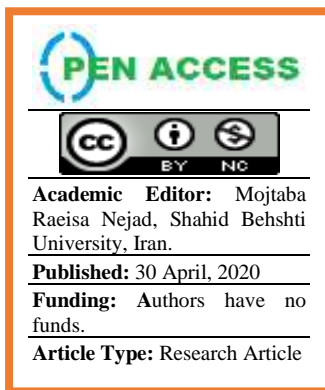


SYSTEMATIC APPROACH IN CHANGE MANAGEMENT FOR ELICITATION AND PRIORITIZATION OF REQUIREMENTS IN AGILE

AMR M. JADI, MOHAMMAD A. ALQARNI, EESA ALSULAMI, ABDULRAHMAN A. ALSHDADI, SHAHID N. BHATTI*

University of Hail, Saudi Arabia
University of Jeddah, Saudi Arabia
Bahria University, Islamabad, Pakistan
Email: snbhattii.buic@bahria.edu.pk*

ABSTRACT



As the numbers of software projects are increasing, large numbers of projects are following agile methods for efficient and agile development. Agile methodology focuses on customer satisfaction instead of conforming to plan. Agile methods show patterns of success and even deficiencies. In agile methodologies as customers are part of development strategies consequently in any project periodically key changes occur during or after the development. Traditional Methodologies do not welcome changes instead they stick to their set of requirements and try to reduce the cost by eliminating change. Change management is one of the pivotal and beside numerous advantages agile methodology offers. Agile methodology is a response to traditional methodologies as their main focus is to appease the customer even if it requires major changes at any stage during the design or development. Although when these corresponding revisions occur, they need to be enduring through a systematic process. The following research paper examines how changes are discovered in any project, their impact and further how these changes are prioritized on the basis of their importance and how they are implemented in agile environment.

Keywords: requirements change management; agile software development; scrum; software requirements prioritization; traditional development, requirements engineering;

1. INTRODUCTION

Several traditional methods also known as requirements engineering methodologies have been developed and implemented with the aim to deliver all the requirements of the customer. Requirements engineering is a process of software engineering which aims to identify the requirements of the system, analyse the requirements, document them and validate the requirements on the developed system, an optimized cycle in this is depicted in the Figure 1 [1]. Requirements engineering main goal is to create a document which contains system requirements specification, as traditional methods rely on documentation for communication [2].

Agile came as a response to this approach and has been used widely during the last few years. Different agile methodologies are being used in projects. Agile methodologies rely on face-to-face communication to reach a common goal. All agile methodologies aim to deliver the software with all the changes customer needs and deliver it faster. These methodologies have the same basic principle which is customer satisfaction, adapting changes in the system and deliver it fast. Agile approach is to welcome change at any stage of during and after the development of the software with the aim to reduce the cost of change. It works on the principle to deliver the first iteration in weeks, take feedback from the customer to see if any changes required and implement those changes while improving quality side by side by testing for any defect detection. Each methodology implements quality.

As agile welcomes changes, this can cause the success or failure of the project as it needs to be handled carefully [3]. Requirements discovery or elicitation is one of the most challenging tasks of the project development. Requirements are usually gathered at the start of the project but it is very unlikely that they are complete at that stage, as requirements changes or evolves through the process and they require to be handled carefully. For customer satisfaction these changes are needs to be implemented in a proper way by organizing them. Requirements are needs to be prioritized according to the customer's needs to discover which needs to be implemented urgently and are most important requirements of the system. Requirements are needed to be prioritized on the basis of many factors like customer needs, their cost, impact on the system etc. by maintaining the system quality [4].

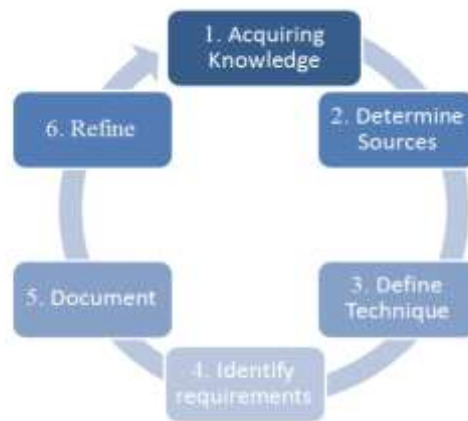


Figure. 1 From requirement identification to documentation [5]

2. MATERIALS AND METHODS

2.1 Agile requirements

In [3] conducted a research on 16 organizations which used agile requirements engineering (RE), they conducted case studies for some organizations, interviews, data analysis and data collection. In agile methodology they analyse major requirements on the priority basis and evolves it during the development to meet customer's need as new requirements emerges, added and modified. By [6], as compare to traditional requirements engineering, in which requirements are only prioritize once on the basis of requirements analysis whereas in agile RE, requirements are prioritized in each development process/ iteration. On contrary to traditional RE, agile methodology is only based on business value whereas traditional RE is based on many factors like risk, business value, implementation dependencies and cost. To satisfy customer needs accommodating requirements changes in the system during developments. It is easier to implement requirements changes in agile methodology and it also costs less. It requires intensive communication between the customers and developers. Agile is dynamic, adaptive and provides iterative discovery approach at each development level.

In [4] discussed that all RE processes like requirements elicitation, requirements analysis and requirements validations are all involved in agile methodology. In Traditional RE these steps are just done once but in agile methodology these are repeated for each iteration of development. Determining requirement upfront what is required is very difficult, it evolved as time which is done in agile methodology. The difference RE and agile makes is the amount of documentation as RE management requires documentation but in agile documentation is kept at minimum.

In [2] proposed from a case study that while dealing with volatile environment organization is the most challenge for the developers. Traditional RE methodologies don't work efficiently in such environments. Most of the development is more likely to be conducted in volatile environments. As most organizations are adapting themselves to changing technologies, social conditions and market which requires requirements to be changed along with that adaptation. As compare to traditional linear methods which try to impose rigid formal controls, agile project management lets developers and customers to adapt changes according to the circumstances.

In [6] suggested that agile methodology allows high degree of uncertainty and change in the software development which results in lower risk project, quality and better productivity. Managing changes on the software can be very costly if they are ungoverned that's why is requires a proper methodology for the change management. There are also some challenges of using agile methodology but these can be offset easily by adding back some formality like documentation. Agile Methodology is most suitable for the projects that require rapid changing requirements as an alternative management and development methodology for developers. Projects that are even questionable for agile methodology can use some underlying agile principles.

2.2 Elicitation of requirements

In [7], as RE is always concerned with the discovery of requirements, analyzing it, specifying and documenting it. In agile RE, agile manifesto to the RE process focal values are applied. Working with stakeholder's agile team discovers the application domain, operational constraints and the services provided by the system. The techniques used for the elicitation of requirements in agile RE are interviews (closed or open), brainstorming (group technique), ethnography (observational technique) and use case analysis (scenario-based technique). The success of RE secret is good agile developers, customer collaborations and project managers.

The different techniques for the requirements elicitation are Customer Interviews, storyboards, questionnaires, question and answer method, brainstorming and idea reduction, use cases, prototyping and requirements management [8, 9]. As some requirements methods are better than others one can use according to the application domain. According to rating the most effective is QA method. Furthermore, the detail account of the different requirements elicitation and prioritization techniques in this is taken in the research work [1, 3, 4, 6, 7, 10-12] as comprehensively depicted in Table I.

2.3 Prioritization techniques

In [13] describes a checklist on how to facilitate different types of variables of requirements and measure them. The explained framework consisted of three common kinds of variables which are dependent variables, independent variables and context variables. First list independent variable's that influences cause or effects the requirements. Some of the independent variables are Goal, Qualitative Process Description, Hierarchy level, Input, Output etc. List all dependent variables which are dependent on independent variables treatment. These variables are those which answers that why, how and when one approach is preferable over another. Some of the dependent variables are scalability, ease of use, understandability, fault tolerance, time, ease of learning, attractiveness, accuracy etc. Finally, the list context variables which are somehow mediating, or intervening the system.

In [10] discussed that we assign values for any prioritization to different requirements which needs to be prioritize. There are many prioritization techniques of the requirements using different scales and types. Of them the least one is ordinal scale, in which we ordered the requirements on the basis of which one is important than the other. The powerful of them is the ratio scale in which we give scale to each requirement ranging 0-100. A more powerful among them is the absolute scale, it is used where we can assign absolute value i.e. scale of hours [12].

AHP (Analytical Hierarchy Process) is a decision-making method that is used for the prioritization of the requirements of the system. It is done by comparing different types of pairs of hierarchical requirements, which determine which requirement has the highest priority. Cumulative Voting or the 100-Dollar Test is a kind of straightforward technique for the prioritization where the user or the stakeholders are given imaginary 100-dollars/points and they are asked to rate each requirement on the basis of their priority. Result shows exact ratio of each requirement prioritization. This technique is used when there are many requirements to prioritize, as also highlighted and validated in the following work [4, 5, 9, 11, 13, 14]. Grouping or Numerical Assignment is used commonly for the prioritization of the requirements as briefly highlighted in Table I, in corresponding of some of the popular techniques in this. In this technique we group different types of requirements into groups. This technique shows results on an ordinal scale.

Ranking technique is also based on the ordinal scale. Each requirement is ranked separately from 1 being the most important to n being the least important. Different ways can be used to obtain the list of ranked requirements e.g. bubble sort, binary search algorithms etc. This technique is mostly used where there is one stakeholder, but it is possible to take multiple views of multiple stakeholders and take a mean of it.

In Top-Ten requirements technique, each stakeholder is asked to pick their top-ten requirements. This technique causes various issues like while taking mean priorities can conflict between stakeholders, which is the challenge to create balance between them.

Table. 1 Comparison of different requirements prioritization techniques [2]

No.	Techniques	Technical Aspects						Business/Client Aspects			
		Citations	Scalability	Ease of Use	Time Complexity	Decision making	Accuracy	Sales	Marketing	Customer Satisfaction	Strategic
1	Analytic Hierarchy Process (AHP)	50		Yes		Yes	Yes				
2	Binary-Tree Prioritize	8		Yes			Yes				
3	Bubble Sort	8		Yes		Yes					
4	Cumulative voting (CV)	20		Yes	Yes		Yes				
5	Kano Analysis	5		Yes		Yes	Yes		Yes	Yes	
6	MoSCoW	6	Yes	Yes							
7	Pair-wise analysis	10		Yes	Yes	Yes	Yes				
8	Numeral Assignment	15		Yes	Yes	Yes	Yes				
9	Ranking	8	Yes	Yes	Yes	Yes	Yes	Yes			
10	Relative weighting	2		Yes							
11	Top Ten Requirements	18		Yes	Yes	Yes	Yes				
12	Wieger's Prioritization	14	Yes	Yes	Yes	Yes	Yes			Yes	

2.4 Change management

In [12] discusses that instead of conforming to the plan of requirements which is not enough, customer satisfaction is the primary goal in agile methodology. In agile Methods the strategy is to minimize the cost of change throughout the project. Extreme programming (XP) produces the first iteration in few weeks to get response back quicker, it works in small solutions so that the change is minimum and easy to change. Pair programming is used for rapid response and feedback by using prototyping. Agile methodology encourages change at any stage of development.

Agile methodology is the mostly applicable to the high change environments. Agile methodology deals with the change in requirements late in development too. Trends in Information technology are going closer to Agile Methodologies due to emergent requirements and change management.

In [15] proposed that the two factors that comprise agile methodologies are response to changes in due time and exploring changes. Agile drivers are the changes from the environment to maintain its competitive advantage in the business. Methodology has the following parts: changes, assessment model, gap analysis, strategy formulation, implementation and feedback. Different types of organizations require different type of change. Agile methodology is the most important condition for the prosperity and survival in the increasing changes of requirements in business environment. The authors in [5, 15] have comprehensively taken into account the steps in change management in context of requirement discovery, application and impact on overall product.

As requirements management is a systematic process in context of the changing requirements while maintaining effective and efficient agreement within the project team and customer, a systematic approach for requirements change when request for a new requirement emerge is depicted in the Figure 2.

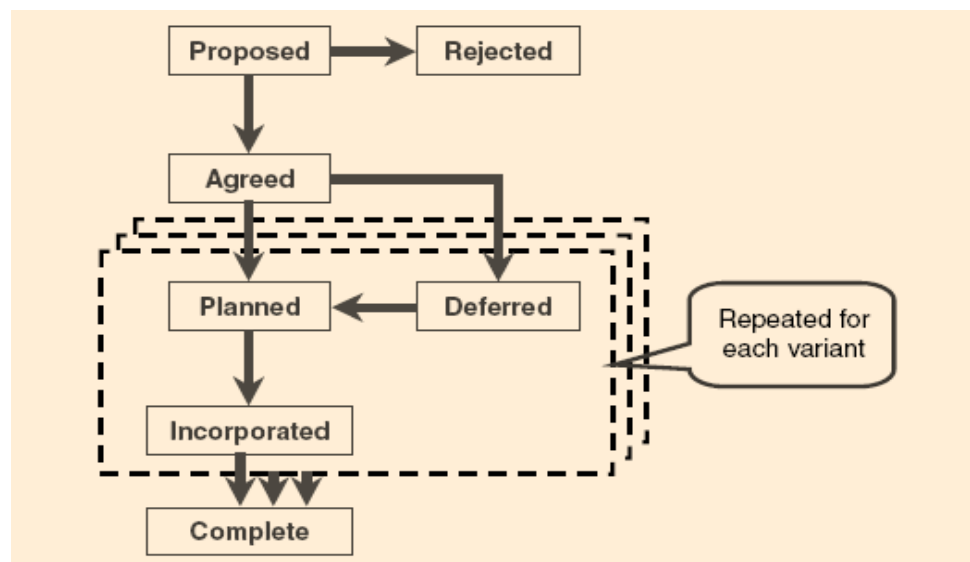


Figure. 2 Requirements management for a new requirement request [5]

3. METHODOLOGY

The goal of the proposed model is to discover requirements and prioritize them on the basis of several techniques to implement changes. For requirements elicitation first step is to have a communication face-to-face with the stakeholders and document their requirements. Interviews can also be conducted on the daily basis during the communication to get their point of view.

Closed interview for the basic requirements of the system to discover the services provided by the system and domain of the system. It is important to discover the domain of the system because it also imposes some requirements on the system, which if not met system may be unworkable.

Open interviews can be conducted to discover non-functional requirements of the system which may affect the overall system. These requirements can be categorized on the basis of system requirements, organizational requirements and external requirements.

Another step in agile methodology is brainstorming, where developers and stakeholders needs to sit together and discover new requirements of the system. Some requirements can be discovered through the ethnography technique, which is to observe the system carefully. All requirements should be complete and consistence to state

the goals of the system and they are verifiable. Next step is to organize them, group the requirements on the basis of related requirements and organize them into clusters.

After the elicitation of the requirements, they need to be prioritized. There are several techniques for the prioritization of the requirements but gives better result when combined together.

First step in prioritization is to group them into clusters on the basis of independent, dependent and context variables. Independent variables are those that influences, cause of effect the requirements of the system. Some of the independent variables are Goal, Qualitative Process Description, Hierarchy level, Input, Output etc., as shown in Figure 3. Next step is to group dependent variables those are dependent on independent variables. It is the measurement criterion while evaluating. Some of the dependent variables are scalability, ease of use, understandability, fault tolerance, time, ease of learning, attractiveness, accuracy etc.

In next step group context variables, which are considered important with regards to the system. Some of them are market type, process model used by the organization, project size, organization, domain etc.

Each requirement is labeled as relative terms to the system such as high, medium and low or critical, standard and optional. Stakeholders will label each requirement with the relative terms according to their requirement. The result will be on ordinal scale. At this time each group is having the same priority of each requirement. For the priority we use the cumulative voting also known as 100-dollar test. Each stakeholder is provided with imaginary 100 dollars and they have to distribute them to each requirement in group separately. The result will be on a ratio scale. If the requirements are large in number the amount can be increased. There is a probability that the stakeholders might bias their evaluation on the basis of their requirements which will result that the selected requirement does not match with other requirements selected.

Next technique is to apply is Ranking technique, which is based on the ordinal scale. The most important technique in each group will be ranked 1 and the least important in each group will be ranked n. This ranking can be obtained through different algorithms such as binary search or bubble sort, as shown in Figure 4, toward the end.

Independent Variables	Dependent Variables	Context Variables
Goal	Scalability	Market type
Qualitative Process description	Ease of use	Process Model
Hierarchy level	Understandability	Organization
Input	Fault tolerance	Project size
Output	Time	Domain
	Ease of learning	
	Attractiveness	
	Accuracy	

Figure. 3 Different variables based on their corresponding classifications

3.1 Change control process

After the requirements are ranked we have the list of requirements that needs to be changed in the system. For any stage of changes we need to follow some steps for change control process which are: (pseudo algorithm is performed below), as provided in steps (i, to iv, (I-IV)).

- 1) Record the change.
- 2) Identify the impact of change corresponding to each requirement within the system.
- 3) Decide if that change needs to be accepted or not (which is already done while ranking the requirements).
- 4) Decide when to implement the change in the system, implement.

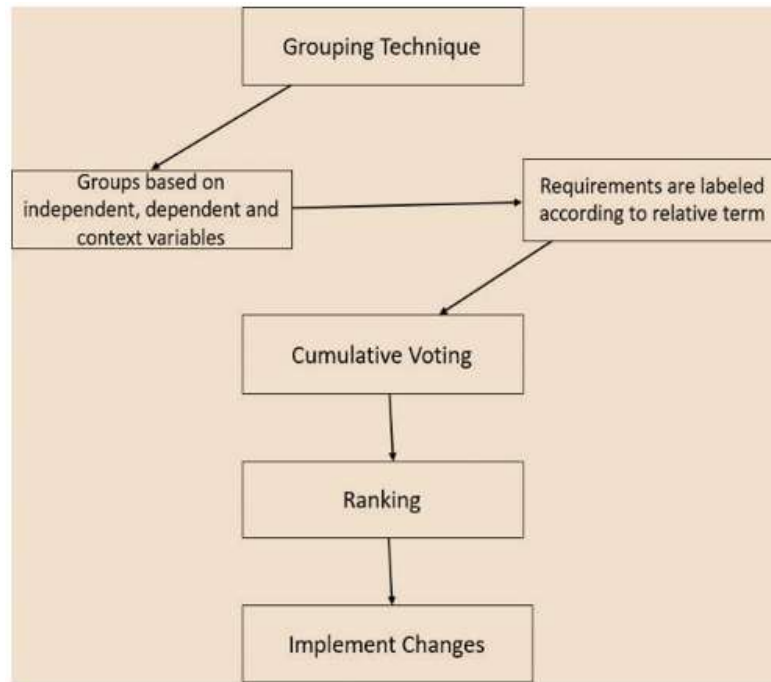


Figure. 4 Adopting the change in context of change management via proposed method

4. CONCLUSION

The agile methodology provides a very efficient requirements engineering process for requirements management via requirements elicitation and prioritization. Beside it provides an approach for a change in the system in software design and development in context of agile methodologies. As requirements often changes or emerge during the development process, these requirements needs to be prioritize and implemented on the basis of highest priority at first. As proposed in this research work when these approaches are implemented in right context resulting in lower risk, better quality and customer satisfaction. Further there are different techniques for discovering requirements and prioritization which can give better results when combined. In this work a systematic methodology has been proposed which can adopted according to the situation, in which we discover new requirements (elicitation), prioritize them and implement them. The proposed model ensures that the right requirements are discovered and implemented at the right time on the basis of priority. In future the proposed scenario can be validated for efficiency via qualitative and quantitative metrics from industrial, software industries data (within KSA) to authenticate the validity of this proposed technique.

REFERENCES

1. Paetsch, F., A. Eberlein, and F. Maurer. *Requirements engineering and agile software development*. in *WET ICE 2003. Proceedings. Twelfth IEEE International Workshops on Enabling Technologies: Infrastructure for Collaborative Enterprises, 2003*. 2003: IEEE.
2. Cao, L. and B. Ramesh, *Agile requirements engineering practices: An empirical study*. IEEE software, 2008. 25(1): p. 60-67.
3. Asghar, A.R., et al., *Role of Requirements Elicitation & Prioritization to Optimize Quality in Scrum Agile Development*. INTERNATIONAL JOURNAL OF ADVANCED COMPUTER SCIENCE AND APPLICATIONS, 2016. 7(12): p. 300-306.
4. Asghar, A.R., et al., *The Impact of Analytical Assessment of Requirements Prioritization Models: An Empirical Study*. INTERNATIONAL JOURNAL OF ADVANCED COMPUTER SCIENCE AND APPLICATIONS, 2017. 8(2): p. 303-313.
5. Abbasi, M., et al., *Requirement Prioritization Techniques: An Empirical Study*. 2017.
6. Alam, S., et al., *Impact and challenges of requirement engineering in agile methodologies: A systematic review*. 2017.
7. Coram, M. and S. Bohner. *The impact of agile methods on software project management*. in *12th IEEE International Conference and Workshops on the Engineering of Computer-Based Systems (ECBS'05)*. 2005: IEEE.
8. Dingsøyr, T., et al., *A decade of agile methodologies: Towards explaining agile software development*. 2012, Elsevier.

9. Karlsson, J. and K. Ryan, *A cost-value approach for prioritizing requirements*. IEEE software, 1997. 14(5): p. 67-74.
10. Berander, P., K.A. Khan, and L. Lehtola, *Towards a research framework on requirements prioritization*. SERPS, 2006. 6: p. 18-19.
11. Butt, F.L., et al., *Optimized Order of Software Testing Techniques in Agile Process-A Systematic Approach*. International Journal of Computer Science and Information Security, 2016. 14(12): p. 509.
12. Bhatti, S.N., M. Usman, and A.A. Jadi, *Validation to the requirement elicitation framework via metrics*. ACM SIGSOFT Software Engineering Notes, 2015. 40(5): p. 1-7.
13. De Lucia, A. and A. Qusef, *Requirements engineering in agile software development*. Journal of Emerging Technologies in Web Intelligence, 2010. 2(3): p. 212-220.
14. Asghar, A.R., et al. *Impact and challenges of requirements elicitation & prioritization in quality to agile process: Scrum as a case scenario*. in *2017 International Conference on Communication Technologies (ComTech)*. 2017: IEEE.
15. Ahmed, A.R., et al., *Impact of Story Point Estimation on Product using Metrics in Scrum Development Process*. 2017.

AUTHORS PROFILE