Available online at www.sciencedirect.com

SciVerse ScienceDirect

Procedia Computer Science 16 (2013) 806 – 813

Procedia
Computer Science

Conference on Systems Engineering Research (CSER'13)
Eds.: C.J.J. Paredis, C. Bishop, D. Bodner, Georgia Institute of Technology, Atlanta, GA, March 19-22, 2013.

# Value-Based Requirements Prioritization: Usage Experiences

Nupul Kukreja*[a], Sheetal Swaroop Payyavula[b], Barry Boehm[a], Srivinas Padmanabhuni[b]

[a]*Center for Systems and Software Engineering (CSSE),University of Southern California, Los Angeles 90089, USA*
[b]*Infosys Labs, Infosys Technologies, Bangalore, India.*

**Abstract**

There are usually more requirements than feasible given budget and schedule constraints. Thus it's important to select the most valuable ones for implementation in order to ensure the delivery of a high value system. Simple prioritization approaches like 1-10 ranking or MoSCoW lead to numerous ties requiring one to repeat the process for the tied items. In a previous study [1] we analyzed 17 different prioritization frameworks that could be used to perform value-based requirements prioritization (VBRP). The *Technique of Ordered Preference by Similarity to Ideal Solution* (TOPSIS) was selected as the framework of choice, as a result of the analysis. TOPSIS was deployed for use by a premier IT company in India. In this paper we present our experiences in using a decision analysis framework like TOPSIS to perform VBRP. We have seen successful applications of using such a decision analysis framework for prioritizing test-cases, scoping to select the most valuable requirements for a release or product roadmap, value focused resource allocation and value-oriented product customization. We aim to provide evidence showing how VBRP and decision analysis frameworks can help channelize various systems engineering activities by focusing on the most valuable items first. Our experience in deploying the framework for various prioritization activities provides strong evidence making explicit the need for VBRP as a first-class citizen for value-based planning, implementation and delivery of systems and software applications.

*Keywords*: requirements; prioritization; decision making; value-based software engineering; usage; experience

## 1. Introduction

For systems with a large number of early high-level requirements (300-500 on average) simple prioritization methods like 1-10 or MoSCoW (must, could, should and want-to haves) are unable to capture the true value of the requirements. There are numerous ties and the process is repeated to re-rank the tied requirements; possibly more than once. These techniques assume that the stakeholders understand the intrinsic value of requirements and can correctly score them on a 1-10 scale or add them to the appropriate priority bucket in case of MoSCoW. We undertook a case study [1] whereby we analyzed seventeen different prioritization frameworks against a set of seventeen different criteria. These criteria were elicited by conducting interviews with 50 stakeholders i.e. business analysts, project, program and portfolio managers. Each prioritization framework was analyzed and scored against

* Corresponding author. Tel.: +1-213-740-6505;.
*E-mail address*: nkukreja@usc.edu

1877-0509 © 2013 The Authors. Published by Elsevier B.V.
Selection and/or peer-review under responsibility of Georgia Institute of Technology
doi:10.1016/j.procs.2013.01.084

each of the 17 criteria. The relative importance of each criterion was ascertained using a Kano Survey [2] conducted with the stakeholders. The analysis was performed by tailoring the House of Quality framework [3] for comparative analysis. Based on the analysis TOPSIS [4] was selected as the framework of choice. We created a tool based on TOPSIS and executed a test pilot to ascertain its value and use for prioritization. Based on the feedback the tool was modified to handle prerequisites for requirement dependencies and the ability to perform hierarchical prioritization. The tool was initially developed as an Excel™ Workbook and has now been integrated into a web-based Distributed Project Management Platform for company-wide deployment.

### *1.1. Research hypotheses*

We originally sought to validate the following research hypotheses, if a VBRP framework was made available to various teams within the company:

- To help clients optimize on scope and help the company manage resources and focus on the highest value items first
- To select highest value requirements to pursue in case of fixed bid projects
- To aid project managers for scoping interim releases – working on the most valuable aspects first
- Ordering the testing activities around most valuable requirements owing to high turnaround time for regression testing

The tool was used on numerous projects with varying levels of success providing indicators of satisfying each of the above hypotheses.

### *1.2. Usage Overview*

Business analysts used VBRP to prioritize the requirements and the testing team prioritized test-cases with respect to the corresponding requirements. The test-case priorities were influenced by that of the corresponding requirement providing a better metric of priority. VBRP was also used for resource planning and allocation to channelize the effort on the most valuable requirements in case of a fixed-bid and fixed-schedule projects. The planning, design and implementation activities were based around the most valuable items.

An independent product team in the organization adopted VBRP to aid in the decision making process of selecting features for the upcoming roadmap of their product. Typical roadmap/release planning meetings took days to complete whereas using a decision analysis model like TOPSIS they were able to have a better idea of the priorities in significantly less time and could better channelize their meetings towards the most valuable items. The same product team also used TOPSIS to better understand which high-value requirements of a prospective client were already satisfied and which would need to be customized in their product. This enabled them to prepare a value oriented business case showcasing the overall value of their product to the client.

The following sections provide a quick overview of TOPSIS and its inner working along with the detailed analysis of its application across the organization. The analysis provides strong evidence for using VBRP for planning and scheduling various systems and software engineering activities around the most valuable requirements.

## 2. TOPSIS Primer

TOPSIS is a method for performing multiple-criteria decision analysis (MCDA). It is based on the vector space model of computation. We provide a quick overview of its underlying algorithm in this section.

In TOPSIS a set of alternatives are scored against a set of criteria. The alternatives maybe scored on an absolute scale (e.g. dollars, effort-hours etc.,) or a relative scale (e.g. 1-9 Likert scale, 1-3-9 or Fibonacci scale) for each criterion. The criteria have a direction of preference i.e. if more or less of that particular criterion is preferred for prioritization. For example, let's assume *Effort* is a criterion for comparing various alternatives. If we wish to have the alternatives ranked by 'low effort' then *Effort* would be depicted with a '–' symbol. If however, we wish to rank the alternatives by 'high effort' then we would depict it with a '+' symbol.

Each alternative is considered to be a multi-dimensional vector in *criteria space*. The theoretical *ideal* alternative *S'* is the one having the best score for each criterion. Consequently, the theoretical *non-ideal* alternative *S** is the one

| Click to Generate Next Level of Prioritization<br>Delete Current Level<br>**Root** | UPDATE<br>**Prerequisites:** Comma separated list of requirement IDs/Serial Number(s) | **Serial Number (Autogenerated)** | **Overall Weighted Ranking (Non Dependency Adjusted)** | **Project Goals/Expectations/Criteria** | **Criterion 1** | **Criterion 2** | **Criterion 3** | **Criterion 4** | **Criterion 5** |
|---|---|---|---|---|---|---|---|---|---|
| **Categorization (Available only if not at root level)** | **Check box to perform dependecy adjusted ranking**<br>☐ | **Select Id Prefix (below)** | **TOPSIS** | **(Relative) Goal Weights** | **1** | **3** | **3** | **6** | **4** |
| | | | | **Direction of Improvement (+/-)** | **+** | **-** | **+** | **+** | **+** |
| | | **R** | **0-1 Normalized Score** | **List of Requirements/Test-Cases/Alternatives** | **For each requirement fill in it's relative re**<br>**Suggested scale: 1-9 (1 = low satisfactior**<br>***(You MAY use numbers greater than 9 if y*** | | | | |
| | | R1 | 0.3051 | Requirement 1 | 1 | 2 | 3 | 8 | 2 |
| | R3, R4 | R2 | 0.1786 | Requirement 2 | 6 | 4 | 9 | 2 | 5 |
| | | R3 | 0.1597 | Requirement 3 | 3 | 5 | 7 | 1 | 7 |
| | | R4 | 0.3566 | Requirement 4 | 8 | 6 | 5 | 7 | 9 |

Figure 1. TOPSIS Excel Worksheet

with the worst score for each criterion. Depending on the direction of preference of the criterion, the best/worst score is the maximum/minimum score in the set of alternatives, for that criterion.

The algorithm rank orders the alternatives so as to minimize their vector distance from the *ideal (S')* and maximize it from the *non-ideal (S*)*. Hence the name, *Technique of Ordered Preference by Similarity to Ideal Solution (TOPSIS).*

## *2.1. TOPSIS Excel™ Template*

The initial version of TOPSIS was developed as an Excel™ Workbook. Figure 1 shows a snapshot of the template. The criteria to be used for comparing the requirements are captured in the topmost row. Each criterion has a relative score and direction of preference indicated by a '+/–' symbol. Each requirement is entered under the column for 'List of Requirements' and is scored on the appropriate scale against each criterion. The figure only shows numbers entered from 1 through 9 but the scores could be absolute measures if they need to be. Each requirement is given an auto-generated ID which is used for specifying any dependencies/prerequisites that may exist. In case of hierarchical prioritization there is also provision for specifying which parent requirement a specific child requirement belongs to and can be specified in the leftmost column of Fig 1. A quick overview of the key features of prerequisite handling and hierarchical prioritization are provided in the following sub-sections.

## *2.2. Prerequisite/Dependency Handling*

Most prioritization methods suggest leaving prerequisites out when prioritizing requirements owing to the additional complexity of the prioritization. However, handling prerequisites was deemed very critical by our stakeholders. It was required to better echo the intended delivery/implementation sequence of the requirements. The absence of prerequisite handling made the tool unusable in practice. Further deliberation with our stakeholders suggested that the final score of a requirement must be less than each of those in its prerequisite graph (i.e. a high-value requirement could not be implemented unless its low-value prerequisites had been satisfied). This required making a pass over the resulting scores, as computed by TOPSIS and changing the score of each requirement to be less than each of its prerequisites/dependencies. The prerequisites are specified using a comma separated list of requirement IDs under the second column from the left as seen in Fig. 1. This feature can be turned on or off to see how the priorities change in case of any dependencies.

### 2.3. Hierarchical Prioritization

Another requirement strongly advocated by the stakeholders, especially by the independent product team, was the ability to perform hierarchical prioritization. That is, being able to prioritize high-level requirements with respect to goals and low-level requirements w.r.t. high-level ones and so on. That is, being able to prioritize requirements at a child level and have their priorities influenced by their parent. TOPSIS doesn't have a capability for performing hierarchical prioritization and was extended to cater for such a scenario as explained in [1] using mathematical normalization. Each child level is independently prioritized using the same TOPSIS template as above and each child requirement can be linked to its parent by selecting it from a drop-down list that appears in the first column of Fig 1 (when at a child level worksheet). The sum of the scores of the children adds up to that of the parent. This allowed one to perform multiple levels of prioritization. Using it beyond three levels proved to be cumbersome and was not encouraged unless required.

## 3. Usage Experiences of VBRP

In this section we detail the application of using a VBRP/decision-analysis framework based on TOPSIS for performing value-based prioritization of the requirements and test-cases and using the results for scoping and resource allocation/planning as well as for creating a business case. These case studies aim to provide strong qualitative evidence of the value of having VBRP practices across the organization.

### 3.1. Value-Based Test Case Prioritization

An independent testing team used TOPSIS to perform value-based test-case prioritization. Owing to high turnaround times because of regression testing, they wanted to know which test-cases were of the highest value so as to shorten the testing cycle time by focusing on the most valuable requirements. This required a great deal of communication overhead with the business analysts to understand the value of each requirement and prioritizing the test-cases accordingly. Changes in the system would require running the complete regression test-suite; however, schedule constraints would necessitate the need for testing the most critical requirements first to decrease turnaround time. To solicit this information the testing team would constantly communicate with the business analysts to ascertain the criticality of each new and changed requirement and reprioritize their test cases.

After deploying TOPSIS the business analysts prioritized the requirements, which were at the same level of granularity, against the goals of the project. The same workbook was then used by the testing team to generate test-case priorities. The testing team generated a 'child level' worksheet for prioritizing their test-cases against a set of criteria more akin to testing (e.g. criticality of failure, effort-hours, business value etc.) along with the project goals. Each test-case was then mapped to the corresponding requirement using the hierarchical capability mentioned above. Thus, the test-case priority was influenced by that of the requirement it belonged to. If a change was made to a feature or a new feature was added or the project goals' priorities changed, it became relatively trivial for the testing team to ascertain the priority of the test-cases without the need for manually reprioritizing everything thus eliminating high communication and manual overhead. The overall test-case priorities were independently validated by the business analysts and were largely seen in to be in accordance with their expectations (except in 10-20% of the cases). The margin of error was acceptable and TOPSIS-based hierarchical prioritization of test-cases was adopted for prioritizing test-cases with respect to the original (prioritized) requirements. The testing team was now able to better plan the testing cycles around the most valuable requirements and perform incremental value-based testing, substantially decreasing turnaround time. Thus, VBRP was also deployed for value-based testing with increased time savings and better planning of the testing activities with high 'return on value'.

### 3.2. Value-Based Roadmap Planning

The hierarchical prioritization capability was suggested by the independent product development team at the company as they used the same mental model for planning the product roadmap and interim releases. Typical product and release planning meetings lasted for days with a high communication overhead. The team decided to test TOPSIS and gauge its viability for ascertaining the priorities of the requirements at various levels of granularity by correlating the output of TOPSIS with the requirements chosen for the current release of the product.

Initially, the team prioritized high-level requirements against the strategic goals of the company along with the needs of their clients. Prerequisites/dependencies were common at this level and were captured accordingly in the tool as shown above. They then created a second level of low-level requirements and mapped it to the corresponding high-level ones. (Another (third) level of requirements was created and mapped to their ‘parents’ at the second level.) Creating the three-level hierarchical model and ascertaining the priorities took significantly less time (a few hours) and provided valuable insights about the overall value/priority of each requirement. The team was able to perform various what-if scenarios (i.e. sensitivity analysis) on the output to better understand the implications of the strategic goals on the requirement priorities.

The output of the tool had high correlation with the current requirements selected for the product roadmap. About 10-20% of the requirements were out of order owing to the mathematical calculation of the priorities and prerequisite post-processing. Nevertheless, TOPSIS was adopted as the decision analysis framework of choice for product and release planning meetings. The resultant priorities were used to channelize the discussions and also conduct various what-if analyses to better ascertain what features to include or exclude for a particular release. The team experienced immense time savings eliminating the high communication overhead involved in ascertaining the priorities as a result of extended discussions. This also provided them with institutional memory and rationale of selecting one requirement over another for a particular roadmap to better argue their case of including a new feature or excluding a prior requested one.

### *3.3. Value-Based Resource Allocation and Planning*

On one of the projects the company was purely an implementation partner i.e. the requirements were captured by another organization and the implementation was to be done by the company. The necessary effort and schedule estimation was completed and the product was to be shipped by a particular date. Schedule and scope had little flexibility of change. VBRP proved useful for planning and resource allocation of activities. The goals/criteria in this case were primarily of the company i.e. implementation and project planning specific goals e.g.: effort, risk, complexity of implementation, familiarity with technology etc. Since all the requirements in the particular scope were to be delivered, the partner organization viewed them as equally important. The project managers however, needed to plan their implementation activities in a value-centric manner. They decided to adopt TOPSIS and prioritize the various requirements against project specific goals as mentioned above. Planning discussions ensued after calculating the resultant priorities of the requirements along with various what-if analyses to gauge the impact of specific criteria on the related activities. The prerequisite handling provided a convenient way to view the delivery sequence owing to the sensitivity analyses conducted by the team. In most cases, the prioritization agreed with the intuition of the project managers. The prioritization output helped them question their assumptions about the various goals as well as the relative contribution of the requirements to those goals. The output of the prioritization thus aided them in the decision making process for sequencing various software development activities.

The project managers were able to use VBRP to help schedule the implementation activities of the project as well as discuss various possible scenarios by changing the weights/importance of the various criteria. Time savings was immediately apparent along with the ability to focus the planning meetings around the requirement priorities with respect to the value propositions of the project stakeholders.

### *3.4. Value-Based Business Case Analysis and Product Customization*

The independent product team mentioned earlier also used VBRP to prepare a business case of their product for a prospective client. The client had sent out an RFP (Request for Proposal) to the company with a large set of requirements for a transformation project.

A Systems Integration team then performed VBRP on the set of client requirements to rank order the requirements according to the client’s goals. The team had conducted a detailed benefits analysis of typical transformation programs in their domain as result of various responses to RFPs. Twenty-two end benefits were identified for use across the entire portfolio. For each prospective client either all or some of the goals were applicable with varying degrees of importance. These goals were created and finalized by subject matter experts along with the past data of their transformation programs for various clients. The client's requirements were then ranked against the set of applicable goals. Once VBRP was performed on the requirements, the results could be used by the product team to match the requirements as per their value to:

- Already existing "out or the box" features
- Product features that would need customization
- Enhancements needed in the product

After this exercise the team could make an effective business case of their product by showcasing to the client that the product already satisfied most of the high value requirements**.**

## 4. Results and Evaluation

For each of the above case studies where TOPSIS was used for practicing VBRP we solicited the feedback of the teams on its value and applicability. Teams unanimously agreed on the tool being extremely valuable for decision making at various levels of granularity within the project. The primary benefits experienced by each team were:

- High correlation of priorities with intuition – in all cases highlighted above the teams were in agreement about the accuracy of the prioritization with that of their intuition as per their experience. They did disagree with some of the priorities (10-20% in most cases). However, the margin of error was tolerable and VBRP was seen as a valuable addition to their planning and negotiation practices.
- Increased time savings to ascertain requirement/test-case priorities – having a decision analysis framework like TOPSIS for VBRP, decreased communication overhead to ascertain the priorities of individual requirements and test-cases. It was especially faster to perform sensitivity analysis and check the resulting change in a matter of minutes instead of long discussions to ascertain the same. There was now increased ability to check for various what-if scenarios in a relatively short span of time.
- Better channelizing of negotiation and discussion sessions – the teams were better able to focus the discussion around the valuable items as put forth by the tool and further deliberate on the correctness of the priorities. In case of change requests it was easier to see how valuable the changes were in comparison to the existing requirements. This provided the teams with strong evidence for ensuing negotiations in deciding whether to include or exclude the particular change from a specific release.
- Better resource allocation and planning around high-value items – previously the ability of being able to do this relied on the skills and experience of the project managers and business analysts. Having a decision analysis framework augment their decision making process helped decrease the cognitive workload required to manage the various priorities, trade-offs and scheduling of activities around high value requirements. The capability of handling prerequisites and having the prioritization output echo a prospective delivery sequence proved immensely valuable for these tasks.
- Better downstream percolation of priorities – the prioritization activities were usually restricted to requirements or test-cases. There was a high communication overhead in deriving test-case priorities from the requirements. The capability of performing hierarchical prioritization within the decision analysis framework reduced the high communication overhead between the testing team and business analysts; providing independent yet influenced priority ratings of test-cases that could be easily derived from the requirements.
- Ability to provide a metric of *percent value delivered* – since each item score was normalized on a scale of 0 to 1 it provided an approximation of percent contribution of each requirement to the overall project. The teams were asked to exercise caution when citing the same since the implication could be easily misinterpreted. In case of product scoping it proved to be a useful measure to compute the *percent value delivered* i.e. how much percent value was delivered by the selected requirements. (For example, one could state *"These 20 requirements contribute to 50% of the overall value."*)

The above feedback helped validate our research hypotheses that were initially laid out. It provides a strong indicator of the applicability of decision analysis frameworks for carrying out value-centric requirements prioritization activities. A VBRP framework based on decision analysis models seems to be a vehicle well suited for channelizing various systems and software engineering activities around the most valuable items. Although various albeit complex frameworks like Multi-Attribute Utility Theory [5], Analytical Hierarchy Process (AHP) [6] exist and are in widespread use, even less complicated frameworks like TOPSIS or Simple Additive Weighting [7] (or the others analyzed by us in the related study [1]) seem to be equally applicable in providing valuable results and aiding the decision making process.

### *4.1. Limitations*

Although we received positive feedback there were a few limitations that were encountered. However, most of these limitations were from the standpoint of the tool rather than the framework itself. We are working on the limitations and trying to alleviate the same by incorporating them into a web-based system owing to the limitations of implementing them in Excel. The following limitations were reported by the teams:

- Ability to hierarchically organize the goals themselves – currently the hierarchical prioritization is only applicable to the requirements. A few teams suggested the ability of being able to have a hierarchy of goals in a manner similar to AHP [6]. The primary reason cited was to have a better mental model of goal hierarchies and select the levels of goals against which to view the prioritization outputs. Different stakeholders would like to consume information at various levels of abstraction. Currently the model was too low level with respect to the goals and required extensive manual labor to abstract away the information into high-level goals
- Better report generation capabilities – currently the reports need to be manually generated through raw Excel data itself. The teams suggested that if a reporting feature existed to pick and choose the types of reports it would lead to further time savings and help provide information in an easy to assimilate manner.
- Better prerequisite handling – currently the prerequisites/dependencies are handled via comma separated lists of requirements. In case of a large number of items a great deal of back-n-forth is required within the sheet itself to help fill in the prerequisites. The teams suggested the ability of being able to drag-n-drop the dependencies instead of comma separated lists.

There is however a major limitation of TOPSIS itself – rank reversals i.e. introducing irrelevant requirements/test-cases can cause the priorities of the existing items to change and may lead to a different prioritization ordering altogether. This is also a known limitation with AHP and some MCDA methods [8]. However, this was not a deterrent in the above case-studies. The teams were reasonably result oriented and were not intent on gaming the system. Irrelevant requirements or test-cases were filtered before performing the prioritization. The teams would primarily use VBRP/TOPSIS as an aid to the decision making process and also exercised their judgment about the expected prioritization order. For majority of the cases, the teams agreed with the prioritization output as laid out by TOPSIS. For the out of order requirements/test-cases, the teams would manually adjust for the discrepancies in the ratings and use them for planning accordingly.

Most of the limitations stemmed from the usability aspects of the tool rather than the framework per se and rank reversals did not jeopardize the overall value of using TOPSIS. We are currently working to alleviate most of the above concerns in a web-based version of the tool that is integrated with a Distributed Project Management Platform. Irrespective of the limitations, the current version of the tool has provided valuable insights in using decision analysis frameworks for various system and software planning and implementation activities. The validation of the research hypotheses coupled with the feedback of the various teams has provided a stronger impetus to achieve company wide deployment of VBRP practices.

## 5. Conclusion

Our initial exploration of the most appropriate VBRP framework to use [1] led us to select TOPSIS as the framework of choice. After executing initial test pilots we decided to expand the outreach owing to positive feedback from the stakeholders. In this paper we have tried to convince the reader that having a decision making framework for VBRP is indeed valuable and can lead to significant time savings and efficient channelizing of resources towards the most valuable requirements.

For collaborative requirements prioritization, TOPSIS's rank reversal concern is minimal, but could be significant for negotiations among competitors. Although TOPSIS may suffer from rank reversals, it still provides valuable insights into the overall prioritization output leading to constructive discussions around the importance of individual requirements and test-cases. It also provides a model to help compare newly found requirements (or change requests) with the existing ones to better judge their overall value.

The use of a VBRP framework was not restricted to requirements prioritization but has also been successfully applied for prioritizing test-cases and resource allocation. It has proven to be a valuable toolkit for the requirements engineers empowering them to better channelize the negotiation and execution efforts of the various system and software development activities.

## References


1. N. Kukreja, S. S. Payyavula, B. Boehm, S. Padmanabhuni, Selecting an Appropriate Framework for Value-Based Requirements Prioritization: A Case Study, IEEE RE Conf. Sept. 2012
2. M. Cohn, “Agile estimation and planning”, 1st ed. Prentice Hall, 2005. Print.
3. J. R. Hauser and D. Clausing, "The House of Quality," Harvard Business Review, 66, 3, May-June 1988, pp. 63-73
4. R. Wang, “Performance evaluation method - Technique for order preference by similarity to ideal solution (TOPSIS)”, researcher.nsc.gov.tw/public/caroljoe/Data/02182133671.ppt.
5. R. L. Keeney and H. Raiffa, “Decisions with Multiple Objectives: Preferences and Value Tradeoffs”, New York, New York: Cambridge University Press, 1993
6. T.L. Saaty, “The Analytic Hierarchy Process”, McGraw-Hill, Inc. (1980)
7. R. S. Martin, “Engineering Project Appraisal: The evaluation of alternative development schemes”, Malden, MA: Blackwell Science, 2001. Print
8. Y.M. Wang, Y. Luo, “On rank reversal in decision analysis”, Mathematical and Computer Modelling, Volume 49, Issues 5–6, March 2009, Pages 1221-1229, ISSN 0895-7177, 10.1016/j.mcm.2008.06.019. (http://www.sciencedirect.com/science/article/pii/S0895717708002860)