System Requirements Prioritization Framework

Michel Högberg, Mira Kajko-Mattsson, Paulina Persson, Anne Håkansson
School of Electrical Engineering and Computer Science, KTH Royal Institute of Technology
Stockholm, Sweden
e-mail: {michelh, mekm2, pauper, annehak}@kth.se

***Abstract*—Prioritization of system requirements is pivotal for coping with limited project resources. A well-structured and adequate prioritization method ensures that the most critical requirements get addressed first. Unfortunately, today, there are very few methods that are dedicated to requirements prioritization. This paper suggests a framework for prioritizing system requirements. The framework is called System Requirements Prioritization Framework (SRPF). It consists of eight components each representing a specific angle of the prioritization effort. Its components are (1) Input, (2) Stakeholders, (3) Prioritization Criteria, (4) Prioritization Methods, (5) Environment, (6) Resources, (7) Priority Scales, and (8) Urgency Levels. Our goal is to create a framework aiding companies in making structured and objective prioritization decisions. The theory on the framework's constituents and structure got educed in four consecutive exploration steps within the industry. The framework then got evaluated within the industry. Altogether, seventeen companies have been involved in the framework's exploration and five companies have been involved in the framework's evaluation. The evaluation results show that the framework is highly relevant and useful to the organizations studied.**

Keywords-Software project; development; prioritization method; decision making; customer benefit; corporate value.

I. INTRODUCTION

Projects have limited resources in terms of staff, time, and budget. Hence, it is not always possible to implement all the requirements in the current release or in the next coming releases [1]. Priorities must be made both by the stakeholders stating the requirements and the stakeholders attending to the requirements. Unfortunately, today, requirements are not always prioritized in an effective manner or they are not prioritized at all [2].

There are many reasons for the ineffectiveness of the requirements prioritizations. Stakeholders that state requirements believe that all their requirements are equally important. Hence, they are not always willing to prioritize them [3]. Stakeholders who attend to the requirements, on the other hand, do not always have adequate support for making priorities. Many try their best by using whatever tools they have. Many, however, still conduct prioritization in an ad hoc manner, often based on the will of some strongly opinionated individuals [4]. Or, as Stephen Covey claims, many companies prioritize what is on their schedule, and they do not schedule their priorities [5].

Lack of prioritization support may lead to many problems, such as (1) disagreements with respect to assigning priority [6][7], (2) too strong a subjectivity when prioritizing [8][9], (3) decisions conducted in uncertain conditions [10]–[12], (4) difficulty to reprioritize due to newly reported acute projects [6], (5) compliance among the prioritized requirements [7][9], (6) difficulties to implement all the requirements in the backlog, and many other problems. At its worst, the resources available will get quickly consumed on implementing less urgent requirements thus leaving scarce resources to the implementation of more urgent, business value adding requirements.

Priorities are very powerful. Even if companies have good resources, they may quickly jeopardize their productivity, if they spend them on requirements that have little bearing on the financial business health or other form of revenue or benefit of the software company and/or its customers. Despite this, requirements prioritization has been, and still is, one of the most difficult tasks in today's strongly chaotic and unpredictable development environments. Prioritization is also one of the most neglected research topics. To the knowledge of the authors of this paper, there is scarce literature about requirements prioritization [13]–[15].

This paper suggests a framework for prioritizing system requirements. The framework is called *System Requirements Prioritization Framework* and is referred to as SRPF. It consists of eight components, each representing a specific angle of the prioritization process. These components are (1) *Input*, (2) *Stakeholders*, (3) *Prioritization Criteria*, (4) *Prioritization Methods*, (5) *Environment*, (6) *Resources,* (7) *Priority Scales, and* (8) *Urgency Levels*. Our goal is to create an effective support aiding companies in making structured and objective prioritization decisions.

SRPF is a framework composed of a basic structure of the constituents required for making requirements priorities. As a framework, it is open for various kinds of adaptations and additions to the companies' own development milieus. Its mission is to support companies in their objective prioritization work within system development.

Altogether, seventeen companies were involved in this study. For confidentiality reasons, we do not disclose their names. Instead, we use fictitious names, whenever necessary.

The remainder of this paper is as follows. Section II describes the research method taken in this study. Section III presents the results of the exploration phase. Section IV describes the framework whereas Section V reports on the results of the framework evaluation. Finally, Section VI makes final remarks and suggestions for future research.

II. Research method

Our research method was a typical qualitative and inductive study. It was carried out in four phases. These are (1) *Exploration phase*, (2) *Design phase*, (3) *Evaluation phase*, and (4) *Fine-Tuning* phase.

Since limited research has been done within the area, we educed as much knowledge as possible about the current prioritization practice within the literature and industry. This phase was quite long and extensive. The results of the *Exploration* phase gave us enough feedback for designing the preliminary version of SRPF in the *Design* phase. The preliminary version was then evaluated in the *Evaluation* phase within the industry. Here, we used six evaluation criteria for assessing the relevance and usefulness of the SRPF. Finally, using the results of the *Evaluation* phase, we fine-tuned our framework and created its new improved version. Below, due to space restrictions, we only describe the *Exploration Phase* and the evaluation criteria.

A. Exploration Phase

We started our study with a thorough investigation of the domain of requirements prioritization. Here, we first made an extensive literature study using the following keywords: *prioritization, system requirements, decision making, customer benefit,* and *corporate value*. Unfortunately, this study resulted in very few sources on which we could base our research. Therefore, we continued to educe knowledge about the prioritization domain by studying the industrial practice. We did our exploration in four consecutive steps via interviews and surveys using the exploration questionnaires as presented in Figure 1.

First, we conducted a case study within *Company 1* using *Questionnaires 1* and *2*. Using *Questionnaire 1*, we interviewed two system development managers. Here, we focused on finding out (1) what the company's prioritization model looked like, (2) what criteria were considered in the prioritization work and how they were weighed, (3) whether the company used any predefined priority scales [16][17], (4) what the communication process looked like, and finally, (5) how they defined corporate value.

Using *Questionnaire 2*, we interviewed three business area managers, one technical manager, and the CEO. Here, we inquired about (1) what information was used when communicating on project prioritizations, (2) whether the business and system managers used any predefined priority scales, (3) whether any supporting tools were used, (4) what criteria were considered when prioritizing, (5) whether and how the company paid heed to the strategic goals, and finally, (6) how the company defined the expected value or benefit of attending to the prioritized system requirements.

To further broaden our insights into the prioritization work, we interviewed another company, *Company 2* using *Questionnaire 3*. By studying the questions, our reader may see that in addition to some questions that had already been asked in *Questionnaires 1* and *2*, we inquired about the project prioritization models and processes, their designs and uses, and the contexts of prioritization. Finally, we asked our interviewees to point out which of the criteria were the most important ones when doing prioritizations.

To assure that we have understood the requirements prioritization domain, we conducted a survey on the web using *Questionnaire 4*. Fifteen respondents were involved in this survey. Here, we first found out whether our respondents had the right competence for answering our questions. We also investigated what their companies and development departments looked like. Regarding prioritization work, we focused on finding out (1) what the respondents' work model looked like, (2) how they made priorities, (3) whether they used any criteria and methods for determining priorities, (4) whether any business strategic priorities were followed, and finally, (5) we inquired about the roles and responsibilities.

B. Evaluation Criteria

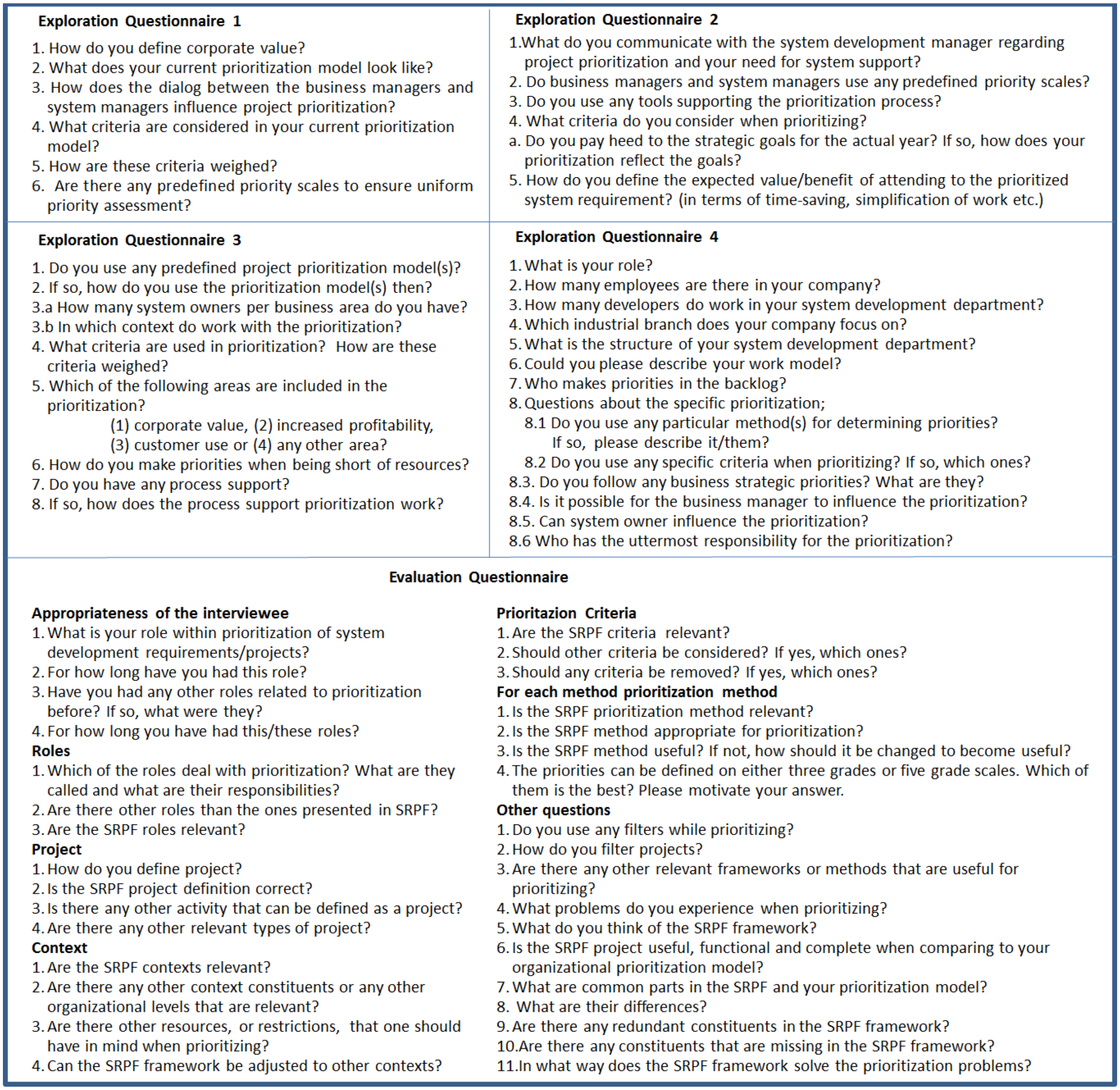
When evaluating SRPF, we used six evaluation criteria. These were:

1. *Appropriateness of the interviewees*: Using *Questions 1–4* in the *Evaluation Questionnaire* in Figure 1, we inquired whether our interviewees were suitable for evaluating our framework.
2. *Roles*: With *Questions 1–3*, we tried to find out what roles were involved in the prioritization work. We also asked our interviewees to express their opinions on the relevancy of the roles as suggested in our framework.
3. *Project*: One of the terms used in our framework is "project". To avoid misunderstanding with respect to its meaning, we inquired how the interviewees defined prioritization projects and whether our definition agreed with theirs. Here, we used *Questions 1–4*.
4. *Context*: Using *Questions 1–4*, we inquired whether the framework's context was (1) relevant, (2) whether anything was missing, (3) whether there were any resources or restrictions one should consider when prioritizing, and finally, (4) whether our framework could be adjusted to other contexts.
5. *Prioritization Criteria:* Using *Questions 1–3*, we wished to find out whether the framework's criteria were relevant or redundant, and whether any other criteria were missing.
6. *Prioritization Methods*: With *Questions 1–4*, we inquired whether the framework's prioritization methods were relevant, appropriate and useful for the interviewees' respective organizations. We also wished to hear their opinions about the number of grading levels to be used.

Other questions: Using *Questions 1–11*, we wished to hear the opinions of the interviewees about SRPF, how much it differed from their prioritization methods, and whether our framework missed any important components.

III. Exploration Results

In this section, we describe the results of the four *Exploration phases*. Due to space restrictions, we only provide additional feedback that got elicited during each consecutive phase.



Exploration Questionnaire 1

1. How do you define corporate value?
2. What does your current prioritization model look like?
3. How does the dialog between the business managers and system managers influence project prioritization?
4. What criteria are considered in your current prioritization model?
5. How are these criteria weighed?
6. Are there any predefined priority scales to ensure uniform priority assessment?

Exploration Questionnaire 2

1.What do you communicate with the system development manager regarding project prioritization and your need for system support?
2. Do business managers and system managers use any predefined priority scales?
3. Do you use any tools supporting the prioritization process?
4. What criteria do you consider when prioritizing?
a. Do you pay heed to the strategic goals for the actual year? If so, how does your prioritization reflect the goals?
5. How do you define the expected value/benefit of attending to the prioritized system requirement? (in terms of time-saving, simplification of work etc.)

Exploration Questionnaire 3

1. Do you use any predefined project prioritization model(s)?
2. If so, how do you use the prioritization model(s) then?
3.a How many system owners per business area do you have?
3.b In which context do work with the prioritization?
4. What criteria are used in prioritization? How are these criteria weighed?
5. Which of the following areas are included in the prioritization?
(1) corporate value, (2) increased profitability, (3) customer use or (4) any other area?
6. How do you make priorities when being short of resources?
7. Do you have any process support?
8. If so, how does the process support prioritization work?

Exploration Questionnaire 4

1. What is your role?
2. How many employees are there in your company?
3. How many developers do work in your system development department?
4. Which industrial branch does your company focus on?
5. What is the structure of your system development department?
6. Could you please describe your work model?
7. Who makes priorities in the backlog?
8. Questions about the specific prioritization;
8.1 Do you use any particular method(s) for determining priorities? If so, please describe it/them?
8.2 Do you use any specific criteria when prioritizing? If so, which ones?
8.3. Do you follow any business strategic priorities? What are they?
8.4. Is it possible for the business manager to influence the prioritization?
8.5. Can system owner influence the prioritization?
8.6 Who has the uttermost responsibility for the prioritization?

Evaluation Questionnaire

Appropriateness of the interviewee
1. What is your role within prioritization of system development requirements/projects?
2. For how long have you had this role?
3. Have you had any other roles related to prioritization before? If so, what were they?
4. For how long you have had this/these roles?

Roles
1. Which of the roles deal with prioritization? What are they called and what are their responsibilities?
2. Are there other roles than the ones presented in SRPF?
3. Are the SRPF roles relevant?

Project
1. How do you define project?
2. Is the SRPF project definition correct?
3. Is there any other activity that can be defined as a project?
4. Are there any other relevant types of project?

Context
1. Are the SRPF contexts relevant?
2. Are there any other context constituents or any other organizational levels that are relevant?
3. Are there other resources, or restrictions, that one should have in mind when prioritizing?
4. Can the SRPF framework be adjusted to other contexts?

Prioritazion Criteria
1. Are the SRPF criteria relevant?
2. Should other criteria be considered? If yes, which ones?
3. Should any criteria be removed? If yes, which ones?

For each method prioritization method
1. Is the SRPF prioritization method relevant?
2. Is the SRPF method appropriate for prioritization?
3. Is the SRPF method useful? If not, how should it be changed to become useful?
4. The priorities can be defined on either three grades or five grade scales. Which of them is the best? Please motivate your answer.

Other questions
1. Do you use any filters while prioritizing?
2. How do you filter projects?
3. Are there any other relevant frameworks or methods that are useful for prioritizing?
4. What problems do you experience when prioritizing?
5. What do you think of the SRPF framework?
6. Is the SRPF project useful, functional and complete when comparing to your organizational prioritization model?
7. What are common parts in the SRPF and your prioritization model?
8. What are their differences?
9. Are there any redundant constituents in the SRPF framework?
10. Are there any constituents that are missing in the SRPF framework?
11. In what way does the SRPF framework solve the prioritization problems?

Figure 1 Questionnaires used in our study

A. *Results of Exploration Step 1*

Exploration Step 1 in *Company 1* revealed that none of the two interviewees used any predefined prioritization model. They neither used any predefined criteria nor any priority scales. All project prioritization was conducted in a merely ad hoc manner and varied among the two individuals being interviewed. Despite this, we received some insight into the company's prioritization work.

Typical evidence for lack of common prioritization method is their individual understanding of corporate value. *Interviewee 1* defines it as *profit, satisfied employees* and *satisfied customers* whereas *Interviewee 2* excludes customer satisfaction. This already automatically provides a basis for non-uniformity of their prioritization efforts.

Prioritization in *Company 1* is conducted on three levels: (1) system level implying development of a new system, (2) functionality level implying major change, and (3) minor change level. The predefined budget always constrains all prioritizations.

There is a very poor communication on setting priorities between business and system development managers. Business managers always set priorities first. System development managers then either accept or change them.

Very seldom do they provide feedback on the changes to the business managers.

The fact that *Company 1* does not have any predefined prioritization criteria implies that *Interviewee 2* follows his own subjective prioritization and project effort estimations. It happens that his prioritization choices are not always well motivated. *Interviewee 1*, on the other hand, uses the Information Technology Infrastructure Library´s four priority levels [18] subdivided into 99 sublevels.

B. *Exploration Step 2*

Just because *Exploration 1* did not provide us with much feedback, we once again interviewed individuals in *Company 1*, this time however, using *Questionnaire 2*. We interviewed five people and one of them *(Interviewee 1* in *Exploration 1)* was interviewed anew for confirming that we had understood him right.

The results of *Exploration 2* confirmed the results of *Exploration 1*. In addition, we found out that *Company 1* was strongly controlled by customers, not always in an orderly manner. The customers "*shouting the loudest get their wills easily satisfied*". This puts system development management in a very difficult position when trying to balance customer satisfaction and company's strategic goals. In some cases, the prioritization requests escalate to high-level management.

Regarding communication on the already prioritized projects, prioritizations and re-prioritizations of their individual requirements are being made on almost a daily basis. Here, project teams know the best how to prioritize them in the most effective manner.

None of the five interviewees uses any prioritization tools. Only two out of five interviewees use one common criterion when prioritizing, which is *company strategy*. This criterion is only used if conflicts arise. Finally, only one interviewee was able to state the expected value/benefit of attending to the prioritized system requirements. The value concerned savings in time and money.

C. *Exploration Step 3*

The results of *Exploration Step 3* reveal that even *Company 2* does not follow any predefined prioritization models. Our interviewees use their own individual models instead. The models are simple. They imply either regular meetings with follow ups or budget-controlled models. In any of the cases, the models include a strong interplay among many roles.

Irrespective of the models, all development in *Company 2* follows the *Phase-Gate* process model [19], which is the context of all its prioritization efforts. Just as in *Company 1*, big focus is being put on more important customers. In addition, the company has defined severity levels for each project to be prioritized. The priority is then defined based on severity value and the revenue to be gained. In cases, however, when several projects compete, the criterion that wins is the "*customer bigness*". When short of resources, the projects that hurt the least get the lowest priority.

When making priority decisions, *Company 2* regards areas such as (1) *corporate value,* (2) *increased profit,* and (3) *customer use*. Especially important is the customer use of the product. Big effort is being made to understand how the product is being used for the purpose of understanding the needs of the customers and the value of customer demands, and for making correct prioritizations.

D. *Exploration Step 4*

Altogether, fifteen people were involved in the survey in *Exploration Step 4*. They had the following roles: (1) six system developers, (2) three project leaders, (3) two managers, (4) two product owners, (5) one Unix administrator, and (6) one undefined. Three respondents came from very large companies with more than 500 employees, another three from large companies with more than 100 employees, and the remaining ones came from companies having more or less ten employees. The industries involved were banking and insurance, e-commerce, public services and various branches, such as general tech, gaming, farming, and amusement parks.

All except for one respondent could identify their work models as agile and lean related. In their respective companies, product prioritization is conducted by product owners (8 responses), project leaders (5 responses), and project teams (1 response).

Only two respondents could claim that they had a prioritization method. The prioritization criteria, as mentioned by the respondents, concerned ROI, customer impact, technical debt, emergency status, and the cost.

Regarding the roles responsible for prioritization, the following was provided: (1) business manager (2 responses) and product/system owners (10 responses) could influence the prioritization process, and finally, product/system owners (7 responses) and CTO or CEO (2 responses) had the uttermost responsibility for the prioritization.

E. *Exploration Phase in Summary*

The exploration phase taught us that prioritization was very complex and included several aspects. These are method, roles, context, prioritization criteria and resources.

Although many companies and roles are involved in prioritization, there are still companies who do not have a proper prioritization method. Lack of the method and lack of mutual criteria steering the prioritization effort imply great risk for subjective prioritization that may not always be aligned with the strategic goals.

IV. SRPF FRAMEWORK

In this section, we describe SRPF. We first provide an overall description of all its components. We then describe in detail the SRPF prioritization methods.

A. *Components in SRPF*

The preliminary SPRF consists of eight parts. As illustrated in Figure 2, these are (1) *Input,* (2) *Stakeholders,* (3) *Prioritization Criteria,* (4) *Prioritization Methods,* (5) *Environment*, (6) Resources, (7) *Priority Scales*, and (8) *Urgency Levels*. Below, we briefly describe the parts.

The SRPF *Input* stands for projects to be prioritized. Here, we include all projects that have not been prioritized and projects that need to be reprioritized for various reasons.

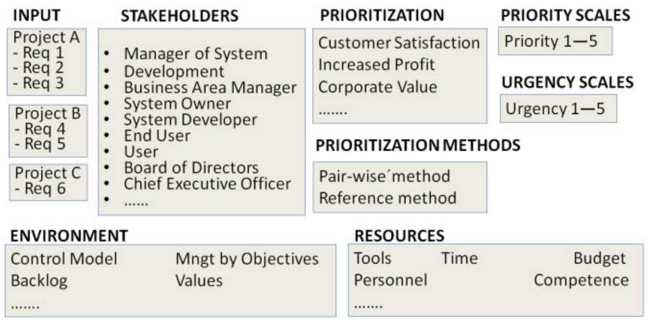


Figure 2. Outline of SRPF. Dots imply that the SRPF users are free to extend the framework parts with their own suggestions

The SRPF definition of a project is a set of requirements to be attended to. A set may consist of at least one requirement. Regarding the term requirement, SRPF defines it as a description of a need to get attended to. This need may either concern an implementation of a new functionality, minor improvement, a corrective or preventive change, and the like.

The SRPF *Stakeholders* correspond to a role or a group of roles that has interest or concern in a prioritization process. Stakeholders can affect or be affected by the prioritization process. Some examples are system managers, product owners, project managers, acquirers, business area managers, users, customers, and the like.

An important component in SRPF are *Prioritization Criteria*. To ensure achievement of strategic goals, the organizations must define criteria that help them identify the most urgent projects at a given point in time. SRPF leaves it open to its users to define their own criteria which they feel are the most suitable ones for their business operation and prioritization contexts. It, however, lists three criteria that are common to most of the organizations. These are (1) *corporate value*, (2) *increased profit,* and (3) *customer satisfaction*. To assure full commitment, the criteria should be well motivated and communicated to all the parties involved. It is only in this way, companies may assure the effectiveness of their prioritization efforts.

Prioritization Methods are the core of our framework. SRPF suggests two methods. These are (1) pair-wise and (2) reference methods. The pair-wise method compares projects pair-wise with all other projects meanwhile the reference method, which is a simplified version of the pair-wise method, compares all projects with one neutrally chosen reference project. The two methods are described in Section IV.A. As marked with dots in Figure 2, the SRPF users are free to extend the framework with their own methods.

A very important SRPF part is *Context*. It shapes the overall prioritization process. Context is very often neglected in many prioritization efforts or, for some reason, the stakeholders involved do not always attempt to explicitly communicate it.

Lack of a common understanding of a context may lead to many problems. For instance, information communicated by one stakeholder having his/her subjective understanding of a context may be easily misunderstood by some other stakeholder having his/her own subjective understanding. Hence, context must be explicitly identified by the company. Context describes what the organization looks like; where in the organization are decisions made, who has the authority to do prioritizations, whether there is a steering model, corporate values, backlog, and other important issues that are relevant for a specific organization. SRPF suggests the following contextual constituents:

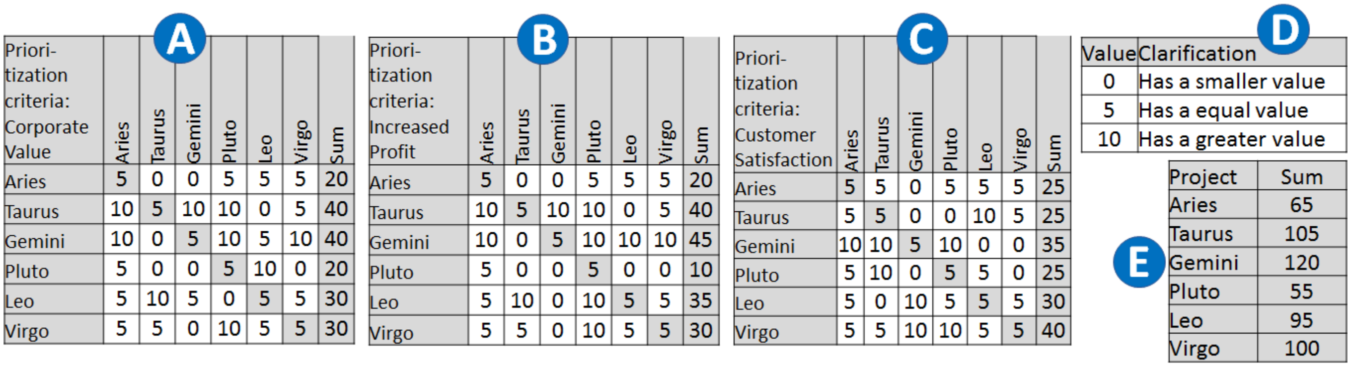
- *Control model* describing the decision making authorities and points in time when decisions are to be made.
- *Management by Objectives* aiming at decision making directed towards specific goals.
- *Backlog* listing all pending projects.
- *Values* referring to the organizational values to be considered in prioritization.

Finally, the SRPF *Resources* are sources of supply and support that are needed for conducting prioritization. Here, we include the following:

- *Tools* assisting the prioritization efforts, such as software, hardware and the like.
- *Personnel* referring to the individuals performing both development and prioritization.
- *Time* assigned to both prioritization and implementation of the pending projects.
- *Competence* standing for the collected organizational capability of attending to the prioritized projects.
- *Budget* referring to the amount of financial resources available for attending to the prioritization.

The SRPF Priority defines the urgency level for corrective action. It should be stated both by the customer and developer. The priority value as stated by the customers indicates how important it is for the customers to get the requirements attended to. Different customers, however, have different needs, different environments, and different safety and security requirements. The development organization cannot consider them all. They must define their own priority values that provide a basis for making their own priorities among the pending projects.

The SRPF Severity measures the effect of the disruption caused by a problem. Severity influences priority. For instance, a problem that could represent danger to human life or could cause failure of a company is most severe, and hence, its resolution should have the highest urgency. The value of priority, however, does not always influence the value of severity. High priority can be assigned even to less severe problems. Less severe yet frequent problems can be very costly and may lead to lowered credibility of the software organization [20][21]. Hence, they should be prioritized.



Prioritization criteria: Corporate Value	Aries	Taurus	Gemini	Pluto	Leo	Virgo	Sum
Aries	5	0	0	5	5	5	20
Taurus	10	5	10	10	0	5	40
Gemini	10	0	5	10	5	10	40
Pluto	5	0	0	5	10	0	20
Leo	5	10	5	0	5	5	30
Virgo	5	5	0	10	5	5	30

Prioritization criteria: Increased Profit	Aries	Taurus	Gemini	Pluto	Leo	Virgo	Sum
Aries	5	0	0	5	5	5	20
Taurus	10	5	10	10	0	5	40
Gemini	10	0	5	10	10	10	45
Pluto	5	0	0	5	0	0	10
Leo	5	10	0	10	5	5	35
Virgo	5	5	0	10	5	5	30

Prioritization criteria: Customer Satisfaction	Aries	Taurus	Gemini	Pluto	Leo	Virgo	Sum
Aries	5	5	0	5	5	5	25
Taurus	5	5	0	0	10	5	25
Gemini	10	10	5	10	0	0	35
Pluto	5	10	0	5	5	0	25
Leo	5	0	10	5	5	5	30
Virgo	5	5	10	10	5	5	40

Value	Clarification
0	Has a smaller value
5	Has a equal value
10	Has a greater value

Project	Sum
Aries	65
Taurus	105
Gemini	120
Pluto	55
Leo	95
Virgo	100

Figure 3. Illustrating pairwise method

B. *Prioritization Methods*

The pair-wise and reference methods are almost similar in their designs. In this section, we first describe the pair-wise method. We then describe the reference method. Finally, we discuss their similarities and differences.

Pair-wise method is a well-known process of comparing various entities in pairs with the purpose of deciding which of them is better. It has been used in various domains such as education, engineering, energy and water resources, management, and environmental applications [22]–[24]. It has also been used within requirements engineering, however, mainly from the cost perspective. In contrast, the SRPF pair-wise method considers all kinds of criteria to be used in comparison, not only the cost.

The choice of the comparison criteria is to be decided by the company using SRPF. Below, we provide an example based on three values: (1) *corporate value*, (2) *profit*, and (3) *customer satisfaction*. Projects get assessed pairwise using only three numerical values. These are (1) zero standing for "*has lower value*", (2) five standing for "*no difference*" and (3) ten standing for "*has higher value*".

Subfigures A-C of Figure 3 show pair-wise evaluation of the projects with respect to corporate value, profit and customer satisfaction respectively using the values specified in *Subfigure D of Figure 3*. These values are then summed up for all the comparison criteria. As shown in *Subfigure E of Figure 3*, the project called *Gemini* gained the highest score which is 120 points. This project should get the highest priority. If the criteria have different mutually important weights, typically 1 to 3, then each score can be multiplied with the weight before summarization. Then the criterion with the highest weight will be more important in the final sum.

Reference method is also a comparison method. To the knowledge of the authors of this paper, no one has used it in within requirements engineering. SRPF suggests that all competing projects are compared to only one neutral project. As illustrated in the third column in Subfigure A of Figure 4, this neutral project is called *reference project*. All other projects that are to be compared are given a value depending on how similar or different they are to/from the reference project.

Five values are assigned to the competing projects. These are (1) '+' better, (2) '++' much better, (3) 'S' similar, (4) '-' lower, (5) '- -' much lower. When a project is better than the reference project, a plus (+) is assigned to the project. If the difference is judged enormously bigger, then two plusses (++) are assigned. If the project is equal, then an 'S' is given standing for the same value. If the project is worse or much worse, then a minus (-) or two minuses (- -) are assigned.

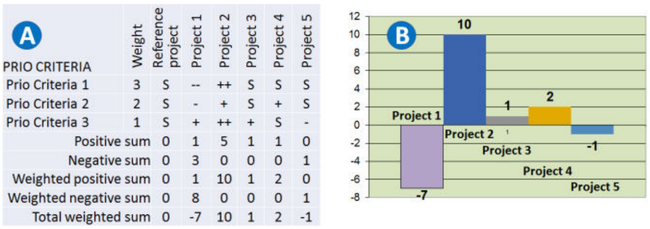
Just as in the pair-wise method, the reference method may use different criteria. Their score may then be summed giving total scores of the projects. An example comparison is illustrated in Subfigure B of Figure 4.

Adding a weight to each criterion is possible. When all projects are compared, a summarization of the respective grading values are made adding the weight and then a total sorting of the points gives the priority value.

C. *Comparing the two methods*

When comparing the methods, reference method is like the pair-wise method. There are however some differences. These are:

- Pair-wise method uses several matrices whereas reference method only uses one for comparison regardless of how many criteria are being compared.
- Pair-wise method does not use any reference project. If the reference project is not a good choice, then the whole prioritization effort is at risk.
- Pair-wise method compares all projects with one another whereas reference method compares all other projects to the reference project. This implies that in the reference method, the number of comparisons is linear to



PRIO CRITERIA	Weight	Reference project	Project 1	Project 2	Project 3	Project 4	Project 5
Prio Criteria 1	3	S	--	++	S	S	S
Prio Criteria 2	2	S	-	+	S	+	S
Prio Criteria 3	1	S	+	++	+	S	-
Positive sum		0	1	5	1	1	0
Negative sum		0	3	0	0	0	1
Weighted positive sum		0	1	10	1	2	0
Weighted negative sum		0	8	0	0	0	1
Total weighted sum		0	-7	10	1	2	-1

Figure 4. Illustrating reference method ((1) '+' better, (2) '++' much better, (3) 'S' similar, (4) '-' lower, (5) '- -' much lower)

the number of projects, while it is quadratic in the pair-wise method.

When using SRPF, companies are free to choose any of the two methods, or use the two methods, or add their own methods. SRPF suggests that reference method be used first for quick filtering of less important projects and then pair-wise method be used for more meticulous comparisons.

V. RESULTS OF THE EVALUATION INTERVIEW

In this section, we report on the results of the evaluation phase following the evaluation criteria as defined in Section II and the evaluation questionnaire.

All our seven interviewees were suitable for evaluating SRPF. Five of them have more than ten years and two of them have more than five years of experience of prioritizing within system development. All seven are managers involved in system development.

The roles that have been mentioned as active within prioritization were project leaders, system architects, product managers, product/system owners, line managers, technology managers, functional managers, quality managers, system development managers, business area managers, users, boards of directors, CEOs, and developers.

All the interviewees agreed with the SRPF project definition. However, they all had their own variants. Important is to say that they pointed out that the SRPF definition missed the concept of goal, scope, time and resource constraints and quality requirements. Also, there have been suggestions for differentiating between projects as temporary created for fulfilling some specific goal and permanent dealing with continuous improvements and defect corrections.

All seven interviewees agree that SRPF's context is relevant. Five interviewees find the context very general and all seven interviewees agree that SRPF is useful in almost any prioritization effort. Three interviewees came with highly valuable suggestions. One of them claimed that the technological context was missing and another one suggested that the methodological context should include the identification of the development method used and the placement of projects within its phases. Certain projects could only be placed at the beginning of a development method whereas others could be placed anywhere within the lifecycle. Finally, the third interviewee mentioned the business context for considering business strategic goals.

All the interviewees agreed that the prioritization criteria in SRPF were relevant and that no criteria were redundant. They however pointed out new criteria that were highly relevant. These are *cultural value, politics, customer, business value, customer value, focus on new markets, risks, time,* and *competence*.

All the interviewees agreed that the SRPF prioritization methods (pair-wise and reference) were relevant, appropriate and useful and that they contributed to the objective prioritization. Neither of the interviewees found the methods to be redundant. Some comments were made regarding the reference method. Two interviewees were of the opinion that the method was difficult. The difficulty lied in the choice of a reference project. Regarding the used scale in the method, five interviewees would choose the five-grade scale as suggested by SRPF, one interviewee would rather use a four-grade scale whereas one interviewee claimed that six grades would be the best.

Regarding the answers to the batch of the remaining questions, the results are as follows. All seven interviewees do filter some projects before prioritization. One interviewee knows about a framework similar to SRPF as conducted within the industry.

When prioritizing, the interviewees encounter many problems. In addition to the problems as listed in Section I, the problems that have been mentioned are (1) difficulties in down-prioritizing projects, (2) agreeing on common criteria to base prioritizations on, (3) inability to prioritize all projects in the backlog, and finally, (4) lack of time for making prioritizations.

When being asked about the overall impression of SRPF, all seven interviewees found it interesting and good. Five of them claimed that they did work in a similar way as suggested in the framework. None of the interviewees found anything redundant in SRPF.

VI. FINAL REMARKS

This paper suggests *System Requirements Prioritization Framework*. Our goal was to create an effective support aiding companies in making structured and objective decisions when prioritizing requirements.

As a framework, SRPF is open for various kinds of adaptations and additions to the companies' own development milieus. Its mission is to support companies in their objective prioritization work within system development. Right now, it only outlines the most important prioritization components. Hence, we suggest it be an initial version for both the industry and the academia. We also propose to further evolve it.

Even if SRPF is in its initial phase, it is already more advanced than the existing prioritization methods [8][25]. These methods are very simple in their designs. They mainly state that requirements should be prioritized and, at its most, suggest priority grades. Hence, they are incomparable to SRPF

During the evaluation phase, we discovered that our definition of a project was too broad. For this reason, we broaden it with project goal, scope, and constraints. The new definition of a project *is a set of requirements to be attended to that has a clearly specified goal and scope, and that is bounded by a set of clearly specified constraints. A set consists of at least one requirement.*

When evolving the framework, we suggest the following issues to be further researched on. Priorities and urgencies are complex issues, and little has been done to identify scales and/or variances of scales to reflect their levels and relationships. We suggest that more effort be put into ways of defining priorities and urgencies.

Regarding the context, we must admit that we have forgotten one very important criterion concerning the ethics. Ethics has not been explicitly mentioned during the evaluation interviews. However, by studying the interview results, we understood that it was implicitly hidden behind many answers. We suggest that this criterion be considered and researched on in the context of prioritization. Also, methodological and technological contexts should be considered in the framework.

Considering the roles, we have noticed that prioritization involved a multitude of roles. SRPF has only identified a subset of them. We believe that more research should be conducted on identifying the roles involved and their responsibility portfolios so that each role may contribute to the prioritization efforts in its best possible way.

So far, SRPF has been explored and evaluated via interviews and surveys. Even if industrial professionals accepted the framework, it would be good if the framework were used in an industrial context. Therefore, we warmly welcome anybody to use SRPF and provide feedback from its real-life usage.

REFERENCES

[1] A. Purnus and C.-N. Bodea, ”Project Prioritization and Portfolio Performance Measurement in Project Oriented Organizations,” *Procedia - Social and Behavioral Sciences,* vol. 119, pp. 339-348, 2014.

[2] M. Alexander, ”CIO from IDG,” IDG Communications, Inc., 23 11 2015. [Online]. Available from: https://www.cio.com/article/3007575/project-management/6-proven-strategies-for-evaluating-and-prioritizing-it-projects.html 2018.3.16

[3] M. Schedlbauer, ”Corporate Education Group,” 2017. [Online]. Available from: http://www.corpedgroup.com/resources/ba/ReqsPrioritization.asp 2018.3.16

[4] Sweden, ” World-class foreign affairs: final report”, The governments official investigations, ”Utrikesförvaltning i världsklass : slutbetänkande,” *Statens offentliga utredningar SOU 2011:21*, Stockholm, Offentliga Förlaget : Publit, 2011, pp. 457-459.

[5] S. R. Covey, ”Goodreads,” Goodreads Inc, 2017. [Online]. Available from: https://www.goodreads.com/quotes/706652-the-key-is-not-to-prioritize-what-s-on-your-schedule 2018.3.16

[6] O. Maassen and J. Sonnevelt, "Kanban at an Insurance Company (Are You Sure?)," in *Agile Processes*, Trondheim, Norway, 2010.

[7] B. Nuseibeh and S. Easterbrook, "Requirements engineering: a roadmap," in Proceedings of the Conference on The Future of Software Engineering, Limerick, Ireland, 2000.

[8] E. T. Game, P. Kareiva, and H. P. Possingham, ”Six Common Mistakes in Conservation Priority Setting,” *Conservation Biology,* vol. 27, nr 3, p. 480–485, 2013.

[9] I. Sommerville and P. Sawyer, ”Viewpoints: principles, problems and a practical approach to requirements engineering,” *Annals of Software Engineering,* vol. 3, nr 1, pp. 101-130, 1997.

[10] C. C. Chou, ”A fuzzy MCDM method for solving marine transshipment container port selection problems,” *Applied Mathematics and Computation,* vol. 186, nr 1, pp. 435-444, 2007.

[11] S. L. Chang, R. C. Wang, and S. Y. Wang, ”Applying fuzzy linguistic quantifier to select supply chain partners at different phases of product life cycle,” *International Journal of Production Economics,* vol. 100, nr 2, pp. 348-359, 2006.

[12] X. D. Luo, J. H. M. Lee, H. F. Leung, and N. R. Jennings, ”Prioritised fuzzy constraint satisfaction problems: axioms, instantiation and validation,” *Fuzzy Sets and Systems,* vol. 136, nr 2, pp. 151-188, 2003.

[13] J. Vähänitty and K. T. Rautiainen, ”Towards a conceptual framework and tool support for linking long-term product and business planning with agile software development,” *Association for computing Machinery,* pp. 25-28, 2008.

[14] J. Karlsson, C. Wohlin, and B. Regnell, ”An evaluation of methods for prioritizing software requirements,” *Information and Software Technology,* vol. 39, no 14-15, pp. 939-947, 1998.

[15] M. Kajko-Mattsson, ”Corrective maintenance maturity model : problem management,” Stockholm, 2001.

[16] M. Kajko-Mattsson, "Conceptual model of software maintenance," Forging New Links, Proceedings of the 1998 International Conference on Software Engineering (ICSE 98) Apr. 1998, pp. 422-425, ISBN:0-8186-8368-6

[17] M. Kajko-Mattsson, ” Experience Paper: Maintenance at ABB (I): software problem administration processes,” 1999 International Conference on Software Maintenance (ICSM 1999), Sep. 1999, pp. 167-176, ISSN: 1063-6773, ISBN: 0-7695-0016-1

[20] B. Beizer, Software System Testing and Quality Assurance, Van Nostrand Reinhold Company, 1984.

[21] C. Kaner, J. Falk, and H. Q. Nguyen, Testing Computer Software, John Wiley & Sons, Inc., 1999.

[22] L. Dias, M. Pirlot, P. Meyer, R. Bisdorff, and V. Mousseau, International handbooks on information systems. Evaluation and decision models with multiple criteria, Berlin: Springer, 2015.

[23] J. Wallenius, J. Dyer, P. Fishburn, R. Steuer, S. Zionts, and K. Deb, ”Multiple criteria decision making, multiattribute utility theory: recent accomplishments and what lies ahead,” *Management Science,* vol. 54, nr 7, pp. 1336-1349, 2008.

[24] C. Zopounidis and P. Pardalos, Handbook of multicriteria analysis, Berlin: Springer, 2010.

[25] V. Ahl, ”An Experimental Comparison of Five Prioritization Methods,” Blekinge Sweden, 2005.