Master Thesis
Software Engineering
Thesis no: MSE-2010:28
September 2010

Empirical Evaluation of a Universal Requirements Engineering Process Maturity Model

Mai Nguyen

School of Engineering
Blekinge Institute of Technology
Box 520
SE – 372 25 Ronneby
Sweden

This thesis is submitted to the School of Engineering at Blekinge Institute of Technology in partial fulfillment of the requirements for the degree of Master of Science in Software Engineering. The thesis is equivalent to 20 weeks of full time studies.

Contact Information:
Author(s):
Mai Nguyen
Address: Studenvagen 1, Rum 20, Ronneby, Sweden 37240
E-mail: mainguyentp@gmail.com

University advisor(s):
Dr. Mikael Svahnberg
Department of Software Engineering and Computer Science
Blekinge Institute of Technology

EMSE Co-supervisor(s):
Dr. Oscar Dieste
Facultad de Informática
Universidad Politécnica de Madrid

School of Engineering
Blekinge Institute of Technology
Box 520
SE – 372 25 Ronneby
Sweden

Internet : www.bth.se/tek
Phone : +46 457 38 50 00
Fax : + 46 457 271 25

ABSTRACT

Context. Software products are usually developed for either a specific customer (bespoke) or a broader market (market-driven). Due to their characteristics, bespoke and market-driven development face different problems, especially in the requirements engineering areas. Moreover, many of these problems are caused by an inadequate requirements engineering process. Hence, in order to improve the process and subsequently the software quality, the maturity of the RE process must be evaluated. Although there are many process assessment initiatives done in bespoke development, there is a need for models covering both approaches. Uni-REPM, which can assess the RE process maturity in all environments, is such a model.

Objectives. This study presents an academic and industrial evaluation of the Uni-REPM model before transferring it to industry.

Methods. The first validation was conducted in the form of interviews with seven academic experts in which the model was scrutinized for its correctness and completeness. Subsequently, the model and the assessment method were applied and validated in 4 industrial organizations locating in Denmark, Spain and Singapore.

Results. Based on the feedback obtained in the validation, refinements were made to the model to improve its quality. In addition, the evaluation result analysis of each industrial project is useful in indentifying weak areas in the process and suggesting possible improvement practices.

Conclusions. The study shows that Uni-REPM is a quick, simple and cost-effective solution to assess the maturity level of the Requirements Engineering process of projects. Moreover, the assessment method using checklist is highly usable and applicable in various international development environments.

Keywords: requirements engineering, process evaluation, empirical

ACKNOWLEDGEMENT

This thesis work was accomplished not only by my solitary effort but also with help and contributions from many people. Hence, I would like to take this opportunity to express my great gratitude to all of them.

First of all, I would like to dedicate this thesis to my beloved parents. Without their love and support, I would not have been here and completed this challenging yet interesting research.

Secondly, I had the good fortune to work under the supervision of two distinguished scholars of the requirements engineering field, Dr. Mikael Svahnberg and Dr. Tony Gorschek. I am deeply grateful for their invaluable advice, guidance and ideas to overcome bottleneck problems encountered during the thesis. Moreover, I want to extend my thanks to Dr. Oscar Dieste, my EMSE Co-supervisor.

In addition, I greatly appreciate the contribution of the following experts to the improvement of the model: Dr. Jürgen Börstler, Dr. Kristian Sandahl, Dr. Samuel Fricker, Dr. Inge Van de Weerd, Dr. Christof Elbert, Mr. Richard Berntsson Svensson and Mr. Krzysztof Wnuk.

I am also indebted to the software professionals participating in the research for their willingness to collaborate and useful feedback to make the work more practical.

Last but not least, I would like to thank all my friends for their moral support, especially Mrs. Nguyen Thi Thanh Loan who has assisted me tremendously throughout the thesis work and Mr. Michael Unterkalmsteiner for the fruitful discussion.

CONTENTS

1 INTRODUCTION

Nowadays, virtually all countries deploy software in their various daily operations spanning from financial to electrical industries [1]. On the one hand, software has traditionally been developed according to customers' requirements written in the contract between producer and customer(s), for example an air traffic control system. This type of development is called "bespoke" where "software is developed with a specific customer in mind and when it is often possible to have direct contact with this one user/customer" [2]. On the other hand, there is an increasing number of software produced for a broader market [3, 4], which is also known as commercial-off-the-shelf (COTS) [5] or packaged software [6]. One example is word processor software. This type of approach is called "market-driven", in which the development costs of the generic software product are divided among many buyers on the open market [7]. There are many differences between market-driven and bespoke development in terms of primary goals, measure of success, main stakeholders, distance to users and lifecycle [8]. While time-to-market is the survival attribute to market-driven organizations to stay competitive, bespoke organizations only need to comply with a fixed requirements specification. Moreover, market-driven organizations measure their success in terms of sales and market share whereas their counterparts use customer satisfaction and acceptance as success criteria. The main stakeholder in the case of market-driven companies is the developing companies themselves. On the other hand, for bespoke organizations, customer organizations are their main stakeholders. Another difference lies in their users and the distance to them. In case of market-driven organizations, the users are unknown and may not exist until the product is first released to the market. They are usually hard to reach as well. In bespoke case, the users are identifiable and easier to reach. Regarding the lifecycle, market-driven development has several releases while bespoke development usually has one release followed by maintenance.

Due to these above differences, the characteristics of the software process and especially, the Requirements Engineering (RE) process applied in each case also vary [9, 10]. The Requirements Engineering process consists of activities to discover, document and maintain a set of requirements for a computer-based system [11]. According to Regnell et al., although both approaches perform requirements elicitation, documentation and validation activities, there is a distinction in the purpose, level of focus as well as methods, techniques and tools used in each case [7]. Moreover, while bespoke RE focuses on negotiation and conflict resolution, market-driven requirements engineering (MDRE) emphasizes prioritization, cost estimation and release planning [8]. Besides, there are some exclusive challenges faced by MDRE process [12]. Among them are release planning [13], choosing a suitable process [14, 15, 16], balancing market pull and technology push [6] and bridging gap between marketing staff and developers [17]. Apart from that, both bespoke RE and MDRE suffer several similar problems [18, 19, 20, 21, 22]. Apart from organizational problems such as user communications and inappropriate skills, there are other problems lying in the requirements process [20]. Poorly specified requirements, undefined process and inadequate traceability are the most common problems identified in practice [22].

In order to overcome those problems and develop better software, organizations should improve their RE processes as almost half of the development problems stem from requirements [20]. According to Villalón et al., assessment of the current process status is the first step of the continuous improvement process [23]. There exists a number of models developed for assessment purpose that are applicable to bespoke development, namely CMMI [24], ISO 9000 [25], REGPG [26] and REPM [27]. On the other hand, MDREPM, as a counterpart of the REPM, is the only assessment tool applicable to market-driven requirements engineering environment [28]. However, the line between bespoke RE and MDRE has become blurred as an increasing number of software companies that previously

developed systems for specific customers is moving towards MDRE as a matter of economic urgency [9]. The opposite way also happens when more and more companies start with a mass market product and offer customizing service to specific customers' needs, for example a SAP system [1, 7]. In these cases, the software producer has to deal with both MDRE and bespoke RE. Hence, there is a need of an assessment model addressing these cases. Uni-REPM, which can assess the RE process maturity in all environments, is such a model [29].

Figure 1 illustrates the research context covered above in a graphical form.

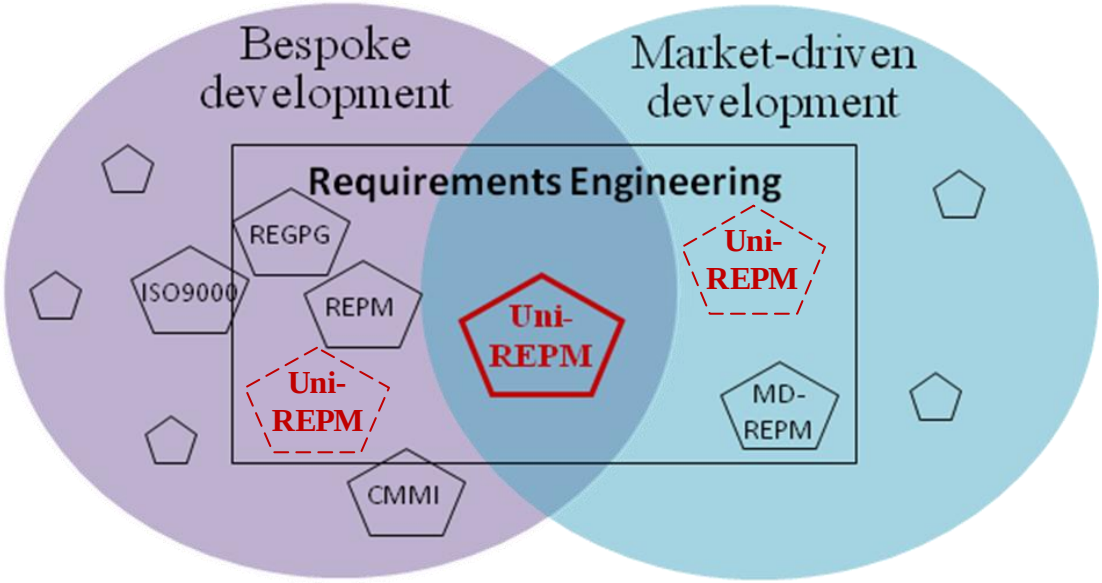


Figure 1. Research context

This study presents the continuation version of the work done in "Requirements Engineering Process Maturity Model: creation" [29]. In that study, a new Uni-REPM model was proposed for assessing the maturity of the requirements engineering process in various development environments. On the other hand, in this study, the model underwent the scrutiny of seven academic experts regarding some keys aspects before being applied in industrial settings. Then the pilot application was performed in four companies in order to assess the understandability and applicability of the new model. The main purpose of this study is to evaluate and refine the Uni-REPM model so as to produce useable and useful results in both academic and industrial settings. The contribution of this study is the improved practical version of Uni-REPM model and the project evaluation results of the participating companies.

1.1 Aims and objectives

The aim is to validate a newly created model for assessing requirements engineering process maturity in industrial organizations and then improve it based on the collected feedback.

The objectives of this study are to:

- Ensure the model quality is suitable for industrial pilot.
- Apply the model in industrial organizations to assess RE process maturity and evaluate its usage.
- Finalize the model by analyzing feedback information and taking response action.

1.2 Research questions

Based on the above aims and objectives, a list of research questions is constructed. These questions will be addressed during the course of the thesis and the methods used to find answers to those questions can be found in Section 1.4 "Research Methodologies".

RQ1: To what extent is the model suitable for industrial pilot, in terms of its correctness, completeness and applicability?
RQ2: To what extent is the model applicable, usable and useful for industry application?
RQ3: What improvements can be done to the model based on the findings in RQ1 and RQ2?

1.3 Expected Outcomes

EO1: Evaluation results of the Uni-REPM model prior to the industrial pilot.
EO2: Evaluation results of the Uni-REPM model in practice.
EO3: Description of the final version of Uni-REPM.

1.4 Research Methodology

The approach of this research follows loosely the technology transfer framework created by Gorschek [30]. In that framework, after the problem formulation, a candidate solution is created based on study of state of the art. Then static and dynamic validations are performed so as to refine the solution and test it for usability and scalability [30]. In this case, a new Uni-REPM model was already established on the findings of literature review and systematic review processes in the previous study [29]. Following that, in this study, the proposed Uni-REPM model underwent several validation steps to assure that the model was of good quality and usable before its release.

Firstly, a static validation was performed in order to answer RQ1 ("To what extent is the model suitable for industrial pilot, in terms of its correctness, completeness and applicability?"). The objective of this question is to uncover defects in the model regarding the three above aspects and ensure the model quality prior to the industrial pilot. Therefore, the static evaluation proves to be a suitable method to attain this objective because it provides fast, early feedback and helps to identify potential problems without using industry resources [30]. In this stage, the Uni-REPM model was reviewed by academic experts to validate its accuracy and adequacy in both content and presentation aspects without being executed. Then semi-structure interviews were conducted either by face-to-face meetings or through audio conference to gather feedback from the experts. The face-to-face meetings were strived for whenever possible because it is a rich means of communication. On the other hand, audio conference through Skype was also conducted in case the experts were located geographically far from the researcher. After the interviews, feedback results were double-checked with the reviewers to avoid any misunderstanding or misinterpretation. Based on the feedback, weaknesses of the model were identified, analyzed and RQ3 ("What improvements can be done to the model based on the findings in RQ1 and RQ2?") was discussed. Response actions were then decided and the model was refined and improved before industry piloting. Detailed information about the static validation design, result and improvement can be found in Chapter 3 – Static Validation.

In the second stage, the improved model was applied in four industrial organizations. The dynamic validation was used to realistically evaluate the model while minimizing risk [30]. This step was performed in order to answer RQ2 ("To what extent is the model useable and applicable for industry practitioners?"). The necessary data was collected from practitioners participating in industrial projects by two means: through structured interviews and a self-administered questionnaire. A self-administered questionnaire is usually used as an instrument for data collection in survey research [31]. The structured interviews were conducted through Skype because all the participants were located in different countries than the researcher. The self-administered questionnaire was used when it was impossible to arrange interviews due to practitioner's busy schedule and different time zones. A model checklist comprises a large part of both the interview questions and the questionnaire. Answers to the checklist were used as inputs for not only the project assessment but also the model evaluation. The collected data was recorded in electronic documents and in audio tape

(if possible). Based on the evaluation results, the feedback from the practitioners were analyzed and refined in order to improve the model. The model was modified once more to become the final version. Detailed information about the dynamic validation design, result and improvement can be found in Chapter 4 – Dynamic Validation.

Table 1. Mapping of research questions, methodologies and expected outcomes

Research question	**Methodology**	**Expected Outcome**
RQ1	Static validation	EO1
RQ2	Dynamic validation	EO2
RQ3	Analysis	EO3

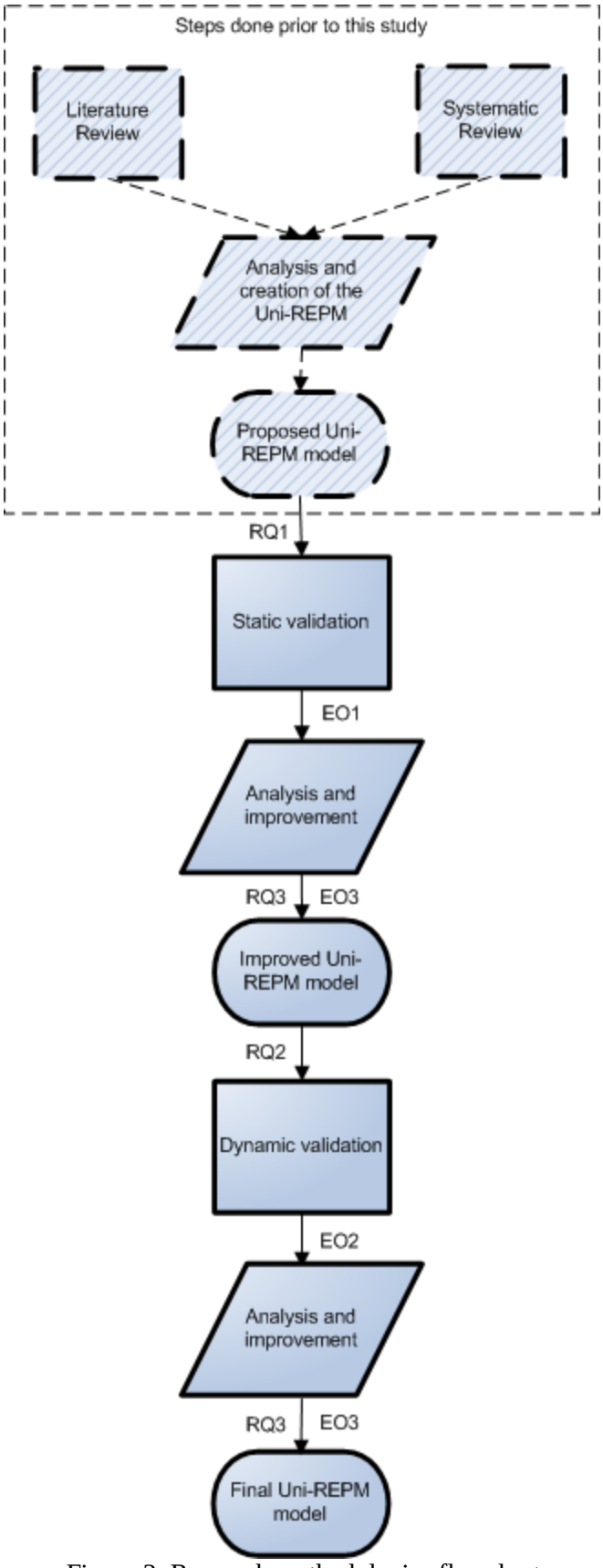


Figure 2. Research methodologies flowchart

1.5 Related work

In the literature, there are several existing models which support the organization to evaluate process maturity. The most renowned models are Capability Maturity Model Integration (CMMI) or more specifically CMMI for Development (CMMI-DEV) [24], and ISO 9000 [25]. CMMI_DEV, which is relatively huge and spans the whole lifecycle of the project, does not cover requirement engineering practices in great detail. This fact also applies to ISO 9000. Moreover, both models consume lots of time and resource to assess the maturity [32, 33].

In addition, there are also several models that are tailored to the RE process. Among those are the Requirements Engineering Good Practice Guide (REGPG) designed by Sommerville and Sawyer [22], and the Requirement Engineering Process Maturity (REPM) developed by Gorschek et al. [27]. Research has reported that REGPG provides a large set of detailed practices with adequate ranks but lacks of guidelines for implementation [24]. The REPM model is a map of sort, covering the requirements engineering process. It can be used to evaluate the maturity of Requirements Engineering process in a project and see what is done, what is not and possible improvements [27].

While all of the above models were created towards bespoke development, MDREPM, which was based on REPM, is the only RE process assessment tool made for market-driven development [28]. It provides software organizations not only a collection of good practices in MDRE but also a step by step process improvement path towards a better requirements engineering process.

There is a need of a RE process assessment tool for software companies that operate in both bespoke and market-driven development. Uni-REPM, which is a universal Requirements Engineering process maturity model, is such a choice [29]. It is aimed to function in either environments or both [29]. However, while other models have been applied or piloted in industry, Uni-REPM has not. It is clear that "software practitioners need supporting evidence from case studies, field studies and experiments before adopting new technologies" [2]. Moreover, there is a need for packaging and transferring research solutions in a more elaborate way to industry together with the real evaluation of their advantages [22]. Therefore, the Uni-REPM model has to be applied to real projects in order to judge its usage. This thesis targets to evaluate the model in industrial settings.

1.6 Thesis structure

The thesis comprises of two parts: the main part and the appendix. The first part presents the Introduction, Uni-REPM overview, Static Validation, Dynamic Validation, Validity Threats and Conclusion.

The second part contains all the appendices that support the first part. The 9 appendices are: Appendix A - Static Validation questions, Appendix B - Uni-REPM brochure, Appendix C - Dynamic Validation introductory questions, Appendix D – Project Evaluation Result 1, Appendix E – Project Evaluation Result 2, Appendix F – Project Evaluation Result 3, Appendix G – Project Evaluation Result 4, Appendix H – Self-administered questionnaire, Appendix I – Uni-REPM checklist, Appendix J – Requirements Engineering Process Maturity Model (Uni-REPM).

Chapter 1 - Introduction
This chapter contains information about the problem that motivates this study, the aim of the study as well as the research questions. It also consists of the related work and research methodologies used to find answers to those questions.

Chapter 2 - Uni-REPM model overview

This chapter presents core concepts of the Uni-REPM model spanning from its structure, maturity levels to its application as an RE process assessment tool. This serves as a foundation so that readers can understand the subsequent chapters.

Chapter 3 - Static Validation
This chapter discusses about the first round of validation of Uni-REPM model using expert opinion. Seven experts were asked to review the model and their feedback was collected through interviews. The aspects of the model that need to be validated are its correctness, completeness and applicability. The chapter covers the interview design, expert feedback and improvements made on the model based on the feedback.

Chapter 4 - Dynamic Validation
The fourth chapter presents the second round of Uni-REPM validation in industry settings. This was done by apply the model on project evaluation in four organizations. In this validation round, the understandability and applicability of the model is the main focus of the evaluation purpose.

The chapter contains the validation design, results as well as the model improvement suggestion obtained indirectly from the interview process.

Chapter 5 - Validity threats
This chapter discusses the validity threats to the static and dynamic validation.

Chapter 6 - Conclusion and Future work
This chapter discusses the overall conclusion of the author drawn out from the whole research project together with the possible follow-up work of this research.

2 REQUIREMENTS ENGINEERING PROCESS MATURITY MODEL UNI-REPM

This chapter represents the Uni-REPM model covering its structure, components and usage. The reason for having this chapter is to lay a foundation for the readers to understand and follow the subsequent evaluation steps, namely Static Validation and Dynamic Validation. As those two are the main contribution of this thesis and not the creation of the model, only the core concepts of the model are discussed here. Should the readers want to learn more details about the model, they can refer to Nguyen L.'s paper for further reading [29]. The final version of the model after being modified in Static and Dynamic Validations can be found in Appendix J.

The chapter has two main parts: the model description and its usage. The first part starts with a simple introduction of the model, followed by the description of its structure and the maturity levels. The second part covers the usage of Uni-REPM model as well as a method for using it to assess process maturity.

2.1 Introduction

Uni-REPM is a light-weight model assessing the maturity of RE process through sets of necessary activities. The assessment is performed by mapping those ideal activities to real work.

Uni-REPM is a means to identify the strengths and weaknesses of the RE process in organizations. Furthermore, it provides organizations recommended practices to improve their requirements engineering process from basic to advanced level.

The model was constructed based on extensive literature reviews of REPM model [27], CMMI [24], ISO 9000-TickIt [25] together with a systematic review on market-driven requirements engineering researches.

2.2 Model Structure

The model hierarchy has three levels, namely ***Main process area*** (MPA), ***Sub-process area*** (SPA) and ***Action***. On the top level of the model, there are seven Main process areas corresponding to requirements engineering main activities. Each MPA is further broken down into several SPAs, which facilitates better understanding. On the bottom level, an action, a smallest unit, denotes a certain activity that should be done or a certain item that should be present.

So as to improve the model structure and its coherence, closely-related actions are grouped together and put under one SPA if possible. Besides, every MPA has one SPA called "General actions" where other actions reside. An action must be attached to a SPA and there is no MPA or SPA located under an action. The model components are summarized and illustrated in Figure 3.

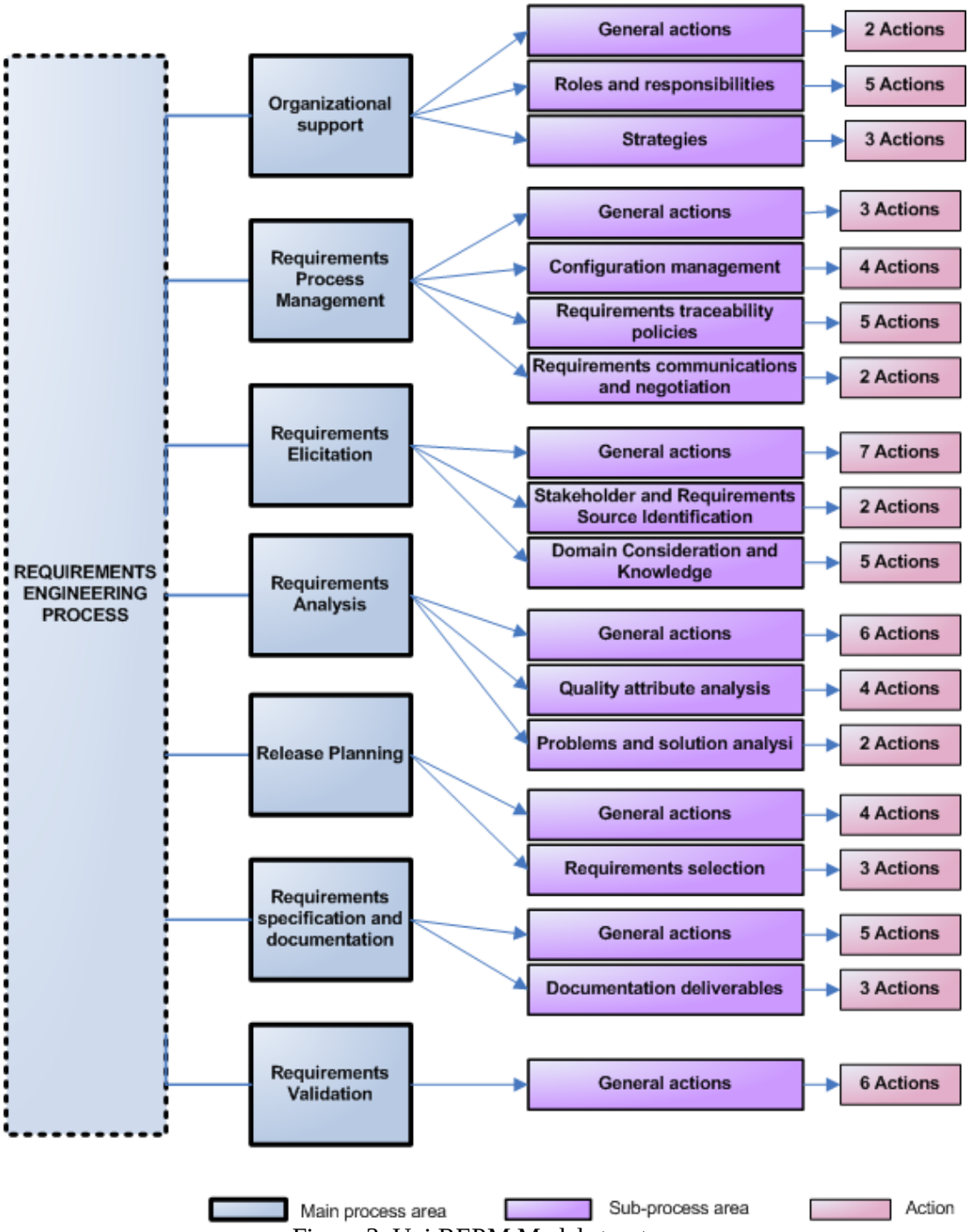


Figure 3. Uni-REPM Model structure

2.2.1 Main Process Area (MPA)

On the top level of the model, a main process area represent a cluster of related practices in one main requirements engineering activity such as Elicitation.

There are seven MPAs in the model: Organizational Support, Requirements Management Process, Elicitation, Requirements Analysis, Release Planning, Documentation and Requirements Specification, and Requirements Validation.

Each MPA has a unique identifier which enables traceability throughout the model. For example, “Organizational Support” MPA is referred to as “OS”.

2.2.2 Sub-Process Area (SPA)

Sub-process area contains closely related actions, which help to achieve a bigger goal. The unique identifier assigned to each SPA is composed of the MPA identifier to which the SPA attaches and its abbreviation. For example, "OS.RR" represents a sub-process area "Roles and Responsibilities" which resides under MPA "Organizational Support".

2.2.3 Action

The smallest unit in the model is called "action" showing a specific good practice. By performing the action, the organization can improve their process and gain certain benefits.

Actions also follow the same format to form their unique identifiers. They are identified by the MPA/SPA under which they reside, followed by an "a" which stands for "action" and their position in the group. For example, "OS.a1" points to the first action which attaches directly to MPA "Organizational Support".

Each action is assigned a certain level depending on its implementation difficulty and essentiality for the requirements engineering process. The level structure is discussed in detail in section 2.3.

Example(s) and Supporting Action(s)

Within the description of each Action, there can be **Example(s)** and **Supporting Action(s)**. The idea of **Example(s)** is to give practitioners suggestions on proven techniques or supporting tools when performing the action. It is worth noticing that the Example item, as the name suggests, is not an exhaustive list. Therefore, companies are not restricted to apply only those in order to fulfill an action. In addition, the **Supporting Action(s)** provided links to other actions which will benefit the practitioners when implementing them together.

2.3 Process maturity levels

Uni-REPM makes use of an ordinal scale to assess the maturity of the process. The levels to assess process maturity is inspired from the REPM [27] and REGPG [26] models; and the book "*Software requirements*" [34]. Concerning the fact that Requirements Engineering Processes applied in industry are usually small-sized and ad-hoc [22], three levels of maturity were defined, namely ***Basic, Intermediate*** and ***Advanced.***

Level 1 – Basic

The aim of this level is to achieve a rudimentary repeatable requirements engineering process. The process in this level is defined and followed. Quality of requirements is managed through relevant stakeholder involvement in elicitation, in-depth requirements analysis and pre-defined document standards.
However, the process does not maintain any kind of communication among stakeholders and within the organization in term of strategies.

Level 2 – Intermediate

In this level, the process is more rigorous because it involves various perspectives and is led by product strategies/goals. Roles and responsibilities for particular tasks are clearly defined and documented. Change requests are handled in a consistent manner. Well-informed decisions about requirement selection can be made by analyzing and prioritizing the requirements systematically.
This process still stays in "*present-state*"; meaning that there is no activity performed to collect and analyze data/feedback for future improvement of the process.

Level 3 – Advanced
This level denotes the most mature process. The improvements in the process are shown in the advanced way of capturing requirements, ensuring their high quality, maintaining communications and common understanding among different stakeholders and pro-actively assessing the decision making process.
The process takes into account the "*future-state*" since it not only covers pre-defined and structured procedures but also adequately pay attention to future works (e.g. reusable materials, port-term evaluation, etc.).

2.4 Model usage

To assess the maturity of a RE process, the users basically perform a mapping from the actions present in the model to the activities in a real process using the checklist. The checklist is actually a direct transformation of the model into question form. A snapshot of the checklist is shown in Figure 4. The whole checklist can be found in Appendix I.

The checklist follows the same structure as the model with questions grouped according to the MPA and SPA. For each action in the model, there is a corresponding question or group of questions to verify if the action is done or not. The Action ID which links the question(s) to the associated action in the model helps the users in case they need to locate the item for further information or clarification.

When answering the questions, the users may encounter one of the following situations:

- The action was deemed vital but was performed partially or not at all in this RE process. It should be marked as **"Incomplete"** (IC)
- The action was completed in this RE process. It should be marked as "**Complete"** (C)
- The action was not necessary or possible to be performed in this process. It should be marked as **"Inapplicable"** (IA)

More about "Inapplicable"

In reality, as organizations and processes vary in their characteristics and environments, they may not benefit from implementing all the actions in the model. Some of the actions are deemed unnecessary to be performed in particular situations of organizations. For example, a company has an internal glossary of terms but it is not released to the users. They find it hard and useless to give the user the whole document and ask them to read it as they might not do so. Instead, the terms and examples are explained directly to the users in the meeting and it was more effective this way. In this case, the action "OS.GA.a1 Create a Product-wide Glossary of Terms (Basic Level)" is not useful for them. If we consider it as "Incomplete", the process may not reach the Basic level because not all actions in this level are fulfilled. This is even more unfair if all other actions in higher maturity levels are completed. Therefore, companies should not be "punished" if they do not perform a certain nonessential action (in their point of view). In order to take into account this factor, the option "Inapplicable" is devised. In this way, the model is more fitting to the real process and the evaluation result is less distorted. Besides, the differences between two types of development settings (bespoke and market-driven) do exist [8]. Therefore, in some cases, the organization may find some actions only applicable in one of the settings.

Whether an action is "Inapplicable" or not is solely based on the judgment of the project evaluator. Reasons for deeming an action "Inapplicable" should be considered carefully to avoid accidentally skipping an important action. Lack of time, resource or unawareness cannot be accounted for an "Inapplicable" action.

Action ID	Question	(C)	(IC)	(IA)	Comment / Reason if Inapplicable
OS	**Organizational Support**				
OS.GA	General Actions				
OS.GA.a1	Do you have a product-wide glossary of terms to ensure that the key concepts in the domain are properly understood by all parties?				
OS.GA.a2	Do you have training about requirements development and management processes as well as necessary skills to perform the job?				

Figure 4. Uni-REPM Checklist snapshot

2.4.1 How to read the result?

After answering all the questions in the checklist, the users can calculate the results for each MPA by summing up the answers according to each level and consider the following rules.

- For each MPA, all actions at a certain level must be **Completed** (or **Inapplicable**) in order for the MPA to achieve such level.
- For the whole process, all actions in the seven MPAs at a certain level must be **Completed** (or **Inapplicable**) in order for the process to achieve the level.

It is impractical that all companies should aim for the highest maturity level as this would require a lot of time, effort and resource. Instead, based on their own needs, they should decide which level will be most beneficial for them and strive for it. However, it is recommendable that the process achieves homogenous Uni-REPM levels across all the MPAs. The reason is that although the MPAs deal with different facets of the process, they together make up the whole process. Taking an organization with Basic level for the Requirements Elicitation MPA and an Advanced level in Documentation and Requirements Specification MPA as an example, this would mean that poorly elicited requirements are specified and documented perfectly. This is an imbalanced process and will not bring much benefit. Therefore, companies should pay more attention to the weaker MPAs and improve them in order for the process to perform consistently and effectively.

An example

The result of MPA "Organizational Support" after evaluating may look like in Table 2.

Table 2. Assessment result in MPA "Organizational Support"

Level	Actions in real process		Total actions in OS in Uni-REPM
	Completed	Inapplicable	
Basic	0	0	2
Intermediate	3	1	6
Advanced	1	1	2

To have a better view, the result can be presented in graph as follows.

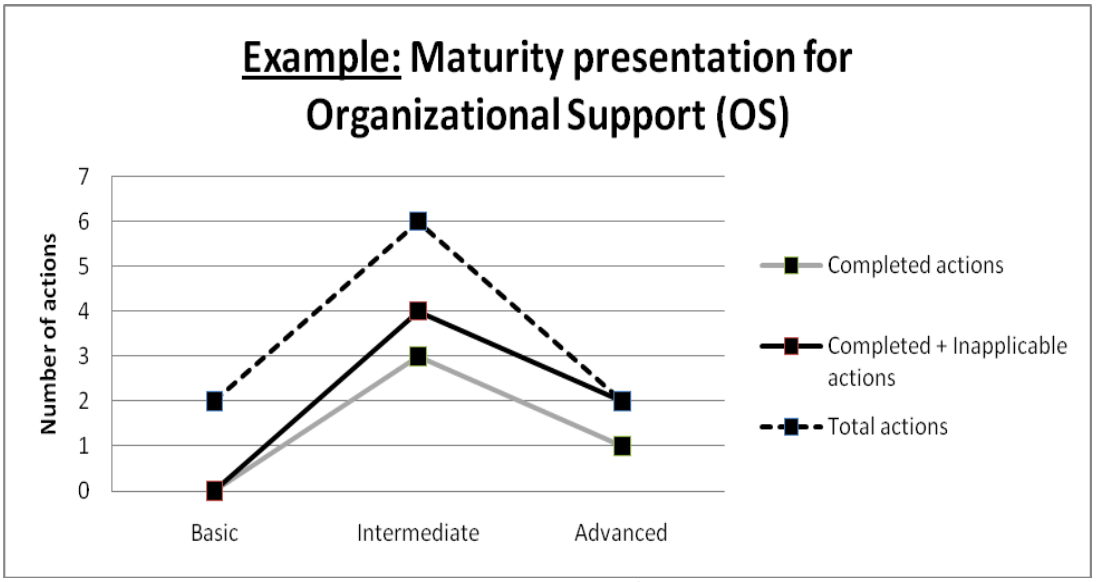


Figure 5. Graphical presentation of assessment results

The grey line presents actions which were completed in the real process. In this case, no action was done at the lowest level; three actions were completed in Intermediate level and one action in the highest level. The black line presents actions completed together with actions that were not performed due to unnecessary or inapplicable reasons in the assessing organization. The distance between the grey line and black line is called the "model lag", which represents the number of inapplicable actions. Hence, the model lag shows the applicability of the model in the real setting. In this case, the model lag is fairly small with only two inapplicable actions. This means a high applicability of the model.

The dashed line in the graph presents the total actions that should be completed at the three levels in "Organizational Support" MPA. For example, at Basic level, there are two actions that should be finished. The difference between the black line and the dashed line is important because it denotes the improvement area of the process. It shows how many additional actions that should be done in order to attain a certain level of maturity.

Overall, the graph denotes that, in this MPA, the process has not completed all the actions at Basic level. Hence, according to the above rule, the MPA resides on **Level 0.** In order to reach the Basic level, two more actions have to be done. If the company aims for Intermediate level, it has to perform two Basic actions and another two Intermediate ones. Similar work can be done with other MPAs to achieve the result for the whole process.

3 STATIC VALIDATION

The purpose of this chapter is to present the static validation of the Uni-REPM model through experts' opinion in order to find out whether the knowledge in literature was reasonably transferred and presented in the model. The reason to choose static validation is that it helps us to obtain feedback and improve the model quickly, hence prepare the most possibly viable model for the next step – dynamic validation. In this step, the model was presented to subjects who participated in the validation and their feedback to improve the model was collected during the interviews.

The validation targets to evaluate Uni-REPM model in terms of the following aspects:

- Completeness: to make sure the model presents all necessary RE practices with adequate information. Besides ensuring that existing practices in the model were covered adequately, this step attempted to detect other necessary RE practices that might not be captured in the previous literature review steps.
- Correctness: The degree to which software, documentation, or other items meet specified requirements [35]. In this case, it is to ensure the model content and presentation, especially the names, maturity levels, placements and descriptions of all necessary RE practices in the model, are correctly presented.
- Applicability: to evaluate whether the model can be applied in industrial settings and to what extent.

This chapter consists of three main parts: the design, the execution and the improvement. In the first part, steps to design the validation process were presented. The execution part describes the results obtained from the interviews with the experts. Last but not least, the improvement part discussed actions taken in correspondence to the feedback in order to enhance the model as well as the conclusion of the whole process.

3.1 Static validation design

3.1.1 Validation process

The evaluation was performed as demonstrated in Figure 6. The overall description is given below and more details about each individual steps are discussed in the separate following sections.

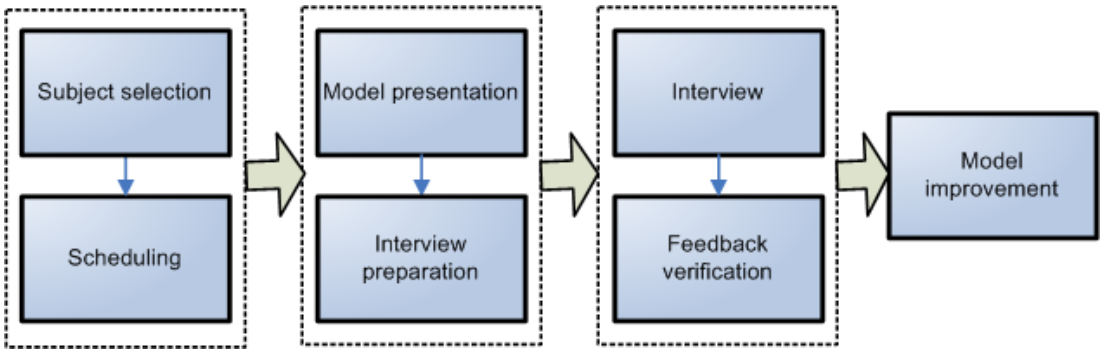


Figure 6. Static validation process

At the first step, the experts were ***selected*** according to some defined criteria and invited to participate in the validation. Once the acceptance was established, a ***schedule*** for the subsequent steps was set and agreed to by the expert and the researcher.

In the second step, the Uni-REPM model and a short list of questions were ***presented*** to the subjects prior to the validation meeting through emails. The rationale for this step is that the subjects could spend their most convenient time to carefully explore the model and ***prepare*** ideas/answers in the validation meeting. Apart from that, information about the subject background was also obtained through email so that the interview questions could be tailored more towards the subject's expertise area prior to the interview.

After that, the main validation step was conducted in the form of ***interviews***. The detailed design of the interview is described in the next section. The interview result was then summarized and sent back to expert to ***verify*** its correctness and avoid misunderstanding. Finally, the feedback from subjects were consolidated and analyzed in order to ***improve*** the model.

3.1.2 Subject selection

Since the static validation results are heavily influenced by the interviewed subjects, the criteria for this step were carefully considered. In order to acquire valuable and useful feedback, the following subject selection criteria were taken into account:

- Subjects should have a research interest in Requirements Engineering or Product management. This is to ensure the commitment of the subject through the whole validation.
- Subjects should have contributed relevant important publications in the study area. This is to ascertain that the subjects have right knowledge to evaluate the model.
- Subjects should have close collaboration with industry. Since the model targets to be applied in industrial organizations, it is necessary that the subjects have industrial experience to be able to evaluate the applicability of the model.

Based on these criteria, the selection was done using random sampling. The researcher searched for experts from many sources (publications, personal recommendation) and contacted them through emails. After one month, seven out of 17 experts contacted accepted to participate in the validation. The list of experts is shown in Table 3.

Table 3. Static validation participants

Expert name	Expert title	Country
Kristian Sandahl	Professor	Sweden
Jürgen Börstler	Professor	Sweden
Samuel Fricker	Doctor	Switzerland
Inge Van de Weerd	Doctor	The Nederland
Christof Elbert	Doctor	Germany
Richard Berntsson Svensson	PhD student	Sweden
Krzysztof Wnuk	PhD student	Sweden

The first interview was conducted with Dr. Inge Van de Weerd. Mrs. Weerd is currently an assistant professor at Department of Information and Computer Science at Utrecht University, The Netherlands. She has five years researching on Requirements Engineering and her research activities focus on mostly requirements management, release planning and product roadmapping and a little bit of portfolio management. She has carried out several case studies (consisting of interviews, assessment and advice) at various companies.

The second expert chosen to interview was Dr. Samuel Fricker. He is Senior Research Associate at the Requirements Engineering Research Group at the Department of Informatics

of the University of Zurich. He has been researching on Requirements Engineering for seven years, mostly on Requirements Communications. He has collaborated with industrial organizations in approximately 50 projects as a requirements engineer, requirements engineering process owner, and requirements engineering coach during the full seven years.

The third interview was performed with the participation of Mr. Krzysztof Wnuk. Mr. Wnuk is a Technical Licentiate PhD Student at Lund University, Sweden. He has been researching on decision making process (scoping) for three years. Before starting the PhD studies, he worked in industry for two years. He has been working with Sony Ericsson throughout his entire PhD studies (three years and ongoing). He has experienced in working with large projects with more than 300 features in the scope, which means 3000 requirements each.

Associate Professor (docent) Jürgen Börstler was the fourth expert that the researcher has an honor to interview with. He is a director of studies at the Department of Computing Science at Umeå University, Sweden. He has been researching on Requirements Engineering for 13 years and his expertise is on OOA (use cases), process assessment (software development in general) and traceability. He has been involved in several industrial projects during the last ten years, of which two are larger projects: QMSE (Quality Management for Small Enterprises) and Requirement for software product lines. In QMSE, he did assessments based on adapted versions of the CMM for software.

In the fifth interview, the model was discussed with Mr. Richard Berntsson Svensson. He is a Licentiate PhD Engineer at Lund University, Sweden. During five years of research, he has been focusing on market-driven requirements engineering, software product management, process improvement, quality requirements and cost/effort estimation. He has been collaborating with industrial organizations in several projects for about four to five years.

The sixth expert contacted was Dr. Christof Ebert. He is managing director and partner at Vector Consulting Services. His consulting and research covers engineering management, process improvement and requirements engineering. He has been directing engineering projects, R&D platform programs and process improvement initiatives, achieving substantial quality and productivity improvements and cycle time reduction.

Last but not least, in the seventh interview, a one-and-a-half-hour discussion was granted with Professor Kristian Sandahl at Linköping University, Sweden. Since 1994, Professor Sandahl has been researching on Requirements Engineering and his expertise covers broadly from prioritization, release planning, traceability to use-cases, elicitation methods and non-functional requirements. He has also worked with Ericsson research for five years with a focus on “early phases”.

3.1.3 Model presentation

A copy of the Uni-REPM model was sent to the experts beforehand so that they could prepare for the interview. In addition, for the better feedback acquisition, a short list of questions (namely ***Participants Background Questions***) was also sent to the expert prior to the interview. These questions aim to extract information about the experts and their experience in the area. The answers help to draw a better view about the experts, hence support semi-structure interview later (i.e. focus the questions on expertise of the subjects). The ***Participants Background Questions*** can be found in Appendix A.

3.1.4 Interview

In order to gather expert opinion about the model, interviews are proven to be a suitable technique. Among many of its objectives, interview is a commonly used technique to collect historical data from the memories of the interviewees and to collect opinions or impressions about something [36].

There are three types of interview: structured, semi-structured and unstructured. In structured interview, the interviewer has a clear view of what information is sought in the interview, hence the questions are very specific [36]. On the other extreme end, interviewer in unstructured interview wants to elicit as much information as possible on a broadly defined topic [1]. Other studies combine these two approaches into Semi-structured interviews. This type combines specific questions, which bring forth the foreseen information, with the open-ended question, which elicit unexpected types of information coming up during the interview [36]. In this study, the interviews were designed to be semi-structured. The reason to choose semi-structured interviews is that every expert has different strength in the field and the semi-structured interview allows flexibility in discovering specific areas in depth. Some of the questions were prepared beforehand to ensure certain themes/points were covered.

As most of the experts are located far from the researcher's location, five interviews were conducted by audio conference while the other two were face-to-face meetings. One week before the interview, the list of questions going to be asked in the interview was sent to the interviewees. This is to allow interviewees to have enough time to think and prepare so that the interviews would be compact and complete.

The interviews were designed to last for two hours so that all the important points could be covered and still fit into the expert's busy schedule. The interviews were conducted by the author with the support of an independent colleague; the author asked questions to the interviewee while the assistant took notes. In case of ambiguity or incomplete information, the assistant also asked questions to clarify. The advantages of having two interviewers are that the second interviewer can focus on what is said, ask follow-up questions and aid the primary interviewer when necessary and the probability of understanding the subject correctly can be increased by discussing and verifying the interpretation of the interview between two interviewers [37].

The interviews were recorded with the interviewees' consent and notes were taken during the interview in the event that recording equipment failed [38]. Moreover, taking note can also provide a real-time "sanity check" to discover aspects that need to be discussed further [39].

After the interview, the interview content was transcribed from the recording under summary form, not a verbatim transcription. The note and transcription was then compared to double-check the consistency as well as to avoid losing information. The summary of the interview result was sent back to the interviewee for review of its correctness before it was integrated with other results.

3.1.4.1 Interview questions

The interview questions aim to uncover improvement areas in the model in terms of completeness, correctness, applicability. They contain big general questions (leading questions) followed by one or more smaller specific questions that delve deeply into certain aspects or actions. In order to ensure the quality of the questions, they were reviewed by both the researcher and Dr. Mikael Svahnberg (Advisor). The interview question list can be found in the following Appendix A.

3.1.5 Piloting

The purpose of piloting the interview was to uncover the problems in the questionnaires and the interview process prior to the actual validation [40]. Mr. Michael Unterkalmsteiner, a PhD student in Software Engineering at BTH was invited to participate and act as an expert during the mock interview. He was provided with the model and the list of abstract questions to prepare before the mock interview so as to simulate the interview as close to the actual one as possible. Through the mock interview, the researcher intended to identify which questions were not clear for understanding, whether the question order was logical and easy to follow and whether the interview process was suitable. Besides, since the interview was

semi-structured, the pilot was also useful for us to train the researcher's interviewing skills in dealing with unprepared situations.

During the mock interview, some problematic questions were identified and needed rephrasing. Moreover, the researcher tried out different ways to ask about the same idea. The questions were also modified so that the desired information could be obtained. For example, instead of asking questions like "Do you feel the content of the model is clearly and adequately described?" which might trigger general answers which were less of interest, it was put in other words as "Can you point out 5 actions that are short of information?". In this way, the answers obtained were more specific and useful to detect the actions that required better elaboration. Moreover, through the discussion in the mock interview, additional questions were created to delve into difficult yet overlooked points such as "Satisfied/Explained" option or "Optional group" concept. On top of that, Mr. Unterkalmsteiner also provided some interesting ideas which could be considered as good hints on how to improve the usability of the model in the next step.

3.2 Static Validation Execution

Overall in this step, there were seven interviews conducted, two of which were face-to-face meetings, another two were performed through landline and the rest were through Skype. In the face-to-face meetings, it was easier for the experts to follow the interviewer when moving around different points of the model. Hence, more information and suggestions were discussed as communicating in a rich channel enabled faster and better common understanding between the parties about the model. On the other hand, there was no difference between interviews through landline and Skype.
Most of the time, only one interview session was done with each expert. However, it was an exception with Dr. Samuel Fricker in which a follow-up interview was scheduled one week after the main one.

Regarding the duration of the interview sessions, the average length is one hour with an exception case of Dr. Christof Ebert which lasted only 15 minutes due to his busy schedule. Regarding the model review duration, some experts spent as much as three hours on the model. On average the experts spent 1 hour 30 minutes in reading and reviewing the model.
The interviews followed closely the static validation design. The content of the question list used in each interview was more or less similar. The only exception was the interview with Dr. Christof Ebert. Due to the short time restriction, ten most doubtful questions were asked.

The detailed view of the interview duration and the model review duration is shown in Table 4. In order to easily refer to expert's comments, an ID was assigned to each expert, for example E1 for Dr. Inge Van de Weerd.

Table 4. Interview and Model Review Durations

ID	Expert name	Interview Duration	Model Review Duration
E1	Inge Van de Weerd	55 mins	1 hour
E2	Samuel Fricker	1 hour 35 mins + 15 mins	45 mins
E3	Krzysztof Wnuk	1 hour	1 hour
E4	Jürgen Börstler	1 hour 20 mins	2 hours
E5	Richard Svensson	1 hour 30 mins	3 hours
E6	Christof Ebert	15 mins	Unknown
E7	Kristian Sandahl	1 hour 25 mins	1 hour

3.3 Static Validation Results

Throughout the interviews, expert's opinions regarding the correctness, completeness and applicability of the model were extracted. Moreover, the experts were also asked about the model structure, the maturity levels as well as its detailed actions. Based on the individual opinions of each expert above, the suggestions for improvement were consolidated. These suggestions were then categorized into different aspects as shown in table 5. Similar opinions regarding the same issue were also grouped. In total, there were 65 suggestions obtained from the experts. The suggestions were found to be equally dispersed among all the parts of the model (for example the model structure, maturity level, specific MPA etc.), which meant that no area was overlooked.

Table 5. Suggestion summary

Correctness	Completeness	Applicability	Others
23 suggestions	19 suggestions	2 suggestions	21 suggestions

The breakdown of the suggestions is shown in the following sections.

3.3.1 Regarding the structure of the model

Suggestion ID	Expert	Problem description
SG1	E2, E4, E7	The concept "Optional group" is confusing as the so-called optional group actions in Quality Assurance are not mutually-exclusive.
SG2	E2, E3, E5, E7	The concept "Satisfied/Explained" is not easy to understand.
SG3	E3	The 7 Main Process Areas (MPAs) seem to be different to what people usually know. Hence, it would be clearer to mention the rationale for categorize it this way. For example: Release planning and Requirements Documentation and Specification is part of Requirements Process Management.
SG4	E1	"Supporting Action(s)" is not clear for readers. This relation should be explained in more details.
SG5	E5	It is hard to find certain activities in the model.
SG6	E4	The figure of structure can be changed with fully present MPAs and SPAs.
SG7	E2	The order of the actions in the model needs to be refined.

3.3.2 Regarding the maturity level structure of the model

Suggestion ID	Expert	Problem description
SG8	E1, E2	The names of the levels should be changed to be more neutral. The current name implies that companies should always aim for the highest level, which is not the real intent.

3.3.3 Regarding the details of the model content

Suggestion ID	Expert	Problem description
SG9	E4	A common template should be used for each action, in which "name", "what it is", "how to do" and "why" are explicitly mentioned.
SG10	E4	Actions which are not part of SPA should be placed under a SPA named "General Actions".
SG11	E1	It is more convincing to add links from the model to literature.

3.3.4 Regarding the specific area "OS Organizational Support"

Suggestion ID	Expert	Problem description
SG12	E4	"OS.a1 Assign owner of Requirements process" should belong to SPA "OS.RR Roles and Responsibilities" because it is more related to roles and responsibilities.
SG13	E1	In the action "OS.RR.a1 Define Roles and Responsibilities for Requirements Engineering Process", the roles are not described in detail, there should be part mentioning about "documenting in a central place and letting everyone knows where to access".
SG14	E7	Regarding "OS.RR.a1 Define Roles and Responsibilities for Requirements Engineering process", it should have a generic view and let companies define the roles.
SG15	E2	The Sub-process area (SPA) "OS.S Strategic" should be moved to a separated MPA.
SG16	E1	The actions in Sub-process area (SPA) "OS.S Strategic" should be renamed in order to highlight the supporting purpose of them to RE.
SG17	E5	The 3 actions in SPA "OS.S Strategic" are overlapped with one another.
SG18	E4	"OS.S.a1 Define product strategies" and "OS.S.a2 Define Product roadmap" should be moved under RP Release Planning.
SG19	E1	The action "OS.S.a1 Define Product Strategies" should also specify about documenting in a central place, and explain more about how often this action should be performed.
SG20	E4	"PM.a3 Train personnel in Requirements Management Process and Specialty" should be moved to MPA "OS Organizational Support".
SG21	E4	"PM.a5 Early connect portfolio connection into requirements engineering process" should be moved to "OS Organizational Support" and action PM.a5 should be placed under "Strategic".
SG22	E5	"OS.S.a4 Communicate strategies in Organization" should be at level 1 since it was very essential (communication is a big problem!).
SG23	E1	Regarding the specific area "OS Organizational Support", additional information should be considered in order to explain how to create product definition and win-lost analysis with Boston Matrix, but they might be too detailed for the model.
SG24	E7	There should be an action for competitor analysis to identify the strength and weakness of companies compared to competitors in MD in order to know where to focus the effort on.
SG25	E7	The action "OS.S.a1 Define organizational strategies" was not well-recognized in industry and should be renamed to "portfolio management".

3.3.5 Regarding the specific area "PM Requirements Process Managements"

Suggestion ID	Expert	Problem description
SG26	E2	The supporting action of "PM.RC.a1 Establish effective communication with requirements issuers" is too far related.
SG27	E2	The actions "PM.a1 Introduce Tool Support for Requirements

		Engineering" and "PM.a2 Define and maintain the Requirements Management process" should be swapped due to the common sense of implementation.
SG28	E4	In the action "PM.RC.a1 Establish effective communication with requirements issuers", there are many more stakeholders than requirements issuers. The communications with issuers should be at level 1 and another action should be added to communicate with all stakeholders at level 2.
SG29	E2	"Involving teams" in action "PM.RC.a2" should be changed into "involving roles" in order to include different involved stakeholders.
SG30	E4	The action "PM.a2 Define and maintain the Requirements Management process" should be the supporting action for SPA "PM.RT Requirements Traceability".
SG31	E4	"RA.a4 Analyze Requirements Functional Dependencies" supports traceability activities; hence it should also be added as Supporting Actions for the SPA "PM.RT Requirements Traceability".
SG32	E5	"PM.CM.a1 Manage versions of requirements" and "PM.CM.2 Baseline requirements" should be merged together.
SG33	E1	Action "Document centrally all the requirements, stakeholders should have access to that" should be added.
SG34	E2	Actions should be added to mention that managers need to follow up with the teams regularly review and give formal acceptance.

3.3.6 Regarding the specific area "RE Requirements Elicitation"

Suggestion ID	Expert	Problem description
SG35	E6	Regarding the action "RE.SI.a3 Distinguish between Customers, End-Users and In-house Stakeholders", the stakeholders should be more generic and not limited to only 3 types as above
SG36	E7	Customer-customer stakeholder (e.g. customer of a banking system) should be added in "RE.SI.a2 Distinguish between Customers, End-Users, and In-house Stakeholders".
SG37	E4, E5	"RE.EP.a2 Consider quality requirements" should be at level 1 and action "Qualify and quantify quality requirements" should be placed at level 2.
SG38	E6	In SPA "Domain Consideration and Knowledge", the verb "Consider" is not an assessment term and should be substituted by a more concrete one.
SG39	E1	The action "RE.EP.a1 Adapt elicitation technique according to situations" is quite obvious and may need to be renamed to express the correct intent.
SG40	E5	"RE.EP.a3 Create Artifacts to facilitate Elicitation and Analysis" is confusing as the artifacts are created in Analysis but can be used in Elicitation to elicit new requirements.
SG41	E4	"RE.EP.a7 Reuse Requirements" is one channel for the requirements sources and should be merged with "RE.EP.a6 Create Elicitation Channels for Requirements Sources".
SG42	E1	The name of the action "RE.DC.a3 Consider co-existing business process" is a bit confusing.

3.3.7 Regarding the specific area "RA Requirements Analysis"

Suggestion ID	Expert	Problem description
SG43	E4	The description of "RA.a7 Perform refinement and abstraction of requirements" is too detailed.
SG44	E5	There should also be more prioritization techniques recommended to the practitioners, e.g. Cost-value approach, Focus-point, etc.
SG45	E5	The negative and positive impacts of ICOST and CVALUE relation should be clearly specified in "RA.a5 Analyze Value-related Dependencies between Requirements".
SG46	E2	Requirements Negotiation should be placed together with Requirements Communication instead of included in Analysis.
SG47	E2	The activities of specifying solution by modeling the context/behavior, etc. should be placed under Requirements Analysis instead of Requirements Elicitation.
SG48	E3	"RA.a3 Perform Requirements Risk Analysis" is more applicable in industry, hence should be placed under level 1 and action "RA.a2 Perform Systematic Requirements Prioritization at Project-level" should be placed under level 2.
SG49	E6	The MPA "Requirements Analysis" should consider 3 different dimensions: Quality attributes, Solution/ problems model and Cost-benefit analysis.
SG50	E4	Requirements Analysis activities and Quality assurance activities may overlap hence they both aim to ensure the quality of the requirements.
SG51	E5	In "RA.a4 Analyze Requirements Functional Dependencies", the phrase "functional dependencies" may make readers confused. It can be simply "dependencies".
SG52	E5	One more Action should be added: Analyze potential relations between functional and non-functional requirements. It should be at level 3.
SG53	E2	The actions in the model should be described clearly for 3 situations: Product development, Project development and Contract development.

3.3.8 Regarding the specific area "RP Release planning"

Suggestion ID	Expert	Problem description
SG54	E5	There should be actions to perform Cost and Value Estimations prior to Prioritization.
SG55	E1	Regarding the action "RP.a4 Involve different perspectives in release planning", the external customers should also be involved into the process.
SG56	E2	The two actions "RP.S.a2 Perform Requirements Prioritization at Pre-project level based on value, cost and effort" and "RP.S.a3 Consider additional advantageous dimensions for prioritization" should be merged because in many cases, they do not use cost and value but perform prioritization based on interdependencies.

3.3.9 Regarding the specific area "QA Quality Assurance"

Suggestion ID	Expert	Problem description
SG57	E1, E6, E7	This MPA should be renamed to "Requirements Validation" while E2, E3 suggested to keep the name.

SG58	E4	The actions "QA.a1 Use checklist to ensure quality of Requirements" and "QA.a2 Review requirements" are pre-requisites of "QA.a4 Organize Inspections to ensure quality requirements".
SG59	E7	"QA.a3 Create preliminary artifacts for Quality Assurance" name is not easy to understand.
SG60	E7	Regarding the specific area "QA Quality Assurance", incremental acceptance test should be included.

3.3.10 Others

Suggestion ID	Expert	Problem description
SG61	E5	The model should consider 3 different types of requirements: raw requirements, product requirements (functional, non-functional, business), and component requirements.
SG62	E4	Percentage can be used to grade actions partially completed.
SG63	E3	There should be sequence for implementation apart from improvement suggestion. For instance, there can be guidelines for process at 1*, 2*and 3*.
SG64	E3, E4	Clearly state which actions are applicable for bespoke or MDRE.
SG65	E1	An automatic tool support should be introduced together with the model; so that the practitioners can easily deploy the assessment.

3.4 Response Action and Model Improvement

After the suggestions were consolidated, they were considered carefully by the researcher. Some suggestions were analyzed together as they were related to the same issue or represented conflicting opinions. The suggestions went through an analysis process shown in Figure 7. Firstly, the classified suggestions according to aspects were validated for their relevance to the model. For correctness suggestions, they were checked against the literature. Regarding completeness suggestions, the scope and lightweight aspect of the model were the main concerns in judging whether the suggestions should be implemented. As for the rest of the suggestions, the suggestions were checked if they were beneficial to the model and if omitting them would make severe adverse impact on the model. Besides relying on the literature to analyze the suggestion, in certain cases, the research also consulted with Dr. Mikael Svahnberg (Advisor) and Dr. Tony Gorschek (Examiner).

After this first step, those suggestions that were deemed irrelevant were dismissed. For the rest of the suggestions, resource and time was dedicated to implement suggestion related to completeness, correctness and applicability first as they affected the model quality more than others. Regarding the "Others" suggestions, they were sorted based on the estimated resource and time taken to implement. If no resource and time available, the suggestion would be left as future work.

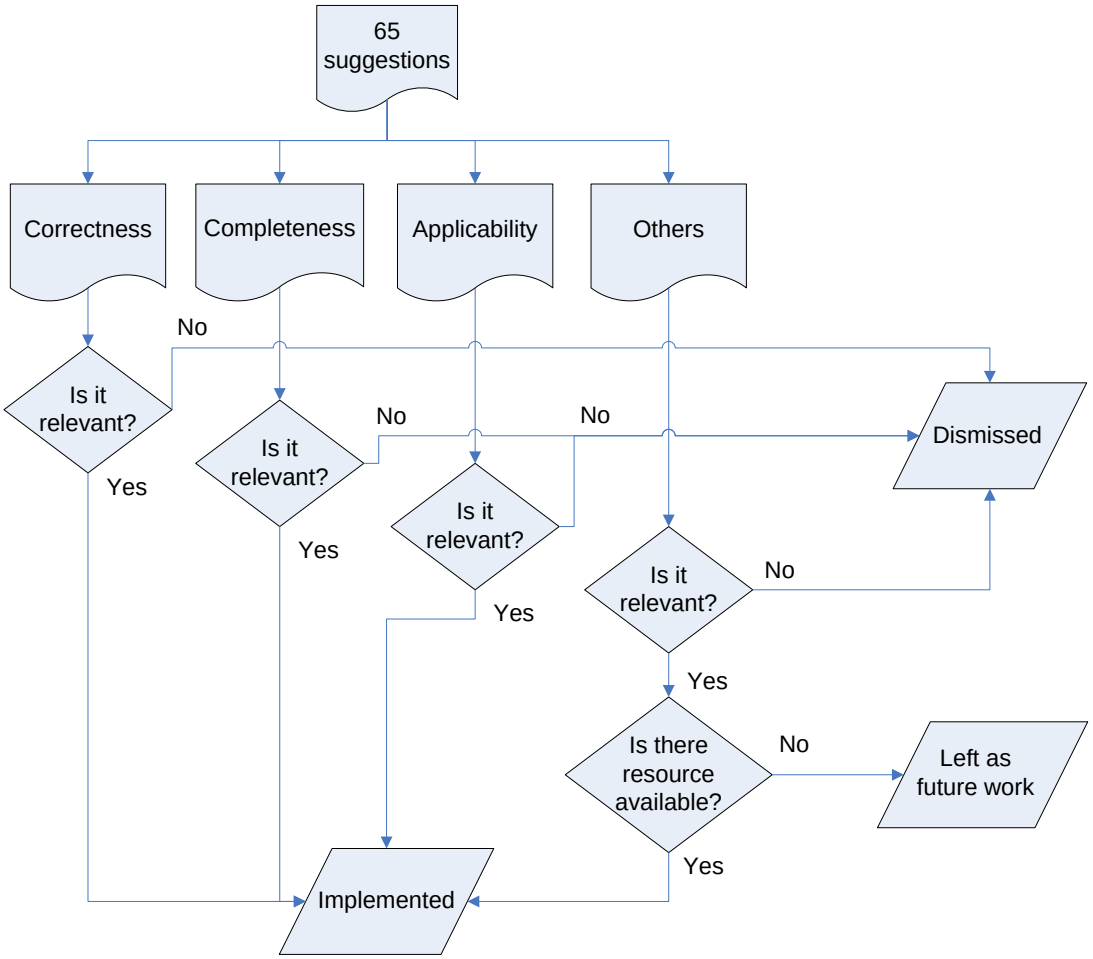


Figure 7. Suggestion analysis flowchart

Keeping in mind the goal of the model and the resources, the researcher tried to accommodate as much feedback as possible. As a result, 47 out of 65 suggestions were implemented. In two cases the researcher postponed the implementation of the suggestions due to restriction of time and effort while ensuring no severe adverse impact on the model. On the other hand, 16 suggestions were deemed irrelevant after serious consideration. Table 6 presents the suggestion classification based on the response action.

Table 6. Suggestion classification in terms of response action and type

Suggestion Type / Response action	Correctness	Completeness	Applicability	Others	Total
Implemented	20	12	0	15	47
Left as future work	0	0	0	2	2
Dismissed	3	7	2	4	16
Total	23	19	2	21	65

The response actions and model improvements in correspondence to the suggestions are presented in detail below.

3.4.1 Regarding the structure of the model

The concept “Optional Group” was re-checked and the researcher agreed that these actions complemented instead of substituting one another. Hence, this concept was removed in the structure and the actions in Quality Assurance were treated as other actions [SG1]. The “Satisfied/explained” option was also changed to “Inapplicable” since this word better presents the intended idea and most people understand it without further explanation [SG2].

Having the relationship between the MPAs and the Requirements Engineering activities mentioned in the literature would be beneficial to readers to understand the model better. Therefore, additional information was included in the MPAs [SG3]. Besides, the description of "Supporting Action" was rewritten so that it would be clearer for understanding [SG4]. The Process Area view of the model was also rearranged to appear before the model description to give a good picture of the whole model. Moreover, it is also updated with page number for each action so that reader could easily locate them in the model [SG5]. Apart from that, a new figure with a more detailed presentation of MPAs and SPAs were added in the new version of the model in order to better illustrate it [SG6]. As part of the attempt to improve the understandability of the model, all the actions in each SPA were considered and re-organized so that actions of lower levels appeared before those of higher level. This order however does not imply the implementation order in the process [SG7].

3.4.2 Regarding the maturity level structure of the model

The researcher decided to make use of more neutral words for the names of three levels: "basic", "intermediate" and "advanced". With these new names, there is no implication that companies have to aim to the highest level to be able to function well [SG8].

3.4.3 Regarding the details of the model content

Regarding the suggestion on having a common template for each action, it would be very advantageous if the idea were implemented. However, in the scope of this research, there was not enough literature resource and time to describe it as detailed as the expert recommended. Hence, this improvement can be realized in the next iteration of the model [SG9].

In order to ensure the consistency of the model, the current actions residing directly under the MPAs are now put under the corresponding SPA "General Actions" in the new version of Uni-REPM [SG10].

Besides, the link to literature was added in the new version of the model since it would help readers to find necessary documents which supported them in implementing the actions [SG11].

3.4.4 Regarding the specific area "OS Organizational Support"

The action "OS.a1 Assign owner of Requirements process" became the first action in "OS.RR Roles and Responsibilities" and its ID was renamed to "OS.RR.a1 Assign owner of Requirements process" as suggested. The rest of the actions in that SPA were also renamed their IDs to reflect the change [SG12].

Regarding the comments [SG13 - SG14], as different companies have different suitable roles needed for their operations, it is impossible to come up with the exhaustive lists of possible roles used in companies. Moreover, the same role can be called by different names and have different responsibilities. Therefore, the roles are kept as generic as it currently is. However, the part "documenting in a central place and letting everyone knows where to access" was added in the description of the action as suggested.

The researcher also considered how much the MPA "Organizational Support" was related to "Product Management" area. Through researching, it was found that this area was huge and deserved a whole study of its own. There already existed a detailed process assessment model created by Weerd [41]. In order to avoid overlapping effort, the scope of the model was re-defined and only covered product management to the extent of a supporting activity of Requirements Engineering process. Consequently, this SPA was kept the same under MPA "OS Organizational Support" [SG15 - SG16]. Apart from that, all the actions in this SPA were rechecked and they were somehow overlapped. In order to improve, they were benchmarked with different publications about Product Management. However, the researcher found out that the knowledge in this area was very fragmented and there was a

lack of common concepts and terminologies. Therefore, the two actions "OS.S.a1 Define product strategies", "OS.S.a2 Define Product roadmap" were rewritten and "OS.S.a3 Define Organizational Strategies" was removed as it was covered by the above actions [SG17].

Regarding the comments about actions related to product roadmap being moved to "Release Planning" process area [SG18], although these two actions support the release planning activity, they also support actions in other areas such as Elicitation and Analysis. Hence, because of their wide coverage, the researcher decided to keep them as they are. Besides, more detailed information was added in the description of these actions in order to specify clearly the work product of these activities [SG19].

In response to the comment that training is one of the activities done on the organizational level and not Requirements Process Management, this action was then moved to MPA "OS Organizational Support" and placed under SPA "General Actions" [SG20].

The action "PM.a5 Early connect portfolio connection into requirements engineering process" was removed because it overlapped with several other actions "RE.EP.a4 Let Business Concerns/Product Strategies guide Focus of Elicitation Efforts" in MPA "RE Requirements Elicitation" and "RP.a1 Synchronize Release Plan with Product Roadmap" in MPA "RP Release Planning" [SG21].

Besides, the researcher agreed that "OS.S.a4 Communicate strategies in Organization" is a weak point in organizations. However, this action has to be done after the other two actions "OS.S.a1 Define product strategies", "OS.S.a2 Define Product roadmap" which are both at level 2. Hence, it has to be at least level 2. The researcher decided to keep it at level 3 because it is beneficial but not very essential to companies to implement [SG22].

Regarding the actions related to competitor analysis, win-lost analysis, etc., the researcher is aware that they belong to the Product Management process and can be used to support the product strategy definition activity. However, as the model scope was already redefined and only covered product management to the extent of a supporting activity of Requirements Engineering process, it was not necessary to add these recommended actions in the model [SG23 - SG24]. Since the SPA "OS.S Strategic" was rewritten to avoid overlap of actions, the action "OS.S.a1 Define organizational strategies" was removed. Hence, there is no need to rename it [SG25].

3.4.5 Regarding the specific area "PM Requirements Process Managements"

Taking the recommendations [SG26] into account, more detailed support such as "implementation proposal" or "prototype", "use rich communication channel" were added in the improved model so that the guideline purpose of the model would function more effectively.

Regarding the comment [SG27] about the order of the actions presented in the model, although there is no intention about the implementation order in the model, it was decided to swap "PM.a1 Introduce Tool Support for Requirements Engineering" and "PM.a2 Define and maintain the Requirements Management process" in the improved version to make the model easier for readers to understand.

The action "PM.RC.a1 Establish effective communication with requirements issuers" remained the same as in the old version of the model and action "PM.RC.a2 Obtain common understanding of requirements among different involving teams" was changed to "PM.RC.a2 Obtain common understanding of requirements among different involving roles". By this improvement, all the communications within requirements process are covered sufficiently [SG28-SG29].

The suggestions [SG30-SG31] were also directly implemented. In the new version of the model, the action to define procedure and roles impact was Supporting Action for the SPA "PM.RT Requirements Traceability". Action "RA.a4 Analyze Requirements Functional Dependencies" was added as Supporting Actions for the SPA since the relation between requirements was one of the important factors affecting traceability.

The Actions "PM.CM.a1 Manage versions of requirements" and "PM.CM.2 Baseline requirements" were not merged together as suggested. The reason is that small companies do not make use of versions control but after sometime, the set of requirements can still be reviewed, approved and fixed as a baseline. Therefore, the expert's idea was not implemented [SG32]. As action in comment [SG33] "Document centrally all the requirements, stakeholders should have access to that" was covered by activities in MPA "Requirements Specifications and Documentation", it was not chosen for the improvement.

Finally, the suggested action in comment [SG34] that managers need to follow up with the teams regularly review and give formal acceptance is a very important one. However, this is more about general management than Requirements engineering process specifically. Hence, this idea was not implemented, either.

3.4.6 Regarding the specific area "RE Requirements Elicitation"

The initial intent of action "RE.SI.a3 Distinguish between Customers, End-Users and In-house Stakeholders" was to emphasize the importance and different expectations of these three common types of stakeholders. However, E6's argument is valid as there are many more types e.g. partners, distributors etc. Therefore, the actions were rewritten and renamed to "Distinguish between different types of stakeholders" [SG35 - SG36].

These two actions "RE.EP.a2 Consider quality requirements" and "RE.EP.a5 Qualify and quantify quality requirements" were also pushed to the lower level to emphasize their importance taking into account the current situation that companies tend to overlook quality requirements and have to pay a high price later [SG37].

In SPA "Domain Consideration and Knowledge", action name starting with "Consider" was changed to "Elicit information about" since that word is quite abstract [SG38].

It was also agreed that the current name of the action "RE.EP.a1 Adapt elicitation technique according to situations" did not reflect the intent correctly. The intent is to suggest companies to use different techniques in different situations because in one circumstance, certain techniques are more suitable than others. Therefore, the action was renamed to "Use appropriate elicitation techniques according to situations" [SG39].

Regarding comment [SG40], to avoid the confusion, the action "RE.EP.a3 Create Artifacts to facilitate Elicitation and Analysis" was renamed to "Use artifacts to facilitate Elicitation" and new actions in Analysis were added to take care of the creation part.

As reusing requirements was one channel for the requirements sources, that information was added in the description of the action "RE.EP.a7 Reuse Requirements" following the suggestion of E4 [SG41]. The reason for not merging it with "RE.EP.a6 Create Elicitation Channels for Requirements Sources" is to emphasize the importance of reusing requirements because it can help to reduce cost, effort and improve the requirements quality.

Finally, the action "RE.DC.a3 Consider co-existing business process" was renamed to "Elicit information about System's Operation Domain" [SG42].

3.4.7 Regarding the specific area "RA Requirements Analysis"

The action "RA.a7 Perform refinement and abstraction of requirements" from the comment [SG43] was considered and the researcher agreed with the expert that the example provided in the action description was too detailed, hence might make the readers confused. In the improved version of the model, this information was moved to the "Examples" section.

Regarding the comment [SG44], although the intention on recommendations section is not to provide a full list of available solutions to implement certain action, the researcher agreed with the expert to add his suggested techniques to this section. The reason under this change is that prioritization is a complex task and varies in different companies. Hence, the guideline function of the model would be more advantageous if the implementing solution for this task is covered adequately. It was also agreed that it would be much clearer and more accurate to describe the impacts of these relations of ICOST and CVALUE in detail [SG45]. Hence, this idea was implemented for the new version of the model. Besides, Requirements Negotiation was also moved to Requirements Communication instead of included in Analysis according to comment [SG46].

The modeling actions were moved to "Requirements Analysis" MPA and split into 2 actions: Prototyping and System modeling. The reason was that initially, modeling was considered as a tool for elicitation (scenarios elicitation…). However, taking into account the idea in [SG47] that modeling is actually analysis activity by which problems and solutions are modeled, the suggestion of the expert was implemented in the improved model. The maturity levels of the two actions "RA.a2 Perform Systematic Requirements Prioritization at Project-level" and "RA.a3 Perform Requirements Risk Analysis were also changed according to the comment [SG48].

The MPA Requirements Analysis was divided into 3 SPAs in correspondence to [SG49]. The first SPA "Requirements quality attribute analysis" covered activities to detect incomplete, incorrect and untestable requirements. The second SPA "Problems and Solution modeling" covered activities to model and demonstrate problems and solutions. The last SPA covered all activities to estimate cost-benefit, priorities and interdependencies among requirements

Regarding [SG50], this issue was solved in the MPA "Requirements Validation" below.
Regarding the recommendation about the phrase "functional dependencies" [SG51], as this phrase may cause confuse to the readers, two types of dependencies were merged into one action called "RA.GA.a3 Analyze Requirements Relations".

Concerning the comment [SG52] about analyzing potential relations between functional and non-functional requirements, it was decide not to introduce a separated action for this purpose since this relation is merely one type of the requirements interdependencies which are stated in the model. The suggestion [SG53] was also not implemented. The reason for this exclusion is that according to some practitioners, they rarely knew whether their environment was "Product development", "Project development" or "Contract development". This view might be clear in academia but not in industry. Hence, these categories were not introduced in the model.

3.4.8 Regarding the specific area "RP Release planning"

As suggested in the comment [SG54] cost-value estimation in prioritization was not included in the proposed model, separated actions were added to perform these estimations.

The expert's opinion in [SG55] is right about the external customer involvement in Release Planning since they are one of the most important perspectives in release planning. Hence, in the improved version of the model, this perspective was added to the action description.

According to [SG56], these two actions "RP.S.a2 Perform Requirements Prioritization at Pre-project level based on value, cost and effort" and "RP.S.a3 Consider additional advantageous dimensions for prioritization" were also considered and merged. The reason of this change is that solutions for prioritization vary largely in companies, especially in prioritization factors. Hence, it may not be flexible to limit them into some fixed factors.

3.4.9 Regarding the specific area "QA Quality Assurance"

Regarding the recommendation [SG57], after checking this MPA and also the overlapping threat of some other actions such as "RA.a1 Analyze for missing, double, incomplete, ambiguous requirements", the MPA name was changed from "Quality assurance" to "Requirements Validation". Another reason for changing is that the existing actions in this MPA only covers the validation of requirements whereas "Quality Assurance" is broader and covers the compliance and improvement the whole RE process as well.

Since the "Optional Group" concept was removed and based on their lower levels, the actions "QA.a1 Use checklist to ensure quality of Requirements"(level 1) and "QA.a2 Review requirements"(level 2) were indeed be implemented before "QA.a4 Organize Inspections to ensure quality requirements"(level 3) [SG58].

Based on [SG59], the name of action "QA.a3 Create preliminary artifacts for Quality Assurance" was also changed to a clearer and more concrete one "Develop Preliminary Test Case or User Manual" to improve the understandability of the model. The suggestion [SG60] was also implemented by adding the action "Define Acceptance Criteria and Acceptance Tests" since having customers define acceptance test is one effective technique of requirements validation. The earlier the user writes acceptance test, the sooner defects in requirements and products are detected.

3.4.10 Others

Regarding the recommendation [SG61], it is acknowledged that there are different types of requirements according to sources, purpose or characteristics. Depending on their needs, different organizations will define different types of requirements as well as the level of details each type should have. Hence, it is hard to specify all types of requirements available as well as which actions associated with each type. The researcher opted not to implement this suggestion in order to keep the model more generic. However, these types of requirements were added in the description of the action "RA.a7 Perform Refinement and Abstraction of Requirements" as it was related to breaking down and working up requirements at different levels of abstraction.

It was a good idea using percentage to grade actions partially completed as it would give more accurate result when judging how many percent of a particular action the company has completed [SG62]. However, this approach lost the overview when actions are judged altogether. As the numbers of actions in each level are not equal, with the same percentage of incomplete actions, the real number of actions in each level need to be fulfilled can be different. For example, in level 1 that would mean 2 actions while in level 2 would mean 5 actions. Due to that disadvantage, the researcher opted not to change the model in this version.

As opposed to [SG63], it is not practical to impose a pre-defined implementation order on the practices of the model. Depending on the company characteristic and situations, each company will have its own process that works best for them. There is no one-size-fit-all or perfect process. Hence, the purpose of this model is to present all the good practices that give company ideas to improve. However, it is the company responsibility to decide whether the recommended practices are indeed beneficial and suitable and when to implement in their situations.

The researcher agreed with [SG64] that there are some actions only applicable to one development environment; for example, "OS.S.a1 Define product strategies" is only applicable in the market-driven settings. However, as there is no clear-cut border between these environments, it is hard to determine which one the actions belong to. Even for the companies, they may not be able to identify clearly which environment they operate in. Therefore, it is undesirable to specify clearly the actions for the two situations. Instead, the company will be asked all the actions and decide for themselves whether the actions are applicable. If the actions are inapplicable, it will be assigned "Inapplicable" and the reason will be noted down.

Finally, the suggestion [SG65] about an automatic tool support is a great idea to implement and it would help reducing time and effort in assessing the maturity. However, due to the time and resource constraint, the researcher was unable to realize in this version. It will be considered in the next iteration of the model.

3.5 Conclusion

Planning, preparing and executing the static validation required extensive effort to analyze and determine among many alternative approaches. The researcher learnt a lot from selecting, contacting experts to conducting interviews through face-to-face meeting and telephone. Looking back at the whole static validation process, the careful design step was crucial and beneficial to ensure that nothing important was missing and no error was made. The execution went smoothly as planned as interviews were kept in the reasonable timeframe (one and a half hour). The researcher also learnt to always opt for a face-to-face meeting when possible, as it was the most effective and efficient way of communication, especially when the topic was new or complex to both parties.

The aim of the static validation is to identify possible improvements that can be done to the Uni-REPM model. And by looking at the huge results obtained from the interviews, the researcher was confident to say that this goal was successfully achieved. With the help and feedback from seven experts coming from various countries (The Nederland, Switzerland, Sweden and Germany) with diversified expertise, numerous good advices were gathered on how to improve the model and all aspects of the model were covered adequately.

Based on the feedback, the correctness, completeness and applicability of the model were scrutinized. The majority of the suggestions were implemented in an improved version of the model. By working on these suggestions, the correctness of the model was enhanced greatly. Moreover, the improved model was fairly complete as there were few suggestions about adding new actions. According to the experts' opinions, all of the actions in the model were applicable in real settings. However, the model has to be validated in the industry in order to confirm the validity of its applicability. Besides the above three aspects, other suggestions were also analyzed and acted on to improve the overall quality of the model.

Aside from lots of suggestions for improvement gathered in the interviews, the model also received good compliments for its well-written structure and description. However, as the purpose of the static validation was to uncover potential problems, the interview questions and results did not focus on the good points of the model. Through the static validation, weak points of the model were identified and then worked on. Hence, the improved model produced after this step has significantly better quality in terms of correctness, completeness and applicability and is suitable to be used to assess Requirements Engineering processes in companies.

4 DYNAMIC VALIDATION

In this chapter, the second round of Uni-REPM evaluation is presented. The objective of the step is to validate the applicability, usability and usefulness of the Uni-REPM model in industrial settings. Unlike the previous chapter in which direct questions about the model were asked, at this stage, the model and the checklist was used to assess the RE process in industrial projects. Hence, information about the model and checklist was obtained indirectly through the project evaluation session. After the project evaluation, a general discussion was brought up in order to capture practitioner's feedback of the session and the model usage.

This validation targets to evaluate Uni-REPM model in terms of the following aspects:

- *Applicability*: the degree to which the model and checklist can be applied in projects with different development environments (bespoke, market-driven or both). By using them in real projects, the results showed how well the model fitted in reality.
- *Usability:* The ease with which a user can learn to operate, prepare inputs for, and interpret outputs of a system or component [35]. In this case, the usability of the model is assessed by :
    - Efficiency: how much time was required for the practitioner to use the checklist and model to assess the RE process maturity.
    - Understandability: how easy it was to understand and answer the checklist correctly. The researcher looked out for "misunderstanding signs" during the project evaluation to detect ambiguous questions or actions.
    - Satisfaction: how pleasant the practitioner felt about the checklist, model and whole validation session.
- *Usefulness:* In this validation, the usefulness of the model and checklist is judged by obtaining the RE process maturity of the projects. The current state of the process is also shown with its strength and weaknesses. Although the recommended improvements are shown as well, their usefulness cannot be assessed as the results has not been released back to the organization. Moreover, it takes time to apply those suggestions and observe their impact. Hence, this perspective was planned as future work.

This chapter consists of three parts. The first part described how the validation was planned and executed. The second part presented the evaluation result for each of the projects followed by the detailed analysis. Then the improvements made on the model and conclusions about the whole dynamic validation were shown in the last part.

4.1 Dynamic validation design

The evaluation process was performed as demonstrated in Figure 8.

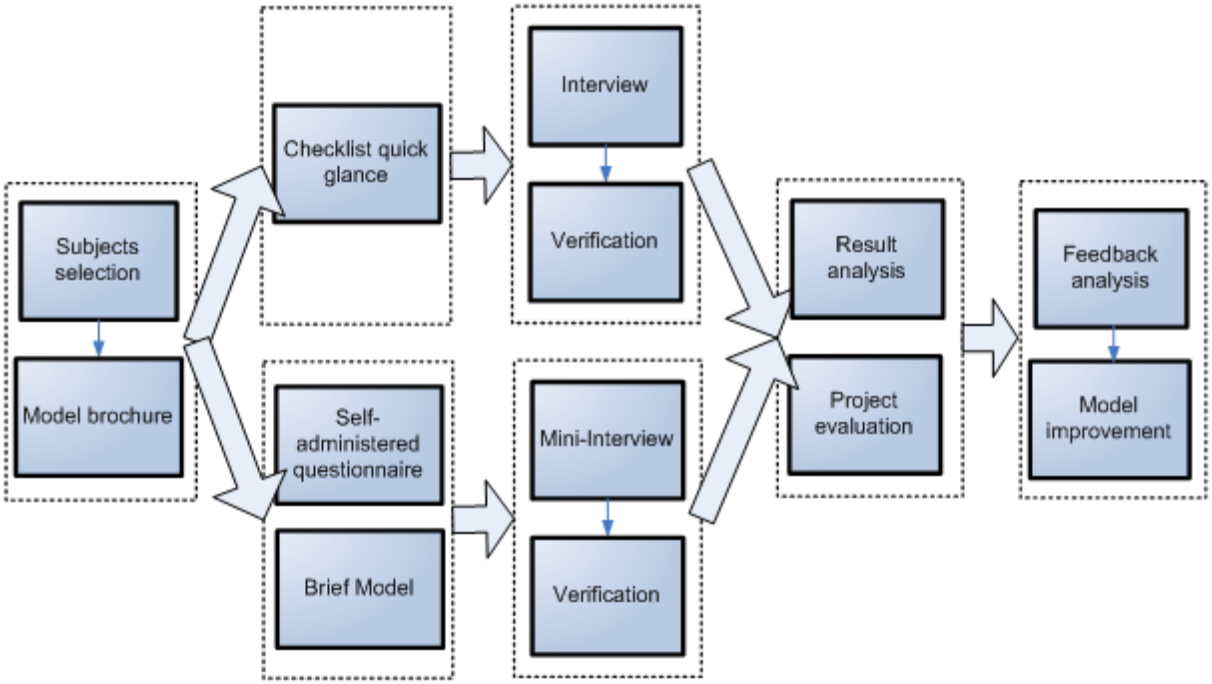


Figure 8. Dynamic validation process

At the first step, the subject were ***selected*** according to some defined criteria and invited to participate in the validation by emails. A ***brochure*** of Uni-REPM model was made and sent in the invitation showing a professional view about the work so as to urge the subject to participate.

After receiving the acceptance from practitioners, the researcher attempted to arrange an interview through Skype with the practitioner. In case this was successful, the ***checklist*** was sent to the subjects prior to the validation meeting through emails. The rationale for this step is that the subjects could spend some time to prepare in advance if they want to. Apart from that, as the checklist is quite long, having it in the interview helps the subjects to keep their interest and follow the interviewer. After that, the main validation step was conducted in the form of ***interviews***. The detailed design of the interview is described in the following section. The interview result was then summarized and sent back to the practitioner to ***verify*** its correctness and avoid misunderstanding.

In case the interview could not be arranged, a ***self-administered questionnaire*** was used as an alternative way. The practitioner was sent the questionnaire together with a ***brief version of the model***. He was expected to follow the instruction in the questionnaire, fill it up and return it to the researcher. After receiving the result, the researcher went over it. If needed, the researcher organized a ***mini-interview*** to clarify doubts and ***verify*** the understandability of the user.

Finally, the project evaluation ***result*** was analyzed to identify the process maturity level and ***feedback*** from subjects was considered in order to ***improve*** the model.

4.1.1 Subject selection

The objects of the validation are requirements engineering processes in companies. Therefore, only companies that have their own RE process are of interest. The subjects of the validation are person(s) involved in the requirements engineering process. They should participate in the project and are knowledgeable about the activities in the RE process. Moreover, as the Uni-REPM model was supposed to be used in all types of environments, no restriction was made on the type of projects being evaluated.

The subjects were asked to select a typical project to be evaluated. This was to give the subject the necessary openness to discuss freely in the interview and avoid giving out confidential information.

The goal is to validate the Uni-REPM on projects with various maturity levels and different development environments. The best situation is to have at least one project with bespoke development setting and one with MDRE setting so that all the actions in the model can be ascertained regarding their applicability. However, this was not guaranteed as the researcher did not know which companies would participate beforehand.

As the goal of the dynamic validation is not to compare the maturity levels of RE process between companies, the company identity was kept anonymous.

4.1.2 Model brochure

A brochure of the model was sent to the subjects along with the invitation to participate in the interview. This overview mainly provides the idea of the Uni-REPM model, its usage and its advantages over other existing models. The rationale for this step is that the subjects could spend their convenient time to explore the idea of the model and accept the invitation. The brochure can be found in Appendix B.

4.1.3 Interview

In this stage, interview was also used as main method for the validation. However, unlike the static validation step above, the interviews were structured in this case. The reason for this choice is that, in structured interview, the interviewer has a clear view of what information is sought in the interview, hence the questions are very specific [36]. This is suitable in this case where the information the researcher aims to extract is basically based on the Uni-REPM model.

As all of the companies involved in this validation are located far from the researcher's location, face-to-face meetings were infeasible due to time and resource constraints. Hence, all interviews were conducted by audio conference (Skype).

4.1.3.1 Interview questions

A list of fixed questions was constructed based on the improved version of the Uni-REPM model (resulted from static validation stage). This checklist is actually the direct transformation of the model into question form. For each action in the model, there is a corresponding question or group of questions to verify if the action is done or not. In order to ensure the quality of this question list, it was reviewed by the researcher and Dr. Mikael Svahnberg (Advisor) independently. Prior to the interview, the checklist was sent to the interviewees. This was to allow interviewees to have enough time to think and prepare so that the interviews would be compact and complete.

Besides this checklist, the interviewees were also asked several background questions in order to gather information about the interviewee himself, the company and the project context. These data are important because they affect the result validity and generalization. The background questions can be found in Appendix C. Moreover, at the conclusion of the interview, the interviewee's opinions about the whole interview were also collected.

4.1.3.2 Interview duration and participants

The interview sessions were estimated to last around 1.5 hours, in which the first five minutes was used for a short introduction and then followed by a structured interview. The interviews were conducted by the author with the support of one independent colleague; the researcher who asked questions to the interviewee and the assistant who took notes. In case of ambiguity or incomplete information, the note taker also asked questions to clarify. The advantages of having two interviewers are that the second interviewer can focus on what is

said, ask follow-up questions and aid the primary interviewer when necessary and the probability of understanding the subject correctly can be increased by discussing and verifying the interpretation of the interview between two interviewers [37].

During the interview, the interviewees were asked to "think aloud" when answering the questions. The idea is that, by this way the information on understandability of the questions list could be extracted as well as the "Inapplicable" reasons could be justified by both the interviewee and interviewer. Besides, the two interviewers also looked out for "misunderstanding signs" from interviewees. Some of those signs can be hesitation in answering questions or irrelevant answers. In case those problems occurred, the explanation from the model was provided to the readers to clarify. This is also a means to validate the model description.

The interviews were recorded with the interviewees' consent whenever possible and notes were taken during the interview in the event that recording equipment failed or recording was not feasible [38]. Moreover, taking note also provided a real-time "sanity check" to discover aspects that need to be discussed further [39].

After the interview, the interview content was transcribed from the recording under summary form, not a verbatim transcription. The note and transcription were then compared to double check the consistency as well as to avoid losing information. The summary of the interview result was then analyzed to assess the process maturity and identify suggestion for the RE process in the next step.

4.1.4 Self-administered questionnaire and brief model

The questionnaire content was almost similar to the ones used in the interview, with background questions and the model checklist. However, because the researcher could not monitor the process as in the interview, additional questions were created to capture the understandability of the model. In addition, a brief version of the model which consisted of only the process area view and the actions was also created to assist the user in understanding and filling in the checklist. The content of the questionnaire can be found in Appendix H.

4.1.5 Result analysis

In this step, the raw data obtained from the interview was analyzed against the actions in the model. The data were then processed to generate a final result. The final result including process maturity and MPA maturity information were sent to the interviewee together with the suggestion for improvement path thereafter. In addition, the interviewees were also provided with the full description of Uni-REPM model as a guideline for implementation.

4.1.6 Feedback analysis and Model improvement

The final step of this validation is to summarize the information extracted from the interviews and feedback to improve the models. The three main aspects of this validation, **Usability, Usefulness** and **Applicability**, were considered.

Usability - Efficiency
The time used for the validation session was recorded.
Usability - Understandability
All the information about misunderstanding and ambiguity was summarized and a root-cause analysis was performed on those. If the problems mostly came from different terminologies, the definitions would be considered for improvement.
Usability – Satisfaction
The feedback from the practitioners about the checklist, model and the validation session was recorded and analyzed to identify improvements.

Usefulness

After the process assessment, a general discussion was encouraged so that feedback of the practitioners on the usefulness of the model was recorded and analyzed.

Applicability
After the final results from all the cases were calculated, the model lag was checked. This information gave an overview of how well the model fitted in reality. If this gap was huge, an analysis on the population of the research would be performed to identify whether it was necessary to change some parts of the model.
In addition, other comments of the interviewees were also taken into account for improvement. In case additional practices (occurring from industries) were suggested, they would be considered in relation to a literature review.

4.2 Dynamic Validation Execution

In the first two weeks, Swedish IT companies were searched and twenty of them were chosen randomly regardless of their locations or sizes. Unfortunately, none of them accepted the interview. Due to the tight schedule, it was impossible to search and wait for random companies. Hence, the researcher opted for convenience sampling, i.e. personal contacts were used to arrange the interviews. However, all the above criteria are still satisfied. There were four companies participating in this validation: two from Singapore, one from Spain and the other from Denmark.

Overall in this step, there were three interviews conducted through audio conference and one self-administered questionnaire through emails. As the interviews were real-time communication, more information about the project, clarification and feedback were discussed. Hence, it was easier for the researcher to detect understandability problems as the researcher worked alongside with the interviewee and tackle the issues immediately. Regarding the duration of the interview sessions, the average length is one hour and a half. The interviews followed closely the dynamic validation design.

One self-administered questionnaire was completed by the practitioner alone in 40 minutes. The practitioner followed the instruction in the questionnaire, read the brief model once to familiarize with all the terms and then filled in the checklist. Because the data about understandability was restricted in this form, the researcher performed a mini-interview after receiving the result to verify the understandability of the practitioner about some "difficult" concepts. The "difficult" concepts were defined as questions that other interviewers stumbled on or concepts that might be confusing/unfamiliar. Besides, the researcher also randomly asked the practitioner about some actions to confirm the accuracy of the result obtained.

The list of participating companies is shown in Table 7 together with their descriptions.

Table 7. Dynamic validation participants

Company	Industry	Means of validation
Company 1	Software	Interview through Skype
Company 2	Software	Interview through Skype
Company 3	Insurance, Banking	Interview through Skype
Company 4	Insurance	Self-administered questionnaire

The first company operates in software industry and there are fifteen employees working in this business unit. Regarding the project under assessment, it involves customizing two different modules for a client, one for internal Invoicing and the other for Space Management. Both are related to facility management. The system is derived from a product

already developed in a MDRE context, and now being customized for the client. The project started four months ago and is still on-going. There are two employees in the team responsible for requirements elicitation, the project manager and project manager assistant.

The second company domain is software industry and the interviewee belongs to the engineering department business unit. Regarding the project under assessment, it involves programming the interface for the robot hardware. There are five team members working in seven months. The project is placed in the bespoke context as there is a specific customer.

The third company domain is in Insurance system, banking system and fund management system. There are around 500 employees working in the company. The project was to develop a system for a small insurance company in Singapore. The system was derived on a generic product developed by the company and new requirements were created by comparing the gap between the expectations of the customer and the existing functionality of the system. The new requirements were implemented by the customization and localization teams. The project lasted for one and a half year and the requirement team comprised three to six people.

The fourth company belongs to the Insurance domain with 800 staff and the IT business unit has 200 employees. The product of the project under evaluation is a Point of Sale system. The product is a system that encapsulates information of insurance product, receives information of clients and the insurance product they choose to buy, and generates benefit illustrations for clients. The system should also have the ability to generate risk profiles of clients based on the fact finding information obtained from them. This is an in-house system and the requirements come from user. The system is used across Singapore, Indonesia, Vietnam, Malaysia, China & Brunei. The project lasts for one year and involved one requirements engineering team of 10 people.

4.3 Dynamic Validation Result

4.3.1 Applicability

In order to have a complete view of the applicability of Uni-REPM, the following information was analyzed:

- The model lag of the whole processes gathered from interviews
- The most common inapplicable actions of the model

The concept “model lag” has been introduced in the design of the model to evaluate the applicability of the model. The idea under this concept is to explore how suitable the model can be in real projects.

4.3.1.1 Model lag summary

The proportion of inapplicable actions (among all actions) is summarized in the following Table 9.

Table 8. Model lag summary

Project name	Model lag percentage (without RP actions)	Total model lag percentage
Project 1	Inapplicable	12% (9 out of 74 actions)
Project 2	13.4% (9 out of 67 actions)	21.6% (16 out of 74 actions)
Project 3	19.4% (13 out of 67 actions)	27% (20 out of 74 actions)
Project 4 (self-questionnaire)	Inapplicable	0.002% (1 out of 74 actions)

In the cases of project 2 and 3, the model lag was calculated with and without the MPA Release Planning. Since the projects were developed in bespoke environment, the actions in RP were not relevant.

As seen from the presented data, the model lags drawn from four projects are moderate. Especially in the case of Project 4, it is closed to 0. This implies that the model is applicable in different development environments. Practitioners can find actions in the model useful and necessary to implement. Besides, in many cases, the reason for skipping certain actions is because the systems being built were derived from existing products hence part of information could be reused.

However, there are some cases in which interviewees claimed that the actions were not suitable for their situations. One example of this case is for action "DS.GA.a1 Define Requirements Attributes". He mentioned that the project size was small and the process was kept simple, hence they did not make use of attributes to manage requirements. Therefore, some of the actions are more applicable to bigger projects. This may lead to the consideration of which projects size would benefit most from using the model.

4.3.1.2 Common inapplicable actions in the model

The following actions have been found as inapplicable in more than one project.

- OS.RR.a5 Define Roles and Responsibilities for Product Management
- RE.SI.a3 Identify Other Requirements Sources
- RE.GA.a7 Reuse Requirements

The interviewees found that it would be too complicated to involve Product manager in their requirements process. It was due to the fact that the requirements were often built by developing teams, hence the involvement of Product managers was not considered as necessary. Moreover, in most of the interviewed cases, the customers are the main and most important source of requirements. Small developing companies did not encounter adverse impact for not identifying other requirements sources such as bug reports. Therefore, this may again lead to the consideration of the applicability of the actions in model in different project sizes. The same reason applies to "reusing requirements" action. Only mature process can plan and manage the reuse process systematically to reap benefits from that.

In the case of Project 2 and 3, all actions in MPA Release Planning were skipped due to the particular situations where they are both bespoke development.

In the researcher's point of view, the model lag presents the possible improvement area of the model in order to make it more fitting to the real situation. Hence, actions that are usually inapplicable in industry like those above should be considered to remove from the model.
On the other hand, practices applied in the industry repeatedly should also be added to the model. However, the decision of adding/deleting action must only be drawn from the evaluation with a large population of projects in order to ensure its accuracy. Therefore, in this case, those common inapplicable actions still remain in the model until further evaluation.

4.3.2 Usability

4.3.2.1 Efficiency

The interviews were completed in the designated time frame of 1.5 hours while it took 40 minutes to fill in the self-administered questionnaire. In addition, the assessment exercise did not require costly resource besides the practitioner and sometimes the interviewer. The detailed result analysis generally took ten hours of work. Based on those data, the Uni-

REPM model is proven to be a quick and cheap way to assess the RE process maturity in organizations.

4.3.2.2 Understandability

In this step, the researcher attempted to detect difficulties of practitioners in understanding the questions by looking for certain problematic signs during the conversation. The signs could be that the interviewee voiced out his understanding of the concept, which was different from the intended meaning, or hesitated in answering the questions or provided unrelated answers. When one of these signs happened, the researcher tried to narrow down the root causes by rephrasing the question and/or providing the concept definition/description in the model. If the misunderstanding problem is solved by rephrasing the question, the root cause may be poor construction of the question or wrong choice of words. If the problem is solved by explanation from the model, the root cause does not lie in either the checklist or the model. The confusion was made because the concept was new to the interviewee. In case more information needs to be given besides the definition in the model, this shows that the action in the model is inadequately described.

Based on the above categories of problem and root cause, the interview feedback were scrutinized and classified into relevant categories. After that, certain response actions were taken in order to improve the model. The below section is divided into three parts: improvements on the checklist and improvements on the model, no further necessary action.

4.3.2.2.1 Improvements on checklist

Based on the root cause analysis, certain misunderstanding problems were caused by the poor construction of the question or the wrong choice of word used in the question. Therefore, in other improve it, the questions were rephrased. The list of changes made on the checklist is shown in the Table 8 together with the reasons.

Table 9. Refinements made on checklist

Action UID	Original question	Improved question	Reason
OS.RR.a4	Change is inevitable. How do you deal with requirements changes, and who are involved?	Change is inevitable. What are the involving roles when requirements changes occur?	Part of the question overlaps with the action "PM.CM.a3 - Do you have a defined process for how to manage change requests?"
OS.GA.a2	Do you train personnel about requirements development and management processes as well as necessary skills?	Do you have training about requirements development and management processes as well as necessary skills?	The interviewee may not be manager who is responsible for organizing training.
PM.GA.a1	Do you define and document the process of how you perform requirements development and management?	Do you define how to process requirements?	The question caused confusion.

PM.CM.a3	Do you have a defined process for how to manage change requests? Do you have a defined process for keeping the requirements up to date with the current development status? Do you have a defined process for communicating changes to the rest of the organization?	Do you have a defined process for how to manage change requests?	The old questions are too long and may confuse the interviewee. The interviewee only focuses on the first question.
PM.RT.a2	Do you document the source of a requirement so that you may go back and get further information? The source can e.g. be documents, process descriptions, competitor products, as well as people.	Do you document where the requirement comes from? E.g. from customer, existing system document, competitor.	The question caused confusion.
PM.RT.a5	Do you document the impact of a requirement on other artifacts such as pre-studies, product design, implementation artifacts, test cases, etc.? When you produce an artifact, do you attach information about affected requirements?	Do you document the impact of a requirement on other artifacts such as pre-studies, product design, implementation artifacts, test cases, etc.?	The old questions are too long and may confuse the interviewee. The interviewee only focuses on the first question.
RE.GA.a3	Do you let the business objectives influence how you conduct your elicitation efforts?	Do you use the business objectives to guide you on how to conduct your elicitation efforts?	The question caused confusion.
RE.GA.a4	Depending on each situation, certain elicitation techniques are more appropriate to use than others. Do you consider different techniques to suit each case?	Depending on each situation, certain elicitation techniques are more appropriate to use than others (e.g. observation, interview, brain storming). Do you consider different techniques to suit each case?	Examples are needed to illustrate the concept.

RE.GA.a6	Do you create different channels to capture requirements from various sources in different form and at different time?	Do you create different channels to capture all forms of requirements from various sources?	The question had poor wording.
RE.SI.a1	Do you identify and involve different stakeholders in elicitation? For example, customers, end users, testers, developers, marketing etc	Do you identify all potential stakeholders and involve relevant ones in elicitation? For example, customers, end users, testers, developers, marketing etc	The question does not reflect fully the intent of the underlying action.
RE.DC.a1	Do you systematically elicit information about, and consider restrictions or possibilities that the domain may impose on your product?	Do you systematically elicit information about restrictions or possibilities that the domain may impose on your product?	The question was bewildering.
RE.DC.a3	When eliciting requirements, do you consider how your system will contribute to the business process in customer's organization?	When eliciting requirements, do you consider how your system will contribute to the organizational business?	The original question only targets to bespoke environment. However, it should be applicable in all environments.
RE.DC.a6	Are you aware of and have ways to deal with the political or organizational influence on the requirements sources when eliciting requirements?	Are you aware of the political or organizational influence on the requirements sources when eliciting requirements?	For political issues, in certain cases, although interviewee is aware of it, he may not have ways or defined process to deal with it.
RA.GA.a2	Do you have a systematic way of prioritizing the requirements so that you know which to focus on? Do you priorities based on several viewpoints (e.g. cost, value, risk, penalty)? Do you priorities with the help of your stakeholders? Do you weigh the importance of different stakeholders?	Do you have a systematic way of prioritizing the requirements so that you know which to focus on? Do you priorities based on several viewpoints (e.g. cost, value, risk, penalty)?	The old questions are too long and may confuse the interviewee. The interviewee only focuses on the first question.

RA.GA.a3	Do you systematically **estimate** whether there are any dependencies or relations between requirements?	Do you systematically analyze whether there are any dependencies or relations between requirements?	The question had wrong choice of word.
RA.GA.a5	Do you estimate and document how much requirements may impact or increase or decrease the value of other requirements?	Do you estimate and document how much requirements may impact positively or negatively the value of other requirements?	The question was bewildering.
DS.GA.a1	Do you have a standardized structure for how the System Requirements Specification should be written, or generated?	Do you follow a template defining how the System Requirements Specification should be written, or generated?	Companies are familiar with the term "template" than "standardized structure"

4.3.2.2.2 Improvements on model

During the interviews, the researcher identified some confusion which is related to the model but not the checklist. Hence, certain actions in the model needed refinements to improve the understandability of the user. The refinements that were made on those are:

Regarding action "OS.RR.a3 Define Roles and Responsibilities for Release Planning", many practitioners did not understand the concept "release planning" due to their bespoke development environment. Hence, a definition of "release planning" was added into the action description to facilitate user understanding.

The similar problem happened with action "OS.RR.a5 Define Roles and Responsibilities for Product Management". Therefore, the definition of Product management was provided in this action description.

In the description of action "RE.DC.a4 Elicit Information about System's Operational Domain", an example was added to illustrate this concept.

When conducting interviews, the researcher found that the answers to two questions "RE.SI.a1 Identify and Involve Relevant Stakeholders" and "RE.SI.a2 Distinguish between Different Types of Stakeholders" overlapped each other. In practice, when the companies identify stakeholders, they are already aware of different types of stakeholders. Therefore, as the action RE.SI.a1 covers RE.SI.a2, the action RE.SI.a2 was removed from the model. Hence, the final version of the model consists of 73 actions in total.

4.3.2.2.3 No further necessary actions

There were times when the interviewee did not understand the question due to a new/unfamiliar concept. However, giving them the action description solved the problem. Therefore, no further action is needed in these cases. The actions that fall into this category are:

- OS.S.a1 Define Product Strategies
- OS.S.a2 Define Product Roadmaps
- PM.CM.a2 Baseline Requirements
- RE.GA.a3 Let Business Concern Guide Focus of Elicitation
- RE.GA.a6 Create Elicitation Channels for Requirements Sources

4.3.2.3 Satisfaction

Apart from collecting the project evaluation results, the author also extracted feedback from interviewees on the usability of the model. The following comments were analyzed and considered in the improvement of the model.

The interviewee from project 1 commented that he liked the question lists. “I thinks your questions are really nice because some questions make me remember some answer from other questions, so you create synergies between the questions” hence “If the interviewed person forgets about something he/she could remember it with some other questions“. However, he also mentioned that some of the questions were long and he only focused on the first part of the question. This comment was considered to refine some questions in the list in order to improve the applicability of the model. It is also recommended to establish the relationships between questions in the list in order to avoid asking inapplicable questions (e.g. if there is no product roadmap in the process, it is not necessary to answer question RP.GA.a1).
Moreover, the interview also suggested that bringing too much knowledge into one single requirement process such as RAM, PARSEQ, etc. might overload the process. He also recommended developing a classification or description of environments in which the model is suitable.

In project 3, the practitioner mentioned that the checklist/model was quite long and involved many more actions than the real process. This, in fact, is also one of the intentions of the researcher, which is to make Uni-REPM a central point of Requirements Engineering knowledge. Instead of being overwhelmed by the large amount of information available in research and not knowing where to start, the practitioner can refer to the model, quickly identify what should be done to improve the process and follow the link in the model to the corresponding literature for more information.

4.3.2.4 The particular case of project 4

In project 4, the interviewee acted herself as an appraiser. She read through the whole model description by herself and used it to assess her project by answering all the questions in the checklist. The interviewee said that all the terms used in the checklist are familiar and the explanation in the model was adequate and easy to understand. Apart from the process assessment, a mini-interview was arranged to post-check and evaluate her result. This interview confirmed the assessment result she had performed. Moreover, it was confirmed that the interviewee had the same understanding of the “difficult” concepts as intended. Hence, the result from this case convinces us of the model usability. It shows that it is possible for an engineer to learn and apply the model in real work without additional help.

4.3.3 Usefulness

Besides the purpose of evaluating the usability and applicability of the model, the dynamic validation also aims to validate the usefulness of the model in assessing process maturity level of the project. Hence, the raw results obtained from the interviews and questionnaire were analyzed and the overall process maturity was determined. Moreover, each main process area was scrutinized to locate the strong and weak points of the process. Based on those findings, specific improvement actions were recommended in order to increase the maturity of the process. The detailed evaluation result and analysis of each project can be found in Appendix D, E, F, G.

The interviewee in project 2 expressed that the checklist and the corresponding actions were useful because they gave him ideas on how to improve the process in the next project. This is also one of the contributions of this study, which is to narrow the gap between the research knowledge and the industrial practice. While there have been a lot of works done in research, not many of the practitioners are aware of and make use of them, which make both sides lose.

In project 4, interviewee commented that the idea of the model was very nice but it would be more beneficial for practitioners to get the information of how to perform actions recommended in the model. Currently, Uni-REPM functions as a guideline mostly with "what to do" guidance. However, the missing information for how to perform actions is considered as more important and preferable from the companies' point of view. Hence, implementing these suggestions will improve the usefulness of the model. However, due to time limitation, the researcher opted to leave this suggestion for the next version of the model. This will be a quite huge improvement on Uni-REPM and will require another validation phase to evaluate the result.

4.4 Conclusion

In this phase, the dynamic validation of Uni-REPM was performed in four companies crossing three countries. The Uni-REPM has shown to be applicable in various development environments in these companies ranging from bespoke to market-driven and sometimes a mix of those twos. Moreover, the diverse in project characteristics and project size, as well as the geographical distribution of the projects proved that Uni-REPM could be broadly used in many different projects.

Generally, in all the three interviews, the practitioners showed a good understanding of the checklist and the actions in the model. They were familiar with most of the terms used and the actions as well. Moreover, the Uni-REPM model is efficient in assessing the RE process maturity in organizations given by the short duration of the interview or the self-questionnaire. In addition, project 4 has indicated that the model is usable by using self-administered questionnaire.

The obtained results showed that in all projects the model could assess the RE process maturity under the designed method. Moreover, it could detect the strong and weak areas in the process and recommend additional actions to improve it.

During the validation phase, the author experienced the difficulties to approach companies due to trust problem. However, most of the interviewees showed quite positive feedback about the idea of the model and its usefulness. Driven by the purpose to product a lightweight solution, the assessment result obtained by Uni-REPM model only helps to identify strong and weak points in the process and provides suggestions on how to improve it. In case the organizations want to analyze root causes of the problems and develop a detailed improvement plan suitable to their own situation, a more rigorous tool should be applied to achieve that.

Overall, the study indicates that Uni-REPM is applicable, usable and useful for practitioners to use in assessing RE process maturity in industrial organizations.

5 VALIDITY THREATS

In this section, the validity threats to both the static and dynamic validation are presented. There are four types of validity threats: conclusion, internal, construct and external [42] and each of them is discussed below.

5.1 Static Validation

5.1.1 Internal validity

There is a threat regarding the instrument used in the validation process. If the instrument is designed badly, it can affect the result negatively [42]. Therefore, in order to minimize this risk, the question list underwent a review by the advisor and a pilot test with a PhD student in Software Engineering to avoid poor wording, confusing questions and overlooked areas. Refinements were made to the question list before being applied in the interviews with the experts.

5.1.2 Conclusion validity

Conclusion validity threats are those that affect the ability of draw correct conclusion about the relationship between the treatment and the outcome [42]. In this study, the threat of searching for a specific result was tackled by recording and analyzing all positive and negative feedback from experts equally.

Another threat is the random heterogeneity of subjects as the subjects range from professors with many years of experience to PhD students and their experience areas are diversified as well. However, this is considered more of an advantage than a threat because the model was then scrutinized from various viewpoints and suitable for people with different levels of knowledge.

5.1.3 Construct validity

One threat identified in this study is the experimenter expectancy, which is the bias the experimenter exert on the result consciously or unconsciously [42]. In order to mitigate it, the interview questions were formulated to uncover weaknesses in a neutral way.

5.1.4 External validity

The external validity relates to the generalization of the result to a larger population [42]. The interaction of selection and treatment can pose a threat in this study as the participating experts are academic researchers and not industrial practitioners. However, all of the experts involved have relevant industry experience which can mitigate the threat. On top of that, as the experts came from different countries, their feedback also represented international experience.

5.2 Dynamic Validation

5.2.1 Internal validity

When the project selection was done by the subjects, it posed a threat as the project selected might not be representative of the whole organization. However, this is a low risk because the assessment result is kept anonymous and the main purpose of the task is to validate the model not the company. Another threat is that the convenience sampling technique made the selection method statistically unsatisfactory. There is another threat to the validity of the project evaluation result caused by one perspective. As only one practitioner participating in the interview/questionnaire, the answers to the checklist was one-sided and may not reflect the whole process fully. If more perspectives were involved in the process, the accuracy of the findings would be improved. Moreover, the threat caused by the instrument used is also

reduced by reviewing and refining the checklist as well as validating the model through static validation before conducting the dynamic validation.

5.2.2 External validity

There is a small threat caused by the interaction of selection and treatment as the validation was only performed in four organizations. The threat is reduced by reporting the characteristics of the environments and providing details about the projects under evaluation as well. Moreover, as the companies diversify as two of them are from Europe (Spain and Denmark) and two are from Asia (Singapore), it implies that the model is applicable internationally and in various contexts.

5.2.3 Construct validity

The threat of evaluation apprehension exists as the subjects may try to perform better when being evaluated [42]. However, the risk is low because the subjects were clearly communicated about the purpose of the exercise which validates and improves the model. Moreover, the result analysis generated will be kept anonymous so the subjects are honest when answering whether an action is performed in the project.

The threat of experimenter expectancy is low as the raw result obtained from the practitioner was not tainted by the interpretation of the interviewer. The interviewer only asked questions in the list and provided explanation when needed. It was the solely the decision of the practitioner of whether an action is “complete”, “incomplete” or “inapplicable”.

5.2.4 Conclusion validity

The threat of searching for a specific result is low because after interview, the raw result is sent back to the interviewee to verify and approve before it is analyzed. This is to ensure that the obtained raw result is correct before further processing.

6 CONCLUSIONS AND FUTURE WORK

This chapter presents the conclusions drawn from the process of validating the Uni-REPM model with academics experts and industrial practitioners as well as the project evaluation method.

The chapter is divided into four parts. The first one summarized what have been done in this study. Then the research questions are revisited to figure out to what extent they were answered, followed by the conclusions deduced from the study. Finally, some future works are identified to further improve the model.

6.1 Study Summary

In this study, the Uni-REPM model proposed in [29] was validated in academic and industrial settings. Uni-REPM model is the product of an extensive research on literature in order to find a quick and cheap way to assess the Requirements Engineering process maturity level in organizations regardless of its size and developments environments. However, before releasing it to the industry to be applied, it needs to be refined and tested. Therefore, the motivation of this study is to prepare for a successful technology transfer of Uni-REPM model into the real environment by multiple steps of validation.

The first step of validation was performed by having seven academic experts review the model. The experts were chosen based on their experience in Requirements Engineering and their close collaboration with the industry. Interviews were arranged to collect their feedback on the model accuracy, completeness. Based on the feedback, modifications were made on the model to improve it before applying it in industry pilot.

After being improved, the model was validated further by using its checklist in assessing real projects in four industrial organizations. Both interviews and self-administered questionnaire were used to collect results and feedback from the practitioners. The model was judged on its applicability and understandability and improvements were identified. Moreover, according to the practitioner's perspective, the model helped to uncover strengths and weaknesses in their process by looking at the detailed analysis of the process evaluation.

6.2 Research Questions revisited

RQ1: To what extent is the model suitable for industrial application, in terms of its correctness, completeness and applicability?

Based on the static validation result in section 3.3, there were in total 65 feedback collected from the experts. Most of them relate to the model correctness in terms of the action name, the maturity level the action resides as well as scope of the actions. This, however, was expected as currently there are a lot of researches going on in this area and there has been no fixed set of terms/activities that everyone agrees on.

The model was quite complete as there were only a few suggestions to add new actions. Moreover, the amount of information presented in each action is adequate for understanding what it does and what benefit it brings without overwhelming the reader. The experts found the "example" section useful for practitioners as it provided ideas on how to implement certain actions and links to other literature for more information.

Regarding the applicability of the model, the experts believed that all actions are useful and applicable and no action should be removed from the model. Although most of the actions are applicable in both bespoke and market-driven development settings, some are more useful and essential in one setting or vice versa. However, according to all experts, because this aspect relates to the real application of the model, it should be evaluated in practice in

order to obtain the accurate result. Therefore, the dynamic validation step performed later will take this aspect into account.

RQ2: To what extent is the model useable and applicable for industry practitioners?
Taking a look at the results obtained from the dynamic validation of the model in real projects (section 4.3), it is worth noticing that the model is highly applicable. This is shown by the small number of inapplicable actions in all the four projects evaluated. Especially in the case of high mature process (project 4), 73/74 of the actions are deemed applicable. Moreover, taking into account the diversified characteristics of the four projects involved, it is safe to say that the model is applicable in various development environments.

Besides the applicability, the aspects of understandability and usability were also evaluated. The aspects were first judged on the usage of the checklist and then the model itself. The checklist is the operational form of the model. Besides some minor difficulties in understanding the checklists, the practitioners claimed that they were familiar with most of the terms and concepts used. They also had no problem in understanding how the evaluation worked and the concepts of "complete", "incomplete" and "inapplicable" options. It is interesting to notice that the project evaluation in project 4 was completely done by the practitioner on her own without any outsider's help. This shows the highly usability of the model and method although further case studies should be done to confirm the result.

Although the model/checklist consists of a large amount of actions (74 actions), all the interviews were done in one and a half hours and the self-questionnaire took 40 minutes. Hence, this is considered a quick way to obtain the assessment result compared to other similar solutions. Moreover, as in the case of the project 4, it does not require training or hiring an appraiser to perform the evaluation. Therefore, Uni-REPM is verified to be a quick, easy and cheap way to assess RE process maturity in organizations.

RQ3: What improvements can be done to the model based on the findings in RQ1 and RQ2?
Regarding RQ1, although all 65 suggestions contained valid arguments, not all of them were implemented in the improved version of the model. The reason is that several trade-offs have to be considered in incorporated those changes and also the available resource allocated to this study. The list of improvements made on the model after static validation can be found in section 3.4.

Regarding RQ2, most of the problems lie in the checklist rather than the model itself. Therefore, modifications were made in the checklist to make it clearer and reflect the correct intent. Some definitions were also added into the model description to explain specific concepts. Additional actions identified in real process were also considered to be added in the model. Section 4.3 contains all the refinements made to the model to improve it.

6.3 Overall conclusions

After the two refinement steps, the final version of Uni-REPM consists of 73 actions which are applicable more or less in both bespoke and market-driven environments, Although the action descriptions focus more on "what to do" rather than "how to do", they do not impact the model usefulness negatively. The reason to keep the action in high-level abstraction is that the implementation of the action varies greatly among organizations depending on its organizational characteristics, development environments and project specifics. Including the action implementation information will make the model extremely bulky which goes against the purpose of a lightweight assessment model. Hence, it was decided to provide reference to the corresponding publications in order to make the model more complete without sacrificing its compactness.

The model performed satisfactory in four case studies. The detailed result analysis revealed not only the current state of the process but also the improvement suggestions that can be considered to implement. The four case studies were conducted with companies in three countries (Denmark, Spain and Singapore), spanning from Europe to Asia. This indicates that the model can be applied in international settings. However, further case studies in other countries need to be done to confirm the result.

Through this study, it is proven that the static and dynamic validation steps are useful and necessary to prepare a research solution to be applied in industry. The static validation is the efficient and effective way to gather early feedback on the model without wasting valuable industrial resource. Moreover, in case the research product is the aggregation effort from multiple publications like Uni-REPM, involving many researchers with diversified expertise is a good approach to ensure that no single area/aspect was overlooked. Regarding the dynamic validation, besides interviews, self-administered questionnaire is shown to be a promising method to conduct the assessment in term of the time, effort and resource spent.

Finally, the Uni-REPM model has managed to lessen the gap between academic researches and industry adoption of the research findings by concentrate and present them in a practical and usable way that practitioners can easily and quickly apply.

6.4 Future work

In this study, the static and dynamic validations of the model were conducted and presented in order to produce a good usable RE process maturity assessment tool. However, so as to ensure a successful technology transfer of the research solution to practice, further steps as follows need to be taken:

6.4.1 Get feedback from the practitioners about the assessment results

Although the project evaluation results and their detailed analysis were already prepared and presented in Appendix D, E, F, G, they have not been released to the respective interviewees. Therefore, the next step is to provide the participating companies the detailed result together with the model and collect their feedback on the applicability and usefulness of recommended actions. This information will help to further improve the model in the next iterations.

6.4.2 Extend the dynamic validation to more organizations

As the dynamic validation was performed on a limited number of companies, there is a need to reach out for more companies in order to have a larger pool of data and feedback for improving the model. Specifically, it is desirable to apply the model in companies working in the market-driven development environment as this area was not verified adequately due to the disadvantage of the chosen sampling technique. Moreover, it would be beneficial to involve more perspectives in the assessment to obtain a well-rounded view of the RE process.

6.4.3 Automate the project evaluation process

Based on the good experience of using self-administered questionnaire as the evaluation means, computerizing the project evaluation process can help to reduce the time, effort and cost of the participating companies. The automation can be done in various steps from answering the checklist to generating the detailed results and graphs. One drawback of this approach is that the participants solely depends on the model for explanation and further information, hence the model must be of good quality. Moreover, the accuracy of the result depends on the participants' self-consciousness in checking the model description when in doubt. The participant should not rely on his assumption in answering the checklist.

6.4.4 Conduct survey to validate the model

Once the automation is done, conducting a survey is a good way to validate the model in the wider population. Many companies can do the assessment at the same time and the result can

be analyzed statistically. In this way, the solution can reach out to more audience and a successful technology transfer can be performed.

6.4.5 Obtain information about Requirements engineering state-of-practice

The gap between current RE practices and published research solutions has always existed [22]. However, there is still little contemporary evidence about what software professionals actually do in RE in order to improve the practices [9, 43]. Fortunately, Uni-REPM can be used as an instrument to collect such data. By conducting a survey and analyzing the results obtained from different organizations, one can extract information about the requirements engineering state-of-practice. Uni-REPM covers all areas of RE in a deep thorough manner. The results can then pinpoint which areas in practice need more focus and researchers can investigate and come up with solutions.

REFERENCES

[1] Somerville I., *Software Engineering*, Addison-Wesley, 1995.

[2] Wohlin C., Aurum A., *Engineering and Managing Software Requirements*, Springer, 2005.

[3] Carmel, E., Becker, S., "A Process Model for Packaged Software Development", *IEEE Transactions on Engineering Management*, Vol. 42, No. 1, pp 50-60, 1995.

[4] Sawyer, P., "Packaged Software: Challenges for RE", *Proceedings of the sixth Int. Workshop on Requirements Engineering: Foundations of Software Quality (REFSQ'00)*, pp 137-142, 2000.

[5] Sawyer P., Sommerville I., Kotonya G., "Improving Market-Driven RE Processes", *Proceedings of the Int. Conf. on Product Focused Software Process Improvement*, Oulu, Finland, pp. 222-236, 1999.

[6] Potts C., "Invented requirements and imagined customers: requirements engineering for off-the-shelf software", *Second IEEE International Symposium on Requirements Engineering (RE'95)*, pp.128, 1995.

[7] Regnell B., Brinkkemper S., "Market-Driven Requirements Engineering for Software Products", *Engineering and Managing Software Requirements*,Springer, New York NY, 2005.

[8] Carlshamre P., "A usability perspective on requirements engineering – From methodology to product development", Ph.D. thesis, Linköping Institute of Technology, Sweden, 2001.

[9] Lubars M., Potts C., Richter C., "A Review of the State of the Practice in Requirements Modelling", *First IEEE Int. Symposium on Requirements Engineering (RE'93)*, pp2-14, 1993.

[10] Gorschek T., "Requirements Engineering Supporting Technical Product Management", Karlskrona : Blekinge Institute of Technology, 2006.

[11] Kotonya G., Sommerville I., *Requirements Engineering – Processes and Techniques*, John Wiley & Sons, Chichester UK , 1998.

[12] Karlsson L., Dahlstedt ÅG., Natt och Dag J., Regnell B., Persson A., "Challenges in market-driven requirements engineering - An industrial interview study", *Proceedings of 8th International Workshop on Requirements Engineering: Foundation for Software Quality (REFSQ'02)*, Essen, Germany, pp.37-49, 2002.

[13] Carlshamre P., "Release Planning in Market-Driven Software Product Development: Provoking an Understanding", *Requirements Engineeging*, Vol.7, pp.139–151, 2002.

[14] Yeh AC., "Requirements Engineering Support Technique (Request): A Market Driven Requirements Management Process", *Proceedings of the Second Symposium on Assessment of Quality Software Development Tools, IEEE*, Los Alamitos CA, pp. 211-223, 1992.

[15] Regnell B., Beremark P., Eklundh O. "A Market-Driven Requirements Engineering Process - Results from an Industrial Process Improvement Programme", *Requirements Engineering*, Vol.3, No.2, pp.121-129, 1998.

[16] Carlshamre P., Regnell B., "Requirements Lifecycle Management and Release Planning in Market-Driven Requirements Engineering Processes", *Proceedings of the 11th International Workshop on Database and Expert Systems Applications, IEEE*, Los Alamitos CA, pp.961-965, 2000.

[17] Davis AM., "The Art of Requirements Triage", *IEEE Computer*, Vol.36, No.3, pp.42-49, 2003.

[18] Curtis B., Krasner H., Iscoe N., "A Field Study of the Software Design Process for Large Systems", *Communications of the ACM*, Vol.31, No.11, pp.1268-1287, 1988.

[19] El Emam K., Madhavji N.H., "A Field Study of Requirements Engineering Practices in Information Systems Development", *Second IEEE Int. Symposium on Requirements Engineering (RE'95)*, pp 68-80, 1995.

[20] Hall T., Beecham S., Rainer A., "Requirements Problems in Twelve Software Companies: An Empirical Analysis", *Proceedings of the Conference on Empirical Assessment in Software Engineering (EASE)*, 2002.
[21] Nikula U., Sajaniemi J., Kälviäinen H., "A State-of-the-Practice Survey on Requirements Engineering in Small- and Medium-Sized Enterprises", TBRC Research Report 1, Telecom Business Research Center Lappeenranta, Lappeenranta University of Technology, 2000.
[22] Juristo N., Moreno A.M, Silva A., "Is the European Industry Moving Toward Solving Requirements Engineering Problems?", *IEEE Software*, Vol.19, No.6, pp. 70-77, 2002.
[23] Villalón C., Agustín C., Gilabert S., Seco D., Sánchez G., and Cota P., "Experiences in the Application of Software Process Improvement in SMES", *Software Quality Journal*, Vol. 10, pp. 261 – 273, 2002.
[24] CMMI for Development, Version 1.2, CMMI-DEV v1.2, CMU/SEI-2006-TR-008, Technical Report, Software Engineering Institute, August 2006, URL: http://www.sei.cmu.edu/pub/documents/06.reports/pdf/06tr008.pdf.
[25] The TickIT Guide – Using ISO 9001:2000 for Software Quality Management System, Construction, Certification and Continual Improvement, Issue 5.0, 2001.
[26] Sommerville I., Sawyer P., *Requirements engineering: a good practice guide*, John Willey & Sons, 1997.
[27] Gorschek T., Tejle K., "A Method for Assessing Requirements Engineering Process Maturity in Software Projects", Blekinge Institute of Technology, Master Thesis Computer Science no. MSC-2002:2, 2002.
[28] Gomes A., Pettersson A., "Market-Driven Requirements Engineering Process Model – MDREPM", Master Thesis Software Engineering, Thesis no: MSE-2007-06, January 2007.
[29] Nguyen L., "The creation of Uni-REPM: a universal model for assessing requirements Engineering Process Maturity", Master Thesis Software Engineering, Thesis no: MSE-2010:27, September 2010.
[30] Gorschek T., Garre P., Larsson S., Wohlin C., "A Model for Technology Transfer in Practice," *IEEE Software*, Vol. 23, No. 6, pp. 88-95, 2006.
[31] Kasunic M., *Designing an Effective Survey*, Pittsburgh, PA, Carnegie Mellon Software Engineering Institute, 2005.
[32] Staples M., Niazi M., Jeffery R., Abrahams A., Byatt P., and Murphy R., "An exploratory study of why organizations do not adopt CMMI", *Journal of System Software*, Vol.80, No.6, pp.883-895, 2007.
[33] Hockman KK., Grenville R., and Jackson S, "Road Map to ISO 9000 Registration", *Quality progress*, Vol. 27, No. 5, pp. 39-42, May 1994.
[34] Wiegers K., *Software Requirements*, Microsoft Press, Redmond, Washington, 2003.
[35] "IEEE standard glossary of software engineering terminology," Tech. Rep., 1990. [Online]. Available: http://dx.doi.org/10.1109/IEEESTD.1990.101064
[36] Seaman C.B., "Qualitative Methods in Empirical Studies of Software Engineering," *IEEE Transactions on Software Engineering*, Vol. 25, pp. 557-572 1999.
[37] Hove S.E. and Anda B., "Experiences from conducting semi-structured interviews in empirical software engineering research," *11th IEEE International Symposium Software Metrics*, p. 10, 2005.
[38] Creswell J., *Research Design: Qualitative, Quantitative, And Mixed Methods Approaches*, Sage Publications, California, 2003.
[39] Pettersson F., Ivarsson M., Gorschek T., and Öhman P., "A practitioner's guide to light weight software process assessment and improvement planning", *The Journal of Systems and Software*, Vol. 81, pp. 972–995, 2008.
[40] Scott L., Jeffery R., Carvalho L., D'Ambra J., and Rutherford P., "Practical software process improvement - the IMPACT project", *IEEE Computing Society*, Los Alamitos, CA, USA, pp. 182-9, 2001.
[41] Weerd I., Bekkers W., Brinkkemper S., "Developing a maturity matrix for software product management", *Proceedings of the 1st International Conference on Software Business (ICSOB 2010), LNBIP 51,* pp.76–89, 2010.

[42] Wohlin C., Runeson P., Höst M., Ohlsson M.C., Regnell B., and Wesslén A., *Experimentation in software engineering: an introduction*, Kluwer Academic Publishers, 2000.
[43] Neill C. J., and Laplante P. A., "Requirements Engineering: The State of the Practice", *IEEE Software,* Vol.20, No.6, pp. 40-45, 2003.

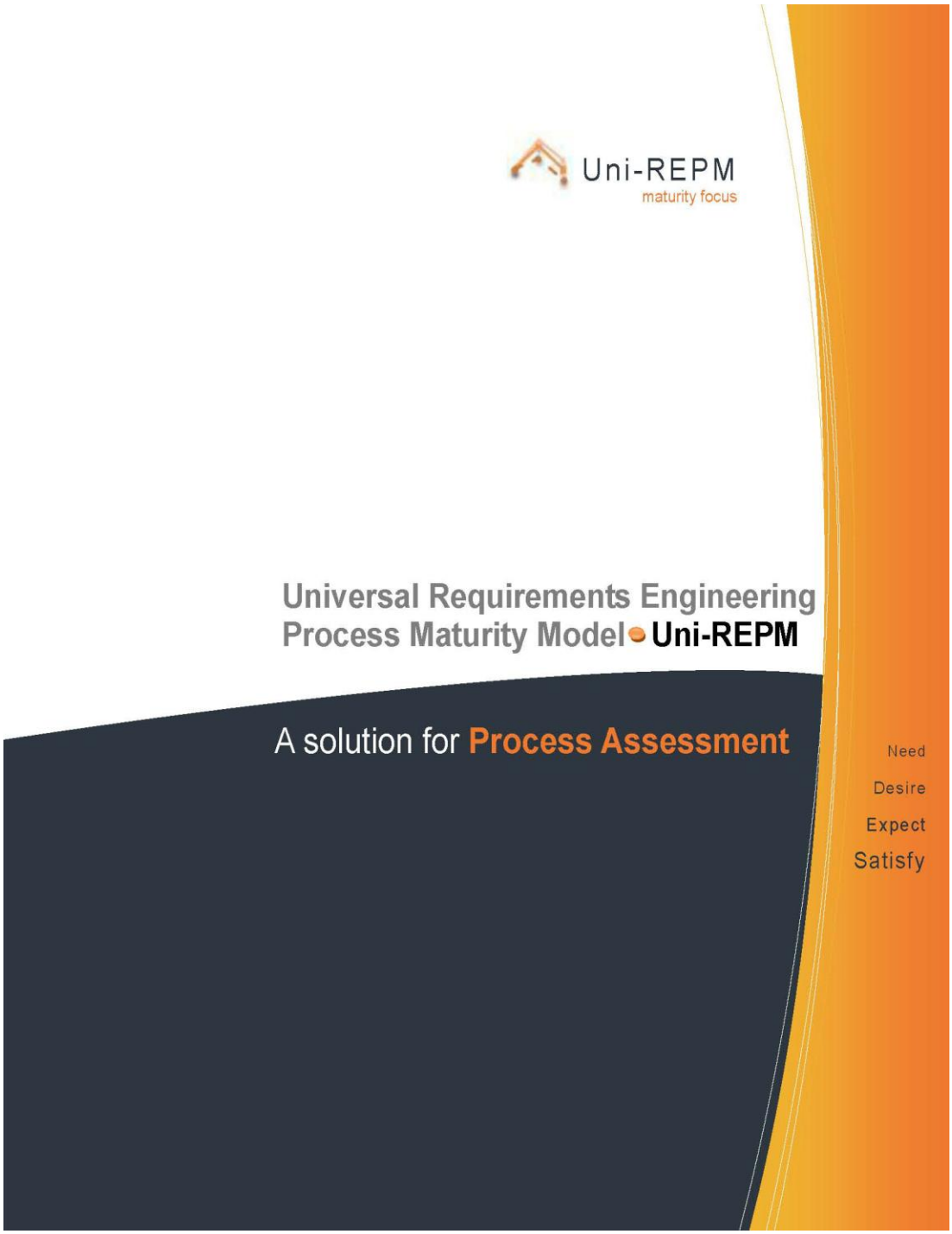
APPENDIX A – STATIC VALIDATION QUESTIONS

Participant Background Questions
1. What is your academic title?
2. How long have you been researching in RE/PM area?
3. Which specific area(s) in RE/PD have you focused your research on? (e.g. decision making, product life cycle…)
4. Have you been working in collaboration with industrial organization? If yes, to what extent (Number of projects, your involvement in the project(s)…) and for how long?

Leading questions	Possible in-depth questions	Aims
How long did you spend to study the Uni-REPM Model?		To evaluate the quality of the static validation
How important do you think the RE process maturity assessment is in practice?		To evaluate the model idea
Do you feel the structure of the model is easy to understand?	-Structure of MPA/SPA/Action in the model? -Model levels?	To evaluate the usability of the model
Do you feel the content of the model is clearly and adequately described?	-Model description? -MPA/SPA/Action name and description?	
Do you think that there is some main/sub process area missing in the model presentation? Why?		To evaluate the completeness of the model
Do you think that the main process areas are covered adequately?	-Is there any Action that is overlooked in *Organizational Support* area? -Is there any Action that is overlooked in *Requirements Process Management* area? -Is there any Action that is overlooked in *Release planning* area? -Is there any Action that is overlooked in *Elicitation* area? -Is there any Action that is overlooked in *Requirements Analysis (and Negotiation)* area? -Is there any Action that is overlooked in *Documentation and Specification* area? -Is there any Action that is overlooked in *Quality Assurance* area?	

Are there any MPAs/SPAs/Actions that are too detailed or abstract?	-Whether they need to be broken down or merged? Why?	To evaluate the completeness of the model
Do the Actions belong to the correct MPA and SPA?	-Which actions should be moved? Why?	Evaluate the correctness of the model
Do you think the Actions in the model are placed at the appropriate levels?	-Which Actions should be placed in different levels? Why?	
Are there any Actions that need to be removed according to your opinion?		
Do you think the model is suitable to be applied in industrial settings?	-What Actions are not suitable to be applied in industrial settings? -Why?	To evaluate the applicability of the model
In what context do you think the model would be used?	-Is the model applicable in various-size companies? -Is the model applicable in MD/bespoke companies? -Is the model applicable in hybrid (both MD and bespoke) companies?	
Can you name and explain 3 (the three most important) potential shortcomings with the model as you see it.		To identify possible improvement
Can you name and explain 3 (the three most important) potential benefits/strong points with the model as you see it.		To summarize the interview in a good way

APPENDIX B – UNI-REPM BROCHURE



They always occur…

AND CAUSE YOU HEADACHE!

- Project delay and low productivity.
- Projects cost increase far from estimation due to rework.
- Or even worse: project failure and wrong product deliverables.

Why let it that late to realize problems? You can always detect potential problems earlier during project duration in order to prevent them and improve your process with **Uni-REPM**.

Why leave it so late?

Fix it soon with **Uni-REPM**

Uni-REPM THE SOLUTION FOR YOUR PROCESS

Uni-REPM is a universal Requirements Engineering process maturity model. Uni-REPM helps project teams to:

- Address potential problems early at RE process
- Point out the strengths and weaknesses of the process
- Identify an improvements path for projects team.

USING **Uni-REPM**, THE PROJECT TEAMS WILL BE ABLE TO

- Build a more desirable product fulfilling customers' needs through Requirements Elicitation means and Domain considerations.
- Achieve common understanding among project teams by improving quality of requirements and maintaining communication channels.
- Efficiently handle complexity through traceability management.
- Improve productivity through well-defined processes and organizational structure.
- Mitigate project risk and avoid rework and duplication through in-depth Requirements Analysis.
- Attain business goals by aligning Requirements Engineering effort with product strategies.

STRONG FOUNDATION

Extensive research base
Well-known scholars support

LIGHT-WEIGHT METHOD

Easy to learn
Simple to use
Require low cost
Quickly get result

SPECIFIC SOLUTION

Adapt to situation
Detect concrete problem
Fully cover all areas
Provide detailed solution

A Glance At **Uni-REPM**

Uni-REPM is a product created at Blekinge Institute of Technology under the supervisions of *Prof. Mikael Svahnberg* and *Prof. Tony Gorschek*. It is the result of an extensive and systematic research under which studies in the field of Requirements Engineering throughout 17 years were aggregated. Moreover, the model also consolidated knowledge from many well-known scholars in the area. Hence, **Uni-REPM** represents the "best practices" of the field.

WHY **Uni-REPM**?

Specific

Simple

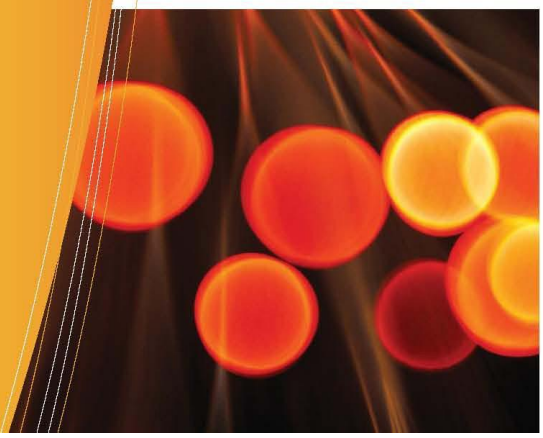
Easy

Accurate

Light & Quick

Uni-REPM FULLY AND ADEQUATELY COVERS SEVEN SPECIFIC AREAS

- ***Organizational Support***: covering activities to ensure adequate support given of the related organizations.
- ***Requirements Process Management***: covering all the activities to manage, control requirements change as well as to assure the enforcement of the process and collaboration among team members.
- ***Requirements Elicitation***: covering activities to discover and forecast the needs and wants of the potential stakeholders in order to convey this information to the system developers.
- ***Requirements Analysis***: covering activities to analyze product requirements and estimate necessary information for later steps (eg. risk, priorities…).
- ***Release Planning***: covering activities to determine the optimal solutions for a certain release to be implemented at an estimated time and cost to achieve pre-defined business goals.
- ***Documentation&Requirements Specification***: covering activities to organize requirements and other knowledge gathered during the process into consistent, accessible and reviewable documents.
- ***Requirements validation***: covering activities to verify product specifications against defined quality standards and the real expectations of various stakeholders.



Uni-REPM Maturity Levels

Uni-REPM assesses the maturity of the process according to 3 defined levels: ***Basic***, ***Intermediate*** and ***Advanced***. The levels represent the improvement of the Requirements Engineering process from an ad-hoc state to a highly-defined state. To reach a certain level, the process has to fulfill all the pre-defined actions in that level as well as those in the lower ones.

➢ LEVEL 0 – AD-HOC

There is no defined process and everything is done in an ad-hoc manner. Even if requirements produced work well, the success cannot be replicated.

➢ LEVEL 1 – BASIC

The aim of this level is to achieve a rudimentary repeatable requirements engineering process. The process are defined and followed. Quality of requirements is managed because of relevant stakeholder involvement in elicitation, in-depth requirements analysis and pre-defined documentation standards.

➢ LEVEL 2 – INTERMEDIATE

In this level, the process is more rigorous because it involves various perspectives and is led by product strategies/goals. Roles and responsibilities are clearly defined. Change requests are handled systematically. Well-informed decisions about requirements selection can be made by analyzing and prioritizing requirements.

➢ LEVEL 3 – ADVANCE

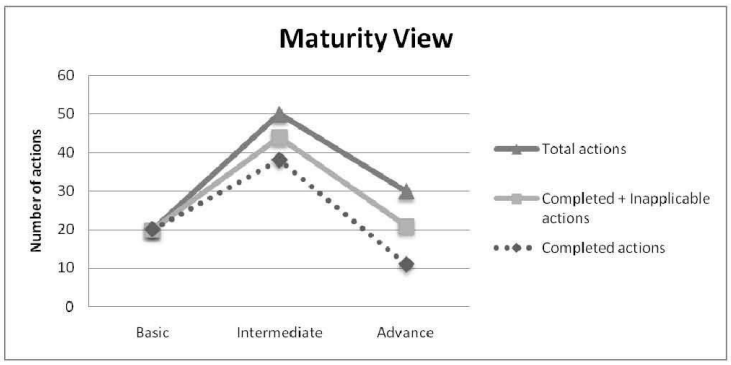
This level denotes the most mature process. The improvements in the process are shown in the advanced ways to capture requirements, to ensure their high quality, to maintain communications and common understanding among different involving roles and to pro-actively assess the decision making process.

Uni-REPM FEATURES

MATURITY ASSESSMENT

Using **Uni-REPM** as a maturity assessment tool, you simply perform a mapping from the actions present in Uni-REPM to the activities in the real process by answering a question list. You will find out certain activities in your process could be "*Complete*", "*Not Complete*" or "*Inapplicable*". "*Inapplicable*" option enables adaptability for you since in many cases, activities are found unnecessary to be carried out due to particular situations of your organization.

Uni-REPM will then generate a maturity view for specific areas within your Requirements Engineering process as well as a summary for the whole process.



The result may look like in the graph. All the activities under **Basic** level area have been completed in the process while it is not true for actions under **Intermediate** and **Advance** levels. Hence, the whole process resides on **Level 1 - Basic**.

MATURITY SUGGESTION

Maturity suggestion feature helps you easily build an improvement plan based on a close look at the result once the current situation of your process is detected. Using the breakdown results on 7 specific aforementioned areas, the strengths and weaknesses of your process will be quickly located in each area and an improvement path will then be generated.

MATURITY GUIDANCE

In order to use the model as a guideline, you simply implement actions following the description in the model. **Uni-REPM** provides very detail information in *Examples* section under each Action which suggests useful tools or techniques for particular tasks.

Uni-REPM

Constructor
Eng. Nguyen Thi Thanh Loan
Eng. Nguyen Thi Phuong Mai

Supervisor
Prof. Mikael Svahnberg
Prof. Tony Gorschek

Uni-REPM
maturity focus

Contacts:
Nguyen Thi Thanh Loan
Tel: (0046) 0708290544
Email: ttng09@student.bth.se
Nguyen Thi Phuong Mai
Tel: (0046) 0765594763
Email: mbng09@student.bth.se

APPENDIX C – DYNAMIC VALIDATION INTRODUCTORY QUESTIONS

No.	Introductory Questions
1	What is your position/job scope?
2	How long have you experienced in Requirements Engineering (RE)?
3	Which specific area(s) in RE does your job focus on?
4	What industry does your company belong to?
5	How many employees in company/business unit?
6	What is the project about? What is the product?
7	How many man-hours for the project? (how many people involved in the project? How long does it last?)
8	Is it as mass-market product? Or do you have specific customer(s) for this product?
9	Additional info you would like to share about the project?

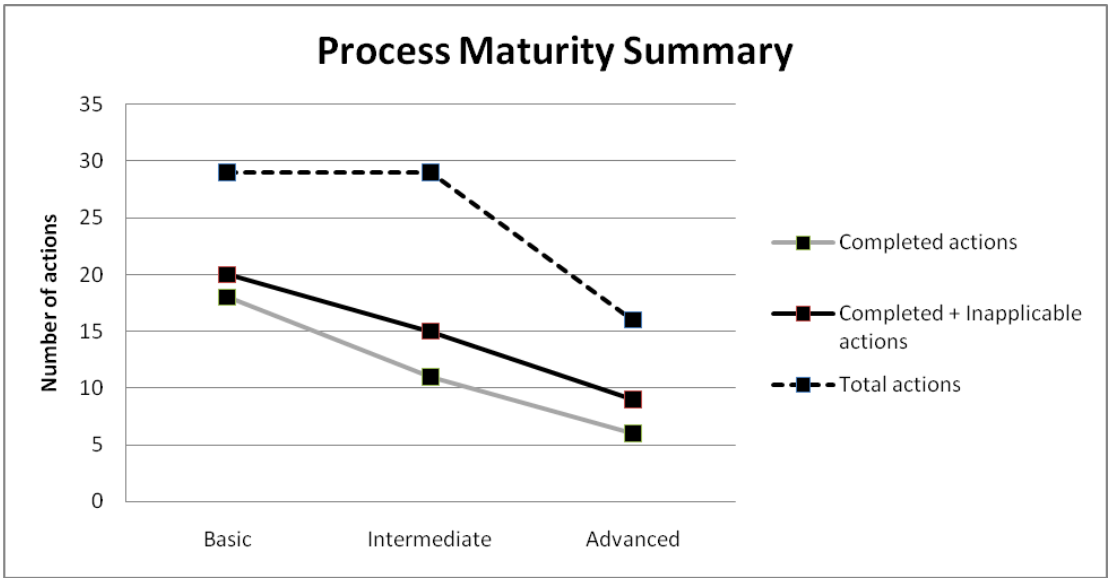
APPENDIX D – PROJECT EVALUATION RESULT 1

The company domain is software industry and there are fifteen employees working in this business unit. Regarding the project under assessment, it involves customizing two different modules for a client, one for internal Invoicing and the other for Space Management. Both are related to facility management. The system is derived from a product already developed in a MDRE context, and now being customized for the client. The project started four months ago and is still on-going. There are two employees in the team responsible for requirements elicitation, the project manager and project manager assistant.

1. PROCESS MATURITY OVERVIEW

Total actions overview in the process is summarized in the following table according to levels.

Level	Completed actions	Inapplicable actions	Total actions
Basic	18	2	29
Intermediate	11	4	29
Advance	6	3	16



General

From the chart and graph above, we can see that the whole process resides at level 0. The reason is that, 9 out of 27 actions at Basic level were not completed in the process. The model lag is fairly small giving that only 9 actions are inapplicable.

Summary

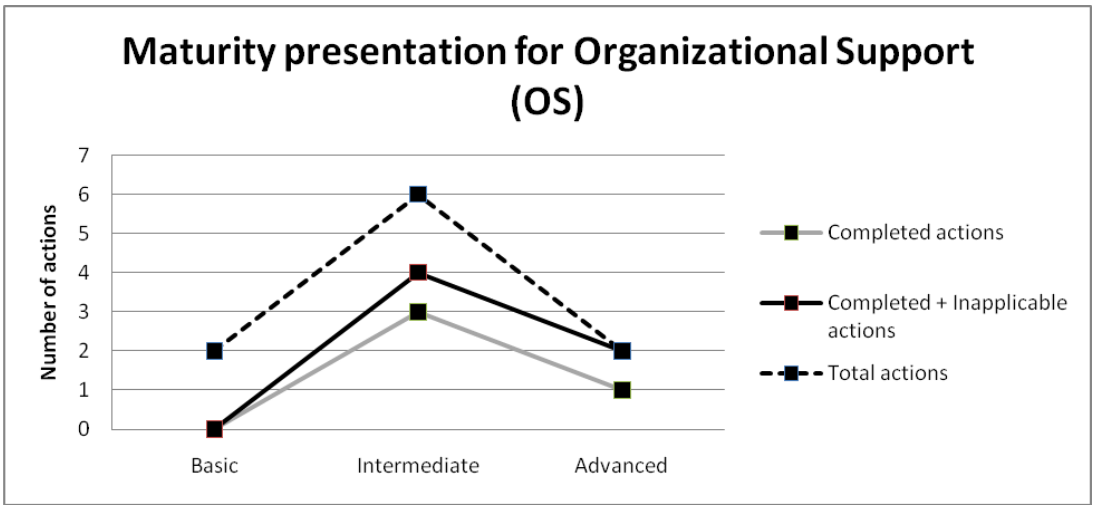
The project according to the interviewee is sort of hybrid project. The product of this project is specifically built for a fixed customer; however it is based on a generic product which was developed previously in the company. This leads to quite a number of inapplicable actions including the release planning MPA (where a large portion of inapplicable actions resides). In the whole process, communication was very well considered, especially with customers. This is a very good approach. However, most of the quality attributes of the requirements were left behind. This is the main missing part. Besides, documenting and versioning are not considered either. These two actions are not costly but bring a lot of benefit for the process. Especially with the glossary of term, the team members will pay much less time to learn the

product context and process. Overall, the process is still ad-hoc. Elicitation is mostly performed based on negotiation in meetings other than using models. Although customers were considered as core factor, many beneficial activities were not performed during the process. Some simple activities such as making use of checklist, etc. will significantly improve the productivity of the process.
However, the model-lag is fairly small in this assessment (excluding the Release planning MPA due to project characteristic). Only a few actions were stated inapplicable. Hence, the model is fairly closed to the real requirements process.
Looking at the whole process across all 7 MPAs, it is advisable to aim for Basic level when improving the process. In order to have a consistent and stable process, all actions in Basic level have to be fulfilled. It also means that the company only needs to perform 7 additional actions. The breakdown of the process in each MPA is presented as below so as to provide more insights and more detailed views of its strengths and weaknesses.

2. MPA ORGANIZATION SUPPORT

The number of completed and inapplicable actions according to each Uni-REPM level in Organizational Support compared to the total actions:

Level	Completed actions	Inapplicable actions	Total actions
Basic	0	0	2
Intermediate	3	1	6
Advanced	1	1	2



General

From the graph, we conclude that the project resides at level 0 in MPA Organizational Support. In order to achieve Basic level, two additional actions need to be completed. It is interesting to point out that although not all Basic and Intermediate actions are completed, all the actions in Advanced level are fulfilled.
We noticed that there were two inapplicable actions:

- *OS.RR.a3 Define Roles and Responsibilities for Release Planning (Intermediate Level)*
- *OS.RR.a5 Define Roles and Responsibilities for Product Management (Advanced Level)*

They are considered Inapplicable because the project is of hybrid type, which means that the product was derived from a market-driven product but was customized for the specific clients. Therefore, the release planning and product management activities happened long before and are not relevant in the current project.
The model lag which consists of only two actions are considered to be small and showed that the actions in this MPA have good applicability in the real setting.

Suggestions for improvement
In order to reach Basic level, two additional actions have to be completed:

- *OS.GA.a1 Create a Product-wide Glossary of Terms (Basic Level)*

There are many different users of the requirements documents ranging from the managers, developers to customers. And because of their different backgrounds, they may have different perceptions about a certain thing. Therefore, In order to unite everyone's understanding of the requirements documents, there should be a term definition documented in a central place that covers everything from domain specific terms to product/project acronyms.

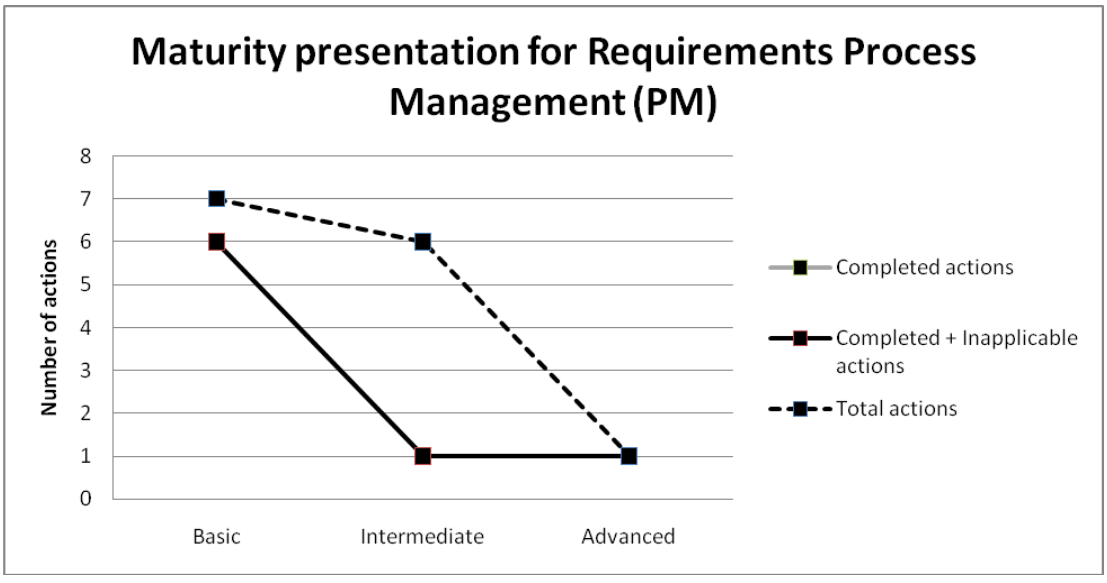
- *OS.RR.a1 Assign Owner(s) of Requirements Development and Management Processes (Basic Level)*

An owner of the requirements development and management processes is needed in order to maintain the processes and keep them up-to-date. He also makes sure that the processes are followed and ensure that support and training are available when needed. Although in this case, the process does not seem to change much, it is still a good idea to delegate this task to someone so that he could reinforce it.

3. MPA REQUIREMENTS PROCESS MANAGEMENT

The number of completed and inapplicable actions according to each Uni-REPM level in Requirements Process Management compared to the total actions:

Level	Completed actions	Inapplicable actions	Total actions
Basic	6	0	7
Intermediate	1	0	6
Advanced	1	0	1



General
In this MPA, the project resides on level 0. It can reach level 1 – Basic by completing one more action and Level 2 - Intermediate with five additional actions. It is well-noticing that there is no model lag in this MPA.

Suggestions for improvement
In order to finish the Basic level, one additional action has to be completed:

- *PM.CM.a1 Manage Versions of Requirements (Basic Level)*

The subject specified that version control was not used in the project. However, version management would benefit the project significantly. Especially in the case the customers were involved quite often as the subject mentioned, version control would help to manage different sets of requirements while changes occur due to customers' expectation.
In case the company wants to move to Intermediate level, a huge step must be performed including 5 actions:

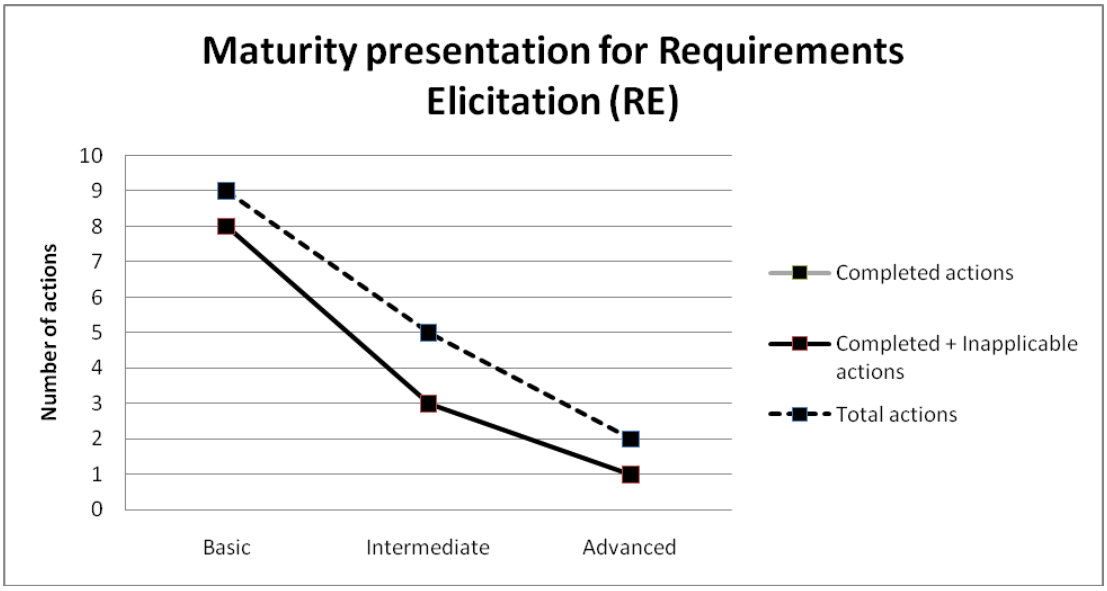
- *PM.CM.a3 Define a Process for Managing Change and Evolution (Intermediate level)*
- *PM.CM.a4 Track change requests (Intermediate level)*
- *PM.RT.a3 Define traceability policies (Intermediate level)*
- *PM.RT.a4 Document Requirements' Relation (Intermediate level)*
- *PM.RT.a5 Document Impact of Requirement on Other Artifacts (Intermediate level)*

Most of the incomplete actions are related to tracing and change control. It would save a lot of effort for the company to have a defined procedure with all the policies to manage changes occurring in most of the meetings. The subject described that most of the changes were approved by the manager. However, benefit from changes is just one aspect. The other aspects such as impacts and risk on the system should also be considered. In order to perform this consideration, tracing policies are required and relations among requirements are needed. This set of actions once implemented would benefit the subject on saving effort and resource on changes.

4. MPA REQUIREMENTS ELICITATION

The number of completed and inapplicable actions according to each UNi-REPM level in Requirements Elicitation compared to the total actions:

Level	Completed actions	Inapplicable actions	Total actions
Basic	8	0	9
Intermediate	3	0	5
Advanced	1	0	2



General
In this MPA, the project resides in level 0. However, it only takes one more action to complete the Basic level and two more to achieve Intermediate level. The model lag does not exist in this MPA as there is no inapplicable action.

Suggestions for improvement
In order to reach Basic level, only one action has to be completed:

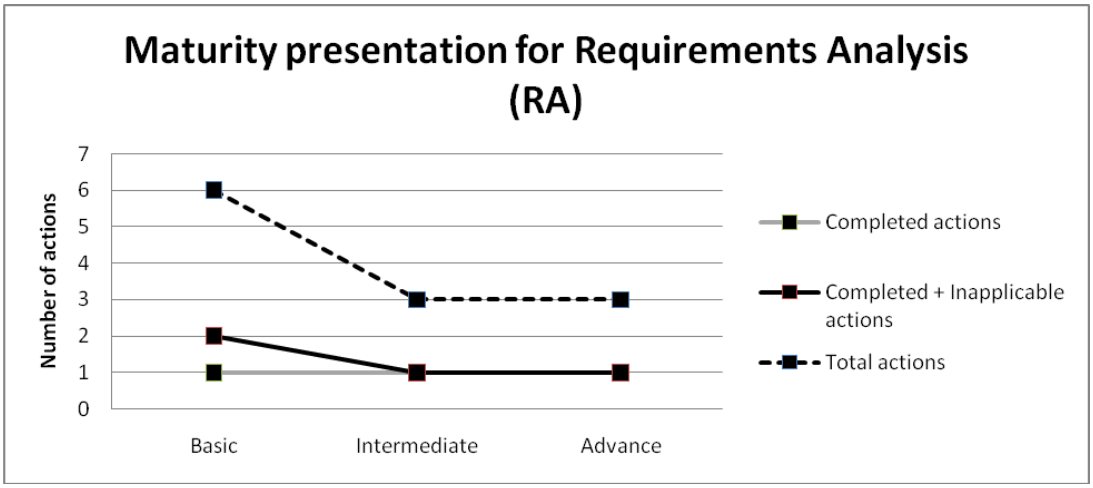
- *RE.GA.a1 Elicit Quality Requirements (Basic Level)*

Usually companies tend to overlook the quality requirements as they do not directly related to the functional requirements. However, ignoring the quality requirements will back fired the companies later as the customer may not be satisfied with the performance, reliability aspects etc. of the system. This is also the same case with the current project under assess. As the quality requirements will have a huge impact on the system, they tend to ignore them. In order to prevent customer disappointment and major rework or product failure, quality aspects that are crucial to the system need to be identified.

5. MPA REQUIREMENTS ANALYSIS

The number of completed and inapplicable actions according to each Uni-REPM level in Requirements Analysis compared to the total actions:

Level	Completed actions	Inapplicable actions	Total actions
Basic	1	1	6
Intermediate	1	0	3
Advanced	1	0	3



General

In this MPA, the project resides on level 0. It can reach level 1 – Basic by completing four more actions and Level 2 - Intermediate with two additional actions. The model lag is very small in this MPA with only one Inapplicable action.

- *RA.QA.a1. Analyze for double and missing requirements*

The reason was that the project mainly derived results from an existing product, hence a huge part of requirements were reused and it was not necessary to identify the double and missing requirements in this set.

Suggestions for improvement

In order to finish the Basic level, the following four additional actions have to be completed:

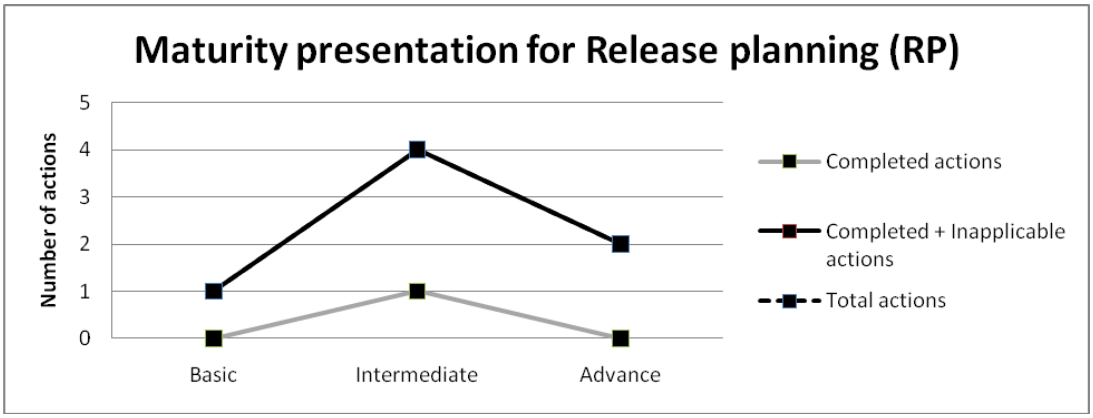
- *RA.QA.a2 Analyze for Ambiguous Requirements (Basic Level)*
- *RA.QA.a3 Analyze for Correctness of Requirements (Basic Level)*
- *RA.QA.a4 Analyze for Testability of Requirements (Basic Level)*
- *RA.PS.a1 Create Prototype (Basic Level)*

Requirements are usually collected from different sources, hence present different levels of expectations. Therefore, it is always necessary to consider the quality attributes of requirements. In this particular process, the requirements are gathered very often from customers (every week) but were never analyzed in the process regarding the mentioned perspectives. This may lead to increasing cost due to the reason that different teams need to communicate more often to clarify requirements. Besides, prototype was not used in the process, which is a very good way to align the understandings among different teams, and also with customer. These practices once implemented together will improve the productivity of the process systematically.

6. RELEASE PLANNING

The number of completed and inapplicable actions according to each Uni-REPM level in Release Planning compared to the total actions:

Level	Completed actions	Inapplicable actions	Total actions
Basic	0	1	1
Intermediate	1	3	4
Advanced	0	2	2



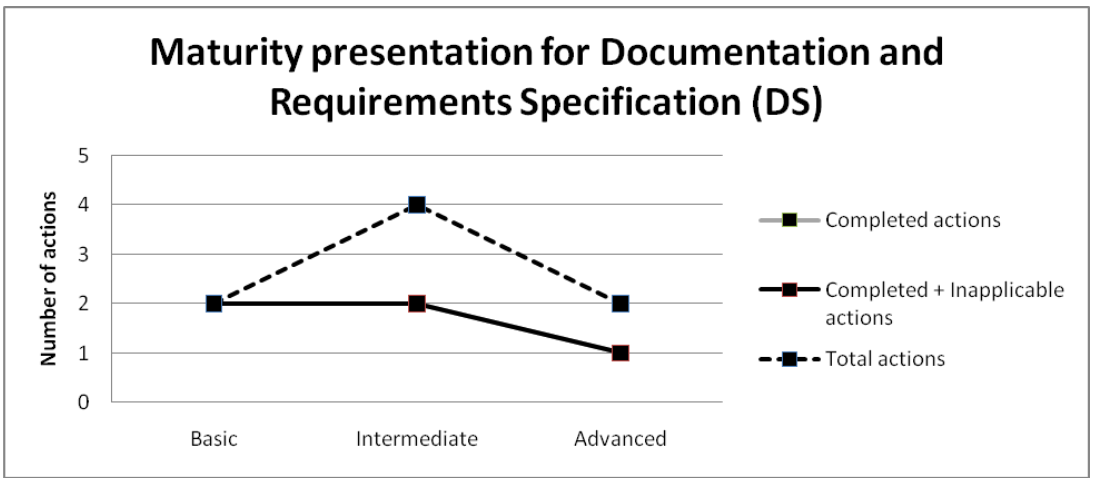
General

According to the interviewee, the project is mainly bespoke in which the customer is specified and they keep the importance in release decisions. Hence, most of the practices in this MPA are inapplicable. The only practice performed was "RP.GA.a2. Involve different perspectives in release planning". This action was done in the meetings with customer. Although decisions were made by the customers, involving in this step would reduce a lot of effort to understand the decisions and reduce risk for the process.

7. MPA DOCUMENTATION AND REQUIREMENTS SPECIFICATIONS

The number of completed and inapplicable actions according to each Uni-REPM level in Documentation and Requirements Specifications compared to the total actions:

Level	Completed actions	Inapplicable actions	Total actions
Basic	2	0	2
Intermediate	2	0	4
Advanced	1	0	2



General

From the chart, the project achieves Level 1 – Basic in Documentation and Requirements Specification MPA. So as to move to the higher maturity level - Intermediate, only two more actions need to be completed. It is interesting to notice that no model lag exists in this MPA.

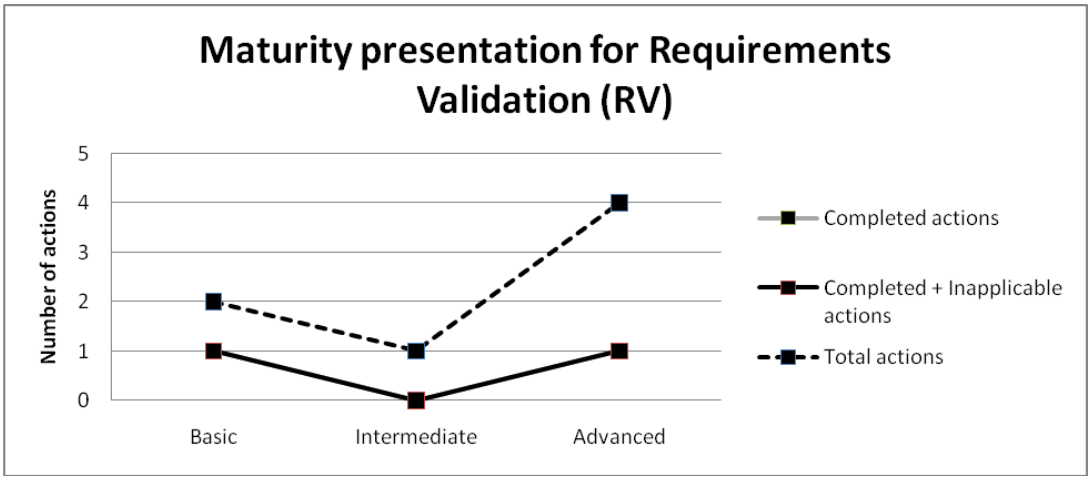
Suggestions for improvement
In order for the whole process to be consistent and stable, all actions in Basic level have to be fulfilled. As for this MPA, this has been achieved. If the company looks for a more mature process, they can consider implementing two more actions in the Intermediate level.

- *DS.GA.a3 Define Requirements States (Intermediate Level)*
- *DS.GA.a4 Document Requirements Rationale (Intermediate Level)*

8. MPA REQUIREMENTS VALIDATION

The number of completed and inapplicable actions according to each Uni-REPM level in Requirements Validation compared to the total actions:

Level	Completed actions	Inapplicable actions	Total actions
Basic	1	0	2
Intermediate	0	0	1
Advanced	1	0	4



General
In this MPA, the project resides on level 0. It can reach level 1 – Basic by completing one more action and Level 2 - Intermediate with one additional action. Once again, there is no model lag in this MPA.

Suggestions for improvement
In order to reach Basic level, one additional action has to be completed:

- *RV.GA.a2 Use Checklist to Ensure Quality of Requirements (Basic Level)*

There is an unofficial checklist created by the staff but it was not in use due to lack of time. Checklist draws the participants' attention to the certain aspects of the requirements document as well as the frequently encountered problems. Therefore, it helps to alleviate the risk of overlooking some parts of the requirements. Moreover, this action does not require much extra resource and easy to implement.

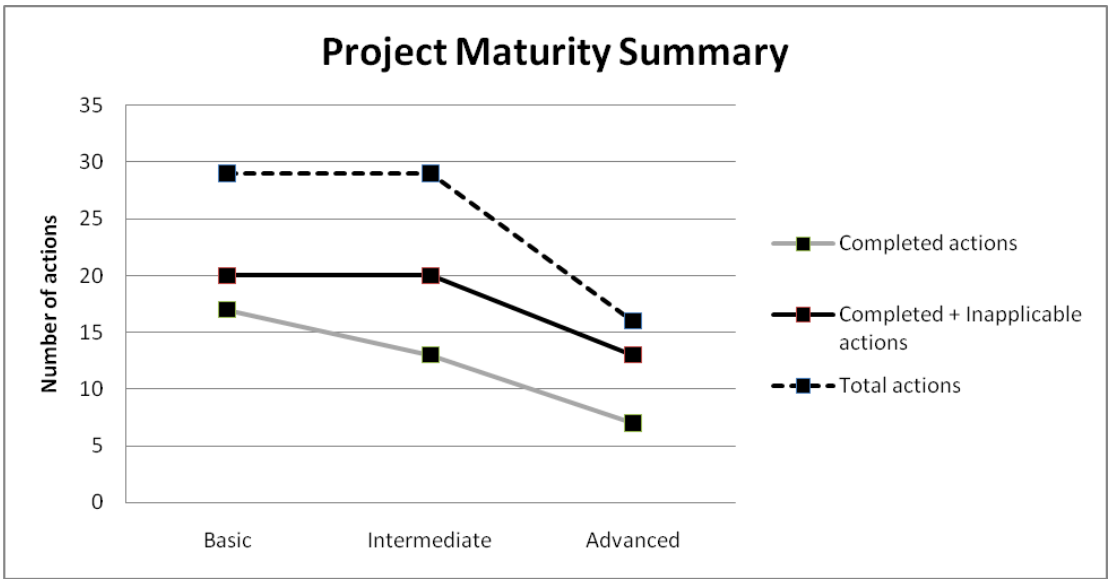
APPENDIX E – PROJECT EVALUATION RESULT 2

The company domain is software industry and the interviewee belongs to the Supporting team esp. engineering department business unit. Regarding the project under assessment, it involves programming the interface for the robot hardware. There are five team members working in seven months. The project is placed in the bespoke context as there is a specific customer.

1. PROCESS MATURITY OVERVIEW

The result for maturity assessment of the whole requirements process using Uni-REPM is demonstrated in the following diagram.

Level	Completed actions	Inapplicable actions	Total actions
Basic	17	3	29
Intermediate	13	7	29
Advanced	7	6	16



General

The project according to the interviewee is pure bespoke project. The product of this project is specifically built for a specified customer. This is also the reason that a significant number of actions are inapplicable including the release planning MPA (where a large portion of inapplicable actions resides). In the whole process, communication was very well considered, especially with customers. The interviewee told that they had meetings with customers every time changes occur and everyday within the team to exchange ideas and problems. It is also necessary to mention that the product of this project was developed under high pressure and performance was considered as the most important factor. This may be the reason why most of other requirements quality factors were overlooked in the project. Documentation was fairly well-considered. Templates and models were made use in the process. Overall, the whole requirements engineering process of this project resides at level 0. A quite big step of improvement (with 10 additional actions) needs to be performed in order to bring the process up to Basic level.

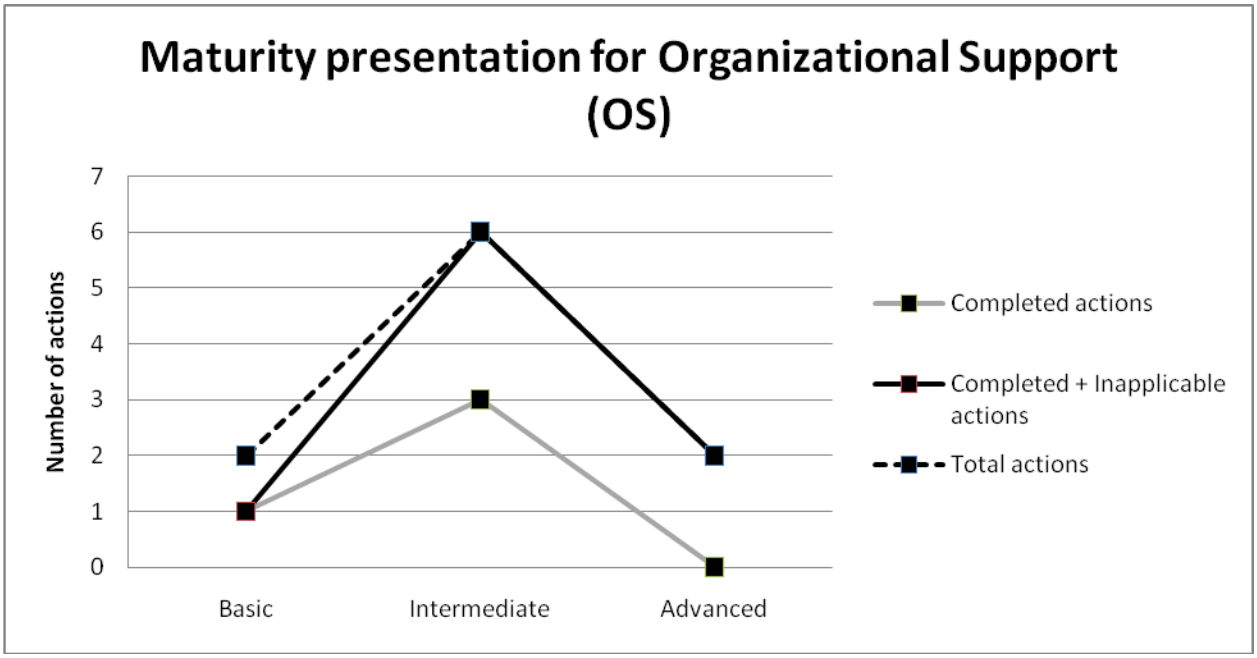
From the diagram, model lag seems to be huge in the diagram. However, the main reason is because the whole MPA Release Planning was considered inapplicable in this case.

Dismissing Release Planning actions, the model lag is fairly small. This convinces that the model is quite closed to real works.

2. MPA ORGANIZATION SUPPORT

The number of completed and inapplicable actions according to each Uni-REPM level in Organizational Support compared to the total actions:

Level	Completed actions	Inapplicable actions	Total actions
Basic	1	0	2
Intermediate	3	3	6
Advance	0	2	2



General

From the chart, we concluded that the project resided at level 0 in MPA Organizational Support. In order to achieve Basic level, one additional action will need to be completed. It is interesting to point out that although not all Basic actions are completed; all the actions in Intermediate and Advanced levels are fulfilled.

We noticed that there were five inapplicable actions:

- *OS.RR.a3 Define Roles and Responsibilities for Release Planning (Intermediate Level)*
- *OS.RR.a5 Define Roles and Responsibilities for Product Management (Advanced Level)*
- *OS.S.a1 Define Product Strategies (Intermediate Level)*
- *OS.S.a2 Define Product Roadmaps (Intermediate Level)*
- *OS.S.a3 Communicate Strategies in Organization (Advanced Level)*

They are considered Inapplicable because the product of the project is built according to the specific client demand. Therefore, the release planning and product management activities are not relevant in the current project. The model lag which consists of five actions are considered to be moderate.

Suggestions for improvement

The only one additional action that has to be completed is:

– *OS.GA.a1 Create a Product-wide Glossary of Terms (Basic Level)*

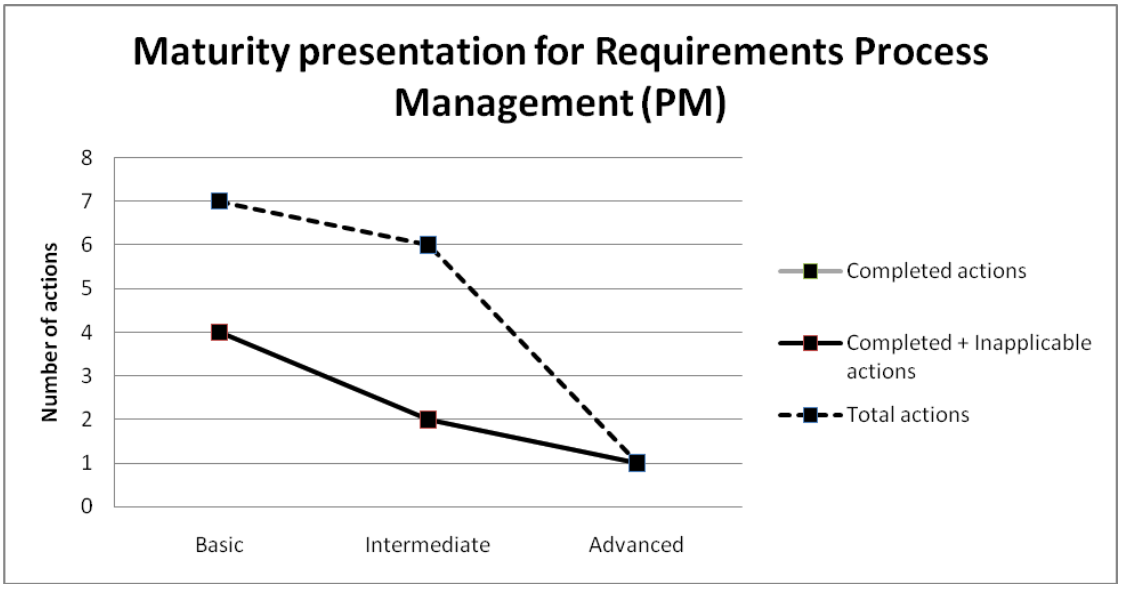
There are many different users of the requirements documents ranging from the managers, developers to customers. And because of their different backgrounds, they may have different perceptions about a certain thing. Therefore, In order to unite everyone's understanding of the requirements documents, there should be a term definition documented in a central place that covers everything from domain specific terms to product/project acronyms.

For this particular MPA, by completing this action, it not only achieves Basic level but also the Intermediate and Advanced levels.

3. MPA REQUIREMENTS PROCESS MANAGEMENT

The following table and figure demonstrate the maturity assessment of this MPA in the process using Uni-REPM.

Level	Completed actions	Inapplicable actions	Total actions
Basic	4	0	7
Intermediate	2	0	6
Advanced	1	0	1



General

The diagram shows that, in this MPA, the process resides at level 0 which means that it is still an ad-hoc process. In order to reach Basic level, it needs to complete three more actions. The next level will require even bigger step with 5 actions. However, the highest level is finished in this process.

Model lag in this case does not exist which means the model closely presents the real works.

Suggestions for improvement

The three following additional actions need to be completed in order to bring the process up to Basic level

– *PM.GA.a1 Define and Maintain Requirements Development and Management Processes (Basic Level)*
– *PM.CM.a2 Baseline Requirements (Basic Level)*
– *PM.RT.a2 Document Requirements' Source (Basic Level)*

Defining the process can help the managers to effectively control, monitor and keep track on the process. Besides, it helps all the team members clearly know steps to follow in the process. Depending on the size of the project, there could be a huge number of necessary steps for requirements process. It is also important to document this information in a central place so that any involved team member could easily access it.

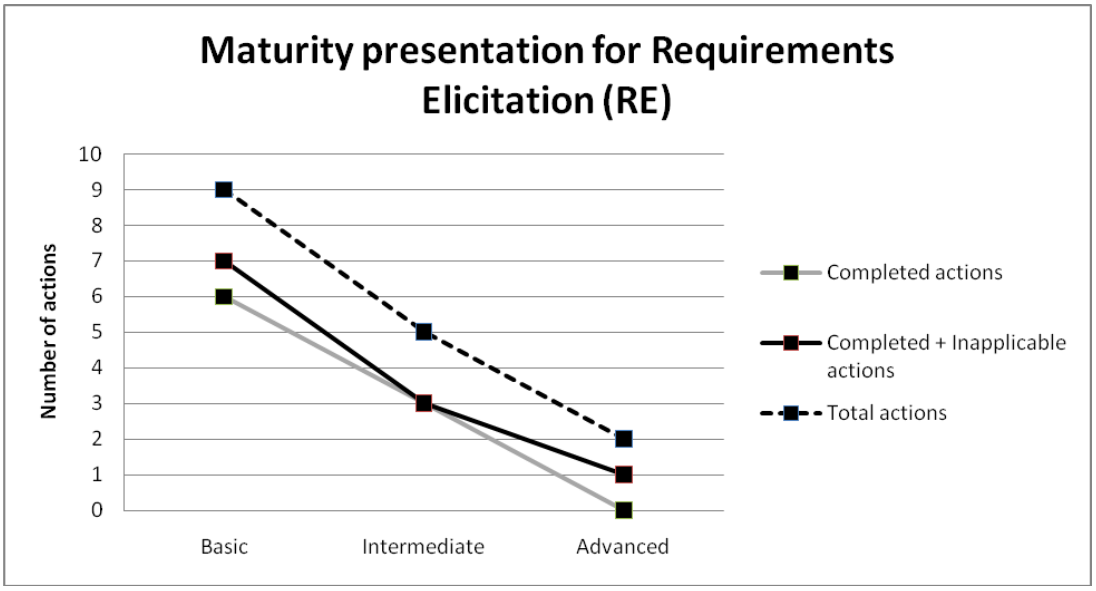
Defining a fixed set of requirements (also called baseline) is important. Due to the fact that expectation of customers often change rapidly, it is very necessary to fix them at some point so that there will be a set of agreed requirements to work on. By implementing this action, the process would be able to avoid dealing with messy systems and solve the problem from different parts.

Documenting requirements source will significantly help the requirements tracing when change occurs. This action once implemented will provide clear information on whom and where to clarify or refer in order to decide and apply changes.

4. MPA REQUIREMENTS ELICITATION

The number of completed and inapplicable actions according to each Uni-REPM level in Requirements Elicitation compared to the total actions:

Level	Completed actions	Inapplicable actions	Total actions
Basic	6	1	9
Intermediate	3	0	5
Advanced	0	1	2



General

In this MPA, from the diagram we can see that the project resides in level 0. It takes two additional actions to complete the Basic level and two more to achieve Intermediate level. This MPA is considered a "weak" spot of the whole process and needs some attention to improve it.

The model lag in this MPA is small as there are two inapplicable actions:

- *RE.SI.a3 Identify Other Requirements Sources (Basic Level)*

This is not relevant for this project because everything was provided including several hardware which is ready to use. The requirements come from the customer only.

- *RE.GA.a7 Reuse Requirements (Advanced Level)*

The company always considers the reuse of requirements as well as other artifacts. However, because it was a total brand new project, there is no other similar system created in the same domain. Therefore, reuse was not considered in this situation.

Suggestions for improvement

Based on the overall view of the project throughout the MPAs, we believe that it is beneficial for the project to aim for Intermediate level. In order to reach the Intermediate level, four additional actions have to be completed:

- *RE.DC.a3 Elicit Information about System's Business Process (Basic Level)*
- *RE.DC.a4 Elicit Information about System's Operational Domain (Basic Level)*
- *RE.DC.a6 Consider Sociopolitical Influences on Requirements Sources (Intermediate Level)*
- *RE.GA.a3 Let Business Concern Guide Focus of Elicitation (Intermediate Level)*

The reason for developing a system is usually because it will contribute to the business of the organization by either making the normal process more efficient or cost-effective. Therefore, gathering general information about the constraints in business process in which the system will function helps to define right and useful system to build.

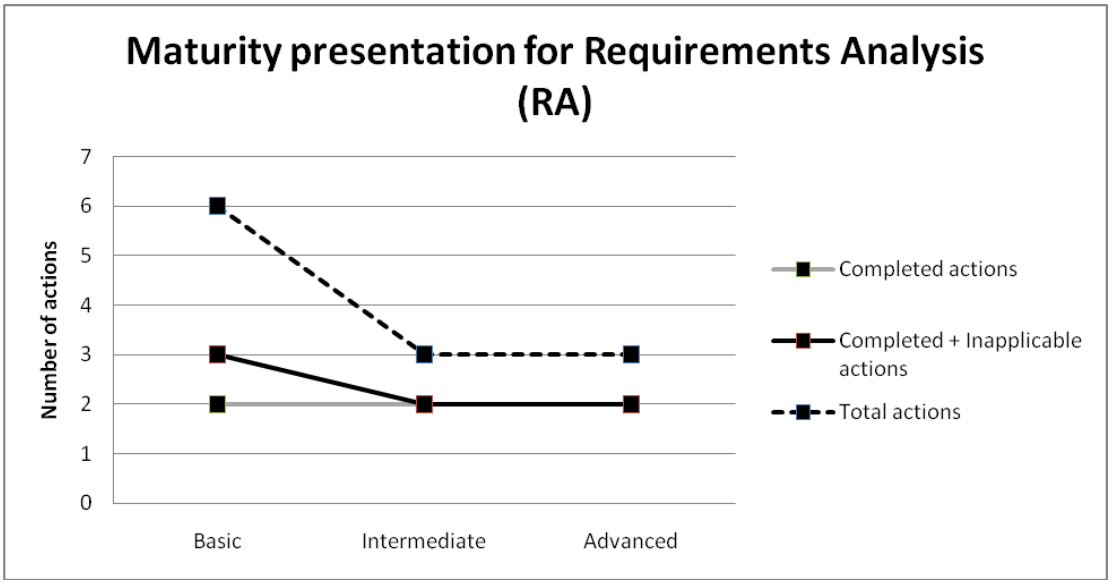
Besides, not only the direct business affects the system. Other business processes which are supported by the system being developed also impose requirements and constraints on the system. Hence, elicitation process should take into account this information as well.
Software systems are not built in a vacuum environment so they can be adversely affected by the human factors. Organizational and political factors can affect or conceal the real system requirements. People may have different hidden agenda and not all of them are willing to contribute to the system being developed. Being aware of these factors can help to understand the real reason for including the requirements.

The business concerns helps to align the elicitation resources, the time and money spent on elicitation with overall goals for the product. This orientation is important as it keep the project stay on track.

5. MPA REQUIREMENTS ANALYSIS

The result of this MPA using Uni-REPM assessment is drawn in the following table and figure.

Level	Completed actions	Inapplicable actions	Total actions
Basic	2	1	6
Intermediate	2	0	3
Advanced	2	0	3



General
From the chart we could see that this MPA resides at level 0 since most of actions in level 1 were not completed. However, it is interesting to point out that level 2 and level 3 are almost completed in the process of this MPA.

In order to reach Basic level (level 1), three more actions need to be done in the process. However, only one more action at level Inter-mediate and one more at level Advanced (level 3) need to be finished to reach the highest maturity level.

Suggestion for improvement
The three missing actions at basic level in this MPA are:

- *RA.QA.a1 Analyze for Missing and Double Requirements (Basic Level)*
- *RA.QA.a3 Analyze for Correctness of Requirements (Basic Level)*
- *RA.QA.a4 Analyze for Testability of Requirements (Basic Level)*

As mentioned in the first part, this project highly focuses on the quality aspect of the product since its product was expected to work in strict environments. Therefore, these three actions are highly recommended to be finished. Detecting double and missing requirements will help to improve the productivity of the process by preventing multiple meetings for clarification.

Besides, inspecting incorrect and testable requirements will help to improve the quality aspect of the product. The reason the interviewee provided us for skipping these activities was that they had meetings every week with customers to align the expectations, hence there was no misunderstanding or missing requirements occurs at late stage. However, implementing these actions will help the team to save a lot of effort and cost holding the meetings every week as well as testing the product in the other phases.

6. MPA RELEASE PLANNING

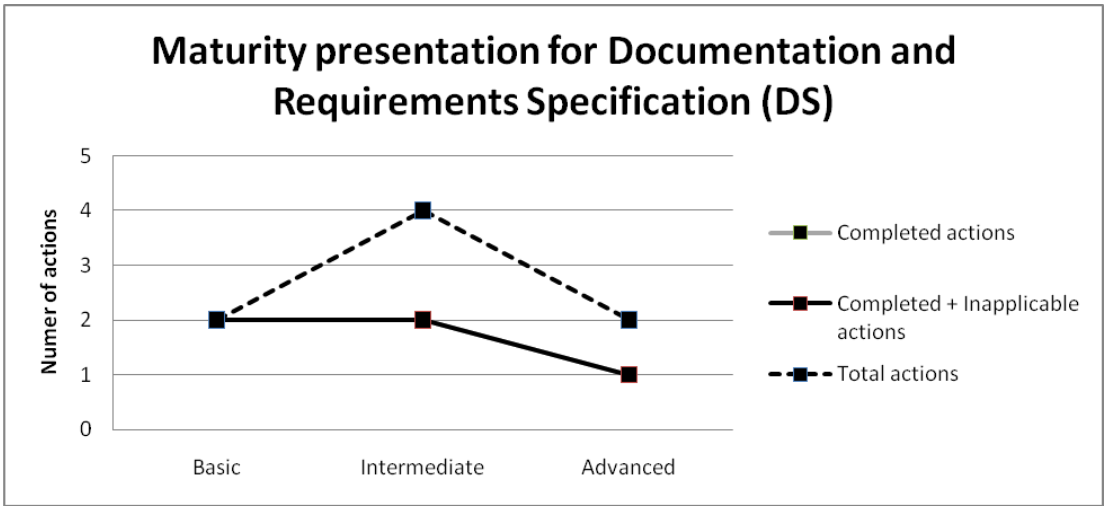
The project followed bespoke development approach, hence all of the actions in this MPA were not performed in the process. Since the reason for this ignore is the particular characteristic of the project, we agreed on marking these actions as inapplicable actions.
MPA Release Planning

As described in the first part, this project is purely bespoke, hence the actions in this MPA are all inapplicable. Therefore, this MPA is not considered in the improvement suggestion.

7. MPA DOCUMENTATION AND REQUIREMENTS SPECIFICATIONS

The number of completed and inapplicable actions according to each Uni-REPM level in Documentation and Requirements Specifications compared to the total actions:

Level	Completed actions	Inapplicable actions	Total actions
Basic	2	0	2
Intermediate	2	0	4
Advanced	1	0	2



General

From the chart, the project achieves Level 1 – Basic in Documentation and Requirements Specification MPA. In order to reach to Intermediate level, two more actions should be performed. It is interesting to notice that no model lag exists in this MPA.

Suggestions for improvement

In order for the whole process to be consistent and stable, all actions in Intermediate level have to be fulfilled. This means that two more actions below need to be completed:

- *DS.GA.a3 Define Requirements States (Intermediate Level)*
- *DS.GA.a4 Document Requirements Rationale (Intermediate Level)*

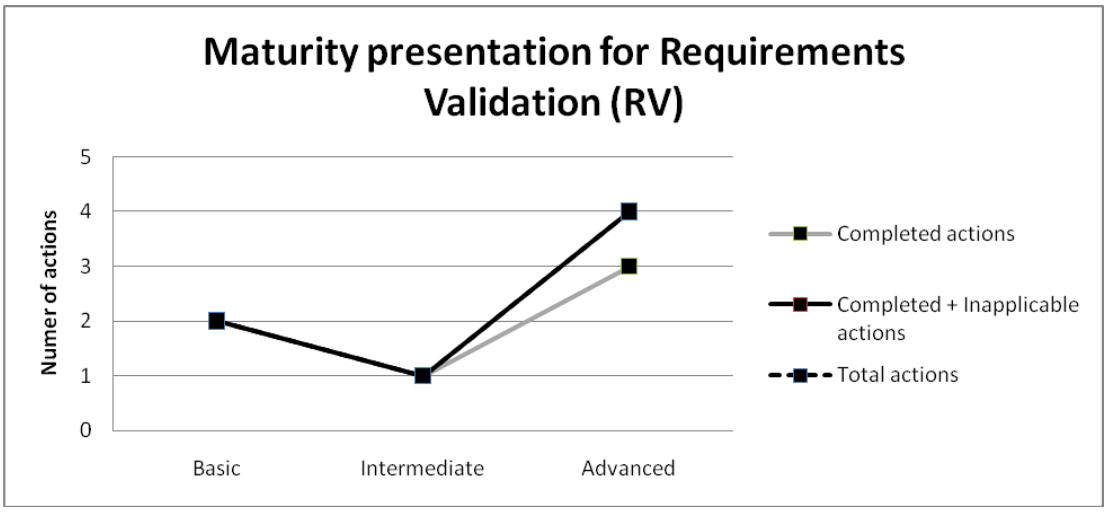
Defining requirements states can help to manage and keep track of requirements better. Depending on each company and project needs, the number of pre-defined states can be big or small. Tracking requirements states help to monitor the requirements and project progress more accurately.

Requirements rationale consists of vital information of why a requirement is included and what function it has. This brings the advantage of maintaining the initial meanings of the requirements in case those initially defined the requirements have left the company. Moreover, it will help problem expert to check if the requirements are consistent with the problem being solved.

8. MPA REQUIREMENTS VALIDATION

The number of completed and inapplicable actions according to each Uni-REPM level in Requirements Validation compared to the total actions:

Level	Completed actions	Inapplicable actions	Total actions
Basic	2	0	2
Intermediate	1	0	1
Advanced	3	1	4



General

In this MPA, the project resides on level 3 – Advanced, which means it has reached the most mature level in Requirements Validation.

There is a relatively small model lag in this MPA with one inapplicable action:

- *RV.GA.a7 Define Acceptance Criteria and Acceptance Tests (Advanced level)*

In this case, the project team defines the acceptance test instead of the client. Based on the result of the acceptance test, the client will decide whether they are satisfied with the product. As this action is done by the project team, we considered it as Inapplicable.

Suggestions for improvement

As all the actions in all three levels are fulfilled, the quality of the requirements is assured. This is the strong point of the whole process. Hence, the team can focus its effort on other MPAs to maintain a well-rounded process.

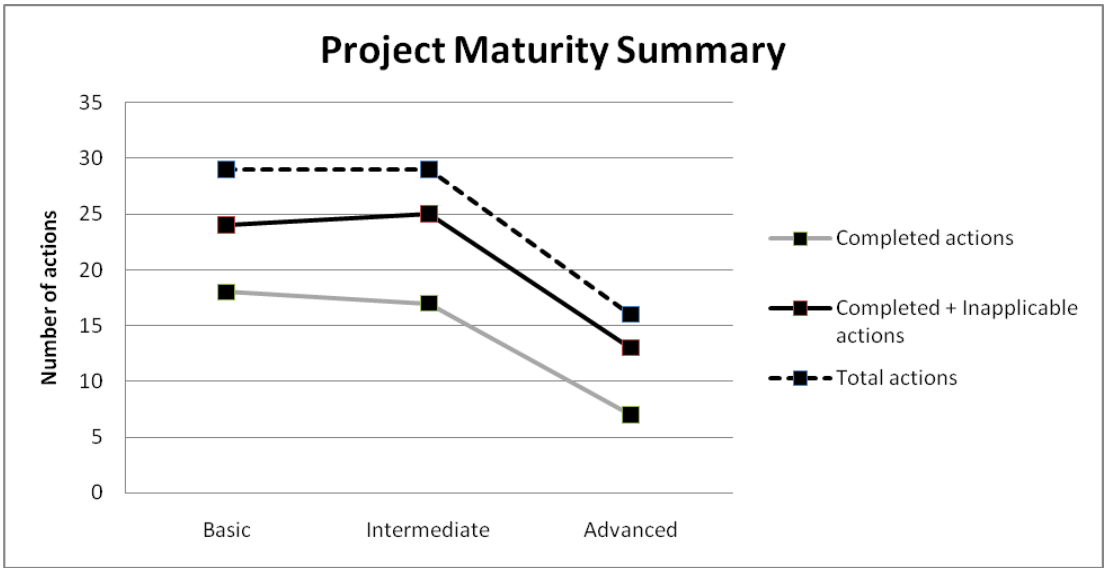
APPENDIX F – PROJECT EVALUATION RESULT 3

The company domain is in Insurance system, banking system and fund management system. There are around 500 employees working in the company. The project was to develop a system for a small insurance company in Singapore. The system was derived on a generic product developed by the company and new requirements were created by comparing the gap between the expectations of the customer and the existing functionality of the system. The new requirements were implemented by the customization and localization teams. The project lasted for one and a half year and the requirement team comprised three to six people.

1. PROCESS MATURITY OVERVIEW

Total actions overview in the process is summarized in the following table according to levels.

Level	Completed actions	Inapplicable actions	Total actions
Basic	18	6	29
Intermediate	17	8	29
Advanced	7	6	16



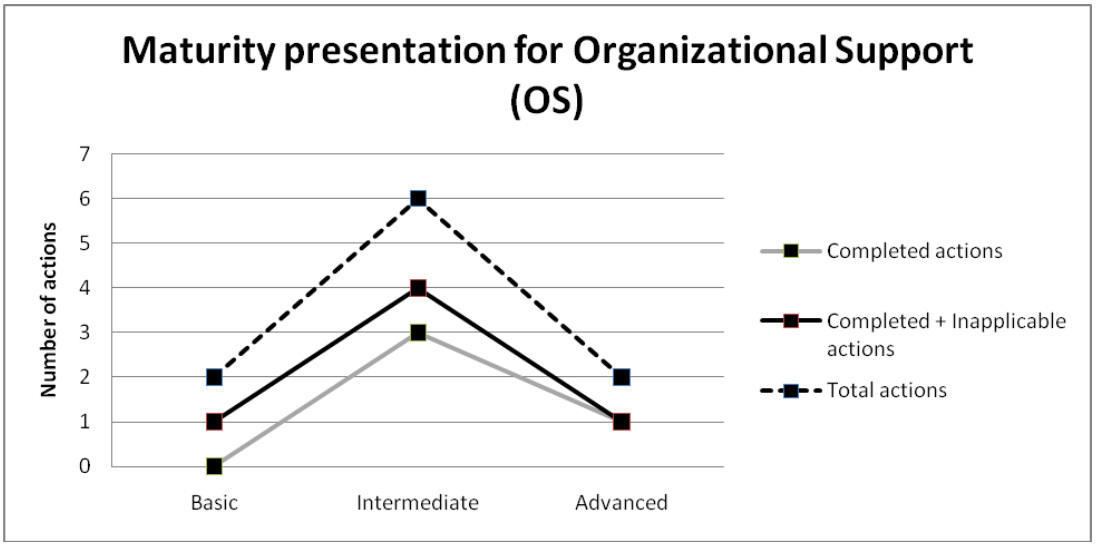
The project according to the interviewee is pure bespoke project. The product of this project targets for a specified customer and part of it is reused from the old product. This is also the reason that a quite large number of actions are inapplicable (20 actions) including the release planning MPA (where a large portion of inapplicable actions resides). In the whole process, documentation was very well considered. Templates and models were made use in the process. However, requirements management and requirements analysis tend to be overlooked Overall, the whole requirements engineering process of this project resides at level 0. A fairly small step of improvement (with 5 additional actions) needs to be performed in order to bring the process up to Basic level.

From the diagram, model lag seems to be huge in the diagram. However, the main reason is because the whole MPA Release Planning was considered inapplicable in this case. This convinces that the model is quite closed to real works.

2. MPA ORGANIZATION SUPPORT

The number of completed and inapplicable actions according to each Uni-REPM level in Organizational Support compared to the total actions:

Level	Completed actions	Inapplicable actions	Total actions
Basic	0	1	2
Intermediate	3	1	6
Advance	1	0	2



General

From the chart, we conclude that the project resides at level 0 in MPA Organizational Support. In order to reach Basic level, 1 additional action will need to be completed.

The model lag is considered to be small as it consists of only 2 actions. This shows that the MPA has good applicability in the real setting.

The two inapplicable actions are:

- *OS.GA.a1 Create a Product-wide Glossary of Terms (Basic Level)*

The company has an internal glossary of terms but it is not released to the users. In this case, the subject felt that it was hard and useless to give the user the whole document and ask them to read it as they might not do so. Instead, the terms and examples were explained directly to the users in the meeting and it was more effective this way.

- *OS.GA.a2 Train personnel in Requirements Development and Management Processes (Intermediate Level)*

There is no formal training, only on-the-job training. In the subject's opinion, the requirements engineering skills are more of soft skills so they cannot be taught formally. The business analyst usually picks up the skills by following the senior ones.

Suggestions for improvement

In order to achieve Basic level, one action has to be completed:

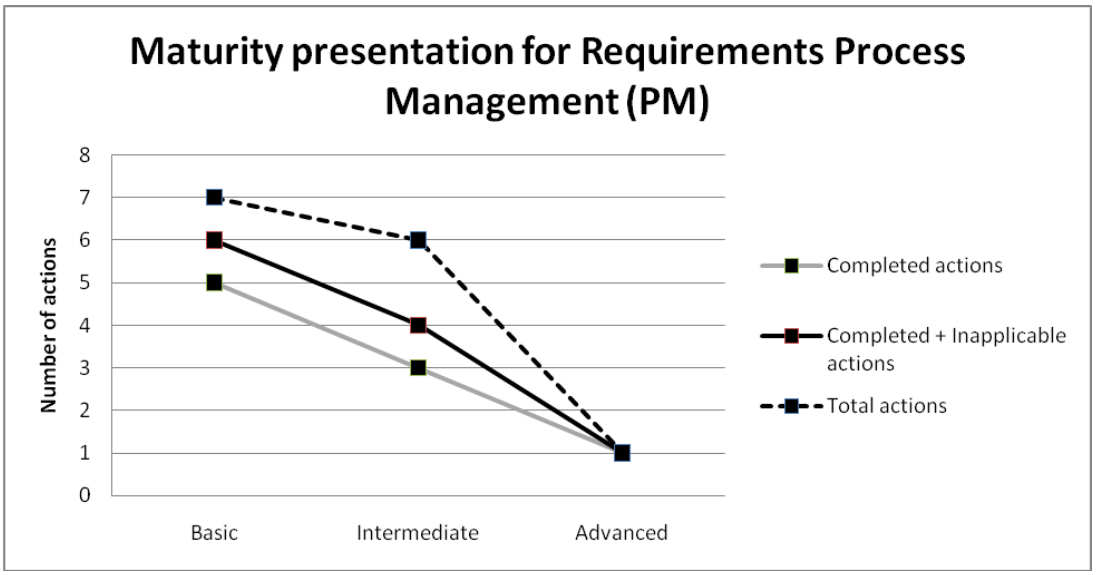
- *OS.RR.a1 Assign Owner(s) of Requirements Development and Management Processes (Basic Level)*

An owner of the requirements development and management processes is needed in order to maintain the processes and keep them up-to-date. He also makes sure that the processes are followed and ensure that support and training are available when needed. Although in this case, the process does not seem to change much, it is still a good idea to delegate this task to someone so that he could reinforce it.

3. MPA REQUIREMENTS PROCESS MANAGEMENT

The maturity assessment result of this MPA using Uni-REPM is described in the following table and diagram.

Level	Completed actions	Inapplicable actions	Total actions
Basic	5	1	7
Intermediate	3	1	6
Advanced	1	0	1



General

The diagram shows that in this MPA, the process resides at level 0. In order to reach the Basic level, it is required to complete one additional action. And two more actions need to be finished to bring it up to Intermediate level. However, it is interesting to see that Advanced level is completed.

The model lag in this MPA is rather small, only two actions (out of 29) are considered inapplicable. Those are

- *PM.CM.a1 Manage Versions of Requirements (Basic Level)*
- *PM.RT.a3 Define traceability policies (Intermediate Level)*

The reason for skipping these two actions was that the project reused a significant part from a developed product. The additional project work is not huge and complicated hence versioning and tracing are not required.

Suggestion for improvement

The process in this area almost reaches Basic level which means it is quite closed to a structured process. The only missing action at level 1 is

- *PM.GA.a1 Define and Maintain Requirements Development and Management Processes (Basic Level)*

Defining the management process for requirements is very important, It helps the managers to control and monitor the process effectively; and the team members to be clearly aware of what to do and who to ask in next steps. This action is considered costly, however once it is implemented, it would significantly effect the productivity of whole process.

It is also necessary to mention that only two following additional actions need to be completed in order for the process to reach the highest maturity level – Advanced level

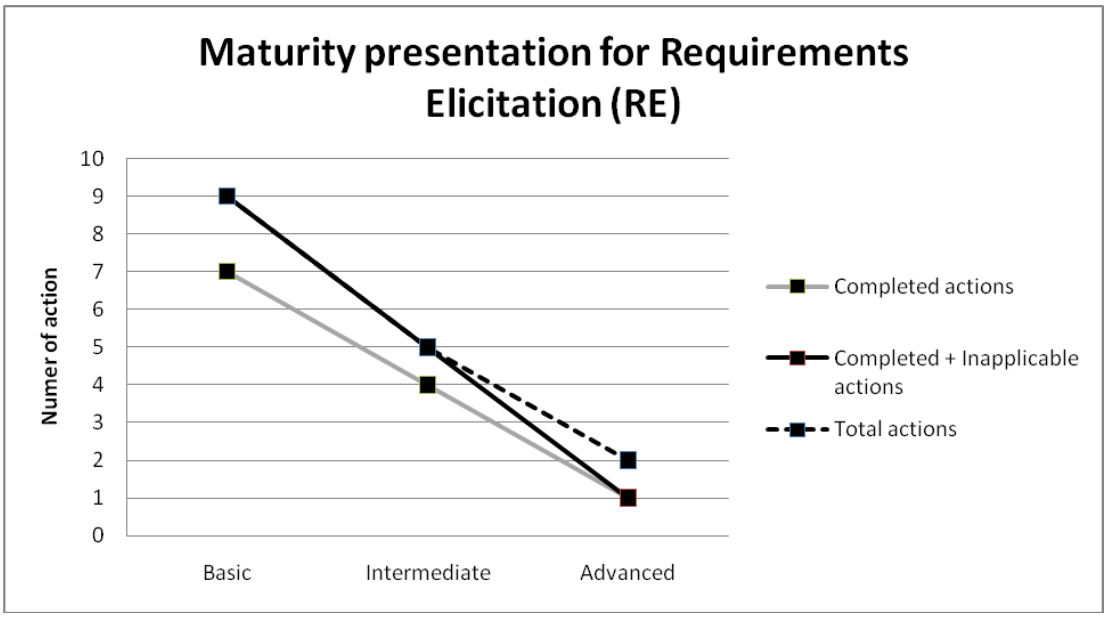
– *PM.RT.a4 Document Requirements' Relation (Intermediate Level)*
– *PM.RT.a5 Document Impact of Requirement on Other Artifacts (Intermediate Level)*

Both of the two actions are documentation related. It is quite common that the documentation activities are tent to be overlooked in processes. However, it is important to perform these actions in this case. Documenting the relations between requirements would help the requirement analyst save a lot of effort in tracing when changes occur. It is also important for the later phases when designers and developers consider which parts of the system will be implemented prior to the other ones. Besides, in some cases where the project product multiple releases of the products, it is very important to see the relations especially the restriction among requirements of the product. A similar reason is applied for the other action. Documenting impact of requirements on other artifacts will also help to keep track on and apply changes when they appear in later phases where more artifacts such as test cases, class diagrams, system model etc. are available.

4. MPA REQUIREMENTS ELICITATION

The number of completed and inapplicable actions according to each Uni-REPM level in Requirements Elicitation compared to the total actions:

Level	Completed actions	Inapplicable actions	Total actions
Basic	7	2	9
Intermediate	4	1	5
Advanced	1	0	2



General

Looking at the graph, the project resides in level 2 – Intermediate level for this Requirements Elicitation MPA. The model lag in this MPA is moderately small as there are 3 inapplicable actions.

The reasons for the following three inapplicable actions are:

– *RE.SI.a2 Distinguish between Different Types of Stakeholders (Basic level)*

Based on the characteristics of the functionalities, the team conducted interviews with specific stakeholders. They did not specify a lot of stakeholders' views into one single requirement.

– *RE.SI.a3 Identify Other Requirements Sources (Basic level)*

It was not necessary to get the requirement from other sources as they only obtained requirements directly from the stakeholders of the functionality.

– *RE.GA.a3 Let Business Concern Guide Focus of Elicitation (Intermediate Level)*

This is deemed inapplicable as it has little impact on the current system. The company wanted to make the system flexible enough to allow the customer to achieve their ever-changing business objectives.

Suggestions for improvement

In order for the whole process to be consistent and stable, all actions in Basic level have to be fulfilled. As for this MPA, this has been achieved. If the company wants to improve their process maturity, they can consider implementing one more action in the Advanced level:

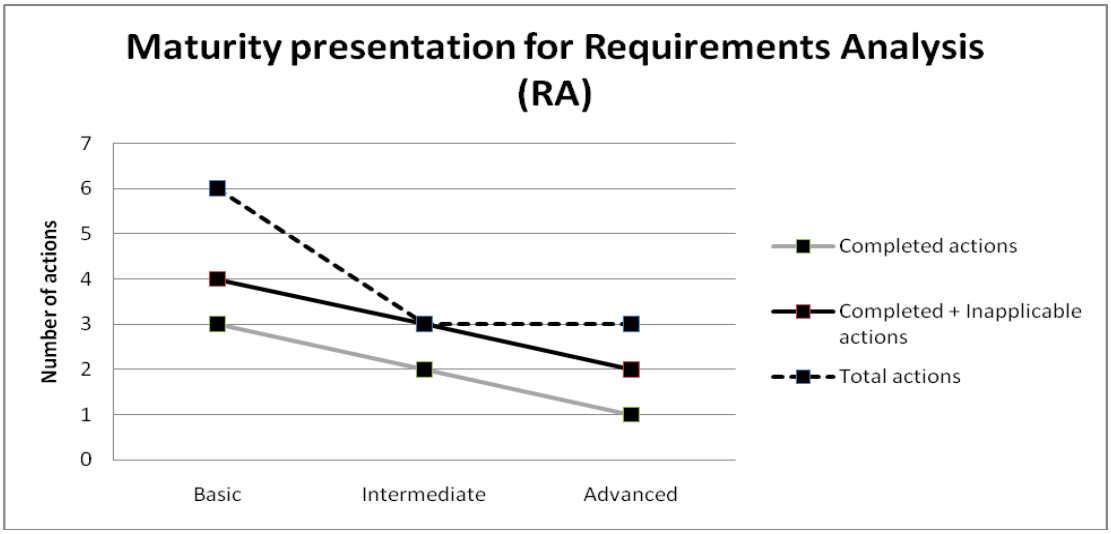
– *RE.GA.a6 Create Elicitation Channels for Requirements Sources (Advanced Level)*

Besides common channels and techniques of eliciting requirements, customer feedback and requests that can be turned into requirements can be captured in many different forms such as incident reports, idea feedback and suggestion. Moreover, company should enable elicitation channels for In-house stakeholders e.g. developers, testers to submit new requirements.

5. MPA REQUIREMENTS ANALYSIS

The following table and diagram draw the assessment result of this MPA of the process using Uni-REPM.

Level	Completed actions	Inapplicable actions	Total actions
Basic	3	1	6
Intermediate	2	1	3
Advanced	1	1	3



General

The diagram shows that the process in this MPA has not finish Basic level (2 missing actions). However, all actions in Intermediate level are finished and only one additional action need to be completed to get to the Advanced level once Basic level is done.

The model lag in this MPA is quite small considering that there are only 3 inapplicable actions (out of 12 actions).

– *RA.GA.a1 Perform Requirements Risk Analysis (Basic Level)*

- *RA.GA.a2 Perform Systematic Requirements Prioritization at Project-level (Intermediate Level)*
- *RA.PS.a2 Perform Systems Modeling (Advanced Level)*

The reason for skipping these actions is, once again, mainly because of the characteristic of the project in which most parts of the product were derived from an existing product.

Suggestion for improvement

As shown in the diagram, the two following actions need to be done in order for the process to finish Basic level

- *RA.QA.a4 Analyze for Testability of Requirements (Basic Level)*
- *RA.PS.a1 Create Prototype (Basic Level)*

Analyzing requirements under testability perspective will help to know whether the requirements are sufficient for implementing. It is also to find out requirements which are not defined how to be tested in later phases. Based on that, it is possible to ensure it is possible in later phase to check if the implemented system is what the customers expected.

Prototyping is another mean to test if the designed system is closed to what customers expected. It is also very useful to test if the requirements are correct and sufficient for implementation. Using prototype, helps to detect misunderstands and misalignments between two sides and prevent delays in later phases.

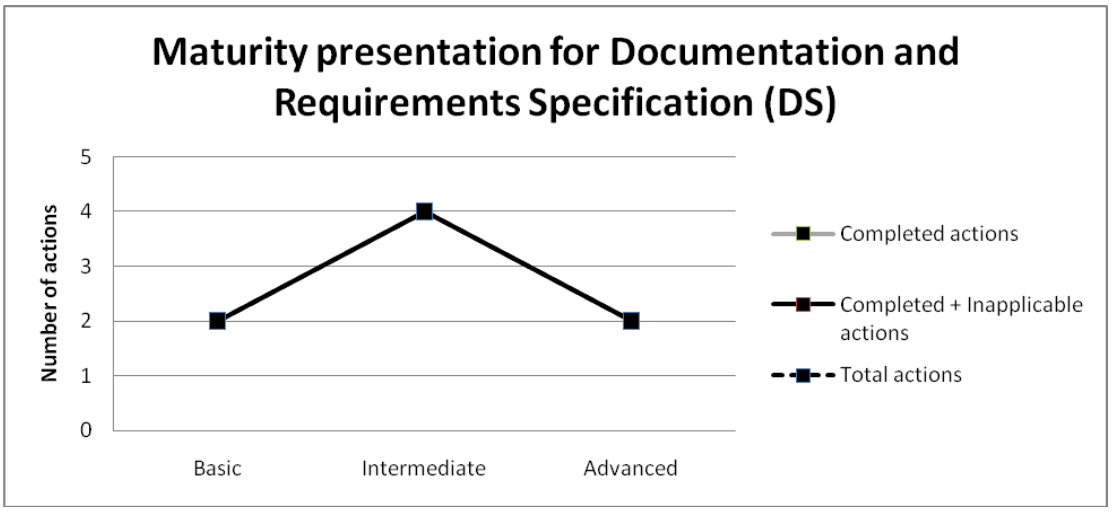
6. MPA RELEASE PLANNING

This MPA was marked as "inapplicable" in this particular case since the product of this project was built for a specified customer. Hence, all the actions in this MPA were not performed in the process.

7. MPA DOCUMENTATION AND REQUIREMENTS SPECIFICATIONS

The number of completed and inapplicable actions according to each Uni-REPM level in Documentation and Requirements Specifications compared to the total actions:

Level	Completed actions	Inapplicable actions	Total actions
Basic	2	0	2
Intermediate	4	0	4
Advanced	2	0	2



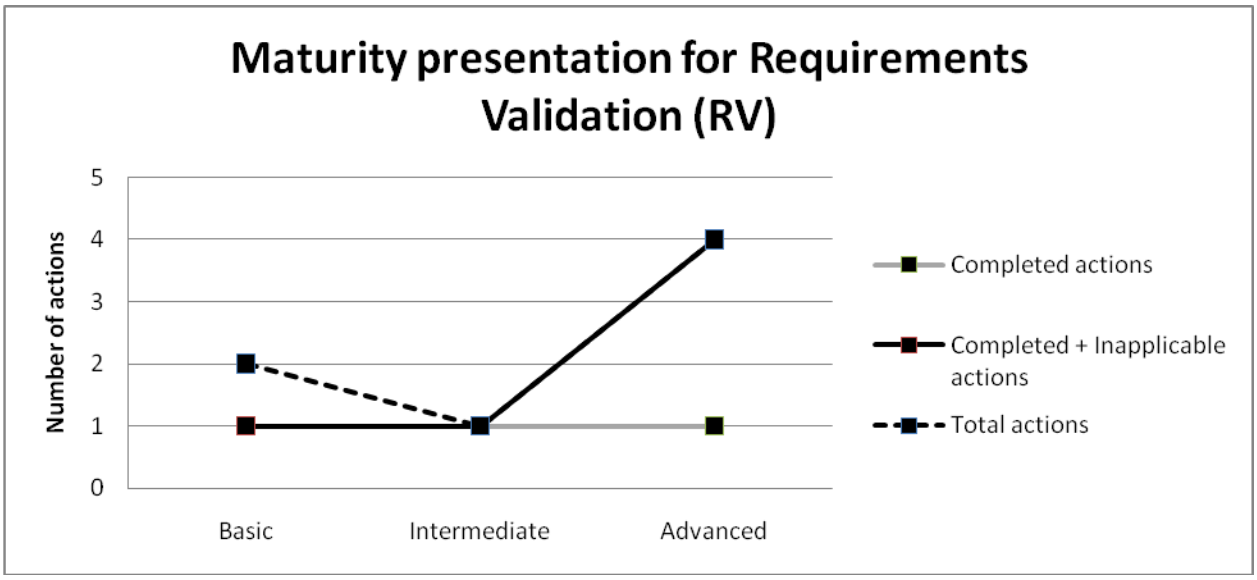
General
From the graph, the project achieves Level 3 – Advanced in Documentation and Requirements Specification MPA. It completed all the required actions and reached the most mature level in this aspect.
It is well-noticing that no model lag exists in this MPA. It proves the high applicability of the actions in this MPA.

Suggestions for improvement
As all the actions in all three levels are fulfilled, it is assured that requirements and other knowledge gathered during requirements engineering process are organized into consistent, accessible and reviewable documents. This is the strong point of the whole process. Hence, the team can focus its effort on other MPAs to maintain a well-rounded process.

8. MPA REQUIREMENTS VALIDATION

The number of completed and inapplicable actions according to each Uni-REPM level in Requirements Validation compared to the total actions:

Level	Completed actions	Inapplicable actions	Total actions
Basic	1	0	2
Intermediate	1	0	1
Advanced	1	3	4



General
In this MPA, the project resides on level 0. It can reach level 1 – Basic by completing one more action. It is interesting to point out that although not all Basic actions are completed; all the actions in Intermediate and Advanced levels are fulfilled.

A small model lag exists in this MPA with three inapplicable actions. The reasons for deeming the three actions inapplicable are:

- *RV.GA.a4 Organize Inspections (Advanced Level)*

They did not need to do this action as they performed other validation activities to ensure the quality of the requirements.

- *RV.GA.a5 Develop Preliminary Test Case or User Manual (Advanced Level)*

They usually used user manual for training and test cases for testing, but not for the purpose of assuring the quality of the requirements.

– *RV.GA.a6 Use System Model Paraphrasing for QA (Advanced Level)*

It is not necessary for them to convert the system model to natural language to verify them.

Suggestions for improvement

In order to attain Basic level, one additional action has to be completed:

– *RV.GA.a2 Use Checklist to Ensure Quality of Requirements (Basic Level)*

Checklist draws the participants' attention to the certain aspects of the requirements document as well as the frequently encountered problems. Therefore, it helps to alleviate the risk of overlooking some parts of the requirements. Moreover, this action does not require much extra resource and easy to implement. When fulfilling this action, the process will automatically reach the Advanced level as well as all other actions have been completed.

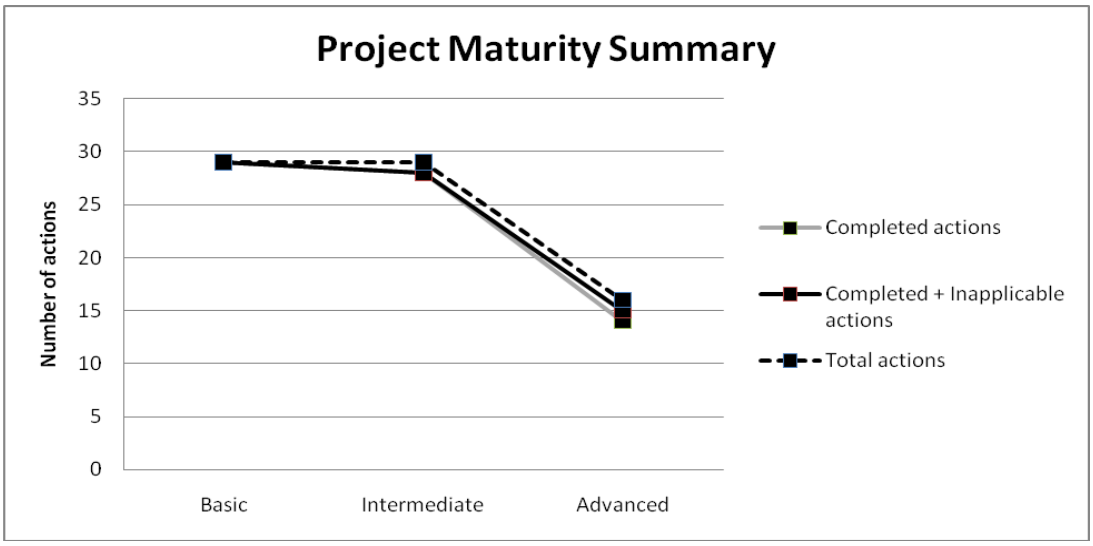
APPENDIX G - PROJECT EVALUATION RESULT 4

The company belongs to the Insurance domain with 800 staff and the IT business unit has 200 employees. The product of the project under evaluation is a Point of Sale system. The product is a system that encapsulates information of insurance product, receives information of clients and the insurance product they choose to buy, and generates benefit illustrations for clients. The system should also have the ability to generate risk profiles of clients based on the fact finding information obtained from them. This is an in-house system and the requirements come from user. The system is used across Singapore, Indonesia, Vietnam, Malaysia, China & Brunei. The project lasts for one year and involved one requirements engineering team of 10 people.

1. PROCESS MATURITY OVERVIEW

The result for maturity assessment of the whole requirements process using Uni-REPM is demonstrated in the following diagram.

Level	Completed actions	Inapplicable actions	Total actions
Basic	29	0	29
Intermediate	28	0	29
Advanced	14	1	16



From the diagram, the whole requirements engineering process of this project resides at level 1 – Basic level. Only one additional action needs to be implemented in order for the process to move to the next level, and one more action needs to be finished to reach the highest level. The process is considered quite mature with 4/7 MPAs reach the Advanced level. Effort was distributed quite broadly in all process areas. The only 2 actions were left behind are:

- *OS.RR.a5 Define Roles and Responsibilities for Product Management (Advanced level)*
- *RE.DC.a6 Consider Sociopolitical Influences on Requirements Sources (Intermediate level)*

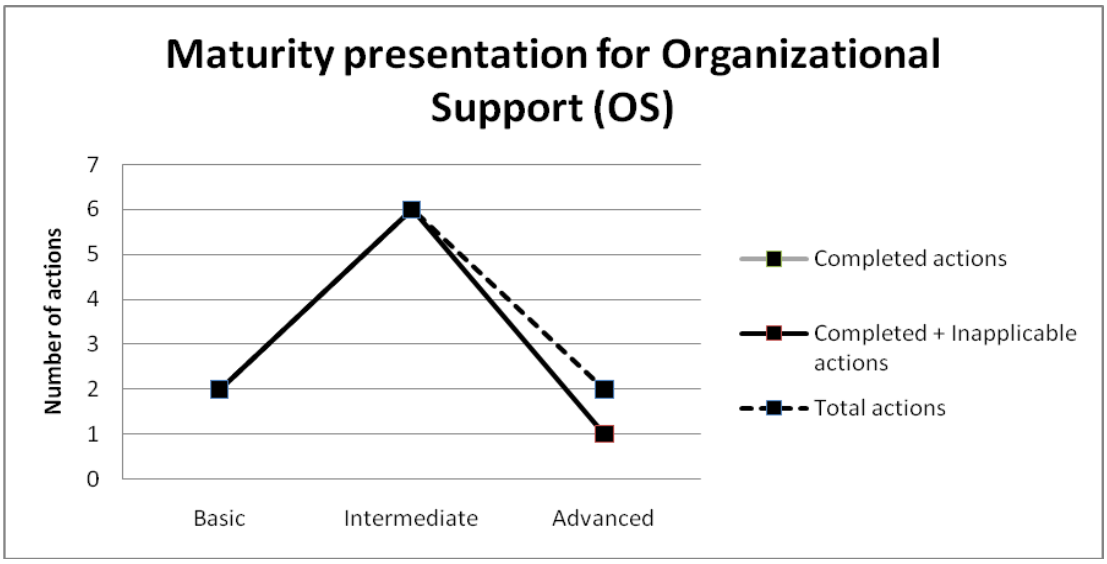
This missing of political impact consideration was claimed that it was considered particularly due to certain case, hence there was not a defined processed for this issue.

Model lag is very small in this case. Most of the actions were either completed or incomplete. This may refer to the fact that the interviewee is very experienced in the area and the organization itself holds a high maturity level.

2. MPA ORGANIZATION SUPPORT

The number of completed and inapplicable actions according to each Uni-REPM level in Organizational Support compared to the total actions:

Level	Completed actions	Inapplicable actions	Total actions
Basic	2	0	2
Intermediate	6	0	6
Advance	1	0	2



General

Looking at the graph, the project resides in level 2 – Intermediate level for this Requirements Elicitation MPA. It is well-noticing that no model lag exists in this MPA. It proves the high applicability of the actions in this MPA.

Suggestions for improvement

In order to achieve Advanced level, one action has to be completed:

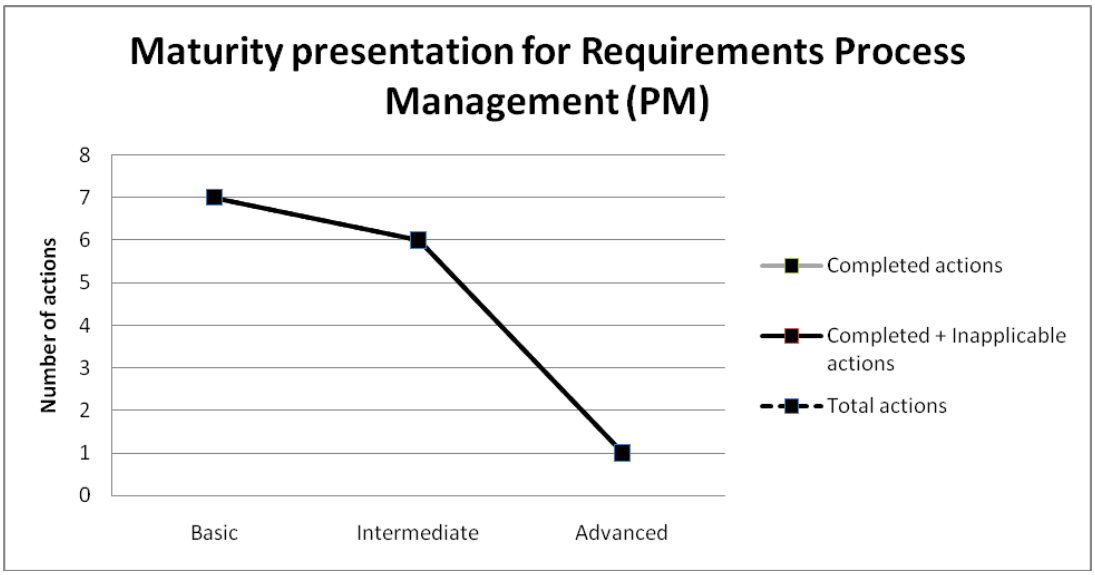
- *OS.RR.a5 Define Roles and Responsibilities for Product Management (Advanced level)*

In project where the product is released to many different market segments or countries, it is necessary to define the product strategies, roadmap and disseminate the information to the affected staff. These are the responsibility of the project management. The roles involved in product management can be contributors, controllers and distributors. The contributors possess future-oriented tacit knowledge regarding the market. The controller responsibility is to combine contributors' knowledge into product strategies and roadmaps. The distributor disseminates the product strategy and roadmap knowledge into the organization by identifying who depend heavily on it. These roles present not only internal perspective but also external one including sale and customers. Depending on each company, there can be many more roles and responsibilities.

3. MPA REQUIREMENTS PROCESS MANAGEMENT

The following table and figure demonstrate the maturity assessment of this MPA in the process using Uni-REPM.

Level	Completed actions	Inapplicable actions	Total actions
Basic	7	0	7
Intermediate	6	0	6
Advanced	1	0	1



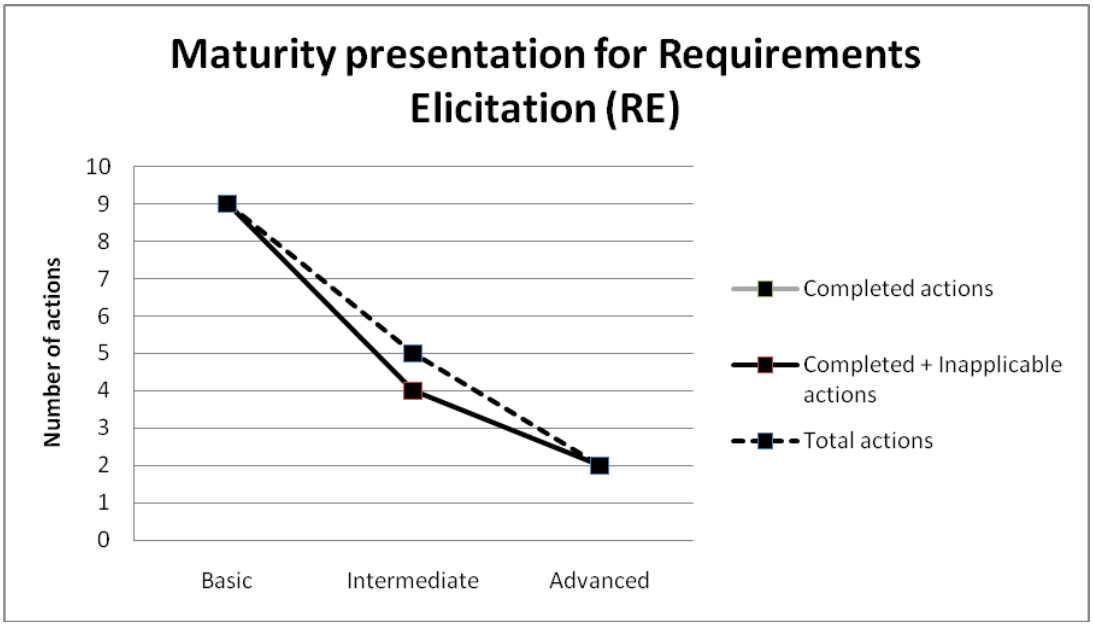
General
The diagram shows that, in this MPA, the process is very mature, it reaches the Advanced level. All the actions in this MPA were completed. Model lag does not exist which means none of the action was inapplicable in the case of the project.

Suggestions for improvement
As all the actions in all three levels are fulfilled, it is assured that the process is very well-structured. Change management and tracing policies are clearly defined at the beginning of the process. Moreover, stakeholders are considerately involved in the process. This is the strong point of the whole process. Hence, the team can focus its effort on other MPAs to maintain a well-rounded process.

4. MPA REQUIREMENTS ELICITATION

The number of completed and inapplicable actions according to each Uni-REPM level in Requirements Elicitation compared to the total actions:

Level	Completed actions	Inapplicable actions	Total actions
Basic	9	0	9
Intermediate	4	0	5
Advanced	2	0	2



General
Looking at the graph, the project resides in level 1 – Basic level for this Requirements Elicitation MPA. It is interesting to point out that although not all Intermediate actions are completed; all the actions in Advanced levels are fulfilled.
Once again no model lag exists in this MPA. It proves the high applicability of the actions in this MPA.

Suggestions for improvement
By completing one action below, the maturity level of this MPA will be Advanced:

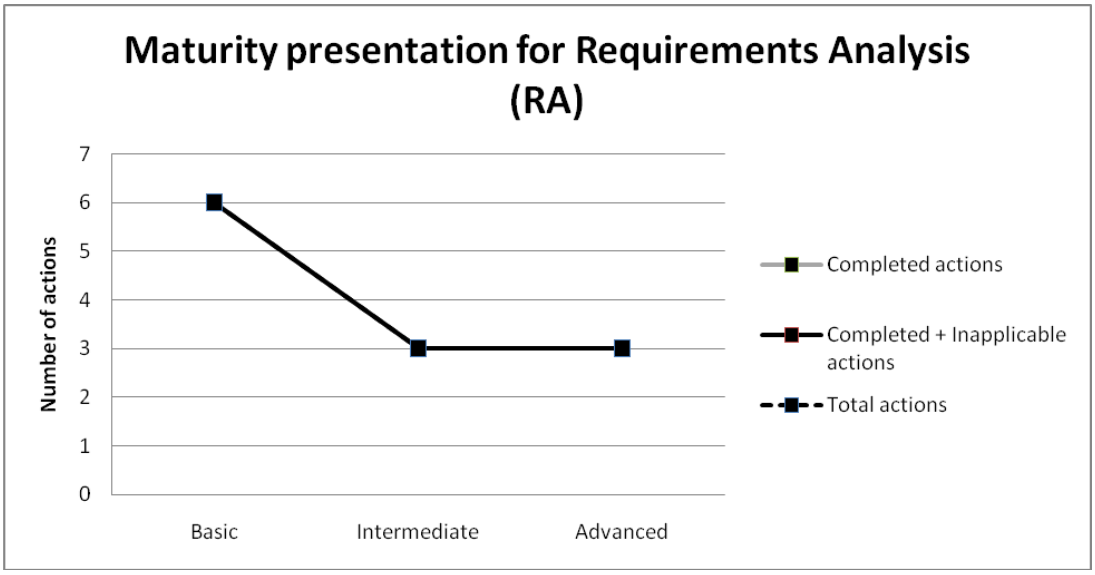
- *RE.DC.a6 Consider Sociopolitical Influences on Requirements Sources (Intermediate level)*

Although the practitioner is fully aware of the political issues in the organization, there is no defined process in dealing with it. The issue is tackled on a case-by-case basis as it is a soft issue. Hence, if the organization gets more mature, it can develop guidelines in how to solve the sociopolitical Influences on requirements sources.

5. MPA REQUIREMENTS ANALYSIS

The following table and figure demonstrate the maturity assessment of this MPA in the process using Uni-REPM.

Level	Completed actions	Inapplicable actions	Total actions
Basic	6	0	6
Intermediate	3	0	3
Advanced	3	0	3



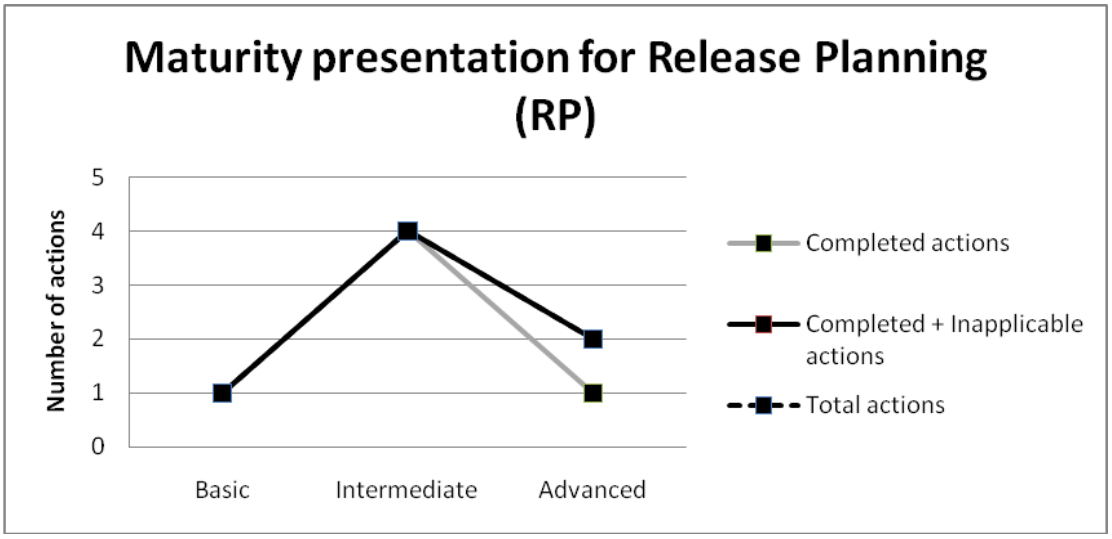
General
As we can see from the diagram, the three lines completely overlap in this MPA the process. This indicates that the process in this MPA is very mature, it reaches the Advanced level. All the actions in this MPA were completed. Model lag does not exist which means none of the action was inapplicable in the case of the project.

Suggestions for improvement
As all the actions in all three levels are fulfilled, it is assured that all quality dimensions of requirements were considered in the process. Moreover, modeling was used to support the requirements engineering process. This is the strong point of the whole process. Hence, the team can focus its effort on other MPAs to maintain a well-rounded process.

6. MPA RELEASE PLANNING

The following table and figure demonstrate the maturity assessment of this MPA in the process using Uni-REPM.

Level	Completed actions	Inapplicable actions	Total actions
Basic	1	0	1
Intermediate	4	0	4
Advanced	1	1	2



General
The diagram shows that in this MPA, the process resides at the highest level - Advanced level.

The model lag exists but very small in this case (only one action) which means most of the designed activities in the model are applicable in the project:

- *RP.GA.a3 Post Requirement Selection Evaluation (Advanced level)*

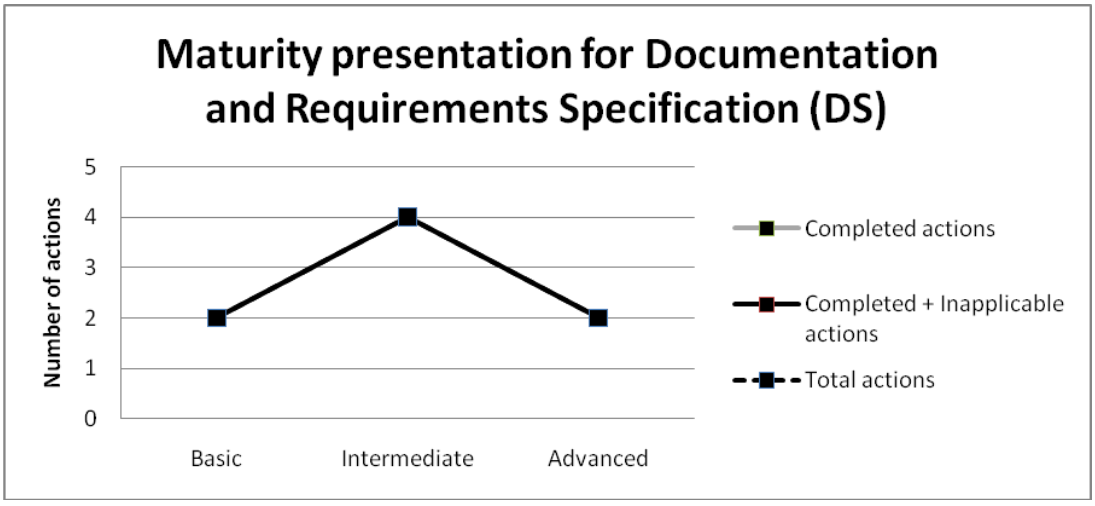
The action is deemed inapplicable because the requirement selection was determined by the delivery team based on the priority of the user and the resource of the project. As long as the user and delivery team agree on the set of the requirements, they will be implemented. Therefore, it is not necessary to perform the post-mortem of the selection to decide whether the selection is correct.

Suggestions for improvement
As all the actions in all three levels are fulfilled, it is assured that decisions were made under a very well-defined process. Moreover, the process involved different perspectives in decision making. It is also necessary to mention that the prioritization with more than one aspects were also implemented. This is the strong point of the whole process. Hence, the team can focus its effort on other MPAs to maintain a well-rounded process.

7. MPA DOCUMENTATION AND REQUIREMENTS SPECIFICATIONS

The number of completed and inapplicable actions according to each Uni-REPM level in Documentation and Requirements Specifications compared to the total actions:

Level	Completed actions	Inapplicable actions	Total actions
Basic	2	0	2
Intermediate	4	0	4
Advanced	2	0	2



General

From the graph, the project achieves Level 3 – Advanced in Documentation and Requirements Specification MPA. It completed all the required actions and reached the most mature level in this aspect.

Once again no model lag exists in this MPA. It proves the high applicability of the actions in this MPA.

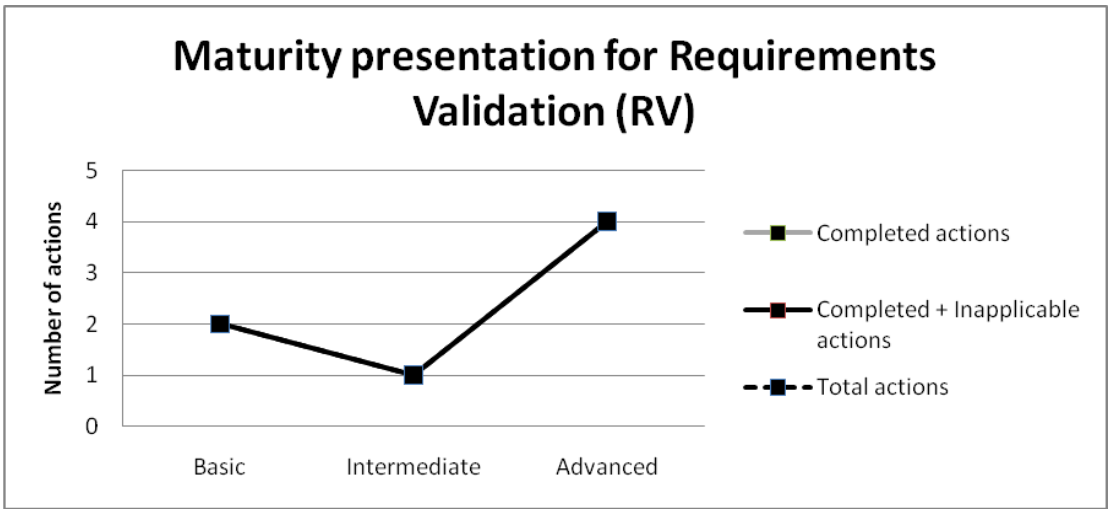
Suggestions for improvement

As all the actions in all three levels are fulfilled, it is assured that requirements and other knowledge gathered during requirements engineering process are organized into consistent, accessible and reviewable documents. This is the strong point of the whole process. Hence, the team can focus its effort on other MPAs to maintain a well-rounded process.

8. MPA REQUIREMENTS VALIDATION

The number of completed and inapplicable actions according to each Uni-REPM level in Requirements Validation compared to the total actions:

Level	Completed actions	Inapplicable actions	Total actions
Basic	2	0	2
Intermediate	1	0	1
Advanced	4	0	4



General
From the graph, the project achieves Level 3 – Advanced in Documentation and Requirements Specification MPA. It completed all the required actions and reached the most mature level in this aspect.
Once again no model lag exists in this MPA. It proves the high applicability of the actions in this MPA.

Suggestions for improvement
As all the actions in all three levels are fulfilled, it is assured that the documented requirements are correct, consistent, clear and complete. Moreover, they comply with defined quality standards and real needs of various stakeholders. This is the strong point of the whole process. Hence, the team can focus its effort on other MPAs to maintain a well-rounded process.

APPENDIX H – SELF-ADMINISTERED QUESTIONNAIRE

Part I. Introductory Questions

Question	Answer
What is your position/job scope?	
How long have you experienced in Requirements Engineering ?	
What industry does your company belong to? How many employees in your company/business unit?	

Part II. Project Questionnaire

Please select one typical project you have done and answer the questions below accordingly:

Question	Answer
What is the project about? What is the product?	
How many man-hours for the project? (how many people involved in the project? How long does it last?)	
Is it a mass market product? Or do you have specific customer(s) for this product?	
Additional info you would like to share about the project?	

Please choose one option "Complete", "Incomplete" or "Inapplicable" and put (X) in the corresponding box for each action below:

- **Complete (C):** The action was performed fully in this RE process.
- **Incomplete (IC):** The action is necessary but was performed PARTIALLY or not at all
- **Inapplicable (IA):** The action is NOT necessary or impossible to be performed in this process. Please provide reason as well.

If you have doubt about a specific question, please refer to the model in the file "Brief_Model.pdf" with the Action ID for further information and explanation.

(Refer to "Appendix I – Uni-REPM checklist" for the questions)

Part III. Your feedback

Question	Answer
Do you perform any additional action(s) that is not covered in the questions above?	
Are you familiar with the terminology used in the questionnaire? Which term(s) do you find hard to understand?	
How often do you need to refer to the model to understand the question?	
Is the explanation in the model adequate and easy to understand?	
Additional comment(s)	

APPENDIX I – UNI-REPM CHECKLIST

Action ID	Question	(C)	(IC)	(IA)	Comment / Reason if Inapplicable
OS	**Organizational Support**				
OS.GA	General Actions				
OS.GA.a1	Do you have a product-wide glossary of terms to ensure that the key concepts in the domain are properly understood by all parties?				
OS.GA.a2	Do you have training about requirements development and management processes as well as necessary skills to perform the job?				
OS.RR	Roles and Responsibilities				
OS.RR.a1	The owner of requirements process is responsible for defining and maintaining the RE process. Do you delegate this task to someone in your organization?				
OS.RR.a2	What roles are involved in the overall requirements engineering process?				
OS.RR.a3	When performing release planning, who should be involved, and what should be their roles?				
OS.RR.a4	Change is inevitable. What are the involving roles when requirements changes occur?				
OS.RR.a5	The product management organization is deeply involved in defining the requirements for the product. What roles exist, and what are their different responsibilities?				
OS.S	Strategies				
OS.S.a1	Do you have the product strategies defined? Which market segments and key customers does this product target?				
OS.S.a2	Do you have product roadmaps defined and documented centrally?				
OS.S.a3	What means of communication is used to disseminate or retrieve the strategy knowledge within your organization?e.g. formal meeting? Informal talking?				
PM	**Requirements Process Management**				
PM.GA	General Actions				
PM.GA.a1	Do you define and document the process of how you perform requirements development and management?				

PM.GA.a2	Do you have tool support for your requirements engineering activities, e.g. a requirements database tool (could be excel sheets saving in a central place), support for requirements prioritization, etc.?				
PM.GA.a3	How do you get the stakeholders and relevant team involved in your Requirements Engineering process?				
PM.CM	Configuration Management				
PM.CM.a1	Do you have a defined process for dealing with new (and old) versions of requirements?				
PM.CM.a2	Do you baseline your requirements at some points? Do you baseline them together with e.g. design artifacts and test cases?				
PM.CM.a3	Do you have a defined process for how to manage change requests? Do you have a defined process for keeping the requirements up to date with the current development status? Do you have a defined process for communicating changes to the rest of the organization?				
PM.CM.a4	Do you have a mechanism to keep track on the change request? E.g. when change request is triggered, analyzed, or approved.				
PM.RT	Requirements Traceability Policy				
PM.RT.a1	Do you have any mean to uniquely identify each requirement? E.g. ID number so that you can refer to them in other artifacts?				
PM.RT.a2	Do you document the source of a requirement so that you may go back and get further information? The source can e.g. be documents, process descriptions, competitor products, as well as people.				
PM.RT.a3	Do you define the policies for tracing requirements when necessary? i.e. requirements need to be traced backward and forward to detect source of problem and consequent changes to apply.				
PM.RT.a4	Do you document relations between requirements? Relations can be e.g. "must be developed together", "implement before", "cannot be implemented together", "influences negatively", "influences positively", etc.				
PM.RT.a5	Do you document the impact of a requirement on other artifacts such as pre-studies, product design, implementation artifacts, test cases, etc.? When you produce an artifact, do you attach information about affected requirements?				

PM.RC	Requirements Communication and Negotiation				
PM.RC.a1	Do you establish and maintain contact with the requirements' issuers to obtain an understanding on the requirements they proposed?				
PM.RC.a2	How do you ensure that the involving roles have the same understanding of the requirements?				
RE	**Requirements Elicitation**				
RE.GA	General Actions				
RE.GA.a1	Do you determine which quality aspects of the system to focus on (such as performance, usability, reliability etc) and explicitly elicit requirements about them?				
RE.GA.a2	Do you describe quality requirements in details such as max, min, average value?				
RE.GA.a3	Do you use the business objectives to guide how you conduct your elicitation efforts?				
RE.GA.a4	Depending on each situation, certain elicitation techniques are more appropriate to use than others. Do you consider different techniques to suit each case?				
RE.GA.a5	When appropriate, do you use additional artifacts such as prototyping or scenario to aid in the elicitation and analysis process?				
RE.GA.a6	Do you create different channels to capture all forms of requirements from various sources?				
RE.GA.a7	Do you have a systematic process to reuse requirements from other systems developed in the same application area?				
RE.SI	Stakeholder and Requirements Source Identification				
RE.SI.a1	Do you identify and involve different stakeholders in elicitation? For example, customers, end users, testers, developers, marketing etc.				
RE.SI.a2	Besides the above stakeholders, do you elicit requirements from other sources (e.g. partners, distributors, company standards, regulations etc)?				
RE.DC	Domain Consideration and Knowledge				
RE.DC.a1	Do you systematically elicit information about restrictions or possibilities that the domain may impose on your product?				
RE.DC.a2	Do you consider the technical infrastructure when developing the system?				
RE.DC.a3	When eliciting requirements, do you consider how your system will contribute to the business process in customer's organization?				

RE.DC.a4	Do you take into account the co-existing business processes which the system should support?				
RE.DC.a5	Do you elicit the information about what is part of your system and what is outside of the scope? Do you use this information to enable yourself to focus on what is within the system boundaries?				
RE.DC.a6	Are you aware of and have ways to deal with the political or organizational influence on the requirements sources when eliciting requirements?				
RA	**Requirements Analysis**				
RA.GA	General Actions				
RA.GA.a1	Do you systematically assess the risks of individual requirements or set of requirements?				
RA.GA.a2	Do you have a systematic way of prioritizing the requirements so that you know which to focus on? Do you priorities based on several viewpoints (e.g. cost, value, risk, penalty)? Do you priorities with the help of your stakeholders? Do you weigh the importance of different stakeholders?				
RA.GA.a3	Do you systematically estimate whether there are any dependencies or relations between requirements?				
RA.GA.a4	Do you have a process for selecting, at an early stage, which requirements to focus on and which to discard immediately, so that you spend your resources on the right requirements?				
RA.GA.a5	Do you estimate and document how much requirements may impact or increase or decrease the value of other requirements?				
RA.GA.a6	Do you classify and group requirements in to different categories based on their goals or levels of abstraction?				
RA.QA	Requirements quality analysis				
RA.QA.a1	Do you systematically analyze whether you have double requirements or whether there are requirements missing?				
RA.QA.a2	Do you systematically analyze whether your requirements are ambiguous?				
RA.QA.a3	Do you systematically analyze whether your requirements are incorrect?				
RA.QA.a4	Do you systematically analyze whether your requirements are testable?				
RA.PS	Problems and solutions analysis				

RA.PS.a1	Do you create prototype for the product to analyze whether the requirements are specified properly?				
RA.PS.a2	Do you create system and architectural models to analyze whether the requirements are specified adequately?				
RP	**Release Planning**				
RP.GA	General Actions				
RP.GA.a1	Is your release plan in line with your product roadmap? Are there areas in your product roadmap where you have no or insufficient requirements? Do you also propose to change this part of the roadmap?				
RP.GA.a2	Do you consult your stakeholders in prioritization and decision making (RP)? Do you weigh the importance of different stakeholders?				
RP.GA.a3	Do you evaluate whether you actually selected the right requirements for certain release?				
RP.GA.a4	How far ahead does your planning stretch? Are you able to foresee and start preparing for subsequent releases already now?				
RP.S	Requirements Selection				
RP.S.a1	Do you clearly define a release with all the necessary information before passing it to the next development stage?				
RP.S.a2	Do you estimate the cost and value of your requirements?				
RP.S.a3	Do you priorities based on several viewpoints (e.g. cost, value, risk, penalty)?				
DS	**Documentation and Requirements Specification**				
DS.GA	General Actions				
DS.GA.a1	Do you follow a standardized structure of how the System Requirements Specification should be written, or generated?				
DS.GA.a2	Do you define attributes for each requirements such as ID,title, descriptions, author etc?				
DS.GA.a3	Do you define and store states that requirements should follow during their lifetime? For example, new, assigned, rejected, implemented, tested, delivered.				
DS.GA.a4	Do you document the reason why the requirement is specified and what function the requirement has?				
DS.GA.a5	Do you record which requirements were rejected and why so as to avoid re-analysis if the same requirements reappear later?				

DS.DD	Documentation Deliverables				
DS.DD.a1	Do you define what user manuals and other user documentation that shall be delivered together with your product?				
DS.DD.a2	Do you define system documentation that shall be delivered together with your product?				
DS.DD.a3	Do you define management documentation that shall be delivered together with your product?				
RV	**Requirements Validation**				
RV.GA	General Actions				
RV.GA.a1	Do you validate requirements with relevant stakeholders to ensure that they reflect the correct intent?				
RV.GA.a2	Do you have and systematically use a checklist for ensuring the quality of your requirements?				
RV.GA.a3	Do you conduct requirements reviews to ensure their quality?				
RV.GA.a4	Do you organize inspections to review the requirements with other stakeholders?				
RV.GA.a5	Do you create artifacts (e.g. user manuals or test cases) to assist you in ensuring the quality of your requirements?				
RV.GA.a6	Do you convert system models into natural language in order to detect requirements errors?				
RV.GA.a7	Do you get the customer to define the acceptance criteria and acceptance test?				

Appendix J - Requirements Engineering Process Maturity Model Uni-REPM

Contents

PART I. Motivation

Requirements engineering (RE) has been acknowledged to be one of the most important processes since it is the critical determinant of the software quality and software development process effectiveness [1]. Currently, requirements engineering has been roughly divided into two main approaches: ***bespoke*** and ***market-driven***. In software development, bespoke approach, also known as traditional development approach, aims to produce software products for specific customers [2]. However, market-driven development tends to provide software products for a massive market [2]. These two main characteristics bring many differences between them apart from the similarities in practices and technologies [3, 4]. For instance in MDRE, requirements engineering usually involves organizational and product strategies. The validation activities with the customers can hardly be performed directly due to the fact that they do not have specific customers, and the elicitation is mostly through market analysis or survey which is never made use of in bespoke [2].

Despite those particular differences, both approaches are still facing many challenges in handling requirements in industry. As in the research of Juristo [5], Beecham [6] and Niazi [7], there still exist many problems. Personnel and tool support are listed as the most pressing problems. Besides, communication in the process, documentation, requirements estimation are also reported as high rank problems in the requirements engineering process in organizations. Specifically, in MDRE, the vast number of stakeholders makes it hard to elicit and manage the requirements since they usually provide a large and continuously growing number of requirements with many different levels of abstractions [3].

However, those problems tend to be reduced in higher maturity RE process [8]. Therefore, the organizations should improve their RE process in order to overcome those challenges. In order to do that, as the first step of the improvement process, the current state of the process needs to be evaluated [9,10]. There have been a number of models developed for assessment purpose, namely **CMM [11]**, **CMMI [12], ISO 9000 [13]**. However, most of them did not pay an adequate attention on RE process. **Good Practice Guidelines (GPG) [14]** and **REPM [15]** models were introduced later to particularly focus on RE process. Nevertheless, these models are only limited to bespoke development. This limits the model applicability in many cases such as in market-driven or hybrid (in which both approaches could be used in development) organizations where RE practices usually start early at product-level. This fact motivated us to upgrade and broaden the original light-weight **REPM** model so that it could be applied intensively in all industrial organizations.

Based on this idea, we have performed research and construct the model named Universal Requirements Engineering Process Maturity model (Uni-REPM). The next sections of the document will provide an overview, structure and usage of the model (Part II) and the description of the model itself (Part III).

PART II. UNI-REPM overview

The purpose of this section is to give a brief view of the Uni-REPM model covering its structure, components and usage.

1. Introduction

Uni-REPM is a light-weight model presenting the maturity of RE process through sets of necessary activities. The assessment is basically an action to map those ideal activities to real work. The activities in the model are divided into 7 areas: Organizational support, Requirements Management Process, Elicitation, Requirements Analysis, Release Planning, Documentation and Requirements Specification, and Requirements Validation.

Uni-REPM is a means to identify the strengths and weaknesses of the RE process in organizations. Furthermore, it provides organizations recommended practices to improve their requirements engineering process from basic to advanced level.

The model was constructed based on extensive literature reviews of REPM model [15], CMMI [12], ISO9000-TickIt [13] together with a systematic review on market-driven requirements engineering researches.

2. Model Structure

The construction of Uni-REPM structure was based on the original model REPM. The model hierarchy has three levels, namely ***Main process area*** (MPA), ***Sub-process area*** (SPA) and ***Action***. On the top level of the model, there are seven Main process areas corresponding to requirements engineering main activities. Each MPA is further broken down into several SPAs, which facilitates better understanding. On the bottom level, an Action denotes a certain activity that should be done or a certain item that should be present.

So as to improve the model structure and its coherence, closely-related actions are grouped together and put under one SPA if possible. Besides, every MPA has one SPA called “General actions” where other actions reside. An action must be attached to a SPA and there is no MPA or SPA located under an action. The model components are summarized and illustrated in Figure 1.

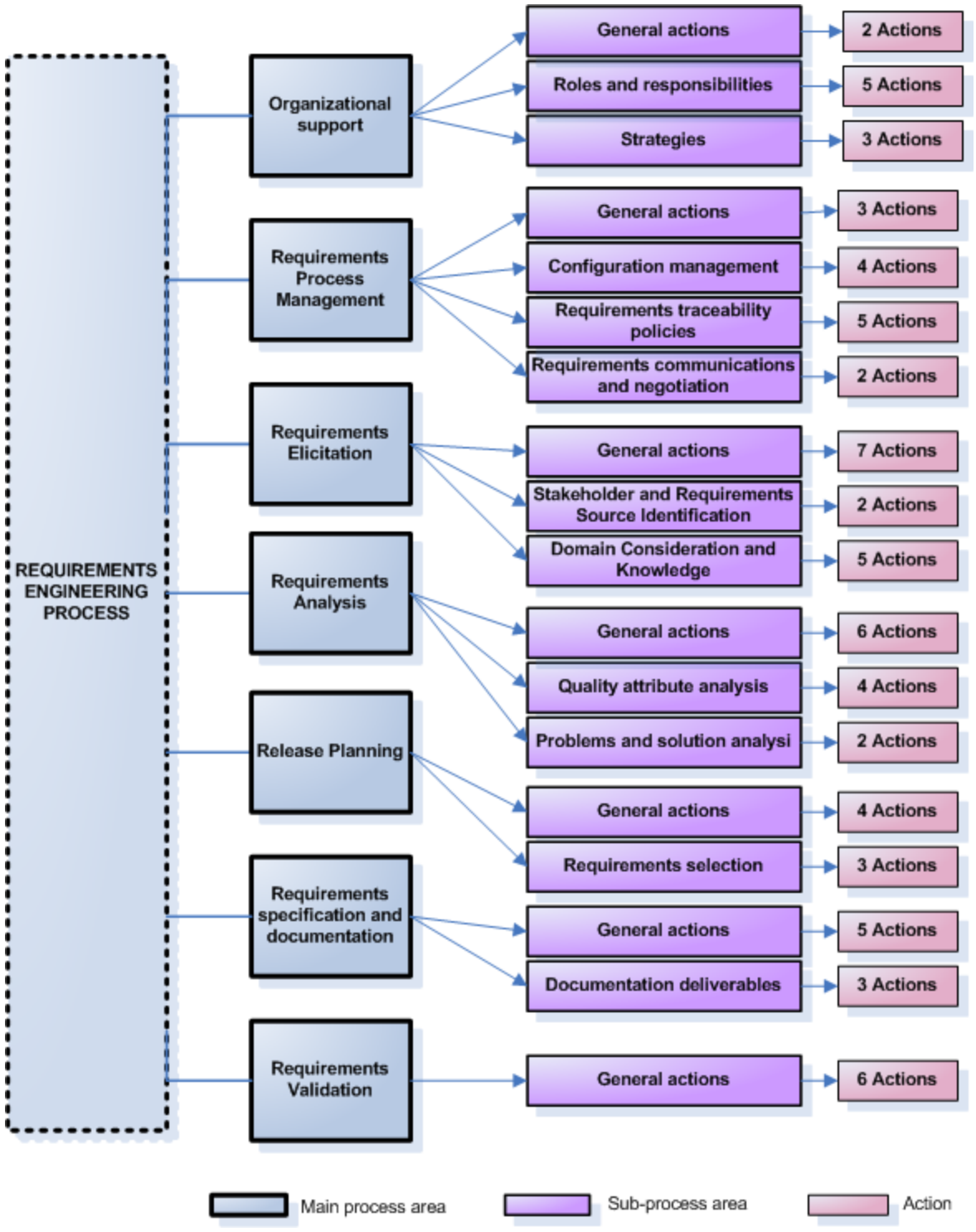


Figure 1. Uni-REPM Model structure inspired by REPM model [15]

2.1. Main Process Area (MPA)

On the top level of the model, a main process area represent a cluster of related practices in one main requirements engineering activity such as Elicitation.

There are seven MPAs in the model, represented here according to the active order in the requirements engineering process:

- **Organizational Support**: This main process area evaluates the amount of support given to requirements engineering practices from the surrounding organizations. This MPA supports both the Requirements Development and Requirements Management processes.
- **Requirements Process Management**: The requirements process management covers all the activities to manage, control requirements change as well as to ensure the organization of the process and coherence among team members. This MPA represents the Requirements Management process.
- **Requirements Elicitation:** Elicitation is the process of discovering, understanding, anticipating and forecasting the needs and wants of the potential stakeholders in order to convey this information to the system developers. This MPA is part of the Requirements Development process.
- **Requirements Analysis**: Requirements after being gathered from different sources need to be analyzed to detect errors, to create detailed view of requirements as well as to estimate necessary information for later activities (eg. risk, priorities…). This MPA is part of the Requirements Development process.
- **Release Planning:** Release planning covers crucial steps aiming to determine the optimal set of requirements for a certain release to be implemented at a defined/estimated time and cost to achieve some goals. This MPA is more applicable to market-driven development.
- **Documentation and Requirements Specification:** Documentation and Requirement specification deals with how a company organizes requirements and other knowledge gathered during requirements engineering process into consistent, accessible and reviewable documents. This MPA is part of the Requirements Development process.
- **Requirements validation:** This process involves checking the requirements against defined quality standards and the real needs of various stakeholders. It ensures that the documented requirements are complete, correct, consistent, and unambiguous. This MPA is part of the Requirements Development process.

Each MPA has a unique identifier which enables traceability throughout the model. For example, “Organizational Support” MPA is referred to as “OS”.

2.2. Sub-Process Area (SPA)

Sub-process area contains closely related actions, which help to achieve a bigger goal. The unique identifier assigned to each SPA is composed of the MPA identifier to which the SPA attaches and its abbreviation. For example, “OS.RR” represents a sub-process area “Roles and Responsibilities” which resides under MPA “Organizational Support”.

2.3. Action

The smallest unit in the model is called “action” showing a specific good practice. By performing the action, the organization can improve their process and gain certain benefits. For example, an action

"Create a product-wide glossary of terms" once implemented will enable readers from different backgrounds to understand the technical jargons used in a requirements document.

Actions also follow the same format to form their unique identifiers. They are identified by the MPA/ SPA under which they reside, followed by an "a" which stands for "action" and their position in the group. For example, "OS.a1" points to the first action which attaches directly to MPA "Organizational Support". Another example is "OS.RR.a1", which means the first action under MPA "Organizational Support" and SPA "Roles and Responsibilities".

Each action is assigned a certain level depending on its difficulty to implement and essentiality for the requirements engineering process. The level structure will be discussed in detail in section 3.

Example(s) and Supporting Action(s)

Within the description of each Action, there can be **Example(s)** and **Supporting Action(s)**. The idea of **Example(s)** is to give practitioners suggestions on proven techniques or supporting tools when performing the action. It is worth noticing that the Example item, as the name suggests, is not an exhaustive list. Therefore, companies are not restricted to apply only those in order to fulfill an action. In addition, the **Supporting Action(s)** provided links to other Actions which will benefit the practitioners when implementing them together. Figure 2 shows a snapshot of the model to illustrate its structure and components.

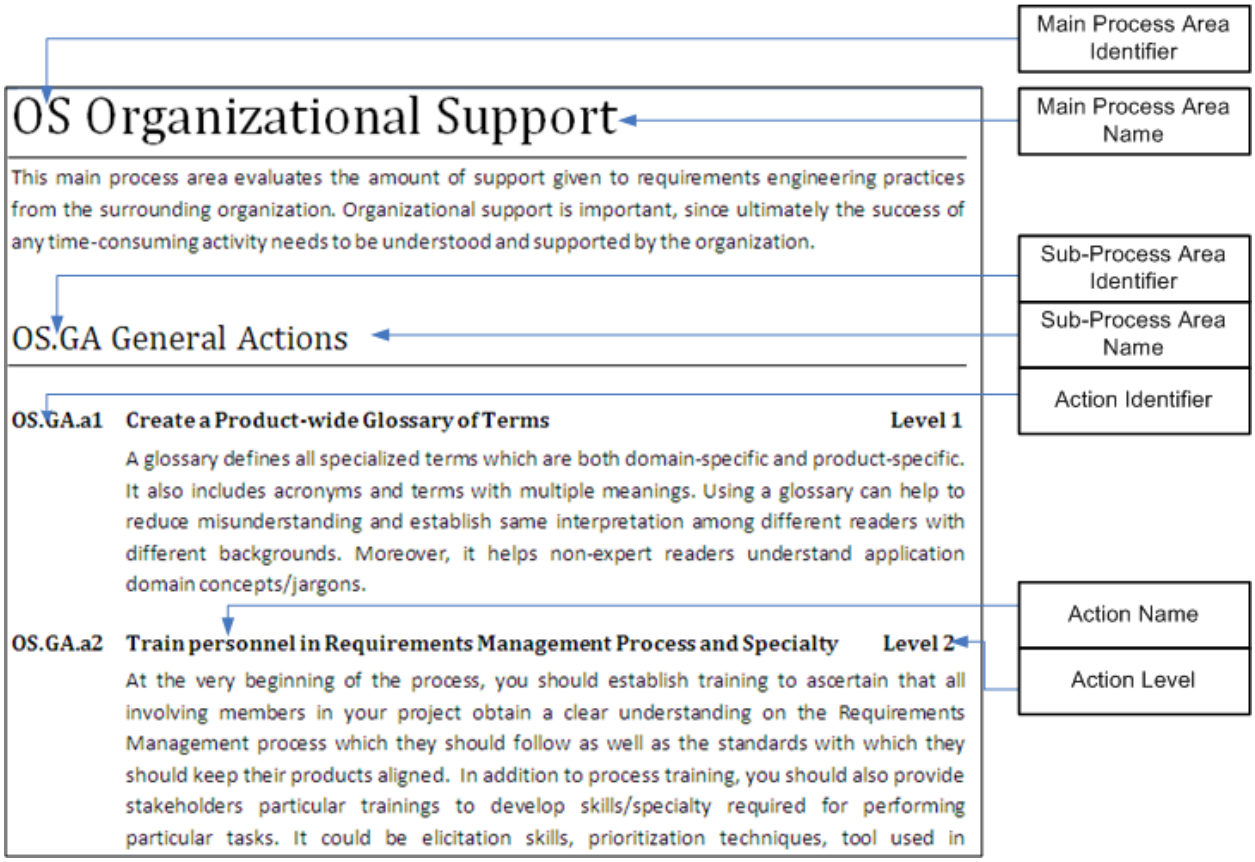


Figure 2. A snapshot of Uni-REPM model

3. Process maturity

Uni-REPM makes use of an ordinal scale to assess the maturity of the process. The levels to assess process maturity is inspired from the REPM [15] and REGPG [14] models; and the book "*Software requirements*" [16]. Concerning the fact that Requirements Engineering Processes applied in industry are usually small-sized and ad-hoc [5], three levels of maturity were defined, namely ***Basic, Intermediate*** and ***Advanced.*** The reason for changing from 5 levels as in REPM model [15] to 3 levels is to make the requirements engineering process significantly better after completing each level. Hence, the benefits gained from moving from one level to another level are more visible as well. The meaning of these levels is they will present how mature the evaluated process is. It is, however, not applicable to the whole organization maturity since the model scope only resides on Requirements Engineering Process. Nevertheless, it is possible to compare two processes in term of maturity using the evaluation results from the model.

The resulting level of a process is constructed from levels of actions performed within such process. In Uni-REPM, each action is placed under a certain level concerning its essentiality and required skills/cost to carry out. We also considered the dependencies among actions when assigning levels to them, e.g. if action A requires another pre-requisite action to be performed, it must be placed at least at the same or higher level than the pre-requisite action.

Level 1 – Basic

The aim of this level is to achieve a rudimentary repeatable requirements engineering process. The process in this level is defined and followed. Quality of requirements is managed because of relevant stakeholder involvement in elicitation, in-depth requirements analysis and pre-defined document standards.

However, the process does not maintain any kind of communications among stakeholders and within the organization in term of strategies.

Level 2 – Intermediate

In this level, the process is more rigorous because it involves various perspectives and is led by product strategies/goals. Roles and responsibilities for particular tasks are clearly defined and documented. Change requests are handled in the consistent manner throughout the project. Well-informed decisions about requirement selection can be made by analyzing and prioritizing the requirements systematically.

This process still stays in "***present-state***"; meaning that there is no activity performed to collect and analyze data/feedback for future improvement of the process.

Level 3 – Advanced

This level denotes the most mature process. The improvements in the process are shown in the advanced way of capturing requirements, ensuring their high quality, maintaining communications and common understanding among different stakeholders and pro-actively assessing the decision making process.

The process takes into account the "***future-state***" since it not only covers pre-defined and structured procedures but also adequately pay attention on future works (e.g. reusable materials, port-term evaluation, etc.).

4. Model usage

4.1. Who will directly use the model?

Uni-REPM aims to assess the RE process maturity; hence it can be used by people who are involved in the process, deeply understand it and be in charge of process improvement in general. Example users can be:

- Software Engineer
- Quality assurance engineer
- Project manager
- Product manager

4.2. How to use the model?

To assess the maturity of a RE process, the users basically perform a mapping from the actions present in the model to the activities in a real process using the checklist. The checklist is actually a direct transformation of the model into question form. A snapshot of the checklist is shown in Figure 3.

The checklist follows the same structure as the model with questions grouped according to the MPA and SPA. For each action in the model, there is a corresponding question or group of questions to verify if the action is done or not. The Action ID which links the question(s) to the associated action in the model helps the users in case they need to locate the item for further information or clarification.

When answering the questions, the users may encounter one of the following situations:

- The action was deemed vital but was performed partially or not at all in this RE process. It should be marked as **"Incomplete"** (IC)
- The action was completed in this RE process. It should be marked as "**Complete"** (C)
- The action was not necessary or possible to be performed in this process. It should be marked as **"Inapplicable"** (IA)

More about "Inapplicable"

In reality, as organizations and processes vary in their characteristics and environments, they may not benefit from implementing all the actions in the model. Some of the actions are deemed unnecessary to be performed in particular situations of organizations. For example, a company has an internal glossary of terms but it is not released to the users. They find it hard and useless to give the user the whole document and ask them to read it as they might not do so. Instead, the terms and examples are explained directly to the users in the meeting and it was more effective this way. In this case, the action "OS.GA.a1 Create a Product-wide Glossary of Terms (Basic Level)" is not useful for them. If we consider it as "Incomplete", the process may not reach the Basic level because not all actions in this level are fulfilled. This is even more unfair if all other actions in higher maturity levels are completed. Therefore, companies should not be "punished" if they do not perform a certain nonessential action (in their point of view). In order to take into account this factor, the option "Inapplicable" is devised. In this way, the

model is more fitting to the real process and the evaluation result is less distorted. Besides, the differences between two types of development settings (bespoke and market-driven) do exist [8]. Therefore, in some cases, the organization may find some actions only applicable in one of the settings.

Whether an action is "Inapplicable" or not is solely based on the judgment of the project evaluator. Reasons for deeming an action "Inapplicable" should be considered carefully to avoid accidentally skipping an important action. Lack of time, resource or unawareness cannot be accounted for an "Inapplicable" action.

Action ID	Question	(C)	(IC)	(IA)	Comment / Reason if Inapplicable
OS	**Organizational Support**				
OS.GA	General Actions				
OS.GA.a1	Do you have a product-wide glossary of terms to ensure that the key concepts in the domain are properly understood by all parties?				
OS.GA.a2	Do you have training about requirements development and management processes as well as necessary skills to perform the job?				

Figure 3. Uni-REPM Checklist snapshot

4.3. How to read the result?

After mapping all the actions present in the model, the user can collect the results for each MPA and consider the following rules.

- For each MPA, all actions at a certain level must be **Completed** (or **Inapplicable**) in order for the MPA to achieve such level.
- For the whole process, all actions at a certain level must be **Completed** (or **Inapplicable**) in order for the process to achieve such level.

An example

The result of MPA "Organizational Support" after evaluating may look like in Table 2.

Table 1. Assessment result in MPA "Organizational Support"

Level	Actions in real process		Total actions in OS in Uni-REPM
	Completed	Inapplicable	
Basic	0	0	2
Intermediate	3	1	6
Advanced	1	1	2

To have a better view, the result can be presented in graph as follows.

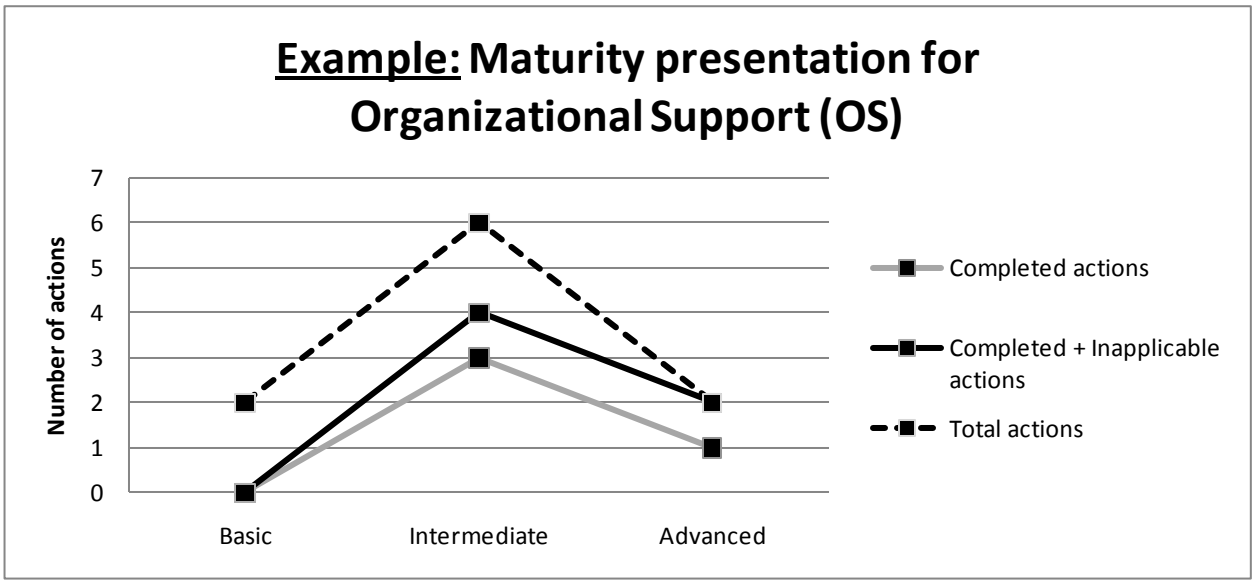


Figure 4. Graphical presentation of assessment results

The grey line presents actions which were completed in the real process. In this case, no action was done at the lowest level, 3 actions were completed in Intermediate level and 1 action in the highest level. The black line presents actions completed together with actions were not performed due to unnecessary or inapplicable reasons in real process of the assessing organization. The distance between the grey line and black line is called the model lag, which represents the number of inapplicable actions. Hence, the model lag shows the applicability of the model in the real setting. In this case, the model lag is fairly small with only two inapplicable actions. This means a high applicability of the model.

The dash line in the graph presents the total actions that should be completed at 3 levels in "Organizational Support" MPA. For example, at Basic level, there are 2 actions that should be finished. The difference between the black line and the dash line is important because it denotes the improvement area of the process. It shows how many additional actions that should be done in order to attain a certain level of maturity.

Overall, the graph denotes that, in this MPA, the process has not completed all the actions at Basic level. Hence, according to the above rule, the MPA resides on Level 0. In order to reach the Basic level, two more actions have to be done. If the company aims for Intermediate level, it has to perform two Basic actions and another 2 Intermediate ones. Similar work can be done with other MPAs to achieve the result for the whole process.

5. References

[1] Somerville I., ”Software Engineering”, Addison-Wesley, 1995.

[2] Wohlin C., Aurum A., “Engineering and Managing Software Requirements”, Springer, 2005.

[3] Karlsson, L., Dahlstedt, A.G., Regnell, B., Natt och Dag, J., Persson, Requirements engineering challenges in market-driven software development - An interview study with practitioners, In the Journal of Information and Software Technology, 49, 6, pp. 588-604, 2007.

[4] Regnell B., Beremark P., and Eklundh O., “A Market-driven Requirements Engineering Process - Results from an Industrial Process Improvement Programme”, Springer, pp. 121-129, 1998.

[5] Juristo N., Moreno A.M, Silva A., "Is the European Industry Moving Toward Solving Requirements Engineering Problems?", IEEE Software, vol.19, no.6, pp. 70-77, Nov/Dec 2002.

[6] Beecham, S., Hall, T., & Rainer, A., Software process problems in twelve software companies: An empirical analysis, Empirical Software Engineering, 8, 7–42, 2003

[7] Niazi, M., An empirical study for the improvement of requirements engineering process, The 17th International Conference on Software Engineering and Knowledge Engineering , pp. 396–399, 2005.

[8] Hall T., Beecham S., Rainer A., “Requirements Problems in Twelve Companies: An Empirical Analysis”, IEE Proceedings for Software, October, vol.149, no.5, pp.153-160, 2002.

[9] Gorschek T., “Requirements Engineering Supporting Technical Product Management”, Karlskrona : Blekinge Institute of Technology, 2006.

[10] Villalón C., Agustín C., Gilabert S., Seco D., Sánchez G., and Cota P., “Experiences in the Application of Software Process Improvement in SMES,” Software Quality Journal, vol. 10, pp. 261 – 273, 2002.

[11] Paulk M.C., Curtis B., Chrissis M.B. and Weber C., Capability Maturity Model TM for Software Version 1.1, 1993.

[12] CMMI for Development, Version 1.2, CMMI-DEV v1.2, CMU/SEI-2006-TR-008, Technical Report, Software Engineering Institute, August 2006, URL: http://www.sei.cmu.edu/pub/documents/06.reports/pdf/06tr008.pdf.

[13] The TickIT Guide – Using ISO 9001:2000 for Software Quality Management System, Construction, Certification and Continual Improvement, Issue 5.0, 2001.

[14] Ian Somerville and Pete Sawyer, Requirements Engineering – A Good Practice Guide, John Wiley & Sons, Chichester UK, 2000.

[15] Gorschek T., Tejle K., “A Method for Assessing Requirements Engineering Process Maturity in Software Projects”, Blekinge Institute of Technology, Master Thesis Computer Science no. MSC-2002:2, 2002.

[16] Wiegers K., Software Requirements, Microsoft Press. Redmond, Washington, 2003.

Part III. Model Description

This part provides the description of the Uni-REPM in two views: Process Area and Maturity Level.

1. Process Area View

In this section, the model will be presented by process area. The process area view helps organizations to focus on practices within a specific requirements engineering area e.g. Elicitation.

OS Organizational Support

This main process area evaluates the amount of support given to requirements engineering practices from the surrounding organization. Organizational support is important, since ultimately the success of any time-consuming activity needs to be understood and supported by the organization.

OS.GA General Actions

OS.GA.a1 Create a Product-wide Glossary of Terms **Level 1**

A glossary defines all specialized terms which are both domain-specific and product-specific. It also includes acronyms and terms with multiple meanings. Using a glossary can help to reduce misunderstanding and establish same interpretation among different readers with different backgrounds. Moreover, it helps non-expert readers understand application domain concepts/jargons.

OS.GA.a2 Train personnel in Requirements Management Process and Specialty **Level 2**

At the very beginning of the process, you should establish training to ascertain that all involving members in your project obtain a clear understanding on the Requirements Management process which they should follow as well as the standards with which they should keep their products aligned. In addition to process training, you should also provide stakeholders particular trainings to develop skills/specialty required for performing particular tasks. It could be elicitation skills, prioritization techniques, tool used in organizations and so on. This is because not all the members of your team can be aware of all the required techniques, and in many cases the techniques and tools are customized to adapt to your organizational situation. Along with the training, it is also necessary that the importance of the tasks is specified. This activity may include:

- Define a training program(s)
- Prepare documents
- Prepare personnel with appropriate knowledge
- Specify mechanism for measuring the effectiveness of the training program

Supporting action(s)

- PM.GA.a1 Define and Maintain Requirements Development and Management Process

OS.RR Roles and Responsibilities

In order to be able to produce repeatable and predictable results, it is important to define the roles, and their responsibilities, that deal with different aspects of requirements engineering. This information should be documented in a central place and everyone should know where and how to access it.

OS.RR.a1 Assign Owner(s) of Requirements Development and Management Processes — Level 1

The owner of the requirements development and management processes has the responsibility of managing the process, assuring that all the requirements engineering activities are executed properly and supporting tools, training are available when needed. The benefit of having the process owner is that the process will be kept alive and updated to changes.

Supporting action(s)

- PM.GA.a1 Define and Maintain Requirements Development and Management Processes

OS.RR.a2 Define Roles and Responsibilities for Requirements Development and Management Processes — Level 2

Roles and responsibilities for requirements development and management processes should be defined explicitly in details. In the case of mass market product, this step is particular important as the process does not follow a phase-oriented development model but an asynchronous fostering of requirements through a life-cycle. Some responsibilities that should be specified are creating, analyzing, specifying, validating and managing requirements.

Supporting action(s)

- PM.GA.a1 Define and Maintain Requirements Development and Management Process

OS.RR.a3 Define Roles and Responsibilities for Release Planning — Level 2

Release planning is the activity in which an optimal collection of requirements is selected for implementation in the next version of a software system. The responsibilities in release planning include deciding which prioritization aspects to consider, how to prioritize, selecting requirements into release, just to name a few. The roles involved in release planning can be product managers, marketing managers, technical managers, experts, customers etc.

OS.RR.a4 Define Roles and Responsibilities for Change Control — Level 2

As change happens all the time throughout the product lifecycle it is necessary to manage changes effectively by defining who is responsible for what in change control process. Some of the possible roles are change control board, change submitter, evaluator, modifier, and

verifier. The change control board is in charge of making decisions whether to approve proposed changes. The change control board should comprise people from different perspectives e.g. project management, product management, marketing, and development in order to have well-rounded and accurate decisions. The evaluator is responsible for analyzing the impact of the requirements change. The modifier executes the approved change on affected artifacts whereas the verifier checks if the change was implemented correctly.

Supporting action(s)

- PM.CM.a3 Define a Process for Change Control

OS.RR.a5 Define Roles and Responsibilities for Product Management Level 3

Product management is related to managing requirements, defining releases, and defining products in a context where many internal and external stakeholders are involved. The roles involved in product management can be contributors, controllers and distributors. The contributors possess future-oriented tacit knowledge regarding the market. The controller responsibility is to combine contributors' knowledge into product strategies and roadmaps. The distributor disseminates the product strategy and roadmap knowledge into the organization by identifying who depend heavily on it. These roles present not only internal perspective but also external one including sale and customers. Depending on each company, there can be many more roles and responsibilities.

OS.S Strategies

Certain strategic decisions and/or practices influence the requirements engineering process, especially in a market-driven product.

OS.S.a1 Define Product Strategies — Level 2

Product strategies can be defined by identifying where a company wants to go (direction of movement), how it will get there (means), what need to be done (tactics) and why it will be successful (rationale). The direction of movement can be determined in terms of profit, growth and market share. The means to reach the goals is by defining the customer targets, competitive targets and differentiated advantage. The tactics cover product, pricing, promotion, distribution, and service. Documenting the rationale is important because it enables replicating the success of the product.

The benefits of the product strategy are that it not only provides the long-term view of the product in the company but also drives the elicitation and analysis processes.

The strategies should be documented in a central place and updated regularly.

Supporting action(s)

- OS.RR.a5 Define Roles and Responsibilities for Product Management

OS.S.a2 Define Product Roadmaps — Level 2

The basic purpose of roadmapping is to explore and communicate the dynamic linkages between markets, products, and technologies over a period of time. It also helps requirements engineers to make business-oriented decisions in release planning, elicitation and analysis. Out of many types of roadmaps, the product-technology roadmap defines what a product tends to achieve over the time in terms of its evolvement and the technology trends.

The roadmaps should be documented in a central place and updated regularly.

Supporting action(s)

- OS.RR.a5 Define Roles and Responsibilities for Product Management Organization

OS.S.a3 Communicate Strategies in Organization — Level 3

Strategies are nothing without implementation. Strategies have to be disseminated to those who need to act on them. The first line consumers of strategic knowledge are people in development or productization teams who take the knowledge as input to their activities. Because of the immediate impact of the strategies on their work, a distributor is needed to communicate the strategies directly to the first line consumers. The second line consumers are those whose inputs are from the first line works, such as customers, partners, sales and

technical support. For these people, a documented form of strategies is enough.

Supporting action(s)

- OS.RR.a5 Define Roles and Responsibilities for Product Management
- OS.S.a1 Define Product Strategies
- OS.S.a2 Define Product Roadmaps

PM Requirements Process Management

The requirements process management covers all the activities to manage, control requirements change as well as to ensure the organization of the process and coherence among team members.

PM. GA General Actions

PM.GA.a1 Define and Maintain Requirements Development and Management Processes **Level 1**

It has been clear the benefit of having pre-defined processes to develop and manage requirements. This is to ensure a well-organized way to control the whole requirements process, and to guide the stakeholders of what to do next and How should it be done in a structured way. At project level, it is quite common to follow the phase-oriented process model in which requirements should be managed in phases such as elicitation, analysis (and negotiation), and documentation before being passed to another process within the project development. However, at pre-project level (product level), there is usually continuous stream of huge amount of requirements. Hence, concurrent approach models are preferable.

Example

You can follow one of the process model studied in research REQUEST [1], REPEAT [2], MDRE [3] or can tailor one based on these model to your organization.

Supporting action(s)

- OS.RR.a1 Assign Owner (s)of Requirements Development and Management Processes

PM.GA.a2 Introduce Tool Support for Requirements Development and Management **Level 1**

The tool support for the whole requirements process should be considered and chosen early. You can consider the following types of tools:

- Storage tools: with a huge number of requirements, especially in market-driven requirements engineering, it is crucial to have a database to store them. It is more advantageous if you can have a centralized repository for requirements so that all the changes will be applied in real time and different stakeholders can have the same view at a set of requirements.
- Version tools: provide automatic assignment of versions.
- Prioritization support tools: support prioritization.
- Elicitation tools: support elicitation

Example

You can find below the table of the support tools which are used quite popularly nowadays [4].

Tool name – Producer name	Purpose
Vital link - Compliance Automation	Database-centric system
RTM system	Provide repository for requirements management
RM Trak - RM Trak	Requirements management at entry level
Caliber RM – Borland	Lifecycle oriented, for large systems, provide traceability
CARE	Database-like view, requirement-centric system
Vital link - Compliance Automation	Database-centric system
DOORS – Telelogic	Integrated management, large projects, API available, High, XML support; PLM, UML tools, MS Project
IRRV (Integral Requisite Analyzer) - TCP Sistemas & Ingeniería	Requirements classification, OO analysis and entity relationship method for database design, traceability, test support, XML support; MS Office
ReqSimile	Finding and linking similar requirements
cost-value tool support	Prioritization using AHP
PARSEQ tool	Tool support for Post-release validation
Reqtify TNI-Valiosys	Traceability and impact analysis; text processing, office tools
Requisite Pro - IBM Rational	Change management, traceability, XML support; work with MS Word, Rational Rose, TeamTest, MS Project
Truereq	Lifecycle-oriented management, team-centric, entry level, XML support;

Supporting action(s)

- OS.GA.a2 Train personnel in Requirements Development and Management Processes

PM.GA.a3 Involve various perspectives in Requirements Development and Management Process — Level 2

It is very important to get different perspective involve appropriately in the Requirements Engineering process. They could be relevant engineers, customers or experts collaborating with the ones responsible for specific tasks. This is to prevent subject views in developing and managing requirements and the RE process.

Supporting action(s)

- RE.SI Stakeholder and Requirements Source Identification

PM. CM Configuration Management

When it comes to working with a large number of or continuously changed requirements, you should always manage the configuration of them to support traceability and avoid confusion.

PM.CM.a1 Manage Versions of Requirements — Level 1

Change happens along the requirements process. Therefore, it is necessary to control the version of your requirements in order to avoid confusion and support traceability. You can choose to use documents to version the requirements or tool support. However, if you work with a huge number of requirements, it is recommended to use a version control system. The historical information of requirements version will help you to trace back when necessary (e.g. when uncovering some mistakes performed on a requirement(s)), and to ensure that the requirements you are working on are the right ones (e.g. the latest requirements instead of an obsolete ones).

Example

You can use CVS, Subversion to support version control [4].

Supporting action(s)

- PM.GA.a2 Introduce Tool Support for Requirements Development and Management

PM.CM.a2 Baseline Requirements — Level 1

This is especially important when you work with a huge number of requirements for the system, and sometime they happen to change continuously. The idea of this activity is that, once your team (and customers) has reached an agreement on a set(s) of requirements, you should capture and save this state of the set(s) as a baseline. This baseline will be served as a stable point for other activities, e.g. implementation, testing, etc... This activity is preferably performed in more stable stages such as after analysis (and negotiation) or when release planning is done.

Supporting action(s)

- PM.GA.a2 Introduce Tool Support for Requirements Development and Management
- PM.CM.a1 Manage Versions of Requirements

PM.CM.a3 Define a Process for Change Control Level 2

Change has been agreed to be the nature of requirements. Since there is no way to avoid it, you had better define a process to control it. The procedure of the change process should be clearly specified. It can cover certain steps a change request must follow and requirements mentioned in the request should be considered. It is also important to specify which factors and which technique to use during re-analyzing process.

Example

You can use tool to support the change process [4]. This will give involved stakeholders a real time view of the change request.

Supporting action(s)

- OS.RR.a3 Define Roles and Responsibilities for Change Control
- PM.GA.a2 Introduce Tool Support for Requirements Development and Management

PM.CM.a4 Track Change Requests Level 2

Since change requests are usually passed through and processed among different stakeholders, you should always keep track on them. The most common and easy way to do so is defining the change request status and keep it up-to-date. Moreover, you should provide a mechanism to ensure issuers can easily and accurately determine the status and disposition of their change requests.

Example

You can use the following status to track the change requests: New, Selected, Implemented, Verified, and Rejected [3].

Supporting action(s)

- PM.CM.a3 Define a Process for Change Control

PM.RT Requirements Traceability Policies

Along the requirements process, you will mostly apply number of changes on requirements. In order to ensure the consistency of the system, it is important that you prepare for these cases so that you can always trace from requirements to other artifacts of the project and vice versa to apply necessary changes.

PM.RT.a1 Uniquely Identify each Requirement — Level 1

Every requirement should have a unique identification. This is especially important when working with a large number of requirements. Having this, the requirements can be easily specified when passing between different stakeholders during the process.

Example

Repository can automatically assign ID for requirements for you [4].

PM.RT.a2 Document Requirements' Source — Level 1

Requirements' source is valuable for traceability, e.g. when need of clarification occurs. In case there is no specific customer, the requirements' source could be the issuers. You can specify and store this information in one of the attributes of individual requirements so that it could be easily found when other stakeholders access the requirement.

Supporting action(s)

- RE.SI.a1 Identify and Involve Relevant Stakeholders
- RE.SI.a3 Identify other Requirements source

PM.RT.a3 Define Traceability Policies — Level 2

Define policies for traceability helps to determine the tracing routines and directions. It is important to specify in the policies the relevant information and artifact which are impacted by requirements changes. It is also necessary to identify the directions such as backward or forward tracing from requirements to other artifacts. The documentation of tracing result also needs to be defined here.

Supporting action(s)

- RE.SI.a1 Identify and Involve Relevant Stakeholders
- RE.SI.a3 Identify other Requirements source
- PM.GA.a1 Define and Maintain Requirements Development and Management Processes

PM.RT.a4 Document Requirements' Relations — Level 2

Requirements' relations are valuable for tracing from requirements to requirements (e.g. when change occurs at requirement A which impacts requirement B). This action should be done together with an analysis of the dependencies between requirements.

Supporting action(s)

- RA.GA.a3 Analyze Requirements Relations
- RA.GA.a4 Identify irrelevant requirements for early dismissal (in/out scope OR Triage)

PM.RT.a5 Document Impact of Requirement on Other Artifacts Level 2

Requirements are the initial images of the system; hence they impact many other artifacts such as test cases, components, modules and so on. Once change occurs, it is important to apply changes in all related artifacts. Therefore, it is necessary to document the impact of requirements on those artifacts. To do so, you can record the related artifacts for each document together with the importance level of this impact. Although this is a quite expensive activity since it requires a lot of effort from different involvements, it is beneficial to ensure the safety of the whole system and to save your effort in later phases (as well as avoid re-work when impact occurs).

Supporting action(s)

- PM.GA.a2 Introduce Tool Support for Requirements Development and Management
- DS.GA.a2 Define Requirements Attributes

PM.RC Requirements communication and negotiation

One important aspect in requirements process management is to ensure the coherence between teams and team members. It is recommended to establish an adequate communication basis among involving parties to reduce gaps (misunderstanding, conflict…).

PM.RC.a1 Establish Effective Communication With Requirements Issuers — Level 1

Requirements are usually gathered from different sources. Therefore, there always exists the need of clarification and verification for them. Apart from that, it is also necessary to observe customers' changes in expectation, especially in MDRE case. In this activity, you should establish an effective communication (i.e. define communication channel, interval…) with the issuers to obtain clear understandings of their desires.

Example

Some means could be used to implement this communication such as: Implementation proposal [5] or prototype or making use of rich communication channels.

Supporting action(s)

- PM.RC.a2 Document requirements source

PM.RC.a2 Obtain common understanding of requirements among different involving roles — Level 3

Common understanding on requirements (i.e meanings, estimation values, prioritization rationale…) should be shared between different involving roles to reduce gaps. This activity may include regular meetings, emails or informal discussions to exchange necessary information. You should consider other teams which will later work with your outputs such as implementing or testing teams.

Supporting action(s)

- RE.SI.a1 Identify and Involve Relevant Stakeholders

RE Requirements Elicitation

Elicitation is the process of discovering, understanding, anticipating and forecasting the needs and wants of the potential stakeholders in order to convey this information to the system developers. The potential stakeholders can include customers, end-users and other people who have the stake in the system development. In the process, the application domain and organizational knowledge are necessary among other things.

RE.GA General Actions

While there is much to say about the actual elicitation practices, in this sub-process area we focus on the overall framework under which you conduct the specific elicitation practices.

RE.GA.a1 Elicit Quality Requirements **Level 1**

Quality requirements, also known as non-functional requirements cover performance, accuracy, reliability, security, usability etc. of the system. Quality requirements are critical because they can affect a large part of the functionality. Not eliciting quality requirements can cause customer disappointment and major rework or product failure.

RE.GA.a2 Qualify and Quantify Quality Requirements **Level 2**

Use appropriate metrics and value to quantify and specify quality requirements in order to understand, measure and test them correctly.

Supporting action(s)

- RE.GA.a1 Elicit Quality Requirements

RE.GA.a3 Let Business Concerns Guide Focus of Elicitation **Level 2**

Business concerns are abstract high-level goals which the product should meet in order to be useful. In the mass market context, business concerns are expressed in the form of product strategies. Critical business concerns such as software reliability, safety and customer service should be elicited. By using the business concerns as means to align the elicitation resources, the time and money spent on elicitation is assured to be aligned with overall goals for the product.

Supporting action(s)

- OS.S.a1 Define Product Strategies

RE.GA.a4 Use Appropriate Elicitation Techniques according to Situation Level 2

Requirements for software-intensive system are complex and varied. Based on each unique case, certain suitable techniques/methods should be chosen and adapted. Some of the method selection criteria are usage context, knowledge types, internal filtering of knowledge and the purpose of requirements.

Example

Some of the useful techniques you can choose to use:

- Observation [6]
- Interview [6]
- Brain storming [6]
- Market survey [7]

RE.GA.a5 Use Artifacts to Facilitate Elicitation Level 2

Additional artifacts like prototypes, scenarios can be used to provide a better understanding of the problems at hand by simulating the interactions of the end-users with the system. By using these artifacts, the end-users can refine their ideas about the system requirements as well as expose their real needs.

Example

You can use scenario analysis [6] and prototype [6] to perform this action.

RE.GA.a6 Create Elicitation Channels for Requirements Sources Level 3

Customer feedback and requests that can be turned into requirements can be captured in many different forms such as incident reports, idea feedback and suggestion. It is recommended to give customers feedback afterwards about their suggested requirements to ensure their continuous contributions. Moreover, company should enable elicitation channels for In-house stakeholders e.g. developers, testers to submit new requirements.

RE.GA.a7 Reuse Requirements Level 3

Reusing requirements is also one of the channels for requirements source. Requirements for a new system can be developed by reusing existing requirements of other systems in the same application domains directly or indirectly. Direct reuse means that minimal modifications will be done to make the existing requirements suitable to the new systems whereas indirect usage means that new requirements are created based on existing ones. In order to effectively reuse requirements, a systematic and planned reuse process has to be defined. Companies can reduce cost, time and risk by using this approach.

RE.SI Stakeholder and Requirements Source Identification

Stakeholders are people who have interests in the product. In order to do successful requirements engineering, it is important to identify whom we would like to listen to and which source of information we can look into in order to elicit requirements for the system.

RE.SI.a1 Identify and Involve Relevant Stakeholders **Level 1**

Explicitly identify all potential stakeholders, who can be customers, end-users, marketing personnel, managers, developers, testers etc, and consult the relevant ones. As different types of stakeholders have different interests and expectations in the system, it is important to distinguish between them in order to elicit all relevant requirements. Among them, customers, end-users and in-house stakeholders are fairly important. Customers are the people who have the authority to purchase/order the system whereas end-users are the ones actually using the system in their work. In some cases, customers may be end-users as well. In-house stakeholders involving in the development/management of the system are often overlooked in the elicitation process. The stakeholders will provide requirements or impose constraints on the system. This ensures that all the concern of affected people will be taken into account.

RE.SI.a2 Identify Other Requirements Sources **Level 1**

Besides stakeholders, other sources of information can also provide requirements for the system. Those sources include regulations, bug reports, market surveys, product reviews, and company standards.

RE.DC Domain Consideration and Knowledge

In order to discover accurately requirements of the systems from various stakeholders, it is required to obtain application domain, organizational as well as other specific knowledge. This SPA consists of many types of knowledge that are necessary to be aware of during elicitation.

RE.DC.a1 Elicit Information about System Domain Restrictions Level 1

Domain experts should be consulted regularly in order to identify the domain constraints imposing on the system. For mass market product, the domain expert should come from inside the organization whereas in customer product, the expert can reside in the customer side. If these constraints are overlooked, it would result in a product failure or legal, organizational, physical obstacles.

Supporting action(s)

- RE.SI.a1 Identify and Involve Relevant Stakeholders
- RE.SI.a3 Identify other Requirements source

RE.DC.a2 Elicit Information about System's Technical Infrastructure Level 1

Technical infrastructure refers to the operating environment in which the system will be installed. It consists of the platform, other hardware and software that interact with the system. Taking into account this information can help to avoid some installation problems.

Supporting action(s)

- RE.SI.a1 Identify and Involve Relevant Stakeholders
- RE.SI.a3 Identify other Requirements source

RE.DC.a3 Elicit Information about System's Business Process Level 1

The system is built because it can contribute to the business of the organization. Therefore, general information about the business process in which the system will function helps to drive the elicitation process forward.

Supporting action(s)

- RE.SI.a1 Identify and Involve Relevant Stakeholders
- RE.SI.a3 Identify other Requirements source

RE.DC.a4 Elicit Information about System's Operational Domain Level 1

A system is developed to support either a new or an existing operational process, for example, producing customer reports. The elicitation process should explicitly describe these business processes in order to reveal process requirements and constraints imposing on the system.

Supporting action(s)

- RE.SI.a1 Identify and Involve Relevant Stakeholders

- RE.SI.a3 Identify other Requirements source

RE.DC.a5 Elicit Information about System Boundaries Level 1

System boundaries define the scope of the system being developed. This information can be obtained by working with the customer or by consulting the product strategies. The information will then be used to focus the effort on the requirements residing within the boundaries.

Supporting action(s)

- RE.SI.a1 Identify and Involve Relevant Stakeholders
- RE.SI.a3 Identify other Requirements source

RE.DC.a5 Consider Sociopolitical Influences on Requirements Sources Level 2

Organizational and political factors can affect or conceal the real system requirements. People may have different hidden agenda and not all of them are willing to contribute to the system being developed. Being aware of these factors can help to understand the real reason for including the requirements.

Supporting action(s)

- RE.SI.a1 Identify and Involve Relevant Stakeholders
- RE.SI.a3 Identify other Requirements source

RA Requirements Analysis

Requirements gathered from different sources need to be analyzed to detect incomplete or incorrect ones as well as to estimate necessary information for later activities (e.g. risk, priorities…). It is also recommended that you should perform some analysis to dismiss irrelevant requirements to avoid wasting effort in next steps.

RA.GA General Actions

RA.GA.a1 Perform Requirements Risk Analysis Level 1

The requirements will need to be analyzed to estimate possible problems arose in the future; hence the managers can have plans to prepare and overcome those risks. In case of products developed for specific customer(s), risk analysis will also provide necessary information for negotiation activities. In case of products developed for mass market, this analysis will be important as an input for release planning. The engineer should perform risk assessment on individual requirements or sets of them or certain selected requirements. In addition, it is also recommended that the probabilities of risks and the effects as well as the significance levels of these effects could be analyzed.

RA.GA.a2 Perform Systematic Requirements Prioritization at Project-level Level 2

In many cases, requirements could be prioritized before they are sent to certain project (e.g. in market-driven development, requirements are usually prioritized at pre-project level to perform release planning). However, at In-project level, there are often more detail requirements introduced in those cases (e.g. features are broken down into implementable and testable requirements). Apart from that, many projects skip the pre-project steps (e.g. in bespoke development). Hence, it is also necessary to prioritize requirements at In-project level. The information of this step will be valuable for negotiation with customer(s) (eg. To eliminate the unnecessary potential requirements) or schedule the implementation of them. The basic aspects can be considered in this step are requirements importance and implementation effort. The requirements priorities analyzed before at pre-project level if available can also be used in this step.

Example

You can choose to use Prioritization techniques [8] such as

- Pair-wise comparisons
- Prioritization working groups
- Scale of 1-to-10 rankings
- Voting schemes (e.g., give each stakeholder a specific number of votes to distribute amongst the requirements or classes of requirements being prioritized)

- Weightings (e.g., weight the votes of different stakeholders)
- Value-Based Software Engineering
- WIN-WIN
- Quality Function Deployment (QFD)
- Cost-value approach
- Focus-point

Supporting Action(s)

- RP.S.a3 Perform Requirements Prioritization at Pre-project level based on various dimensions

RA.GA.a3 Analyze Requirements Relations **Level 2**

It is important during the analysis to consider the relations among requirements. This is based on the fact that requirements are usually not singular. There are requirements that require the others requirements to be implemented before, or exclude the others. Hence, having an overview of these relations will help you in later phase, e.g. in release planning, change control, etc. Besides, it is also very important to consider the relations between quality requirements (also known as non-functional requirements) and functional requirements.

Example

The usual considered relations (mentioned in [9]) are:

- "***Require***" relation: Requirement A depends on requirement B (B need to be implemented before A)
- "***And***" relation: a two-way "require" relation
- "***Or***" relation: when requirement A is similar to requirement B

Apart from the aforementioned requirements dependencies, there are more types of them that should be considered as well. These types of dependencies are more value-related, hence mostly support product planning perspective. Those are:

- **"*ICOST***": A requirement stating that "the system should be able to serve 1000 users concurrently" will typically increase the cost of implementing many other requirements. ICOST relation could cause both negative and positive impacts on other requirements.
- "***CVALUE***": Requirements A may impact value of requirement B (A CVALUE B). For example, a real-time sharing document may decrease the customer value of exporting documents. CVALUE relation could cause both negative and positive impacts on other requirements.

RA.GA.a4 Identify Irrelevant Requirements for Early Dismissal (in/out scope OR Triage) — Level 2

This step is to early dismiss the irrelevant requirements so that the huge amount of initial requirements could be reduced to avoid wasting time and effort for future works. In order to perform this step, the requirements should be aligned with the boundaries of the developing system. Requirements which are not in-scoped should be eliminated. The boundaries of the system can be defined by discussions with customers. In case there is no specific customer, requirements should be compared to the strategies and plans of the product.

Example

Model for Early Requirements Triage and Selection (MERTS) can be used as a tool for requirements early dismissing in case there is no specific customer [10].

Supporting action(s)

- RE.DC.a4 Elicit Information about System Boundaries
- OS.S.a1 Define Product Strategies

RA.GA.a5 Analyze the Strength of Relations between Requirements — Level 2

In addition to the types of relations, you should also estimate the strength of the detected relations. This information will be very useful for later activities in negotiation or release planning when you need to consider and make the trade-off.

However, it is always difficult to keep track of all kinds of dependencies and to visualize them. Hence, it is also important that you should choose the necessary aspects that you are interested in and will use in later activities.

RA.GA.a6 Perform Refinement and Abstraction of Requirements — Level 3

Requirements often come from various sources hence they are usually diverse in levels of abstraction. By performing refinement/abstraction to synchronize their abstraction levels, it will be easier for understanding, managing and further activities such as prioritization. It is also recommended to keep multiple levels of abstraction, for instance: high level which consists of more general and goal-like requirements, and low level which contains more specific requirements for implementation.

Example

One stakeholder requires “System must look user-friendly” while another asks for something like “Waiting time does not exceed 5 seconds”. You can make use of Requirements Abstract Model (RAM) to perform this step [11]. This model is validated in industry and very useful for this purpose.

RA.QA Quality attributes analysis

In this step, the elicited requirements will be analyzed in order to ensure their quality aspects such as completeness, correctness and testability.

RA.QA.a1 Analyze for Missing and Double Requirements **Level 1**

After elicitation phase, the raw requirements need to be analyzed to detect missing and overlapped requirements. This step is to uncover the incomplete requirements so that a clarification could be made with the source of requirements to obtain the correct desires or expectations of the stakeholders.

RA.QA.a2 Analyze for Ambiguous Requirements **Level 1**

The requirements will also be analyzed to uncover volatility. At this step, the requirements will be checked whether they are clear enough for readers to understand and to be implemented. If volatility is detected, a clarification could be made with the source of requirements to obtain the clearer demand from the stakeholders.

RA.QA.a3 Analyze for Correctness of Requirements **Level 1**

The requirements also need to be checked in term of correctness since many of those are proposed from “non-it” users. Some may even conflict to the other requirements. Hence it is necessary to investigate the incorrect requirements and clarification can be made if necessary.

RA.QA.a4 Analyze for Testability of Requirements **Level 1**

This step is to uncover the inadequate requirements meaning requirements in which information is not sufficient for testing in next phases. It is usually the case of quality requirements. A clarification could be made with the source of requirements to obtain more detail desires or expectations of the stakeholders.

RA.PS Problems and solutions analysis

RA.PS.a1 Prototyping **Level 1**

Additional artifacts like prototypes, scenarios can be used to provide a better understanding of the problems at hand by simulating the interactions of the end-users with the system. By using these artifacts, the end-users can refine their ideas about the system requirements as well as expose their real needs.

Example

You can use Scenario analysis [6] to perform this action.

RA.PS.a2 Perform Systems Modeling **Level 3**

System modeling covers models of system specification information, system environment and system architecture. Different parts of the system can be modeled, in the context of business processes that may use the system. The different sub-systems existing within the system and the links between them are also necessary to be described here.

Example

You can make use of data processing models, composition models, classification models, stimulus-response model and process model to demonstrate system models [12].

RP Release planning

Release planning covers crucial steps aiming to determine the optimal set of requirements for a certain release to be implemented at a defined/estimated time and cost to achieve some goals. Performing this step carelessly would lead to high risky situations or fail to achieve planned goals. For example, placing important features at a too late release would make the product miss the right moment to gain the customers' impression.

RP.GA General Actions

RP.GA.a1 Synchronize Release Plan with Product Roadmap Level 2

Product roadmap is important to support the planners in determining the contents of a release. By aligning the requirements with the product plan (including strategies and time) in the roadmap, the planners could easily consider whether the requirements should be included or excluded in a certain release.

Example: If the roadmap states that the upcoming release should target at Chinese market; requirements investigated from Chinese market such as Chinese language feature, Chinese keyboard feature, etc. should be of higher priorities.

In addition, product-technology road map gives planners an overview of the relationship among product releases and their evolvement along the time axis. Hence, it is beneficial for planners to use product-technology road map along their decision process to decide which requirements need to be postponed or excluded, etc. in a certain release due to technology constraints.

Vice versa, the planners should also consider how new features will impact the existing product.

Supporting action(s)

- OS.RR.a3 Define Roles and Responsibilities for Release Planning Activities
- OS.S.a2 Define Product Roadmaps

RP.GA.a2 Involve Different Perspectives in Release Planning Level 2

As mentioned above, release planning itself is a trade-off among customer-value, financial value, developing cost, risk, etc. The nature of it clearly shows a need of involving different perspective in the process. The four perspectives: product management, marketing, development and finances preferably participate in this step. Besides, it is also beneficial to involve external customers in the process to achieve external view of the products. It is also recommended that the different perspectives could form a cross-functional team and work together along the product life cycle in order to achieve a mutual understanding and improve the decision making quality.

One way to perform this diverse involvement is to get different perspectives participate in prioritization and give them appropriate weights based on their importance.

Supporting action(s)

- RE.SI.a1 Identify and Involve Relevant Stakeholders
- OS.RR.a2 Define Roles and Responsibilities for Release Planning Activities

RP.GA.a3 Post Requirement Selection Evaluation Level 3

Post-release evaluation is the step to assess the quality of the requirements selection in the previous step. This is to uncover previous mistakes/misunderstanding, gain experiences in decision making, hence ensure the quality of decisions for future releases.

The evaluation can be done by analyzing the measurements such as customer value, market penetration, profit, and revenue etc. of different product releases after they were released. Based on this information, the planners could see whether he/she had made a correct decision at that time and further investigate the mistakes.

Example

PARSEQ (Post- Release Analysis of Requirements Selection Quality) is an industrial-validated method supporting this step [13].

Supporting action(s)

- OS.RR.a2 Define Roles and Responsibilities for Release Planning Activities

RP.GA.a4 Plan Multiple Releases at Pre-defined Interval Level 3

Although it is obvious that not all of the potential requirements could be included in one release, it is always recommended to show the plan to implement them in next few releases. The reason is that, requirements present customers' desires, and excluded requirements indicate that customers might be disappointed. Hence, having few releases ahead enable sale personnel to be able to show that they may be compensated in next releases. This is very important for customer-relationship development.

The planning should be undertaken at pre-defined interval since market-driven planning is more like a trade-off between current state and future. Therefore, regularly reviewing the plans will give more chances to evaluate the decisions, hence re-plan to adapt to the current situation.

RP.S Requirements Selection

RP.S.a1 Pack Requirements into Release — Level 1

Requirements after being prioritized will be selected to certain releases. The selection activity usually requires the involvement of different perspectives from marketing, developing and management, etc. Besides, you should also consider the interdependences when packing requirements into release. There are requirements with low priorities but mandatory for other higher priority ones, hence taking the relations into account is very important. Currently, there is no tool that fully support for this activity even it is considered a very challenging one.

Supporting action(s)

- OS.RR.a3 Define Roles and Responsibilities for Release Planning Activities
- RP.S.a3 Perform Requirements Prioritization at Pre-project level based on various dimensions
- RA.GA.a3 Analyze Requirements Functional Dependencies
- RA.GA.a5 Analyze Value-related Dependencies between Requirements

RP.S.a2 Estimate Cost and Value of Requirements — Level 2

The cost for implementing each requirements and their value need to be estimated in order to identify benefit can be obtained on individual requirements and set of them. Besides, this information also helps the decision making in release planning process to make the trade-off between cost and benefit of releases.

RP.S.a3 Perform Requirements Prioritization at Pre-project Level based on Various Dimensions — Level 2

Requirements prioritization at pre-project level helps to determine the relative necessity of the requirements. With a huge number of mandatory requirements which are impossible to be implemented all at the same time, it is crucial to specify which are more critical than others.

In addition, requirements need to be prioritized along more than one dimension (related or even opposing ones). And these dimensions can be valued differently by different stakeholders. Usually, customer-value, cost and interdependencies are considered as the basic dimensions. Customer-value present customer preference of the requirements while cost presents how much would be spent to implement the requirements (in finance and man month).

Apart from the aforementioned dimensions, the prioritization can also take into account additional ones such as business value, risk, harm avoidance, legal mandate, etc.

Usually the result of prioritization is served as an input for requirements selection. Hence, the more aspects are considered, the more carefully the selection can be performed which

can result in a better decision.

Example

Several prioritization techniques [8] are available and validated for the engineers to choose, namely:

- Pair-wise comparisons
- Prioritization working groups
- Scale of 1-to-10 rankings
- Voting schemes (e.g., give each stakeholder a specific number of votes to distribute amongst the requirements or classes of requirements being prioritized)
- Weightings (e.g., weight the votes of different stakeholders)
- Value-Based Software Engineering [Boehm 2003]
- WIN-WIN [Boehm 2001]
- Quality Function Deployment (QFD)
- Cost-benefit approach
- Focus-point

DS Documentation and Requirements Specification

Documentation and Requirement specification deal with how a company organizes requirements and other knowledge gathered during requirements engineering process into consistent, accessible and reviewable documents. The software requirements specification (SRS) contains the product's detailed functional and quality requirements.

DS.GA General Actions

DS.GA.a1 Establish Standardized Structure for SRS **Level 1**

Companies should define a common standard structure which reflects the best practice to organize the requirements document in the companies. The best structures vary among companies as they are influenced by the custom of companies, the type of products developed and the development processes. The common structure helps users to understand the document faster and assure high quality of documents.

DS.GA.a2 Define Requirements Attributes **Level 1**

Each requirement is specified with a number of attributes associated with it. Attributes are assigned values to reflect what is known about the requirement such as estimated cost, priority, state. Different attributes are specified and utilized depending on the various needs of the companies. The benefit of having attributes is to separate important pieces of information about a requirement from its description. Hence, companies can manage requirements more effectively and efficiently by looking at different properties of the requirements. In case requirements are stored in a database, managers can use tool support to simply query, sort or filter the requirements.

Example

Some of the attributes that can be present are ID, Title, Description, Requirement Source, Status and Rationale [14] [11].

Supporting action(s)

- PM.GA.a2 Introduce Tool Support for Requirements Development and Management

DS.GA.a3 Define Requirements States — Level 2

The states of the requirements represent their refinement levels in the progress towards release. Some possible states are New (requirement is issued), Selected (requirement is analyzed and selected for implementation), Implemented (requirement is successfully realized), Rejected (requirement is excluded). Tracking requirements states help to monitor the requirements and project progress more accurately.

Supporting action(s)

- DS.GA.a2 Define Requirements Attributes
- PM.GA.a2 Introduce Tool Support for Requirements Development and Management

DS.GA.a4 Document Requirements Rationale — Level 2

The reason why a requirement is included should be recorded in order for the readers to understand the requirements. It is extremely useful in case those who initially defined the requirements have left the company. Moreover, it will help problem expert to check if the requirements are consistent with the problem being solved.

Supporting action(s)

- DS.GA.a1 Define Requirements Attributes
- RE.DC Domain Consideration and Knowledge

DS.GA.a5 Record Rationale for Rejected Requirements — Level 3

When requirements are rejected after analysis or negotiation, the reason for rejection and who rejected it should be recorded immediately to avoid being forgotten. This information will be helpful for future reference when dismissed requirements resurface as they can be checked without spending effort in re-analysis.

Supporting action(s)

- DS.GA.a3 Define Requirements States
- RA.GA.a4 Identify irrelevant requirements for early dismiss (in/out scope OR Triage)
- RP.S.a1 Pack requirements into release

DS.DD Documentation Deliverables

The documentation deliverables imply all deliverables the requirements engineering process supposes to produce. You should clearly define the expected deliverables of RE process at the beginning of it as requirements for the project itself. This activity can also be done together with members from other phases that will use the information later.

DS.DD.a1 Define User Documentation Deliverables **Level 2**

The user deliverables may consist of user manual, user dictionary, etc. This aims to describe the system from user points of view and how to use it.

DS.DD.a2 Define System Documentation Deliverables **Level 2**

Management deliverables cover all the necessary documents for managing the system such as: system design, technical specification, etc.

DS.DD.a3 Define Management Documentation Deliverables **Level 3**

Management deliverables cover all the necessary documents for managing the system such as maintenance, administrative manual, etc.

RV Requirements Validation

This process involves checking the documented requirements against defined quality standards and the real needs of various stakeholders. It ensures that the documented requirements are complete, correct, consistent, and unambiguous.

RV.GA General Actions

RV.GA.a1 Validate requirements with relevant stakeholders **Level 1**

Requirements must be validated with the relevant stakeholders in order to ensure their consistency, completeness and adequacy. Moreover, the intent and interpretation of the requirements can also be verified.

RV.GA.a2 Use Checklist to Ensure Quality of Requirements **Level 1**

Checklist draws the participants' attention to the certain aspects of the requirements document as well as the frequently encountered problems. Checklist should not be too long to prevent people from referring to the list too often.

RV.GA.a3 Review Requirements **Level 2**

Reviewing is the technique involving peers (someone other than the author) to examine the requirements and identify defects. The author is then responsible for correcting the found problems.

Supporting action(s)

- RV.GA.a2 Use Checklist to Ensure Quality of Requirements

RV.GA.a4 Organize Inspections **Level 3**

Inspections are formal meetings in which a small team of inspectors with different perspectives (e.g. customer, analyst, developer, tester) carefully examine the requirements, detect errors and resolve them together. This technique can identify a high percentage of requirements errors but requires all parties to be present at the same time.

Supporting action(s)

- RV.GA.a2 Use Checklist to Ensure Quality of Requirements

RV.GA.a5 Develop Preliminary Test Case or User Manual **Level 3**

Creating possible test cases or writing a draft user manual can force a detailed look at the requirements and uncover problems with the requirements document related to ambiguities, inconsistencies or usability. The test cases or draft user manual can be used

later as a basis for actual artifacts.

RV.GA.a6 Use system model paraphrasing — Level 3

Converting system models into natural language enables general stakeholders to understand these models more clearly and comment on them. In this way, additional requirements or problems will be detected.

Supporting action(s)

- RE.EP.a4 Create Artifacts to Facilitate Elicitation

RV.GA.a7 Define Acceptance Criteria and Acceptance Tests — Level 3

An effective technique to validate requirements is by having customers define the acceptance criteria. The acceptance criteria and acceptance test determine if requirements are right and the product satisfies them. They are used to validate the most commonly used and important use cases and requirements but they do not replace system testing.

2. Maturity Level View

In this section, the model can be viewed by maturity level. This view shows the practices from all process areas which the organization should implement in order to achieve a specific maturity level.

Level 1 – Basic		
ID	**Title**	**Level**
OS	**Organizational Support**	
OS.GA	General Actions	
OS.GA.a1	Create a Product-wide Glossary of Terms	1
OS.RR	Roles and Responsibilities	
OS.RR.a1	Assign Owner(s) of Requirements Development and Management Processes	1
PM	**Requirements Process Management**	
PM.GA	General Actions	
PM.GA.a1	Define and Maintain Requirements Development and Management Processes	1
PM.GA.a2	Introduce Tool Support for Requirements Development and Management	1
PM.CM	Configuration Management	
PM.CM.a1	Manage Versions of Requirements	1
PM.CM.a2	Baseline Requirements	1
PM.RT	Requirements Traceability Policies	
PM.RT.a1	Uniquely Identify each Requirement	1
PM.RT.a2	Document Requirements' Source	1
PM.RC	Requirements Communication and Negotiation	
PM.RC.a1	Establish Effective Communication With Requirements Issuers	1
RE	**Requirements Elicitation**	
RE.GA	General Actions	
RE.GA.a1	Elicit Quality Requirements	1
RE.SI	Stakeholder and Requirements Source Identification	
RE.SI.a1	Identify and Involve Relevant Stakeholders	1
RE.SI.a2	Identify Other Requirements Sources	1
RE.DC	Domain Consideration and Knowledge	
RE.DC.a1	Elicit Information about System Domain Restrictions	1
RE.DC.a2	Elicit Information about System's Technical Infrastructure	1
RE.DC.a3	Elicit Information about System's Business Process	1
RE.DC.a4	Elicit Information about System's Operational Domain	1
RE.DC.a5	Elicit Information about System Boundaries	1
RA	**Requirements Analysis**	
RA.GA	General Actions	
RA.GA.a1	Perform Requirements Risk Analysis	1
RA.QA	Quality attributes analysis	
RA.QA.a1	Analyze for Missing and Double Requirements	1
RA.QA.a2	Analyze for Ambiguous Requirements	1
RA.QA.a3	Analyze for Correctness of Requirements	1
RA.QA.a4	Analyze for Testability of Requirements	1

RA.PS	Problems and solutions analysis	
RA.PS.a1	Create Prototype	1
RP	**Release Planning**	
RP.S	Requirements Selection	
RP.S.a1	Pack Requirements into Releases	1
DS	**Documentation and Requirements Specification**	
DS.GA	General Actions	
DS.GA.a1	Establish Standardized Structure for SRS	1
DS.GA.a2	Define Requirements Attributes	1
RV	**Requirements Validation**	
RV.GA	General Actions	
RV.GA.a1	Validate requirements with relevant stakeholders	1
RV.GA.a2	Use Checklist to Ensure Quality of Requirements	1

Level 2 - Intermediate		
OS	**Organizational Support**	
OS.GA	General Actions	
OS.GA.a2	Train personnel in Requirements Development and Management Processes	2
OS.RR	Roles and Responsibilities	
OS.RR.a2	Define Roles and Responsibilities for Requirements Development and Management Processes	2
OS.RR.a3	Define Roles and Responsibilities for Release Planning	2
OS.RR.a4	Define Roles and Responsibilities for Change Control	2
OS.S	Strategies	
OS.S.a1	Define Product Strategies	2
OS.S.a2	Define Product Roadmaps	2
PM	**Requirements Process Management**	
PM.GA	General Actions	
PM.GA.a3	Involve various perspectives in Requirement Development and Management Process	2
PM.CM	Configuration Management	
PM.CM.a3	Define a Process for Managing Change and Evolution	2
PM.CM.a4	Track change requests	2
PM.RT	Requirements Traceability Policy	
PM.RT.a3	Define traceability policies	2
PM.RT.a4	Document Requirements' Relation	2
PM.RT.a5	Document Impact of Requirement on Other Artifacts	2
RE	**Requirements Elicitation**	
RE.GA	General Actions	
RE.GA.a2	Qualify and Quantify Quality Requirements	2
RE.GA.a3	Let Business Concern Guide Focus of Elicitation	2
RE.GA.a4	Use Appropriate Elicitation Techniques according to Situation	2
RE.GA.a5	Use Artifacts to Facilitate Elicitation	2
RE.DC	Domain Consideration and Knowledge	

RE.DC.a6	Consider Sociopolitical Influences on Requirements Sources	2
RA	**Requirements Analysis**	
RA.GA	General Actions	
RA.GA.a2	Perform Systematic Requirements Prioritization at Project-level	2
RA.GA.a3	Analyze Requirements Relations	2
RA.GA.a4	Identify Irrelevant Requirements for Early Dismissal (in/out scope OR Triage)	2
RA.GA.a5	Analyze the Strength of Relations between Requirements	2
RP	**Release Planning**	
RP.GA	General Actions	
RP.GA.a1	Synchronize Release Plan with Product Roadmap	2
RP.GA.a2	Involve different perspectives in release planning	2
RP.S	Requirements Selection	
RP.S.a2	Estimate Cost and Value of Requirements	2
RP.S.a3	Perform Requirements Prioritization at Pre-project Level based on Various Dimensions	2
DS	**Documentation and Requirements Specification**	
DS.GA	General Actions	
DS.GA.a3	Define Requirements States	2
DS.GA.a4	Document Requirements Rationale	2
DS.DD	Documentation Deliverables	
DS.DD.a1	Define User Documentation Deliverables	2
DS.DD.a2	Define System Documentation Deliverables	2
RV	**Requirements Validation**	
RV.GA	General Actions	
RV.GA.a3	Review Requirements	2

Level 3 -Advanced		
OS	**Organizational Support**	
OS.RR	Roles and Responsibilities	
OS.RR.a5	Define Roles and Responsibilities for Product Management	3
OS.S	Strategies	
OS.S.a3	Communicate Strategies in Organization	3
PM	**Requirements Process Management**	
PM.RC	Requirements Communication and Negotiation	
PM.RC.a2	Obtain common understanding of requirements among different involving roles	3
RE	**Requirements Elicitation**	
RE.GA	General Actions	
RE.GA.a6	Create Elicitation Channels for Requirements Sources	3
RE.GA.a7	Reuse Requirements	3
RA	**Requirements Analysis**	
RA.GA	General Actions	
RA.GA.a6	Perform refinement and abstraction of requirements	3
RA.PS	Problems and solutions analysis	

RA.PS.a2	Perform Systems Modeling	3
RP	**Release Planning**	
RP.GA	General Actions	
RP.GA.a3	Post Requirement Selection Evaluation	3
RP.GA.a4	Plan multiple release at pre-defined interval	3
DS	**Documentation and Requirements Specification**	
DS.GA	General Actions	
DS.GA.a5	Record Rationale for Rejected Requirements	3
DS.DD	Documentation Deliverables	
DS.DD.a3	Define Management Documentation Deliverables	3
RV	**Requirements Validation**	
RV.GA	General Actions	
RV.GA.a4	Organize Inspections	3
RV.GA.a5	Develop Preliminary Test Case or User Manual	3
RV.GA.a6	Use System Model Paraphrasing for QA	3
RV.GA.a7	Define Acceptance Criteria and Acceptance Tests	3

Reference

[1] A.C. Yeh, “Requirements Engineering Support Technique (REQUEST) A Market Driven Requirements Management Process,” 1992, pp. 211-223.

[2] B. Regnell, P. Beremark, and O. Eklundh, “A market-driven requirements engineering process: results from an industrial process improvement programme,” UK: Springer-Verlag, 1998, pp. 121-9.

[3] P. Carlshamre and B. Regnell, “Requirements lifecycle management and release planning in market-driven requirements engineering processes,” Los Alamitos, CA, USA: IEEE Comput. Soc, 2000, pp. 961-5.

[4] R. Wieringa and C. Ebert, “RE'03: Practical requirements engineering solutions,” IEEE Software, vol. 21, 2004, pp. 16-17.

[5] Samuel Fricker, Tony Gorschek, Carl Byman, Armin Schmidle, "Handshaking with Implementation Proposals: Negotiating Requirements Understanding," IEEE Software, vol. 27, no. 2, pp. 72-80, Mar./Apr. 2010, doi:10.1109/MS.2009.195

[6] N.A.M. Maiden and G. Rugg, “ACRE: Selecting methods for requirements acquisition,” Software Engineering Journal, vol. 11, 1996, pp. 183-192.

[7] CMMI for Development, Version 1.2, CMMI-DEV v1.2, CMU/SEI-2006-TR-008, Technical Report, Software Engineering Institute, August 2006

[8] Firesmith, “Prioritizing requirements,” *Journal of Object Technology*, vol. 3, 2004, pp. 35-47.

[9] P. Carlshamre, K. Sandahl, M. Lindvall, B. Regnell, and J. Natt och Dag, “An industrial survey of requirements interdependencies in software product release planning,” Los Alamitos, CA, USA: IEEE Comput. Soc, 2000, pp. 84-91.

[10] M. Khurum, K. Aslam, and T. Gorschek, “A method for early requirements triage and selection utilizing product strategies”, Piscataway, NJ, USA: IEEE, 2008, pp. 97-104.

[11] T. Gorschek and C. Wohlin, “Requirements abstraction model,” *Requirements Engineering*, vol. 11, 2006, pp. 79-101.

[12] Gorschek T., Tejle K., “A Method for Assessing Requirements Engineering Process Maturity in Software Projects”, Blekinge Institute of Technology, Master Thesis Computer Science no. MSC-2002:2, 2002.

[13] L. Karlsson and B. Regnell, “Introducing tool support for retrospective analysis of release planning decisions,” Berlin, Germany: Springer-Verlag, 2006, pp. 19-33.

[14] P. Sawyer, I. Sommerville, and G. Kotonya, “Improving market-driven RE processes,” Espoo, Finland: Tech. Res. Centre Finland, 1999, pp. 222-36.