IJCSC Volume 5 • Number 1 March-Sep 2014 pp.23-28 ISSN-0973-7391

Estimating Regression Effort in Agile Environment

Rashmi Popli[1], Priyanka Malhotra[2], Naresh Chauhan[3]
[1]Assistant Professor,Department of Computer Engineering, YMCAUST, Faridabad
[2]M.Tech Scholar,Department of Computer Science, YMCAUST, Faridabad
[3]Professor, Department of Computer Engineering, YMCAUST, Faridabad
rashmimukhija@gmail.com, jayant.malhotra1@gmail.com,nareshchauhan19@gmail.com

Abstract: In the last few years Agile methodologies appeared as a reaction to traditional software development methodologies. The Scrum is the widely used method of Agile. Scrum divides a project in several iterations called as sprints of approximately 24 weeks. The goal of a sprint is typically to develop a potentially shippable product increment. The error free sprint must be delivered to the customer. For this it is required that the sprints needed to be tested again and again for making it completely free from the defects. This lead to the need of regression testing. The rework associated with the regression testing is a big problem so in this work effort and cost associated with the rework are calculated. Also the formula for defect density has been proposed that calculates how the defects massiveness varies with the increasing number of sprints. Effectiveness and feasibility of the proposed formula has been shown by considering a case study.
Keywords- Sprint; Scrum; Rework; Rework Cost; Rework Effort; Defect Density

I. INTRODUCTION

Scrum works in an incremental way in an Agile management mechanism for building software. The actual work for the sprint starts with planning meeting which leads to creation of sprint backlog. The developer works on all the items of the sprint backlog in the sprint and then releases it to the tester, tester tests it and the cycle for testing continues until the sprint is thoroughly tested and is completely error free. Meanwhile testing one sprint other can be ready for development and after that it is also ready for testing, so the second sprint is retested and testing the previous sprint is called as regression testing.

In Section II the life cycle of agile is described. In Section III describes working of scrum. section IV related work in this field is discussed, section V dictates the research problem which is the basis of writing this paper, section VI proposes scenario for regression testing in agile. In Section VII shows evaluation and results of proposed algorithm, section VIII concludes the paper.

II. AGILE LIFE CYCLE

Meaning of Agile is "moving quickly". When applied to software development, it means that delivering the software that meets the customer requirements in shortest possible time. The success and failure of a software project is determined by accurate estimation. This is the process to calculate the time that will be taken to finish the project, cost of the project and the effort required to complete the project.

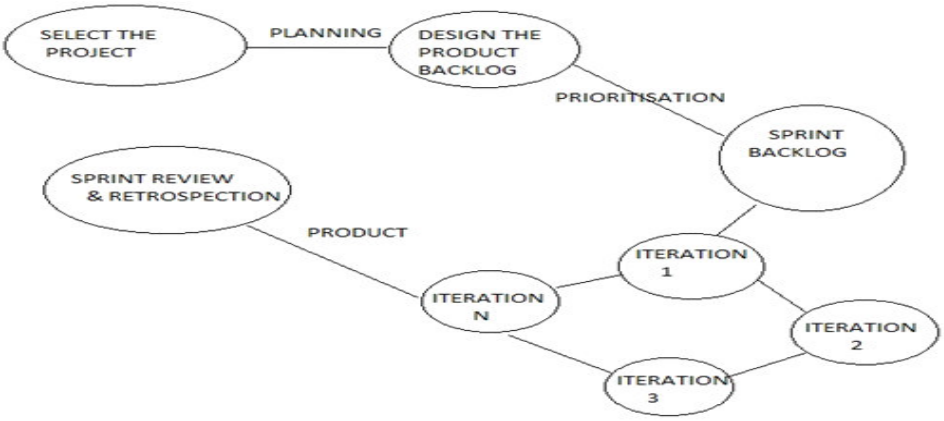


Fig 1:Agile Life Cycle

Estimation is very important task as improper estimation may lead to failure of the software project. It may also increase the budget of the customer and sometimes the nature of the project is also affected [1]. The estimation in the Agile environment is a difficult task due to the changing requirements. The figure 1 shows the Agile software development life cycle.
Regression testing is done in Agile to make sure that the new incorporated changes should not have side effects on the existing functionalities and thereby finds the other related bugs. Thus regression testing may consume much time, cost and effort in the project and therefore it is considered as an expensive process. So there is a need for a technique to calculate regression testing efforts in agile.
Agile software lifecycle is an iterative process where software is ready at each iteration but can always be improved in next iteration. Or in agile terms a part of the project is ready at each iteration and that part itself can be improved at each iteration or can be free from bugs at each iteration.
There are two methods of agile:

1. Extreme programming (xp)
2. Scrum method
3. Crystal Family

III. WORKING OF SCRUM

In scrum the project is divided into small parts known as sprints. The duration of a sprint can range from one to 3 weeks. At the end of sprint the team member, stakeholders and scrum master gather to access the progress of project and identify further plan of action. The assessments helps in taking stalk of the current state and reworks the line of work and complete the project on time and not just speculate and predict the further outcome. In this fig 1 the functions performed by the scrum master are being shown.

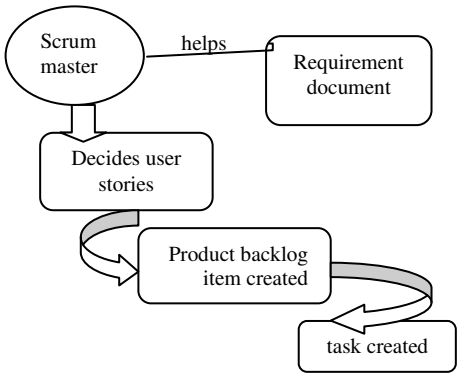


Fig 2:scrum master management of team

In this figure 2 it is being shown that the whole work of agile is dependent on scrum master. On the basis of requirement document the scrum master decides the user stories, which are the baseline of the project. After creating the product backlog item (PBI), tasks are added to the user stories. Product backlog item works as an entry point for the whole project. After creating product backlog item we can add tasks further as per our requirements being updated after regular meetings with the customer.
This whole management of scrum is called as sprint planning. The requirement document describes about all the sprints in a particular project, it also describes which user story will be covered under which sprint.

IV. RELATED WORK

Gregg Rothermel and Mary Jean Harrold proposed the issues relevant to selective retest approaches and presents a framework within which such approaches can be evaluated. This framework is then used to evaluate and compare existing selective retest algorithms. The evaluation reveals strengths and weaknesses of existing methods, and highlights problems that future work in this area should address.
Rashmi popli[2] proposed an algorithmic method for cost and effort estimation in Agile by considering project as well as people-related factors like quality requirements, complexity, transaction etc.

Michael Ruth and Shengru Tu [5] proposed a control flow graph-based approach that makes it possible to apply a safe RTS technique to Web services in an end-to-end manner. Safe RTS techniques ensure that no modification revealing tests will be left unselected.

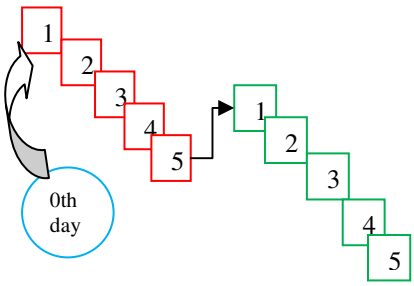
Dharmendra **S.** Modha[9] , proposed abstract minimum complexity regression estimators for dependent observations,which may be adapted to a particular list of parametric models, and establish upper bounds on the statistical risks of the proposed estimators in terms of certain deterministic indices of resolvability. Assuming that the regression function satisfies a certain Fourier-transform-type representation, it was examined that minimum complexity regression estimators adapted to a list of parametric models based on neural networks and, by using the upper bounds for the abstract estimators, and rates of convergence for the statistical risks of these estimators were established.

V. RESEARCH PROBLEM

The tester has to test the first sprint when development for first sprint is under work , and when second sprint is ready , it may happen that the first sprint is still not free from all the defects , so it needs to be retested again , this retesting is called as regression testing . The testing of the sprints again and again even if the developer has done some new silly mistakes while removing the previous ones is called rework. This rework has no exact formula for calculation of its effort and cost .so this paper proposes formula for rework cost and effort, along with the defect density and project efficiency matrix.

VI. PROPOSED WORK

When planning about first sprint, at least 80% of the backlog items are estimated to build a reasonable project map. These backlog items consist of user-stories grouped in sprints. But user-stories based estimation is done using hours, days or weeks which are also including other activities.



1. 1st , 2nd 3rd and 4th day of each sprint the process of development is done.
2. On 5th day the first sprint is released to tester for testing .
3.Regression testing is started for the previous sprint and rework effort is calculated

0th day of a particular sprint means the first day when sprint planning is done, and also the meetings with developer stake holder , tester and scrum master .

1. On 1st day development is done for 2nd sprint and testing for previous sprint.
2. On 2nd day also same thing happens
3. On 3rd day bug fixing is done for 2nd sprint.
4. On 4th day bug fixing for 2nd sprint and released to tester for re testing .
5. On 5th day retesting for 2nd sprint and regression testing for previous

On each 5th day same thing which was done on 0th day is repeated for next sprint, and the work done in previous sprint is handed over to tester.

Fig3: Proposed scenario of regression testing

.

When a team member estimates that a given task can be completed within 8 hours it does not mean that he can complete the task in one day. Because no one can sit in one place for the whole day. So the concept of Sprint-point

is proposed. A Sprint-point basically calculates the effective time which a team member spends in Sprint-related work in that iteration.

Effective time per day for Sprint Related work = Average work day time - Time allocated for other activities. For example Average work day time = 10 hours
Time Allocated for other activities = 5 hours
Emails and Phone: 1 Hrs
Lunch: 1 Hrs
Meetings: 2 Hrs
Bug fixes: 1 hrs
Available time for Sprint Related work = 5 hours

1. **Rework effort=** total number of defects /defect fixing effort(DFE)
2. **Defect Fixing Effort(DFE)=** no of defects fixed per hour*no of working hours per day *no of persons working
3. **Rework Cost**=rework effort*cost of one employee*no of employee.
4. **Defect Density of one iteration**=defect in that iteration / total story points that iteration is covering.
5. **Project efficiency=** defects fixed before regression testing/defects fixed after regression testing defects fixed after regression testing.

The proposed scenario of regression testing in agile shows that the iterations in agile are of short duration, the very first day is fixed for sprint planning, decision making and for meetings with the developer and tester. Thereafter the actual work starts, the development takes place for one sprint in first iteration and in second iteration it is sent to the tester for testing work. The tester tests the first sprint meanwhile the developer starts his work on the second sprint, after testing the first sprint is ready for the retesting and second sprint is ready for the testing and along with third sprint is under development. This process continues till all the sprints are developed and are completely tested.

Proposed Algorithm

- Scrum master decides user stories $u_1, u_2, u_3 \ldots\ldots u_n$ on the basis of requirements described in requirement document.
- Assign story points to each user story.
- Assign sprint number to each user story.
- Calculate total defects in each sprint after performing regression testing
- Calculate total defects for all sprints.
- Find out Defect fixed per hour .
- Defect fixing effort(DFE)=no of effective working hours per day*number of employees working*defects fixed per hour.
- Rework effort=total number of defects/defect fixing effort(DFE)

Proposed Regression Testing Activity diagram

The estimation diagram explains the various steps involved in estimating rework effort in regression testing in agile environment as shown in figure 3.

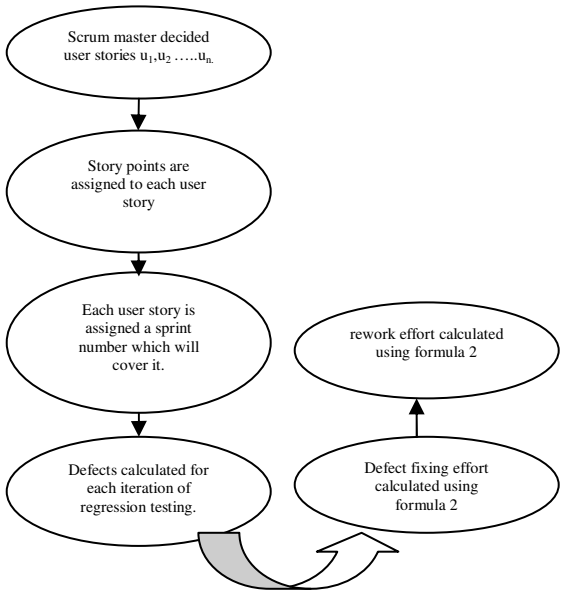


Figure 2: Proposed regression testing in scrum

VII EVALUATION AND RESULTS

In this section the feasibility of our algorithm is shown by calculating the estimated values of rework effort, rework Cost, defect density and project efficiency for a small project. We had considered the number of effective working hours as 5 per day by using [10] .In this section the feasibility of our algorithm is shown by a case study in which the values for rework cost, rework effort and defect density order is calculated. We have considered the user stories of enable quiz which is a **technical quizzing solution**; for companies that hire engineers and it will allow them to better screen job employees and assess their internal talent for skills development. A graph can be drawn taking the defect density values against the sprint values , which will show the defects amount distribution in overall project.

Table 1: **Showing user stories of the project under taken and related story points and sprint number**

Serial no	User story	No of Story points in user story	Sprint covering that user story
1	As a manager, I want to show all the existing quiz questions	4	1
2	As a manager, I should be sure that I'm subscribed to all the related topics of my skills	5	2
3	As a manager, I can add more questions and topics to my quizzes.	7	1
4	As a manager, I can create a customized questionnaire	4	2
5	As a manager, I can share quiz with my staff	8	1
6	As a manager, I can create a list of students for all the related topics	9	2
7	As a manager, the quiz can be made online to students	4	4
8	As a manager, I can invite foreign university and student	6	3
9	As a manager, I can see the quiz to check students	4	4
10	As a manager, I can start a skill improvement program	5	3

.

Table 2: summing up the table 1 and calculation of defect density for each sprint.

Sprint no	User stories covered in that sprint	No of defects found after regression	Total story points covered	Defect density for that sprint
1	1,3,5	0	19	0
2	2,4,6	20	18	1.01
3	8,10	37	11	3.36
4	7,9	44	8	5.5

Table 3: Exact statistics of the project

bug fixed	2
no of testers	2
no of developers	1
Scrum manager	1
defect fixing effort	40 person per days
Rework effort	2.5days per person

Numerical Analysis

Total number of employees working=4

Defect fixing effort=no of persons*working hours per day*defects fixed per hour=4*5*2

Rework effort= total defects/defect fixing effort=101/40=2.5 days per person.

VIII. CONCLUSION

In this paper, for exact calculation of the rework in regression testing in the agile environment the formula for rework effort is proposed. Estimated value of rework will highly affect the working of the tester and he will work more smartly. The formula for the defect density describes the exact distribution of the defects in the project and it can guide tester how he should perform his work. After having exact values it would be easier for the manger to read out the reports because the work is shifted from the theoretical portion to the exact numerical data.

REFERENCES

[1] Gary Gack "Powerful Metrics: Software Cost of Quality + Defect Containment" Published in ASQ Software Quality Professional, March, 2011

[2] Rashmi Popli "An Agile Software Estimation Technique based on Regression Testing Efforts" 13th Annual International Software Testing Conference in India 04 – 05 December 2013, Bangalore, India [3] Mike Cohn,

[4] Per Runeson and Mats Skoglund A Case Study of The Class Firewall Regression Test Selection Technique on a Large Scale Distributed Software System

[5] Michael ruth and Shengru Tu**. "**A Safe Regression Test Selection Technique for Web Services"

[6] David Willmor and Suzanne M. Embury "A safe regression test selection technique for database–driven applications" School of Computer Science, University of Manchester, Oxford Road, Manchester, M13 9PL, United Kingdom

[7] Gregg Rothermel, Kent Sayre, Filippos I. Vokolos, Phyllis G. Frankl "An Empirical Comparison of Two Safe Regression Test Selection Techniques."

[8] Jamieson, D., K. Vinsen, G. Callender, "Agile Procurement to Support Agile Software Development, Proceedings of the 35th IEEE International Conference on Industrial Informatics,2005.

[9] Dharmendra **S.** Modha, **Member,** IEEE, and Elias Masry, Fellow, IEEE "Minimum Complexity Regression Estimation with Weakly Dependent Observations" IEEE TRANSACTIONS ON INFORMATION THEORY, VOL. 42, NO. 6, NOVEMBER 1996 2133

[10]Rashmi Popli, Naresh Chauhan," **"**Scrum- An Agile Framework", International Journal of Information Technology and Knowledge Management (IJITKM) ISSN: 0973-4414", Vol-IV, Number-I, 20 Aug 2010**.**