A Review on Software Requirement Parameter Optimization

Anika Bisht[1], Madan Kushwaha[2],
[1]M.Tech Scholar, [2] Assistant Professor,
[1,2]Dept of CSE Bansal Institute of Engineering and Technology.

Abstract: Software Engineering (SE) is one of the freshest engineering domains emergent and developed within the past four decades or so. Still, a huge amount of research work has gone into shaping it the way we see it functioning today. As a result, we have an remarkable knowledge repository to work with in the form of software development models, the software engineering theories and practices etc. The aim of SE is to create software products, services or their artifacts in order to meet the requirements posed by stakeholders or buyers while meeting quality constraints imposed on them. In order to meet both these objectives, any software development derives its purpose and meaning from the requirements posed by various stakeholders. Requirement Prioritization and Classification is a very critical but often neglected area of requirement engineering. Experience has shown that without proper prioritization and classification of requirements presented by various stakeholders, the end product usually fails to meet its objectives optimally.

Keywords: Fuzzy Logic, Requirement Prioritization, Requirements engineering, rRCF

1. Introduction:

Software Engineering (SE) goes for making programming items or their relics in a manner that these will meet the prerequisites postured by the partners while it is satisfying the quality of requirements forced on them. By keeping in the mind the end goal to go through both these destinations and any product advancement infers its significance and motivation from the prerequisites postured by different partners. Prerequisite Engineering is a set up area of information inside of programming designing which sets up practices and standards for powerful necessity elicitation, displaying, detail, documentation and so forth. One imperative however regularly ignored routine of programming necessity building is prerequisite prioritization. A few necessity prioritization procedures had been introduced by the creators. These systems are both subjective and quantitative in their own tendency. Many understood prerequisite prioritization procedures to incorporate Numerical Assignment, AHP (Analytical Hierarchy Process), Ranking, Cumulative Voting, Requirement Triage, Theory W, Wieger's Method and so forth. What's more, there are a few different methods which we should examine in this paper. Necessity prioritization empowers us to comprehend the importance of prerequisites opposite the framework to be created and among prerequisites also. With necessity prioritization, we can distinguish the center zones which require the vast majority of our consideration so as to build up an item which ideally meets the prerequisites of the partners. In the vast majority of the circumstances, because of spending plan and time limitations, it gets to be difficult to actualize every one of the prerequisites postured by partners. Likewise the way of numerous undertakings is such that necessities are actualized in an organized situation. In such situations, optimization of prerequisite is required [1]. We may organize the prerequisite to acknowledge which necessities may be either deferred or changed so that the other earnest necessities may be actualized and to what degree. We could likewise utilize necessity optimization to figure out which of the prerequisites to be actualized in later stages or before stages. We had been working with a few supported undertakings amid our examination. These ventures are confronted with both of the aforementioned circumstances. We have discovered it critical to organize necessities in their actual sense by keeping in mind the end target to building up a significant plus fruitful item.

2. Related Work:

In order to build on a cost-effective quality software, it's certainly required to pick out the right requirements from the set of each and every requirement. It would be more advantageous if all requirements are grouped release wise. In this manner, we will be able to decide which requirements are essential to be focused in any specific release. Prioritize requirements based on some parameters is one way to categorize them. But accomplishing such goal is not generally an easy task because there are several issues that should be directed as it should be. These all issues consist of the assortment of stakeholders or buyers whose recommendation should be worn out to give priority (value). Likewise, an aspect, which shall be worn out as parameters to allocate priorities necessities to be solemnised. Matters like organization's system, market value and stakeholder's personalities as well as their agendas need to be catered too while performing the requirement prioritization. Working in situations like this needs numerous trade-offs to be reached amongst the diverse stakeholders or buyers in incompatible environments. Then only we can expect to turn up to a proposed and an agreed system which put up to the needs of stakeholders. To support on decision making procedures with reference to each and every aspect of system development is the major objective of prioritization. This decision-making procedure is an important thing in all indices of daily life from the very complex things such as politics, military, economics, administration to as trouble-free as running your day-to-day life's responsibilities. In SE, the primary importance of prioritization is that it is used in almost every phase its software project life cycle i.e., select a project, finalize the requirements, design and develop of modules, testing, implementation, and even in post implementation activities. Though, the key constraint in optimization procedure is its difficulty as subject develops its base criteria. In a very simple form, project demands no prioritization but with very simple changes, it becomes very difficult to make a decision based on beyond one criterion. When there are number of alternatives reaches thousands, then the decision making and prioritization becomes almost impossible. This gives detail that manual and human based prioritization is not the solution for advanced complex systems. There are number of prioritization techniques are available that deal with the different level of complexity. These techniques are worn out in different domains for supporting the decision making.

2.2 Benefits of Prioritization

Followings are some key factors that advocate including prioritization process in requirement engineering:

2.2.1 Constraint Driven strategy:

Time and budget constraints usually do not allow entertaining all requirements. Therefore, project manager can use prioritization as a tool to help him in selection of those requirements that can be implanted in a certain time and budget constraints.

2.2.2 Planning the Releases:

Prioritization helps the stakeholders to decide the type of requirements of system and categorize them in the optimal requirements sets. Based up on these optimal sets a project manager can also plan the releases of product. In its normal flow; the system's core requirements will be implemented in the first release and based up on their priority values next group of the requirements will be implemented in the successive releases.

2.2.3 Balancing benefits against cost:

Key stakeholders or buyers should identify the related requirements and their corresponding benefits before starting the implementation. In the same way, each of these requirements shall be evaluated for the implementation cost. The prioritization is based on the benefit as well as the related cost; may help business to balance their benefits of each and every requirement against the cost of implementing it.

2.2.4 Negotiating conflicts:

The viewpoints of the stakeholders for most of the requirements are different which arise conflicts between the stakeholders. Requirements prioritization every so often involves negotiation procedures to handle the contradictory requirements. Therefore, it also supports in resolving conflicts as well as disagreement amongst stakeholders.

2.2.5 Better Understanding:

To some extent requirements optimization is an expensive activity, because it requires time, experts' opinions meeting, stakeholders' involvement, etc. It also helps all stakeholders to recognise all the requirements further carefully. Each requirement is assessed in detail and their implementation feasibility is tested out at the prioritization stage. If specific requirements are found not-feasible then the alternatives can be discovered.

2.3 Prioritization Parameters:

There must be a few parameters that shall be worn out to assign values to each requirement (Requirements should be prioritized objectively). Given below are some chief parameters for the prioritization:

2.3.1 Time:

In any project management process time is an important factor. So, it is essential to consider it when planning for the releases of the software. For the most part, time constraint helps in the identification of those requirements that could be distract in the short duration as well as in long duration.

2.3.2 Cost:

Estimation of cost of requirement requires previous experience. Generally developing organization estimates the cost which is regularly articulated in terms of the man-hours. The prioritizing requirements relating to cost supports project to be within the given budget.

2.3.3 Penalty:

In case requirement is not fulfilled penalty can be viewed as a negative point that can be launched. In normal way, the core requirements have higher PV (penalty value) than those that are in 2ndlevel to the core requirements.

2.3.4 Risk:

To cope with both internal and external risks Risk management is very useful; thus, it should be considered when planning the requirements. Based on risk impact for each and every requirement, the risk level of project should also be considered.

2.3.5 Functionality

Functionality in Requirement Engineering relates to the work/s that proposed system has been allotted to accomplish. Any system to be established inserts inside itself numerous functionalities which could be visualized as the programmed representation of several requirements posed by the stakeholders or buyers. Just as priority of the requirements fluctuates, so does priority of the functionalities of system. That is the major reason why both in the linear or iterative approaches, our focus is always to implement the high priority functionalities first and after that the lower ranked one. There are other parameters also that can be take into account when prioritizing the requirements like competitors, financial benefit, release theme, strategic benefit, ability to sell, competence/resources, etc.

2.4 Requirement Prioritization Techniques: an Overview

There are various requirement prioritization procedures as mentioned in literature review. Though, no assessment of these procedures has been made so far so as to their relevance and utility may be finalised. We have widely analysed the existing mechanisms as mentioned earlier [12, 13, 19]. In this segment, we will be giving a comprehensive overview of several requirement prioritization procedures.

2.4.1 Analytical Hierarchy Process (AHP)

Analytical Hierarchy Process is a relative evaluation based measurable system to organize the prerequisites for programming an item. In an event we have n number of necessities, Analytical Hierarchy Process makes n x (n-1)/2 correlations at every pecking order level. All the things are considered, we are generally working with the prerequisites which have many goals. Analytical Hierarchy Process fills in as effectual method in this sort of situations by creating a pair savvy relationship to determine the relative esteem as well as the cost of every necessity against the other one. All these expansive number of investigations makes strategy less compelling as the increment in number of correlations consistently occurs at the rate of O(n2). Analytical Hierarchy Process is viewed as five stage approach.

1. Establishing completeness of the requirements.
2. Applying pair-wise comparison method to assess relative value.
3. Applying pair-wise comparison method to assess relative cost
4. Calculating each applicant requirement's relative value and the implementations cost, and then plotting each on a cost-value diagram.
5. Using cost-value diagram as map for analysing candidate requirement.

Countless have been made in past to decide adequacy of Analytical Hierarchy Process for prerequisites the prioritization.

Karlsson [13] has made various different studies which have demonstrated viability of this method in modern settings. In meantime, some different studies [15] have discussed Analytical Hierarchy Process as being troublesome, less effective plus tedious. Analytical Hierarchy Process can be taken into account as a profoundly refined as well as complex procedure which can build up the prioritization at the level of individual necessities. Endeavors have been made to diminish the quantity of examinations. In any case, this has constantly improved the room for give and take. As we would like to think, this tradeoff is essential since a few correlations might very be required.

2.4.2 Cumulative Voting (CV)

Sometimes Cumulative Voting is also referred as 100 $ test or 100 point technique, it resembles very much with the voting mechanism of a brainstorming sessions. Each stakeholder has been provided 100 points that he or she can spread out amongst the requirements as they seem like it fits. It seems straightforward mechanism but it can become complex as number of requirements increases or the stakeholders involved increases. This scheme has some drawbacks related with it. First of all, this scheme treats all requirements as equal opportunity candidates. Then, element of bias can never be over governed. It has been observed that in subsequent voting, stakeholder or buyer allocate more votes to their favourite requirements with the intention to move them up. Many researchers [24] have indicated out shortcomings in this mechanism. Cumulative voting (CV) technique can be considered as one which is complex in nature but attempts to prioritize the requirements at their discrete level.

Table 2.1: Sample Cost- Value Diagram for Analytical Hierarchy Process

	SR-1	SR-2	SR-3	SR-4	SR-5	SR-6	SR-7	SR-8	SR-9	Scores	Product	Ratio
SR-1	1	8	1/5	3	1	2	2	3	1	0.1373	1.5427	11.234
SR-2	1/8	1	1/5	1/7	1/7	1/7	1/7	1/9	1/9	0.0146	0.1549	10.5917
SR-3	5	5	1	1	2	1	3	1	1	0.1717	1.9647	11.4415
SR-4	1/3	7	1	1	1/2	1/2	3	1/2	1	0.0968	1.0743	11.0955
SR-5	1	7	½	2	1	3	3	1	1/3	0.1259	1.4065	11.1681
SR-6	1/2	7	1	2	1/3	1	1/3	1	1	0.0911	0.9550	10.4813
SR-7	1/2	7	1/3	1/3	1/3	3	1	3	2	0.1155	1.2740	11.0301
SR-8	1/3	9	1	2	1	1	1/3	1	1/6	0.0887	0.9134	10.2961
SR-9	1	9	1	1	3	1	1/2	6	1	0.0887	1.7547	11.088

2.4.3 Numerical Assignment (NA)

The most well-known optimization method which is similarly simple to utilize is the Numerical Assignment method. In initial step, requirements are ordered in to diverse gatherings. These necessities are given to every partner. Each and every necessity inside of these gatherings is distributed out a number on size of 1 to 5 by discrete associates. The last calculating with the intention to place is dictated normal of all positioning given to every prerequisite by each partner. This procedure due to its convenience has likewise been proposed by IEEE Std. 830-1998. Since necessities are initially ordered into gatherings and afterward organized therefore, we can express that this Numerical Assignment method does not organize the prerequisites at the level of singularity. Rather one level of the deliberation is presented. In spite of its wide materialness, this strategy additionally represents a few issues. Clear definition of the gatherings is one noteworthy disadvantage. Another problem is that even with the clear definitions, associates will tend to put enormous majority of their prerequisites into basic gatherings due to their inclination. Another truth that we have to be careful about the inside of every gathering, every one of necessities are at first at the equivalent need level. The majority of these downsides in numerical task procedure have been very much recorded in [24].

2.4.4 Ranking

The suitable method in nature where a solitary associate is included is ranking. On the off chance that there are n number of the prerequisites, these necessities are placed from 1(most huge) to n (slightest noteworthy). This placing is select in its temperament on the grounds that prerequisites are not placed respecting dissimilar necessities comparable to the instance of Analytical Hierarchy Process or total voting. Different methods like air pocket sort, brisk sort or twofold inquiry procedures can be utilized to accomplish this positioning. There are the two noteworthy downsides associated with this process. To start with an important problem, it can bring about greater number of disagreements than declarations when associated in a domain of several associates. The 2nd downside is that the necessities are realized and situated in segregation. Effect of one necessity over the other does not take for granted any part in overall prioritization. Since the necessities can have different measurements to them so scientists have conceived an instrument of joining these measurements and ascertaining a mean need for every prerequisite [25]. This adjustment has its own particular confinements and additionally has been appeared in [25].

2.4.5 Top-Ten requirements

Top ten requirement technique prioritizes only the most significant requirements into a set of top-ten from a larger set of requirements. Selection of the most important requirements is subjective to the project environment and so it can be erroneous if based on human judgment. Since we create only a set of top-ten requirements, no prioritization within this set takes place. This can be termed as a shortcoming in many situations. The technique can be applied in conjunction with other techniques to achieve better results. According to Lausen, it is mostly helpful in situations where there are multiple stakeholders with uniform or similar significance.

2.4.6 Theory

The fundamental defender of this hypothesis is Dr. Barry Boehm who presented this idea [13] in 1989. Prominently known as Win-Win demonstrate, this procedure depends vigorously on transaction to determine any distinctions of conclusion among different partners. The transactions are led in a manner that every partner is in a "Win" circumstance. The standards of this system are advancement taking into account predefined arrangement, hazard appraisal and danger taking care of. In this system, clients are requested that rank their necessities before genuine transactions begin. Clients are asked to painstakingly consider which prerequisites they are willing to arrange and which they are definitely not. Hypothesis W has been a dynamic region of exploration among researchers which has been connected in necessity designing as well as in different spaces of programming building. Hypothesis W is a noteworthy constituent of Value Based Software Engineering (VBSE) plan and standard too.

2.4.7 Planning Game (PG)

This specific requirement prioritization technique is very suitable to extreme programming. In this specific technique, requirements are prioritized in consultation with customers. This is a variation of numerical assignment technique as discussed. However it provides more flexibility than numerical assignment where users are queried to significantly divide the requirements into three groups. Some other novel and innovative techniques to emerge recently include Requirement Triage (RT) [13] and Wieger's Method (WG) [14]. In requirement Triage, each requirement is prioritized relative to the resources that are necessary to meet that specific requirement. In this way, a subset of requirements is selected which can optimize the probability of success of product while using the allocated resources efficiently. In Wieger's method, the priority of each requirement is set by determining the utility of that requirement to the customer as well as penalty incurred by the customer if that requirement remains unfulfilled. In this section, we have presented a brief overview of existing requirements prioritization techniques. In this next section, we shall present a theoretical evaluation of these techniques as well as present the brief idea of our proposed and implemented approach.

2.5 Literature Review:

Necessity Engineering is one of the soonest and extremely basic periods of the programming designing. RE (Requirement Engineering) as a learning stream is fundamentally gone for securing, displaying and documenting prerequisites for product item. RE is one of a kind orders as in it consolidates the ideas of building as well as of human and sociologies. At times, alluded to as necessities investigation, RE is dealt with as a sub control of framework designing and programming building. Necessity designing means to characterize decisively the prerequisites that should be met. This is not a standard undertaking. As indicated by Fred Brooks, choosing what should be manufactured is the most troublesome piece of programming improvement. We can envision one programming prerequisite as one archived need that product item ought to fulfill. Typically, necessities are delegated either by way of procedure based and item based or useful and non-practical prerequisites. Programming prerequisite can best be characterized as the portrayal of framework usefulness alongside its quality is relate to.

The Prerequisite prioritization is the following coherent errand as soon as necessities have been elicitated and legitimately dissected. By and large, it is truly hard to meet every one of the prerequisites that have been given by different partners. The vast majority of the times, elicitated necessities are dubious, clashing or out rightly false. Over timeframe, as our comprehension of the framework turns out to be more clear, the prerequisites begin accomplishing their real or particular shape. Likewise, by and large, necessities are executed in an amazed manner. In such circumstances, it gets to be critical to organize the prerequisites in an organized request to add to the framework in more practical way. This assignment turns out to be much more troublesome when performed right on time in the lifecycle. One of the best issues of programming specialists is advancement of such an item which doesn't fulfill the needs and desires of partners. In this way, numerous different analysts underlined upon the hugeness of necessity prioritization. Prioritization of necessities is a critical issue where expressed that prioritization of prerequisites was one noteworthy subject of dialog amid the overview that they attempted.

There are different systems for organizing necessities. Some significant strategies are Binary Search Trees, Analytical Hierarchy Process (AHP), 100 focuses strategy, arranging diversion, numerical task method and hypothesis W 20 and so forth. The accord of these studies is that the venture's prosperity or the disappointment is emphatically dictated by how successfully, we may organize the necessities.

The Computational Intelligence and the delicate registering are built upon the methods which have determined numerous true issues. These systems incorporate Fuzzy Logic, Evolutionary Computing, and Artificial Neural Networks and so forth. Fuzzy logic is a system fixated on fuzzy set hypothesis. Therefore, it is taken into account as an augmentation of outdated set hypothesis. Idea of the fuzzy sets as presented by Lotfi Zadeh may be taken into account as speculation of the traditional sets which were fresh in their temperament. Motivation behind the fuzzy logic is to lessen the many-sided quality of current arrangements as well as in addition expanding the availability of control hypothesis. Computational insight based procedures including fuzzy logic have been broadly utilized as of late to handle numerous genuine issues. Fuzzy logic has additionally discovered its way in programming designing where it has most as of late been utilized as a part of exertion estimation, programming undertaking closeness, programming advancement, venture assessment, programming development and so forth. Fuzzy logic has been utilized as a part of necessity designing also for different assignments. Chengdong Li et al. displayed a new methodology for the utilization of former learning and tested information in the fuzzy frameworks. Some other late advancements in the space of fuzzy logic have likewise exhibited new vistas of examination in programming designing. What is apparent subsequent to concentrate broadly is the way that scientists in programming designing need to apply counterfeit consciousness in different spaces of learning of Software Engineering (SE) to recommend these methods which can come out and can insightfully create the proficient results. In this thesis, we have also proposed another use of the fuzzy logic in programming building. We have also projected to utilize the fuzzy logic in the space of necessity building. We have also proposed acquainting fuzzy logic with decide the need of necessities. The following area is given to elaborate this proposed procedure.

In this thesis, another application of fuzzy logic in domain of the requirement engineering has been proposed. We propose the introducing fuzzy logic to establish priority of requirements with an integration of Bayesian network.

3. Conclusion:

Requirement optimization is a significant activity of requirement engineering phase in the software development. There are various prerequisite optimization methods in literature and practice. However, no substantial relative evaluation of such procedures has been made so far. In this paper, we conclude additional application of fuzzy logic in the domain of the requirement engineering. We suggest introducing fuzzy logic to determine priority of requirements with integration of Bayesian network.

References:

[1] F. Brooks, No silver bullet: Essence and accidents of software engineering, IEEE Computer, vol.20, no.4, pp.10-19, 1987.

[2] J. Karlsson and K. Ryan, Supporting the selection of software requirements, Proc. of the 8th International Workshop on Software Speci_cation and Design, 1996.

[3] J. Karlsson and K. Ryan, A cost-value approach for prioritizing requirements, IEEE Software, vol.14, no.5, pp.67-75, 1997.

[4] X. Liu, C. C. Veera, Y. Sun, K. Noguchi and Y. Kyoya, Priority assessment of software requirements from multiple perspectives, Computer Software and Applications Conference, vol.1, pp.410-415, 2004.

[5] L. Fellows and I. Hooks, A case for priority classifying requirements, The 3rd International Conference on Requirements Engineering, pp.62-65, 1998.

[6] E. Yourdon, Death March Projects, Prentice Hall, 1997.

[7] M. Lubars, C. Potts and C. Richter, A review of the state of the practice in requirements modeling, Proc. of the IEEE Intemational Symposium of Requirements Engineering, pp.2-14, 1993.

[8] V. Ahl, An Experimental Comparison of Five Prioritization Methods, Master Thesis, School of Engineering, Blekinge Institute of Technology, Ronneby, Sweden, 2005.

[9] T. Saaty, The Analytic Hierarchy Process: Planning, Priority Setting, Resource, Allocation McGraw-Hill, New York, 1980.

[10] T. L. Saaty and G. Hu, Ranking by eigenvector versus other methods in the analytic hierarchy process, Applied Mathematical Letter, vol.11, no.4, pp.121-125, 1998.

[11] F. Hartwich, Weighting of agricultural research results: Strength and limitations of the analytic hierarchy process, Research in Development Economic and Policy, Discussion Paper, Grauer Verlag, Stuttgart, no.9, 1999.

[12] F. Hivert, J. Novelli and J. Thibon, The algebra of binary search tree, Theoretical Computer Science, vol.339, no.1, pp.3-10, 2005.

[13] L. Xiang, K. Ushijiam, T. Zhao, T. Zhang and C. Tang, O(1) time algorithm on BSR for constructing a random binary search tree, Proc. of the 4th International Conference on Parallel and Distributed Computing, Applications and Technologies, pp.599-602, 2003.

[14] I. Al-furaih, S. Aluru, S. Goil and S. Ranka, Parallel construction of multidimension binary search tree, IEEE Trans. on Parallel and Distributed Systems, vol.11, no.2, pp.136-148, 2000.

[15] C. Lee, L. Hung, M. Chang, C. Shen and C. Tang, An improved algorithm for the maximum agreement subtree problem, Information Processing Letters, vol.94, no.5, pp.211-216, 2005.