rules of scrum and it progresses as planned. Scum master interacts with the development team as well as with the customer and the management during the project.

- Scrum Team

Scrum team is the team that has authority to decide on the necessary actions and to organize itself in order to achieve the goals of sprint. The team is involved in estimating the effort and creating and reviewing the Sprint backlog and removing the impediments that need to be removed from the project.

- Developer, Tester, Customer: They serve their respective roles but according to the sprint and work is divided in each iteration.

ii. Artifacts

- Product Backlog

A product backlog is ordered list of requirements that is maintained for the product. It contains backlog items that are kept by product owner based on consideration like risk, business value, and dependencies. The features added to the backlog are commonly written in story format. It can be edited by anyone but the product owner is responsible for ordering the stories. The product backlog contains rough estimates of both business values and development effort; these values are stored in story points. These estimates help the product owner to gauge the timeline and may influence ordering of backlog items. The product backlog, and business value of each listed item is the responsibility of Product owner. The effort that is estimated to complete each backlog item is decided by development team. The team estimates items and User stories either in story points or estimating hours.

- Sprint Backlog

Sprint Backlog is list of items to be used in the next list. These items have been extracted from top priority in product backlog. The extraction process is fairly simple in which the development team ask questions like Can we do this? And adding stories and features in sprint backlog. Same features selected in the previous sprint will not be selected in the sprint for which the selection of items is taking place now.

- Increment

This is the sum of all product backlog items of this sprint and all previous sprints. The increment should

be in usable condition whether or not it is to be re-

leased by the development team.

- Burn Down

The sprint burn down chart is a displayed chart showing remaining work in sprint backlog. It gives an updated view of sprint progress.

2 DIFFERENCE IN XP AND SCRUM

Table 1 shows the differences between scrum and XP. Scrum is ideally a management practice while XP is an engineering practice. As scrum involves various tasks like sprint backlog, release backlog, product backlog which is formed after planning. Tracking is possible with the help of daily burn down charts , review meeting and sprint retrospective meeting. On the other hand XP being totally an engineering activity involving activities like TDD, refactoring , continuous testing, continuous feedback. XP which is more adaptive to change in each iteration is different from scrum wherein changes take place after a sprint ends not in the middle of sprint. In case of XP unit testing takes place while scrum has both integration and system testing carried about in each sprint. Also the unit test cases in case of XP are designed much before than the code is written , based upon which the coding takes place. As XP has activities on iteration basis feedback is essentially quick as against scrum in which the feedback takes place after daily scrum. The people involved in scrum are product owner, scrum master, and team of developers, tester, and designer while in case of XP the team is of developers, tester, analyst and the customer.

XP has following practices which separates it from the ideal scrum model which are, test driven development, pair programming, refactoring

3 THE IXSCRUM MODEL

Agile testing involves the tester to perform testing based upon their knowledge as much of the things have to be assumed in the absence of proper specifications. For this reason “agile testing is not for Dummies” [10]. The integration model proposed is as shown in Fig 1.

Parameter	Scrum	XP
Practice	Management practice	Engineering Practice
Refactoring	NO	YES
Adaptive to changes within iteration	NO	YES
Testing	Integration and System	Unit testing
Feedback	After daily scrum	Quick
Documentation	More	Less
Pair Programming	NO	YES
Test driven development	NO	YES
Project size	Medium to large	Small to medium
Each Iteration is called	Sprint	Iteration
Roles	Scrum master, product owner, scrum team [5]	Team including developers, testers ,analyst, customer
Timeboxes/Meetings	Better defined	Not so much defined
Artifacts	Sprint backlog, product backlog, release burn down chart, sprint burn down chart [5]	Code, user story, release plan, iteration plan , coding standard, unit test , build [3]
Rate of Development	Sprints	40- hour week and small releases [9]

TABLE 1
DIFFERENCE IN SCRUM AND XP

i. Building Product backlog

Initially, alignment of the development team with the business takes place. If it is a single business unit it's easier but if it is multiple units developing different products it becomes difficult to align and group the units together. Proper planning should be done in that case.

Next step in the process will be to nominate the product owner [10]. Unless a product owner is appointed, one should not give a go ahead to scrum as issues and problems may arise later. Decide on a scrum master so that the scrum team can be supported, guided through this process. Then product backlog will be created which is a list of prioritized items form product attributes which needs to be implemented. Items to be listed in product backlog are of some value to the user or customer. It may include functional and non functional requirements. Finally, the product owner prioritizes the backlog items, the priority in no mean decided serially like 1, 2, 3 etc. It's like the items that keep on coming in the product backlog are ordered the way they come in the list. The items in backlog which are down the order are the ones which are lowest in priority so they will at last or may never be considered.

Next step in line will be providing high level estimates which tells about the size of backlog items and giving a perspective about what all items in the backlog are worthwhile. Always the product backlog items should be estimated in points rather than in time. So we are in a way not asking the question how long will it take? We are asking how big it is? [10]. Always the estimates should be built as a team.

ii. Sprint planning / clarify requirements

The next step is to start a sprint planning meeting comprising of the all the individuals involved in the project like analysts, developers, testers. The decision about duration of each sprint is taken which should be unanimous and taken as a team.

Decision about sprint duration is done which is roughly 30 days. Teams with automated testing, deployment can have shorter sprint duration. Also set up sprint goal which is the overall objective of sprint. The items in the backlog are explained by the product owner to the entire sprint team. The team discusses among themselves about the item in question. The outcomes of the discussion should be noted down on a dashboards or diagrams or user stories.

iii. Sprint planning / estimate tasks

The next part of workshop deals in breaking up of requirements into tasks and estimating the hours required to complete it. The meeting is attended by many including testers, developers, analysts while not by the user and customer or business representative as it is more technical in nature. Calculation of sprint budget should take place. Available number of hours the team has to work on sprint. Multiplication of number of hours in sprint duration by the number of people in the sprint takes place. Hours not working comprising of any holiday, any known meeting should be deducted and any everything is kept transparent to the entire team.

iv. The team sprints to achieve the sprint goal

The beauty of scrum is that any methodology adapts here so XP will be adapted here. Testing is integrated throughout the lifecycle. In agile the whole cycle of analyze, design, develop and test takes place from feature to feature basis. Testing starts at the start of the sprint. *Test driven development* starts here Fig 2. The process followed will be designing the tests first, the test should fail as there is no code to run onto, next process will be building the code based upon these tests as our idea is the code developed in this way will be well structured and designed. The process follows for the entire module. Code refactoring will also be required at constant intervals.

A daily stand-up meeting is held in which the whole team is present including the product owner. The team generally stands in half-circle around the sprint whiteboard [10]. This is called *scrum*. Each team member reports back to the team and only one person should speak at a time. Their report should be concise and focused and should address three questions:

What was achieved since last meeting?

What will be achieved before the next meeting?

Is there anything that is holding up the progress?

The scrum master takes responsibility of whole team and removes any impediments which can be placed on the whiteboards. Late arrival of scrum is penalized. A daily burn down chart is prepared, in which a sprint backlog-estimated time to complete (ETC). Each team member is responsible to update their ETCs. The horizontal axis of the burn down chart shows the sprint, the vertical axis shows the amount of work remaining at the start of each sprint. Work remaining can be shown in story points.

v. Review, reflect, and repeat:

A sprint review meeting is held at the end of sprint in which the whole team including business representatives participate. Review of software takes place. A complete demo of working software is performed. The purpose of sprint review is three fold: The team members become aware of the fact what they have achieved and demonstrate their contribution. It also provides an insight to the key stakeholders about the progress of product. It also helps to team to stay focused.

A retrospective meeting is held in which all the team members participate in which all the things that went wrong is addressed. It's a chance to discuss how

things went during the sprint. Together the team reviews the final burn down chart, review the team's velocity. Discussion on what went well, what could have gone better, decide what team will do differently in the next sprint. The entire process is repeated.

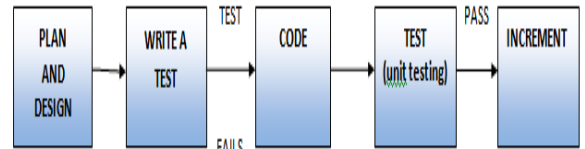


Fig. 1 Each sprint of IXSCRUM Model

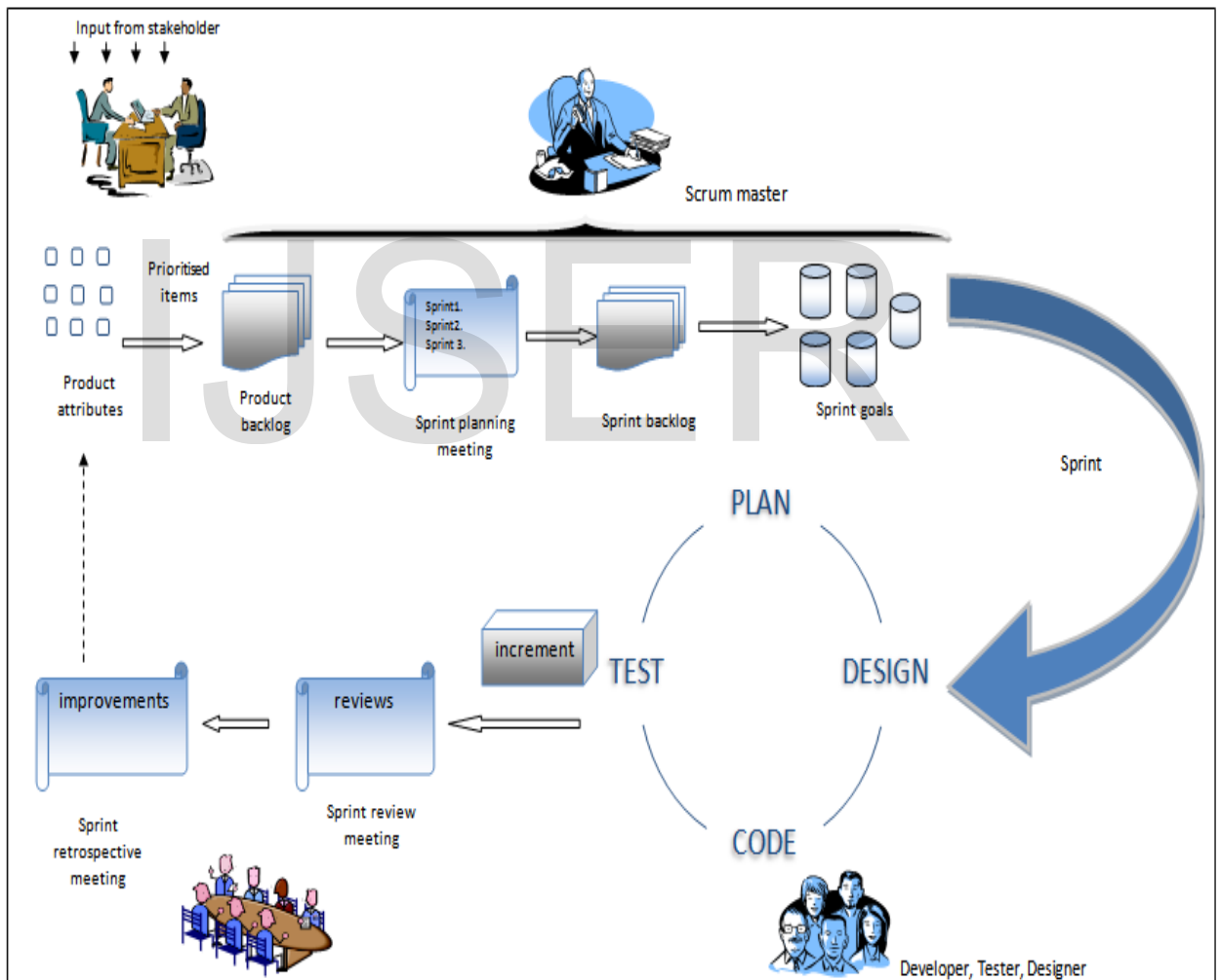


Fig. 2 IXSCRUM Model

4 EXPERIMENTAL SETUP

The setup involved such that all the stages of project development followed a well laid out plan. For this purpose it was very important to carry out the planning activity. The planning activity consisted of developing the plan document comprising of the scope, strategy, approach, timeline, resource involvement of the project. It was also important to decide on the task to be considered from sprint backlog of the project. It was also necessary to bring the product owner, scrum master and the team to be on a same pace on the project. The decision about which product backlog items would be considered in current iteration constructed from a list of product attributes. Also the decision about which items to be constructed from sprint backlog in each iteration was also taken. Formation of sprint backlog from the product backlog was also done [7].

The design was as simple as possible as XP idealizes in keeping the design simple and designing only the features which is in need of the day. Programming only the features which have been designed will lead to system getting stuck. Therefore it was important for a constant maintenance at regular basis. It was also important to reduce the interdependency and that changes in one part not affecting other part, it was important to have constant refactoring by changing the architecture [8].

5 EXECUTING THE PROPOSED MODEL

The work started in iteration by iteration basis wherein it was important to build the control with the help of burn down charts which also helped to carry out tracking activity. It took around 3 iterations to finally get the work done. The application chosen was a Shopping application which involved buying products from the categories like hardware goods, software goods etc. HP's Quick Test Professional was chosen as a testing tool. The team had their work allocated with the help of scheduling activity with a scrum management tool. Testing was intended with the introduction of test driven development wherein it was important to have unit tests built first then the code. It was important to have code built upon the unit tests. These tests failed due to non availability of

code. The coding was then done based upon the finalized design. Again the tests were run which now passed as the code was available. If any of the test failed, code refactoring was done, features were updated and the tests were again repeated with the changed features. The activity helped in producing a quality software at the end which was thoroughly tested and was finished much earlier as it was developed using test driven development approach. Review meeting was held on regular basis with each sprint. This helped the team members to become aware of the fact that they have achieved what they have planned. Retrospective meeting addressed the things that went wrong during the sprint and the changes were suggested, the changes were incorporated and again the whole cycle started until the final product was ready and well approved by the whole team.

TABLE 2

EVALUATION DATA FOR ALL RELEASES

Parameter	Release 1	Release 2	Release 3	Total
Calendar Time(weeks)	2	1	1	4
Number of module	5	2	2	9
Scheduled tasks performed	55	16	9	80
Total allocated effort(hours)	290	40	30	360
Actual hours spent	230	35	20	290
Defects found	7	3	1	11
Number of scenarios	6	2	2	10
Post release defects	2	1	1	5
Sprint retrospective	6	2	1	9

6 CONCLUSION

The implementation of the model was done and it was verified with the help of case study. The model was very well adaptable as it was involving the two methodologies i.e. scrum and XP. It was evident that scrum as being a very popular agile management activity when clubbed with some other engineering activity like XP works very well and is very well suited to current development need of a product. The results of the case study proposed that this integration model enhanced quality of both scrum and XP thus reducing their drawbacks.

REFERENCES

- [1] <http://agilemanifesto.org/principles.html>
- [2] V.Dattatreya, Dr K.V.Chalapati R, V.M. Rayudu "AGILE PROGRAMMING AND DESIGN PATTERNS IN WEB DEVELOPMENT -A CASE STUDY" International Journal of Software Engineering & Applications (IJSEA), Vol.3, No.1, January 2012
- [3] Scott Ambler "Agile Testing Strategies" www.ambysoft.com/scottAmbler.html, 2002
- [4] David Cohen, Mikael Lindvall, and Patricia Costa "An Introduction to Agile Methods" ADVANCES IN COMPUTERS, VOL. 62 © 2004 Elsevier Inc., ISSN: 0065-2458/DOI 10.1016/S0065-2458(03)62001-2
- [5] Ken Schwaber and Jeff Sutherland "Scrum: Developed and sustained", 2008
- [6] Pete Deemer, Gabrielle Benefield, Craig Larman "The Scrum Primer" Version 1.2 ,2010
- [7] Zaigham Mushtaq, M. Rizwan Jameel Qureshi "Novel Hybrid Model: Integrating Scrum and XP" IJ. Information Technology and Computer Science, 2012, 6, 39-44 Published Online June 2012 in MECS (<http://www.mecs-press.org/>) DOI: 10.5815/ijitcs.2012.06.06
- [8] http://en.wikipedia.org/wiki/Extreme_programming
- [9] Marcello Visconti, Curtis R. Cook "An ideal process model for Agile developments" Product Focused Software Process Improvement, 2004 – Springer
- [10] <http://allaboutgile.com>
- [11] Chhavi Malhotra, Anuradha Chug "Agile Testing with Scrum-A Survey" International Journal of Advanced Research in Computer Science and Software Engineering, Volume 3, Issue 3, March 2013 ISSN: 2277 128X