ASIAN JOURNAL OF INFORMATION TECHNOLOGY
Journal Home Page: https://medwelljournals.com/journalhome.php?jid=1682-3915

Prediction of Agile Release Planning and Agile Methodology

Komal Sharma and H. Abdul Gaffar
School of Computer Science and Engineering, VIT University, Vellore, India

Key words: Iterative plan, factors, approach, planning, k-NN classifier

Corresponding Author:
Komal Sharma
School of Computer Science and Engineering, VIT University, Vellore, India

Page No: 187-192
Volume: 18, Issue 7, 2019
ISSN: 1682-3915
Asian Journal of Information Technology

Abstract: Agile development improvement system got significance as of late. The Agile reasoning advances incremental and iterative plan and usage. In iteration one or more product feature delivers. Release planning is a principle action in any of Agile approach. Fundamental factors that should be considered are the specialized priority intrinsic in the requirements; the component's business value seen by project stack holders, team limit and effort required to finish the requirement. To accomplish higher adaptability and to better fulfil real client requirements, there is an expanding propensity to create and convey development in an incremental manner. In embracing this procedure, requirements are conveyed in releases thus a choice must be made on which requirements ought to be conveyed in which release. Three principle contemplations that should be assessed are the technical precedence's innate in the requirements, the commonly clashing priorities as controlled by the delegate stakeholders and also the harmony amongst required and accessible exertion. The technical precedence limitations identify with circumstances where one prerequisite can't be actualized until the point when another is finished or where one requirement is executed in an indistinguishable addition from another. Stakeholder inclinations might be founded on the apparent esteem or earnestness of conveyed requirements to the diverse stakeholders included. The technical priorities and individual stakeholder priorities might be in struggle and hard to accommodate. There are different devices accessible in the industry to oversee project yet they are missing to give arranging while at the same time thinking about every one of these variables. Previously Genetic Algorithm (GA) has emerged from ideas, presented from the regular procedure of natural development. GA utilizes selection, cross-over and mutation to develop an answer to the given issue. In this study, a research has been made to formalize the release planning. At that point, an approach is proposed to do release planning utilization using the top-k random Gaussian Genetic algorithm and predict the accuracy of the different Agile method using k-NN classifier.

INTRODUCTION

These days in any field like business, instruction, sports and so forth the achievement rely upon the product being utilized because of the quick improvement in innovation the companies need its product refreshed and the product should meet all the business needs. The quick developing fulfilments between the companies have made a great challenge for the product improvement organizations. The companies require its business programming on time with in the financial plan and furthermore amid the improvement time the authoritative prerequisites changes because of the adjustment in the business procedure which makes issues for the advancement team as it turn out to be exceptionally troublesome and costly for the advancement team to roll out improvements amidst the improvement stage and in this manner the product wind up finished spending plan and late. So as to defeat those issues in the product improvement the product advancement organizations must need to utilize adaptable programming improvement techniques that can suit any progressions at any phase of the product advancement. Agile improvement is exceptionally adaptable for changed prerequisites at any phase of the product advancement and that is the reason extremely appropriate for the associations to take the advantage from the Agile strategies. The word Agile means light weight; the primary subject of Agile technique is the straightforwardness and speed. The principle purposes of Agile strategies are incremental cooperative straightforward.

As Agile strategies turn out to be increasingly prevalent day by day, industry see iterative, transformative and incremental programming improvement as the "cutting edge" substitution of the waterfall show. Presently associations are being moved from customary programming advancement philosophies like waterfall models to Agile procedure. Agile methodology centres on delivering working product instead full-featured product this encourages fast conveyance of ideas to frame into products. By releasing working products to its clients, associations get snappy and speedier input from client. Therefore, they can anticipate the market in a quick and proficient way (Karlsson and Agerfalk, 2004). The full release of the item is partitioned into little emphasizes called sprints. Each sprint contains at least one completely created and finishes highlights or requirements. These includes in the Agile world are called client stories. Arranging of all the sprints which recounts what number of and what stories will be part of various sprints is called release planning. This planning relies upon exertion, team limit, business esteem and specialized priority between various stories (Gupta and Dwivedi, 2012). The point is to convey most extreme business esteem as quickly as time permits. This is an NP-hard combinatorial advancement issue (Qumer and Henderson-Sellers, 2006). Numerous Agile methodologies are utilized as a part of the business, for example, DSDM (Data System Development Method), XP (Extreme Programming), RAD (Rapid Application Development), SCRUM, FDD (Feature Driven Development) and so on. Huge software like Windows, Google Drive can't be produced utilizing Agile methods as there is appropriate documentation required for the system being created. Because of these inconveniences, there is a need to a device that decides fitting programming improvement technique for the project close by in view of the qualities of the projects. The estimation of the project qualities changes as indicated by projects like for portable application specialized hazard is less, however, for aircraft framework specialized hazard is more. This study analyse for developing software which one is most suitable Agile methodology based on the input parameters and using feed forward neural network and extract total requirement to be completed with minimum time.

Agile methodology: Agile software development is a transformative, very collective, quality-engaged, taught, approach of software development and conveyance. In Agile, conceivably shippable working software is delivered at standard interims for audit and revision in light of client criticism (Stapleton, 1997). Agile have its long history of development. It takes best of variousiterative developments systems accessible in 2001. Craig and Victor (Coad *et al*., 1999) gave a point by point history of this. In February, 2001, a gathering of seventeen process experts speaking to: Dynamic Systems Development Method (DSDM), Feature Driven Development (FDD), Scrum, Extreme Programming (XP) and others who were occupied with advancing present day Iterative and Incremental Development (IID), methods and standards met in Utah to talk about regular future grounds of IID (Gupta and Dwivedi, 2016). From this meeting they consented to frame a gathering called 'Agile Alliance'.

Agile conveys finish rundown of necessities into various cycles. These cycles are called sprint. Ordinary sprint span is taken between two to a month. Each sprint contains at least one story (or features).

Every story contained a short portrayal and an arrangement of acknowledgment test criteria that must be utilized to exhibit that the story had been effectively and completely done. Stories are kept up in item overabundance. Item owner is mindful to keep up this item overabundance. Whenever, another necessity comes, the client story is composed, for that prerequisite and item owner place this into item overabundance. Item owner allocates some business esteems to these stories after dialog with various partners. This business esteems may rely on showcase demands, anticipated income after conveyance, hazard-related with this conveyance of the feature and numerous more parameters (Gupta and Dwivedi, 2012). A more demandable feature may be relegated higher esteem. In the focal point of Agile, there is a team which is actually skilled to finish these

necessities. Team does exertion estimation of various stories (Agile Manifesto, 2001). This estimation is done as a story point. A more mindboggling story will be assessed as the higher story focuses. Story point for the most part assigned as Fibonacci numbers (Avison, 1996) like 1, 2, 3, 5, 8, 13 and so on. On the off chance that there are any contentions at that point planning poker might be used to go ahead a last conclusion for estimation (Brinkkemper, 1996).

Team picks the stories from this release overabundance and do the sprint arranging. A colleague called, 'Scrum Master' encourages the whole required foundation and Agile information base to team. Scrum master fills in as correspondence connects between product owner and team. Amid the Sprint, the team completes the daily scrum meeting which is 15 min in length. In this gathering team talk about the work done today, arranged work for tomorrow and assuming any leap emerges. This encourages the straightforwardness in the team where each part knows what the others are doing. Toward the end of each sprint, the team does the item demo of the finished usefulness, trailed by a sprint audit and review meeting. Conveying necessities as sprints give following favourable circumstances:

- Necessities can be organized
- As expectations are littler, the cost and calendar can be effortlessly evaluated
- Clients get some portion of the framework early and in this way are more probable that clients begin to utilize item early
- Client criticism can be acquired at each sprint and arranging might be balanced if required
- It underpins better response to changes or increments to necessities
- It effortlessly consolidates item with changing and developing client prerequisites

In an Agile situation, planning occurs on two levels:

Release planning: Release planning occurs on the in the first place day of each sprint and spotlights on the more drawn out-term, vital objectives for the task. On the off chance that necessity or on the other hand economic situations are changed, this will be reevaluated. Release planning tends to all choices identified with the choice and task of stories to a grouping of sequential item releases.

Sprint planning: Sprint planning additionally happens the to begin with a day of a sprint, yet manages the specifics of each sprint. In light of group speed (i.e., limit), the team picks most elevated conceivable business esteem stories for that sprint.

The principle contemplations that should be taken care at the time of release planning might be given as following (Qumer and Henderson-Sellers, 2008):

Effort: The exertion is story direct doled out toward the given story. This indicates the unpredictability of story and time taken to finish the specific story.

Business value: Business esteem signified needs of a given story. This might be connected to real income increase after the development of this story.

Dependency: Dependency indicates, the specialized priority imperatives identified with circumstances where one prerequisite can't be executed until another is finished.

Group velocity: Velocity is the normal feature or story focuses a group finishes amid a sprint. The group knows the group speed from its past experience.

Problem statement: During the release planning backlog a set of whole requirements R comes into existence and they are directly mapped to user stories USi where n is total no. of requirements and i = 1: n:

$$R = \{US1, US2, US3, ..., USn\}$$

And to complete these stories how much effort required is denoted by Eff:

$$Eff = \{Eff1, Eff2, Eff3, ..., Effn\}$$

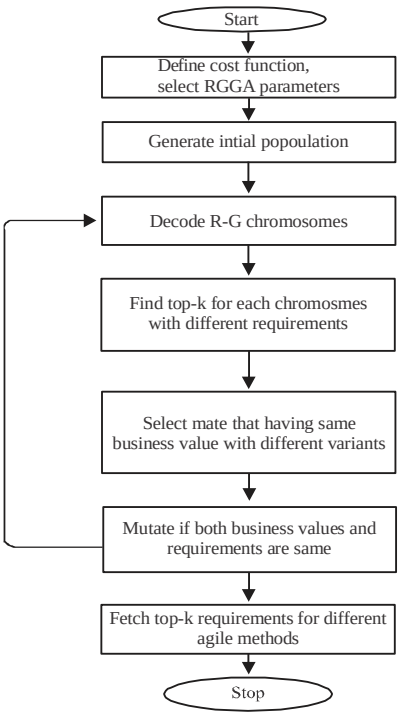

Fig. 1: Framework

Stack holders of project assigned some business values to these user stories:

$$BV = \{BV1, BV2, BV3, ..., BVn\}$$

So, to generate the highest business value we should consider team capacity, total number of requirements, project stack holder time and to predict which Agile methodology is best or accurate for the given set of requirements (Fig. 1).

MATERIALS AND METHODS

Genetic algorithm: Genetic Algorithms (GA) have a place with a stochastic class of problems motivated from the procedure of common development. GAs was first presented by Brinkkemper (1996) at University of Michigan, USA in 1970's. GAs starts with a populace of arrangements called chromosomes. Two arrangements are then chosen as parents to play out the crossover activity. In the crossover procedure the data between the two parents is swapped to create at least one youngster arrangements. In the subsequent stage, transformation process is played out that arbitrarily adjusts a few qualities inside the chromosome. The entire procedure is guided by the standard of survival of the fittest. The hunt continues until the point that a predetermined halting foundation is come to. For each progressive generation the fitter arrangements are chosen to shape another populace. A thorough presentation alongside different applications has been given by Qumer and Henderson-Sellers (2008).

Top-k algorithm using random Gaussian Genetic algorithm: In this study, we consider the systems of a versatile Genetic algorithm utilizing two parameters: runs executed and population measure. The mutation rate isn't considered in our framework. Littler mutation rates make for shorter running circumstances, however, the inquiry space will be littler. At the point when the mutation rate is higher, the merging pace will moderate, implying that the mutation rate must be set higher to discover better outcomes. In our tests, we found the best mutation rate to be in the vicinity of 30% and the half. We watched that there is no connection between finding the best naturally versatile pattern and the mutation rate.

The quantity of executed runs was chosen to guarantee that the last esteem is the best. The framework endeavours to influence the best answer for show up at any rate K times. On the off chance that this edge isn't achieved, the quantity of runs executed is naturally expanded. The framework additionally endeavours to abstain from executing excessively numerous runs while guaranteeing that the esteem acquired is ideal. The method of the versatile instrument is given in an algorithm. To start with it sorts the values produced by GAs (Genetic algorithms) into a world-class set. It at that point checks whether the first K things of significant worth are indistinguishable. On the off chance that indistinguishable, the framework finds the best arrangement. Something else, the framework will expand the number of keeps running the number of runs at first set, until the limit is come to. On the off chance that the quantity of runs achieves R times the quantity of beginning runs, the framework will twofold the populace (from the underlying size) to extend the space for search. The algorithm is recorded as takes after.

Pseudo code of proposed algorithm:

1. for all submitted release plans in the set; Ti
2. for all resources; Rj
3. Ctij = Etij+rtj; End For; End For
4. Prepare whereas jobs set is not unfilled
5. Find release plan Tk that cost minimum execution time
6. Allocate Tk to the reserve Rj which contributes smallest predictable complete time
7. Remove Tk from the release plans set
8. Update ready time rj for select Rj
9. Update Cij for all Ti
10. End Do
11. Choose initial population
12. Evaluate the separate fitness's of a confident quantity of the population
13. Select pairs of best- ranking individuals to reproduce
14. Apply crossover operator
15. Apply mutation operator
16. until terminating condition
17. Reload the release plan
18. Do while the most heavy load resource is considered then no need for reload release plan
19. Find release plan Ti that cost minimum execution time on the heavy load resource Rj
20. Find the minimum completion time of Ti produced by resource Rk
21. If such minimum completion time < Release plan
22. Reassign Release plan Ti to Resource Rk
23. Update the ready time of both Rj and Rk
24. End If
25. End Do //where Random Gaussian represents maximum completion time of all release plans which equals to the completion time of the most heavy load resource

We say that a random variable is normally distributed Agile methodology with mean μ and standard deviation σ if the probability density function is given by:

$$f(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(x-\mu)^2}{2\sigma^2}} \quad \text{for } -\infty < x < \infty$$

Because of its importance in statistics, it is worthwhile to memorize the form of the density function. The cumulative distribution function $F(x) = P(X \leq x)$ needs to be calculated by a computer or looked up in tables.

The probability density function of a normal distribution with mean 0 and standard deviation 1 is displayed in Fig. 2. Note the bell-shaped curve.

Excel has a function that returns the cumulative distribution function and the density function for a normal distribution with a specified mean and standard deviation. The syntax is:

NORMDIST(x, mean, standard_dev, cumulative)

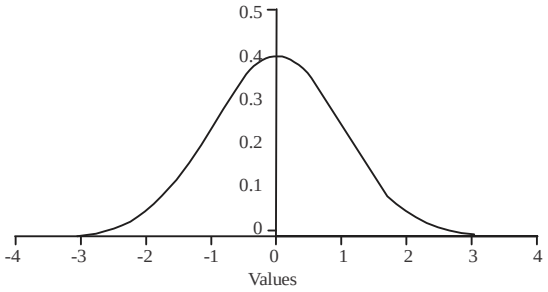

Fig. 2: The probability density function of a random Gaussian distribution with mean 0 and standard deviation 1 (normal density)

Where, x is the value at which if want to evaluate the distribution, mean is the mean, standard_dev is the standard deviation and cumulative is a logical value (TRUE for the cumulative distribution function and FALSE for the probability density function).

kNN classifier: The k-Nearest Neighbour (k-NN) is an arrangement procedure which arranges an obscure information by looking at a separation between the obscure information and existed information (Gupta and Dwivedi, 2016). The 'k' esteem speaks to the closest neighbour. There are numerous strategies utilized for compute the separation between information. Be that as it may, the most widely recognized technique is utilizing Euclidean distance. In this study, we are using k-NN just to find out the accuracy of different Agile methods like (Crystal, Scrum, FDD, KNBN) based on the input requirement, business value used and corresponding stack holder time.

The Genetic algorithm as:

Step 1: we set $G_{counter} = 0$
Step 2: generate the initial population by choosing chromosome in form random
Step 3: appraise the objective and fitness function of each individual
Step 4: selection operator such as sort in a descending order all the population according to some value such as their fitness
Step 5: perform crossover operation and mutation operator
Step 6: choice of population size of chromosomes from parents
Step 7: increment counter $G_{counter} = G_{counter}+1$
Step 8: if $G_{counter} = \max G_{counter}$ and exit with best solution and stop else $G_{counter}$<generation size goes to 3 and continues

RESULTS AND DISCUSSION

Random Gaussian Genetic algorithm has been used to resolve the difficulties in three different test cases for discovering its optimization possible where the objective function was incomplete within power ranges of the gene release units and transmission losses were also taken into account. In some environments, software may be transported to users or at least a subsection of users, incrementally at the end of each iteration or every couple of 1000 iterations. After a first feature list has been recognized, prioritized and potentially estimated the team

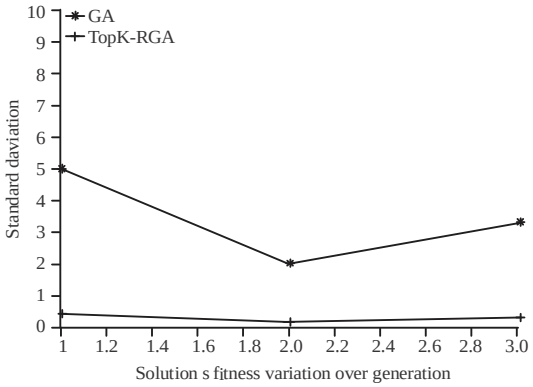

Fig. 3: Best fitness variations over generations

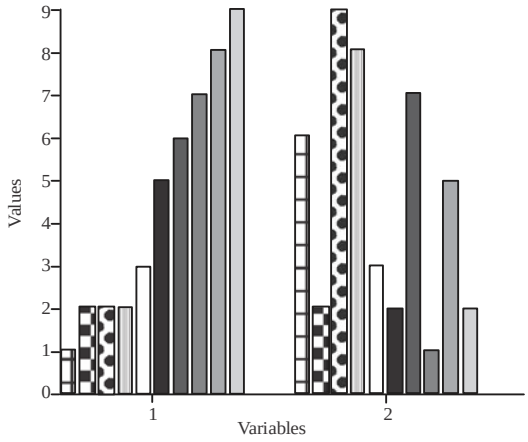

Fig. 4: No. of requirements is balanced in Topk-GA

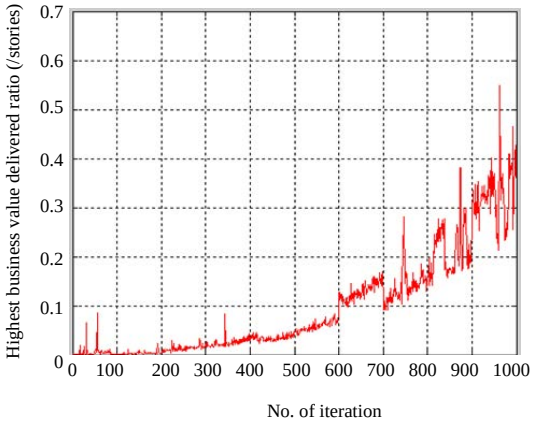

Fig. 5: Highest business value delivered ratio

holds a release planning meeting to start the general release list and control which features can likely be distributed (Fig. 3-6). The general release plan in relationships of ordered features is then used to

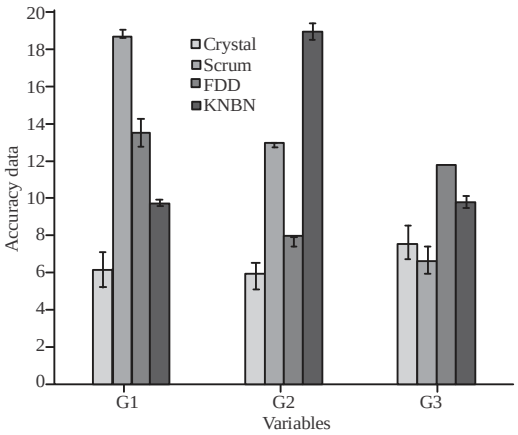

Fig. 6: Prediction accuracy of different Agile methods

nonstop feed separate iteration plans. Some Agile methods pressure clear split-up of release plans between programmers and customers. During planning, only the client is accountable for business decisions and prioritization and only the programmers are accountable for release plan approximation and expansion execution. Agile methods also powerfully discourage organization from impulsively imposing technology choices on the growth group, instead giving designers as much autonomy as possible to choose the best tools for the scheme and the scheme (Fig. 3-6).

CONCLUSION

The proposed solution is truly able to give a sequential order of stories by user which will be followed by release planning. Additionally, Agile methods accuracy will be predicted according to the set of input.For a team that's just adopted the Agile methodology or even embarked on a new product, the velocity estimates of the first few sprints will probably be erratic. But as teams gain experience, the velocity will peak and then reach a plateau of predictable flow and performance expectancy. A decrease in consistent flow will indicate problems in the development and reveal the need for change.

REFERENCES

Agile Manifesto, 2001. Agile manifesto and principles: Manifesto for agile software development. http://agilemanifesto.org.

Avison, D.E., 1996. Information Systems Development Methodologies: A Broader Perspective. In: Method Engineering, Brinkkemper S., K. Lyytinen and R.J. Welke (Eds.). Springer, Boston, Massachusetts, ISBN: 978-1-4757-5824-5, pp: 263-277.

Brinkkemper, S., 1996. Method engineering: Engineering of information systems development methods and tools. Inf. Software Technol., 38: 275-280.

Coad, P., E. Lefebvre and J.D. Luca, 1999. Java Modeling in Color With UML: Enterprise Components and Process. Prentice Hall, Upper Saddle River, New Jersey, USA., ISBN: 9780130115102, Pages: 218.

Gupta, D. and R. Dwivedi, 2012. A step towards method configuration from situational method engineering. Software Engi. Int. J. (SEIJ.), 2: 51-59.

Gupta, D. and R. Dwivedi, 2016. Configurable method model of agile methods-for creating project-specific methods. Proceedings of the International Conference on Software Engineering Research and Practice (SERP'16), July 25-28, 2016, Las Vegas, USA., pp: 86-91.

Karlsson, F. and P.J. Agerfalk, 2004. Method configuration: Adapting to situational characteristics while creating reusable assets. Inf. Software Technol., 46: 619-633.

Qumer, A. and B. Henderson-Sellers, 2006. Crystallization of agility: Back to basics. Proceedings of the ICSOFT 2006 1st International Conference on Software and Data Technologies, September 11-14, 2006, Setubal, Portugal, pp: 121-126.

Qumer, A. and B. Henderson-Sellers, 2008. A framework to support the evaluation, adoption and improvement of agile methods in practice. J. Syst. Software, 81: 1899-1919.

Stapleton, J., 1997. DSDM: Dynamic Systems Development Method: the Method in Practice. Cambridge University Press, Cambridge, USA., ISBN: 9780201178890, Pages: 163.