

Machine Learning Based Software Effort Estimation of Suggestive Agile and Scrumban Methodologies

B V Srikanth¹, Dr. P.V.Bhaskar Reddy²

^{1,2} School of Computer Science and Engineering, Reva University, Bangalore, India

¹ r21pcs30@reva.edu.in, ² bhaskarreddy.pv@reva.edu.in

Abstract— The dynamic IT business and industries are welcoming the customer software requirements to adopt with new changes as per needs. With the recent advancement of technology in this era, software industries have grown outrageously. Software industry has shown such great hike in technology which is non-comparable to any other industries. Various methods have been established which improves the software quality one such method is Agile. Agile software development has gained a lot of attention because of its simplicity and ease of use. Agile software development is an approach which produces quality software with remarkable team interaction and more of customer involvement. Agile method is basically ideally suited for a scenario where requirements are changing in continuous manner. One of the most important advantages of using Agile is, it takes less time for software release, easy to understand and require less documentation. This research deals with various agile methods, their comparison, advantages, shortcomings and suggestive SCRUMBAN, a new proposed framework to overcome those mentioned shortcomings.

Keywords— SCRUMBAN, SRCUM, Extreme Programming, Feature Driven Development, Crystal, Adaptive Software development, Dynamic System Development Method.

Introduction

Agile provides an organization with the acceptance of welcoming set of methods, adoptions of various parameters and the software products those choose to design. Agile implementation became most reputed. By addressing the drawbacks in normal implementation, initiating better versed in procedures and quality, agile introduced. The agile methods is to for the client and customer for speedy developments for better inputs, accepting, producing more results.

The agile provides a company, the feasibility to accept a subset of principles based on their needs and requirements. Agile is to produce higher quality in less time. Agile streamlines the procedure and always welcomes the complexities to adopt project needy changes in the code at any time. Agile methodologies for the customer needs and develop needs be assigned for the duration of code developing period. In the code writing time, the problem assumed which is resolved while using a collecting major aspects of different models, this model is hybrid model.

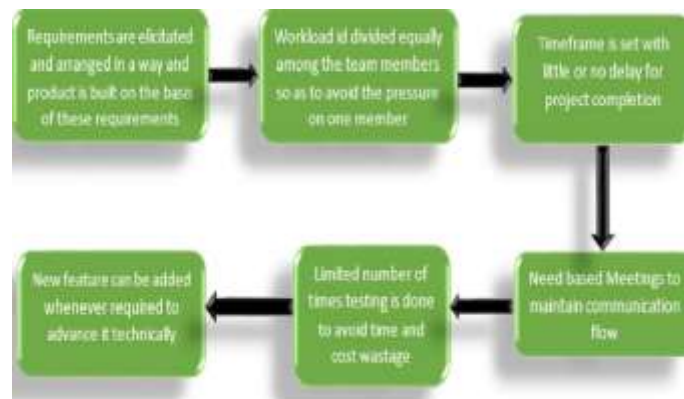


Fig.1. Proposed Research framework on SCRUMBAN

- The policy design phase is the defining of research oriented objectives belongs to an agile Expecting efficiency.
- RQ#1: which is implemented for efficiently proved in Agile?
- RQ#2: Working principle for implement efforts are estimated effort in ?
- RQ#3: The parameters which are Considered in the process?
- RQ#4: Extent that are in large-scale agile principles are studied empirically in the literature?
- RQ#5: The level of abstraction those are principles, methods, tools, and metrics are the major considering scale agile methods which are identified in RQ4 the resulting knowledge gaps?
- RQ#6: The challenges those are reported for the large-scale agile methods found in RQ4?
- RQ#7: The success factors found in RQ4?

II. Background of Research

- Mostly, projects not successful because poor monitoring hence it was found that defined project management is used, that is Implementation of an agile re development. Several process have been practiced from the invention of an agile agenda with various used for success.
- Development thru iteration of agility methods and aspects and includes the exacting methods from needed three variations – Adorns, ability, efficiency. process is implemented to expectation of agile principles. Validated is a goal process of framework.

The high efficient of the AM is to implement that have been related to the factors, train the customers, ability, perception of company deviated scheduling. Some of agility are Scrum, XP, Kanban, LSD, FDD, AUP, and DSDM.

The AGILITY will project management, all cycles in the project, team, engineering timely delivery. Example is DSDM gives all disciplines .the success factors of the agile are organizational, people, process, technical and application factors.

This estimation is a mix of resource with given time, is the number of keenly efficient working time work done. The efficient units are decided in resource calendar hours. Agile is not same as the traditional model.

An agile that focus on delivery of projects, effort of hourly work efficiency use to find the number of iterations to expect the schedule and delivery date. The software focusses for finding the project cost. This research aim is to review the methodologies and Approaches of software cost of work efficiently, to make sure the most useful Agility.

• **AGILE METHODOLOGIES**

The below is the list of main agile methodologies with their definitions and area of focus during applicability:

Scrum: The agile scrum methodology promotes more cross functional and co-location teams to ensure effective cooperation in teams that are working on complex products. The sprint period can be changed based on stakeholder’s assessment to meet working product time. However, the shorter is the sprint the better to ensure progress review.

Kanban: Kanban used to use Agility and Development Operations in the project. It needs transparency in coordination work. The main focus of Kanban is visualize work ensure work in progress limit based on team capacity to avoid too tasks in progress mode.

I. **Extreme Programming (XP):** This methodology advocates for frequent releases in short development cycles, which is intended to improve productivity. In additional to frequent releases, the methodology advocates on engineering best practices such pair programming, code review, test driven development, refactoring, etc.

II. **Lean Development:**The lean development advocates mainly on providing prototypes as minimum then when customers or users appreciate then continue improving the product.

III. **Crystal:**The team interaction, communication, transparency, accountability is key for crystal methodology.

IV. **Dynamic System Development Method (DSDM):** The method that focuses on the full project lifecycle.

V. **Feature Driven Development (FDD):** This is to organize software development around making progress on features.

Kanban:

Kanban is an ASD is signaling device that teaches the flowing as ‘pull’ in the development. Kanban gives workflow steps on the whiteboard.

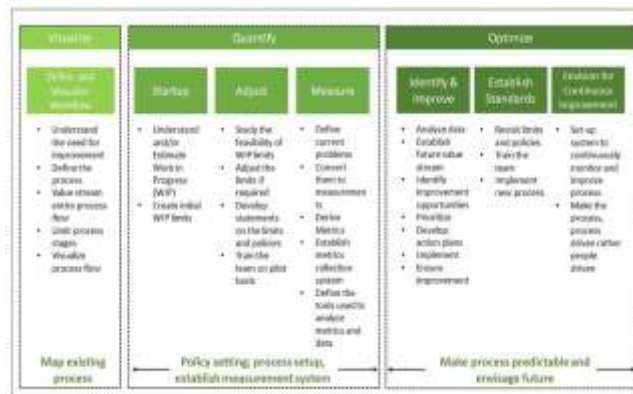


Fig. 2 Kanban Approach

The aim of the Kanban is to work activities move to the next level vastly to do business. This is to from evolution to incremental and process changes in the companies. The progress of work pull system is

used as the method to unfolded system the cooperation for best system. Kanban used in Jira, SwiftKanban, Lean Kit Kanban, etc. Tools.

The following are the six core Kanban practices:

1. Limit WIP
2. Visualize
3. Manage flow
4. Improve Collaboratively, Evolve Experimentally
5. Implement Feedback Loops
6. Make Policies Explicit

Critical Review of Literatures and identification of Research gaps

Abrahamsson et al. suggested a technique for effort prediction using user stories. They proposed that this technique can be applied to Agile software project estimation as in Agile the requirements are developed iteratively. The proposed technique was applied to two industrial Agile software projects and it was concluded that effort estimation is accurate for well-structured user stories. The study identified that estimation becomes harder for Agile methods as requirements are not completely specified at the beginning or before the start of the development. To resolve these issues this research suggested a novel method for effort prediction based only on predictors automatically extracted from user stories. Usman et al. performed a systematic review of existing literature where total 25 studies were analyzed and the main observations were regarding the estimation techniques, size metrics, accuracy metrics and the cost drivers used in existing techniques. Wen et al. Conducted a systematic review of existing literature comprising of empirical studies based on various soft computing models published in 2011-2021. The work analyzed machine learning models on various factors like Machine Learning model used, estimation accuracy, selection of model and context of estimation. This research analyzed 84 primary publication and found that eight ML models are used in Software Development Effort Estimation. Further it was concluded regarding the prediction accuracy that accuracy of ML models outperformed the basic models. C. Lopez Martin discussed the issues which arise when machine learning models are applied for predicting the development effort and compared the accuracy of prediction of various neural network models with statistical regression model. The dataset used was published benchmarking dataset which included function point as independent variable depicting the size of project and actual effort. The results in the research showed that the estimation accuracy of suggested model based on neural network is better as compared to the accuracy of the mathematical model based on Regression. S. Dragicevic, S. Celar, M. Turic proposed an effort estimation model based on Bayesian network model for projects specifically developed in Agile. This research uses data of projects developed by a software company and assesses the prediction accuracy of proposed model with the statistics like Mean Magnitude of Relative Error, Mean Absolute Error and prediction accuracy. The observations indicate a very good prediction accuracy. J. Moeyersoms indicated that the prediction model must be accurate as well as comprehensible i.e., it should be easy to understand. In this research, software faults and effort are predicted by applying various data mining techniques. Rule extraction is applied and tree structure of random forest is used and regression of SVR is used to predict the fault and effort. P.C. Pendharkar et al. used Bayesian Network Model for prediction of software development effort and compared its performance with other models like neural network and regression tree models and shows that this proposed model gives competitive efficiency with the other models.

Ziauddin et al. Mentioned a mathematical model for estimation of measures like effort in software projects developed in Agile. The model was verified with dataset of 28 projects dataset having number of story points which represents the size and team speed denotes for user requirements an completes in a particular cycle. The story point is a measure of user story (user requirement). The user story is associated

with two features size and complexity. Both size and complexity have values for 1 to 5, where 1 shows a very small and least complex story that is it can be completed only in a few hours of work, is very straightforward with few unknowns, requires no research and effects are localized to that story itself while 5 denotes an extremely large story, extremely complex, requires an expert skill set and has many dependencies on other stories. Each user story is thus equal to $\text{Size} \times \text{Complexity}$ of the requirement and total user stories of the project is sum of all user stories, which gives the count of story points. The calibration of velocity is done using factors like Friction and dynamic forces which calculates Deceleration. The effort of the project is then calculated. Then the accuracy was checked using the evaluation measures MMRE, PRED. The MMRE observed was 7.19% and Prediction accuracy observed was 57.14%. S. Kheiri discussed the impact of various machine learning techniques like ANN, Bayesian, SVR with kernel RBF on the diagnosis and prediction of various factors affecting dermatological diseases. Further, they concluded that machine learning application increases the accuracy of prediction and decreases the errors in terms of various factors.

Agile Adaptability Challenges in Corporate Organization

The above applicability of Agile Scrum methodology in software development is a combination of from my experience together with best practices of the methodology itself. The applicability of Scrum best practices in corporate organization was not sufficient to extend where we could borrow some practices from other agile methodologies and even comprise some of the agile practices.

The challenges are grouped based on software development phases, to ensure readers will link them easily with their current process. In addition to the grouped challenges there are some overall ones in different areas.

Overall

This section will provide readers some of key challenges and issues corporate organization go through while in Software development using agile methodology:

- Lack of document or Incompleteness
- Enterprise architecture standards Legacy systems capability and architecture
- Adjustment of organization operating model
- Agile impact on organization existing standards
- Lack of technical Skills
- Gaps in team's communication

Planning

The below is the list but not limited to, for agile methodology adaptability challenges in first phase of Lack of requirements business ownership.

Unclear Minimum Viable Product (MVP) definition.

Endless planning loop (Program & Sprint).

Analysis

The below is the list but not limited to, for agile methodology adaptability challenges in second phase of analysis:

Lack of business owners' involvement

Incomplete business analysis (epics, features & user stories)

Disconnect between Product backlog and MVP

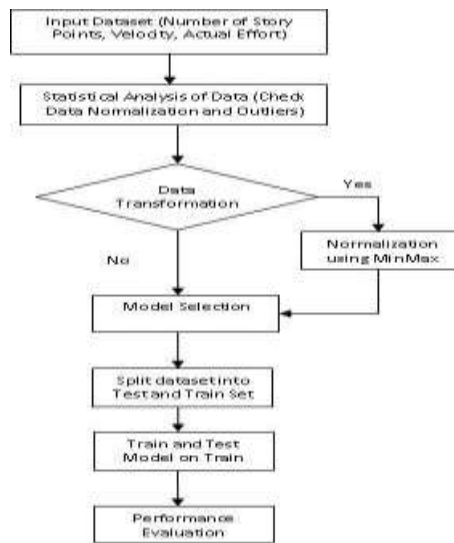


Fig. 3 Kanban Approach

Design

The below is the list but not limited to, for agile methodology adaptability challenges in third phase of design:

- Design focusing on UX and UI only
- Lack of technical feasibility impact of design
- Endless changes in design creates loop

Development

The below is the list but not limited to, for agile methodology adaptability challenges in fourth phase of development:

- Engineering practices compromised due to delivery timelines
- Lack governance principles and standards
- Endless Changing Requirements and testing

Maintenance (Operations & Support)

The below is the list but not limited to, for agile methodology adaptability challenges in fifth phase of maintenance and operation support:

- Development
- DevOps setup maturity level
- Continuous changes impact on standards
- Lack of business ownership of products

Objectives of Research work Agile software development methodologies have many characteristics including iterative, prototyping and needy documentation. Agility is iterative, develops from subsequent improvements in sprint lifecycle.

Reasons for adopting Agile from “A state of Agile survey 2011”

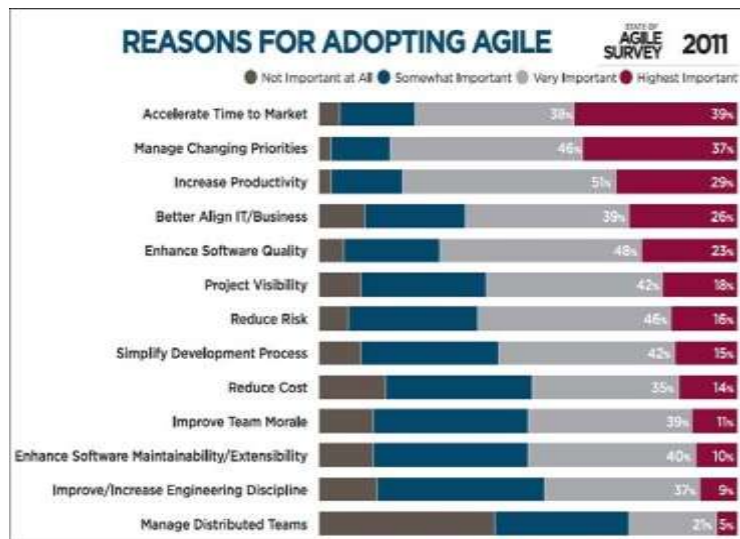


Fig. 4. Objective of Research based on Performance Evaluation

Key Benefits of Scrum Agile Methodology

It is necessary to highlight some of the benefits Agile Scrum methodology brings on board in software development. The below are key listed benefits but not limited:

Time to completion is significantly faster than the traditional waterfall approach. This is a result of breaking down the work packages into small deliverables often completed within a 2 to 4 -week period. Flexibility and adaptability to change allows the ownership and development teams to deliver software that is relevant and customer-centric. This eliminates the risk of obsolete requirements driving development.

There is top-down transparency between squads and the key stakeholders throughout the development lifecycle. This improves accountability and quality of deliverables.

Incremental delivery drastically reduces risks associated with cumbersome development projects, and also has a positive impact on cost to the organization.

Following techniques are used in this research to predict the effort of projects. The story point approach is used in the SCRUM model of agile development framework. The regression-based models are used so as to predict learning from the historical dataset which is considered for training the model.

- **Story Point Approach:** A user story is basically a user requirement. These stories are developed in iterations, usually termed as Sprints in SCRUM Agile framework. These user stories are measured in story points. Total number of user stories in an iteration gives the number of story points. Velocity is another independent variable which gives the number of resources tasks the group completes in a single iteration. Using these two variables effort is calculated in man-hours.
- **Linear Regression:** Linear regression is a linear approach which models the relationship between a response variable referred to as target variable and one or more dependent variables referred to as predictor variables.
- **Logistic Regression:** Logistic regression is used when the relationship can be non-linear.
- **Ridge Regression:** It compresses the various calculative variables; it is mostly used to avoid multi same linearity. It reduces the model complexity by coefficient shrinkage.

Experimental Evaluation

Presently, Information technology cost calculating models are calculating are COCOMO,FPA,, SLOC, SEER. In this research introducing in Linear based, Multiplicative scenarios, and Putnam Methods of calculation, Brake down cost, by using Machine Learning with the help of python.

This describes the experimental evaluation of the proposed model which is based on regression. The considered techniques, dataset used by Ziauddin et al. Used. The dataset consists of three columns, the first column and the second column represent the story points, the initial or the raw velocity and the third column is the actual effort for completing the project. This dataset is used to determine the software development effort. Machine Learning Models takes input of both story points and velocity and the accuracy of the predicted effort values as output.

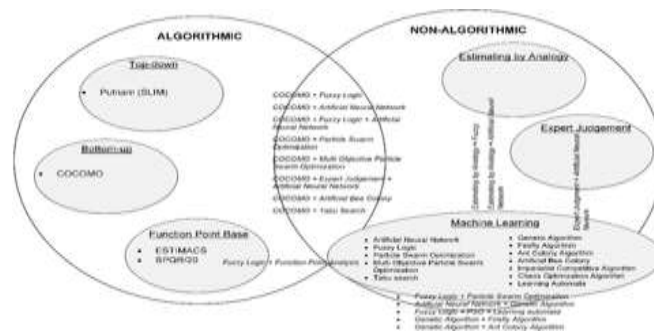


Fig 5 Experimental Evaluation for Research outcome.

The following steps are performed:

Step 1. Statistical Analysis of Dataset

- (i) Plot a Scatter plot for dependent and independent variable (No of Story points Vs. Effort)
- (ii) Plot a histogram of Effort values

Step 2.

- (iii) Calculation of statistical measures like Kurtosis and Skewness
- (iv) Outliers are checked with the help of Box Plots

b) If Data is Normalized proceed to step 3 else Step 2.

Step 3. Model Building Build the model using Regression.

Step 4. Partitioning of Dataset and Cross Validation

Step 5. Calculation of Prediction Accuracy and Error Rate.

Methodology Adopted In The Execution Of The Research

Methodology adopted Considered below steps:

- Verification of needy methodology
- Verification of organization requirements, accepting and making use of Agility.

The acceptance of the Agility verified and elaborated with various aims. The adaptation method of the agile methodology differ from agile specific software development to general engineering principles.

Acceptance done by recognizing roles, activities, functions and procedures that uses to be exact for present case studies. Situation is recognized with various factors related to the cubicles of combined both internal and external work culture, aims, modeling levels and earlier knowledge. As so many parameters

are verified size, distribution, turnover, and previous good cooperation, in various working areas, automated tool, and technical aspects highly acquire knowledge.

Project aspects are such as internal cooperation and self-acquiring are unrevised in present acceptance principles of the agile referred to cut short the processes to every company.

Hence, purpose of this is to accept the method for acceptance and AM as per the project members.

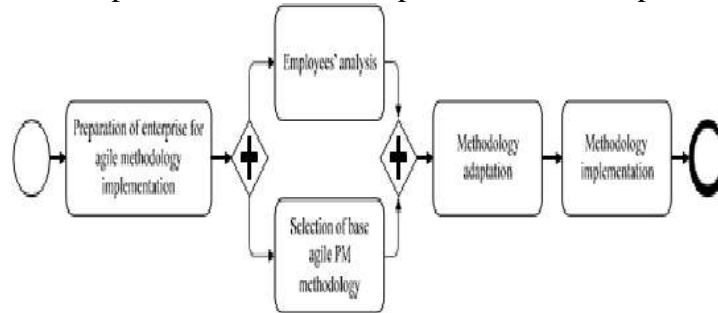


Fig 6. Methodology of Agile Approach

The proposed research needs the principles of revised strategy, methodology acceptance and use socio-metric and adaptation of research principles for analysis of the project team. The major contribution of this method is integration of both internal, external with success towards the company team in the acceptance of requirements of the APM with focus to make better team efficiency and self-realization. Assessment of the future developing method will be useful mainly in Information Technology Industries.

Expected Outcome

From the literature, found that Agile is the most popular framework for development of project now-a days in software industry, also, it is evident that effort estimation is an important task in project management. This research considers the story point approach for effort calculation and for optimization, Linear, Ridge and Logistic regression models are used. The results obtained are compared based on certain evaluation measures like MMRE, RMSE and PRED. It has been observed that the Ridge Regression outperformed other models. Further the suggested regression-based model is compared with a statistical model given. The computations in this research are performed using Python and the outputs are generated.

This work can be further extended using

other machine learning techniques as well as the use of ensemble algorithms may give better prediction accuracy.

Results And Comparative Analysis

The regression technique takes a training dataset indicates the input pattern space and finds the function $f(x)$ which optimizes the training data. In this research, our target is to create a regression model which estimates the effort of the software projects in man-hours on the training data further it is evidenced.

From the literature, that the models which are based on machine Learning techniques mostly proves to be advantageous over other statistical and non-algorithmic models which are based on intuition of experts for prediction of measures for software projects like effort, cost etc.

Considering the metrics RMSE, MMRE and PRED as evaluation measures and applying the Linear Regression, logistic regression, and Ridge Regression the results are obtained.

The mean magnitude of relative error and root mean square error along with the prediction accuracy at 25% level of significance of all the applied models .The proposed regression-based model is then

compared with the results of the model proposed by Zia and the results show that regression-based models outperform the model proposed by Zia which is a statistical model.

The scatter plots depicting comparison of actual effort values vs. predicted effort values using all three models of Regression. Comparison of proposed model with existing model Approach RMSE MMRE PRED, Linear Regression Logistic Regression Ridge Regression.

REFERENCES

1. D. Cohen, M. Lindvall, and P. Costa, *Advances in Software Engineering*, chapter An Introduction to Agile Methods. Elsevier, Amsterdam, 2004.
2. Amit Sharma and Rajesh K. Bawa, A Multilevel Hybrid Approach for Selection of Agile Development Method Using AHP, PROMETHEE and Fuzzy Logic, *STRUCTURAL INTEGRITY AND LIFE* Vol. 17, No 1 (2017), pp. 49–54 March 2017.
3. Sharma, A., Sharma, R., A systematic review of agile software development methodologies, *Proc. of the Nat.*
4. Conf. on Innov. and Develop. in Engineering and Management, 2015.
5. Keramati, H., Hosseinabadi, S.H.M., Integrating software development security activities with Agile methodologies. 6th IEEE/ACS Int. Conf. on Comp. Systems and Appl., AICCSA 2008, Doha, Qatar, pp. 749- 754
6. Sandhya Tarwani and Anuradha Chug, Agile Methodologies in Software Maintenance: A Systematic review, *Informatica* 40 (2016) 415–426, November 2016.
7. University Science, 1989. Sonia Thakur and Amandeep Kaur, Role of Agile Methodology in Software Development, *International Journal of Computer Science and Mobile Computing*, Vol. 2, Issue. 10, October 2013.
8. Giulio Concas, Ernesto Damiani, Marco Scotto and Giancarli Succi, Agile Processes in Software Engineering and Extreme Programming, 8th International Conference, XP 2007 Como, Italy, June 2007.
9. Orit Hazzan and Yael Dubinsky, Agile Software Engineering, Undergraduate Topics in Computer Science ISSN 1863-7310, 2008.
10. A.H. Mohammad et al., "Agile Software Methodologies: Strength and Weakness," *Int. J. of Engineering Science and Technology*, Vol. 5, No. 03, March 2003, pp. 455-459.
11. L. Williams, "A Survey of Agile Development Methodologies", ©Laurie 2007.
12. M.R.J. Qureshi, "Agile software development methodology for medium and large projects," *IET Software*, Vol. 6, No. 4, ©2012 The Institution of engineering and technology, pp.358-363.
13. F.K.Y. Chan and J.Y.L. Thong, "Acceptance of Agile Methodologies: A critical Review and conceptual framework," *J. Decision Support Systems*, Vol. 46, No. 4, 2009, pp. 803-814.
14. L.R. Vijayarathy and D. Turk, "Agile Software Development: A survey of early adopter," *Journal of Information Technology Management*, Vol. 11, no. 2, 2008, pp. 1-8.
15. S. Ilieva et al., "Analyses of an agile methodology implementation," in *Proc. 30th EUROMICRO Conference*, ©IEEE, 2004, pp. 326-333.