

# 4A Frameworks for Requirements Prioritization

Persis Voola A

Department of Computer Science  
Adi Kavi Nannaya University  
Rajahmundry Andhra Pradesh, India

Vinaya Babu

Department of Computer Science  
Jawaharlal Nehru Technological University  
Hyderabad Andhra Pradesh, India

## ABSTRACT

Prioritization decisions in general aim at conducting assessment of several alternatives that are characterized by multiple conflicting attributes, which are intertwined by the competing preferences of multiple assessors. These assessments personifying various forms of ambiguity such as uncertainty, ignorance, vagueness and fuzziness have to be aggregated to generate reliable collective priorities. The objective of this paper is to introduce 4A prioritization frameworks with alternatives at the centre surrounded by the four facets: Attributes, Assessors, Ambiguity and Aggregation. Elements constituting the framework are discussed in a general context and then related to software requirements. The frameworks introduced have confronted a wide scope of further research.

## Keywords

Requirements prioritization, prioritization attributes assessors, ambiguity, and aggregation.

## 1. INTRODUCTION

Establishing priorities is a decision making problem that happens in everyday life. When several tasks have to be done and practically it is not possible to do all of them because of the restricted resources, prioritization comes into picture. It is narrowing down the big list of actions into a smaller and implementable list, which assists in sensing what will help or hinder in accomplishing a goal. If this kind of sensing is not done, the result might be an unsuccessful conclusion. The applications of prioritization are diverse, and can be found almost in any real world situation ranging from simple day-to-day activities to complex marketing strategies evaluation.

Prioritization decisions involve the assessment of several alternatives, according to their outcomes. These outcomes are affected along dimensions of multiple conflicting attributes and multiple stakeholders [2]. Prioritization as a decision making problem has to deal with ambiguous nature of human judgment. This ambiguity may arise for 3 reasons: inadequate understanding, incomplete information and undifferentiated alternatives [44] as found by Raanan Lipshitz, 1997 et al through post analysis of 102 reports related to decision making under uncertainty. Individual assessments obtained thus have to be aggregated to form collective decisions.

The basic elements of prioritization: alternatives, attributes, assessors, ambiguity and aggregation mentioned in a general context are more closely connected to Requirements Prioritization (RP) of a software system, application or product. Alternatives are requirements for prioritization. RP enables to select the right set of requirements for implementation out of a larger set with the aim of maximizing customer satisfaction simultaneously fulfilling the restrictions imposed by budget, delivery deadlines, personnel availability and many other factors. The requirements for prioritization are characterized by

multiple *attributes* of conflicting nature. Multiple stakeholders who claim to have a stake in the product act as *assessors* of the requirements' priorities. Assessments come with various forms of *ambiguity* like uncertainty, ignorance, imprecision and vagueness. Individual assessments obtained thus have to undergo *aggregation* to generate reliable requirements' priorities.

These elements of RP are designated as 4A (read as four A's): Alternatives at the center surrounded by four facets Attributes, Assessors, Ambiguity and Aggregation. An RP framework as shown in Figure 1 must be flexible enough to integrate the 4A in a consistent manner.

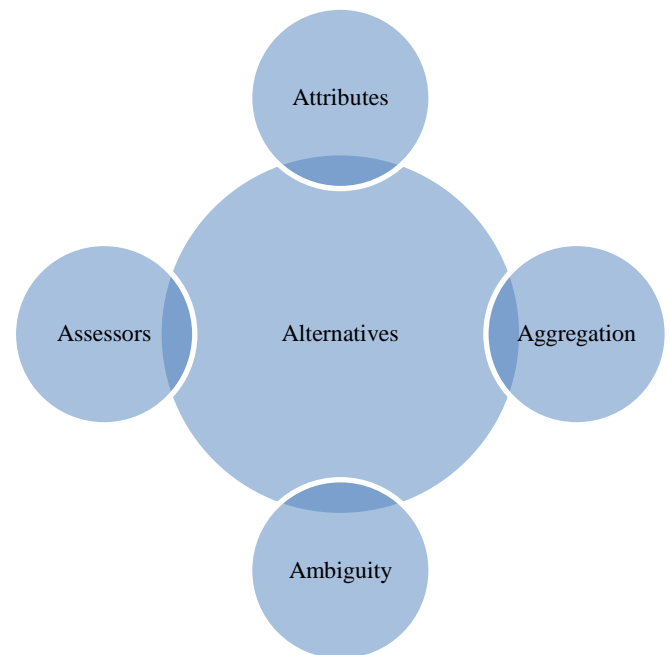


Figure 1.5 4A Framework for Prioritization

Section 2 discusses the four prioritization elements in a general context. Section 3 discusses the prioritization elements in the context of software requirements. Section 4 introduces RP frameworks that are constituted by 5A. Section 5 concludes with an outline on future work.

## 2. 4A IN A GENERAL CONTEXT

This section discusses the significance of decision alternatives and how decision alternatives for prioritization have an impact on several attributes. Further discussion is focused on participation of multiple stakeholders in the prioritization process. Ambiguity, the pivotal element of human judgment is presented with two real world examples. This section concludes with the list of aggregation methods.

## 2.1 Attributes: General context

Ben Franklin [3] recognized the presence of multiple attributes in every day decisions over 200 years ago. For example if you want to buy a house, cost is not the only attribute but several other attributes like locality, dimensions, current market condition, loan facility etc impact the decision to purchase. These are conflicting in nature since some are positively oriented attributes and some others are negatively oriented. For example, cost is negatively oriented whereas a dimension is positively oriented. Appropriate trade-off decisions facilitate to maximize satisfaction of the buyer by determining the best house out of a set of houses in terms of low cost and broad dimensions.

Decision problems that prioritize alternatives influenced by multiple conflicting attributes are Multi Attribute Decision Making (MADM) problems. Attributes are also called criteria, aspects, characteristics or objectives. MADM problems were popular decision methodologies in business, sciences and engineering domains. Yoon and Gytal, 1989 defined MADM as 'technical decision aids for evaluating alternatives which are characterized by multiple conflicting attributes'.

Keeney and Raiffa, 1976 characterized a MADM problem in the following way:

- A set of decision alternatives for evaluation.
- A set of criteria or attributes along which the alternatives are to be evaluated.
- The ranking of alternatives in the order of preference.

Roy 1985, Ahire and Rana 1995, Jacquet-lagrece and Siskos, 2001 pinpointed the three elements of MADM in the same style as defined above.

Once alternatives are evaluated, it is possible to find the best alternative or a set of alternatives in the order of preference. Eom, 1989 describes how MADM models have been used in a variety of practical applications such as: employee appraisal, organization evaluation; resource allocation; engineering design evaluation; supplier evaluation; urban and community planning and credit analysis. Fred S Azar, 2000 recommended that the attributes characterizing the MADM problem should be the most relevant ones and also measurable in a meaningful way [4]. The attributes should further satisfy the constraint of mutual exclusion, i.e. the attributes should be viewed as independent entities among which appropriate trade-offs may later be made.

## 2.2 Assessors: General Context

MADM problems are even more complex when the alternatives characterized by multiple conflicting attributes are evaluated by multiple decision makers with competing personal preferences (Hwang and Lin, 1987; Chen and Hwang, 1992). This sort of multiple person MADM problems are recommended in many practical situations since group decisions are less prone to error rather than individual decisions (Evans, 1984). Seaver, 1976 concluded that group judgment is more accurate than individual judgment primarily due to decrease in error variance around the true value [41].

It is important to note that all decision makers are not equally important. The varying influence on the decisions has to be discriminated by dividing them into groups carrying relative weights. Mendelow's power interest grid [25] categorizes stakeholders into 4 groups: High Power-High Interest, High Power-Low Interest, Low Power-High Interest and Low Power-

Low Interest. Another grouping of stakeholders is as exerting high, medium and low impact on the system [26].

## 2.3 Ambiguity: General Context

Ambiguity is a well found concept in the literature on decision making (Kahneman, Slovic, & Tversky, 1982; March & Olsen 1976). In a multiple person MADM problem, decision makers are humans whose judgments are based on intuition, experience, intelligence, assumptions, opinions and beliefs. Human judgment inevitably involves various types of ambiguities such as uncertainty, ignorance and fuzziness. The manifold nature of uncertainty is depicted in Table1 [44], gathered from the decision making literature between 1960 and 1990.

**Table1. Different Interpretations of Uncertainty**

Author	Term	Interpretation
March and Simon (1958)	Conflict	Absence of arguments which clearly favor a particular course of action.
Arrow (1965)	Risk	A positive function of the variance of the probability distribution of expected +ve and -ve outcomes.
Terreberry (1968)	Turbulence	Unpredictable changes in system-environment relations.
Weick (1979)	Equivocality	The multiplicity of meanings which can be imposed on a situation.
Anderson (1981)	Uncertainty	A situation in which one has no knowledge about which of several states of nature has occurred or will occur.
Hogarth (1987)	Ambiguity	Lacking precise knowledge about the likelihood of events.

Precise judgment using a single number proves to be difficult, sometimes unacceptable and has the probability of being incorrect [5]. Even if an expert makes an attempt to convert a subjective judgment or a probability distribution or an incomplete piece of information into a single number, it might result in information loss or distortion. Two real world examples discussed below depict the imprecise nature of human judgment.

### 2.3.1 Student Performance in Exam

Universities and educational institutions are setting away old, stagnant and faulty evaluation system of giving marks and moving towards grades. In the marking system, students are evaluated precisely on a scale from 0 to 100. It is likely that the student who scored 73 marks would feel superior to the one with a 72. But the difference between a 72 and a 73 is statistically insignificant, especially if what is being graded is subjective material, such as essays.

The research results indicate that the evaluation is indeed imprecise as the standard of examination differs from subject to subject and evaluation of answer script differs from examiner to examiner. Further, evaluation of an answer script by the examiner differs from time to time [15]. Therefore, grades were introduced acknowledging the imprecision during evaluation. University Grants Commission[16], being the authorized agency

to maintain the standard of higher education and co-ordination among the universities in India, advocates for a 7 point scale of grades as given in Table 2. By using the broader grades, it is possible to be more realistic in marking [17]. Hence, it is reasonable to treat the grade A student superior to grade B student.

**Table 2. Seven Point Grading**

Grades	Grade Point	Percentage Equivalent
O-Outstanding	5.50-6.00	75-100
A-Very good	4.50-5.49	65-74
B-Good	3.50-4.49	55-64
C-Average	2.50-3.49	45-54
D-Below Average	1.50-2.49	35-44
E-Poor	0.50-1.49	25-34
F-Fail	0-0.49	0-24

### 2.3.1 Employee Performance Appraisal

Performance appraisal is the assessment of an employee's job performance using one of the rating scales. The objective is to align the employee's performance and results with the organizational goals. Traditional rating scales based on precise values were in popular practice, which were simple and easy to use. But these scales suffered from rating errors as evaluations varied too widely from rater to rater. One manager might assign a ranking of '1' to perfection, whereas another manager might rate '1' as a much lenient standard. This kind of assessment produces distorted and unreliable data. So, a better method of rating employees' performance is devised by Smith and Kendall, 1963 called Behaviorally Anchored Rating Scale (BARS) [18] as shown in Table3. BARS uses intervals to minimize evaluator's impreciseness and maximize reliability of assessment.

**Table 3. Behaviorally Anchored Rating Scale**

Numerical scale	Relative Amount of Activity
1 to 3	Less than usual amount of typical activity
4 to 6	Usual amount of typical activity
7 to 9	More than usual amount of typical activity

Hence, decision making problems must properly acknowledge uncertainty as a central aspect.

## 2.4 Aggregation: General Context

Simple activities can be prioritized pleasantly without the help of any methods or tools, even without being aware of the prioritization process. However, to be truly effective some prioritization approach must be employed for complex evaluations, where several people influence the decisions characterized by several criteria, articulated by imprecise judgments. Although MADM as a discipline has a shorter past record history of about 40 years, more than 70 MADM techniques have been developed for facilitating the decision making process [46]. Different methods are applicable in

different settings based on the important characteristics of information received [1]. Numerous studies have been conducted on multiple person MADM problems with the challenge of arriving at fair decisions, which properly balance competing personal preferences (Hwang and Lin, 1987; Bui, 1987; Chen and Hwang, 1992; Saaty, 1994; Yen and Bui, 1999; Nurmi and Meskanen, 2000).

Several MADM methods which are based on precise and imprecise judgments are: Weighted Sum Model (WSM), Weighted Product Model (WPM), Analytic Hierarchy Process (AHP) and Multiplicative AHP [34], Multiple Attribute Utility Theory [32], Dempster Shafer Theory of evidence for attribute aggregation [31], Evidential Reasoning (ER) and Interval Evidential Reasoning (IER) [28]. Multiple person MADM problems, which use fuzzy set theory for representing imprecision, have gained much prominence (R R Yager 1977; D Dubois 1980; Tanino, 1984, 1990; Chen and Hwang, 1992; Chiclana *et al*, 1998).

## 3. 4A IN THE CONTEXT OF SOFTWARE REQUIREMENTS

This section discusses the basics and importance of prioritization followed by a discussion of the four elements of prioritization problem in the context of software requirements.

One of the key challenges to software engineering community is achieving quality. This challenge is hit through proper RP, which helps to focus the best efforts of developers on the features that matter most for the customer satisfaction [38]. Priorities must be assigned to the several requirements cropped up during elicitation since tight deadlines may preclude the implementation of every software requirement (Roger S Pressman, 1997). Through RP, it is possible to utilize limited resources efficiently by focusing on the requirements that are most critical to the success of the project. Further, proper prioritization and implementation of requirements is important for overall survivability and economic growth of the company. The added advantages of RP can be found in [41]. The significance of RP as stated by Patrik Berander and Anneliese Andrews, 2007:

*The correct requirements and planning suitable releases with the right functionality is a major step towards the success of a project or product. If the wrong requirements are implemented and users resist using the product, it does not matter how solid the product is or how thoroughly it has been tested.*

The significance of RP is understood by highlighting the direct and indirect consequences observed if prioritization is not done [20, 30, 41].

- It may not be possible to ensure the product developed resonates with the core expectations of the stakeholders.
- Available limited resources may be consumed by the requirements not so important.
- It may be difficult to dissect requirements into the ones to be delivered immediately and the ones that can be postponed for later increments.
- Expected functionality may not be guaranteed to the customers leading to the rejection of the product.
- Project success may be questionable.
- It may not be possible to develop better systems and hence better value to stakeholders cannot be guaranteed.

- Trade off among conflicting attributes that characterize requirements' evaluation decisions is not possible.
- Quality of the software product may get unnoticed.
- It may not be possible to find the most cost effective way of developing software.

The list above clearly shows that RP is the force that drives all the subsequent activities of software development and is of paramount importance, irrespective of the type, complexity and size of the project.

### **3.1 Attributes: Software Requirements Context**

RP problem is characterized by multiple attributes like value, cost, schedule, effort, risk etc, which are conflicting in nature. All of them cannot be achieved at the same time because of the restricting constraints on each. Hence, appropriate trade-off decisions have to be made. For example, if requirements are assessed along the dimensions of value and cost, trade-off decisions to be made in order to determine which set of requirements with high value can be provided at the low cost in order to maximize customer satisfaction [19].

Different RP techniques in the literature are developed to work with different attributes. The attributes for prioritization using AHP [19] are along the dimensions of value and cost. Wiegers [20] proposed an approach for prioritizing requirements along the dimensions of risk, penalty and cost. Several criteria like value, cost, risk, difficulty of implementation, likelihood of success etc to be taken into account as argued in [21]. Hermann and Daneva found 240 papers based on benefit and cost estimation [42].

In addition, different concerns of different stakeholders have to be accounted for. Generally speaking, end users concern is the value or benefit where as developers concern is the technical difficulties. Budget is the primary concern of the financial representatives [30]. The participation of an assorted group of stakeholders, comprising developers who provide cost and risk values, customers who provide benefit and penalty values and project manager who mediates conflicts and makes trade-off decisions, is recommended [20].

Even if a single attribute is considered, different stakeholders may interpret in different ways. For example, the different interpretations of the value attribute are: increasing sales, increasing profit, finding new customers, beating competitors etc [37]. Hence, to achieve consistent results, it is very important to properly understand which set of attributes are relevant to the problem under consideration. It is equally important to realize their interpretations as viewed by different stakeholders.

### **3.2 Assessors: Software Requirements Context**

Assessors are evaluators of the requirements' priorities. They are stakeholders carrying some expectations of the upcoming product. All the direct and indirect benefits of RP are highly geared towards stakeholder satisfaction, which is the success criterion for any project (Kotonya, 1997). The prominence of stakeholder involvement in RP is well accepted for the reason that the quality of the software product is largely determined by the proper identification of stakeholders and their needs [22]. Stakeholder identification precedes any other requirements engineering activity with the objective of determining who they are and how important they are [23].

CMMI defines stakeholder as "a group or individual who is affected by or is in some way accountable for the outcome of an undertaking" [35]. Stanford Research Institute defines stakeholders as "those groups without whose support the organization would cease to exist". IEEE-1471 standard defines the stakeholder as "An individual, team or organization with interests, or concerns relative to a system".

They encompass business manager, project manager, marketing representatives, developers, end users, project sponsor or client or customer, architect, tester, quality engineer, product manager, operator and maintainer each with their own perspective of the product. Stakeholder categorization can be done in different ways [24-26] depending on the problem characteristics and the environmental setting. One sort of categorizing is as in [27]: *Critical*: if neglect might kill the project or render the system useless, *Major*: if neglect would have a significant negative impact on the system and *Minor*: if neglect would have marginal impact on the system. Another way is to categorize into key users, secondary users, and unimportant users [6]. Participation of at least three groups of stakeholders: customers, developers and financial representatives is emphasized [29]. If felt appropriate, even more people that claim to have a stake in the product can be involved [30].

Once relevant stakeholders are identified, they have to be differentiated in order to determine the varying influence they exert on the project, by assigning weights [23]. The weighted impact helps to achieve trade off among the competing personal preferences. This differentiation further helps to discriminate the drivers for the prioritization process [33]. One of the guidelines to be kept in mind when assigning weights is that people who actually use the product should have greater impact [43].

On the other side, if only a single stakeholder is involved in RP rather than a group, it may not be possible to satisfy the diverse group of stakeholders associated with the project. Chaos Report 2009 disclosed ten main factors for project failures with one amongst them is lack of user involvement. Another worry with individual estimates is that they are more prone to extreme outliers when compared with team estimates. [45]. Hence, participation of a proper mix of stakeholders as assessors of the requirements' priorities is appreciated.

### **3.3 Ambiguity: Software Requirements Context**

RP is a challenge for software organizations because, it demands a significant amount of decision making [36] where uncertainty plays a central role. Ambiguity shrouded in the forms of uncertainty, incompleteness and vagueness do exist. Assessors have to cope up with different conceptualizations of uncertainty described in Table 1. Ambiguity during RP may arise for several reasons [40, 41]:

- Description of the requirements in natural language.
- Guesses to be made about upcoming product where only partial knowledge is available.
- Decision makers who judge the priorities of requirements are human beings and human judgments may not be absolutely sure.
- The requirements are vaguer earlier in the project and become clearer as understanding of the product grows.
- Assessors may be in a position to provide only partial information or no information at all.

- Assessments about the priorities of the requirements are based on perception of the system, which cannot be precise always.
- The inability of assessors to provide precise judgments or the failure of some assessors to provide judgments in a group decision situation.

If uncertainty is ignored, success of the product may be affected [39]. Hence, RP problem that does not take various ambiguities into account is of minimal use. Ambiguity brought on by lack of knowledge has to be modeled in some form during RP. The dire need for the acknowledgment of uncertainty in RP is complemented by the observations made by several researchers as noted below:

- Grant Ruhe, 2000 says estimating is about predicting in the face of uncertainty and incomplete knowledge. If uncertainty is not taken into account, precise answers exhibit the high probability of being incorrect. The major contribution factor to the well reported high percentage of failed software projects is the built in human trait to prefer precision over accuracy [7].
- Moisiads, 2002 introduces a RP tool, which captures stakeholders' opinions using graphical fuzzy rating scale. The argument is that people tend toward ranges that are flexible to express uncertainty rather than single points [11].
- Paolo Avesani et al, 2004 present the limitations of recent approaches to RP as their inability to address uncertainty and incompleteness. Rating scales used for requirements evaluation based on discrete categories is another limitation [8].
- Ruhe et al, 2005 mention the current challenges of requirements prioritization as incompleteness and uncertainty of information, conflicting requirements, not enough stakeholder involvement, geographically distributed stakeholders, resource bottlenecks etc. Essay on estimation perspectives records software schedule and cost estimation models as optimistic, most likely and pessimistic estimation ranges rather than point estimates because upfront estimates cannot be precise always [6].
- Ruhe et al, 2005 portray RP activity both as an art and a science. As art, it depends on human intuition. As science, it depends on computational algorithms to produce best solutions. A release planning framework for RP which combines the human experience and knowledge with the strength of computational algorithm is launched. Ruhe concluded that hybrid approaches that integrate in this manner have proven most promising [9].
- Rudolf Vetschera, 2006 characterized software development projects as decision problems under risk. Hence, it is recommended to evaluate the decision alternatives that occur in the software process in terms of likelihood [2].
- B Regnell et al, 2007 argue that human judgment is imprecise by nature irrespective of absolute or relative judgment in view of the fact that some requirements can be estimated precisely, some others with reasonable precision and some others cannot be estimated at all [14].
- Jane Cleland- Huang, 2008 has written that decision support mechanism for RP has to be more sophisticated in

order to accommodate the partially ignorant or fully ignorant information provided by some stakeholders [10].

- Hermann A Daneva, 2008 have put forth the idea of comparing two RP techniques empirically once with an approximation and once without. The intention is to find whether people like approximations or the opposite [13].
- Annabella Loconsole et al, 2011 introduced a novel distributed and automated RP technique and the paper concluded with one of the future challenges as acknowledgment of uncertainty during RP [12].

Hence, as stated above many a requirements prioritization techniques and related concepts in the literature address the compelling need for acknowledging ambiguity, uncertainty, incompleteness and imprecision of the priorities of requirements, but few of them provide a workable solution. The necessity for modeling uncertainty in RP is established.

### **3.4 Aggregation: Software Requirements Context**

It is quiet easy task to determine the priorities of requirements if a single stakeholder provides precise priorities over a set of requirements along the dimension of a single attribute. However, it is more challenging to aggregate the imprecise judgments of diverse group of stakeholders over a set of requirements characterized by multiple attributes, and generate reliable requirements' priorities. Applicability of the aggregation methods discussed in section 2.4 has to be studied and adopted after adaptation, taking into account the relative strengths and weaknesses.

## **4. REQUIREMENTS PRIORITIZATION FRAMEWORKS USING 4A**

The 4A have substantial influence on the RP activity as shown in Figure 1. This section specifies how different RP frameworks can be designed by integrating the four elements of prioritization: Attributes ( $A_t$ ), Assessors ( $A_s$ ), Ambiguity ( $A_m$ ), Aggregation ( $A_g$ ). The four frameworks are presented below:

1. Multiple Attribute Multiple Assessor Ambiguity Aggregation ( $MA_tMA_sA_mA_g$ ):

*Preference decisions of ambiguous nature by multiple assessors, with competing personal preferences, made over the software system requirements characterized by multiple conflicting attributes are aggregated with the goal of deriving the most important requirements.*

2. Single Attribute Multiple Assessor Ambiguity Aggregation of Alternatives ( $SA_tMA_sA_mA_gA_l$ )

*Preference decisions of ambiguous nature by multiple assessors, with competing personal preferences, made over the software system requirements characterized by a single predominant criterion are aggregated with the goal of deriving the most important requirements.*

3. Multiple Attribute Single Assessor Ambiguity Aggregation of Alternatives ( $MA_tSA_sA_mA_gA_l$ )

*Preference decisions of ambiguous nature by a single assessor, made over the software system requirements characterized by multiple conflicting criteria are aggregated with the goal of deriving the most important requirements.*

4. Single Attribute Single Assessor Ambiguity Aggregation of Alternatives ( $SA_tSA_sA_mA_gA_l$ )

*Preference decisions of ambiguous nature by a single assessor, made over the software system requirements characterized by a single predominant criterion are evaluated with the goal of deriving the most important requirements.*

The basic frameworks listed above can be adapted based on characteristics of the RP problem under consideration.

## 5. CONCLUSION AND FUTURE PLAN

RP as a decision problem is found to be tied with four primary elements: attributes, assessors, ambiguity and aggregation. If any one of them is neglected or considered insufficiently, it may result in poor customer satisfaction. Four RP frameworks are launched, all having the 4A in common, but differing in the way they occur. Meaningful frameworks can be developed by observing different characteristics of the problem and the environment. The topics of interest with respect to attributes are: single versus multiple, flat versus hierarchical structure, dependent versus independent attributes and relative versus absolute weightings. Efficient combination of attributes appropriate to the problem at hand can also be explored. The topics of interest with respect to assessors are: number, type and grouping of stakeholders, geographically colocated versus distributed stakeholders, differing weights to stakeholders that aid in sensitivity analysis. The topics of interest with respect to modeling ambiguity are: representation using intervals or probability distributions or fuzzy data or a combination of them, the ease and attractiveness with which ambiguity can be modeled. The topics of interest with respect to aggregation mechanisms are: linear versus nonlinear aggregation, rich outputs for data analysis, rank reversals with addition of new requirements, scalability, find the best alternative or an optimal set of alternatives.

Studies and experiments have to be made how the requirements prioritization frameworks can be adapted to the various problems. Results and observations to be supported by the empirical results accompanied with theoretical background. Frameworks must integrate 4A in a consistent manner with the aim of generating reliable requirement's priorities. Guidelines, constraints, tools, methods and models have to be developed indicating the potential usage of frameworks. These may help to break down the complex RP problem into small manageable tasks and integrate the results back to arrive at optimal decisions. The RP frameworks outlined can be further evolved, by investigating current approaches and their applicability in different settings with the aim of narrowing down the gap between theoretical research and industrial needs. Nevertheless, these can be improved to align with the 4A or new ones can be suggested. A wide scope of research is unwrapped for further applicability of these frameworks in other relevant areas of software engineering.

## 6. REFERENCES

- [1] Hwang C. L and Yoon K. Multiple Attribute Decision Making Methods and Applications: A State of the Art Survey. Springer-Verlag, Berlin, Heidelberg, New York, 1981.
- [2] Rudolf Vetschera. Preference-Based Decision Support in Software Engineering. Value-Based Software Engineering 2006 Springer. pp 67-89. doi: 10.1007/3-540-29263-2\_4.
- [3] MacCrimmon K R. An Overview of Multiple Objective Decision Making. Published In Multiple Criteria Decision Making (pp. 18-43). Columbia, SC: University of South Carolina Press, 1973.
- [4] Fred S Azar, 2000. Multi attribute Decision-Making: Use of Three Scoring Methods to Compare the Performance of Imaging Techniques for Breast Cancer Detection. University of Pennsylvania. Technical Report No. MS-CIS-00-10
- [5] B.W. Boehm and R.E. Fairley. Software Estimation Perspectives. IEEE Software, Nov/Dec pp 22-26.
- [6] Ruhe, G and Saliu, M. O. The Science and Practice of Software Release Planning. IEEE Software (2005).
- [7] G. Ruhe. Bees and the Art of Estimating. IEEE Software, Vol. 17, No. 6, page 2, November 2000.
- [8] Paolo Avesani, Anna Perini, and Angelo Susi. Supporting the Requirements Prioritization Process: A Machine Learning Approach. In Proc. of 16th International Conference on Software Engineering and Knowledge engineering (SEKE 2004), June 2004, Banff, Alberta.
- [9] G. Ruhe and M. O. Saliu. The art and science of software release planning. IEEE Software, vol. 22, pp. 47–53, November 2005.
- [10] Jane Cleland-Huang and Bamshad Mobasher. Using Data Mining and Recommender Systems to Scale up the Requirements Process. ULSSIS 2008, Leipzig, Germany.
- [11] Frank Moisiadis. The Fundamentals of Prioritizing Requirements. In Proc of Systems Engineering, Test and Evaluation Conference, Sydney, Australia, October 2002.
- [12] Annabella Loconsole, Hannes Gruber, et al. Construction and Evaluation of an Algorithmic and Distributed Prioritization Method. Second Workshop on Requirements Prioritization for Customer-Oriented Software-Development RePriCo'11.
- [13] Hermann and A. Daneva. Requirements Prioritization Based on Benefit and Cost Prediction: An agenda for Future Research. In Proc of the International Conference on Requirements Engineering (RE 2008), pp 125-134.
- [14] B. Regnell, M. Host, J. Nattoch Dag, P. Beremark and T. Hjelm. An industrial case study on distributed prioritization in market driven requirements engineering for packaged software requirements. Requirements Engineering, Vol 6, 2001, pp 51-62, doi:10.2007/s007660170015.
- [15] Biren Das. Examination Reforms: Marking vs. Grading. Published in University News, 45(13), March 26-April 01, 2007.
- [16] [www.ugc.ac.in/new\\_initiatives/academic.pdf](http://www.ugc.ac.in/new_initiatives/academic.pdf).
- [17] Barbara Gross David. Grading Practices. Jossey Bass Publishers, San Francisco, 1993. [teaching.berkeley.edu/bgd/grading.html](http://teaching.berkeley.edu/bgd/grading.html).
- [18] Scott Parrill. Revisiting Rating Format Research : Computer based Rating Formats and Components of Accuracy. Ph.D Thesis Virginia Polytechnic and State University.
- [19] J Karlsson and K Ryan. A Cost Value Approach for Prioritizing Requirements. IEEE Software, 14(5) pp 67-74.
- [20] K. Wiegers. First Things First: Prioritizing Requirements. Software Development. Vol. 7 no. 9, Sep 1999.
- [21] <http://www.theiba.org/AM/Template.cfm?Section=Body> of Knowledge BABOK Guide 2.0.

- [22] A Aurum and C Wohlin. Aligning Requirements with Business Objectives: A Framework for Requirements Engineering Decisions. Proceedings Requirements Engineering Decisions Support Workshop, Paris, France 2005.
- [23] Glinz, Martin. Stakeholders in Requirements Engineering .IEEE Software . 28(1). Pp 18-20 ISSN 0740-7459.
- [24] Freeman, R. Edward et al. Stockholders and Stakeholders: A New Perspective on Corporate Governance. California Management Review, Spring 83, Vol 25, Issue 3, pp 88-106.
- [25] Ryan A. Mc Gee . Stakeholder Identification and Quality Attribute Prioritisation for a Global Vehicle Control System. Proc of the Fourth European Conference on Software Architecture. ISBN/ISSN: 978-1-4503-0179-4.
- [26] Mendelow A. (1991). Stakeholder Mapping. Proceedings of the Second International Conference on Information Systems, Cambridge, MA.
- [27] Daniela Damian. Stakeholders in Global Requirements Engineering: Lessons Learned from Practice. IEEE Software. Vol 24. No.2 March 2007. pp 21-27. doi :10.1109/MS.2007.55.
- [28] Dong-Ling Xu, Jian-Bo Yang, Ying-Ming Wang, "The evidential reasoning approach for multi-attribute decision analysis under interval uncertainty," European Journal of Operational Research 174(2006) 1914- 1943. Elsevier. doi: 10.1016/j.ejor.2005.02.064.
- [29] Davis .A.M.: The Art of Requirements Triage, IEEE Computer ,36(3), pp 42-49.
- [30] Patrik Berander. Evolving Prioritization For Software Product Management, Blekinge Institute of Technology Doctoral Dissertation Series No 2007:07 ISSN 1653-2090 ISBN 978-91-7295-108-2.
- [31] Glenn Shafer. A Mathematical Theory of Evidence. Princeton University Press, Princeton, NJ, 1976.
- [32] L Keeney and H Raiffa. Decisions with Multiple Objectives –Preferences and Value Trade-Offs. 2<sup>nd</sup> ed Cambridge UK: Cambridge Univ Press, 1993.
- [33] Value Creation by Agile Projects :Methodology or Mystery ? Zornitza Racheva, Maya Daneva and Klaas Sikkel.
- [34] Evangelos Triantaphyllou and Khalid Baig The Impact of Aggregating Benefit and Cost criteria in Four MCDA methods. IEEE Transactions of Engineering Management Vol: 52, Issue: 2 p 213-226.
- [35] <http://www.sei.cmu.edu/cmml/start/faq/models-faq.cfm>
- [36] Björn Regnell, Barbara Paech et al. Requirements mean decisions! –Research issues for understanding and supporting decision making in requirements engineering", Proc. 1st Swedish Conference on Software Engineering Research and Practice (SERP'01).
- [37] Björn Regnell , Martin Host. et al. An Industrial Case Study on Distributed Prioritisation in Market Driven Requirements Engineering for Packaged Software. Requirements Eng(2001) Vol 6, Issue 1, pp 51-62 Springer Verlag ISSN:09473602
- [38] Karlsson J., Wohlin C., Regnell B. (1997). An evaluation of methods for prioritizing software requirements. Published in Elsevier, New York, 7 Feb. 1997, pp 939-947.
- [39] Lena Karlsson, Björn Regnell, Joachim Karlsson and Stefan Olsson. Post-Release Analysis Of Requirements Selection Quality: An Industrial Case Study. In Proceedings of the 9th International Workshop on Requirements Engineering: Foundation for Software Quality (REFSQ'03), Velden, Austria, June 2003, pp. 47-56.
- [40] S.D. Sandri, D. Dubois, and H. Kalfsbeek. Elicitation, Assessment and Pooling of Expert. Technical Report IRIT/93-24-R, IRIT, Univ. P. Sabatier, Toulouse, France, 1993.
- [41] Donald .G. Fire Smith Prioritising Requirements. *Journal of Object Technology* . Volume 3, Number 8, Sep-Oct 2004.
- [42] R B Svensson et al. Prioritization of quality requirements : state of practice in eleven companies. In proceedings of the 2011 IEEE International Requirements Engineering Conference. ISBN: 978-1-4577-0921-0.
- [43] B Regnell, Martin Host et al. Visualization of Agreement and Satisfaction in Distributed Prioritization of Market Requirements. In proc of 6th Int. Workshop on Requirements Engineering: Foundation for Software Quality, June 5-6 2000, Stockholm, Sweden.
- [44] Raanan Lipshitz and Orna Strauss. Coping with Uncertainty: A Naturalistic Decision-Making Analysis. Organizational Behaviour and Human Decision Processes Vol 69. No 2 Feb pp. 149-163.
- [45] Stefan Biffl. Using Inspection Data for Defect Estimation. IEEE Software, v.17 n.6, p.36-43, November 2000 doi>10.1109/52.895166
- [46] Li Y. "An Intelligent Knowledge-based Multiple Criteria Decision Making Advisor for Systems Design," Ph.D. Dissertation, School of Aerospace Engineering, Georgia Institute of Technology, 2007.