An Experiment on Different Requirements Prioritization Methods

Lena Karlsson, SERG,
Department of Communication Systems, Lund University

Market-driven organizations need to structure, store and prioritize their requirements since they often implement the product in successive releases. Through interviews we have seen that the storage can be implemented as a database or a simple Excel sheet open for new requirements to be entered [1]. However, the prioritization is still difficult to manage in practice. In particular, immature organizations, which do not have a structured process in place, find this difficult and there are not enough resources to buy or develop a tool for it. These new and small organizations need a fast and easy way to prioritize their requirements.

Frequently the decision of what requirements to select for the next release lies at the product manager, who might not have sufficient information at hand. Furthermore, there is a need to inform the product project on *what* requirements to prioritize and most importantly *why*, a choice that can be difficult to motivate without a prioritization method.

There are several prioritization methods based on more or less structured types of sorting algorithms, for instance bubblesort and insertion sort, but the most efficient might be the Analytical Hierarchy Process [2]. Another method of current interest is the Planning game used in Extreme Programming (XP), since the agile methods are getting a lot of attention in many software organizations. Other less complex methods include arranging the requirements into a few groups based on a quick estimation and then only prioritize the group in the middle or even quicker, only picking the group considered most urgent.

Analytical Hierarchy Process (AHP)

The requirements prioritization in AHP involves comparing all possible pairs of requirements and for each pair determine which of the two requirements that is of higher priority, and to what extent. The priority can be based on different criteria, but the most common ones are *value for the customers* and *estimated development effort*.

The AHP is based on a ratio scale and thus can be used for cost-value calculations; furthermore it includes a consistency check and is fault-tolerant [3]. Among its drawbacks is that it requires n(n-1)/2 comparisons, thus it is very time-consuming when scaling-up to larger projects with a larger amount of requirements. There are sophisticated tools that simplify the procedure. However, small or immature organizations might not afford or even need this, although some strive for it.

Planning game sorting

The requirements prioritization in an XP organization is carried out in the Planning game, where the scope of the next release is quickly determined by combining business priorities and technical estimates [4]. In practice, marketing people or customers sort the requirements (written on story cards) *by value* into three piles (1) necessary, (2) significant business value providers and (3) nice-to-have. Then developers sort the requirements *by risk* into three piles: those that they can (1) estimate precisely (2) estimate reasonably well and (3) not estimate at all. Marketing chooses a set of cards, either by setting a release date and choosing cards based on their estimates, or by choosing the cards and calculating the release date.

An addition to the Planning game is to sort the cards within the piles. The method will increase the usefulness and value of the prioritization, using a reasonable amount of time and getting a list of sorted requirements.

Execution

The experiment will be carried out with a *simple repeated measures design, using counter-balancing* [5]. The participants will be assigned randomly to the two groups – one starting with pair-wise comparisons (AHP) and one with Planning game sorting, and then the groups change method with each other. The participants of the experiment are intended to be 18 PhD Students.

This design involves some threats to internal validity, such as order effects and fatigue or practice effects. Since it is a repeated design it is possible that the order in which the methods are used may affect the result. This can be investigated since the orders in which the two groups perform the task differ and thus the significant difference between the two groups can be calculated and evaluated.

It is also possible that the participants get tired from using both methods. However, since the Planning Game sorting is not very time consuming the exhaustion increase will be minimal compared to only using the AHP. The other possibility is that the participants get "practice" during the experiment and unconsciously get an opinion on the context using the first method, which will affect the result for the second method.

The selection threat is decreased due to the random allocation. Since the two groups will be tested at the same time, the threats of maturation and history can also be eliminated, i.e. the participants' situation and attitude does not change during the experiment. Due to measuring each participant's time to complete the task some might feel pressure, which can affect the results. It might also be the case that the researcher presenting the experiment reveals personal preferences, which might affect the result.

The number of participants in each group being a maximum of only 9 persons reduces the generalizability, i.e. there is an obvious threat that the findings are specific to this particular group or context. Perhaps some of the participants already have used one of the methods in practice or favors one of them. It is also likely that some participants have more experience of the context than others, and thus have a better foundation for the prioritization. Nevertheless, this can also be the case in an industrial situation where the product managers are not equally educated or experienced. However, since the experiment is of such a straightforward design it will be possible to replicate with a larger sample, possibly in an industrial setting, hence the credibility is considered.

The requirements to prioritize will be mobile phone features, for instance WAP, Color display, Calendar, Games etc. This context should be reasonably well known to most people and thereby easy to prioritize. There are also many features on a similar level of detail on a mobile phone and they are thereby easier to extract than for instance TV or VCR features.

The prioritization criteria will be *Importance for me*, which corresponds to how important and interesting the respondent find the feature, and *Added price on the phone*, which is an estimation of how much the feature might add to the price. Note that this is not the same as development cost, which would be difficult for non-developers to estimate. The importance criterion has probably been regarded by most of the participants when buying or considering buying a mobile phone. If the participants are not familiar with some features this implies that it is not as important or valuable to them either. The price criterion might also be accounted for since considering buying and comparing mobile phones gives a clue of what features are most costly.

The group using the AHP will be given one sheet for each criterion with all possible pair-wise combinations of the features to compare. For instance, with 10 different features it will be 45 combinations to compare. In between each pair is a ratio scale [5 4 3 2 1 1/2 1/3 1/4 1/5] where the ratio of the requirements' value is to be circled. A 5 means that the one to the left is very much more important, a 1 means that the features are just as important, while a 1/5 means that the feature to the right is very much more important, and similarly for the second criterion. Some matrix calculations will then create a list with sorted features for each criterion, which can be illustrated in a diagram.

The group using the Planning game will be given two sets of cards with one mobile phone feature written on each. The cards shall be sorted into three piles, first for the importance criterion and then for the cost criterion. The piles will be thought of as (1) necessary, (2) adds to the value and (3) unnecessary, for the first criterion, and (1) very high cost, (2) reasonable cost and (3) low cost / for free, for the second criterion. Within the piles, the cards shall be arranged so that the most important one is at the top of the pile and the less important are put underneath. Putting the three piles together will construct a single list of prioritized features for each criterion.

The time needed for these prioritizations will be no longer than one hour, which will give the participants as much time as necessary to think about each prioritization. When the participants have finished the task, the result will be computed and showed in a diagram. Afterwards, the participants will be given a questionnaire together with the resulting individual prioritization lists so that the subjective views can be captured.

Hypotheses

No 1: The average time to conclude the prioritizations is larger for the participants using the AHP.
No 2: The AHP is better at reflecting the participants' views.
No 3: The ease of use is considered lower for AHP.

The quantitative dependent variable, average time to conclude the prioritizations, will be captured by measuring each participant's time to conclude the task. For each method the mean time for all participants will be calculated. The qualitative dependent variables, reflection of prior views and ease of use, will be captured by a questionnaire where the participants can grade each variable from 1 to 10.

To summarize, this experiment will investigate the difference between two widely used prioritization methods to see if it is possible to receive similar results from a more uncomplicated method as from a more refined one. The results can be very interesting for organizations that need easy and inexpensive methods for the requirement prioritization. If the experiment gives a clear result it will be possible to recommend a method to organizations that are in the position of searching for a prioritization method.

References:

[1] *Challenges in Market-Driven Requirements Engineering - an Industrial Interview Study* by Lena Karlsson, Åsa G. Dahlstedt, Johan Natt och Dag, Björn Regnell and Anne Persson

[2] *An Evaluation of Methods for Prioritizing Software Requirements* by Joachim Karlsson, Claes Wohlin and Björn Regnell

[3] *A Cost-Value Approach for Prioritising Requirements* by Joachim Karlsson and Kevin Ryan

[4] *Extreme Programming Explained* by Kent Beck

[5] *Real World Research* by Colin Robson