

# **Rigorous Support for Flexible Planning of Product Releases - A Stakeholder-Centric Approach and its Initial Evaluation**

## **Notice of Originality**

I declare that this paper is my own work and that information derived from published or unpublished work of others has been acknowledged in the text and has been explicitly referred to in the list of references. All citations are in the text between quotation marks (“ ”). I am fully aware that violation of these rules can have severe consequences for my study at Utrecht University.

Name: Mario Narwan  
Signed:

Date:  
Place:

## Introduction

Every time a product is released the product manager has to make sure that the limited resources, assigned to the project, are balanced out between the needs of the market and those of the stakeholders. If the planning process is not addressed correctly, important decisions may be based on intuition, which decreases the reliability of the given products. The main goal of this method is to include the stakeholders in the planning process of product releasing. This way stakeholders will have control or influence on the different features of the product. However, when many stakeholders are involved a face-to-face communication is not feasible, therefore there is a need for a decision support system, which will aid the stakeholders in communicating as well as deciding on the features of the product (Heikkilä, Jadallah, Rautiainen, Ruhe, 2010). The authors of this paper Heikkilä, Jadallah, Rautiainen and Ruhe created a method to address this very issue called SCERP.

The proposed SCERP method identifies the key features of the to-be-released product. In this case the definition of feature is: a product feature as a set of logically related requirements that provide a capability to the user and enable the satisfaction of business objectives (Wiegers, 2003). At the very beginning a group of stakeholders (usually the most important ones) is chosen in order to assess the different features of the system. Each of the stakeholders is given a certain vote weight for the ranking process and all the features are ranked on a nine-point scale. "Possible criteria for prioritization are overall business value, urgency (time dependency), dissatisfaction if feature is not included in a release, risk (using an inverted scale), frequency of use" (Heikkilä et. al, 2010, p. 3). Furthermore, the features are divided into several release plans, each having a different combination of the given features. The aim of each plan is to implement the most important features of the products, while still being within the budget of the project. It is important to take into account that different features have different costs, so the perfect balance has to be found. As a final stage of the process the different plans are again assessed by the stakeholders on a nine-point scale. All of the ranking and assessment is done using an integrated tool called ReleasePlanner a web-based decision support system. The tool enables different stakeholders from different locations to be involved in the decision making process. All of the release features and the next resulting plans are ranked using this tool. The main deliverable of this method are the different alternatives for the final release plan, which will include all the features picked by the stakeholders. To sum it up the main goal of this method is to include the stakeholders in the decision making process for next release of a certain product. As well as to support the judgment of the different features, which are to be included in the next release.

The authors of this paper all come from two universities. Dr. Günther Ruhe and Anas Jadallah are both from the University of Calgary and Kristian Rautiainen together with Ville Heikkilä are from the Helsinki University of technology. Dr. Ruhe is a researcher and his main field of interest is software engineering and release planning (Ruhe, 2010). Jadallah is a former university student and now works as a Software Engineer at Sandvine inc (Jadallah, 2011). Kristian Rautiainen works as a researcher and his main research focus is software product development and planning (Rautiainen, 2010). Another one of the contributors is Ville T. Heikkilä a former Helsinki university student, who now works in Aalto University, School of Science and his specialization is software development methodologies (Heikkilä et. al, 2011).

## Example

When a new product is to be launched the company has to decide on what kind of features and functionality will the product have. The aim of this method is to help in the decision making process by

including the stakeholders. Therefore, a good cooperation between the stakeholders, project manager and developers has to exist for the SCERP method to be effective.  
The whole SCERP process is divided into five steps.

### Step 1: Selection of critical stakeholders and pre-selection of candidate features

The main purpose of this method is to include the stakeholders in the decision making. Therefore, the first step is to select the stakeholders that will be included (usually the stakeholders with the most shares are picked). Each stakeholder is also given a different weight for his vote in the following stages. Next is a pre-selection phase, where the requirements and customer needs are combined to create a list of features. Several features are pre-selected from this list to be included in the next release of the product. The result of this step is the pre-selected list of candidate features. An example list can be seen in Table 1.

ID	Feature	Description
<b>1</b>	<b>New security measures</b>	<b>New hash functions will be included</b>
2	3D GUIs	Every GUI will be made into 3D format
<b>3</b>	<b>Enhancing the user interfaces</b>	<b>New buttons, scroll menus and other functionality</b>
<b>4</b>	<b>Automatic backups</b>	<b>Automatic backup of all server data every hour</b>
5	Advanced Search	Advanced functions for search: by date, by author

**Table 1** - List of all features from which several will be pre-selected. (The bold ones are the selected ones)

### Step 2: Prioritization of features

As it can be seen in Table 1 only several features are selected to move to the next phase (the bold ones). Next the stakeholders assess the features according to the needs, priority, costs or the undertaken risk when not implementing the given feature. All of the assessment is done using the ReleasePlanner tool as I have described in the beginning. Each feature is judged on a nine-point scale (Likert 9 point scale). The scores are then added together, as each stakeholder's vote has different value, all of the data is calculated accordingly and the resulting values can be seen in Table 2, column "Mean". The result of this step is a table with all the ranked features, together with their scores from the stakeholders. An example result can be seen in Table 2.

ID	Feature	Mean	STDV*	Rank
1	New security measures	6.33	1.7	1
2	Enhancing the user interfaces	3.1	2.5	3
3	Automatic backups	5.5	2.1	2

**Table 2** - result of the step 2: table with all the features to be included in the next release, together with their scores.

### Step 3: Collective effort estimation

The next step is to calculate the effort estimation of each of the features. Jørgensen (2002) describes expert estimation as the most commonly used method for effort estimation, which is conducted by experts in the given field. Important parts of the estimation process are based on non-explicit, non-recoverable reasoning processes, i.e., "intuition". Therefore expert estimation is a combination of educated judgment, which is supported by historical data, process guidelines or other data and on the other hand the intuition or "gut feeling". The experts work in groups to critically discuss the effort needed

\*STDV - Standard deviation

for implementing a certain feature. Using this method the experts (usually the developers) make an effort estimation for each of the features picked in Step 2. An Example Result can be Seen in Table 3.

ID	Feature	Mean	STDV	Rank	Effort Est.
1	New security measures	6.33	1.7	1	<b>4 weeks</b>
2	Enhancing the user interfaces	3.1	2.5	3	<b>1 week</b>
3	Automatic backups	5.5	2.1	2	<b>5 days</b>

**Table 3** - Result of Step 3: table with all ranked features now including their effort estimation.

#### Step 4: Calculation of optimized release plan alternatives

One of the greatest advantages of this system is that it uses the optimization-based planning method, which creates not only one plan for the release, but rather several different plans. An automated algorithm creates five different alternative plans, each having a different combination of features. This way the stakeholders can decide not only which features will be included, but also what combination of features. These plans are later judged by the stakeholders. The result can be seen in Table 4 the left side. The different plans are marked as Os (on the top) and the different features are on the left side as Fs. In the center we can see the different combinations of features for each of the plans.

#### Step 5: Prioritization of alternative plans

The final step of SCERP is the stakeholders' assessment of the five alternatives created in Step 4. The assessment is carried out the same way as in Step 2 using the ReleasePlanner. The result of this Step is a table with all the different alternatives together with the scores from the stakeholders.

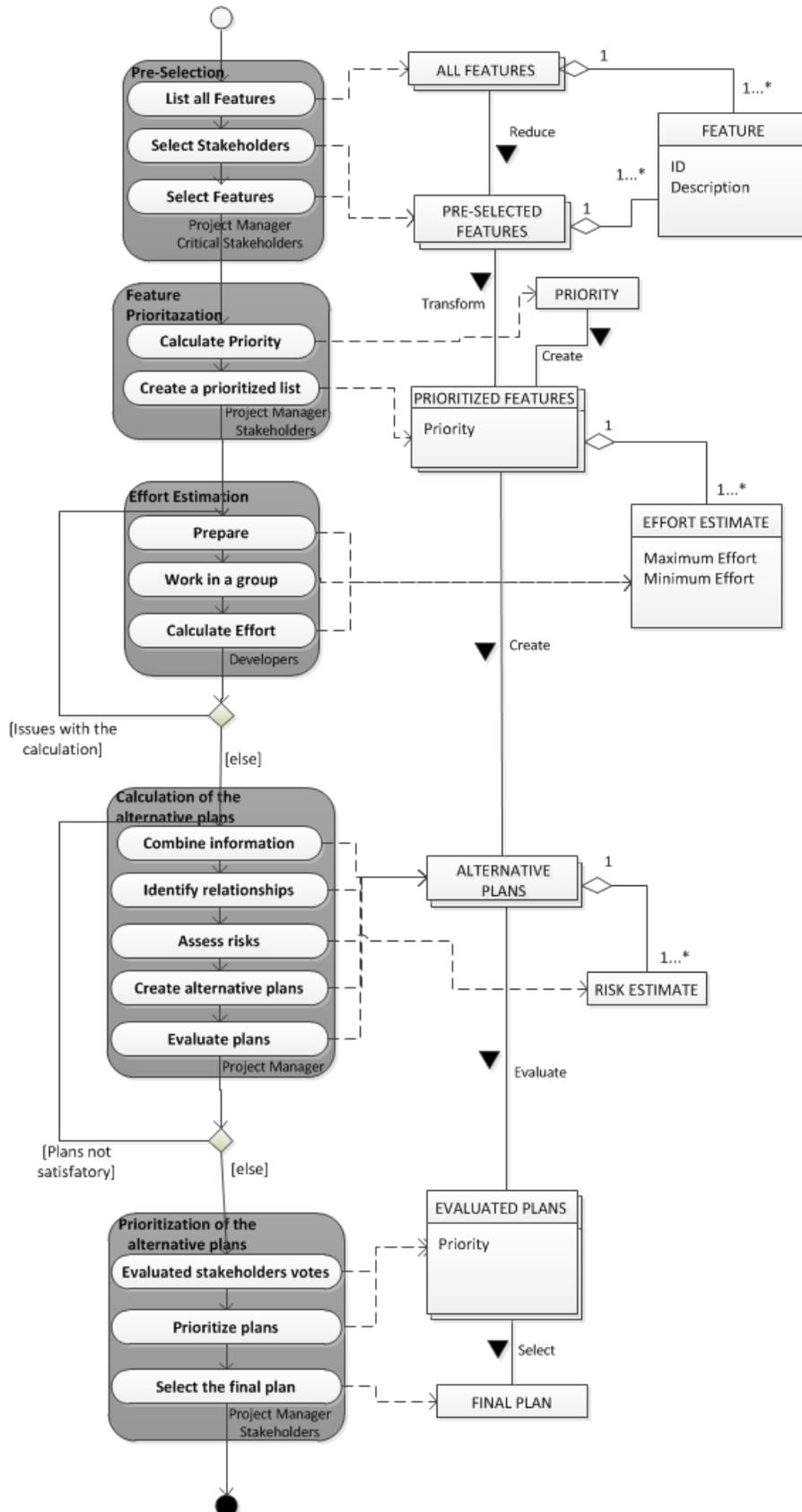
ID	O1	O2	O3	O4	O5	M	Release plan						
							Stakeholder	O1	O2	O3	O4	O5	M
F01	2	2	3	2	2	3	S01	4	3	9	7	1	2
F02	3	3	2	2	3	1	S02	7	5	5	7	7	4
F03	2	2	1	3	1	2	S03	2	2	8	1	7	3
F04	3	1	2	3	3	2	S04	6	9	9	5	8	8
F05	1	3	2	1	1	3	S05	3	6	3	3	5	3
F06	1	1	1	1	1	1	S06	6	5	4	7	5	2
F07	1	1	1	1	2	1	S07	4	4	5	7	4	4
F08	1	1	3	1	2	2	S08	3	2	5	1	6	4
F09	1	1	1	1	1	1	S09	5	4	7	3	6	8
F10	1	1	1	1	1	1	S10	2	3	9	1	5	8
Opt. (%)	100,0	98,9	98,6	97,1	96,8	93,4	Sum	42	43	64	42	54	46
							Median	4,0	4,0	6,0	4,0	5,5	4,0
							STDV	1,8	2,1	2,3	2,7	2,0	2,5

**Table 4** - Table with all the different alternative solutions for the next release

The final deliverable of this method is displayed in Table 4. The left is a table with all the alternative plans for the product release. At the top we have O1 - O5, which represents the alternatives and M which is a manual plan created just for control. On the left side there are all the features, which are to be included. The numbers 1, 2 and 3 inside the table mean 1 - the feature is to be released in the first release 2 - in the second release and 3 - the feature is postponed. At the very bottom of the left table are the percentages of how well is the specific plan fits the ideal situation (as it can be seen the first plan O1 has a 100% correlation, which means it is the most suitable plan). As for the table on the right we can see the different scores that the alternatives got from the stakeholders. This table is the final deliverable, for which the template can be seen in Appendix A. Here the most important variable is the Sum, which is the total score for each of the plans. Based on these scores the stakeholders make their final decision.

\*STDV - Standard deviation

## **PDD**



\*STDV - Standard deviation

## Activity Table

Activity	Sub-Activity	Description
Pre-Selection	List all features	The Project Manager gathers all the user requirements and needed FEATURES and creates a list of all the FEATURES to be included in the next release, resulting in a list of ALL FEATURES.
	Select Stakeholders	Several stakeholders are selected (usually the most important ones) for the latter evaluation process.
	Select Feature	This is the first pre-selection phase, where the project manager and stakeholders gain a deeper understanding of the features and requirements. Furthermore, several features are pre-selection for further evaluation. Resulting in the PRE-SELECTED FEATURES list.
Feature Prioritization	Calculate Priority	Each of the stakeholders uses the ReleasePlanner to judge the different features. The features are judged on a nine point scale (9 - being the most prior feature). As a result is the PRIORITY concept, which holds all the information about priorities.
	Create a prioritized list	The data calculated in priority calculation (the PRIORITY concept) are used to create a list of features together with their priority evaluation. Resulting in the PRIORITIZED FEATURES list.
Effort Estimation	Prepare	Each of the developers makes an individual preparation. They gather a better understanding of the requirements, features and the system as a whole
	Work in a group	After the preparation the developers engage in a group session, where they evaluate each feature individuality. For each feature there are two values calculated. The maximum need effort and the minimum effort (can be seen as the properties of EFFORT ESTIMATE)
	Calculate Effort	The developers put all the gathered information together in order to create a concept called EFFORT ESTIMATE, which holds all the information about the effort need to implement a given feature.
Calculation of alternative plans	Combine Information	This is the first stage of calculating of the alternative plans. Here all the data from previous steps is put together in one table for further evaluation.
	Identify Relationships	There are two types of relationships between features. Positive or negative. A positive interaction means that implementing both of those features will result in addition profit or on the other hand negative profit for negative relation (Ngo-The & Ruhe, 2007).
	Assess Risk	The project manager calculates the potential risk when a given feature is or is not implemented. This results in the RISK ESTIMATE concept.
	Create Alternative	Using an optimization-based method five different

	plans	alternative plans are created. Each of the plans has a different set of features included (Heikkilä et. al, 2010).
	Evaluate plans	Each of the plans is evaluated by the project manager. Different plans are suitable for different situations, therefore the project manager ranks the plans in the order they are most suitable for the next release needs. The data can be seen in Table 4 left bottom side. The final result of this phase is the ALTERNATIVE PLANS.
Prioritization of the alternative plans	Evaluate Stakeholders votes	Each of the stakeholders has different number of company shares and therefore have different level of important. At this step each stakeholder is given a weight for his vote (depending on how important the stakeholder is) for the latter evaluation.
	Prioritize plans	Similarly as in the case of feature prioritization the alternative plans were created in the previous steps are now judged by the stakeholders on a nine point scale. This will result in the EVALUATED PLANS.
	Select the final plan	After each of the plans takes a certain priority, the stakeholders and the project manager consider the priority levels as well as how suitable is the plan for the given situation and chose the best fitting plan; the FINAL PLAN.

## Concept Table

Concept	Description
ALL FEATURES	The project manager uses the user requirements and needs in order to form a list of all FEATURES, which should be included in the next release (Heikkilä et. al, 2010).
FEATURE	FEATURE is a set of logically related requirements that provide a capability to the user and enable the satisfaction of business objectives. Each feature has to have a Description and an ID (Heikkilä et. al, 2010).
PRE-SELECTED FEATURES	This is a result of the first step of SCREP. The initial set of features (ALL FEATURES) is reduced by the stakeholders for prioritization, which is along the lines of good requirements engineering practices (Heikkilä et. al, 2010).
PRIORITY	PRIORITY is a concept that holds all the information about the priority given to FEATURES by stakeholders. The higher the priority the more important the feature is (Heikkilä et. al, 2010).
PRIORITIZED FEATURES	The stakeholders evaluate the FEATURES on a nine-point scale, which results in a list of ranked FEATURES. Prioritization can be either free or cumulative. Free prioritization allows stakeholders to give any priority to any feature, while cumulative prioritization forces stakeholders to select priorities between features (Heikkilä et. al, 2010). PRIORITY is an important property of the PRIORITIZED FEATURES.
EFFORT ESTIMATE	The EFFORT ESTIMATE is done by the developers of the system. A technique

	called expert estimation is used to calculate the maximum and the minimum (properties of EFFORT ESTIMATE) effort needed to implement a given feature (Jørgensen, 2002).
RISK ESTIMATE	RISK ESTIMATE is done by the project manager. It contains information about the risks associated with implementing or not implementing a given FEATURE. Risk associated with the implementation of a FEATURE should be balanced across releases, e.g., no concentration of the most risky features into one release (Ngo-The & Ruhe, 2007).
ALTERNATIVE PLANS	With the use of the optimization-based planning method, several ALTERNATIVE PLANS are created, each plan holds different set of features. Each plan represents a possible option for the next release of the product. Furthermore, the project manager calculates how suitable are the plans for the situation. These plans are then further evaluated by the stakeholders. (Heikkilä et. al, 2010)
EVALUATED PLANS	Similarly as in prioritization of the feature, here the stakeholders evaluate all the alternative plans on a nine point scale. This scores are stored as a property (priority) of the EVALUATED PLANS. There is the assumption that stakeholders have different additional aspects to be looked at. For example, how cohesive the selected features of a release are considered or how risky their combined implementation is expected to be. Finally resulting in the EVALUATED PLANS (Heikkilä et. al, 2010).
FINAL PLAN	From the EVALUATED PLANS only one can be chosen as the next release. Therefore the stakeholders consider all the EVALUATED PLANS together with their suitable values (how suitable are they for the next release) and make a final decision, which will be the FINAL PLAN (Heikkilä et. al, 2010).

## Related Literature

The main focus of this method is prioritization of the product features according to costs, value and the risk of not implementing the given feature. "Software developers are often faced with many requirements and not enough time or money to implement them all. They must therefore find a way to prioritize the requirements to select those that will add the most to a system implementation and eliminate those not worth the cost" (Jung, 1998, p. 74). Jung's idea was to create a list of features together with their value for the product release as well as the costs of the features. The resulting features are picked according to their relative cost to value ratio.

Natt och Dag, Regnell, Gervasi and Brinkkemper (2005) argue that requirements management bridges market interaction with product development. They distinguish between two types of requirements: customer wishes and product requirements. Customer wishes are expressed via the market demands and needs where as the product requirements are those that the company finds important to include for the next release. While the requirements are being gathered many of the customer wishes and product requirements have the same basics or are literally the same. Therefore, the linguistic-engineering approach, which is being used in this method, identifies the similarities in the requirements and adjusts them appropriately. After all the similarities are sorted out the relationships between the requirements are identified. Based on relationships the requirements are evaluated and sorted in a list by value. To support this method a program called ReqSimile is used, which is used to store the list of all requirements

together with their information. The main purpose of this program is to calculate the similarities between the requirements as well as to evaluate them accordingly.

"The challenge for the company is to select a set of requirements that is deliverable within their own budget and which meets the demands of their (important) customers." (Bagnall, Rayward-Smith, Whittle, 2001, 884) This model is focused on involving the customer representatives into the process of decision making. It has some important insides on which features to include, however the involvement of the stakeholders is still minimal.

Greer and Ruhe (2004) are one of the first ones to include stakeholders inside the decision making process. They proposed a method called EVOLVE. Given a set of requirements with their effort estimations and a their categorization into priorities by representative stakeholders, the method uses a genetic algorithm to derive potential release plans within predefined technical constraints. This can be considered as one of the most influential papers for the SCERP method due to the fact that these two methods are very similar.

Ruhe (2005), also being one of the authors of the SCERP method, created a very similar method to SCERP. In his paper "The Art and Science of Software Release Planning" he created a method, which focuses on stakeholders involvement in the decision making process. The specific steps of this method are almost identical to SCERP, with one exception. Ruhe introduces dependencies between the features: "two types of dependency constraints are considered: Features can be in a coupling relation called C, or in a precedence relation called P" (Ruhe, 2005, p. 49).

## References

van den Akker, M. (2008). Software product release planning through optimization and what-if analysis. *Information and Software Technology*, 50(1–2), 101–111.

Bagnall, A. J., Rayward-Smith, V.J. Whittle, I.M. (2001). The next release problem. *Information and Software technology*, 43(14), 883 - 890.

Barney, S. (2008). A product management challenge: Creating software product value through requirements selection. *Journal of Systems Architecture*, 54(6), 576–593.

Greer, D., Ruhe, G. (2004). Software release planning: an evolutionary and iterative approach. *Information and Software Technology*, 46(4), 243–253.

Heikkilä, V., Jadallah, A., Rautiainen, K., Ruhe, G. (2010) Rigorous Support for Flexible Planning of Product Releases - A Stakeholder-Centric Approach and its Initial Evaluation. *System Sciences (HICSS), 43rd Hawaii International Conference*. 5-8 Jan, 1 - 10

Heikkilä, V. T. (2011). Aalto University. Retrieved March 25, 2012, from <https://wiki.aalto.fi/display/~vtheikki@aalto.fi/Home>

Jadallah, A. (2011). Software Engineering Decision Support Laboratory. Retrieved March 25, 2012, from <http://www.seng-decisionssupport.ucalgary.ca/alumni.htm>

Jørgensen, M. (2002). Top-down and bottom-up expert estimation of software development effort. *Information and Software Technology*, 46(1), 3 - 16.

Jung, H. W. (1998). Optimizing value and cost in requirements analysis, *IEEE Software*, 15(4), 74-78.

Karlsson, L. (2006). Pair-wise comparisons versus planning game partitioning—experiments on requirements prioritization techniques. *Empirical Software Engineering*, 12(1), 3-33.

Natt och Dag J., Regnell B., Gervasi V., Brinkkemper S., (2005). A linguistic-engineering approach to large-scale requirements management. *Software, IEEE*, 2(1), 32-39.

Ngo-The, A., Ruhe, G. (2007). A systematic approach for solving the wicked problem of software release planning. *Soft Computin*, 12(1), 95-108.

Rautiainen, K. (2010). Kristian Rautiainen. Retrieved March 25, 2012, from <http://www.soberit.hut.fi/~kqr/>

Ruhe, G. (2005). The art and science of software release planning. *IEEE Software*, 22(6), 47-53

Ruhe, G. (2010). Software Engineering Decision Support Laboratory. Retrieved March 25, 2012, from [http://www.seng-decisionssupport.ucalgary.ca/research\\_chair.htm](http://www.seng-decisionssupport.ucalgary.ca/research_chair.htm)

Sharp, H. (1999). Stakeholder Identification in the Requirements Engineering Process. *Database and Expert Systems Applications*, 387 - 391.

Wieggers, K. (2003). *Software Requirements*, Microsoft Press, Redmond, WA

## Appendix A

### SCERP Template

	Release Plan					
Stakeholder	O1	O2	O3	O4	O5	M
S01						
S02						
S03						
S04						
S05						
SUM						
Median						
STDV*						

This is a template for the final deliverable of the SCERP method (see Table 4), which lists all the possible plans for the next release of the product together with their respective scores from the stakeholders.

\*STDV - Standard deviation