Requirements Engineering Process Maturity Model for Market Driven Projects

- The REPM-M Model

Rashid Awan
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 16 weeks of full time studies.

Contact Information:
Author(s):
Rashid Awan
E-mail: rashid.awan@gmail.com

University advisor(s):
Göran Fries
Department of Computer Science and Software Engineering

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

Internet : www.tek.bth.se  
Phone : +46 457 38 50 00  
Fax : + 46 457 271 25
Several software projects are over budgeted or have to face failures during operations. One big reason of this is Software Company develops wrong software due to wrong interpretation of requirements. Requirements engineering is one of the well known discipline within Software engineering which deals with this problem. RE is the process of eliciting, analyzing and specifying requirements so that there won’t be any ambiguity between the development company and the customers. Another emerging discipline within requirements engineering is requirements engineering for market driven projects. It deals with the requirements engineering of a product targeting a mass market. In this thesis, a maturity model is developed which can be used to assess the maturity of requirements engineering process for market driven projects. The objective of this model is to provide a quick assessment tool through which a company would be able to know what are the strengths and weaknesses of their requirements engineering process.

**Keywords:** Requirements Engineering, Market-Driven Projects, Product, Maturity Model.
1 INTRODUCTION

This chapter provides basic information about software engineering and its importance for software development. In addition, a discussion over the process of requirements engineering and maturity models is given here so that readers can easily understand the information given in the rest of the document. An introduction of REPM-M model and the purpose of this thesis are also presented in this chapter. A road map can be found in the last section.

1.1 Software

Computer software plays an important role in our daily life. Whether it is a manufacturing industry or educational institute, finance or government sector, entertainment or health care, you can find software everywhere. Software is often seen as computer programs, which fulfill the needs of specific people or of a general market. This definition is a little limited. Software also has some associated documentations [5]. Those documentations may either be for developers of the software telling them what to develop, usually called system specification or for end users describing them how to use the software, often called user documentation or end user manual. Development of a software product is not a straightforward process. It involves several technical and managerial aspects. To deal with those aspects, a discipline is introduced called “Software Engineering”.

1.2 Software Engineering

Software engineering is an engineering discipline, which deals with all aspects of software development, including technical and managerial, throughout whole life cycle [5]. Software engineering provides different ways of how effectively a software product can be produced in terms of cost and quality. Those ways include common methods, techniques and tools. A primary goal of software engineering discipline is to provide a cost effective way of development of a software product.

1.2.1 Bespoke and market Driven Products

From the perspective of a software engineer, a software can be categorized in two categories i.e. Bespoke and market driven. Bespoke products are developed for a specific customer or a group of customers. Bespoke products are also termed as Customer driven or customized products. We will use the term bespoke products in the rest of this document for these kinds of products. Market driven products are intended to develop for an open market and can be sold to any customer. Market driven products are also called “Generic products” because these products are developed in a way which covers a broader domain. On a very high level, we can say that target customers are defined in bespoke products while it is not the case for market driven products. Usually bespoke products are ordered by a specific customer while a market driven product is produced by a development organization to sell it in an open market [5].
1.2.2 Software Requirements

Before going into the detail of requirements engineering, we should have a clear definition of what a software requirement is. A requirement is something which is obligatory. According to SEI [6], a requirement is

"Function or characteristic of a system that is necessary...the quantifiable and verifiable behaviors that a system must possess and constraints that a system must work within to satisfy an organization’s objectives and solve a set of problems”

A software requirement is a characteristic or functionality of a system which a system should have in order to work properly. On the basis of their nature, software requirements can be categorized into two categories i.e. Functional and non functional requirements. On a very high level, functional requirements depict what the system should do and non functional requirements describe how those functional requirements should be implemented [1]. Non functional requirements can be seen as constraints over functional requirements. For example, a functional requirement can be system should be used by an authenticated user and non functional requirements may be system should authenticate the user within 5 seconds after he enters his information.

1.3 Requirements Engineering

Several authors and researchers have given different definitions of requirements engineering. According to Sommerville [1],

“Requirements engineering is a process which covers all the activities of discovering, documenting, and maintaining a set of requirements for a computer-based system.”

Dean Leffingwell [10] presents a similar definition but call the process of requirements engineering as requirements management process instead of requirements engineering. Despite all of these different definitions, the main goal of requirements engineering is to develop unambiguous and desired systems requirements. A typical requirements engineering process comprises eliciting requirements, analysis and negotiation, documentation, validation and management [2].

Stakeholder is an important term in requirements engineering. Stakeholder may be any person or a system which has an impact/stake over the prospective system. Stakeholders may include Buyers, Managers, requirements engineers, developers, testers, end users, competitors, similar systems, consultants and also any other person who has domain knowledge of the system [1] [3].

1.3.1 Requirements Elicitation

Requirements elicitation is usually considered as first phase of requirements engineering in which requirements are captured from different sources. The process of requirements elicitation seems straightforward. One can simply think it as collecting all the stakeholder and future users of the system and asking them about what they actually want. But the results show that poorly elicited requirements are one of the drawbacks in software development.

There are several techniques have been introduced in order to discover requirements from customers i.e. interviews, surveys, questionnaires, requirements
workshop, brainstorming sessions, storyboards [2] [10]. The choice of techniques are usually depend on the type of the project company is dealing with.

Reusing requirements from previous projects are also considered one of the techniques for requirements elicitation. This technique is quite effective in terms of cost and time. The reason is because reused requirements are already verified. It has been observed that around 80% of the total requirements are same for similar systems.

1.3.2 Requirements Analysis and Negotiation

Once the requirements are elicited, they should be analyzed in order to find conflicts, overlaps, omissions and inconsistencies. These activities are covered in the phase of requirements analysis and negotiation [2]. Negotiation with customers is also carried out during this phase. The objective of this phase is to develop an agreed set of requirements which are complete and consistent [1]. System boundaries are defined in this phase in order to eliminate unnecessary requirements. Companies often use a checklist for conflict resolution and completeness checking. The checklist may change from company to company and project to project.

Purpose of requirements negotiation is to keep the most important requirements in the requirements document. Requirements are usually assigned priorities in order to negotiate them easily. Robertson [3] keeps two fields in specification of a particular requirement in order to prioritize requirements. The two fields are customer satisfaction and customer dissatisfaction. Both fields may have value from 1 to 5 according to customers’ priority for the requirement. The higher the value of those fields, the higher the importance of requirement for the customer. One of the major problems with requirements negotiation is occurred when it is asked from the customer that which requirements are most important, they reply that every requirement is important.

1.3.3 Requirements Documentation

Requirements are described in a document by using a natural language for example, English, Swedish, French. In some cases, requirements are also described in any other special language depending on the culture of the organization. Requirements should be specified in a way so that every one can understand it easily. In addition, one should take care of the fact that reader may interpret different meanings of one sentence which may cause contractual disagreements [1]. Writing requirements is a continuous process and often termed as requirements documentation. Description of requirements may also be accompanied through figures, tables and graphs.

According to Summerville [1], three factors should bee kept in mind while developing a requirements document. First, writer should invest enough time and effort in writing requirements because requirements will be read several times. Second, writer should not assume that reader will be from same background and would have same knowledge. Third, writing requirements in a clear and concise way is not an easy task. It requires more time and concentration to write a requirement in a good meaningful way.

1.3.4 Requirements Validation

Requirements validation is the process of checking the requirements document for omission, conflicts and ambiguities. Also, the quality of requirements document is assured in this phase whether the produced document is up to the organization standard. In contrast with the process of requirements analysis, requirements validation is a rather formal process in which requirements document is formally inspected and reviewed. In requirements analysis, more emphasize is given over individual
requirements. On the other hand, in requirements validation, requirements document is validated in order to verify any lack of conformance to quality standards, ambiguous requirements, and requirements conflict which may be overlook in requirements analysis phase [1]. Techniques which are common for requirements validation are, requirements inspections, validation checklist, requirements review, developing test cases and/or user manual draft etc.

1.3.5 Requirements Management

The process of managing change in system requirements is called Requirements Management [2]. Requirements management also includes how to trace connected requirements as well as how to use a database to store requirements.

Typical activities which are carried out during this phase are, establishing traceability policies, use of database to manage requirements, defining change management policies, identification of global and volatile requirements, record of rejected requirements, etc.

Even though according to researchers Requirements documentation, requirements validation and requirements management are three different activities, we keep these three activities under the same MPA (See Chapter 2 for detail). The reason of keeping it in this way is also discussed in chapter two.

1.4 Issues in Requirements Engineering

Bad elicited and analyzed requirements cost a lot when identified later in the software development life cycle. There are different issues which are usually involved in the process of requirements engineering [7] [8] [10]. Certain requirements are usually ignored by assuming that these requirements won’t be of that much importance. An adequate and continuous interaction with customers is always required in order negotiate requirements with customers. A number of issues are occurred when process of requirements elicitation is carried out.

Those issues should be overcome in order to produce good basis of requirements. Problems of scope, problems of understanding and volatile requirements are some major issues which should be carefully dealt with. A major problem is to understand what customer actually wants. It happens since analyst and customers are from different backgrounds. Changing requirements is also a big factor which causes problems in requirements engineering.

1.5 Maturity Models (CMM and ISO 9000:2000)

US department of Defense’s Software Engineering Institute developed a model to assess the capabilities of companies. The model is known as Capability Maturity Model (CMM) [11]. According to SEI [11], immature organizations focused more on immediate crises solutions. The processes are usually executed on Ad-Hoc basis and processes are improvised by practitioners and managers. There is no defined method to judge the quality of a produced document in immature organizations. On the other hand, mature organization has an organization wide defined process and ability for managing software development and maintenance processes.

ISO 9000 is another maturity model which can be used to assess the maturity of software organizations. In contrast to CMM, which focuses software development exclusively, ISO standards focus relatively broader set of standards intended to cover other business process activities from domains other than software development as
well [10]. The ISO standards apply in almost all aspects of operations from sales to customer support.

In this section, we will focus more towards CMM as compared to ISO. One and the most obvious reason is that the REPM model is much similar to CMM rather than ISO.

1.5.1 Capability Maturity Model (CMM)

Continuous process improvement is based on small defined steps rather than a revolutionary innovation. By following the same fact, CMM provides a framework consisting small revolutionary steps to build the foundation of continuous process improvement. CMM rates the companies from level 1 to level 5.0. These levels are called Maturity Levels. The higher the rating, the higher the maturity of the software organizations. To achieve a certain maturity level, company should fulfill a certain action given for that particular level. The five maturity levels are [11]

1.5.1.1 Initial level – 1

An organization at this level completes software processes on Ad-Hoc basis, and even chaotic. Some processes are defined and success is mostly dependent on individual efforts. The process, for these kinds of organizations, is just like “Fire Fighting”. Data collection and its analysis are on ad-hoc basis. Even though these companies can successfully and quite frequently develop products, chances of exceeding budget and time is quite high.

1.5.1.2 Repeatable Level – 2

Basic project management capabilities for example, costing and scheduling are established in an organization at level 2. They have started improving their process by learning from their previous experiences. Processes are defined, documented, practiced, trained, measured, enforced and improvable. Planning and tracking of the
A software project is defined. Achievable plans are maintained based on the performance of previous projects. Processes may differ in between different projects for an organization at level 2.

### 1.5.1.3 Defined Level – 3

In a company at defined level, software process for management and engineering activities are documented, standardized and integrated into a standard software process for organization. Problems are predictable and avoided before they become severe. Organization relies on team work rather than the performance an individual. Training environment is planned and established. New technologies are evaluated at a qualitative basis. Collection of data is defined and their analysis is used in processes. Information is systematically shared across projects.

### 1.5.1.4 Managed level – 4

At managed level, detailed measurements of software process and product quality are collected. Process is rather predictable at this level. Company relies on quantitative measurements of processes. Modern technologies are evaluated on quantitative basis.

### 1.5.1.5 Optimizing level – 5

An environment of continuous process improvement is established in an organization at optimizing level. Innovative ideas and quantitative feedback helps in improving the process within organization. Defects are identified and prevented in software projects. Their causes are analyzed in order to keep them away in future projects. New technologies are introduced by using a technology change management process. Team work is strongly appreciated and an environment is created to support team work.

Each level in CMM has a certain number of key process areas, KPA in short, except level 1. Each KPA has defined goals with a set of activities. These activities require to be performed in order to achieve those goals.

### 1.5.2 REPM Model

By keeping the idea of maturity models like CMM and ISO in mind, REPM was developed specifically for the area of requirements engineering. Requirements Engineering Process Model, REPM in short, is a model which is used to assess the maturity of requirements engineering process in software projects. The model was developed by Kaarina Tejle and Tony Gorschek for the work of their master thesis in 2002 [4].

### 1.5.3 The REPM-M Model

This thesis is a continuation of their work. In this thesis, REPM model is revised and mapped on market driven projects. Certain changes are incorporated in this model in order to make it more suitable and usable for market driven software projects. Enhancements which were made are

- SPAs, actions and their descriptions are reviewed and modified in order to make it more usable and understandable.
• Try to keep the SPAs and actions more general so that it covers broader requirements engineering techniques and methods.
• REPM-M model is specifically developed for market driven projects. So, the requirements engineering activities which is useful for market driven projects are incorporated in REPM-M model.
• Another column is added in the evaluation questionnaire while evaluating the projects. Currently questionnaire contains four columns i.e. Question, Yes/No, If No then Why. Now, another column will be added which ask if company performs a certain action then how it is performed. In this way, we would be able to know about current state of practice in the industry regarding requirements engineering. It would also be helpful in further updating the REPM-M model. (you will see the detail of these addition later in this document)

These changes will result requirements engineering process maturity model for market driven projects, REPM-M in short, version 0.1. Once these changes will be incorporated, then we will do the following steps to validate and evaluate the model.

1. Validate the model by taking an interview from a senior personal in the area of requirements engineering.
2. Evaluation of REPM-M model by using an example project.
3. Analysis and conclusions

1.6 Road Map

This document comprises of two main parts. First part contains different chapters discussing the whole work done in this thesis. Second part consists of Appendices containing updated version of REPM-M model along with interview questionnaire and evaluation questionnaire. A road map of the chapters of this thesis is given in the following section.

Part 1 -- Chapters

Chapter 1 Introduction – This chapter includes some basic information about software, Software Engineering, requirements engineering and issues related to requirements engineering. Later sections of this chapter discuss some maturity models and a brief introduction of REPM-M model.

Chapter 2 The REPM Model – In chapter 2, REPM model is described on which REPM-M model is based on. Basic purpose of this chapter is to give users a brief overview about REPM model, which will help them in understanding REPM-M model as well.

Chapter 3 The REPM-M Model – REPM-M model is discussed in this chapter. Its evolution from REPM to REPM-M model is also described. Different features of REPM-M model are presented by using different examples and diagrams. The later sections in this chapter include what approach we used in making REPM-M model.

Chapter 4 REPM-M Model Validation – This chapter includes why we validate REPM-M model and the technique that we used to validate the REPM-M model. The results of the validation can also be found in the same chapter.

Chapter 5 REPM-M Model Evaluation – REPM-M model was evaluated over an example project. This was another way of validation of REPM-M model. Chapter 5 contains the evaluation techniques and results.

Chapter 6 Discussions and Conclusion – Chapter 6 contains the discussions about what was observed during the whole thesis. Future works related to REPM-M can be found in this chapter. A conclusion can be found at the end of the chapter.
Part 2 – Appendix

Appendix I The REPM-M Model Manual – This appendix contains the manual for REPM-M model.

Appendix II The REPM-M Model – This appendix contains the REPM-M model including all the MPA, SPA and actions.

Appendix III Validation Questionnaire – This appendix contains all the questions that were used to validate the REPM-M model.

Appendix IV Evaluation Questionnaire – This section includes the questions that can be used to evaluate a software project for REPM-M model.
2 The REPM Model

This chapter briefly describes REPM model on which REPM-M model is based on. REPM model was developed in order to assess the maturity of requirements engineering process for software projects. Basic objective of this instrument is to provide a quick assessment tool which is easy to use and can provide enough information based on which a company can improve its requirements engineering process. Following sections will provide an overview of different ingredients of REPM model. It will help user in understanding how REPM model works.

2.1 Organization of REPM Model

REPM model comprises of different components which we will discuss in following sections to elaborate the characteristics of REPM model. The components and/or characteristics of REPM model that will be discussed are

- Structure and Notations
- Maturity Levels
- Relation
- Optional Groups and Actions
- Satisfied-Explained
- Enhancements in REPM model

2.1.1 Structure and Notations

REPM model comprises of three components i.e. Main process Areas (MPAs), Sub Process Areas (SPAs) and Actions. Main process area lies on the top of REPM model. Sub process areas may come either under main process areas or under other sub process areas. Actions may also come directly under main process areas or under sub process areas. The hierarchy of REPM model can be represented by following figure 2.1.

Following figure presents an example of how REPM model is structured. In above figure, white boxes represent SPAs while grey boxes represent actions. In this example, Main Process area (MPA), M, lies on top of REPM model. It may contain n number of SPAs and n number of actions. SPAs under MPA, M, are denoted as M.1, M.2… M.n which tell that this is an SPA under the MPA, M. Each action is denoted by the alphabet “a” in order to differentiate it with SPAs. An action which comes directly under an MPA is denoted as M.a1, M.a2, etc. Actions are usually reside under SPAs and denoted as M.1.a1, M.1.a2 and so on. SPAs can also come under other SPAs as shown in the above figure. Those SPAs are denoted as “M.1.1” which tells that this SPA lies under SPA “M.1”.

9
2.1.2 Main Process Areas (MPA)

Main process areas, MPA in short, lie at the top layer of REPM model. It represents main phases for requirements engineering process. In REPM model, requirements engineering process is categorized in three main process areas. The three main process areas are

- Requirements Elicitation
- Requirements Analysis and Negotiation
- Requirements Management

MPAs in REPM model are further categorized into sub process areas and actions. Each MPA comprises of different sub process areas and actions. In the following section, each MPA is discussed in detail.

**MPA 1: Requirements Elicitation (E)**

Requirements elicitation is first MPA in REPM model. It is also considered as first activity in requirements engineering process. Requirements are gathered from stakeholders in this phase. Requirements elicitation phase is denoted as "E" in REPM model.

**MPA 2: Requirements Analysis and Negotiation (A)**

Requirements analysis and negotiation is considered as the second phase of the requirements engineering process as well as it’s the 2nd MPA of REPM model. Goal of this phase is to remove ambiguous, incomplete and conflicting requirements. These types of requirements are also called bad requirements. Activity of negotiation is also carried along with the activity of requirements analysis. The goal of requirements negotiation is to extract the most important requirements among the other requirements. It will help when the time and cost is limited. Often, customers say that
each and every requirement is important. This is the responsibility of requirements analyst to negotiate with the customer in this regard.

**MPA 3: Requirements Management (M)**

In the MPA of requirements management, all the activities, other than Requirements elicitation and requirements analysis and negotiation, are included. Typically, this MPA comprises of three important activities of requirements engineering process i.e. Requirements documentation, requirements validation and requirements management (i.e. traceability, change management, release planning etc.).

2.1.2.1 **Sub Process Areas (SPA)**

Sub process areas are a group of related actions. The main purpose of SPA is to differentiate it with other actions so it would be quite easier for an evaluator or a user to evaluate, enhance and/or analyze requirements engineering process and results. In REPM, an SPA is denoted by a unique identifier e.g. E.1 which tells that this SPA lies in MPA E and is the first SPA.

2.1.2.2 **Actions**

An action represents an activity which is usually performed in order to carry out process of requirements engineering. As an example, ask executive stakeholders, in order to identify stakeholders, is an activity which is performed in the process of requirements engineering, so in REPM this activity is represented by an action. Each action lies on a certain REPM maturity level. That level suggests how mature the requirements engineering process is for a software project. We will discuss maturity levels in more detail in the following section. An action may be optional or correspond to an optional group. We will discuss optional actions and optional groups in more detail in the later sections.

An action is denoted by the letter “a”. As previously discussed, an action may lie directly under an MPA or under another SPA. For example, E.a1 tells that this action lies directly under MPA and it doesn’t link with any SPA while E.1.a1 tells that this action lies under SPA E.1.

As a note, while evaluating a project using REPM, one can avoid the divisions of actions under SPA. But division of MPA is important in order to analyze the results.

2.2 **REPM Maturity Levels**

Each action in REPM model has a certain maturity model. The main factors which are considered when setting an action on certain maturity level are cost and complexity. Cost denotes how much resources have to spend in order to perform an action. Complexity denotes how difficult it is to perform an action. Some other factors were also considered as well but on a less priority, for example, common sense, importance of an action for the process of RE or benefit. Cost and complexity for an action was the primary factor in order to keep the action on a certain level. Other factors were also considered for example, importance of that action for the process of requirements engineering, benefit of an action, or common sense. Tony [4] wrote in his thesis that only cost and complexity was considered in assigning actions to a particular REPM model. It is a bit hard to judge the maturity of a process just by calculating how much cost is spent in the process and how complex process is. Maturity can also be judged from other factors. For example, how intelligently a process is used so that with less cost, more benefits can be taken.
REPM maturity levels denote how mature and advance the process of requirements engineering is for a certain project. The higher the cost and/or complexity of the action, the higher its maturity level would be. There are five maturity levels in REPM model. The five levels are Wood, Bronze, Silver, Gold and Platinum. These levels help in evaluating requirements engineering process for software project.

The objective of a software project should not be always to get the highest maturity level. It may depend on a lot of factors e.g. how feasible it is to spend money or time in order to perform a certain action. It won’t be a wise decision to spend a big part of the whole project to certain activities in order to achieve a higher level. So, organization, handling the project, should efficiently decide what a level suits to a particular project.

2.2.1 Maturity Level 1: (Initial)

This level represents that company follows process of requirements engineering on an Ad-Hoc basis. Only basic activities are performed. Experience plays the key role behind the success of successful requirements engineering process. Usually project goes over budgeted and a lot of bad requirements occur in the later phases of software life cycle. Only the most important activities for example ask executive stakeholders for requirements origin identification, analysis through checklist and only documentation of accepted requirements are some activities which are carried out.

2.2.2 Maturity Level 2: (Basic)

A software project at this level depicts those basic activities of requirements engineering is performed. Market survey, analysis through checklist and requirements prioritization is some of the typical examples of this phase. Even though project fulfills basic activities of requirements engineering, there is still a big gap in terms of continuous improvement and measurements for requirements engineering process. Checklist for validating requirements is developed to find defects in requirements document. User manual draft is developed to facilitate the end users of the system. Requirements that are expected to be changed identified earlier in the development life cycle. Information is interchanged by using software applications.

2.2.3 Maturity Level 3: (Formulated)

A project on this level tells most of the activities are fulfilled and project is governed under experienced supervision. Processes are documented. Most of the activities are planned. A plan for detecting defects in captured requirements is prepared. Requirements are classified/categorized and risk assessment is carried out. Requirements are selected for current release. Suitable steps are carried out for resolving requirements overload problem. A standardize document structure is followed to document requirements. Requirements document is reviewed to find defects. Test cases are also proposed while specifying requirements. A change request mechanism is followed in order to gather changed requests from different sources. Requirements are handled through a database or any software application. User and system documentation is produced.

2.2.4 Maturity Level 4: (Developed)
A project on this level reflects that the process is planned and most of the activities are measured. Human and business factors are considered for requirements elicitation. Proper analysis and actions are taken on the basis of data collection. Cost/impact estimation is carried for release planning. A formal inspection is carried out to find defect in requirements document. A change management plan is followed about how to identify volatile requirements including defining traceability plans. Documentation is produced for management.

2.2.5 Maturity Level 5: (Advanced)

A project on maturity level 5 represents that company realize the importance of continuously improving process of requirements engineering. Eye is kept on future projects and mistakes done in previous projects are not repeated. Requirements reuse and post mortem meetings suggest the maturity of software project. Company is focused towards process improvement. Rejected and postponed requirements are also documented for future referencing. Requirements in graphical formats are translated into normal language.

2.3 Relation

Relation depicts the dependencies among different actions. Relation will help when a certain action is going to be changed then we should also look into the related actions and check whether a certain change affects the related actions. If yes, then one must take the necessary actions. There are three ways how relations between actions are defined.

1. A certain action can be a prerequisite of another action. So lets say, if an Action “X” is a prerequisite of Action “Y”, then Action should be an a lower level than Action “Y”.
2. One action may be helpful to execute another action, for example, reusable requirements can be used to specify scenarios or prototypes in order to further elicit or analyzing the requirements.
3. There may be some relation between two actions in general.

This property was present in a previous version of assessment model [TNY 2002] but removed in REPM.

2.4 Optional Group and Optional Actions

There are certain actions in REPM model which are optional which means that it is not necessary for a project to complete an action to reach a certain maturity level. The optional actions are denoted as “Opt” unless they don’t fall in an optional group.

Optional group comprises of more than one optional action. Actions in an optional group are denoted as OG1.01, OG1.02 and so on where OG1 refers to first optional group and 01 in OG1.01 tells that this is the first action of OG1. At least one action in optional group must be satisfied in order to achieve a certain maturity level.

2.5 Satisfied-Explained

There may be certain situations in which an action is not necessarily be performed. As an over simplified example of satisfied-explained, if a company is working on first project of his history, then it can’t reuse the knowledge from previous projects. As another example, for an in-house development project, there may not be necessarily
research for general stakeholders. An action is considered as Satisfied-Explained if a company thinks that this action should not be performed in order to successfully execute the process of requirements engineering process. The reason may not be for example lack of knowledge, or lack of enough resources. A thing to be noticed is there may be certain conditions in which action can be treated as Satisfied-Explained even though the reason is due to lack of resources. For example, if 50% of total cost is utilized in executing a particular action, then that action can be included in Satisfied-Explained list. If an action is included in Satisfied-Explained, then it is equivalent that the company is successfully performing that action.

A checklist is available to check whether a certain action can be considered as Satisfied-Explained or not. We left this thing up to the evaluator that how honestly he will evaluate requirements engineering process. As we will discuss later, a questionnaire is used to evaluate the maturity of RE process. The questionnaire contains a column in which evaluator will specify the reason of why the company is not executing a particular action for that particular project. That particular column will be used to decide whether this action will lie in the category of Satisfied-Explained or not.

2.6 Enhancements in REPM model

REPM model can be enhanced by adding more SPA and action in it according to preferences of the type of project and culture of the company. The reasons of adding further actions may be any one of them

- If one feels that an action can be split up to more than one action.
- If one feels that a certain action is important for the maturity if requirements engineering process and not included in this version of REPM model.

When adding SPA or action, following things must be taken into consideration

- Make sure that an already present MPA, SPA or action is not similar to one that is going to be added.
- Make sure that description of SPA or action is complete and adequate.
- One should decide on which level a certain SPA or action would reside.
- Name identifier policy must be followed.

While adding further SPA or action, one should follow the same rules as described in this manual. It is recommended that actions of higher maturity must be included in REPM model otherwise it will make the model complex and big. It would be time consuming to assess a software project by using a bigger REPM model.

2.7 Evaluation of Projects using REPM Model

Software projects can be evaluated using REPM model through a list of questionnaire. The list of questionnaire is included in Appendix III, which was used to evaluate REPM-M model (see chapter 3). There may be different purposes of evaluation of a software project. Primary reason is to verify the status of requirements engineering process for a software project. Companies may use REPM model in order to find defects or areas of improvement in their RE process. REPM model can be used as a framework in order to what things should be done during requirements engineering process. Even though in contrast to CMM which covers the maturity of whole organization, REPM model focuses more towards individual project. In order to verify the maturity of whole organization, it is recommended to evaluate all the projects under REPM model. It is not necessary that a software project must reach at a
maturity level 5. To reach a certain maturity level, resources are also required and it is not always the best thing. Company should execute their own cost benefit analysis before executing a certain action because one action, which is necessary for one project, might be irrelevant for another project. The primary goal of REPM model is to give developers and managers an easy and cheaper way to assess the requirements engineering process.

<table>
<thead>
<tr>
<th>E Requirements Elicitation</th>
<th>Action (UID)</th>
<th>Check</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Do you reuse requirements from other systems developed in the same application area?</td>
<td>E.a1</td>
<td></td>
</tr>
<tr>
<td>2. When determining whom the stakeholders are for a system, do you ask the people ordering the system, whom they think are the stakeholders?</td>
<td>E.1.a1</td>
<td></td>
</tr>
<tr>
<td>3. Do you conduct your own research determining who the stakeholders are?</td>
<td>E.1.a2</td>
<td></td>
</tr>
</tbody>
</table>

Figure 2.2: Extract from REPM evaluation questionnaire

Above figure is an extract from REPM evaluation questionnaire. First column comprises of a list of questions. Each question relates to a corresponding action. The answer of the question tells whether project satisfies the particular action or not. Second column contains action identifiers. It facilitates the evaluator to locate the description of particular action. Third column is for whether project fulfills certain activity or not. This column would have yes/no.

2.8 Analysis of Results

2.8.1 Technique for Interpretation of Results

By using the results of REPM model, maturity of requirements engineering process can be assessed. The results tell what has been done, what is not and what can be done in the future to improve the process of RE. If a software project fulfills all the actions under an MPA, ultimately it achieves maturity level 5. Since REPM comprises of three MPA, so there may some confusion while analyzing the results. For example, if the results of an evaluation of a project, say Project A, is

\[
\text{MATURITY Level of Requirements Elicitation} = 3 \\
\text{MATURITY level of Requirements Analysis and Negotiation} = 3 \\
\text{MATURITY level of Requirements Management} = 3 \\
\frac{(3+3+3)}{3} = 3
\]

Then, one can say that project A has successfully reached MATURITY level 3, but this may not be the case each time. As another example for project B

\[
\text{MATURITY Level of Requirements Elicitation} = 4 \\
\text{MATURITY level of Requirements Analysis and Negotiation} = 2 \\
\text{MATURITY level of Requirements Management} = 3 \\
\frac{(4+2+3)}{3} = 3
\]
Apparently, it seems that project B has reached the MATURITY level 3 but one can observe that MPA of requirements analysis and negotiation couldn’t reach MATURITY level 3 while requirements elicitation MPA has much better results than MATURITY level 3. Another example may be something like Project C,

\[
\begin{align*}
  \text{MATURITY Level of Requirements Elicitation} &= 4 \\
  \text{MATURITY level of Requirements Analysis and Negotiation} &= 3 \\
  \text{MATURITY level of Requirements Management} &= 3
\end{align*}
\]

\[
\frac{4+3+3}{3} = 3.33
\]

Above calculation concludes that project C has reached on a level 3.33, if we follow the same calculation rules as we did in the previous two examples. MATURITY level 3.33 are not defined anywhere in the REPM model and may result in some misleading results.

Unlike other maturity models like CMM and ISO9001, results of maturity level in REPM model are analyzed separately for each MPA. The maturity level for each MPA is calculated separately and their results are also interpreted separately. So, if a software project is on level 3 for MPA, Requirements Elicitation, on level 2 for requirements analysis and on level 1 for requirements management, it doesn’t mean that it is on REPM maturity level 2 i.e. the average of all the maturity levels. Separate analysis of each MPA will help company in getting exact knowledge about the strengths and weaknesses in the requirements engineering process.

2.8.2 Result Presentation

Results of REPM model can be presented through diagrams. A typical example of this presentation is given in the following figure. It’s quite easier for a person to analyze the strengths and weaknesses of the process of requirements engineering by using this figure. An important thing to note that to analyze the results of evaluation, three diagrams, one for each MPA, will be made in order to facilitate the task of analysis.

![Actions/REPM-M Level for MPA 'E'](image)

Fig 2.3: REPM Result Representation
Above figure illustrates a sample result for Requirements elicitation MPA during evaluation of a project. X-axis represents the maturity levels i.e. five in our case and y-axis represents no. of actions in the MPA i.e. requirements elicitation in this example. In REPM Level 1, total actions are three while completed actions are also three. In maturity level 2, total actions are four, while completed actions are three. But this project still reaches maturity level two because the incomplete action falls in the category of Satisfied-Explained. The area between the completed actions lines and satisfied explained line is termed as Model lag. The area between total actions and satisfied-explained line is termed as “Improvement Gap”. In the same way other two MPAs’ can be analyzed and presented.

Once an analyst gets used to with the technicalities and terminologies of above REPM model and presentation, it is quite easier for him to analyze the situation of requirements engineering process maturity within the company.

2.9 Conclusion

In this chapter, a brief overview of REPM model is given to the user. Goal of REPM model was to provide a quick way to assess the maturity of requirements engineering process for software projects. Tony Gorschek and Kaarina Tejle originally developed REPM model.

In the subsequent chapters, we will study how REPM-M model was evolved from REPM model.
3 THE REPM-M MODEL

This chapter describes how REPM-M model evolves from REPM model. Research methodology is discussed which was used to update REPM model. Factors which lead to transformation of REPM model into REPM-M model are also discussed.

3.1 Research Methodology

Two research techniques were used while studying REPM model so that it can be further improved. First technique was literature review. Literature related to requirements engineering, process models, maturity models, product/market driven development were studied. It helps in finding state of the art in the related topic. Second technique was unstructured interview from different researchers at BTH. These researchers are also working for some local companies of Sweden. These interviews were conducted in order to know opinion of different researchers about requirements engineering, specifically how they think about the two domains bespoke and market driven. During the interviews, our main focus was to find out what the current state of art in market driven requirements engineering and how much the proposed maturity model may contribute in the industry. The next sections will briefly describe the results of literature surveys and interview and how it affects in the development of REPM-M model.

3.2 The REPM-M Model

Requirements engineering process maturity model for market driven projects, REPM-M in short, was developed on the basis of REPM model originally developed by Tony and Kaarina while conducting their master thesis [4]. REPM-M model was developed to give the requirements engineers, developers, managers, etc. a way to assess their project’s requirements engineering process maturity. The projects specifically to market driven were considered. It is important to remember that REPM model, and of course REPM-M model, is used to assess the maturity of the individual project not for the whole organization unlike other famous maturity models like CMM [11], CMMI [17] and ISO [19]. So, the practices which assess the maturity for the whole organization are not included in REPM-M model.

During the literature review and unstructured interviews from researchers, it was observed that there are existing models for requirements engineering of market driven projects [21] [22] [23]. But most of them are customized by the companies for their own use. Some companies prefer time to market constraint while other prefers to produce high quality products [14]. So, literature lacks a generic process model which deals with market driven projects.

3.3 From REPM Model to REPM-M Model

In this thesis, REPM model version 1.0 was studied thoroughly and it is tried to find out the weaknesses in this model. Next section contains a discussion over those areas which were considered while developing the REPM-M model.

3.3.1 Main Process Areas

REPM 1.0 comprises of three MPAs i.e. Requirements elicitation, Requirements Analysis and Negotiation and Requirements management. It was noted that the third MPA is rather big MPA consists of several SPAs and actions. Requirements management MPA was comprises of three major activities of RE process i.e.
Requirements documentation, requirements validation and requirements management (Traceability, change management etc.). A thorough discussion was held whether to break this MPA into two or more. In the next section, some suggestions are discussed which were the result of discussions held. Each suggestion had its own benefits and drawbacks.

First suggestion, about breaking the third MPA into two or more, was to break the requirements management MPA into two different MPAs that are Requirements Document MPA and Requirements management MPA.

Third MPA could also be broken into two categories that are Pre Requirements management and Post Requirements management Process. Pre requirements management process includes the activities which were carried out initially before the phase of design, implementation and testing. On the other hand, post requirements management contains the RE activities which were carried out once the design and implementation phase has been started. The major drawback of this suggestion was there are certain activities which would be overlapped if the MPA is broken according to these criteria. For example, requirements negotiation can be included in pre requirements management process and post requirements management process.

In the end, it is finally decided to keep the third MPA as it is in order to keep the model a little simpler and relatively easier to analyze.

3.3.2 Specific Actions

Certain actions in REPM 1.0 were very specific so that it doesn’t cover a bigger domain of different requirements engineering techniques. For example, if a company uses any other activity instead of scenario elicitation then it doesn’t satisfy a certain level of REPM 1.0. This fact was also considered when changes were made in REPM. It is tried to make the SPAs and actions a little more general in order to cover broader range of requirements engineering techniques. This thing was also discussed by Tony and Kaarina [4] in their thesis.

3.3.3 Inclusion of Market Driven RE activities

In REPM version 1.0, most of the actions cover the area of bespoke projects. Most of the activities which are carried out in market driven projects were overlooked. To overcome the weaknesses of REPM model, domain of market driven projects was also considered and activities which are usually executed in requirements engineering process for market driven projects are included in REPM-M model. Inclusion of certain actions, related to the domain of market driven projects, grow the model. This continuous growth is directly proportional to the model Lag because larger model will decrease the applicability of model over certain projects [4].

3.3.4 Inclusion of Columns in Validation Questionnaire

Two more columns have been added into validation questionnaire. Fourth column in the questionnaire is used to check whether this action is eligible to include in satisfied-explain or not. This column will contain the answer why a particular action is not executed. Fifth column is optional. Evaluator can put any comments or notes for future guidance. In this column, for example, how a particular action was carried out can be written. It will help in creating a database of current state-of-art in the requirements engineering and different requirements engineers may get benefit from this database (see Section 6.7).

3.3.5 Result Presentation

The presentation of results after evaluation is done by using diagram. This presentation provides a quick method of analyzing the results of the questionnaire. In REPM model, it was proposed to use a single diagram to show the results. In REPM-M model, we proposed to use separate diagrams for each MPA. It helps in identifying the weak and strong area of process maturity more clearly.
3.3.6 Bespoke vs. Market Driven Projects Requirements Engineering

Market driven RE has some aspects different as compared to traditional requirements engineering for customer driven projects. Some of the factors are pressure of short time to market, incremental releases, almost error free release, more focus on requirements prioritization [12].

**Customer Involvement** – Since market driven product is developed for a mass market, there is not a discrete set of customers.

**Time to Market** – one of the primary goals of market driven project is time to market. If product is not launched at correct time, your competitor may win the race [14]. Usually, the time of a release for market driven project is fixed and low priority requirements are excluded in order to meet the release date. Short time to market is crucial in those circumstances where threat from a competitor is present.

**Requirements are Invented** – Because there is not a discrete set of customers or users for a market driven project, requirements are not elicited by using the traditional RE techniques. Before first release, market survey is the main weapon for gathering requirements. Usually, Development Company is responsible of inventing requirements [14]. Once first release is launched, the users and customers will post their feedback or in other words, new requirements.

**Requirements are rarely written** – because requirements are invented on run time, in market driven projects, requirements are rarely document [14]. Another reason of not documenting the requirements is because there is no necessity of a contractual agreement which is usually base on requirements specification document.

3.3.7 Challenges in Market Driven Requirements Engineering

Requirements in market driven projects always change due to various reasons. Reasons may be due to change in market, competitors improvements and customers are not certain of their requirements. Requirements are always volatile so it’s better to get earlier feedback from customers [13] [14]. So, a company should have methods to cope with these changes. Beta test releases can be one method of getting an early feedback from customers and can be used to manage changing requirements.

In MRE, there are two major sources of requirements. First are market and end users and second are developers. If focus would be more towards market, then chances of getting unrealistic will increase e.g. those requirements can not be produced in the available resources [13]. Also, there would be a lack of inventive requirements. If more requirements are taken from developers, then chances would be there that those requirements would not solve the customer requirements. The process of MRE should be such that it should keep a trade-off between these two sources. Create Technical Inventions (Sources in this case would be developing company, It is recommended when project is dealing with a stable market.). Satisfy Customer Needs (Sources in this case would be end users. It is recommended for an immature market.) [14]

Another issue which is common is the communication gap between developers and marketing staff [13]. The perspective of developers and marketing department, about what a good requirement, is usually different. This difference should be overcome by establishing an environment in which developers and marketing staff can easily communicate with each other. It will facilitate for producing a good quality end product.

Another dispute in an organization is usually about whether they should have an elaborate process or not. A defined process may limit the developers from their creativity and freedom. On the other hand, developers will know about their
responsibilities when using a defined process [13]. It is recommended that mature organizations have an elaborate process while an elementary process will be enough for immature and small organizations.

Traditional requirements specification vs. requirements management tool. When managing a steady stream of requirements, it is recommended to use a database other than traditional requirements specification document [14].

Requirements Overload – due to a lot of requirements suggestions, requirements database are usually overloaded, and thousands of requirements are usually in the queue for the next release [13]. Some companies try to overcome this problem by setting top 10 most important requirements. But this solution also has some risk. Because it’s not necessary that the selected requirements are the most important ones. One should have a method of avoiding this overloading of the requirements. An important suggestion of dealing this problem is always carefully deal with the feedbacks of customers and developers.

Small companies usually avoid using a database in order to specify requirements. It is always better to use a standard format while describing requirements. A checklist can be used for this purpose. The checklist will tell which items or attributes should be included in describing the requirements [14].

Requirements bundling can be referred as a process in which related requirements are implemented all together. This is usually done when interdependencies between requirements need to be resolved. Even though it is not a recommended way of dealing interdependencies among requirements, but in some cases, it’s sufficient enough to make decisions [13].

3.4 Conclusion

Main objective of REPM-M model is to provide a quick way to assess the maturity of requirements engineering process. Another objective of REPM-M model is to provide companies certain activities that they should carried out while introducing requirements engineering process within their company. REPM-M model covers the area of requirements engineering for market driven projects. Presentation of the results can be done in terms of tables or in the form of diagram. In this chapter, an example was given to show how one can present the results. These pictorial diagrams provide a quick way to analyze the results.

In the subsequent chapters, REPM-M model would be validated. First by interviewing a senior personal related to requirements engineering and specially having some experience in market driven projects. Second, an example is presented in chapter five which describes how the results of REPM-M model can be analyzed.
4 REPM-M Model Validation

REPM-M model was validated so that to overcome any hidden problem in the proposed model. This chapter will describe how REPM-M model was validated.

First section describes what method was chosen to validate the model. Second section of this chapter describes how the interviewee was chosen. Third section describes validation questionnaire which was used to interview the subject. Fourth section comprises of the results of the interview, contains the suggestions from the subjects and how those suggestions were treated.

4.1 Validation Method

REPM-model was based on literature reviews and a couple of unstructured interviews with researchers in the related domain. It was required to validate the model before it is used to evaluate projects. The validation is necessary in order to check

- Whether it covers all the activities necessary for the relevant domain?
- Is there any activity which is not that important?
- Is there any other anomaly in the model which was overlooked by the creators of the model?

In short, validation was required to find problems or defects in the REPM-M model. Two methods were used to validate the model. In the first method, which we will discuss in this chapter, was to use a structured interview technique from a senior personal in the area of requirements engineering for market driven project. Second method was by evaluating different industry projects by using proposed model. It is covered in more detail in the next chapter. A questionnaire was used in the interview to validate the REPM-M model. That questionnaire can be found in the Appendix II of this thesis. The REPM-M model manual and questionnaire was first sent to subject so that he can read it thoroughly and make his own notes. Later, he was interviewed by us. The interview was recorded through a recorder. In addition, separate notes were taken in order to further analyze the issues raised by the subject.

4.2 Choosing the Subject

The ideal person, who validate REPM-M model, should be a senior personal from Industry who has an experience of requirements engineering processes for around ten years and who had been engaged in the projects of over 100 person.

REPM-M model was validated by interviewing a senior personal from academia. Tony Gorschek, one of the persons who were involved in the formation of REPM-M model, was interviewed in order to review the REPM-M model.

The risk was involved in choosing Tony as our subject because he originally developed the model. But the risk was a little less this time since there wasn’t a major change in REPM-M model except on SPAs and actions level. An advantage of choosing him was he has experience of industry and academia. He is a PhD student at BTH and his area of research is requirements engineering for market driven projects. In addition, Tony is working with local company along with his research.
4.3 Validation Questionnaire

Validation questionnaire consists of six sections. Starting with a warm up section in which general questions were asked. Second section comprises the section about structure and notation of the REPM-M model. Third section includes the question regarding presentation of the results. Fourth and the main section of the questionnaire consist of five questions which were going to apply on each SPA and actions in REPM-M model. Fifth section includes the general questions about if anything missing in the model etc. Sixth section includes cool down section in which interviewee opinion about the model was asked.

4.4 Interview

Interview questionnaire along with REPM-M model and manual were sent to subject well in advance so that he can read the model thoroughly and prepare his notes. An initial version of REPM-M model was sent to subject i.e. version 0.1 which was then evolved to version 1.0 after validation. Interview was recorded on the tape so that if any thing missed during the interview can be rechecked through the recorder. Interview was held at subject’s office.

4.5 Interview Results

In the validation interview, subject pointed out some problems in the model which need to be improved. In the following section, each improvement suggestion was described and the corresponding action which was taken according is also described.

1. It was observed that language used in the model was a bit trivial. Several sentences include words like “Should” or “must”.

   This suggestion was well taken by keeping the factor who are the audience of the model i.e. who will use this model. Since this model is used by software organizations to assess the maturity of their requirements engineering process, so the target audience is project managers, requirements engineers etc. so, the model was restructured so that the language is soft and non-offensive for those personals. A typical example may be “Analytical Hierarchy Process (AHP) must be used to prioritize requirements within your organization”. This sentence can be replaced as “There are several techniques exist for prioritizing requirements e.g. Analytical Hierarchy Process (AHP) or Game planning”.

2. Different unplanned activities are also carried out during requirements engineering process which may lie on smaller levels e.g. level 1 or level 2. Those activities were overlooked in REPM model.

   One of the major changes which were done in the model was to make the action more generic on a rather abstract level so that it will be applicable for larger domain. By keeping this factor in mind, both ad-hoc and more structured techniques were merged in the model. By following the advice of the subject, certain ad-hoc activities are also included in the REPM-M model. For instance, under the SPA E.2 Requirements Capturing, a rather ad-hoc requirements engineering activity of idea generation is included.
3. **Maturity levels of the actions were revised. Certain actions were advised to include on a rather higher level based on cost and complexity.**

   As maturity levels of actions were one of the critical aspects, which decide on which level company is. The prior experience of the subject was used and welcomed in setting REPM-M levels. According to his experience in the relevant field, certain REPM-M levels were revised for example, action E.1.a2 Use of Identification Technique was set on level 2 in REPM-M version 01. But according to subject it should be on level 3 because it often costs more.

4. **Maturity levels of the actions were revised. Certain actions were advised to include on a rather higher level based on cost and complexity.**

   As maturity levels of actions were one of the critical aspects, which decide on which level company is. The prior experience of the subject was used and welcomed in setting REPM-M levels. According to his experience in the relevant field, certain REPM-M levels were revised for example, action E.1.a2 Use of Identification Technique was set on level 2 in REPM-M version 01. But according to subject it should be on level 3 because it often costs more.

5. **There are certain companies which don't specify requirements. So, those companies should reach at least maturity level 1.**

   Another action (M.2.1.a3) has been added in order to support those companies which follow rather ad-hoc process. Action is kept under REPM level 1 as it is rather cheaper to follow this action. It is presumed that almost all the companies fulfill this action. It is also concluded that if a certain company follows a higher technique then this action will be fallen into satisfied explain (see section 2.8).

6. **Validation questionnaire which was sent to subject contain the questions preceding by headings like “Warm Up” and “Cool Down”.

   Those headings have been omitted from the validation questionnaire so one can learn from this that those headings should not be specified in the questionnaire. So, questionnaire should start from a warm up session for example how is you and how have you been working in this subject.

7. **It was also advised by the subject that while taking interview for the model evaluation, you should be present in front of the interviewee.**

   Model evaluation which is the last part of this thesis is done by taking interviews from representatives of different companies working on a market driven project. It was asked from subject whether is it enough to send the evaluation questionnaire to the interviewee but it was recommended by the subject that you should be there in person. Reason being, there may be certain ambiguities in the questions which need to be clarified for example there are different terminologies being used for requirements inspection. So, it is better to be there in person.

8. **There are certain companies which don't follow any systematic way to prioritize their requirements but they are still running their business. There should be a room for such companies.**

   Above suggestion from subject was well taken and a new SPA (A.2.a3) has been added in order to support those companies which follow a rather ad-hoc process for prioritizing their requirements. There may be different reasons of not following a
systematic process of requirements prioritization for example it is quite expensive to prioritize a bunch of requirements for a small company.

4.6 Conclusion

In this chapter, REPM-M model is validated by a senior personal having vast experience in requirements engineering specifically market driven projects which make him an ideal person to validate this model. Validation of a model is always recommended in order to get a different perspective of a person based on his experience. In our case, Subject found various defects in the initial version of REPM-M model version 0.1. it could be nice if we would have validated this model from another person from the industry in order to get an industrial point of view about this model, but due to lack of time and resources, it wasn’t possible. On the basis of suggestions made by subject, REPM-M model has been further updated to version 0.2 which is used to evaluate different market-driven projects from industry (see chapter 4).
5 REPM-M PROJECT EVALUATION

In this chapter, second phase of REPM-M model validation is described. A project is evaluated by using the evaluation questionnaire for REPM-M model. On this basis of the results of that questionnaire, further analysis is presented and improvements suggestions are made.

5.1 Evaluation Method

An evaluation questionnaire for REPM-M model is used to evaluate projects. Evaluation questionnaire consists of set of questions. Each question represents an action in the REPM-M model. A detailed description about how one can evaluate a project can be found in the second chapter of this thesis while evaluation questionnaire can be found in Appendix III of this document. A necessary condition while choosing the project for evaluation was that it should be developed for market as REPM-M model was specifically designed for market driven projects.

To prove the applicability of REPM-M model, a project from industry has been evaluated using REPM-M model. REPM-M questionnaire was sent to project manager responsible of requirements engineering for the particular project. After the discussion, Project manager sends back the answers. In the later sections, those results would be analyzed so that requirements engineering maturity can be gauged.

5.2 Project Description

The project was related to embedded systems involving more than 100 persons involved. Project targets mass market and wasn’t intended to be developed as a bespoke project which is the mandatory condition to use this model.

5.3 Results Presentation in Tabular form

After evaluating the project following results were found.

<table>
<thead>
<tr>
<th>MPA</th>
<th>Total Actions</th>
<th>Completed Actions</th>
<th>Satisfied-Explained</th>
</tr>
</thead>
<tbody>
<tr>
<td>Elicitation</td>
<td>12</td>
<td>5</td>
<td>1</td>
</tr>
<tr>
<td>Analysis &amp; Negotiation</td>
<td>10</td>
<td>6</td>
<td>0</td>
</tr>
<tr>
<td>Management</td>
<td>33</td>
<td>16</td>
<td>4</td>
</tr>
</tbody>
</table>

Figure 4.1: Project Evaluation Results

5.4 Result Presentation in diagrams

A presentation of results can be viewed in the following graphs. Each main process area(MPA) has been illustrated in separate graphs.
Figure 5.1: Graph for Requirements Elicitation

Figure 5.2: Graph for Requirements Analysis and Negotiation
5.5 Analysis for MPA: Requirements Elicitation

MPA Requirements elicitation for REPM-M model comprises of 12 actions. While evaluating the project one of those actions falls into satisfied-explained which was E.3.a4 (performing a study about other business processes with whom the system will contact). As project is not directly for business use, it was an embedded system. As we can see in the table 4.1, that theoretically project doesn’t complete all the actions of REPM level 1, so it doesn’t fall on Level though but it completes some actions from maturity level 3 and 4. from the total of 12 actions, 5 of the actions are completed while one falls in satisfied-explained, so 50% of total action are dealt with. Thus, we can say that the project follows an average requirements elicitation process.

<table>
<thead>
<tr>
<th>Maturity Level</th>
<th>Total Actions</th>
<th>Completed Actions</th>
<th>Satisfied-Explained</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>3</td>
<td>2</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>3</td>
<td>4</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>4</td>
<td>3</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>5</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

Table 5.1: Evaluation Results for Requirements Elicitation

5.6 Analysis for MPA: Requirements Analysis and Negotiation

Ten actions have been included in the MPA requirements analysis and negotiation MPA of REPM-M mode. 6 of those actions are completed successfully in the project while none of them falls into satisfied-explained category. As 60% of the total actions are performed. One point, which is interesting for this MPA in the project, is, it performs 3 actions on level 3 out of 5 actions which tells us that they have a considerably good process of requirements analysis. Especially requirements prioritization is emphasized in order to have correct set of requirements for a particular release.
5.7 **Analysis for MPA: Requirements Management**

Requirements management MPA for REPM-M model comprises of 33 actions, the biggest MPA of the entire three MPAs of REPM-M model. 16 of those actions are completed successfully while 4 falls under the category of satisfied-explained. Therefore, we can say that 20 out of 33 actions are done in REPM-M model, which gives a percentage of 60.6. All the 3 actions from maturity level 1 are completed in the requirements engineering process. While, 4 out of 7 actions are performed for level 2. One of them is in satisfied-explained. A tally of 20 actions for this huge requirements management MPA can easily tells us that it is the most strong MPA for the project and they have a good requirements engineering process. Four actions which are treated as satisfied explained were:

1. Recording of requirements rational as it is “customer wants it” in this case.
2. Documentation of user manual draft as they produce technical document for their customers (embedded system)
3. Writing System models to describe requirements as they have more sophisticated system of writing requirements.
4. Deliver user documentation as they deliver technical documentation (embedded system)

5.8 **Improvement consequences**

In requirements elicitation phase for the project, two of the actions can improve their requirements engineering process considerably i.e. are if they ask executive stakeholder about the sources of requirements elicitation and if they carry a study of competitor analysis. In requirements analysis and negotiation phase for the project, again two actions may have a good impact on requirements analysis and negotiation process for the project. First, one is if
they use software to prioritize requirements. Second one is if the re-prioritize requirements because of new requirements.

In requirements management phase, most of the basic actions are performed successfully. There is some space of improvement if they follow a couple of actions of REPM level 2 for requirements management MPA. First action which is considered as an important one is to find requirements interdependencies in order to include all linked requirements in the same release. Second action, which may affect is if they identify volatile requirements. Identification of those volatile requirements would help them in order to select correct set of requirements for a particular release.

5.9 Conclusion

Evaluation of project explained in this chapter to give an idea about how requirements engineering process for projects, should be evaluated and how results should be analyzed using REPM-M model. A factor, which influenced on the results of REPM-M model, is Satisfied-Explained. Evaluator should be very careful while including an action into category of satisfied-explained.
6 DISCUSSIONS AND CONCLUSION

6.1 Discussion

Requirements are considered one of the problematic areas if it is not elicited and analyzed correctly. Requirements engineering provides techniques and practices which are used to elicit requirements correctly and how one can manage those requirements efficiently. In bespoke projects, requirements are captured from a specific set of stakeholders while in market driven projects, there is not a specific set of stakeholders [12][13]. The product is usually developed for a mass market [14] which is one factor, makes the process of requirements engineering a bit difficult [13], or different we should say. Other factors which differentiate requirements engineering for market driven projects and bespoke projects are time to market and high quality due to presence of competitive market, in the case of market driven projects [13][14].

REPM-M model proposed in this thesis provides a way to assess maturity of requirements engineering process for market driven projects. Even though there are certain models which are already present in literature which is used for assessing the maturity of requirements engineering process [1][16][17] but those models are quite large and requires a lot of resources to assess the maturity [18]. In addition, those models focus more towards a general requirements engineering for bespoke projects, overlooking important aspects of market driven requirements engineering e.g. release planning, or covering the requirements of mass market. REPM-M focuses on these two issues.

Creation of generic actions decreases the possibility of creating optional actions. The basic idea behind optional actions was for example, a company may use a specific technique to elicit requirements while another company may use another. Due to this fact, optional action was included in the REPM model [4] so that it covers both techniques without affecting the final result of the assessment. Due to the presence of more generic actions, the necessity of generic actions decreases automatically.

6.2 From REPM to REPM-M

As we have discussed before, REPM-M model was based on REPM model by keeping some considerations for example, more generic actions and market driven projects. Even though requirements engineering for market driven projects have some extra actions to fulfill the requirements e.g. release planning, but still actions are almost similar in number if we compare both models. The reason is actions in REPM-M model are more general which actually combines similar actions into one action. Most of the closely related actions from REPM model were combined in one action when included in REPM-M model.

6.3 Threats to Validity

There were a couple of threats which can be observed throughout to the validity of the REPM-M model.

REPM-M model was developed on the basis of REPM model [4]. REPM-M model was developed by existing literature review of requirements engineering specifically market driven and product development. In addition, some unstructured interviews were conducted from the researchers at BTH. So, it can be observed that there is more contribution of academia research in the formation of REPM-M model even though experienced personal from the industry could also be consulted but due to lack of
time, it was not possible. This threat was somehow overcome in the second phase of model validation in which projects from industry was evaluated using REPM-M model.

Another threat was about the first phase of the validation. In first phase of REPM-M model validation, the interviewee was one of the founders of the REPM model which was the basis of REPM-M model.

Third threat was in the second phase of the validation in which different projects from industry was evaluated by using REPM-M model. A questionnaire was sent to the senior person involved in a particular project to fill in the answers in terms of yes/no. It depends on the person, who filled the questionnaire, about how carefully and honestly he filled that questionnaire. Deviation from what was actually performed in the project may lead to wrong results. There is another possibility of misinterpretation from the evaluation questionnaire. Evaluator may infer differently as it was asked in the question.

6.4 Analysis of Results

After evaluating a project by using REPM-M model, its result can be analyzed so that weaknesses in the requirements engineering process can be pointed out. In the previous chapter, we have discussed how results can be analyzed. One important thing to be note here that it is not necessary to achieve the highest maturity level, because there might be certain projects whose budget won’t allow to carried out certain expensive requirements engineering activities. There must be a tradeoff between what activities must be carried out in order to get the highest benefit out of it. Managers must be careful while introducing a certain activity for a certain project. Another thing which is important to note here is there might be one action which can’t be executed due to various reasons, but the similar action may be very beneficial for another project.

6.5 Strengths of REPM-M Model

One of the strength of REPM-M model is that it covers domain of market driven projects. In literature requirements engineering for market driven projects are overlooked and conventional requirements engineering processes and techniques are also tried to be use in the market driven projects. But as it was discussed earlier, bespoke and market driven projects, have their own differences when talking about requirements engineering. This model provides a detailed description of the activities which should be carried out during RE of market driven projects.

Second strength is ability to add customized actions in the REPM-M model. Once an evaluator uses the questionnaire to evaluate the project, he can easily analyze what is missing in his process. In addition he can add what is missing in the REPM-M model. This updating in the model will help them in guiding for the future projects.

Dual benefit of using REPM-M model is obvious. First, it points out the weak area of requirements engineering process, second it provides a road map which can be used to start the process of requirements engineering.

Unlike other maturity models, REPM-M model focuses small to medium size software organizations. Even though, it may be used by a larger organization deals in large scale products, but larger organizations may require some sophisticated requirements engineering activities, for example, distributed negotiations, which were overlooked in current version of REPM-M.
6.6 Weaknesses of REPM-M Model

There are certain aspects in REPM-M model which still needs to be refined and future work can be done on this. First of all, some industry projects should be used to evaluate the REPM-M model in order to find holes in the model. Due to insufficient literature about requirements engineering for market driven projects, some important activities might have been overlooked in REPM-M model.

There is no hard and fast rule to include an action on a particular maturity level. Even though, three factors were considered while setting an action on a certain maturity level. The factors were Cost, Complexity and importance (see chapter 2). In almost all cases, level is set by using a common sense and gut feeling.

6.7 Future Works

- An automated tool can be developed so that companies can evaluate the maturity of their requirements engineering process. What companies have to do is to fill the automated evaluation questionnaire, and that tool can give presentations of the results through diagrams and in tabular form so that analyst can point out which parts of requirements engineering process need to be worked out.
- Model can be evaluated on different projects from industry with different sizes of teams. This evaluation helps in refining this tool so that it becomes more accurate and workable.

6.8 Conclusion

The main purpose of REPM-M model is to provide a quick tool to the industry for the assessment of requirements engineering process maturity for market driven software projects. REPM-M model hardly takes two to three hours in order to evaluate a project since it was simple to use and not so big in terms of actions. Another advantage of REPM-M model is companies can use this tool to introduce requirements engineering process in their projects. It provides a road map to those companies, which do not have formal and continuous requirements engineering process.
7 REFERENCES


APPENDICES

Appendix II - Requirements Engineering Process Maturity Model for Market Driven Projects
Appendix III - REPM-M Validation Questionnaire
Appendix IV - REPM-M Model Evaluation Questionnaire

Requirements Engineering Process Maturity Model for Market Driven Software Projects, REPM-M model in short, is developed to assess the maturity of requirements engineering process for market driven software projects. It is a light weight maturity model which will help software companies to verify the maturity of requirements engineering process in their software projects. Primary objective of REPM-M model is to provide an assessment instrument to the industry which quickly investigates the maturity of requirements engineering process for a particular project. It is always better to have something as compared to have nothing. To accomplish this objective, we have to compromise over certain things, for example, REPM-M model doesn’t cover each domain of software development otherwise a lot of more actions have to be included which make the model more complex. On an average, it takes two to three hours to complete the whole process of evaluation. REPM-M model is based on a previous model [4] titled as “REPM Model”.

This manual is written for those persons who will use REPM-M model to assess the maturity of requirements engineering process. The manual is divided into three sections. This manual can also be used as a road map for a company which wants to introduce a process of requirements engineering in a project. First section will discuss the structure of REPM-M model so that user of the REPM-M model will easily understand technicalities of REPM-M model. Second section will discuss how one can evaluate a software project by using REPM-M model. In third section, we will discuss how one can analyze the results of the evaluation in order to improve requirements engineering process.

Organization of REPM-M Model

REPM-M model comprises of different components which we will discuss in following sections to elaborate the characteristics of REPM-M model. The components and/or characteristics of REPM-M model that will be discussed are

1. Structure and Notations
2. Maturity Levels
3. Relation
4. Optional Groups and Actions
5. Satisfied-Explained
6. Enhancements in REPM-M model

Structure and Notations

REPM-M model comprises of three components i.e. Main process Areas (MPAs), Sub Process Areas (SPAs) and Actions. Main process area lies on the top of REPM-M model. Sub process areas may come either under main process areas or under other sub process areas. Actions may also come directly under main process areas or under sub process areas. The hierarchy of REPM-M model can be represented by following figure 1.
The above figure presents an example of how REPM-M model is structured. In above figure, white boxes represent SPAs while grey boxes represent actions. In this example, Main Process area (MPA), M, lies on top of REPM-M model. It may contain n number of SPAs and n number of actions. SPAs under MPA, M, are denoted as M.1, M.2…. M.n which tell that this is an SPA under the MPA, M. Each action is denoted by the alphabet “a” in order to differentiate it with SPAs. An action which comes directly under an MPA is denoted as M.a1, M.a2, etc. Actions are usually reside under SPAs and denoted as M.1.a1, M.1.a2 and so on. SPAs can also come under other SPAs as shown in the above figure. Those SPAs are denoted as “M.1.1” which tells that this SPA lies under SPA “M.1”.

Main Process Areas (MPA)

Main process areas, MPA in short, lie at the top layer of REPM-M model. It represents main phases for requirements engineering process. In REPM-M model, requirements engineering process is categorized in three main process areas. The three main process areas are

- Requirements Elicitation
- Requirements Analysis and Negotiation
- Requirements Management

MPAs in REPM-M model are further categorized into sub process areas and actions. Each MPA comprises of different sub process areas and actions. In the following section, each MPA is discussed in detail.

MPA 1: Requirements Elicitation (E)

Requirements elicitation is first MPA in REPM-M model. It is also considered as first activity in requirements engineering process. Requirements are gathered from
stakeholders in this phase. Requirements elicitation phase is denoted as "E" in REPM-M model.

**MPA 2: Requirements Analysis and Negotiation (A)**

Requirements analysis and negotiation is considered as the second phase of the requirements engineering process as well as it’s the 2nd MPA of REPM-M model. Goal of this phase is to remove ambiguous, incomplete and conflicting requirements. These types of requirements are also called bad requirements. Activity of negotiation is also carried along with the activity of requirements analysis. The goal of requirements negotiation is to extract the most important requirements among the other requirements. It will help when the time and cost is limited. Often, customers say that each and every requirement is important. This is the responsibility of requirements analyst to negotiate with the customer in this regard.

**MPA 3: Requirements Management (M)**

In the MPA of requirements management, all the activities, other than Requirements elicitation and requirements analysis and negotiation, are included. Typically, this MPA comprises of three important activities of requirements engineering process i.e. Requirements documentation, requirements validation and requirements management (i.e. traceability, change management, release planning etc.).

**Sub Process Areas (SPA)**

Sub process areas are a group of related actions. The main purpose of SPA is to differentiate it with other actions so it would be quite easier for an evaluator or a user to evaluate, enhance and/or analyze requirements engineering process and results. In REPM-M, an SPA is denoted by a unique identifier e.g. E.1 which tells that this SPA lies in MPA E and is the first SPA.

**Actions**

An action represents an activity which is usually performed in order to carry out process of requirements engineering. As an example, ask executive stakeholders, in order to identify stakeholders, is an activity which is performed in the process of requirements engineering, so in REPM-M this activity is represented by an action. Each action lies on a certain REPM-M maturity level. That level suggests how mature the requirements engineering process is for a software project. We will discuss maturity levels in more detail in the following section. An action may be optional or correspond to an optional group. We will discuss optional actions and optional groups in more detail in the later sections.

An action is denoted by the letter “a”. As previously discussed, an action may lie directly under an MPA or under another SPA. For example, E.a1 tells that this action lies directly under MPA and it doesn’t link with any SPA while E.1.a1 tells that this action lies under SPA E.1.

As a note, while evaluating a project using REPM-M, one can avoid the divisions of actions under SPA. But division of MPA is important in order to analyze the results.

**REPM-M Maturity Levels**

Each action in REPM-M model has a certain maturity model. The main factors which are considered when setting an action on certain maturity level are cost and
complexity. Cost denotes how much resources have to spend in order to perform an action. Complexity denotes how difficult it is to perform an action. Some other factors were also considered as well but on a less priority, for example, common sense, importance of an action for the process of RE or benefit.

REPM-M maturity levels denote how mature and advance the process of requirements engineering is for a certain project. The higher the cost and/or complexity of the action, the higher its maturity level would be. There are five maturity levels in REPM-M model. The five levels are Wood, Bronze, Silver, Gold and Platinum. These levels help in evaluating requirements engineering process for software project.

The objective of a software project should not be always to get the highest maturity level. It may depend on a lot of factors e.g. how feasible it is to spend money or time in order to perform a certain action. It won’t be a wise decision to spend a big part of the whole project to certain activities in order to achieve a higher level. So, organization, handling the project, should efficiently decide what a level suits to a particular project.

**Maturity Level 1: (Initial)**

This level represents that company follows process of requirements engineering on an Ad-Hoc basis. Only basic activities are performed. Experience plays the key role behind the success of successful requirements engineering process. Usually project goes over budgeted and a lot of bad requirements occur in the later phases of software life cycle. Only the most important activities for example ask executive stakeholders for requirements origin identification, analysis through checklist and only documentation of accepted requirements are some activities which are carried out.

**Maturity Level 2: (Basic)**

A software project at this level depicts those basic activities of requirements engineering is performed. Market survey, analysis through checklist and requirements prioritization is some of the typical examples of this phase. Even though project fulfills basic activities of requirements engineering, there is still a big gap in terms of continuous improvement and measurements for requirements engineering process. Checklist for validating requirements is developed to find defects in requirements document. User manual draft is developed to facilitate the end users of the system. Requirements that are expected to be changed identified earlier in the development life cycle. Information is interchanged by using software applications.

**Maturity Level 3: (Formulated)**

A project on this level tells most of the activities are fulfilled and project is governed under experienced supervision. Processes are documented. Most of the activities are planned. A plan for detecting defects in captured requirements is prepared. Requirements are classified/categorized and risk assessment is carried out. Requirements are selected for current release. Suitable steps are carried out for resolving requirements overload problem. A standardize document structure is followed to document requirements. Requirements document is reviewed to find defects. Test cases are also proposed while specifying requirements. A change request mechanism is followed in order to gather changed requests from different sources. Requirements are handled through a database or any software application. User and system documentation is produced.
Maturity Level 4: (Developed)

A project on this level reflects that the process is planned and most of the activities are measured. Human and business factors are considered for requirements elicitation. Proper analysis and actions are taken on the basis of data collection. Cost/impact estimation is carried for release planning. A formal inspection is carried out to find defect in requirements document. A change management plan is followed about how to identify volatile requirements including defining traceability plans. Documentation is produced for management.

Maturity Level 5: (Advanced)

A project on maturity level 5 represents that company realize the importance of continuously improving process of requirements engineering. Eye is kept on future projects and mistakes done in previous projects are not repeated. Requirements reuse and post mortem meetings suggest the maturity of software project. Company is focused towards process improvement. Rejected and postponed requirements are also documented for future referencing. Requirements in graphical formats are translated into normal language.

Relation

Relation depicts the dependencies among different actions. Relation will help when a certain action is going to be changed then we should also look into the related actions and check whether a certain change affects the related actions. If yes, then one must take the necessary actions. There are three ways how relations between actions are defined.

1. A certain action can be a prerequisite of another action. So lets say, if an Action “X” is a prerequisites of Action “Y”, then Action should be an a lower level than Action “Y”.
2. One action may be helpful to execute another action, for example, reusable requirements can be used to specify scenarios or prototypes in order to further elicit or analyzing the requirements.
3. There may be some relation between two actions in general.

This property was present in a previous version of assessment model [4] but removed in REPM-M.

Optional Group and Optional Actions

There are certain actions in REPM-M model which are optional which means that it is not necessary for a project to complete an action to reach a certain maturity level. The optional actions are denoted as “Opt” unless they don’t fall in an optional group.

Optional group comprises of more than one optional action. Actions in an optional group are denoted as OG1.01, OG1.02 and so on where OG1 refers to first optional group and 01 in OG1.01 tells that this is the first action of OG1. At least one action in optional group must be satisfied in order to achieve a certain maturity level.
Satisfied-Explained

There may be certain situations in which an action is not necessarily be performed. As an over simplified example of satisfied-explained, if a company is working on first project of its history, then it can’t reuse the knowledge from previous projects. As another example, for an in-house development project, there may not be necessarily research for general stakeholders. An action is considered as Satisfied-Explained if a company thinks that this action should not be performed in order to successfully execute the process of requirements engineering process. The reason may not be for example lack of knowledge, or lack of enough resources. A thing to be noticed is there may be certain conditions in which action can be treated as Satisfied-Explained even though the reason is due to lack of resources. For example, if 50% of total cost is utilized in executing a particular action, then that action can be included in Satisfied-Explained list. If an action is included in Satisfied-Explained, then it is equivalent that the company is successfully performing that action.

A checklist is available to check whether a certain action can be considered as Satisfied-Explained or not. We left this thing up to the evaluator that how honestly he will evaluate requirements engineering process. As we will discuss later, a questionnaire is used to evaluate the maturity of RE process. The questionnaire contains a column in which evaluator will specify the reason of why the company is not executing a particular action for that particular project. That particular column will be used to decide whether this action will lie in the category of Satisfied-Explained or not.

Enhancements in REPM-M model

REPM-M model can be enhanced by adding more SPA and action in it according to preferences of the type of project and culture of the company. The reasons of adding further actions may be any one of them

- If one feels that an action can be split up to more than one action.
- If one feels that a certain action is important for the maturity if requirements engineering process and not included in this version of REPM-M model.

When adding SPA or action, following things must be taken into consideration:

- Make sure that an already present MPA, SPA or action is not similar to one that is going to be added.
- Make sure that description of SPA or action is complete and adequate.
- One should decide on which level a certain SPA or action would reside.
- Name identifier policy must be followed.

While adding further SPA or action, one should follow the same rules as described in this manual. It is recommended that actions of higher maturity must be included in REPM-M model otherwise it will make the model complex and big. It would be time consuming to assess a software project by using a bigger REPM-M model.

Evaluation of Projects using REPM-M Model

Software projects can be evaluated using REPM-M model through a list of questionnaire. The list of questionnaire is included in Appendix III. There may be different purpose of evaluation of a software project. Primary reason is to verify the
status of requirements engineering process for a software project. Companies may use REPM-M model in order to find defects or areas of improvement in their RE process. REPM-M model can be used as a framework in order to what things should be done during requirements engineering process. Even though in contrast to CMM which covers the maturity of whole organization, REPM-M model focuses more towards individual project. In order to verify the maturity of whole organization, it is recommended to evaluate all the projects under REPM-M model. It is not necessary that a software project must reach at a maturity level 5. To reach a certain maturity level, resources are also required and it is not always the best thing. Company should execute their own cost benefit analysis before executing a certain action because one action which is necessary for one project might be irrelevant for another project. The primary goal of REPM-M model is to give developers and managers an easy and cheaper way to assess the requirements engineering process.

<table>
<thead>
<tr>
<th>E Requirements Elicitation</th>
<th>Action (UID)</th>
<th>Yes/NO</th>
<th>Comment if No</th>
<th>Comment if Yes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Do you reuse requirements from previous projects?</td>
<td>E.a1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Do you have any method to resolve requirements overload situation?</td>
<td>E.a2</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Do you classify requirements in order to differentiate different requirements?</td>
<td>E.a3</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Figure 2: Extract from REPM-M evaluation questionnaire

Above figure is an extract from REPM-M evaluation questionnaire. First column comprises of a list of questions. Each question relates to a corresponding action. The answer of the question tells whether project satisfies the particular action or not. Second column contains action identifiers. It facilitates the evaluator to locate the particular action. Fifth column in the questionnaire is used to check whether this action is eligible to include in satisfied-explain or not. This column will contain the answer why a particular action is not executed.

Analysis of Results

Once a project is evaluated by using the evaluation questionnaire, its result can be used to analyze the position of requirements engineering process. This analysis should be carried out in careful and intelligent manner in order to invest resources where it is required.

Technique for Interpretation of Results

By using the results of REPM-M model, maturity of requirements engineering process can be assessed. The results tell what has been done, what is not and what can be done in the future to improve the process of RE. If a software project fulfills all the actions under an MPA, ultimately it achieves maturity level 5. Since REPM-M comprises of three MPA, so there may some confusion while analyzing the results. For example, if the results of an evaluation of a project, say Project A, is

*MATURITY Level of Requirements Elicitation = 3*
MATURITY level of Requirements Analysis and Negotiation = 3  
MATURITY level of Requirements Management = 3  

(3+3+3)/3 = 3

Then, one can say that project A has successfully reached MATURITY level 3, but this may not be the case each time. As another example for project B

MATURITY Level of Requirements Elicitation = 4  
MATURITY level of Requirements Analysis and Negotiation = 2  
MATURITY level of Requirements Management = 3  

(4+2+3)/3 = 3

Apparently, it seems that project B has reached the MATURITY level 3 but one can observe that MPA of requirements analysis and negotiation couldn’t reach MATURITY level 3 while requirements elicitation MPA has mush better results than MATURITY level 3. Another example may be some thing like Project C,

MATURITY Level of Requirements Elicitation = 4  
MATURITY level of Requirements Analysis and Negotiation = 3  
MATURITY level of Requirements Management = 3  

(4+3+3)/3 = 3.33

Above calculation concludes that project C has reached on a level 3.33, if we follow the same calculation rules as we did in the previous two examples. MATURITY level 3.33 are not defined anywhere in the REPM-M model and may result in some misleading results.

Unlike other maturity models like CMM and ISO9001, results of maturity level in REPM-M model is analyzed separately for each MPA. The maturity level for each MPA is calculated separately and their results are also interpreted separately. So, if a software project is on level 3 for MPA, Requirements Elicitation, on level 2 for requirements analysis and on level 1 for requirements management, it doesn’t mean that it is on REPM-M maturity level 2 i.e. the average of all the maturity levels. Separate analysis of each MPA will help company in getting exact knowledge about the strengths and weaknesses in the requirements engineering process.

Result Presentation

Results of REPM-M model can be presented through diagrams. A typical example of this presentation is given in the following figure. It’s quite easier for a person to analyze the strengths and weaknesses of the process of requirements engineering by using this figure. An important thing to note that to analyze the results of evaluation, three diagrams, one for each MPA, will be made in order to facilitate the task of analysis.
Figure 3: REPM-M Result Representation

Above figure illustrates a sample result for Requirements elicitation MPA during evaluation of a project. X-axis represents the maturity levels i.e. five in our case and y-axis represents no. of actions in the MPA i.e. requirements elicitation in this example. In REPM-M Level 1, total actions are three while completed actions are also three. In maturity level 2, total actions are four, while completed actions are three. But this project still reaches maturity level two because the incomplete action falls in the category of Satisfied-Explained. The area between the completed actions lines and satisfied explained line is termed as Modal Lag. While area between total actions and satisfied-explained line is termed as “Improvement Gap”. In the same way other two MPAs can be analyzed and presented.

Once an analyst is used to of technicalities of above presentation, It is quite easier for him to analyze the situation of requirements engineering process maturity within his company.
References


APPENDIX II

REQUIREMENTS ENGINEERING PROCESS
MATURITY MODEL FOR MARKET DRIVEN PROJECTS

REPM-M Version 1.0

E Requirements Elicitation

The process of capturing requirements is called requirements elicitation. In bespoke projects, requirements are usually collected from stakeholders. A stakeholder may be any person who has a stake on the developing system. End users, developers, consultants, marketing personals are typical examples of stakeholders. In market driven projects, requirements are invented at developing company as well, in most of the cases by marketing personals and developers.

Actions

E.a1 Requirements Reuse  Level 5
When developing a new system, requirements can be reused from other systems already developed by the development company in the same application area when possible. One can save money and time by reusing requirements because reused requirements are already verified, designed, implemented and tested.

E.1 Requirements Origin Identification

Identification for the source of requirements is the key step to initiate the process of elicitation. There are different techniques that may be used in order to discover requirements from different sources.

Actions

E.1.a1 Ask Executive Stakeholders  Level 1
One can ask from executive stakeholder about who are the actual end users and other persons who have a stake on the developing system. In bespoke projects, executive stakeholder is one who is ordering the system. In market driven, there is not a specific set of executive stakeholders but in most of the cases marketing personals of the developing company act as executive stakeholder.

E.1.a2 Use of Identification Technique  Level 2
Several techniques, for instance market survey, can be used in order to locate the sources from where requirements can be captured.

E.2 Requirements Capturing

There are different techniques that may be used in order to discover requirements from different sources.

Actions
E.2.a1 **Competitor Analysis**  
Competitors’ products play an important role in capturing requirements in the area of market driven. Different requirements can be captured or influenced from competitors products already in market.

E.2.a2 **Planning for Requirements Capturing**  
A plan about how requirements will be captured facilitates the process of requirements elicitation. Different techniques can be used to collect requirements from a mass market. Focus groups, user surveys, interviews, conferences are some of those techniques. Requirements can also be collected by collecting the response of system users.

### E.2.1 Requirements Invention

Requirements invention within developing company is considered as one of factor which is advised. Especially, when a product is going to be launched for a stable and mature market.

E.2.1.a1 **Requirements Generation from Ideas**  
Requirements can be captured on the basis of the ideas launched by different people involved in software development.

E.2.1.a2 **Structured Requirements Invention**  
An environment which supports innovative ideas always leads to requirements invention. Regular internal meetings, Workshops and brainstorming are some of the techniques for requirements invention.

### E.3 Domain Knowledge

Domain knowledge is the general knowledge of all the different aspects and viewpoints of the system. This area is divided into several sub-areas depending on viewpoint.

**Actions**

E.3.a1 **Business Domain Consideration**  
Awareness of how system will make contribution to the organization, which will going to buy it, is called Business Domain Consideration. Those goals drive the process of requirements elicitation. These concerns are more general than the specific business case.

E.3.a2 **Human Domain Consideration**  
Human domain consideration is about knowing organizational and political factors which influence requirements sources. These factors may lead to gathering incomplete or conflicting requirements.

E.3.a3 **Technical Domain Consideration**  
Study of all the hardware and software, which have an impact on the developing system, is called technical domain consideration. This
includes other third party systems as well for example a database system.

E.3.a4 Operational Domain Consideration REPM 3
A computer based system usually supports other business processes. This business processes might be something like systems producing customer reports or technical activities such as navigating an aircraft. Study of those business processes strengthens the process of requirements elicitation.
A Requirements Analysis and Negotiation

Requirements analysis is the process of analyzing requirements in order to find conflicts, overlaps, omissions and inconsistencies. Requirements are analyzed by communicating with stakeholders and by arranging internal meetings with developers. For market driven projects, absence of stakeholders directly affect the process of requirements analysis. Especially for market driven projects, cost/time estimates are also usually made in this phase. These estimates help in making final decisions of either requirement will be implemented in the current release or not.

Once requirements are analyzed, requirements are also prioritized. The prioritization of requirements helps in collection of most important requirements. The activities carried out in this phase highly affect a later phase of requirements engineering process, for market driven projects, known as “release planning”

Actions

A.a1 Requirements Classification
Requirements are classified or grouped in order to find related requirements and for traceability. By using those groups, it is easier to see which group is affected due to a change in a requirement. Second, related requirements are important to make a decision for a particular release.

A.a2 Assessing Requirements Risk
A risk analysis is carried in order to find whether the elicited requirements are possible to implement and the problems that may arise in the implementation of those requirements. Assessing requirements risk also lead to find the completeness of the requirements.

A.1 Requirements Anomalies Detection
Requirements are reviewed in order to discover problems. Typical problems are incompleteness, conflicting requirements, un-testable etc. Several techniques are used to find those problems.

Actions

A.1.a1 Analysis through Checklist
A checklist is an easy to use, effective and economical in terms of finding out anomalies in requirements. The checklists are usually based on experiences of the company.

A.1.a2 Plan for Anomalies Detection
Incompleteness, conflicts, overlapping are major problems in requirements engineering which are identified early in the software development life cycle. A plan to detect anomalies from the elicited requirements makes some room for improvement in the overall process of requirements engineering. Several techniques are used to find those defects for example, Interaction matrices, brainstorming; regular meetings within teams and with stakeholders are some techniques that can be used to avoid problems.

A.2 Requirements Prioritization
Requirements prioritization helps in finding out the most important requirements from the customers. Each requirements is prioritized according to reflect the importance for stakeholders and overall success of the system.
Actions

A.2.a1 **Requirements Prioritization**  
REPM 3
It is always important to prioritize requirements especially in the case of market driven projects. High quality product in the given time and available resources highly depend on how efficiently requirements were chosen to implement. It is recommended to maintain a must and wish list of the requirements in order to facilitate release planning phase. Typical techniques for requirements prioritization are planning game and Analytical Hierarchy Process (AHP).

A.2.a2 **Provide Software for Prioritization**  
REPM 1
Providing support for electronic systems for example electronic mails and video conferencing facilitates the process of negotiation.

A.2.a3 **Ad-Hoc Prioritization**  
REPM 1
No systematic process is used to prioritize requirements. Requirements are usually prioritized on guesses.

A.2.1 **Requirements Re-Prioritization**

Certain factors, for example legal restrictions, change requirement priority over time. Re-prioritization is necessary for the success of the system.

A.2.1.a1 **Prioritization due to New Requirements**  
REPM 2
Requirement is re-prioritized whenever a new requirement occurs. New requirements may change the preferences of the customers or managers.

A.2.1.a2 **Prioritization due to Change**  
REPM 2
Requirement is prioritized when a change occurs. It is important because change may result in already developed prioritized list.

A.2.1.a3 **Prioritization due to New Release**  
REPM 3
Requirement is prioritized when a new release going to be launched.
M Requirements Management

Process of requirements management for requirements engineering can be seen as process of project management for software projects. Requirements management is the process consisting requirements documentation, requirements validation, release planning and change management.

M.a1  Post Mortem  REPM 5
Post mortem meetings are helpful in pointing out the mistake usually done in the projects. In market driven projects, a most mortem meeting after each release will affect the process of requirements engineering for the future releases. A typical output of post mortem is a documentation which drives the future ways.

M.a2  Requirements Overload Resolution  REPM 3
Requirements overload problem occurs when a lot of requirements are raised by system end users. It usually happens when a new release is launched and a lot of end users posted requirements. It is always good to have methods in order to resolve this problem.

M.a3  Requirements Metrics Collection  REPM 4
Different metrics can be collected during the process of requirements engineering in order to view the status of requirements engineering process. A Goal-Question Metrics approach is usually beneficial for measurements. Typical metrics may be no. of change requests, no. of inspections meetings, etc.

M.1  Release Planning

To better satisfy customer requirements and to achieve high quality in time, market driven products are usually delivered in releases. Proper planning is required about which requirements are kept in the current release in the available resources.

Actions

M.1.a1  Cost/Impact Estimation  REPM 4
Cost/impact is estimated in order to make a decision of which requirements will be included in the current release. Cost can be estimated in terms of how many day it would take to implement the requirement. Impact describes how many requirements will be affected by implementing this requirement.

M.1.a2  Requirements Selection  REPM 3
Most important or wanted requirements are advised to select to implement in current release. Requirements are usually selected from the prioritized list which was prepared in requirements analysis and negotiation phase.

M.1.a3  Finding requirements interdependencies  REPM 2
It is beneficial for release planning to find interdependencies among requirements. This will help in deciding that which requirements will be put in the current release. None of the interlinked requirements are forgotten to put in the same release.
M.1.a4 Release Scheduling  
A release schedule is an important aspect for market driven projects when a product is going to be launched for a mass market. Two factors on which release scheduling usually relies are time-to-market and quality.

M.2 Requirements Document
A requirements document is used to communicate system requirements to customers, system users, managers and system developers. Requirements document comprises of all the documentation produced during RE process.

Actions

M.2.a1 Record Requirements Rational  
The rationale is the basis for the requirement. Information about why requirement was specified in the first place and what function it has is specified. This specification is made at an early stage so that the initial rationale is documented.

M.2.a2 Rejected Requirements  
Requirements that are rejected are also documented. The reasons of rejection may be due to cost and time considerations. This documentation offer clarity as well as material for future releases.

M.2.a3 Postponed Requirements  
The requirements which are not selected for current release are documented. This documentation will help in the next release of the product and will act as references.

M.2.a4 Standardize Document Structure  
A standardize document structure are maintained for writing documents. A standardize structure helps all the readers and authors to follow a same strategy and lead to save time and efforts. Typical structure of a standardize document contains document summary, document usage description, business case and term definition.

M.2.1 Describing Requirements

It is easier for a reader when natural language used to describe requirements is concise, understandable and unambiguous. The requirements need to be written in a manner that will help all the readers to immediately understand the requirements meaning and placement.

M.2.1.a1 Selected Requirements Specification  
Selected requirements for current release are specified in unambiguous and clear natural language. In addition, each requirement is specified separately i.e. several requirements are not described in one text body.

M.2.1.a2 Requirements Description Template  
The use of a template to organize the description of requirements makes for a standardized specification. If a certain way of describing requirements (and what information is present) is used at all times the reader will be
familiar with the way the information is written and can more efficiently absorb the contents.

M.2.1.a3 **Ad-Hoc Requirements Specification** REPM 1
Requirements are not described and documented specifically in a certain format or by a certain process. Rather, requirements are directly delivered to design and/or development team in order to implement the change request.

**M.2.2 Requirements Validation**

Requirements validation concerns with checking the requirements for omissions, conflicts and ambiguities and for ensuring that requirements document follow quality standards.

M.2.2.a1 **Define Validation Checklist** REPM 2
A validation checklist consists of points thatvaluators is focused upon while validating the requirements document. This checklist provides a structured way to the validation process. This checklist also helps those persons who are not experienced in validation.

M.2.2.a2 **Requirements Review** REPM 3
Requirement reviews are conducted in peers (independent from the development in hand). Requirements reviews are organized at certain stages when a certain milestone is achieved. Requirements reviews are held to find the errors present in the document and to verify that certain standards are followed.

M.2.2.a3 **Requirements Inspection** REPM 4
Requirements inspection is a formal meeting which is arranged in order to find defects in requirements document. A group of people systematically checks the requirements, meets to discuss problems with the requirements and agree on how these problems are fixed.

M.2.2.a4 **Propose Requirements Test Cases** REPM 3
Proposing requirements test cases result in revealing requirements problems. These test cases also act as a basis of test planning. Proposing one test case for each requirement gives dual benefits i.e. validation of requirements and a test case.

M.2.2.a5 **Draft User Manual** REPM 2
A user manual draft explains the system facilities describe in requirements document. The target readers of user manual are end users. Writing a user manual during requirements validation phase forces a detailed analysis of the requirements document and help in revealing bad requirements.
M.2.2.a6 Paraphrasing System Model

A system model in a graphical notation is converted into natural language representation. Different stakeholders who are unable to recognize notations used in system models can easily understand natural language representation. Also, while explaining models into natural language, chances of finding errors, inconsistencies and incompleteness is increased.

M.2.3 System Modeling

System modeling acts as a supplement natural language description. A system model describes a particular aspect of a system. Those system models are used in the requirements document to add information about natural language descriptions. The key benefit is to help understand the requirements during writing detailed system specifications.

M.2.3.a1 Environmental model

Environmental model shows how your system will communicate with other automated systems which are interfaced to it and with other business processes.

M.2.3.a2 System Models

Different parts of the system are illustrated in system models. It helps reader in not only understanding particular parts of the system but also results in finding out problems like inconsistencies and omissions. Typical examples of system models are data flow diagrams, entity relationship diagrams, stimulus-response model and timing models.

M.2.3.a3 Architectural Models

An architectural model represents how the system is decomposed into sub-systems. In addition how the subsystems communicate with each other. An architectural model helps in partitioning the system requirements.

M.3 Change Management

Change management deals with the processes involved in managing changes in system requirements. It comprises the activities like how changes are formally proposed, analyzed and reviewed.

Actions

M.3.a1 Identify Volatile Requirements

The requirements that are likely to be changed during RE process or for the whole development process are volatile requirements. Identifying those requirements decrease the chances for future problems.

M.3.a2 Change Request Mechanism


Changes may occur during the implementation of the system or after a release. A proper change request mechanism can be used to gather those change requests. Typical techniques that are usually used are for example, a bug report button within application or web request form on the company’s or project’s web site which is accessible to all the effective persons.

M.3.a3 Change Management Plan  REPM 4
A plan for change management is all about how one deals with volatile requirements. Plan comprises of identification of changing requirements, controlling those changes, how to implement those changes and auditing those changes.

M.3.1 Requirements Traceability

Traceability information is the information which allows finding dependencies between requirements. These dependencies are known specially when dealing with the changing requirements. The relationships can also be defined among other parts of the system for example, design, components and documentation.

M.3.1.a1 Unique requirements Identifier  REPM 1
Each requirement is identified through a unique identifier. It helps in not only maintaining related requirements but also if you want to store the requirement in the database, it will act as a primary key.

M.3.1.a2 Defining Multidimensional Traceability  REPM 4
Mechanism for storing traceability information is followed which help in future dealing with the requirements change and controlling cost etc. Traceability information may be between requirements with its source, with other requirements and with other artifacts of the system.

M.4 CARE Tool usage

Computer Aided Requirements engineering tools (CARE) can be used to facilitate the process of requirements engineering. Typical examples of CARE tools are requirements database, graphics tools, communication programs etc.

Actions

M.4.a1 Information Interchange through CARE  REPM 2
The use of computer Aided tools in the RE process in order to exchange information saves time and money. Typical examples of those tools may be simple email programs, video conferencing etc.

M.4.a2 Information handling through CARE  REPM 3
Use of software application for handling requirements facilitates process of requirements engineering process. Storing requirements in database make the life easier when a change is requested.

M.5 Documentation Deliverables
During the process of software development, several documentations are prepared for the final delivery either to the customer or personals within the company. Typical examples of those deliverables may be end project plan, user manual or test plan.

Actions

M.5.a1 User Documentation REPM 3
This documentation comprises of end user manuals, user dictionaries, training manuals etc. target users of these documentations are end users of the system.

M.5.a2 System Documentation REPM 3
This group covers all system documentation from pre-study to complete system design with all pertaining documents e.g. Design documentation, technical specifications, use case diagrams and so on.

M.5.a3 Management Documentation REPM 4
This group contains all management documentation to handle finished system for all kind of upgrades or administrative actions, e.g. how to maintain the system, run diagnostics and optimize the system.
APPENDIX III

REPM-M VALIDATION QUESTIONNAIRE

This section consists of a list of questions, which is used to validate the REPM-M model version 0.1. These questions were asked in an interview by a senior personal in the area of requirements engineering for market driven projects.

Questions

1. Tell us about yourself i.e. your academic background, your achievements, interests and your experience both from industry and academic.
2. Why did you choose requirements engineering area and in particular market driven?
3. What do you think about importance of RE in market driven projects?
4. How long time did you take to study REPM-M Model?

Structure and Notations

5. Is structure of the manual understandable and meaningful?
6. Is the description of the MPA, SPA, and actions in the manual complete and adequate?
7. Do you understand what the idea behind different maturity levels is?

Result Presentation and Analysis

8. Is the way of presenting results adequate and helpful for analysts?

Validation for SPAs and Actions

For Each SPA and action all the three MPAs, interviewee will answer these three questions?

a. Whether SPA or action is applicable or relevant for market driven requirements engineering process?
b. Whether description of SPA or action is adequate, complete and relevant?
c. Do you think that a particular SPA or action is so specific and it must be generic or vice versa?
d. Is the action is on the right maturity level?
e. Is there any SPA or action missing in this MPA?

General Questions

9. Is there any thing missing as a whole in the model?
10. Is there any action, which is an overlap with any other action?
11. What are strengths of the model?
12. What is the most conflicted area in the model?
13. Are there any weaknesses in the model? If so please elaborate.
14. What do you think how much this model can contribute to the industry?
15. Would you like to give any additional comments?
APPENDIX IV
REPM-M MODEL EVALUATION
QUESTIONNAIRE
Name:  
Designation:  
Company:  
Project:  
No. of Persons Involved:  
Any other description if you want to give related to this project:  

General Questions (please answer next two questions once you finish with the remaining questionnaire).

Do you follow any RE activity which is not included in this questionnaire, if yes please specify?
Do you think that any activity is not necessary for RE process, if yes please specify?

Requirements Elicitation

<table>
<thead>
<tr>
<th>Question</th>
<th>Action UID</th>
<th>Yes/No</th>
</tr>
</thead>
<tbody>
<tr>
<td>Do you reuse requirements from previous projects?</td>
<td>E.a1</td>
<td></td>
</tr>
<tr>
<td>Do you ask executive stakeholders about sources from where requirements can be captured?</td>
<td>E.1.a1</td>
<td></td>
</tr>
<tr>
<td>Do you use any technique for identification of requirements sources?</td>
<td>E.1.a2</td>
<td></td>
</tr>
<tr>
<td>Do you plan any competitor analysis activity to capture requirements?</td>
<td>E.2.a1</td>
<td></td>
</tr>
<tr>
<td>Do you make and follow any requirements capturing plan?</td>
<td>E.2.a2</td>
<td></td>
</tr>
<tr>
<td>Do you invent requirements in-house?</td>
<td>E.2.a2</td>
<td></td>
</tr>
<tr>
<td>Do you generate requirements from ideas?</td>
<td>E.2.1.a1</td>
<td></td>
</tr>
<tr>
<td>Do you follow an structured way based on those ideas to create requirements?</td>
<td>E.2.1.a2</td>
<td></td>
</tr>
<tr>
<td>Do you perform a pre-study about how system is going to contribute to the organization?</td>
<td>E.3.a1</td>
<td></td>
</tr>
<tr>
<td>Do you perform a study about how human factors are going to affect the developing system?</td>
<td>E.3.a2</td>
<td></td>
</tr>
<tr>
<td>Do you perform a study about the operating environment of the developing system?</td>
<td>E.3.a3</td>
<td></td>
</tr>
<tr>
<td>Do you perform a study about other business processes with whom the system will contact?</td>
<td>E.3.a4</td>
<td></td>
</tr>
</tbody>
</table>

Requirements Analysis and Negotiation

<table>
<thead>
<tr>
<th>Question</th>
<th>Action UID</th>
<th>Yes/No</th>
</tr>
</thead>
<tbody>
<tr>
<td>Do you classify or group requirements to find related requirements or for traceability?</td>
<td>A.a1</td>
<td></td>
</tr>
<tr>
<td>do you conduct a risk analysis to find out possible problems in the elicited requirements?</td>
<td>A.a2</td>
<td></td>
</tr>
<tr>
<td>Do you use checklist for analysis to find problems in the requirements?</td>
<td>A.1.a1</td>
<td></td>
</tr>
<tr>
<td>Do you follow a plan to detect possible problems in the requirements?</td>
<td>A.1.a2</td>
<td></td>
</tr>
<tr>
<td>Do you follow any structured process to prioritize requirements?</td>
<td>A.2.a1</td>
<td></td>
</tr>
<tr>
<td>Do you use any software, which can help in prioritizing requirements?</td>
<td>A.2.a2</td>
<td></td>
</tr>
<tr>
<td>Do you prioritize requirements in any way?</td>
<td>A.2.a3</td>
<td></td>
</tr>
</tbody>
</table>
Do you re-prioritize requirements due to introduction of new requirements? A.3.a1
Do you re-prioritize requirements due to any change in requirements? A.3.a2
Do you re-prioritize requirements when working in new release? A.3.a3

Requirements Management

Do you conduct post mortem meetings when a release is finished? M.a1
Do you follow any specific method in order to resolve the problem for requirements overload? M.a2
do you collect metrics to keep an eye on the status of the project? M.a3
Do you conduct cost/impact estimation in order to include particular requirements in a release? M.1.a1
Do you follow a plan to select requirements for a specific release? M.1.a2
Do you find requirements interdependencies in order to verify which requirements are linked with each other M.1.a3
Do you prepare and follow a schedule for a particular release? M.1.a4
Do you record requirements rational? M.2.a1
Do you document rejected requirements? M.2.a2
Do you document postponed requirements for the next release? M.2.a3
Do you follow a standardize document structure for requirements specification? M.2.a4
Do you specify selected requirements of the current release? M.2.1.a1
Do you document description of the requirements document template? M.2.1.a2
do you specify requirements in any way? M.2.1.a3
Do you use a checklist for requirements validation? M.2.2.a1
Do you conduct review on requirements document? M.2.2.a2
Do you conduct requirements inspection to find problems in requirements document? M.2.2.a3
Do you propose requirements test cases? M.2.2.a4
Do you document user manual draft? M.2.2.a5
Do you paraphrase pictures used in the requirements document? M.2.2.a6
Do you write environmental models to describe requirements? M.2.3.a1
Do you write system models to describe requirements? M.2.3.a2
Do you write architecture models to describe requirements? M.2.3.a3
Do you identify volatile requirements? M.3.a1
Do you follow a change request mechanism? M.3.a2
Do you follow a change management plan? M.3.a3
Do you use a unique identifier to identify requirements uniquely? M.3.1.a1
Do you define multidimensional traceability? M.3.1.a2
Do you use computer-aided requirements engineering tools to interchange requirements? M.4.a1
Do you handle requirements by using computer-aided requirements engineering tools? M.4.a2
Do you deliver user documentations along with product? M.5.a1
Do you deliver system documentations along with product? M.5.a2
Do you deliver management documentation along with product? M.5.a3