# A Comparative Study for Prioritization Technique Identification for Software Project

Nur Razia Mohd Suradi[1], Mohd Fahmi Mohamad Amran[2], Nor Azliana Akmal Jamaluddin[3]
Department of Computer Science, Universiti Selangor, Bestari Jaya Campus, Selangor
[1]razia@unisel.edu.my, [2]fahmi@unisel.edu.my, [3]norazliana@unisel.edu.my

**Abstract—This paper presents an overview and comparison of available and effective prioritization technique used in handling requirement. Seven prioritization strategies have been stated: Analytic Hierarchy Process, Value Oriented Prioritization, Cumulative Voting, Numerical Assignment Technique, Binary Search Tree, Planning Game and B tree prioritization. Each of the technique being discussed based on its characteristics. The study aimed to examine each technique characteristics, limitation and the advantages.**

***Keywords—Requirement Prioritization, Requirement Analysis, Prioritization Technique***

## I. INTRODUCTION

A software project may comprise many or hundred requirements but in reality not all requirements could be implemented in same phase. Different people may see the importance of requirements prioritization from different viewpoints. Ref. [1] mentioned that a valid Requirement Engineering (RE) process must produce a core subset that balances customer needs, business values, cost and schedule; reflecting an agreement between customers and developers of what constitutes the current project.

An important aspect of managing the requirement engineering process is the choosing of a proper set of requirements from the gathering of competing and inconsistent expectations elicited from the numerous stakeholders in any project. This is because of too many requirements to fulfill compared to the available resources, deadline to rush, risks, market strategy and etc. Requirements prioritization has been accepted as one of the most important decision activities in the requirements engineering area supporting such decisions [2].

## II. PROBLEMS STATEMENTS

Although there are several empirical studies, there is still a lack of evidence of which prioritization approaches to be preferred, since different studies have resulted in different decisions. This research aims to propose a suitable technique for prioritizing requirements and focusing especially to aid higher learning formalize their requirements prioritization process. With this, the requirements prioritization seems by the stakeholders and developers provides the requirements engineer with a meaningful, grounded approach to the decision making process efficiently.

## III. RESEARCH QUESTIONS

The purpose of this study is to identify areas for further research in order to complement the existing techniques. To achieve this aim, 3 research questions (RQs) were formulated as presented below:

- What are the existing techniques used for prioritizing requirement?
- What is the limitation of each technique?
- What are the best techniques suitable to apply for various size of project?

## IV. RESEARCH QUESTIONS

The objectives of this research are:

- To examine the existing prioritization technique for software project.
- To identify the weakness of each prioritization technique.
- To determine the prioritization technique that applicable for different size of software project.

## V. RESEARCH OBJECTTVES

The contribution from this research:

- Provide best techniques for stakeholder to manage the selection of software requirement towards the success of a project.

## VI. LITERATURE REVIEW

### A. Requirement Engineering

According to Ref. [3], Requirement engineering (RE) is known as a structured process of elicitating, defining, negotiating, prioritizing and validating requirements of a system. Requirements Prioritization is one of the most vital activities of requirements engineering that is concerned with selecting the most important requirements out of an ample collected list of all significant or insignificant requirements.

Ref. [4] describes requirements prioritization as the process to decide the implementation order of the requirements for implementing the system or the process to determine the order of importance of the requirements to the stakeholders. Ref [5] defines requirements prioritization as the activity during which the most important requirements can be revealed.

Requirements prioritization has been known as a critical and crucial but inspiring activity for any product development. The pressure on time-to-market and being able to plan for successive release of the software product has posed many challenges to the software engineering process.

### B. Requirement Prioritization Technique

Research has been done to show the various techniques available in selecting the most critical requirement. Numerous methodologies occur to help requirements engineers select this core requirements subset. Most are based on some forms of prioritization. A systematic literature review done by Reference [6] discussed the most cited and utilized techniques of requirement prioritization in managing requirement of software as shown Figure 1.

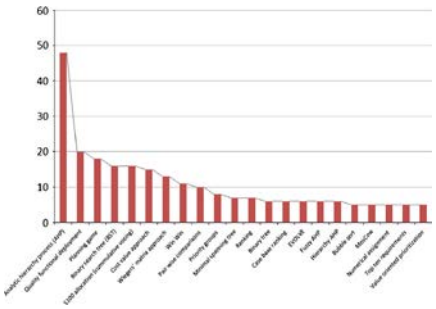


**Fig. 1: Data from Achimugu et al. (2014). The most cited and utilized requirement prioritization technique.**

Figure 1 above depicts the available prioritization techniques in handling requirements. Each of the technique has their own characteristics. Seven popular techniques are Analytical Hierarchical Process (AHP), Quality functional deployment, Planning game, Binary Search Tree, $100 allocation (accumulation voting), Cost Value approach and Wiegers Matric approach.

### C. AHP

The Analytic Hierarchy Process (AHP) is a systematic decision-making method that has been adapted for prioritization of software requirements Reference [7]. It is conducted by comparing all possible pairs of hierarchically classified requirements, in order to determine which has higher priority, and to what extent (usually on a scale from one to nine where one represents equal importance and nine represents absolutely more important). The total number of comparisons to perform with AHP are $n \times (n-1)/2$ (where n is the number of requirements) at each hierarchy level, which results in a dramatic increase in the number of comparisons as the number of requirements increases. Ref. [8] state that even though this is a good technique with many advantages like reliability. According to Reference [6], this technique cause major disadvantage of not being able to cater with environment having multiple stakeholders, hence it has to be modified in one way or another.

### D. Quality functional deployment

Ref. [9] had mentioned that Quality function deployment (QFD) is ‘‘an overall concept that provides a means of translating customer requirements into the appropriate technical requirements for each stage of product development and production (i.e., marketing strategies, planning, product design and engineering, prototype evaluation, production process development, production, sales)’’.

### E. Planning game

Ref. [10] introduces a prioritization method, named Planning Game, which is based on a combination of prioritization techniques. Planning Game is mostly used in agile projects. The idea of Planning Game is that it combines the numerical assignment technique and ranking technique together to perform the requirements prioritization. Requirements are first prioritized into three groups: (1) those without which the system will not function, (2) those that are less essential but provide significant business value, and (3) those that would be nice to have. After assigning the requirements into three groups, requirements are simply ranked in each group.

### F. Binary Search Tree

This method used for sorting elements that is mentioned by Reference [11] and known as binary search tree. A binary search tree is a tree in which each node contains at most two children. Ref. [12] introduce this technique to the requirements prioritization area for ranking requirements.

The idea of the binary search tree method for ranking requirements is that each node represents a requirement, all requirements placed in the left subtree of a node are of lower priority than the node

priority, and all requirements placed in the right subtree of a node are of higher priority than that node priority. First choose one requirement to be the top node. Then, select one unsorted requirement to compare with the top node. If that requirement is of lower priority than the top node, it searches the left subtree, but if that requirement is of higher priority than the top node, it searches the right subtree. The process is repeated until no further node needs to be compared and at that time the requirement can be inserted into the right position.

### *G. $100 allocation (accumulation voting)*

The 100-dollar test is a very straightforward prioritization technique where the stakeholders are given 100 imaginary units (money, hours, etc.) to distribute between the requirements [13]. The result of the prioritization is presented on a ratio scale.

A problem with this technique arises when there are too many requirements to prioritize. Assume, if you have 25 requirements, there are on average four points to distribute for each requirement. Ref. [8] encountered this problem when there were 17 groups of requirements to prioritize. Reference [7] claimed that another possible problem with the 100-dollar test (especially when there are many requirements) is that the person performing the prioritization miscalculates and the points do not add up to 100.

### *H. Cost Value approach*

Ref. [14] propose a Cost-Value approach based on the Analytic Hierarchy Process (AHP). However, the crucial difficulty with AHP is its use of pair-wise requirements comparisons. The pair-wise comparisons are time-consuming and suffer from explosive growth as the number of requirements increase [15] .Cost-Value approach uses the AHP method to compare requirements pair-wise according to their relative value and cost. Ref. [14] use two case studies to evaluate the Cost-Value approach. Nevertheless, they also find that the users find comparing all requirements in a pair-wise manner tedious. It is found that the Cost-Value approach contains a scale-up problem.

### *I. Wiegers matric approach*

This technique describes a semi-quantitative analytical approach that uses a simple spreadsheet model to help estimate the relative priorities for a set of product feature. Ref. [16] suggested that the requirement could be scaled and everyone involved must agree on the meaning of each level in the scale they use. Table 1 depicts two requirements prioritization scales.

**Table 1: Two requirement scales. Data from Wiegers (1999).**

| Names | Meanings |
|---|---|
| ***High*** | ***a mission critical requirement; required*** |
| ***Medium*** | ***for next release*** |
| ***Low*** | ***supports necessary system operations; required eventually but could wait until a later release if necessary a functional or quality enhancement; would be nice to have someday if resources permit*** |
| ***Essential*** | ***the product is not acceptable unless these requirements are satisfied*** |
| ***Conditional*** | ***would enhance the product, but the product is not unacceptable if absent*** |
| ***Optional*** | ***functions that may or may not be worthwhile*** |

Although there are seven popular requirement techniques discussed above, there is also techniques which involve between user Reference [17] addressed multi-aspects based requirement prioritization techniques for value-based software (VBS) development. VBS combine between aspect requirement and business aspects requirement. Figure 2 shows the Technical expert focused such as risk value, cost ,speed and time and Figure 3 represents VBS on business aspects requirement.

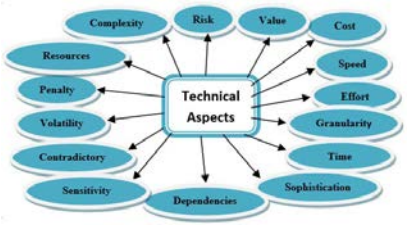


**Fig. 2 – Technical Aspects**


**Fig. 3 – Business Aspects**

## VII. CONCLUSION

Requirements prioritization is known as a challenging decision-making activity that requires support. Many approaches for prioritization of software requirements are presented in the literature. Also it helps the stakeholder to choose the best technique of managing the requirement especially the crucial requirement. With this, the requirements prioritization seems by the stakeholders and developers provides the requirements engineer with a meaningful, grounded approach to the decision making process efficiently.

## Acknowledgment

The authors appreciate Universiti Selangor (UNISEL) for supporting this research as it is in an early stage.

## References

[1] Azar,J., Smth, R. K. and Cordes, D. IEEE Value-Oriented Prioritization: A Framework for Providing Visibility and Decision Support in the Requirements Engineering Process. 0740-7459/07. 2007.

[2] Ngo-The, A., and Ruhe, G.'Decision Support in Requirements Engineering', in Engineering and Managing Software Requirements, ed. Aurum, A., and Wohlin, C., Springer Verlag, Berlin, Germany, pp. 267-286, 2005

[3] Nuseibeh, B., & Easterbrook, S. Requirements Engineering, "A Roadmap", 2000 22nd International Conference on Software Engineering (ICSE 2000), 2000.

[4] Firesmith, D. Prioritizing requirements. Journal of Object Technology, 3(8), 35-47. 2004.

[5] Sommerville, I. Software Engineering, Addison-Wesley.

[6] Philip Achimugu, Ali Selamat, Roliana Ibrahim, Mohd Naz'ri Mahr (2014). A systematic literature review of software requirements prioritization Research, Journal Information and Software Technology 56 (2014) 568–585, 2006

[7] T.L. Saaty . The Analytic Hierarchy Process: Planning, Priority Setting, Resources, Allocation, McGraw-Hill, Inc. 1980.

[8] Regnell B, Höst M, Natt och Dag J, Beremark P, Hjelm T . "An Industrial Case Study on Distributed Prioritisation in Market-Driven Requirements Engineering for Packaged Software. Requirements Engineering 6(1):51-62, 2001

[9] Sullivan, L.P. Quality function deployment. Quality Progress 19 (6), 39–50 ,986b

[10] Beck K `Extreme programming explained: embrace change. Addison-Wesley Professional, 2000.

[11] A.V. Aho, J.E. Hopcroft, J.D. Ullman, Data Structures and Algorithms. Addison-Wesley, Reading, MA, 1983.

[12] Karlsson J, Wohlin C, Regnell B , "An evaluation of methods for prioritizing software requirements." Inf Softw Technol 39(14):939–947, 1998.

[13] Leffingwell D, Widrig D , "Managing Software Requirements – A Unified Approach". Addison-Wesley, Upper Saddle River, 2000

[14] J. Karlsson, K. Ryan, Prioritizing requirements using a cost-value approach, IEEE Software 14 (5) (1997) 67–74.

[15] Karlsson, J.,Wohlin, C. and Regnell, B. An evaluation of methods for prioritizing softwar requirements, Information and Software Technology, 39(1998), 939-947.

[16] Wiegers, K. First things first: prioritizing requirements, Software Development, 7(9) (September 1999),

[17] http://www.processimpact.com/pubs.shtml#requirements

[18] Sher F, Jawawi DNA, Mohamad R, Babar MI. Multi-aspects based requirements priortization technique for value-based software developments. In Proceedings - 2014 International Conference on Emerging Technologies, ICET 2014. Institute of Electrical and Electronics Engineers Inc.2015. p.1-6. 7021007. Available from:10.1109/ICET.2014.7021007