Does the Prioritization Technique Affect Stakeholders' Selection of Essential Software Product Features?

Hans Christian Benestad
ExpertWare AS
NO-0274, Oslo, Norway
benestad@expertware.no

Jo E. Hannay
Simula Research Laboratory
Pb. 134, NO-1325 Lysaker, Norway
jo@simula.no

ABSTRACT

Context: To select the essential, non-negotiable product features is a key skill for stakeholders in software projects. Such selection relies on human judgment, possibly supported by structured prioritization techniques and tools. *Goal*: Our goal was to investigate whether certain attributes of prioritization techniques affect stakeholders' threshold for judging product features as essential. The four investigated techniques represent four combinations of granularity (low, high) and cognitive support (low, high). *Method*: To control for robustness and masking effects when investigating in the field, we conducted both an artificial experiment and a field experiment using the same prioritization techniques. In the artificial experiment, 94 subjects in four treatment groups indicated the features (from a list of 16) essential when buying a new cell phone. In the field experiment, 44 domain experts indicated the software product features that were essential for the fulfillment of the project's vision. The effects of granularity and cognitive support on the number of essential ratings were analyzed and compared between the experiments. *Result*: With lower granularity, significantly more features were rated as essential. The effect was large in the general experiment and extreme in the field experiment. Added cognitive support had medium effect, but worked in opposite directions in the two experiments, and was not statistically significant in the field experiment. *Implications*: Software projects should avoid taking stakeholders' judgments of essentiality at face value. Practices and tools should be designed to counteract biases and to support the conscious knowledge-based elements of prioritizing.

Categories and Subject Descriptors

D.2.1 [**Software Engineering**]: [Requirements/Specifications]; D.2.9 [**Software Engineering**]: [Management]

Keywords

Requirements, Prioritization techniques, Essential features, Stakeholders, Field experiment, Robustness

1. INTRODUCTION

Making sound judgments about the importance of proposed product features is a key skill for stakeholders in software projects, from project inception, through construction and evolution. In this study, we wish to draw attention to *essential* product features (denoted *essentials* in this paper); i.e., those product features that, according to stakeholders, cannot be left out if the product vision is to be fulfilled.

Different stakeholders may have different perspectives and interests and may assess the importance of a given feature differently. With prioritization techniques and tools, individual stakeholders' assessments are collected in a structured manner; the goal being to support experts in setting priorities that reflect their knowledge and best judgment.

However, it is well known that expert judgments are subject to contextual biases. For example, in software cost estimation, question format and irrelevant information have been shown to affect estimates significantly [17, 18], and there are a host of other general biases form the judgment and decision-making literature [19, 1] that have been shown to apply also to software estimation [11]. Due to the cognitive commonality between cost estimation, priority judgments, and judgments in general, it is reasonable to assume that judgments of priorities of product features are subject to similar biases. However, there is little empirical evidence on how and when such biases affect priority judgments.

The goal of this study was to investigate how different prioritization techniques affect stakeholders' threshold for judging product features as essential. A lower threshold means that more features are rated as essential. Therefore, a natural outcome measure for the study is the number of features reported as essential from a list of candidate features presented to the subjects.

Prioritization techniques may be classified according to how they differ along the dimensions of *measurement scale*, *granularity*, and *sophistication* (e.g., *cognitive support*) [3]. With regards to these three dimensions, our study focuses on techniques that elicit priorities on an ordinal measurement scale, according to varying granularity (low, high), and according to cognitive support (low, high). We will elaborate on the precise meaning of these dimensions below. The research question for this study is:

> *Does the granularity and cognitive support in prioritization techniques affect stakeholders' thresholds for judging software product features as essential?*

In two controlled experiments we investigated four ordinal prioritization techniques representing the combinations of low/high granularity and low/high cognitive support. We denoted the techniques *Simple dropdown*, *Drag into bins*, *Sortable table*, and *Pairwise comparisons & ranking*.

In addition to offering input to the reflective practice of

ESEM 2012 Lund, Sweden

	high granularity	low granularity
low cognitive support	*Simple dropdown* (T1)	*Sortable table* (T3)
high cognitive support	*Drag into bins* (T2)	*Pairwise comparisons & ranking* (T4)

Table 1: Investigated Prioritization Techniques

software practitioners, this study offers input to empirical software engineering research. First, the study contributes a new objective outcome measure for prioritization studies. Objective outcome measures are important for the validity of empirical studies investigating judgmental biases, and such measures have been scarce in earlier prioritization studies. Second, the study uses underlying characteristics of prioritization techniques as independent variables, rather than the technique itself. This is important, because it facilitates a deeper understanding of the results by making it possible to draw lines from theories in behavioral decision science and psychophysics. We suggest these research design elements as complements to a proposed framework for prioritization studies [5]. Methodologically, the study demonstrates the viability of conducting a decently scaled controlled experiment in a live software engineering context, and illustrates the usefulness of comparing results (effect sizes in particular) with replications in a more artificial context.

The rest of this paper is organized as follows: Section 2 presents the investigated techniques, Section 3 reviews related work, Section 4 discusses the theory underlying the techniques and proposes possible effects in play, Section 5 describes the experiment and the results, Section 6 discusses implications, Section 7 discusses validity issues, and Section 8 concludes.

2. THE INVESTIGATED TECHNIQUES

Prioritization techniques can be classified in terms of the following three dimensions [3]:

1: The *measurement scale* used for prioritizing features. Priorities given on an ordinal measurement scale hold information simply about the relative ordering of features (A is more important than B). Interval or ratio scale priorities add information about the magnitude of differences between features (A is three units more important than B, C is three times more important than D).

2: The *granularity* of the scale denotes how many categories or values are available on which to rate features. For example, on an ordinal scale, higher granularity means that the expert can choose from a larger set of possible ratings (e.g., "Essential", "Important", "Not important" rather just "Essential", "Non-essential")

3: The degree of *cognitive support* for the technique. Several modes of cognitive support exist, and many of them are designed to help the expert prioritize more reliably or consistently. For example, prompting for multiple and overlapping pairwise comparisons should reduce the likelihood for accidental inaccurate assessments.

In this study, our focus is on essentials. This entails that it suffices to offer experts an ordinal scale. The study is designed to investigate if there are effects of varying granularity and cognitive support. For this purpose, it suffices to vary each of these two independent variables binomially (high and low) which gives 2x2 factorial design with four distinct techniques T1–T4, as indicated in Table 1.

The four techniques were implemented using the tool EstimationWeb (`estimationweb.com`), a publicly available web-based tool for estimation, prioritization, and scheduling, developed by our research group. The functionality and visual appearance of the techniques are presented below. In Section 4 we will discuss the theory underlying the design of the techniques, and in Section 5 we will present details on how the techniques were used in the experiment.

2.1 Simple dropdown

With *Simple dropdown* (T1), each line on the prioritization page contains a feature description and a dropdown box offering choices to give the priorities "Essential", "Significant", "Limited", and "Insignificant". Figure 1 shows the tool page for *Simple dropdown* configured for the general experiment.

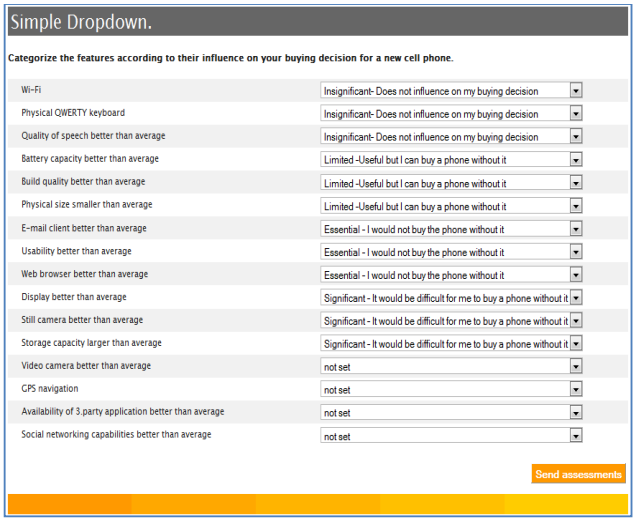

Figure 1: Simple dropdown (T1)

2.2 Drag into bins

With *Drag into bins* (T2), the user drags features into or between categories. The categories and their descriptions in T2 were identical to those in T1, implying equal granularity for these techniques. However, the spatial grouping of features may lead the expert to repeatedly compare features of equal or adjacent ratings, possibly resulting in more reliable priorities. Therefore, T2 is said to offer more cognitive support than T1. Figure 2 shows the tool page for *Drag into bins* configured for the general experiment.

2.3 Sortable table

With *Sortable table* (T3), the features are presented in a table view in which the rows can be dragged and re-arranged according to priority. After sorting the table, the subjects in our experiments used a simple input field to indicate the

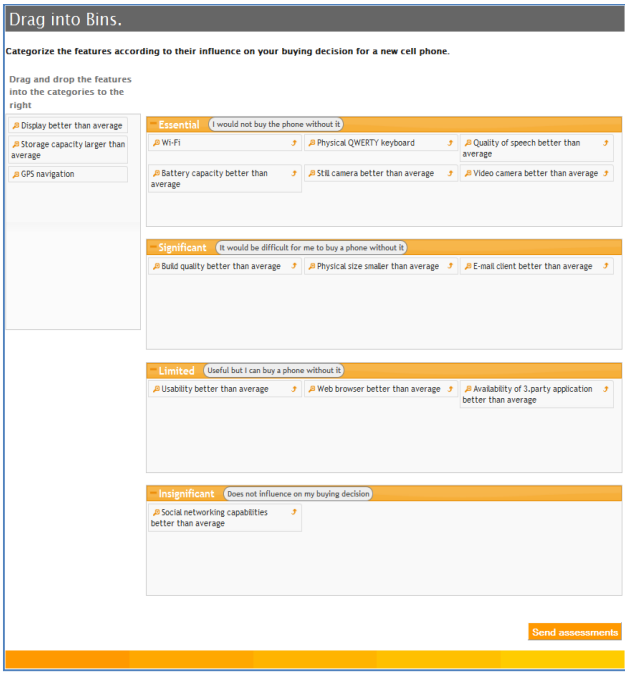

Figure 2: Drag into bins (T2)

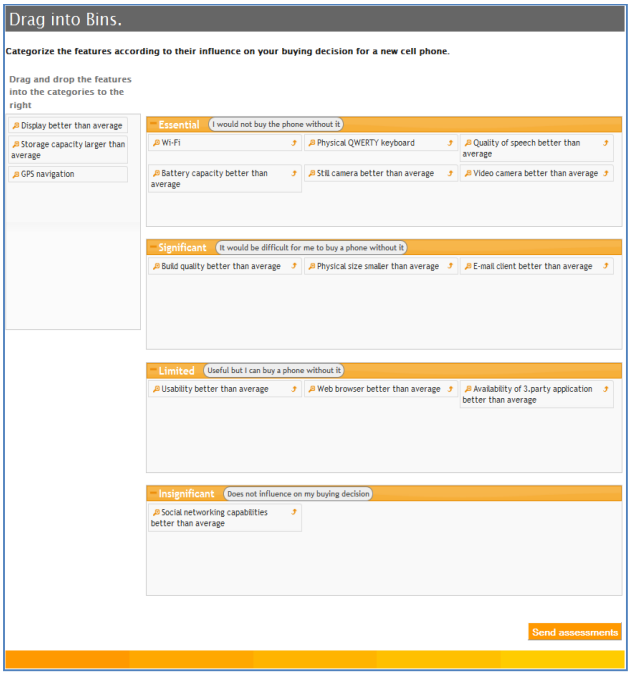

Figure 3: Sortable table (T3)

number of essential features in the now prioritized list. Implicitly, this method classifies features into two categories, "Essential" and "Not essential". Figure 3 shows the tool page for *Sortable table* configured for the general experiment.

2.4 Pairwise comparisons & ranking

Figure 4 shows the tool page for *Pairwise comparisons & ranking* (T4) configured for the general experiment. With *Pairwise comparisons & ranking*, the user is presented with pairs of features and is prompted to indicate the difference in importance on a scale from 1 (equal) to 9 (extreme difference) in either direction of the two features. After these pairwise comparisons, the tool computes a global ranking and displays the results in the table to the right. In this table, the user can adjust the calculated ranking. After any adjustments, the subjects in our experiments indicated the number of essentials in the now sorted table. As with T3, this method implicitly classifies features into two categories, "Essential" and "Not essential". With the pairwise comparison step, some degree of redundancy is introduced, leading to reduced sensitivity for accidentally inaccurate or arbitrary assessment. Also, through the repeated contrasting of feature pairs, users can possibly uncover important criteria relevant for the prioritization. Technique T4 is therefore assumed to offer more cognitive support than T3.

The tool's calculations of ranks in T4 uses the algorithms of the analytical hierarchy process (AHP) [34, 33], complemented with Harkers method to calculate weights and ranks from incomplete pairwise comparisons [15]. AHP is designed to reduce the sensitivity for accidental inaccurate assessments by prompting for redundant pairwise comparisons. The tool was configured to fix the number of pairs to compare to $1.5n$, with n candidate features, i.e., the subjects were asked to perform 24 pairwise comparisons with 16 features. With complete pairwise comparisons, 120 comparisons would have been required. The chosen number was a compromise between obtaining more stable rankings by way of more comparisons, and avoiding fatigue and experiment drop-outs due to too many comparisons. Simulations have shown stable ordinal ranks even below $1.5n$ [15].

3. RELATED WORK

A number of case studies [4, 21, 22, 23, 27, 28] and controlled experiments [6, 24, 25, 31] have been conducted to evaluate prioritization techniques with respect to outcome variables such as time usage and accuracy, but none of the studies directly address our research question. However, since our interest is in the accuracy of essential ratings, studies on the effects of prioritization techniques on the overall accuracy of priorities are still relevant.

The results from such studies do not yet give grounds for strong conclusions. For example, techniques based on pairwise comparisons performed well in some studies, e.g., [23], while in other studies simpler techniques performed equally well [25]. One problem with comparing results for prioritization techniques may be the possible confounding impact of tool support [25]. Another problem is that no objective measures of accuracy have been used in the studies. Subjective measures of accuracy fall short when measuring effects due to factors of which subjects are not consciously aware.

A source of theory related to our research question is the study of judgment and choice within behavioral decision science, see, e.g., [32], with links to psychophysics, a discipline within psychology that investigates the relationship between physical stimuli and perceptions, see, e.g., [9]. We discuss effects predicted by this theory in the next section.

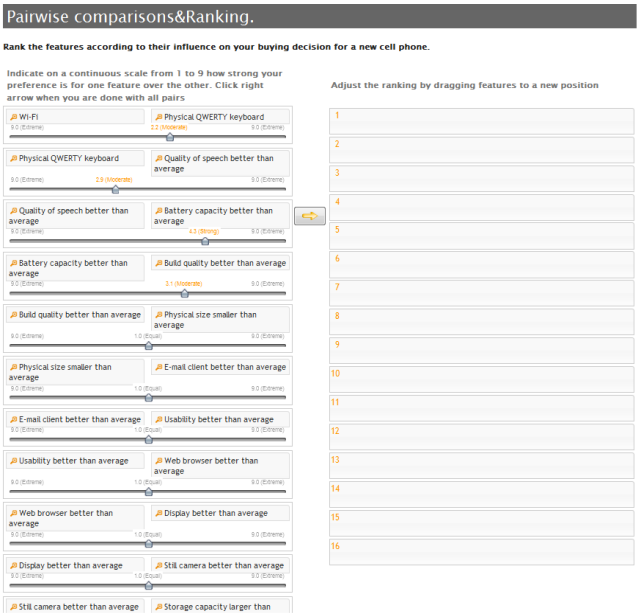

Figure 4: Pairwise comparisons & ranking (T4)

4. THEORETICAL PROPOSITIONS

If properties of the prioritization technique do not affect subjects in their assessment, there should be no difference in the judgment of essential features between techniques. However, based on a substantial body of research on unconscious factors that produce biases in related domains and on related tasks [11], we propose that unconscious factors produce biases also in prioritization, and, further, that various prioritization techniques embody these factors. However, we also propose that cognitive support represents a counterweight to these unconscious mechanisms, in that it strengthens the deliberate and conscious part of the prioritization task. In the following, we discuss the theoretical foundation underlying this study's variation in granularity and cognitive support.

4.1 Effects of granularity

The theoretical basis for effects of variation in granularity is *range-frequency theory* [30, 32]. This theory explains judgments of a set of stimuli as a compromise between two mental principles: Distributing features equally over categories (the frequency principle) and distributing categories over equally sized ranges of features (the range principle).

The frequency principle gives rise to the *equal frequency bias*, by which there is a tendency in people to distribute stimuli equally over available categories in a rating situation. This equal frequency bias affects judgments in the presence of *contextual skewing* [30]: Consider a distribution of features according to a person's unbiased view of importance. If this distribution is positively skewed so that more features are viewed as less important than more important, this will induce a tendency in the subject (a bias) to rate a given feature higher (to equalize the distribution), than if the same feature occurred in the context of a balanced or negatively skewed distribution. Figure 5 exemplifies a positively skewed distribution. In our setting, it is reasonable to expect positive skewing in unbiased priorities, since we are asking for essentials, but this is not known *a priori*.

As a counterpoint to the frequency principle, the range principle induces a tendency in people to perceive categories as constituting equal ranges of features. This entails that categories are spaced out equally along a person's perception of ranked features, and thus the "Essential" category will be perceived as smaller and more extreme the more categories are introduced. The range principle also pertains to the stability of the perception of "Essential". If the term is not solidly based in a person's mental model [8, 16], it is more likely that the category denoting "Essential" and the rating scale itself will change meaning in different circumstances. Roughly, for skewed distributions and with few available categories, more weight is put on the frequency principle, rather than the range principle [30].

Together with variations in granularity, the frequency principle and range principle produce the *category effect*. This effect is the tendency of the effect of contextual skewing to diminish when increasing the number of categories. Thus, the count of features in the highest category (essential) would decrease with higher granularity, because of the tendency to distribute equally. This effect is directly relevant in our study because the number of available categories differs between the treatment groups. The use of an explicit "Essential" threshold in T3 and T4 may increase the influence of the range principle on these techniques, since they prompt the user to set this category boundary explicitly.

If subjects distributed features uniformly across categories, twice as many essentials for two-category techniques (T3 and T4) than for four-category techniques (T1 and T2) would be measured. We note that in situations where categories are described with relative labels, e.g., 1 to 4 or low to very high, such a result could be considered rational, rather than biased. In contrast, our observation in many software engineering contexts is that one attempts to assign a precise definition to categories, particularly to the category for essentials. (Assigning specific meanings to ordinal categories actually gives the categories a nominal flavor.) The tendency to distribute features equally over the available categories in such cases is correctly denoted a bias.

Even when influenced by the equal frequency bias, subjects may be able to reliably (i.e., consistently) rank the features on an ordinal scale. In contrast, imagine a situation where features were allocated to available categories by a random process. The predicted result of such a process would be a uniform distribution over the available categories. Hence, *randomness* and the equal frequency bias are distinct effects that coincide. It is reasonable to assume that more randomness will be present when people have less knowledge or weaker opinions about the subject matter. In software projects, where the assumption is that people know product goals and the features needed for their realization, indications of substantial randomness in priorities would be a cause for concern.

4.2 Effects of cognitive support

The theoretical basis for the cognitive support expressed in T2 and T4 is the *selective accessibility process* [29, 37]. This process explains the cognitive steps in comparisons as an initial holistic judgment in which a person determines whether two things to be compared are holistically similar or dissimilar, followed by a step in which a person's knowledge is accessed in a confirmatory way according to the holistic judgment, which then leads to either an assimilation or a

contrast as the end result of the comparison. As in many cognitive mechanisms, there are unconscious mechanisms at play in this process as well, but a substantial distinction is that the strength of the effects of this process depend on domain-specific knowledge. In contrast to the unconscious biases-inducing factors in this study, which relate to general psychological mechanisms that are robust, the unconscious mechanisms of the selective accessibility process are influenced by knowledge of the features under judgment and can therefore be modified to improve performance. This is arguably the basis for building deliberate and conscious processes (i.e., expertise) in planning tasks such as estimation and prioritization [12].

Comparisons are the essence of virtually all judgment tasks, and T2 and T4 prompt explicit comparisons. One would therefore expect stronger signals over randomness by this fact alone, and therefore more reliable ratings. For example, with T2 the subject can more easily compare a feature with features of equal ranking. In addition, T2 and T4 prompt repeated comparisons which should successively make feature-relevant knowledge more accessible. With less randomness, it can be predicted that the average number of essentials will decrease in the case of a bell-shaped or positively skewed distribution of features to importance.

4.3 Summary

Figure 5 summarizes the discussed effects. First, the graph exemplifies a skewed distribution, where the proportion of tasks is shown as a function of feature importance; the latter measured by some objective measure, such as the return of investment for implementing a feature.

When a subject sets a threshold for what constitutes an essential feature with a low-granularity technique (dashed line), the equal frequency bias, the category effect, and randomness pull in the direction of a larger number of essentials, i.e., a larger area under the curve, when compared to a judgment made with a high-granularity technique (rightmost solid line). These effects are to an unknown extent counteracted by the cognitive support of the techniques and by the definition of "Essential" being kept constant.

The figure is not intended to suggest a precise position for the dashed line; rather it illustrates that its actual position is some context-dependent compromise between the effects illustrated as arrows, with a higher barrier at the rightmost solid line and a lower barrier at a position corresponding to a uniform distribution over categories (leftmost solid line).

5. EXPERIMENT

5.1 Overview

The cognitive mechanisms that underlie prioritizing, are basic mechanisms in the sense that they have been demonstrated to exist in artificial settings where other effects are controlled for. In naturalistic settings, however, such effects might interact with other effects to give combination effects, emergent effects [36] or cancellation of effects [35]. This brings uncertainty to what one is observing in field settings [10, 26] in terms of such biases. In field studies, it is therefore important to control for the variation in artificiality by subjecting the experiment design to both an artificial setting and the intended field setting. If proposed effects that are demonstrated in the artificial setting recur in the field setting, it is likely that the effects are robust. If

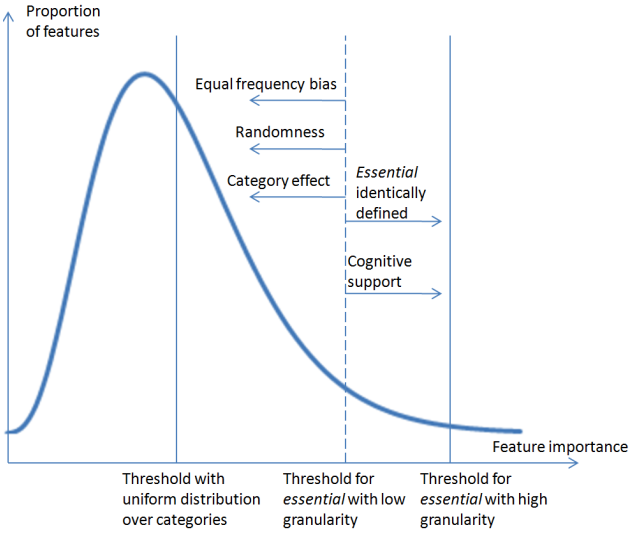

Figure 5: Summary of proposed effects

the proposed effects do not recur, or other effects manifest themselves in the field setting, then this gives grounds for further deliberations [14].

We therefore conducted both an artificial experiment and a field experiment using the same prioritization techniques. The experiment material was prepared so that subjects across treatments received identical definitions of what it is that a feature is essential. In the first experiment, 94 subjects were asked to rate the importance of 16 cell phone features that would pertain to their decision for purchasing a new phone. In the second experiment, 44 subjects were recruited from the pool of functional experts at a large development project in the public sector in Norway. The subjects were asked to assess the contribution of 16 proposed features to the fulfillment of the project's vision for the software product. In both experiments the subjects were randomly allocated to one of four treatments groups, corresponding to the four prioritization techniques investigated.

The experiments followed a 2x2 factorial design, with Granularity (low, high) and Cognitive support (low, high) as independent variables and the count of features judged as essential as dependent variable. Two-way ANOVA on the rank-transformed dependent variable was used to analyze the data.

5.2 Recruiting subjects

Experiment 1. The subjects were convenience sampled. We explained the goal and the basics of the experimental design to the head of administration of our research institution. All 120 employees were invited by email, explaining the overall goal and the relevant procedures. This population represents employees in the domain of ICT research. The average age of the invited employees was 39 years, and they represented 25 nationalities. Fifty-nine colleagues agreed to participate. Concurrently, we contacted a Polish software house with which our research group has collaborated. The same procedures were followed. The average age of the invited developers was slightly below 30 years, and all of them were Polish nationals. Thirty-five developers agreed to participate. We offered compensation based on standard hourly

rates and an estimate of 20 minutes time usage per subject.

Experiment 2. We recruited participants from a large agile development project which was about to engage in feature prioritization for an upcoming release. The authors already had a research collaboration established with this project [2, 13]. We sent a request to the project manager, explaining the goal and the basics of the experimental design. The request was forwarded to one of the project's three product owners who suggested a list of 60 project members who participate in prioritization activities on a regular basis. An email was sent to the potential subjects, explaining the purpose of the experiment and the procedures to follow. We explained that neither agreeing to, declining, or ignoring the invitation would have negative consequences for the invited person. We did not offer incentives for participation other than contribution to project-relevant research and experience with prioritization techniques and tools. Eventually, 44 people agreed to participate in the experiment. The subjects were free to perform the task at the time and location of their own choice within 8 days.

5.3 Experimental material

Experiment 1. 16 features of modern cell phones were identified by the authors based on the authors' judgment of what might be important to the experimental population in a buying situation. The actual features are displayed in Figure 1.

Experiment 2. We selected 16 features from the product queue recorded in the project's issue tracker. The product owner assisted in making the selection based on two criteria.

1: Expected effort is in the same order of magnitude for all features because it makes less sense to prioritize between features at different abstraction levels [3].

2: It is possible for the subjects to understand the basics of the feature through a brief description.

This was to ensure that experimental conditions w.r.t. time and restrictions in collaboration were identical over the two experiments, while at the same time ensuring realism in the field setting. The short description already given in the issue tracker could be used almost unchanged, but some of the texts were improved with respect to clarity and consistency. Three examples follow, translated from Norwegian:

> *As a user/employer, I can execute a check of the salary file for logical errors so that these can be corrected before the file is submitted to the system.*
>
> *As an agency official, I can reconstruct NAV and AORD information so that I can see which data was registered at a certain point in time.*
>
> *As a customer service operator I want to have phone calls automatically recorded in the person log when members call SPK and identify themselves with a social security number, so that I spend less time recording the call.*

Table 2 outlines the instructions given to the subject for the four techniques in the two experiments.

5.4 Execution

In both experiments, subjects were randomly allocated to four equally sized treatment groups, corresponding to each of the four techniques under investigation. The study administrator sent an email containing a brief general description of the study and a personalized link to the web form containing the experimental material:

> *[intro]...The purpose is to investigate whether the format of various prioritization techniques affects the priorities. The participants have been divided into four groups, and through random allocation you have been assigned the technique [Simple dropdown | Drag into bins | Sortable table | Pairwise comparisons & ranking]. Once you have started, it is important that you complete without interruptions. Please spend the time needed for you to feel comfortable about your answers. We ask that you work individually and not use other sources than the material we present. Click the link below to start.*

Technique Experiment	Prioritization instruction	Categorization
T1 and T2 Experiment 1	*Categorize the features according to their influence on your buying decision for a new cell phone*	Essential – I would not buy the phone without it Significant – It would be difficult for me to buy a phone without it Limited –Useful but I can buy a phone without it Insignificant– Does not influence my buying decision
T3 and T4 Experiment 1	*Rank the features according to their influence on your buying decision for a new cell phone*	How many of the above features are essential – you would not buy the phone without it. Please type in one number between 0 and 16
T1 and T2 Experiment 2	*Categorize the tasks according to their contribution to the project's vision, the way you perceive the vision*	Essential– The vision will not be fulfilled without it Significant – The vision will be hard to fulfill without it Limited – Useful, but the vision can be fulfilled without it Insignificant–Does not influence the fulfillment of the vision
T3 and T4 Experiment 2	*Rank the tasks according to their contribution to the project's vision, the way you perceive the vision*	How many of the above elements are essential – the vision will not be fulfilled without it. Please type in one number between 0 and 16

Table 2: Experiment instructions (translated from Norwegian for Experiment 2)

In Experiment 1, all 94 subjects submitted their priorities within 8 days. Reminders were sent by e-mail to nonrespondents after 3 and 7 days. In Experiment 2, 44 out of 60 invited subjects replied within 8 days. One reminder was sent to nonrespondents after 6 days. Sixteen invited subjects did not submit their responses. A web-based experiment that gives subjects freedom to choose time and place for completing the experimental tasks imposes some threats to validity, as further discussed in Section 7. Such validity threats would have been reduced or eliminated had all subjects been under our supervision. However, having 44 domain experts from

one project meet at the same place and time to participate in an experiment was neither possible, nor desirable, in the field setting.

5.5 Analysis

in the analysis, the two independent variables are denoted *gr* (granularity) and *cogn* (cognitive support), and the dependent variable is denoted *ess* (the number of features judged as essential). Two-way ANOVA is used to identify effects and interaction effects of the independent variables on *ess*. A rank-converted measure of *ess* was used to avoid sensitivity to the ANOVA normality requirements. A post-hoc Normal QQ plot indicated a non-normal distribution of residuals in Experiment 1 but not in Experiment 2. However, we employed rank-converted measures of *ess* in the ANOVA for both experiments. The analysis was executed using the statistical analysis package R, Version 2.10.1.

5.6 Deviations

In three cases (Experiment 1) and four cases (Experiment 2), the subject misunderstood the instruction of providing the number of essential features. We noticed this misunderstanding on reception of the web forms (within a few minutes after submission) and asked the subject by e-mail to update the response.

5.7 Results

Descriptive statistics for variable *ess* in the two experiments are shown in Tables 3 and 5, while the ANOVA results are shown in Tables 4 and 6.

Experiment 1: The average number of essential ratings increased by 56% (statistically significant) with low-granularity techniques (T3 and T4) compared to high-granularity techniques (T1 and T2). The effect size measured by Cohen's d is 0.73, which is in the medium category of effect sizes reported in software engineering experiments [20].

The average number of essential ratings increased by 32% (statistically significant) with low cognitive support (T1 and T3) compared to high cognitive support (T2 and T4). The effect size measured by Cohen's d is 0.43, which is in the lower end of medium effect sizes reported in software engineering experiments [20].

Statistic	Overall	T1	T2	T3	T4
Mean	4.1	3.4	2.9	5.8	4.1
Stddev	2.7	1.8	1.8	3.1	2.9
Range	0-12	0-7	0-7	1-11	0-12

Table 3: Descriptive statistics for *ess* Experiment 1

	Df	Sum sq	Mean sq	Sq F value	Pr(>F)
gr	1	7458	7458	11.18	0.0012**
cogn	1	3812	3812	5.71	0.019*
gr : *cogn*	1	574	574	0.86	0.36
Residuals	90	60057	667		

Table 4: ANOVA results for Experiment 1

Experiment 2: The average number of essential ratings increased by 195% (statistically significant) with low-granularity techniques (T3 and T4) compared to high-granularity techniques (T1 and T2). The effect size measured by Cohen's d is 2.40, which is in the highest category of effect sizes reported in software engineering experiments [20].

The average number of essential ratings increased by 33% (not statistically significant) with high cognitive support (T1 and T3) compared to high cognitive support (T2 and T4). The effect size measured by Cohen's d is 0.50.

Statistic	Overall	T1	T2	T3	T4
Mean	5.8	3.0	2.9	7.8	9.50
Stddev	3.75	2.22	2.18	2.15	2.78
Range	0-14	0-7	0-7	5-10	5-14

Table 5: Descriptive statistics for *ess* Experiment 2

	Df	Sum sq	Mean sq	Sq *F* value	Pr(>*F*)
gr	1	4420	4420	64.20	<0.001***
cogn	1	55	55	0.80	0.38
gr : *cogn*	1	110	110	1.60	0.21
Residuals	40	2754	69		

Table 6: ANOVA results for Experiment 2

In summary, the results showed that the number of essential ratings significantly increased with lower granularity. The effect size was greater in Experiment 2 (the software engineering context) than in Experiment 1. Added cognitive support affected the number of essential ratings in both experiments but in opposite directions and with statistical significance only in Experiment 1. The results showed no interaction effect between granularity and cognitive support. Re-doing the analysis with the original outcome measure *ess* (the ANOVA analyses used a rank-converted measure) did not change any of the significance levels.

6. DISCUSSION

The results showed that with lower granularity significantly more features are rated as essential, in line with the equal frequency bias. In Experiment 1, the effect on essential ratings was of a magnitude that could be predicted from the discussion in Section 4 and summarized in Figure 5. Halving the granularity increased the essential ratings, but not by 100% as a pure uniform distribution strategy would have predicted. It may be the case that the definition of "Essential" (*I would not buy the phone without it*) is relatively stable in people's mental models, thus helping in limiting, e.g., the category effect that pulls in the direction of more reported essentials. Also, a higher level of cognitive support had the predicted direction in Experiment 1.

The result for granularity in Experiment 2, on the other hand, is extreme. The effect of reducing granularity is larger than even a full adherence to the principle of equal distribution over categories would have accounted for. In this case it seems that the definition of "Essential" (*The vision will not be fulfilled without it*) is more volatile and perhaps subject to the category effect. In the studied project, the term "vision" is central in the release planning and prioritization processes. Project management assumes that the vision is

understood and shared between the key stakeholders; however, the present result indicates that this assumption may not be met. Indeed, in a post-hoc analysis, we measured the intra-rater agreement of the priorities by Kendalls W, giving values of 0.39 and 0.33 for the two experiments, respectively. These are both in the range of low to medium correlation, and it is interesting to note that individuals ranking their personal cell phone preferences agree more than do software project stakeholders assumed to share a common vision. A qualitative study confirms that there may be challenges in maintaining a common vision in the project [13]. The effect of cognitive support was negative in the field experiment, but not statistically significantly so.

In summary, the correspondence in results between the artificial experiment and the field experiment support the proposition that general biases also occur in our field setting. The differences between the two experiments give valuable insight into the specificities of the field setting that should be investigated further.

An important question is whether the subjects were able to assess reliably the relative importance of features. If so, the results can be explained by the equal frequency bias. Alternatively, the results were influenced by a high degree of randomness, which would be a cause for larger concern. With reliable ordinal rankings, the most important features would still be selected for development, but this would not necessarily be the case with randomness in priorities.

The differences in effect sizes between the two experiments could possibly be explained by extensive randomness in the priorities given in Experiment 2. Future plans are to re-execute the experiment sessions, which would allow us to measure the intra-rater agreement scores of the essentiality ratings. A lower score in the field context than in the phone feature context would support the proposition that more randomness was present in the field.

At the outset, the implications of the study results seem important. If stakeholders' threshold for judging features as essential is sensitive to the prioritization technique and perhaps also to the questioning format in general, this could be an important obstacle against collecting trustworthy judgments as input to release planning. Indeed, it is possible that the results point to a root cause for software failure archetypes such as "Unable to meet the user's needs".

It is important to note that any direction of bias can be harmful. Too strict thresholds for judging features as essential can imply that required functionality is not prioritized over more dispensable features or even gold-plating features. Thresholds set too loosely could lead to projects being prematurely stopped due to outlooks of severe cost overruns or could mean that falsely reported essentials take the place of true essentials when the project budget becomes tight.

In practice, projects have mechanisms to compensate for judgmental biases of the kind demonstrated in this study. First and foremost, group discussions and other forms of broad-banded communication can enable stakeholders to counteract the biases through group discussions and clarifications. Such communication is more difficult to achieve in the largest projects and in projects where stakeholders cannot meet frequently. Unfortunately, these are exactly the kinds of projects that are already considered at risk.

The results are also relevant for discussions on how to handle change and feature requests in the context of a commercial development contract. Such contracts sometimes describe different procedures and conditions for handling change or feature requests of different importance. For example, a contractor might commit to expediting the development of essentials or to develop them as part of a fixed-price contract. Biases in judgments of essentiality could therefore easily have commercial and legal consequences.

Providing recommendations concerning the biases are not straightforward. Since biases in both directions can be harmful, and the "correct" prioritization is generally not known, it is not possible to give normative recommendations, such as "Use a large number of categories". Furthermore, improvements in the accuracy of essential ratings are more difficult to assess then in the context of cost estimation. While cost estimates can be compared with an objective measure of actual expenditure, it remains a matter of subjective opinion to determine whether a requirement was eventually essential. Currently, our best advice would be to triangulate priorities by combining different prioritization techniques. On major deviations between stakeholders or between techniques, stakeholders should meet to clarify their views so that the responsible product owner can make the final decisions based on more and better information. Being as precise as possible in the priority guidelines, category definitions, and descriptions of features is likely to help, but as the results from the present study demonstrate, such measures are unlikely to fully remove the biases.

On the other hand, more effort should be spent on investigating ways to improve cognitive support so as to influence the cognitive elements that rely on knowledge and expertise. This would strengthen the conscious signal in prioritization to overcome the noise of unconscious biases [12]. This would also open up the possibility to train people in prioritizing, by deliberatively targeting [7] the appropriate elements in the selective accessibility process [29].

7. VALIDITY ISSUES

Conclusion validity. Unreliable measurement induced by the unsupervised experiment context can be a threat to conclusion validity. With unsupervised execution, subjects may more easily break the rules by collaborating, answering at random, answering destructively, or consulting information outside of the experimental material. We have no explicit reason to believe such problems were prevalent, given the well-willingness to participate and the professionalism of the participants. To some degree, the use of robust methods of analysis would have counteracted effects of outliers in the data due to such problems.

Internal validity. At present, we have not identified any large threats to establishing internally valid treatmentoutcome relationships from the experiment. However, future investigations may reveal confounding or missing variables that should have been included in the analysis. We cannot obtain more information about underlying relationships from the experimental results. We would have liked to have gathered qualitative data to complement the quantitative analyses; however, we preferred to concentrate our limited time of access to subjects on collecting better quantitative data.

Construct validity. Although our study is theoretically based, our constructs are only very informally defined; if at all. This is a general short-coming in empirical software engineering. In particular, there are many ways of adding cognitive support to prioritization techniques, and

effects might differ between different operationalizations of the concept. Further research is needed to establish valid constructs for the informal concepts involved in our study. For now, the results can only very informally be generalized through construct validity. However, the results from Experiment 1 strengthens the case that the observed effects are indeed domain independent and instances of robust psychological effects between meaningful constructs, at least for variations in granularity.

External validity. For Experiment 2, the population was restricted to one specific development project. Hence statistical inference can be used to generalize to this population but not automatically to other software development projects. For granularity, we have shown effects of varying between two and four categories. Whether similar effects occur for other specific levels of granularity remains to be investigated. We do not believe the experimental material or context have provoked or exaggerated the results. On the contrary, we paid significant attention to articulating the instructions, the category definitions, and the feature descriptions precisely and in an understandable manner. It is likely that our observations can cautiously be transferred to variations over our experiment variables; hence we postulate a modest degree of external validity.

It is also possible to use the logic of case studies to discuss external validity [38]. Earlier, we have argued that the investigated project can be seen as a critical case in the class of large and agile software development projects [13]. The project is a prestige project in the Norwegian public sector, attracting the best skilled workers, both on the client and contractor side. Great attention has been put on sharply defining the scope and vision for the project. Critical case logic implies that other, less fortunate projects in the same class are likely to face similar or more severe challenges.

8. CONCLUSIONS

We have conducted two controlled experiments to investigate whether certain attributes—granularity and cognitive support—of prioritization techniques affect stakeholders' thresholds for judging product features as essential. In an experiment asking subjects to pick essential mobile phone features, the number of reported essentials increased by around 50% when granularity decreased from four to two categories. The effect was extreme195% in the experiment conducted in a realistic software engineering context.

It seems that subjects in both experiments had a tendency to distribute features equally across available categories (known as the equal frequency bias), despite a clear and constant definition of what "essential" means across treatments. The extreme effect in the software engineering context indicates that stakeholders were able to make absolute judgments of essentiality only to a very limited degree and instead resorted to, at best, ordinal ranking. Additionally, randomness in assigned priorities can explain such results.

For cognitive support, the results are less conclusive. In the phone feature experiment, adding cognitive support resulted in a statistically significant decrease in the number of essential ratings. An opposite, but not statistically significant, effect occurred in the field experiment. The most immediate explanation is that the potential effect of cognitive support was overshadowed by the equal frequency bias and randomness in the latter context.

Incorrectly picking essential product features can have harmful effects for software projects. This study has shown that contextual biases can have large effects when stakeholders assign priorities to software product features. We believe this study has shown the importance of designing and employing practices that counteract or manage such effects, and also the necessity to strengthen the conscious elements of the task of prioritizing.

Acknowledgments

The authors are grateful to Krzysztof Rolak, Mona Hegreberg, and Ottar Hovind for accepting and helping us to conduct the experiment in their organizations and to the participants from PGS Software, the SPK/Perform project, and Simula Research Laboratory, respectively. We thank Torleif Halkjelsvik at the Department of Psychology, University of Oslo, for pointing out relevant empirical studies within psychology. The work was partly funded by the Simula School of Research and Innovation.

9. REFERENCES

[1] J. S. Armstrong, editor. *Principles of Forecasting: A Handbook for Researchers and Practitioners*. Kluwer Academic Publishers, 2001.

[2] H. C. Benestad and J. E. Hannay. A comparison of model-based and judgment-based release planning in incremental software projects. In *Proc. 33rd Int'l Conf. Software Engineering (ICSE 2011)*, pages 766–775. ACM, 2011.

[3] P. Berander and A. Andrews. Requirements prioritization. In *Engineering and Managing Software Requirements*, chapter 4, pages 69–94. Springer, 2005.

[4] P. Berander and P. Jönsson. Hierarchical cumulative voting (hcv) prioritization of requirements in hierarchies. *Int'l J. Software Engineering & Knowledge Engineering*, 16:819–849, 2006.

[5] P. Berander, K. A. Khan, and L. Lehtola. Towards a research framework on requirements prioritization. In *Proc. 6th Conf. Software Engineering Research and Practice in Sweden*, pages 39–48, 2006.

[6] A. S. Danesh and R. Ahmad. Study of prioritization techniques using students as subjects. In *Int'l Conf. Information Management and Engineering*, pages 390–394. IEEE Computer Society, 2009.

[7] K. A. Ericsson. The influence of experience and deliberate practice on the development of superior expert performance. In K. A. Ericsson, N. Charness, P. J. Feltovich, and R. R. Hoffman, editors, *The Cambridge Handbook of Expertise and Expert Performance*, chapter 38, pages 683–703. Cambridge Univ. Press, 2006.

[8] D. Gentner and A. L. Stevens, editors. *Mental Models*. Lawrence Erlbaum Associates, Inc., 1983.

[9] G. A. Gescheider. Psychophysical scaling. *Annual Review of Psychology*, 39:169–200, 1988.

[10] G. Gigerenzer and P. M. Todd, editors. *Simple Heuristics that Make Us Smart*. Oxford University Press, 1999.

[11] T. Halkjelsvik and M. Jørgensen. From origami to software development: A review of studies on judgment-based predictions of performance time. *accepted to Psychological Bulletin*, 2011.

[12] J. E. Hannay. Better software effort estimationa matter of skill or environment? *To be submitted to IEEE Trans. Software Engineering*, 2012.

[13] J. E. Hannay and H. C. Benestad. Perceived productivity threats in large agile development projects. In *Proc. 4th Int'l Symp.Empirical Software Engineering and Measurement (ESEM)*, pages 1–10. IEEE Computer Society, 2010.

[14] J. E. Hannay and M. Jørgensen. The role of deliberate artificial design elements in software engineering experiments. *IEEE Trans. Software Eng.*, 34:242–259, Mar/Apr 2008.

[15] P. T. Harker. Incomplete pairwise comparisons in the analytic hierarchy process. *Mathematical Modelling*, 9(11):837–848, 1987.

[16] P. N. Johnson-Laird. *Mental Models: Towards a Cognitive Science of Language, Inference, and Consciousness*. Cambridge Univ. Press, 1983.

[17] M. Jørgensen and S. Grimstad. The impact of irrelevant and misleading information on software development effort estimates: A randomized controlled field experiment. *IEEE Trans. Software Eng.*, 37(5):695–707, 2011.

[18] M. Jørgensen and T. Halkjelsvik. The effects of request formats on judgment-based effort estimation. *J. Systems and Software*, 83(1):29–36, 2010.

[19] D. Kahneman and S. Frederick. A model of heuristic judgment. In K. J. Holyoak and R. G. Morrison, editors, *The Cambridge Handbook of Thinking and Reasoning*, pages 267–294. Cambridge Univ. Press, 2004.

[20] V. B. Kampenes, T. Dybå, J. E. Hannay, and D. I. K. Sjøberg. A systematic review of effect size in software engineering experiments. *Information and Software Technology*, 49(11–12):1073–1086, Nov. 2007.

[21] J. Karlsson. Software requirements prioritizing. In *2nd Int'l Conf. Requirements Engineering (ICRE'96)*, pages 110–116. IEEE Computer Society, 1996.

[22] J. Karlsson, S. Olsson, and K. Ryan. Improved practical support for large-scale requirements prioritising. *Requirements Engineering*, 2:51–60, 1997.

[23] J. Karlsson, C. Wohlin, and B. Regnell. An evaluation of methods for prioritizing software requirements. *Information & Software Technology*, 39(14–15):939–947, 1998.

[24] L. Karlsson, M. Höst, and B. Regnell. Evaluating the practical use of different measurement scales in requirements prioritisation. In *Proc. 2006 ACM/IEEE Int'l Symp. Empirical Software Engineering*, ISESE '06, pages 326–335. ACM, 2006.

[25] L. Karlsson, T. Thelin, B. Regnell, P. Berander, and C. Wohlin. Pair-wise comparisons versus planning game partitioning—experiments on requirements prioritisation techniques. *Empirical Software Engineering*, 12:3–33, 2007.

[26] G. Klein. Developing expertise in decision making. *Thinking & Reasoning*, 3(4):337–352, 1997.

[27] L. Lehtola and M. Kauppinen. Empirical evaluation of two requirements prioritization methods in product development projects. In T. Dingsøyr, editor, *Software Process Improvement*, volume 3281 of *Lecture Notes in Computer Science*, pages 161–170. Springer, 2004.

[28] L. Lehtola and M. Kauppinen. Suitability of requirements prioritization methods for market-driven software product development. *Software Process: Improvement and Practice*, 11:7–19, 2006.

[29] T. Mussweiler. Comparison processes in social judgment: Mechanisms and consequences. *Psych. Review*, 110(3):472–489, 2003.

[30] A. Parducci and D. H. Wedell. The category effect with rating scales: Number of categories, number of stimuli, and method of presentation. *J. Experimental Psychology: Human Perception and Performance*, 12(4):496–516, 1996.

[31] A. Perini, F. Ricca, and A. Susi. Tool-supported requirements prioritization: Comparing the ahp and cbrank methods. *Information and Software Technology*, 51(6):1021–1032, 2009.

[32] E. C. Poulton. *Behavioral Decision Theory: A New Approach*. Cambridge University Press, 1994.

[33] T. L. Saaty. *The Analytic Hierarchy Process: Planning, Priority Setting, Resource Allocation*. McGraw-Hill, 1990.

[34] T. L. Saaty. *Multicriteria Decision Making: The Analytic Hierarchy Process: Planning, Priority Setting, Resource Allocation*. RWS Publications, 1990.

[35] W. R. Shadish, T. D. Cook, and D. T. Campbell. *Experimental and Quasi-Experimental Designs for Generalized Causal Inference*. Houghton Mifflin, 2002.

[36] H. A. Simon. *The Sciences of the Artificial*. MIT Press, third edition, 1996.

[37] F. Strack and T. Mussweiler. Explaining the enigmatic anchoring effect: Mechanisms of selective accessibility. *J. Personality and Social Psychology*, 73(3):437–446, 1997.

[38] R. K. Yin. *Case Study Research: Design and Methods*, volume 5 of *Applied Social Research Methods Series*. Sage Publications, third edition, 2003.