Volume 5, Issue 9, September 2015 ISSN: 2277 128X
International Journal of Advanced Research in Computer Science and Software Engineering
Research Paper
Available online at: www.ijarcsse.com

Analysis of Scrum Based Mutation Testing for Safety Critical Projects

Fathima M M.E.
Assistant Professor
Department of CSE
Sri Krishna College of Technology
Kovaipudur, Tamil Nadu, India

Uthra V
UG Student
Department of CSE
Sri Krishna College of Technology
Kovaipudur, Tamil Nadu, India

Abstract- The Agile methods are the evolving strategies for real time projects in industries and research sectors. The agile models combine conventional methods with random approaches to provide high performance applications. Every level of improvement in the product is considered as an iteration which is then made into a complete system. The agile methodology is distinct from other process models because it welcomes changing requirements, it has a self-organizing team where attention is given for technical calculation and good design. The best feature about Agile is its adaptability under unexpected expectations from stakeholders. Agile promotes the success of any project by ensuring abstraction and openness. This paper reviews all agile methodologies like XP, Scrum, Crystal, FDD, ASD, DSDM and Agile modeling to determine their specific purposes for real time applications. Testing has a high probability of finding errors. The recently used mutation testing specifies the common and uncommon mistakes made by the programmers. This paper also analyses the concepts and benefits of mutation testing and then combines the strategy of Scrum along with mutation testing for its efficient use in building safety critical projects.

Key words- Extreme programming, Feature Driven Development, Adaptive Software Development, Dynamic Software Development Model, agile methodologies, Mutation testing.

I. INTRODUCTION

Software Engineering is the key to the advancements in research and industry based projects [5]. The foundation of a complete software process regardless of their size or complexity is coupled with disciplined structure and integration of varying business components. [10]. Software engineers play different roles with respect to the needs that arise while adopting a perspective approach. Recently, a modern software engineering approach called 'agile' is brought to light for impartment of products with customer satisfaction [10]. Agile methods demand working models as a measurement of progress made by the self organizing teams. To attain sustainable development, effective agile methods can be combined with varying testing procedures for procuring competent safety critical applications.

II. BACKGROUND OF THE STUDY

The conventional process models are Waterfall, Prototyping, Incremental, Spiral, Win-Win spiral, Object oriented and Rapid application development.

1. Traditional Process models

The Waterfall Model is a heavy, micro managed approach that lacks a formative environment leading to negative significance within the team [5]. Though the process is unit tested, documented sequentially and patches are released to fix bugs, high amount of risks and uncertainty are not handled for major projects. However, prototyping model is mainly used for interaction based designs involving customers and allows them to define additional requirements,

It induces poor time, cost estimation and high developmental costs [6]. Similarly, incremental model undergoes testing for each increment [6]. But it creates complexity in the interface and the development cost is higher as the testing causes high cost.

Spiral model which is famous and effective in reducing high priority risks by blending the strengths of earlier models ensures the software is reusable yet it is only suitable for larger projects and it needs experts to manage the software [8]. Win - win model extends the spiral model enhancing the importance in negotiation. Though both the developer and the clients are satisfied, meeting deadline is a great challenge and also if the stake holder is not satisfied then the entire work is in vain [9]. Object Modeling encourages re-use of the code, but it does not show efficiency in reducing the errors also the development cost is higher [8].

Rapid modeling is a business modeling approach that focuses mainly on business. It encourages rapid or faster release of the software hence includes Rapid Fire activities [7]. The application code generated has value so that it can be re-used for new existing real time applications. But it requires higher man power .Maintenance and performance is not as

efficient as the other models. Based on the study made on all the conventional process models, the table 1 is given to show the pros and cons of the software models, taking the important criteria that has to be taken into account while building software.

Table 1. Comparison of basic models

Models	Feedback	Cost	Quality	Time	Error reduction
Waterfall	-	✓	✓	-	-
Prototyping	✓	-	-	-	✓
Incremental	✓	-	✓	-	✓
Spiral	-	-	✓	-	✓
Win-win	✓	-	✓	-	✓
Object model	✓	-	-	-	✓
Rapid model	✓	-	-	✓	-

2. Agile principles

The agile methodology overcomes all the disadvantages of the conventional process models. The agile methodology was formulated by Kent Beck. It is a self organizing team with the highest priority of welcoming changes and obtaining customer satisfaction. Agile promotes continuous development [10]. The agile methodologies are listed as follows.

In XP, user stories are considered important because the check points are used to welcome and accept the changing requirements from customers [11]. XP uses automated Unit Tests and the pair programmers try to break the code to improve efficacy. The ability and expertise in technical aspects enables the team to develop solutions which matches the customer's desires. XP projects become hectic and lead to vanity in the long run if the planning is improper. So the system may become complex and projects could come to a halt.

3. Scrum

Scrum is a holistic passage that made a major transition from basic models to unconventional fashion in software product development by accelerating the performance in deploying various real time services by assuring a rapport with the customers as part of business relations [12]. To make certain of this, scrum meetings are conducted habitually under the leadership of the skilled agile engineers [13]. Events are planned so as to keep track of the desired time duration. Priorities are set with respect to the emphasis given for the particular stages that leads to the expansion of the product without violation of time limit. Sometimes numerous teams may work on the single product. To avoid miscommunication and overlap, regular meetings are organized to set out synchronization [10]. The product stock pile is maintained to achieve cross team systematization. It also holds the details of the efforts made by the core team and the computational business values made by the client. After examining multiplex agile models, scrum always provide better outcomes for profitable (industry) and non-profitable (safety critical) ventures.

4. Other Agile models

Feature driven development (FDD) project analyzes the scope of the project along with its context [14]. Depending on the subject areas, domains are broken down to identify list of features. Since it highlights on features of the project alone, it fails to focus on other functionalities. Crystal is a process in which different projects with different process [16]. It is mainly used for gaming applications but fails for the development of safety concerned projects. Massive projects involve more people and lead to lack of better communication among those people [16] resulting tediousness in maintenance and testing. Adaptive Software Development (ASD) engulfs speculation, learning and collaboration [14]. Assumptions are made to discover mistakes in the domain and corrections are made from the knowledge obtained through iterations. The problem arises with increased team size or project size.

Dynamic Software Developmental Model (DSDM) is used to construct time constraints systems, Uses incremental prototyping and it is an Iterative process [15]. Design and build iteration revisits prototype and provides operational business value [14]. In implementation cycle latest software increment operation is done, it may not be 100% as changes may occur. Agile modeling is a practice based method .It is mainly used for the effective implementation of software system [17]. The principles of this system are unique as the methodology followed need not be perfect. It is a model developed with a purpose as the content is more important than representation. It adapts locally but is more documents oriented.

5. Testing

Testing is done to uncover the errors in the source code .The primary goal is to fix bugs without exhaustive testing.

6. Random testing

Independent inputs are used to test the programs randomly [3].The data are selected randomly to verify the test output. It is better than manual testing but can only find basic bugs. Comparison of the test output is done to ensure that the specifications are exact. But generally if the specifications are improper so does the results. Though this testing is cheap it doesn't carry much advantage.

7. Agile testing

Agile testing is done to consign the business value desired by the customer at frequent intervals [4]. This testing also identifies undesired behaviour, monitors each feature to improvise the coding. Testing challenges still arise from the changes in the definition of the product and its correct behaviour. The test team should be kept informed of the changes; otherwise this change produces testing inefficiencies.

8. Mutation testing

Mutation testing is used to test the test cases in order to locate the shortcomings that occur in the test suites [2]. It is specifically applied to procedural and non procedural languages. The unambiguous parts of the source code are selected where the live mutants are introduced and tested. Concurrently, the mutants can be categorized as equivalent mutants; non equivalent mutants or dead mutants based on their survival in the test process [1]. Although the mutation testing is costlier many industries apply this testing strategy. It is mature testing strategy that has proven to be the best for software and software testing techniques.

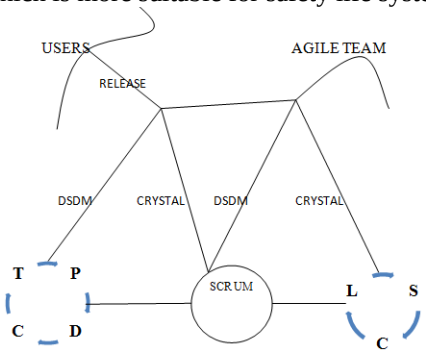
III. CONTRIBUTIONS

Table 2. Comparison of agile models

PARAMETERS	SAFETY LIFE PROJECTS	MISSION PROJECTS	DEFENSE PROJECTS	INDUSTRIAL PROJECTS
CHECK FOR ERRORS	SCRUM	XP	ASD	XP,SCRUM
SUCCESS RATE	ASD	DSDM	AM	XP,SCRUM
COST EFFECTIVE	FDD	DSDM,ASD	DSDM,ASD	FDD
REQUIREMENTS	CRYSTAL	XP	ASD	DSDM
FEEDBACK	ASD,XP	XP	ASD	SCRUM

XP-EXTREME PROGRAMMING
ASD-ADAPTIVE SOFTWARE DEVELOPMENT
DSDM-DYNAMIC SOFTWARE DEVELOPMENT MODEL
AM-AGILE MODELLING
FDD-FEATURE DRIVEN DEVELOPMENT

From the above table 2. we say that agile methodologies are proven to be effective for constructing time boxed systems, XP, Scrum, ASD is more effective than other agile methodologies in terms of customer satisfaction, welcoming the changes in the requirements and in reducing the errors. Mission and defense oriented projects use hard real time operating systems which will not accept even a minutes' delay, DSDM is effective for time constraint systems.FDD is good for moderate and large projects in cost wise. Crystal is a resource limited methodology that aids a co-operative game of communication and invention which is more suitable for safety life system projects



T-Testing P-Planning D-Designing C-Coding…..>XP
L-Leaning S-Speculation C-Collaboration……>ASD

Fig 1. Agile methodologies

1. Scrum and Mutation testing

This paper approaches a strategy that combines scrum and mutation testing due to the following reasons.
In Scrum, quality and functionality come hand in hand and progressively reduces time and budget. Through incremental implementation of functionality, new emerging requirements are added at the beginning of each sprint. Then with proper planning, scrum technique perceives and culminates with a formal review. To remain flexible and to control unnecessary changes without exceeding the cost constraints, incremental approach is used here. In turn, it avoids jeopardizing large chunks of extraneous works [12]. To maximize the efficiency at the end of daily sprints, communication with customers is conducted. This elasticity applies to the operational processes itself.
Mutation testing provides a complementary course of action for measuring the sufficient test data, but industry wide reach ability for safety critical systems is not supported [2]A survey on mutation testing was conducted to check its applicability in airborne software systems. The scrutinized data satisfied the coverage requirements. Using high integrity

subsets of data, we have employed mutation testing to safety-critical software. It then identifies valid mutant types, and inquires the major reasons for failures in test cases. The case study stated that mutation testing is effective in such projects where traditionally used manual reassessment process have failed enormously.

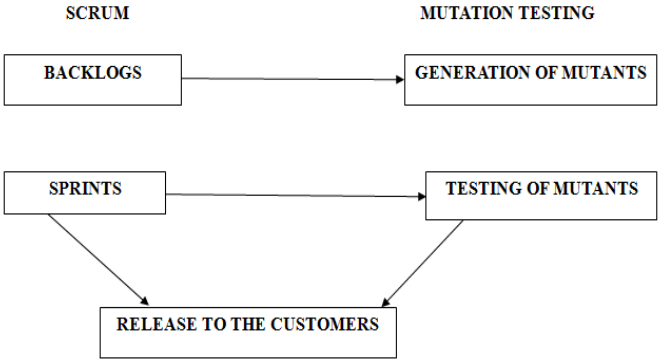


Fig 2. Block diagram

From the figure 2.,In Scrum the backlogs is created by the scrum master, and the sprints are the requirements needed for the backlogs, the backlogs and the sprints are governed during the scrum meeting. Based on the extensive study, the generation of test mutants in mutation testing is related to the creation of backlogs and the testing of mutants is similar to the sprints. In both scrum and mutation testing, delivery is never made without completing or evaluating the features in the final versions of developed product.

IV. PERFORMANCE EVALUATION

Based on the study regarding mutation testing which was applied to safety critical project air borne system software C, some mutants are still born mutants which are illegal and caused because of failed program compilation. Sometimes the initial state of live mutants cannot be determined and it affects the output of the program. These are called equivalent mutants [1]. Defects in the test are identified effectively by non-equivalent mutants. The source code of the C system was divided into code items, lines of code and the cyclomatic complexity. The generation phase was divided into the total mutants generated and the syntactically incorrect mutants [2]. The test execution was based on the total number of the mutants re-tested and the number of mutants that were not killed.

The final analysis was based on the mutation score which is obtained from the number of equivalent mutants. Simultaneously, the number of mutants that survived was also obtained from the above analysis. Mostly, in complex projects it is found that even a single error in one test cannot make a profound impact on the mutation score.

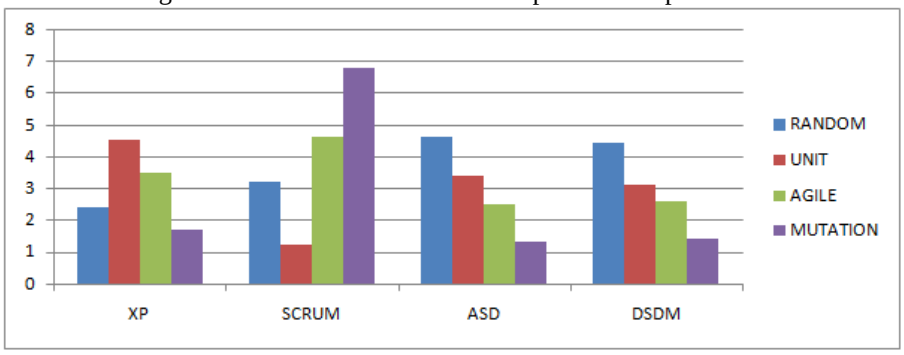


Fig 3. Comparison graph

From the above comparison graph 3. It is clear that XP uses unit testing at its best while agile testing suits XP model also. But it performs poorly using mutation testing. Similarly both ASD and DSDM perform better using random testing but are still poor in performance regarding mutation testing. Scrum is the only methodology that performs well under mutation testing. Thus it is clear through analysis that Scrum along with mutation testing is best adaptable to the safety critical projects.

V. CONCLUSION

This paper has analysed the various traditional and agile process models that are widely used in the real time projects. Every successful development and delivery of product entirely depends on testing. So several testing strategies such as random, agile and mutation testing are compared and based on the implementation, detection of bugs and fixing the errors, they are categorized for specific projects. As a result of the analysis, Scrum model and mutation testing are compatible to generate test cases in unexpected real time scenarios for better performance. Thus Scrum based mutation testing can be used for safety critical projects. There are several areas for further work.

Several mutation data can be obtained from other real time projects namely mission critical applications. The mutation data can be used to evaluate various test cases, coverage levels and their influence on test quality. Analyzing test results and determining equivalent mutant behaviour is still a manual overhead and therefore requires further investigation.

REFERENCES

[1] An Empirical Evaluation of Mutation Testing for Improving the Test Quality of Safety-Critical Software Richard Baker and Ibrahim Habli

[2] Mutation Testing Cost Reduction Techniques: A Survey Macario Polo Usaola and Pedro Reales Mateo, University of Castilla-La Mancha

[3] Formal Analysis of the Probability of Interaction Fault Detection Using Random Testing Andrea Arcuri, Member, IEEE, and Lionel Briand, Fellow, IEEE Lech Made ski, Member, IEEE, Wojciech Orzeszyna, Member, IEEE, Richard Torkar, Member, IEEE, and Mariusz Jozala

[4] Agile testing challenges by Xui Len, International Conference on Software Engineering, 2000.

[5] A Waterfall Model to Achieve Energy Efficient Tasks Mapping for Large Scale GPU Clusters Wenjie Liu Dept. of Computer. Sci. & Technol., Tsinghai Univ., Beijing, China Zhou Du ; Yu Xiao ; Bader, D.A. ; Chen Due

[6] Research on the virtual prototyping technology of the electric-hydraulic-mechanical system based on Syncope Yadong Mange; Yi Fang; Chinghai Liu Advanced Computer Theory and Engineering (ICACTE), 2010 3rd International Conference on Year: 2010, Volume: 1

[7] Telecontrol of rapid prototyping machine via Internet for automated telemanufacturing Lou, R.C.; Wei Zen Lee; Jim; Hoe Robotics and Automation, 1999. Proceedings. 1999 IEEE International Conference on Year: 1999, Volume: 3

[8] A collaborative spiral software process model based on Theory W Boehm, B.; Bose, P.

[9] A Win-Win Multi-issue Service Negotiation Model Based on Bayesian Learning Min Li; Li Pan; Shijun Liu; Lei Wu Cloud Computing and Big Data (CloudCom-Asia), 2013 International Conference on Year: 2013 Tung, L.; Ramirez, R.G.; St.Louis, R.D. System Sciences, 1991. Proceedings of the Twenty-Fourth Annual Hawaii International Conference on Year: 1991, Volume: iii

[9] Security model consistency in secure object-oriented systems Keefe, T.F.; Tsai, W.T. Computer Security Applications Conference, 1989. Fifth Annual

[10] Application of agile method in the enterprise website backstage management system: Practices for extreme programming Lingui Liu; Yao Lu Consumer Electronics, Communications and Networks (CECNet), 2012 2nd International Conference on Year: 2012

[11] Extreme programming: a university team design experience Kevin, J.; Haydon, D.; Hayes, J.; Schneider, R.; Succi, G. Electrical and Computer Engineering, 2000 Canadian Conference on Year: 2000, Volume: 2

[12] Practical Scrum-Scrum Team: Way to Produce Successful and Quality Software Mandrel, A.; Mira, S.; Dawdle, C.A.on Year: 2013

[13] Scrum But, But Does it Matter? A Mixed-Method Study of the Planning Process of a Multi-team Scrum Organization Heikkila, V.T.; Paasivaara, M.; Lassenius, C. Year: 2013

[14] Comparison between Adaptive Software Development and Feature Driven Development Chowdhury, A.F.; Huda, M.N.Computer Science and Network Technology (ICCSNT), 2011 International Conference on Year: 2011, Volume: 1

[15] An extended adaptive software development process model Sadiq, M.; Hassan, T.Issues and Challenges in Intelligent Computing Techniques (ICICT), 2014 International Conference on Year: 2014

[16] A crystal model of knowledge based application design Perot, F.C. Developing and Managing Expert System Programs, 1991., Proceedings of the IEEE/ACM International Conference on Year: 1991

[17] Novel lightweight engineering artifacts for modelling non-functional requirements in agile processes Farid, W.M.; Mitropoulos, F.J. Southeast on, 2012 Proceedings of IEEE Year: 2012

