Master Thesis
Software Engineering
Thesis no: MSE-2010-27
September 2010

The creation of Uni-REPM

A universal model for assessing requirements engineering process maturity

Thi Thanh Loan Nguyen

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

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

Contact Information:
Author(s):
Thi Thanh Loan Nguyen
E-mail: thanhloan2812.nguyen@gmail.com

University advisor(s):
Mikael Svahnberg
Department of Software Engineering and Computer Science

EMSE Co-supervisor(s):
Giancarlo Succi
Free University of Bolzano

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

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

ABSTRACT

Context. The empirical study of Somerville in 2005 has shown that potential business benefits could be achieved by assessing and improving the Requirement Engineering (RE) process. However, currently there has not been an adequate instrument for practitioners to perform this work. Most known process assessment models such as CMMI and ISO do not pay intensive attention on RE whereas tailored models such as GPG and REPM do not cover the mentioned area extensively.

Objectives. This thesis presents a conceptual research of RE process assessment in which the researcher aims to develop a practical model for evaluating the maturity of RE processes in industrial settings.

Methods. A major part of this study consists of a systematic review and a literature review to explore all RE "good practices" as a profound basis for the new model. Together with identifying potential activities, the two reviews also highly regarded the feasibility of certain practices and their context to assure the validity of the model.

Conclusions. Based on the results of the reviews, Uni-REPM is formulated and introduced. The objective of Uni-REPM is twofold. Firstly, it is expected to be applicable for assessing the maturity of RE processes in various scenarios where an organization would use different development approaches. Secondly, it instructs practitioners with what to perform in RE processes and what they would benefit from such activities. As an assessment instrument, Uni-REPM serves a simple and low cost solution for practitioners to identify the status of their RE process. As a guidance tool, Uni-REPM is believed to lessen the gap between theoretical and practical worlds by transferring the available RE technologies from research to real work.

Keywords: Requirements Engineering, bespoke, market-driven, RE process assessment.

ACKNOWLEDGEMENT

I would like to use this particular page to express my sincere gratitude to my professors Dr. Tony Gorschek and Dr. Mikael Svahnberg who have provided me with valuable advice and support. It was a great pleasure for me to work under their supervisions during which I have learnt a lot from them about the research world. I am also thankful to my co-supervisor, Professor Giancarlo Succi who has given me many appreciated suggestions on my systematic review. In particular, I would like to thank my colleague, Nguyen Thi Phuong Mai, who has been beside supporting and discussing about problems with me during the whole thesis work. Last but not least, I am deeply grateful to my family and my friends for always supporting me to finish this thesis.

CONTENTS

1 INTRODUCTION 1

1.1 AIMS AND OBJECTIVES 2
1.2 RESEARCH QUESTIONS 2
1.3 EXPECTED OUTCOMES 2
1.4 RESEARCH METHODOLOGY 3
1.4.1 Systematic Review 3
1.4.2 Literature Review 4
1.4.3 Technology transfer 4
1.5 STRUCTURE OF THE THESIS 6

2 BACKGROUND AND RELATED WORK 8

2.1 BACKGROUND 8
2.1.1 Software process maturity assessment 8
2.1.2 Requirements Engineering 9
2.2 RELATED WORK 11

3 SYSTEMATIC REVIEW 13

3.1 NECESSITY OF A SYSTEMATIC REVIEW 13
3.2 SYSTEMATIC REVIEW DESIGN 13
3.2.1 Define the Research Questions 14
3.2.2 Development of a Review Protocol 15
3.2.3 Evaluate the Review Protocol 22
3.3 SYSTEMATIC REVIEW EXECUTION 22
3.3.1 Search result 22
3.3.2 Selection result 23
3.4 SYSTEMATIC REVIEW RESULT 24
3.4.1 General characteristics of the primary studies 24
3.4.2 Answers for the stipulated research questions 27
3.4.3 Mapping the indentified practices to the acknowledged challenges in MDRE 36
3.5 CONCLUSION 39

4 LITERATURE REVIEW 41

4.1 THE NEED OF LITERATURE REVIEW 41
4.2 LITERATURE REVIEW RESULTS 41
4.2.1 Additional actions 41
4.2.2 Updated actions 45
4.3 SUMMARY 45

5 UNI-REPM MODEL CONSTRUCTION 46

5.1 CONSTRUCTION METHODOLOGY AND OBJECTIVES 46
5.2 MATURITY MEASUREMENT 47
5.3 MODEL STRUCTURE 48
5.3.1 Main Process Area (MPA) 48
5.3.2 Sub-Process Area (SPA) 48
5.3.3 Action 48
5.4 MODEL CONTENT 49
5.4.1 Determine the MPAs of the model 49
5.4.2 Determine the Actions of the model 50
5.4.3 Overlapping actions between Bespoke and Market-driven RE 51
5.4.4 Specify levels for Actions in the model 52
5.4.5 Gather closely related Actions into SPA(s) 52
5.4.6 Model checklist 53
5.5 MODEL USAGE 54
5.5.1 Who will directly use the model? 54
5.5.2 How to use the model? 54
5.5.3 How to read the result? 54

5.6 UNI-REPM PROCESS AREA VIEW 56
5.7 CONCLUSION 60

6 THREATS TO VALIDITY 61

6.1 THREATS TO VALIDITY IN STUDY METHODOLOGIES 61
6.2 PUBLICATION BIAS 61
6.3 THREATS TO DATA SELECTION AND EXTRACTION CONSISTENCY 61

7 CONCLUSION 62

7.1 RESEARCH QUESTIONS AND FINDINGS 62
7.1.1 Research question 1: What are good practices for MDRE? 62
7.1.2 Research question 2: What are good practices for bespoke RE? 62
7.1.3 Research question 3: Based on the answers for RQ1 and RQ2, what major practices could be used to assess the maturity of the RE process? 62
7.2 CONTRIBUTION SUMMARY 63
7.3 FUTURE WORK 63

REFERENCES 64

APPENDIX A: LIST OF IDENTIFIED PRACTICES 67

APPENDIX B: LIST OF IDENTIFIED MODELS AND FRAMEWORKS 71

APPENDIX C: LIST OF IDENTIFIED TECHNIQUES AND METHODS 72

APPENDIX D: LIST OF IDENTIFIED TOOLS 75

REFERENCES INCLUDED IN THE SYSTEMATIC REVIEW 76

APPENDIX E: LIST OF GOOD PRACTICES 79

APPENDIX F: RESULTS OF COMBINATION ANALYSIS 83

APPENDIX G: UNI-REPM DESCRIPTION 90

APPENDIX H: UNI-REPM LEVEL VIEW 113

APPENDIX I: UNI-REPM CHECKLIST 117

List of tables

TABLE I. Research questions 2
TABLE II. Research methodologies 3
TABLE III. Research questions RQ.1 breakdown 14
TABLE IV. Search strings 16
TABLE V. Quality Assessment criteria 21
TABLE VI. Data extraction form 1 21
TABLE VII. Data extraction form 2 21
TABLE VIII. Primary search results 22
TABLE IX. Context distribution of primary studies 26
TABLE X. Quality assessment results 27
TABLE XI. Practice distribution according to activity areas 28
TABLE XII. Most proposed practices in each activity area 28
TABLE XIII. Identified supports 29
TABLE XIV. Generalized practices 30
TABLE XV. Additional actions 42
TABLE XVI. Updated actions in REPM 45
TABLE XVII. SPAs list according to MPAs 53
TABLE XVIII. Assessment result in MPA "Organizational Support" 55
TABLE XIX. Uni-REPM Process Area view 57

List of figures

Figure 1. Technology transfer framework [25] 5
Figure 2. Research methodologies overview 5
Figure 3. Structure of thesis 6
Figure 4. Systematic Review Process 14
Figure 5. Relation among research questions 15
Figure 6. Study selection process 19
Figure 7. Primary study section execution 24
Figure 8. Distribution of the publications according to the publication years 25
Figure 9. Research methodology distribution of primary studies 26
Figure 10. Evaluation process to detect "good practices" 34
Figure 11. Distribution of practices according to validation context 36
Figure 12. Uni-REPM creation process 46
Figure 13. Uni-REPM model structure 48
Figure 14. Practices combination process 50
Figure 15. Overlapping actions between Bespoke and market-driven RE 52
Figure 16. Graphical presentation of assessment result 55

1 INTRODUCTION

The importance of software in our daily life has been increasingly confirmed recently. Together with the increase of software development, software industry started facing various challenges. This has led to a trend investing large efforts in improving the quality of a software process. A number of process improvement frameworks have been introduced for this purpose such as CMM, CMMI, ISO, and so on. The main idea of these frameworks is to assess current state of processes in the organizations, detect existing problems based on which provide an improvement path in order to fulfill missing gaps and advance the processes. However, a question has been triggered: Why uncover problems that late? It has been shown that significant business benefits could be achieved by preventing problems earlier at Requirements Engineering (RE) phase, instead of waiting until the project finished [1]. The rationale is that RE has been acknowledged to be the critical determinant of the software quality and software development process effectiveness [2]. Problems in this area *remain* a profound effect on project cost and system functionality. Hall et al. has reported that a large proportion of the development problems are from requirements (48%) [3]. In addition, fixing requirement related problems consumes a high cost of rework in later states [4] [5]. In Leffingwell's research, it is shown that this effort could be from 70 percent to 80 percent of the project's rework cost [4]. Boehm has reported that requirements defects if remained would cost from five to ten times more to repair during the coding phase and from 100 to 200 times more during the "maintenance" phase [5]. Clearly, significant improvement on product quality will be perceived by advancing the RE process.

Despite its important role, industrial projects still indicate poor practices in RE process [3][6][7][8][9]. As in the survey of Juristo [6], RE process in organizations is often not well-defined and lacks guidelines for using tools, methods and proper specification documents [6]. The user involvement is low in the process and traceability is usually overlooked [6]. Another study which performed survey on 150 organizations in the US showed that almost none of the modeling techniques was used in their RE process [10]. In market-driven RE where the products are built for mass market, the vast number of stakeholders makes it difficult to elicit and to manage the requirements since they usually provide a large and continuously growing number of requirements with different levels of abstractions [11][12]. Moreover, those requirements are often volatile and changed [9]. Besides, the competition in the market is also a characteristic that impacts the RE process. For instance, it's usually very challenging to make the trade-off between market-pull and technology-push in order to have an adequate set of requirements for each release [11].

In an attempt to help practitioners to early identify problems and improve RE process, this study has been performed to provide a particular instrument for assessing RE process maturity as the first step in the improvement path [13, 14] as well as to bring practitioners a more concrete and complete view of current RE knowledge. The study aims to extract practical knowledge from literature to develop the so-called Uni-REPM model. The function of the model is twofold. Firstly, it is designed to be a universal instrument for evaluating the maturity of the RE Process. Secondly, it is a guideline showing implementation steps for practitioners in such process. Uni-REPM is conducted based on the studies of "good practices" which include an extensive systematic review on market-driven RE and an intensive literature review on bespoke RE.

"Good practices" in this thesis mainly imply activities in RE which could benefit the practitioners by implementing them in industrial processes. The "benefit" here refers to schedule and effort impact.

1.1 Aims and Objectives

The aim of this study is to identify the recommended "good practices" in RE literature based on which to develop a new model for assessing organizational RE process maturity. This aim will be achieved by addressing the following objectives:

- Identify "good practices" in RE process for bespoke development.
- Identify "good practices" in RE process for market-driven development.
- Formulate the obtained "good practices" and develop an RE process maturity assessment model.

1.2 Research questions

Table I gives an overview of the research questions of this study with respect to the goals they cover.

TABLE I. Research questions

Research Question	Goal
RQ.1. What are "good practices" for market-driven RE process?	To identify practices which are recommended as beneficial actions for RE process in market-driven development.
RQ.2. What are "good practices" for bespoke RE process?	To identify practices which are recommended as beneficial actions for RE process in bespoke development.
RQ.3. Based on the achieved answers, what major practices could be used to assess the maturity of RE process?	To analyze and develop a practical model for assessing the maturity of RE processes based on the previous answers.

1.3 Expected outcomes

The expected outcomes of this report contain the following products:

- **EO1**: A synthesis of the conducted systematic review which gives an overview of "good practices" in RE literature for market-driven development.
- **EO2**: A summary of "good practices" achieved from the literature review on REPM, CMMI and ISO for RE process in bespoke development.
- **EO3**: A description of a maturity assessment model which will be constructed based on the previous study results. The model is expected to function as an evaluation tool to measure the RE processes maturity as well as a guide tool for practitioners.

1.4 Research methodology

Table II shows the research methodologies used in this study in relation with the stipulated research questions.

TABLE II. Research methodologies

Research question	Research Methodology	Expected Outcome
RQ1. What are "good practices" for market-driven RE process?	Systematic Review	EO1
RQ2. What are "good practices" for bespoke RE process?	Literature Review	EO2
RQ3. Based on the achieved answers, what major "good practices" could be used to assess the maturity of the RE process?	Technology Transfer	E03

1.4.1 Systematic Review

Systematic Literature Review (SLR) also known as Systematic Review (SR) is "a means of identifying, evaluating and interpreting all available research relevant to a particular research question, topic area, or phenomenon of interest" [15]. Kitchenham mentioned that it is a form of secondary study gathering information from included individual studies which are called primary studies [15]. The procedure of Systematic Review consists of three main steps: planning the review, conducting the review and reporting the review [15]. In the planning step, a review protocol covering the purpose and procedure of the systematic review is developed. In the conducting step, the primary studies are selected from the retrieved data using defined search strategies and the extracted data is synthesized. Lastly, the findings are present in reporting step.

In this thesis, the researcher chose to adopt the Systematic Review described by Kitchenham due to several reasons. Firstly, the main purpose of the methodology itself conforms to the goal of this study. Since market-driven RE (MDRE) has been gaining increased interest in software development community [16][9] and the research studies are scattered around various sources [16], a systematic review is a fair and thorough means to find solution to RQ.1, in compare to the traditional literature review [15]. The first reason is that Systematic Review gives advantage of indentifying all the suggested MDRE practices as well as evidences on practices' evaluation across a wide range of publications. Secondly, incorporating this methodology in the thesis helps to minimize the researchers' bias by addressing clearly formulated research questions, systematically collecting and identifying relevant studies, and critically evaluating the obtained studies [15]. Hence, it provides a profound basis for understanding current state of MDRE practices in research. Thirdly, considering MDRE is still a fairly new area it is not advantageous to use snow ball technique in this study since it is hard to identify the key papers in the area. Therefore it is highly possible to end up with a huge ball. Besides, due to the poor practices reported from industries [8, 9], survey will not provide the right/adequate answer for the study.

The systematic review in this study focused on two evaluation aspects of the identified market-driven RE practices which are: 1) Empirical Support and 2) Rationale. Empirical support in this study refers to validation results reported from applying the indentified practices in case studies, sample projects and in industries. Rationale in this study is mainly indicated by motivation (e.g. empirical data support for the need of such practices) and supports. These two aspects were considered as driven criteria throughout the whole systematic review plan and execution. An analysis

on the obtained practices regarding these two criteria will be performed to detect a set of "good practices" which is expected to aid the researcher in constructing the assessment model in later stage. In order to better obtain the answer, the research question RQ.1 is broken down into three sub-questions in the systematic review. The detailed design and findings obtained from this study are formulated and presented in chapter 3.

It is also necessary to mention that in this study, the researcher received support from an independent colleague in the systematic review process. The support was mainly in primary study selection and data extraction steps.

1.4.2 Literature Review

Beside the systematic review on market-driven RE practices, a traditional literature review was conducted to obtain a more sufficient and up-to-date knowledge on RE activities in traditional development. The reason for performing a literature review is that traditional RE had emerged and been discussed for many years. "Good practices" in this area have been introduced in many books [17] [18] [19] [20] and assessment in these processes has been supported by a significant number of methods [21] [22] [23] [24]. Hence, extracting body of knowledge in bespoke RE from renowned and up-to-date sources would be sufficient to a degree considering the schedule of the thesis.

The literature in this study is mainly conducted based on the original assessment model REPM [23] and the two most renowned models CMMI version 1.2 [21] and Tickit [22]. The rationale for choosing those sources is twofold. Firstly, the collection of RE good practices for bespoke RE in the grounded model, REPM, was already validated in the industry as useful and usable [23]. Secondly, the consideration of other creditable sources (CMMI and ISO 9000) ensures the latest good practices are covered. The result of this literature review is presented in Chapter 4.

1.4.3 Technology transfer

The approach to identify the answer for RQ3 follows loosely the technology transfer framework created by Gorschek [25]. In the mentioned framework, the process starts by identifying the potential problems in real work following with problem formulation. Thereafter, a study on state-of-the-art needs to be performed and the candidate solution is created based on the study of state of the art as shown in Figure 1. Therefore, the RE assessment model of this study is constructed using the findings from previous reviews on bespoke and market-driven developments (RQ1 and RQ2). The RE "good practices" are then analyzed in order to build the content of the new model.

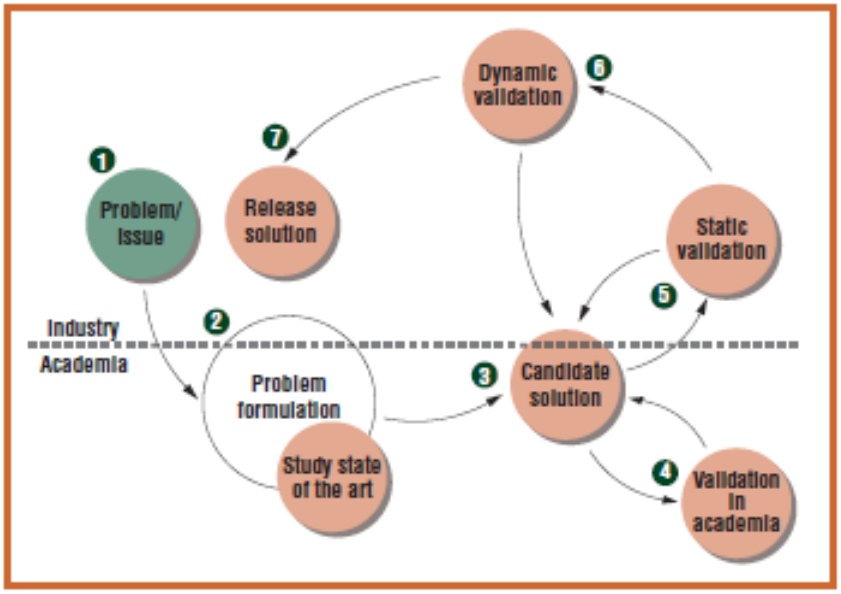

Figure 1. Technology transfer framework [25]

Figure 2 depicts an overview of the whole process in relation with the applied research methodologies in this study. The two sets of "good practices" extracted from the answers of research questions RQ1 and RQ2 will serve as input for RQ3 to create the Uni-REPM model.

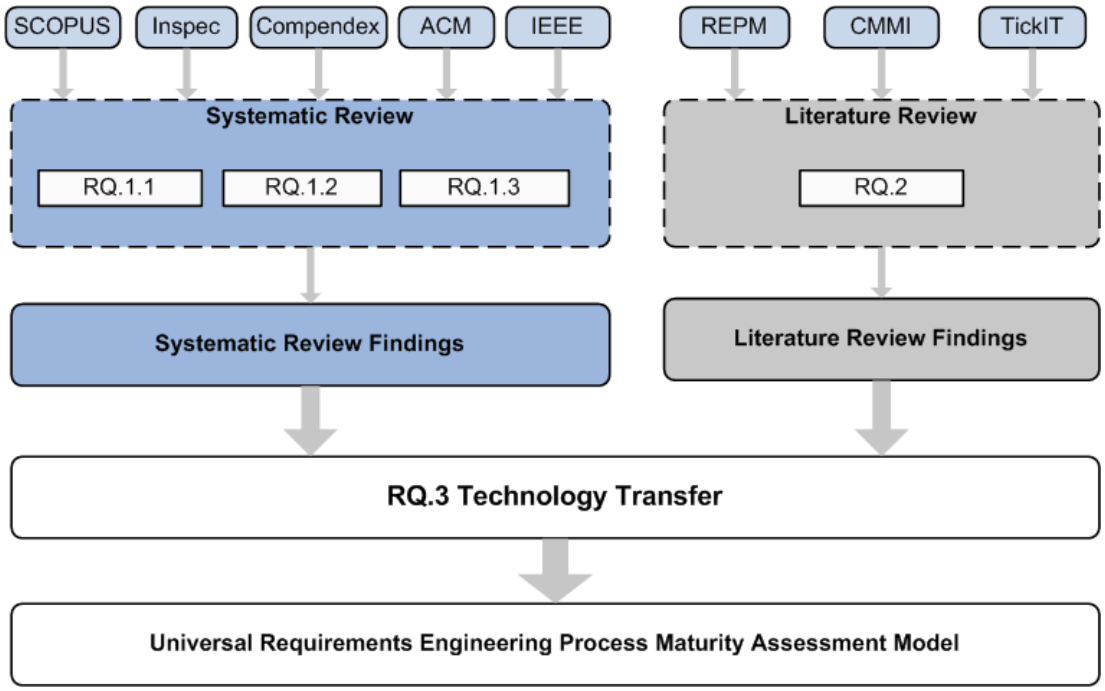

Figure 2. Research methodologies overview

1.5 Structure of the thesis

The content of this thesis is divided into two main parts: Content and Appendices. In the *Content* part, there are 7 chapters as depicted in Figure 3.

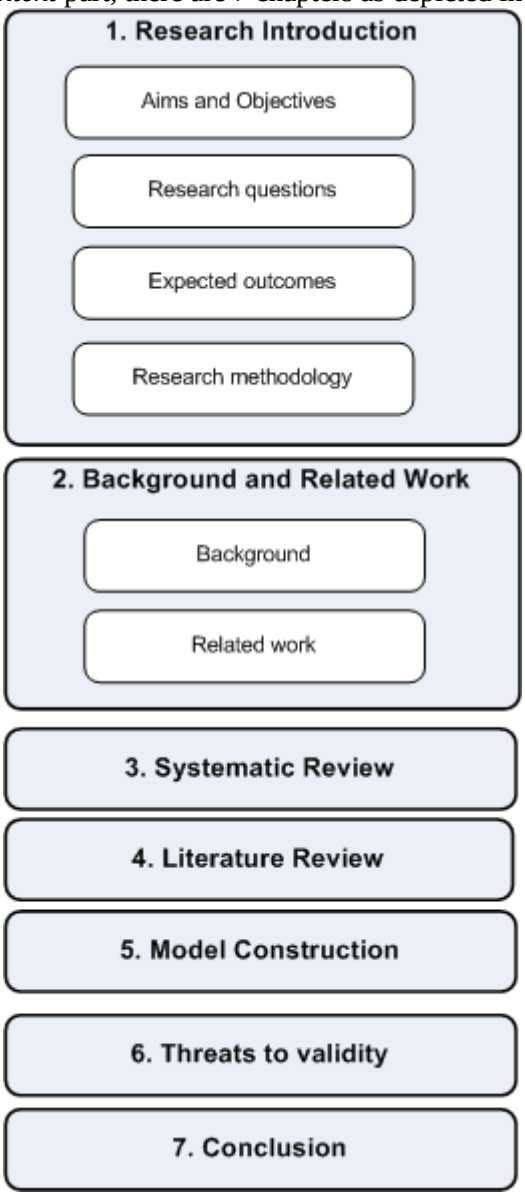

Figure 3. Structure of thesis

In the first chapter, *Research Introduction*, the readers will find introductory information about the study area and motivation, the research questions of which this study targets to find answers, the research methodologies used in this study and the expected outcomes.

The second chapter, *Background and Related Work*, aims to provide the readers with essential information to follow the topic of this study. There are two sections in this chapter. First section, *Background*, will equip readers with fundamental knowledge on Software Process Assessment and RE concepts. The next section brings the readers a brief summary of relevant work which has been done under similar motivation and existing gaps in the specified area.

In the next chapters of this research, the contribution of the study is presented. In *Systematic Review* and *Literature Review* chapters, the readers will find details on review strategies and conduction. A synthesis on the gathered information is analyzed

and findings are presented in these parts. Chapter 5 depicts the construction of the resulting model (namely Uni-REPM). Findings from previous studies will be analyzed and used to develop the RE process maturity assessment model.

In chapter 6, the threats to validity in this thesis work regarding the two reviews and the model creation will be discussed.

In chapter 7, *Conclusions*, the findings of this study will be revisited and final conclusion will be drawn. Future work will also be discussed in this last chapter.

In the Appendices part, 8 appendixes will be provided to denote the explored study results in details.

APPENDIX A: A list of identified practices from the research question 1.1 is summarized in this appendix.

APPENDIX B: A list of identified models and frameworks from the research question 1.2 is summarized in this appendix

APPENDIX C: A list of identified techniques and methods from research question 1.2 is summarized in this appendix

APPENDIX D: A list of identified tools from research question 1.2 is summarized in this appendix

APPENDIX E: In this appendix, a list of 125 good practices resulted from the research question 1.3 can be found.

APPENDIX F: In this appendix, the results from the combination analysis in the construction of Uni-REPM can be found.

APPENDIX G: This appendix provides the description of Uni-REPM model.

APPENDIX H: This appendix provides the description the checklist of Uni-REPM model.

2 BACKGROUND AND RELATED WORK

This section equips the reader with fundamental knowledge to follow the topic of this study. The first subsection describes general concepts of Software Process Maturity Assessment and briefly introduces some of the available frameworks and models for this purpose. The second subsection provides essential information specifically on RE processes. Related work is discussed in the last subsection to point out existing gaps in the area where the study targets to fulfill.

2.1 Background

2.1.1 Software process maturity assessment

The history of process improvement originated from the research of Deming in 1982 [26]. In his work, the result showed that a significant consequent business benefit would be obtained through improving the industrial process. The repeatability of the process was mentioned in the notion of process improvement where it helped to advance a process to be predictable. Deming then successfully introduced an improvement cycle for industrial process.

Adopting Deming's idea, Humphrey et al. in 1989 developed the notion of software process improvement where process maturity assessment was introduced [27]. This approach was the basis for many process maturity assessment models and frameworks. Process maturity assessment then became one of four essential steps in improvement process as mentioned in basic principle of [13, 15]. The main purpose of this particular step is to address the current status of a process based on which an improvement path could be elaborated and developed for the organizations.

In the literature, there have been a number of models and frameworks introduced following this approach which support the organizations to evaluate the maturity of their processes. The most renowned models are Capability Maturity Model (CMM) [28], Capability Maturity Model Integration (CMMI) or more specifically CMMI for Development (CMMI-DEV) [21], and ISO 9000 [22].

The development of CMM [28] was established in the 1980s at Software Engineering Institute (SEI) at the Carnegie Mellon University in Pittsburgh, Pennsylvania. Originally, CMM was developed as a method to objectively evaluate the contractors for software projects in military. Later on, the model was published outside the military boundary and became well-known. The goal of CMM is to provide practitioners methods to assess the maturity of their organizational processes and identify the improvement path. The model presents best practices which were extracted from various collected industrial data on a five-level maturity scale [28]. The first level which is called "*Initial*" demonstrates organizations which hold ad-hoc or even chaotic processes. The second level is "*Repeatable*" where organizations can manage to solve reoccurred problems based on their previous experiences. The third level, "*Defined*", presents a higher maturity where processes within organizations are predefined and documented so as involved individuals are aware of the working procedure and policies. The fourth level namely "*Managed*" indicates an improved state where organizations could statistically measure the performance and use that information to control their processes. The final level in CMM is "*Optimizing*" which represents the state where quality and productivity of the organization are continually improved. However, several encountered problems while using CMM have been reported. The major one is lack of integration. CMM serves different models for different functions. This causes confusions and conflicts for practitioners and increase training and appraisal costs.

ISO 9000 [22] is a family of standards published in the late eighties which can also be used to assess the maturity of software organizations. The idea of this model is to emphasize the importance of planning and documenting. It is a list of necessary steps that conform to Software Quality Management System. Based on that, the companies are able to ensure the quality of the system. However, in contrast to CMMI_DEV which focuses on development perspective, ISO standards tend to broaden to other business process activities [22]. For instance it also covers activities in sales and customer support area.

In 1991, the ESPRIT project BOOTSTRAP [29] has developed an assessment and improvement methodology by adapting the SEI experiences to European software industry using ISO 9000. In 1993, the SPICE project [30] was started in Dublin to develop an assessment framework for evaluating the organizational processes.

In 2006, CMMI was developed in order to overcome the drawbacks of CMM by integrating and standardizing the separate models of CMM. CMMI or specifically, CMMI-DEV is a collection of best practices to improve the effectiveness, efficiency, and quality of product and service development work throughout the whole lifecycle from conception to maintenance. The model derived the five-level scale from CMM, with one as the lowest level and five as the highest level of maturity. The organization with a high maturity level is assumed to develop a better software system or service.

There are more models and frameworks introduced for the purpose of evaluating process maturity [31] [32]. These investments have shown an increasing interest in and the real need of software process assessment in industries nowadays.

2.1.2 Requirements Engineering

RE process is “the process by which the requirements for software products are gathered, analyzed, documented, and managed throughout the SE lifecycle” [17]. Currently, RE has been roughly divided into two main approaches: bespoke and market-driven. Traditional software development usually involves specific customers. This type of development is so-called bespoke development [17]. The RE process of this development, known as bespoke RE process, basically deals with the needs of fixed customers. Differing from that, market-driven development targets to provide software products for a massive market [17]. This different characteristic introduces many new challenges in RE process such as continuous flow of requirements [33], shortage of time to market [34], release planning challenges [35], gap between market staffs and developers [11] and so on. Furthermore, in bespoke development, a specific project will be established once the contract between producer and customer is signed in order to produce the desired software product according to the customers' requirements. The project lifecycle covers RE process, together with other processes such as development, validation and so on [36]. Therefore, the RE activities mostly reside at project level. Nevertheless, in market-driven development, also referred as packaged development [37], the dedicated projects are created only when the subset of requirements are generated for the specific releases [14]. The main RE activities are usually performed earlier at product management phase. In other words, most of them start from higher level, at product-level.

Discarding the differences, common activities that could usually be found in RE processes are: elicitation, interpretation and structuring (also called analysis and documentation), verification and validation, change management and requirements tracing [17] [38]. There are many ways to classify the aforementioned activities into categories. This paper follows slightly the classification of the base model – REPM – in order to have a consistent view of the process. According to the study of Gorschek [23], the activities are grouped into three main areas: Elicitation, Analysis and Negotiation, and Management.

2.1.2.1 Requirements Elicitation

Requirements Elicitation basically aims to understand and gather the needs of the stakeholders (usually are the users and project sponsors) [17] [38]. In bespoke RE, the major part of this step is concerned with communicating with the specified stakeholders (especially the customers) in order to identify and extract the requirements for the software product. However, market-driven RE targets massive market, hence, there is no specific customer for directly eliciting requirements. Therefore, various sources such as internal stakeholders, competitors, distributors, subcontractors and so on are often used with assumptions to handle this step especially when developing the first release of the product [39]. It is also important to identify the key customers who have large impact on the market [14]. Furthermore, the multitudes of stakeholders often come up with new requirements rapidly [33]. This demands elicitation to be continuously conducted throughout the development process [14].

2.1.2.2 Requirements Analysis and Negotiation

Requirements Analysis and Negotiation is the activity where the collected information from the previous step is investigated [17] [38]. The purpose of this investigation is to detect conflicts, overlaps, omissions, and inconsistencies existing in the retrieved requirements in order to ensure a right set of requirements for later phases. In bespoke RE, the activities in this step are mainly to identify problems and find agreements with customers to change [17]. For instance, when conflicts are detected, negotiation with stakeholders will be required in order to solve these problems [17]. Negotiation can be performed to make tradeoffs between the stakeholders' requests and capabilities of the existing technology or provider [40]. Dependencies between requirements are also analyzed and estimation on consumed effort and risk for each requirement is also made in this step [17]. In market-driven RE, a large amount of continuously changing requirements is a big challenge for requirements analysis [33]. Hence, early classification can be used to dismiss irrelevant requirements before applying deep analysis [41]. In addition, there is no specific customer for negotiation. Therefore trade-off must always be made considering priorities of requirements in case there are conflicts.

2.1.2.3 Requirements Management

Requirements Management is the process that covers organizational and documentation activities. This process usually starts at the beginning of the RE process and remains until the end [17]. The main objectives of these activities are to ensure the quality of the process as well as the documents which are the main results of the whole process [17]. One of the most crucial activities in Requirements Management is change control. In bespoke RE, changes usually originate from customers. Hence, tracing backward can be easily performed and meetings can be hold for negotiating changes. However, in MDRE, it is a fact that working in an environment with many stakeholders would be much harder than working with a specific customer since they introduce more changes [11]. The research of Karlsson [11] showed that there are a number of factors impacting the requirements in MDRE such as changing market demands, rougher competition and key-customers who are usually not clear of their expectation. This raises a challenge in MDRE which is how to deal with these volatilities of the market.

2.1.2.4 Requirements Validation

Requirements Validation attempts to answer the question "Have we got the requirements right?"[42] [38]. In bespoke RE, the complete requirements specification are checked during this step in order to assure that the defined requirements are

correct, clearly and sufficiently defined for later phases and follows quality standards [43]. Due to the characteristic of bespoke RE where customers are specified, meetings directly with customers can be hold to perform inspection or review on the requirements specification. The output of this step is a list of problems and corresponding solutions [42]. However, since this contact does not exist in MDRE, the validation is often performed within the organizations and is usually left until the late phase in the development process [17].

2.1.2.5 Release Planning

Release Planning is one of the most important activity areas particularly in market-driven RE. Activities in this area mainly focus on making decisions to generate optimal sets of requirements for different releases of the product [17]. Unlike in bespoke development where plans for releases are usually defined by the customers [44], in market-driven RE, this timeframe is usually set by the organization regarding business concerns and is often strongly influenced by the market [45]. Hence, this area often involves requirements prioritization and business strategies in decision making. The result of these activities will be the input for project implementation.

2.2 Related work

A preliminary study in the literature indicates that apart from process maturity assessment models such as CMM, CMMI, ISO and so on which are introduced generally for the whole process; there are only few models that are tailored to RE process. Among those are the RE Good Practice Guide (REGPG) designed by Sommerville and Sawyer [43], the Requirement Engineering Process Maturity (REPM) developed by Gorschek [23] and the market-driven RE Process Model (MDREPM) [24].

REGPG is a three-level model in which RE good practices are collected and ranked according to their cost of implementation. It is served more as a guideline for process improvement rather than a tool for assessing the process maturity level.

REPM takes inspiration from CMM, ISO 9000 and the former as well as other references [23]. This model is used as a tool to measure on which level the project is regarding the RE process. It also shows what has been done, what is missing and identifies room for improvement.

While REGPG and REPM focus on the traditional customer-developer relationship, MDREPM ventures into the newly explored area of RE which targets on products for the whole market. MDREPM is not only a collection of good practices in market-driven RE but also the assessment tool [24]; therefore, it is the counterpart of the REPM. This model provides software organizations with a step by step process improvement path towards a better RE process.

A closer look at the assessment models and frameworks reveals that the requirement process has not been covered adequately. CMM, CMMI and ISO 9000 were mainly conducted based on experiences and industrial data without considering technologies suggested in research. Furthermore, the three models mostly focus on contract development approach hence did not cover market-driven RE. GPG model is a more systematic solution where research works were taken into account. However, the model encountered similar issues when exploring only problems in bespoke RE [42]. Considering the significant differences between market-driven and bespoke RE as well as the new challenges introduced in market-driven RE, it is hardly effective to apply these methods on organizations where market-driven development is involved.

Apart from that, experiences using the aforementioned models and frameworks for assessing the RE process in industries have been reported with various problems. CMMI_DEV does not give companies the answers for "how to conduct requirements analysis or what technique to use" in order to achieve a certain maturity level [46]. In addition, it also lacks of many activities for each Process area, for instance practices related to domain knowledge or stakeholder identification [21]. In the research of Smith, it was stated that none of the used perspectives (CMM and ISO 9001) "contained the total right answer for RE process" [47]. Besides, research has reported that GPG provides a numerous set of detailed practices with adequate ranks but is lack of guideline for implementation [48]. An empirical study of Nizia in 2003 [49] also showed that GPG caused confusions to practitioners due to the ambiguity of the measurement process and lack of systematic procedure when giving scores for different practices. Overcoming these disadvantages, REPM and MDREPM grouped the activities in MPAs and SPAs and presents necessary steps at each certain maturity level. Therefore, the companies could easily identify a set of activities they need to perform as well as skip non-applicable steps in order to "jump" to next level [23]. Although the two models have been validated in industries, both of them cover RE partially (either bespoke or MDRE). This limits the applicability of these models in many cases especially in hybrid development where both bespoke and market-driven approaches could be used.

Overall, there is no existing model that can effectively address both the bespoke and market-driven RE. As there are more and more organizations adapting market-driven software development [16], there is a need of an assessment model that can be used in both bespoke and market-driven developments. Therefore, the study of this thesis targets to create a universal process assessment model specifically for RE process in order to fulfill that need. Moreover, it has been shown that none of the mentioned assessment methods is constructed based on a widely systematic review on RE research. This has led to the lack of adequate knowledge in those models as reported from practical use. In an attempt to enhance the model credibility and capacity, the researcher aims to perform a systematic review on MDRE. The purpose of this review is to identify a set of MDRE "good practices" as the basis for the new model as well as to evaluate their applicability. By covering all the recommended RE "good practices", the model is deemed to be able to detect all the problems in the whole area hence the capacity of the model is expected to be high. This model can help organizations in evaluating their RE process as well as in guiding the improvement process with an up-to-date best practice collection.

3 SYSTEMATIC REVIEW

In this part, the systematic review on market-driven RE (MDRE) performed in order to gather good practices for the Uni-REPM model construction stage is described in details.

3.1 Necessity of a systematic review

As aforementioned in the introduction part, the aim of this study is to develop a maturity assessment model specifically for RE process. This caused a need of gathering all recommended activities in MDRE process. In order to decide whether a systematic review is necessary for this purpose, a preliminary search on the area was performed with the following details:

- Search string: *"Requirements Engineering"* AND *Systematic Review* AND *market*
- Search database: Inspec and Compendex
- Search fields: Article Title, Abstract, Keywords

The search targeted to identify all available review results on MDRE practices in Inspec and Compendex. However, no relevant result was returned. One systematic review on particular sub areas Release Planning was conducted but there has not been any extensive synthesis on MDRE practices.

The low response of the preliminary search confirmed the need to execute this systematic review.

3.2 Systematic review design

The main expected outcomes of this review are the "good practices" which are empirically supported so that it would be suitable and feasible to apply them in industry. Apart from that, the researcher is also interested to find out about the supported implementation for the practices and their validation in industry and academia. The validation context is also considered to judge the generalization of the solutions.

The systematic review process of this study mainly follows the approach of Kitchenham [15]. Moreover, the researcher considered all the lessons learned from the paper [50] (including lessons learned in specifying research questions, developing review protocol, validating review protocol, identifying relevant research, selecting primary studies, assessing study quality, extracting required data and synthesizing data) in order to tailor the original process so that it could be better adapted.

Figure 4 demonstrates the overview of the systematic review process performed in this study.

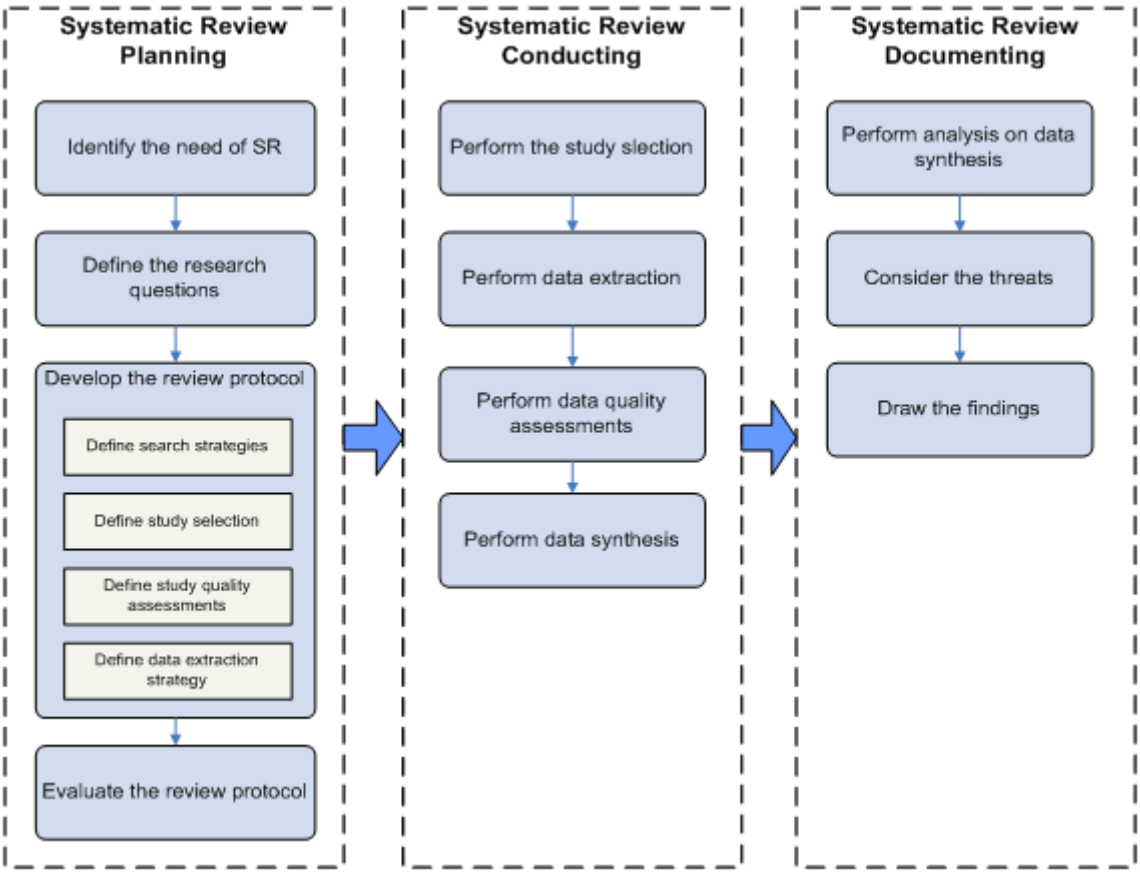

Figure 4. Systematic Review Process

Besides, as described in the Research Methodologies section, the researcher received support in this work from a colleague. Hence, in some parts of the systematic review, additional steps were performed in order to assure the consistency of the study.

3.2.1 Define the Research Questions

The main question for this study is: ***RQ.1. What are the good practices for market-driven RE?***

In order to obtain the answer better for RQ.1, the researcher attempted to break RQ.1 down into three sub questions. The sub questions and the objective of each question are described in the table III.

TABLE III. Research questions RQ.1 breakdown

Sub question	Objectives
RQ.1.1: What practices are explicitly suggested for market-driven RE?	To identify all the practices that are explicitly suggested in the research for MDRE
RQ.1.2: What practices could be extracted from the existing techniques/tools/methods/models for market-driven RE?	To identify the practices that have not been explicitly discussed in literature but could be derived/generalized from tools/techniques/methods/models for MDRE
RQ.1.3: Which practices from RQ.1.1 and RQ.1.2 are justified by empirical validation or/and rationale for market-driven RE?	To identify the practices could be considered as good practices based on the findings (practices that have quality validation or well supported in research)

Question RQ.1.1 aims to address all the practices that are suggested in the publications whereas RQ.1.2 targets to discover practices which have not been explicitly mentioned in literature but could be derived/generalized from tools/techniques/methods/models proposed for market-driven RE. Thereafter, the

results retrieved from RQ.1.1 and RQ.1.2 would be used as the input for studying and answering RQ.1.3 as shown in Figure 5.

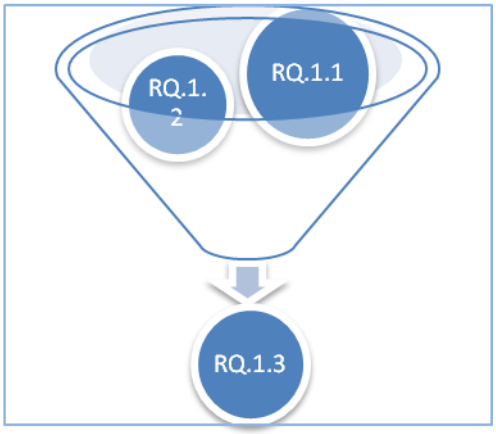

Figure 5. Relation among research questions

3.2.2 Development of a Review Protocol

3.2.2.1 Generating the search strategy

The purpose of this step is to construct the search strategy which consists of the terms, the resources and the process for searching.

a. Resources

In this study, considering the schedule of the thesis, the author attempted to enhance the sufficiency of the study by covering as many creditable databases as possible. Three of the largest reference databases were used for primary search following with two other ones:

- Compendex
- Inspec
- IEEE
- ACM
- Scopus

The reason to choose Conpendex and Inspec is that they are both extensive reference databases with a large numbers of well-known publications related to Software Engineering. Compendex has 11.3 million records from 1970-present across 190 engineering field [51] while Inspec has 11 million abstract records from 1969-present [52]. Both databases are updated weekly. Moreover, most of the papers found in these databases are peer-reviewed which are in the scope of this study.

This reason is also applied the case of SCOPUS, which is the largest abstract and citation database containing 38 million records since 1823 and it is updated daily [51]. In addition, SCOPUS provides a powerful search engine and facility.

Besides, the two databases IEEE and ACM are also included in the search resources. Although many of the publications found in these two databases duplicate with those obtained from the first three resources, searching is still performed on them in order to ensure the completeness of the results.

In addition, it is also important to mention that grey literatures were not considered in this study. The main reason is regarding the credibility of the new model. Since in this thesis, the research aimed to search for a one-fit-all solution which could provide assessment solution for all the organization regardless their characteristics. Therefore, it is not advantageous to include particular experiences and lessons learned reported from specific situations. Besides, grey literatures are usually not easily available. Hence, they were discarded from the primary studies.

b. Generating search string process

In this study, the researcher performed two-step generating process for search strings.

- **Step 1**: Defined initial search terms by:
  - Extracting major terms from research questions
  - Extracting keywords from known papers
  - Brainstorming
  - Indentifying synonyms of the terms
  - Constructing strings by Boolean AND/OR, double-quote
- **Step 2**: Iterated and refined search term. In this step, the researchers got in a loop to conduct searches on different combinations of initial terms and refine them until the retrieved results were considered adequately relevant.
  - Performed test searches on different combinations of initial search terms in the five chosen databases
  - Evaluated the search results
    - Pick up randomly some papers from the obtained results
    - If many papers in the result were found irrelevant then the search criteria and terms needed to be refined.
  - Repeated step 2 until the result was considered as good enough.

By following this process, it is ensured that expected publications are found using the search terms.

c. Final search strings

Along with the search process, the researchers performed refinement loop to evaluate and refine the search strings. Besides, as part of the attempt to ensure the quality of the search result, the identified publications were also checked against already known papers. The known relevant papers were selected from the materials of the course "Large-scale requirements engineering" and the repository for the group assignment which was part of former studies. Based on this, a list of 20 papers was assembled. Thereafter, this list was checked against the search result in order to identify how many percent of the papers in the list could be detected. The result showed that the formulated search strings could identify 17/20 papers. The remaining three papers were then reviewed to discover the reason and solution for refining the search strings. However, it was revealed that these three papers were about marketing strategies and value analysis in market-driven development. The researchers then had a discussion to determine whether marketing related articles would be filtered and finally decided to exclude those. The reason is that the researcher aims to keep the scope of the new assessment model specifically focus on RE activities. Marketing activities could support but not directly belong to this process area hence they are considered out of the scope. Therefore, no more search term was added to extend the search result and the search strings remained as defined before.

The final search strings after applying the above modifications are listed in the table IV. These search strings were applied to search in Title/Abstract/Keywords in all the selected databases. The evaluation step above together with the result showed a high level of confidence of the search strings.

TABLE IV. Search strings

market-driven "mass market" "consumer market" "release plan*"	AND	Requirements	AND	*practice* OR *technique* OR *method* OR *tool* OR *model* OR *approach* OR *solution*
"off-the-shelf" "packaged software" "large-scale"	AND	requirements engineering		
"product management"	AND	Software		

3.2.2.2 Study Selection

a. Study selection criteria

In order to accurately and effectively extract all the valuable data from identified studies as well as to ensure the consistency of the study, inclusion and exclusion criteria were generated for the study selection process. Inclusion criteria define the relevance of publications in which answers for the research questions could be achieved. Exclusion criteria were composed so that all effort could be spent efficiently on relevant studies.

Inclusion criteria

The publications are included if they confirm all of the following criteria:

- C.I.1. The article is in English language.
- C.I.2. The publication year of the paper is from 1993 to the point of conducting the search (February 2010). The reason for this limitation is that this study aims to focus on up-to-date practices while minimizing the risk of losing important classic papers. A study on market-driven research history was performed to determine when MDRE started to get increasing attention. Based on this research, one of the most well known studies of software engineering practice early performed in real organizations was conducted by Curtis, Krasner and Iscoe in 1998 [40]. However the idea of market-driven was not mentioned until 1993 in the research of Lubars [54] where he proposed the two categories: *customer-specific* and *market-driven* project. Since then, this area gained more and more attention, namely in the research of Carmel-1995 [55], Hutchings 1995 [56], Potts 1995 [37], Regnell 1998 [57] and so on. Therefore, year 1993 was chosen as the starting year for data collection.
- C.I.3. The article is peer-reviewed.
- C.I.4. The title or abstract discusses about the MDRE or topics related to research questions.
- C.I.5. The article is available in full text.
- C.I.6. The introduction discusses about the practice/model/method/technique/tool in MDRE.
- C.I.7. The content of the article discusses about:
    - Summary of practices, models, methods, techniques or tools in market-driven RE process
    - Comparison between practices, models, methods, techniques or tools in market-driven RE process
    - Validation of practices, models, methods, techniques or tools in market-driven RE process
    - Proposal of practices, models, methods, techniques or tools in market-driven RE process

Exclusion criteria

The publications are excluded if:

- C.E.1. The article is duplicated to a chosen one.
- C.E.2. The article discusses about practices of business analysis, marketing or resource scheduling in market-driven requirement engineering.
- C.E.3. The article is related to specific systems, which have special characteristics that are not widely applied e.g. grid computing systems, spacious systems, COTS-based system.
- C.E.4. The article is specifically about Product line development, or Notation construction, or component selection for COTS-based systems or Aspect-oriented approaches.
- C.E.5. The article mainly discusses about challenges and problems in MDRE but does not provide any beneficial solution or suggestion to solve such problems.

b. Study selection process

After applying the search strings on the designated databases and obtaining the search results, the researchers excuse primary study selection from the identified results. The whole retrieved publications were divided equally to each researcher based on the list order. In order to ensure that the selection process was performed homogenously by the two researchers, pilot studies were conducted and the Copen's Kappa coefficient was calculated. Thereafter, each researcher accomplished the selection independently using the inclusion/exclusion criteria. Any unsure article was discussed by both researchers before final inclusion/exclusion decision was made.

Firstly, the publications were checked against criteria ***C.I.1 "Is it in Engish?"***, ***C.I.2 "Is it from 1993 to Feb 2010?"*** and ***C.I.3 "Is it peer-reviewed?"***. These three steps were done automatically by incorporating the criteria with the corresponding options provided by the database search engines while applying the search strings. Therefore, the publications captured from the search process already satisfied those criteria. After that, the selection process follows loosely the two-stage process mentioned in Brereton's study [50]. The titles of the identified studies were reviewed first against a set of criteria and in case of uncertainty, the abstracts were read later to ensure that no relevant paper was excluded due to poor title. Subsequently, the researchers attempted to retrieve full texts of the papers not previously rejected. For those papers that could be retrieved, their introductions were reviewed. Finally, the whole publications were read and checked against the inclusion/exclusion criteria in order to obtain the final primary studies. Figure 6 demonstrates the selection process using the developed inclusive and exclusive criteria.

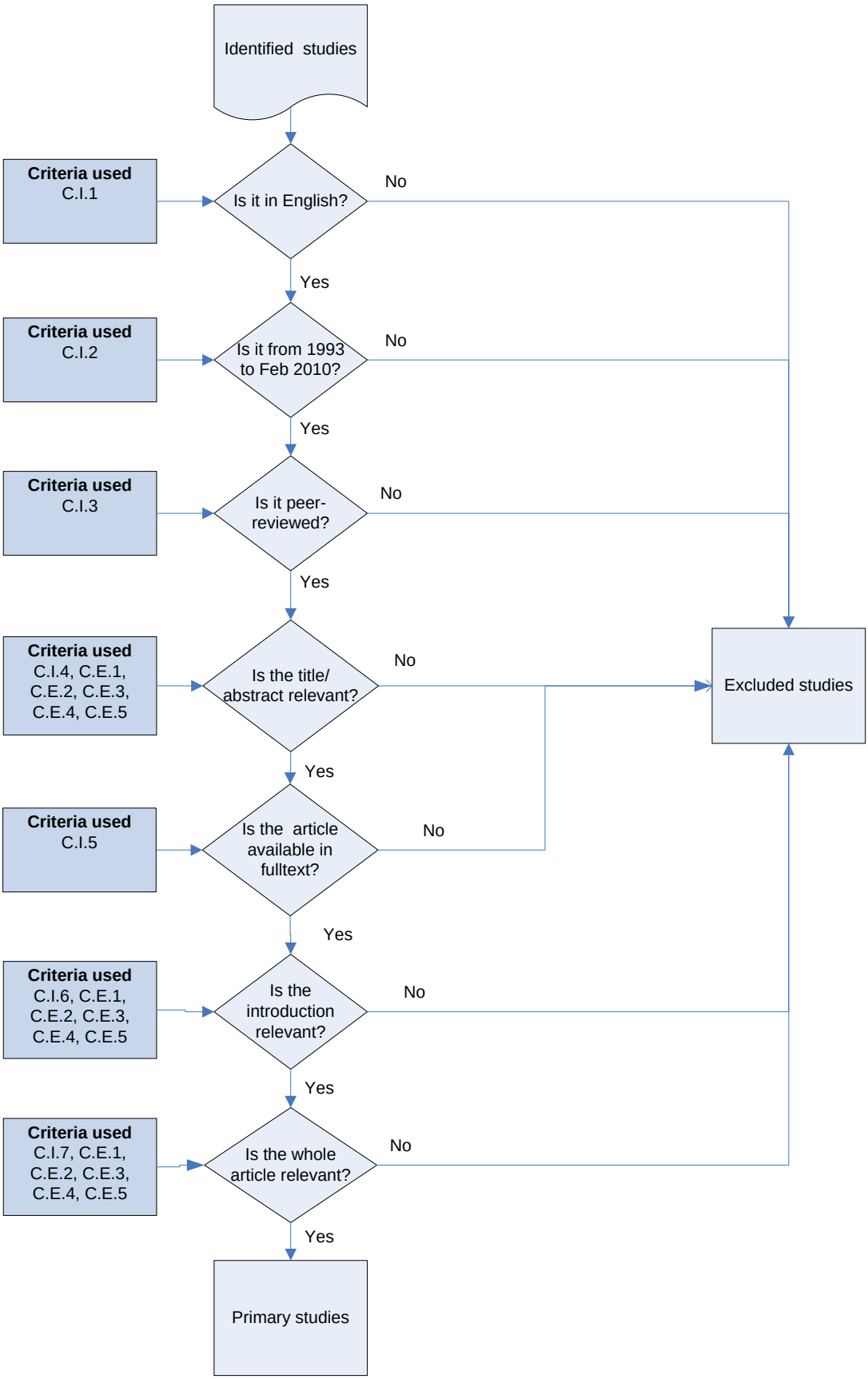

Figure 6. Study selection process

According to Brereton [50], piloting the protocol is essential to reveal problems in different stages of the review process. Hence, in order to ensure the quality of the selection, a two-stage piloting study was proposed. Between these two stages, the

researchers had a meeting to discuss, confirm and construct the common understanding.

First pilot selection study

In the first pilot selection, the papers were extracted from the search result with the third search string on Scopus database. Two researchers randomly picked 50 papers from the search result to individually study the title and abstract (C.I.4). The objective was to classify those papers to one of the three categories:

- Chosen papers: papers which are definitely relevant to the study of this systematic review.
- Rejected papers: papers which are irrelevant to the study of this systematic review.
- Unsure papers: papers of which relevance to the study of this systematic review is doubtful.

After performing the selection, the Cohen's Kappa coefficient was calculated in order to evaluate the agreement level between the two researchers. Cohen's Kappa coefficient is a statistical measure which could be used to quantify the agreement level between two raters [58]. The reason to choose this method is that it is a simple, quick and easy method which is appropriate for appraising agreement in nominal rating [58]. The calculated Cohen's Kappa coefficient returned 0.78 which indicated a quite good agreement level and common understanding of the two researchers. However, exclusion criteria were considered since many of the papers could satisfy those but did not seem to be relevant.

Therefore, the researchers had a discussion on what is "*in*" and what is "*out*" of the scope of this study. Finally, the two researchers decided to add the following exclusion criteria:

- Those papers which are specifically about *Product line development, Notation construction, Component selection for COT-based systems* and *Aspect-oriented approaches* will be excluded.

Second pilot selection study

In the second pilot selection, 50 papers were extracted from the Compendex database by the first search string. The method of the selection was the same as the first pilot selection.

The Cohen's Kappa coefficient was calculated and again showed a good agreement between us (0.76). This result motivated us to divide the work for the primary selection.

3.2.2.3 Study Quality Assessment

Quality assessment is an important step in systematic review to evaluate the quality of the selected publications [15]. In this research, the researcher's interests are practices well-supported by empirical validation and rationale. Hence, this step was performed mainly to judge the quality and reliability of the rationale or validation proposed in the selected articles.

The quality assessment criteria were developed in a form of a checklist and shown in Table V. The criteria can be answered by "Yes", "No" or "Partially". While studying full-text of the selected articles for data extraction, these criteria were assessed and recorded. The idea of a paper was considered as fully explained once the whole context of the study, the motivation under it, the research methodology and the findings are clearly described. Since major part of publications in this study was expected qualitative (due to the nature of this area), the generalizability of these papers was mainly regarded by validation context, evaluation method and findings from the validation steps.

TABLE V. Quality Assessment criteria

Quality Assessment Criteria	Answer
Is the idea of the solution fully explained in the study or in other referred studies?	Yes/Partially/No
Can the findings of the study be generalized?	

3.2.2.4 Data Extraction Strategy

The purpose of this step is to accurately collect all the relevant information from the selected studies in order to answer the research questions and the study quality assessment [15]. The contents of the data collection form and the extraction procedure are explained below.

a. Contents of Data Collection Forms

The data collection forms were designed to include standard information such as name of data extractor, title of publication as well as other data values related to research questions. They were compiled by both researchers and piloted on a sample of studies to ensure its usability and completeness. Table VI and VII present the data extraction forms composed for this study.

TABLE VI. Data extraction form 1

Data Item	Value
Data extractor name:	
Data extraction date:	
Article Title:	
Authors:	
Article Type:	Journal/Conference/Conference proceeding
Publication year:	

TABLE VII. Data extraction form 2

Data Item	Value	Mapping to RQs
Context type:	Academia/Industry	RQ.1.1
Research methodology:	Experiment/Case study/Survey/Action research/Other	RQ.1.3
Name of practice/model/tool:		RQ.1.1, RQ.1.2
Type:	Practice/Model/Method/Technique/Tool	RQ.1.1, RQ.1.2
Purpose:		RQ.1.2
Related software process area(s):		RQ.1.2
Validated in:	Academia/Industry	RQ.1.3

b. Data extraction procedures

A pilot running on data extraction was performed to detect any misinterpretation between the researchers and ensure a common understanding in filling in the form. During piloting step, the researchers selected one sample article in the primary studies and each person extracted data independently. The complete data extraction forms were then compared and discrepancies were discussed to reach consensus. After that, in the actual run, the selected publications were equally divided between the two researchers. Because of the time and resource constraints, each researcher independently performed the data extraction using the above forms. So as to ensure that the extracted data were consistent and reliable throughout the process, random cross-checks in a similar way of the pilot run was also performed.

In case same data are reported in several studies, only the most recent one will be used in order to avoid result bias.

3.2.2.5 Data Synthesis

After data were extracted from the selected primary studies, they needed to be collected, analyzed and summarized [15]. The researcher thereafter performed descriptive synthesis as the extracted data related to MDRE practices are mostly qualitative. The results from the data extraction fields were presented in tabular form in order to show the similarities and differences between the study outcomes. If two or more studies discussed the same practice, their data would be synthesized to some extent. In case two or more studies discussed techniques which shared many common characteristics, they could be grouped together to form a common practice.

3.2.2.6 Duplication avoidance

In this study, the researchers intended to cover also systematic review and literature review results in MDRE. Hence, there was possibility that single studies could be replicated in the review(s). In order to avoid this issue, systematic review and literature review studies were separately considered in the analysis step. Two following questions were regarded throughout this work:

- *If the review covers studies which have been fully included in the selected results?* In case the review was a synthesis of several papers in the primary studies, it would be discarded and practices presented in such study would not be counted.
- *Do the conclusions of the review contribute other findings than those of the primary studies?* This implies that the review may cover more than the selected papers. Hence, practices captured from the review need to be checked against identified practices from other papers. Duplicated activities would then be removed.

3.2.3 Evaluate the Review Protocol

During the planning phase, the review protocol was frequently refined so as to correctly document the process. The research questions search terms, selection criteria and data extraction forms were subject to most of the refinements. The initial protocol was sent to the supervisor Dr. Svahnberg and the professor Dr. Gorschek for review and feedback. The inputs from the review were then be analyzed and incorporated into the protocol.

3.3 Systematic review execution

3.3.1 Search result

After the search strings were finalized, they then were applied in the five selected databases to identify papers for the study. The systematic review was conducted by both researchers. The databases were divided equally between the researchers, so each researcher independently applied the same three search strings on two databases (Inspec + Compendex + IEEE or SCOPUS + ACM). There are totally 1620 publications retrieved from the search. The breakdown of the search result according to each database is shown in Table VIII.

TABLE VIII. Primary search results

Sr. No.	Name of database	Total number of results found
1	Engineering Village (Inspec, Compendex)	617
2	SCOPUS	571
3	ACM	254
4	IEEE	178
	Total	**1620**

3.3.2 Selection result

In order to manage the references, EndNote Web was used. The raw collection was then filtered to remove duplicated publications by using “Find Duplicates” function in EndNote Web. However, as there were remaining duplications detected, the collection underwent a manual duplication removal. The publications were sorted by title and identical ones were detected and removed. This step resulted in 432 removed papers and 737 papers left for selection.

Within 737 remaining papers, 585 papers were excluded by reading their title and/or abstract. Because the next step of selection based on reading introduction, the full text of the 152 selected papers had to be retrieved. The researcher managed to obtain 140 papers in full-text from the school library and by other means (Google, other school libraries). The inclusion/exclusion criteria were then applied on the introduction and full-text reading, which resulted in 48 selected papers for data extraction. Although there were a significant number of articles found in MDRE, a quite large portion of them mainly discussed on challenges and problems in the area. Since the thesis focused specifically on searching for the solutions, those studies were discarded. Only studies with concrete answers on how to solve problems were considered which resulted in such limited number of identified studies.

Figure 7 illustrates the complete process and the step-by-step result of the systematic review from searching to final selection.

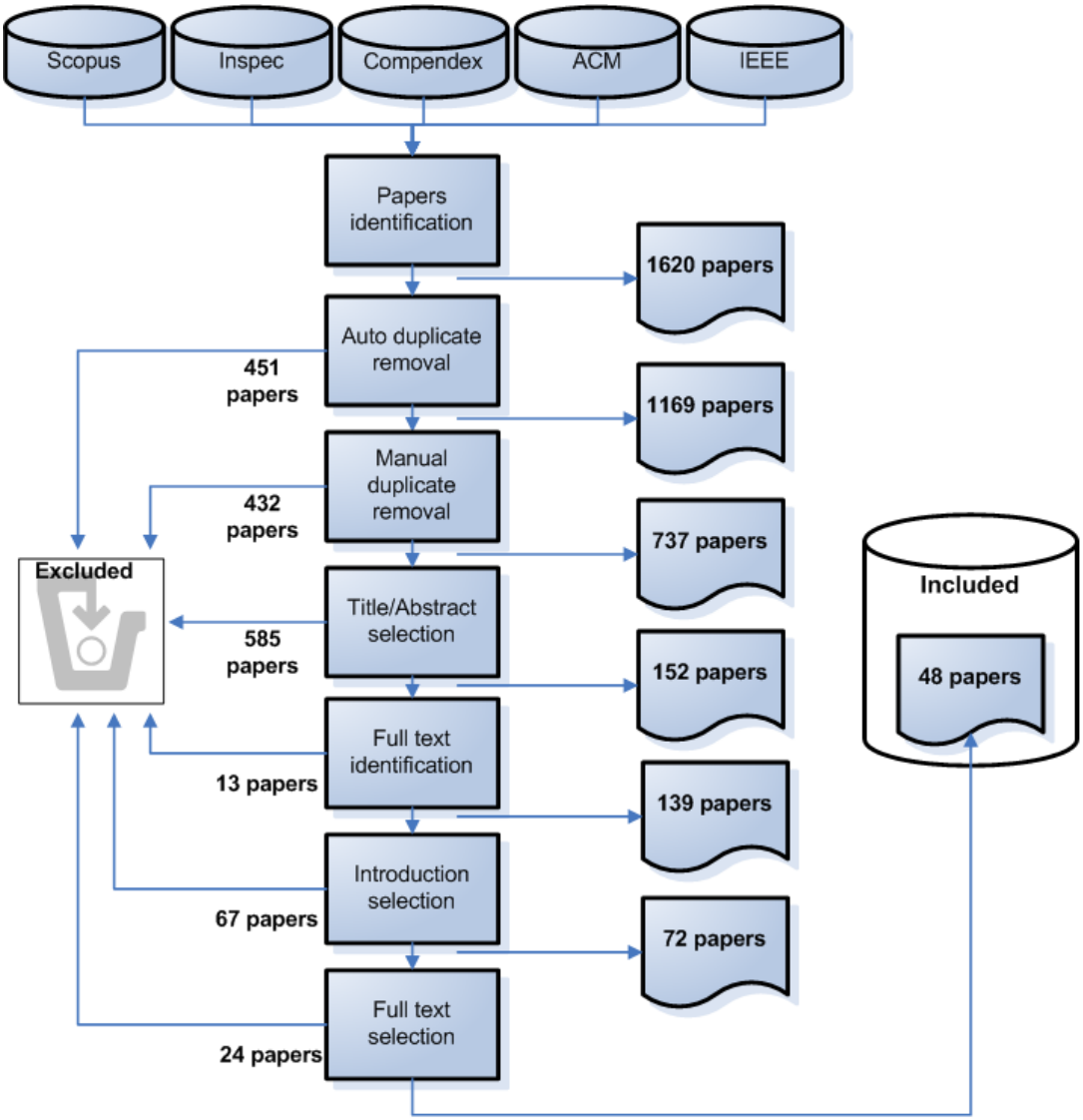

Figure 7. Primary study section execution

3.4 Systematic review result

3.4.1 General characteristics of the primary studies

The primary study selection process has returned a considerable amount of research in the area of market-driven RE since the year 1993. The section gives an overview of the general characteristics of the identified studies following with the detailed analysis and answers for the stipulated research questions.

3.4.1.1 Publication years

A total of 48 papers published from the year 1993 were found to discuss about market-driven RE. Figure 8 shows the distribution of the publications along the published years. An interesting finding emerges as 28 out of 48 papers (58%) were published from 2006 onward though there were relatively a few number of papers studied this area during the previous years. This phenomenon may be the result of a trend starting to increase investments on market-driven RE in research from 2006. This reinforces the quality of the aimed model because they will serve as up-to-date inputs for the model construction in next step.

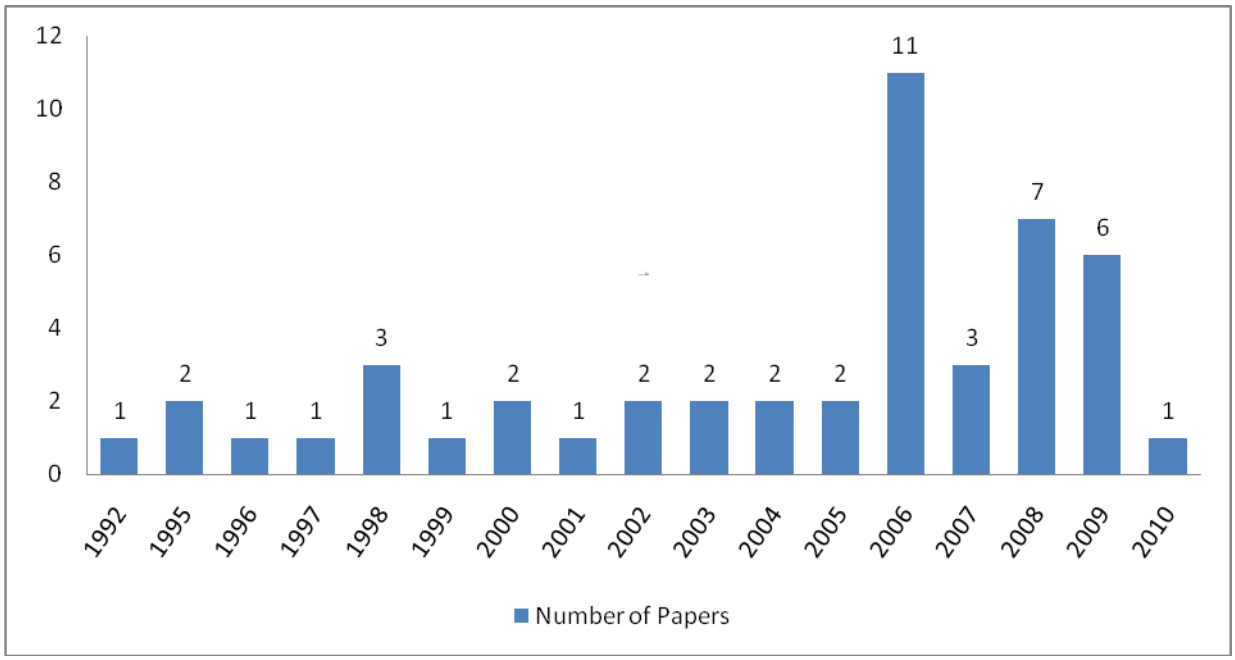

Figure 8. Distribution of the publications according to the publication years

3.4.1.2 Context and Research methodologies of the primary studies

It was quite surprising that the contexts of primary studies were not easy to extract. The reason is that many of the studies were conducted not purely either in academic or industrial environment (e.g. academic study with support of empirical data). Hence, the following categories were defined for the study context:

- **Academia**: includes studies those were purely performed in academic environment.
- **Industry**: the studies were performed in industrial organizations (or in collaboration with academia), or with support of industrial data (e.g. from case study, interview or survey with industrial practitioners).

From Table IX we could see that 33 out of 48 investigated papers (67%) were carried out in the industry context. This shows that the proposed solutions are, to some extent, close to industrial needs. The result of the analysis supports the objective of the systematic review which is identifying "good practices" that can be applied in real work.

TABLE IX. Context distribution of primary studies

Context	Number of papers
Academia	15
Industry	33

The researchers also experienced that the research methodologies were usually not mentioned in the research. Even if they were mentioned, it was done poorly and inconsistently. In many cases, the authors only stated that qualitative methodology was used. Hence, in order to have a clear and consistent overview of the studies, the research methodologies were classified into 7 categories with the following criteria:

- **Systematic review**: Study performed to systematically synthesize relevant researches to answer pre-defined research questions
- **Literature review**: Review on several relevant studies conducted to answer predefined research questions.
- **Case study**: The study declares one or more research questions which are answered by applying a case study
- **Others**: The research proposes new theory from author's experience (without empirical data support).
- **Experiment**: The study conducts an experiment
- **Grounded theory**: The study proposes new solutions based on empirical data collected
- **Action research**: is a reflective process of problem solving to improve the way issues are addressed and problems are solved .The study is usually conducted in industrial settings.

Figure 9 shows the distribution of papers according to the previously defined categories. "*Action research*" (35%) and "*Grounded Theory*" (17%) constitute the majority of the 48 selected papers for this systematic review. This result confirms the researcher's intention to focus on the feasibility of the research findings to strengthen the model applicability.

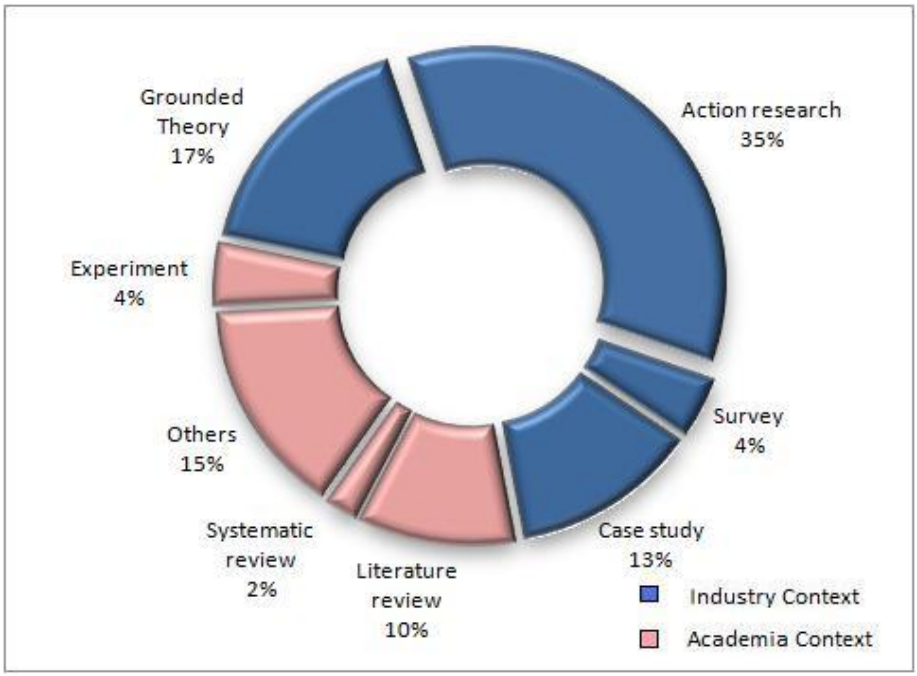

Figure 9. Research methodology distribution of primary studies

3.4.1.3 Quality assessment of individual studies

The quality assessment of the selected publications was conducted according to the protocol defined in the systematic review plan. The purpose of quality assessment is to know if the selected studies have good quality and their findings are reliable to base the model upon in the later stage. Table X shows the summary of the study quality assessment.

TABLE X. Quality assessment results

Quality Assessment Criteria	Yes	Partially	No
Is the idea of the solution fully explained in the study?	40	7	1
Can the findings of the study be generalized?	18	26	4

As we can see from Table X, in most of the cases (40 out of 48 papers), the solution presented in the study are described adequately and clearly. Moreover, the findings from 18 selected papers can be generalized while those in 26 other papers can only do so to a certain extent. This is due to the fact that most of the selected studies used qualitative research methodologies, which does not focus on generalizing the result [59]. However, many of the results can be transferable given that the research participants are selected carefully and the research field and result are described in great details [60].

In general, a majority of the selected studies demonstrated an acceptable quality as they positively responded to the two quality assessment questions (either yes or partially).

3.4.2 Answers for the stipulated research questions

Based on the data extraction field designed in the plan, a Microsoft Access database was set up to store the data extracted from the 48 selected papers. Each field in the data extraction form corresponds to a field in the MS Access database. The criteria in the quality assessment form are also mapped to fields in the database in order to facilitate the assessment logging during data extraction. Each selected paper is associated with one record in the database.

3.4.2.1 Research question 1.1

RQ.1.1.What practices are explicitly suggested for market-driven RE?

a. Results

The focus of this question is to find out practices which are suggested to be performed in market-driven RE process.

By inspecting the primary studies, 163 practices were uncovered as beneficial for market-driven RE processes. Those practices are:

- Suggestions in the papers which aim to help companies face challenges in market-driven RE (MDRE).
- Solutions proved/validated in the papers which aim to be usable and useful for market-driven RE.
- Activities mentioned in the framework or process model for MDRE development.
- Solutions proposed to perform particular tasks in MDRE.

During the review, some practices were found both in single studies and in literature/systematic reviews. Furthermore, some practices present similar ideas or perform similar tasks even when their names differ. Therefore, a large effort was invested in analyzing, detecting and removing overlapping practices. In the end, 137 practices remained.

In addition, those practices were usually proposed under diverse contexts and to different extent. The practices themselves also came from different levels of abstraction. Hence, in order to solve this problem, a categorization based on the goals of RE activities was established following the classification in Wohlin's book [17]. It consists of five activity areas:

- Requirements Elicitation
- Requirements Analysis
- Requirements Management
- Release Planning
- Requirements Validation

The number of captured practices based on categories is presented in Table XI. The detailed information of these 137 practices can be found in the Appendix A.

TABLE XI. Practice distribution according to activity areas

Activity Area	Number of practices
Requirements Management	55
Requirements Elicitation	17
Requirements Analysis	35
Release Planning	19
Requirements Validation	11

The duplicate removal step conducted in prior to this summary however retained the number of mention of each practice. Hence, a list of most proposed practices in each activity area could be extracted as shown in Table XII.

TABLE XII. Most proposed practices in each activity area

Practices	Activity Area	Number of mention
Collect/extract requirements and save to repository	Requirements Elicitation	2
Estimate effort for requirements implementation	Requirements Analysis	2
Estimate resources	Requirements Analysis	2
Identify interdependencies between requirements	Requirements Analysis	3
Construct roadmap	Requirements Management	3
Note Requirement state	Requirements Management	2
Consider different dimensions in Requirements prioritization	Release Planning	4
Perform prioritization	Release Planning	6
Plan for more than 1 release ahead	Release Planning	4
Requirements selection	Release Planning	2
Write detailed requirements specification	Requirements Management	2
Emphasize whole-product thinking	Requirements Management	2
Explicate planning levels and time horizons	Requirements Management	2
Integrate a reuse measurement process into the RE process	Requirements Elicitation	2

b. Analysis

From Table XI, it is noticed that the 137 captured practices spread broadly to all the areas in RE. However, they do not distribute equally. A large portion of identified practices (40%) were proposed for Requirements Management area while only a few of them (20%) were suggested for Requirements Validation. This may due to the reason that Requirements Management is considered as a complex area containing various tasks. This may also imply that this area will need to be broken down in the new model so that problems in this area could be discovered adequately. Furthermore, from Table XII, it could be seen that the most proposed activities found are about requirements prioritization (6 + 4 times of mentioning) and planning multiple releases (4 times of mentioning). This shows a high interest in Release Planning area where activities are severely complicated and difficult. In particular, prioritization is the most critical activity that remains huge influence on the decision making and later steps.

c. Summary

In total, 137 practices have been detected for market-driven RE. These practices cover almost all the concepts in the area, namely Requirements Management, Requirements Elicitation, Requirements Analysis, Requirements Validation and Release Planning. This result together with the answer from RQ 1.2 was analyzed further in order to retrieve the answer for RQ 1.3.

3.4.2.2 Research question 1.2

RQ.1.2. What practices could be extracted from the existing models, frameworks, methods, techniques or tools for market-driven RE?

a. Results

As shown in Table XIII, through data extraction, 12 models and frameworks, 38 methods and techniques as well as 10 tools were discovered for requirements activities in MDRE.

TABLE XIII. Identified supports

Identified data	Amount	Reference
Models and Frameworks	12	Appendix B
Methods and Techniques	38	Appendix C
Tools	10	Appendix D

b. Analysis

Based on these models, frameworks, methods, techniques, tools extracted from the publications, the researchers attempted to identify those with similar purposes and group them together. Supporting artifacts that had same names and/or descriptions were only presented once in the data synthesis in order to avoid duplication.

Through investigation and synthesis, 16 practices were generalized from the extracted models, frameworks, methods, techniques and tools. In all cases, several artifacts supporting the same goal were generalized to one practice which can summarize them, e.g. "*Define and Maintain a Requirements Management Process*" was derived from 3 process models and frameworks defining the procedure for managing requirements.

The details of 16 extracted practices are shown in Table XIV. The descriptions of these practices are presented in the following section.

TABLE XIV. Generalized practices

Name	Type	Name of generalized practice
Roadmapping	Technique	Define Product Roadmaps
Agile roadmapping	Technique	
Quper	Technique	
Software Product Management process	Model	Define and Maintain a Requirements Management Process
Software Product Management Workbench	Framework	
REQUEST	Model	
REPEAT	Model	
RTM system	Tool	Consider Tool Support for Requirements Engineering
RM Trak - RM Trak	Tool	
Caliber RM - Borland	Tool	
Requisite Pro - IBM Rational	Tool	
Vital link - Compliance Automation	Tool	
ReqMan	Tool	
A market Driven REQuirements Management Tool (MDREQ)	Tool	
Lightweight replanning process model	Model	Define a Process for Managing Change and Evolution
Traceability model	Model	Define Traceability Policy
Goal-oriented requirements communication	Model	Establish Communication among Different Involving Teams
Enhanced requirements elicitation and mobile system construction	Method	Adapt Elicitation Technique according to Situation
Customer involvement factory	Model	
Customer Participation Sessions (CPSs)	Method	
Acquisition of Requirements (ACRE)	Framework	
Observation	Technique	
Unstructured Interview	Technique	
Structured Interview	Technique	
Protocol Analysis	Technique	
Card sorting	Technique	
Laddering	Technique	
Repertory Grids	Technique	
Brainstorming	Technique	
Rapid Application Development (RAD) Workshop	Technique	
Ethnographic Methods	Technique	
Rapid Prototyping	Technique	Create Artifacts to Facilitate Elicitation and Analysis
Scenario Analysis	Technique	
Requirements abstraction model	Model	Perform Refinement and Abstraction of Requirements
A Method for Early Requirements Triage and Selection	Method	Perform Requirements Triage
ReqSimile	Tool	Analyze for Double Requirements
Visualization of interdependencies	Technique	Document Requirements

		Dependencies
Value Point estimation	Method	Perform Requirements Estimation on Value and Cost
FSM (functional size measurement)	Model	
Analytical Hierarchy Process	Technique	Perform Systematic Requirements Prioritization
Binary search tree	Technique	
Bubble sort	Technique	
Cost benefit analysis	Technique	
Cumulative Voting	Technique	
Hierarchical Cumulative Voting	Technique	
Minimal spanning tree matrix	Technique	
Multi-Attribute Utility Theory	Technique	
Numeral assignment	Technique	
Outranking	Technique	
Planning game	Technique	
Pair-wise comparisons	Technique	
Priority groups	Technique	
Top 10 Requirements	Technique	
Tool-supported pair-wise	Tool	
Weighting Method	Technique	
Weiger's	Technique	
Cost-value approach	Model	Pack Requirements into Releases
Cost-value tool support	Tool	
Post-release Analysis of Requirements Selection Quality	Method	Post Requirement Selection Evaluation
PARSEQ tool	Tool	

Detailed information of the generalized practices

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

- **Define and Maintain a Requirements Management Process:** Having a pre-defined process to manage requirements ensures a well-organized way to control the whole requirements process and to guide the stakeholders of what to do next and how should it be done in a structured way. At pre-project level (product level), there is usually continuous stream of huge amount of requirements. Hence, concurrent approach models such as REPEAT or REQUEST are preferable.

- **Consider Tool Support for Requirements Engineering:** With a huge number of requirements in market-driven RE, it is crucial to have a database to store them. It is more advantageous if a centralized repository for requirements e.g. RTM system is installed so that all the changes will be applied in real time and different stakeholders can have the same view at a set of requirements. Moreover, powerful tools such as Requisite Pro - IBM Rational can improve and facilitate requirements handling, change management, traceability etc.

- **Define a Process for Managing Change and Evolution:** Change has been agreed to be the nature of requirements. Since there is no way to avoid it, a process to control it should be defined. The change process should cover certain steps a change

request must follow and requirements mentioned in the request should be considered. When there is a market change, a re-planning method should be used in order to accommodate the change in the requirements selection for release.

- **Define Traceability Policy:** Along the requirements process, numbers of changes on requirements will mostly occur. In order to ensure the consistency of the system, it is important that there is preparation for these cases so that it is possible to trace from requirements to other artifacts of the project and vice versa to apply necessary changes.

- **Obtain Common Understanding of Requirements among Different Involving Teams:** Common understanding on requirements (i.e. meanings, estimation values, prioritization rationale…) should be shared between different involving teams to reduce gaps. This activity may include regular meetings, emails or informal discussions to exchange necessary information. The practitioners should consider other teams which will later work with their outputs such as implementation or testing teams.

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

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

- **Perform Refinement and Abstraction of Requirements:** Requirements commonly come from many sources with different ways of expecting; hence they are usually diverse in levels of abstraction. By performing refinement/abstraction to synchronize their abstraction levels, it will be easier for understanding, managing and further activities such as prioritization.

- **Perform Requirements Triage:** This step is to early dismiss the irrelevant requirements so that the huge amount of initial requirements could be reduced to avoid wasting time and effort for future works. In order to perform this step, the requirements should be aligned with the strategies and plans of the product.

- **Analyze for Double Requirements:** In MDRE, as new requirements are continuously issued, there exists a need to compare them with old existing ones in order to reduce effort for reanalysis or update the old requirements with new information. Therefore, a technique or tool like ReqSimile can help in finding and linking similar requirements more efficiently and correctly.

- **Document Requirements Dependencies:** It is important during the analysis to consider the relations among requirements. There are requirements that require the others requirements to be implemented before, or exclude the others. Hence, having an overview of these relations will help practitioners in later phase, e.g. in release planning, change control, etc. Having the visualization of requirements dependency can show clusters of interdependent requirements, as well as highly dependent requirements at a quick glance.

- **Perform Requirements Estimation on Value and Cost:** In order to perform release planning, there is a need to estimate the value and cost of the requirements. Currently, very few estimation approaches are used by managers. The value point estimation method is used to evaluate the gained value (covering financial and non-financial value) of the software product. On the other hand, the FSM model could be used to estimate size and cost of requirements (both Functional and Non-functional).

- **Perform Systematic Requirements Prioritization:** Requirements prioritization at pre-project level helps to determine the relative necessity of the requirements. With a huge number of mandatory requirements which are impossible to be implemented all at the same time, it is crucial to specify which are more critical than others. There are many available techniques and methods for requirements prioritization such as AHP, Cumulative Voting and Planning Game.

- **Pack Requirements into Releases:** Requirements after being prioritized will be selected to certain releases. The selection activity usually requires the involvement of different perspectives from marketing, developing and management, etc. The selection can take into account many different aspects such as cost, value, resources. The cost-value approach can help the selection process considering the requirements cost and value.

- **Post Requirement Selection Evaluation:** Post-release evaluation is the step to assess the quality of the requirements selection in the previous step. This is to uncover previous mistakes/misunderstanding, gain experiences in decision making, hence ensure the quality of decisions for future releases. The evaluation can be done by analyzing the measurements of different product releases after they were released. Based on this information, the planners could see whether he/she had made a correct decision at that time and further investigate the mistakes. The PARSEQ method and tool can be used to support this practice.

c. Summary

Based on the models, frameworks, methods, techniques, tools extracted from the publications, 16 practices were derived. These practices are considered as highly feasible in real work since they were proposed with concrete supporting tool(s). Therefore, these practices are served as input for the RQ1.3 to find out the last set of "good practices" for the new model.

3.4.2.3 Research question 1.3

RQ1.3. Which practices from RQ.1.1 and RQ.1.2 are justified by empirical validation or rationale for market-driven RE?

a. Results

In order to answer this question, 137 practices from RQ.1.1 and 16 practices from RQ.1.2 were joined in the input set (153 practices) for inspection. In addition, the researchers performed a preliminary analysis on this set in prior to the inspection to avoid duplication. The study returned 9 overlapped practices which were then removed from the set. 146 remained practices were further investigated for this question. The 9 overlapped practices can be found in Appendix A with grey background.

The main purpose of this systematic review is to identify "good practices" to be served as input for the model creation in the next stage. The "good practices" here mainly imply feasible practices which are well-motivated or well-supported by

empirical data and rationale. Hence, the researchers conducted an analysis to evaluate the credibility of the detected 146 practices from RQ.1.1 and RQ.1.2. Figure 10 illustrates the evaluation process.

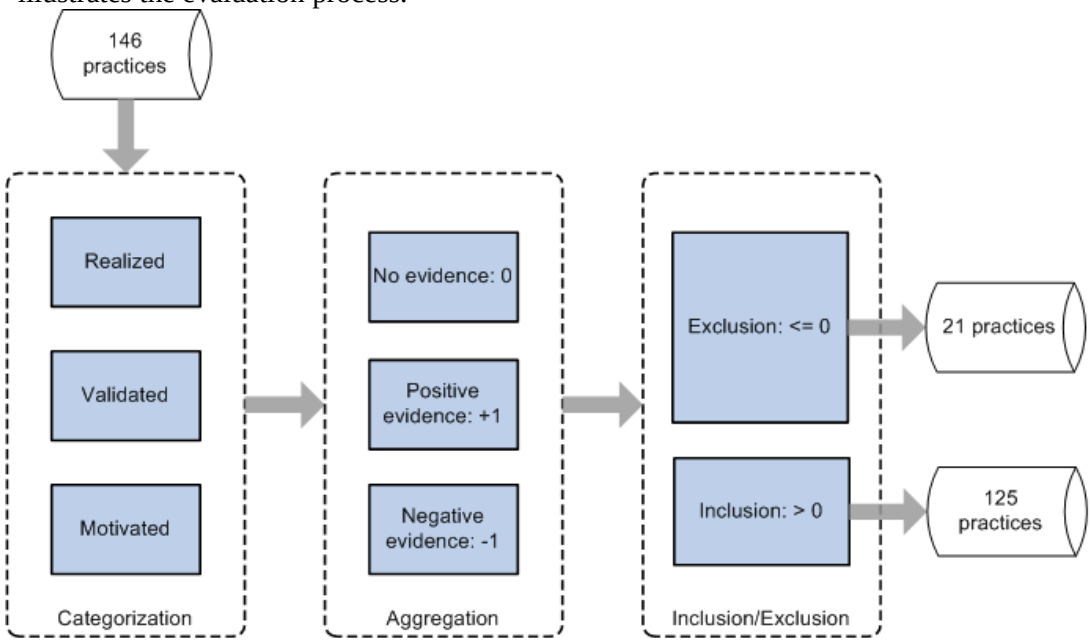

Figure 10. Evaluation process to detect "good practices"

There are 3 steps in the evaluation process.

- **Step 1 – Categorization**: in this step, 146 identified practices were classified into 3 groups.
    - Practices which were realized in form of concrete tools were included in the "***Realized***" group.
    - Practices which were validated in any context (including industrial settings, academic projects or static validation) were filtered into "***Validated***" group.
    - The rest of the practices were then gathered into "***Motivated***" group.

  One important note is that the order of the group implies the priorities of such group. Hence, if a practice was validated in some context, it would be classified into "***Validated***" group and the practice would be analyzed in the next step using solely the validation findings (other motivation would be discarded). The reason is that, result when applying one practice in certain context is considered more creditable to evaluate such practice in compare to rationale. Hence, it is reasonable to discard the motivation under certain practice once the validation result of it was reported.
- **Step 2 – Aggregation**: Once practices were categorized into three groups, their credibility was then investigated. The judgment was made within each group by studying each practice and the publication(s) in which it was proposed.
    - **"Realized" group**: Practices in this group were all realized in form of concrete tools hence receive +1 point for each.
    - **"Validated" group:** Practices in this group receive 0 point if no information regarding validation context was reported. Practices with provided validation context and explicit positive findings receive +1 point. Practices with provided validation context and explicit negative findings receive -1 point
    - **"Validated" group**: The researchers investigated the motivation under the practices and their validation details. Practices with no motivation supported receive 0 point. Practices with positive motivation supported by rationale using empirical data, professional knowledge or industrial

experience receive +1 point. Practices mentioned with negative motivation (i.e. it is not beneficial to perform such practices) receive -1 point. In some cases, it was quite hard to extract the information since the author provided very little information about the data source he used. In order to solve this problem, quality information was taken into account. For instance, if the paper stated that empirical data was used to propose solutions but it was not described and no validation was conducted, the motivation for such solutions would not be considered. In the end, if no evidence for the credibility of a practice was found, such practice then received 0 point.

- **Step 3 – Inclusion/ Exclusion**: Once the aggregation step was performed, the results would be synthesized. Practices from "Realized" and "Validated" groups with positive grades are considered with high credibility and included in *the Highly Applicable "good practice"* set. . Practices from "Motivated" group with positive grades are included in *the Applicable "good practice"* set. The rest of the practices including those with zero or minus point were discarded.

This analysis returned 125 practices "good practices" for MDRE including 9 practices from "Realized" group, 90 practices from "Validated" group and 26 practices in "Motivated" group. This implies that they are either supported by rationale or validated in industrial context and encourage by empirical evidence hence they are considered as "good practices" for the answer of question RQ.1.3. The details of these practices can be found in Appendix E.

The researcher then was interested in having a deeper look at the applicability of the validated practices. Therefore, another study was performed to assess the applicability of those practices by judging whether the practice is validated in industrial company(s) or simulated environment by case study, interview or survey and so on, or is used in real work by industrial company(s). This judgment has uncovered that, out of 90 validated practices:

- 80 practices were validated in industrial organizations.
- 9 practices were checked against experts' opinion in academia and later validated in industry
- 1 practice was validated in academia (through experiments with PhD and Master students)

Promisingly, from those validated in industrial settings, 25% of them (20 practices) were actually in use and 25% of them were validated in more than 1 company. Although none of the practices was extracted from quantitative research, this shows a very promising future of transferring these practices from research to industries. Therefore, we consider all of these practices as ***highly applicable*** "good practices" and will use them as inputs for the model construction.

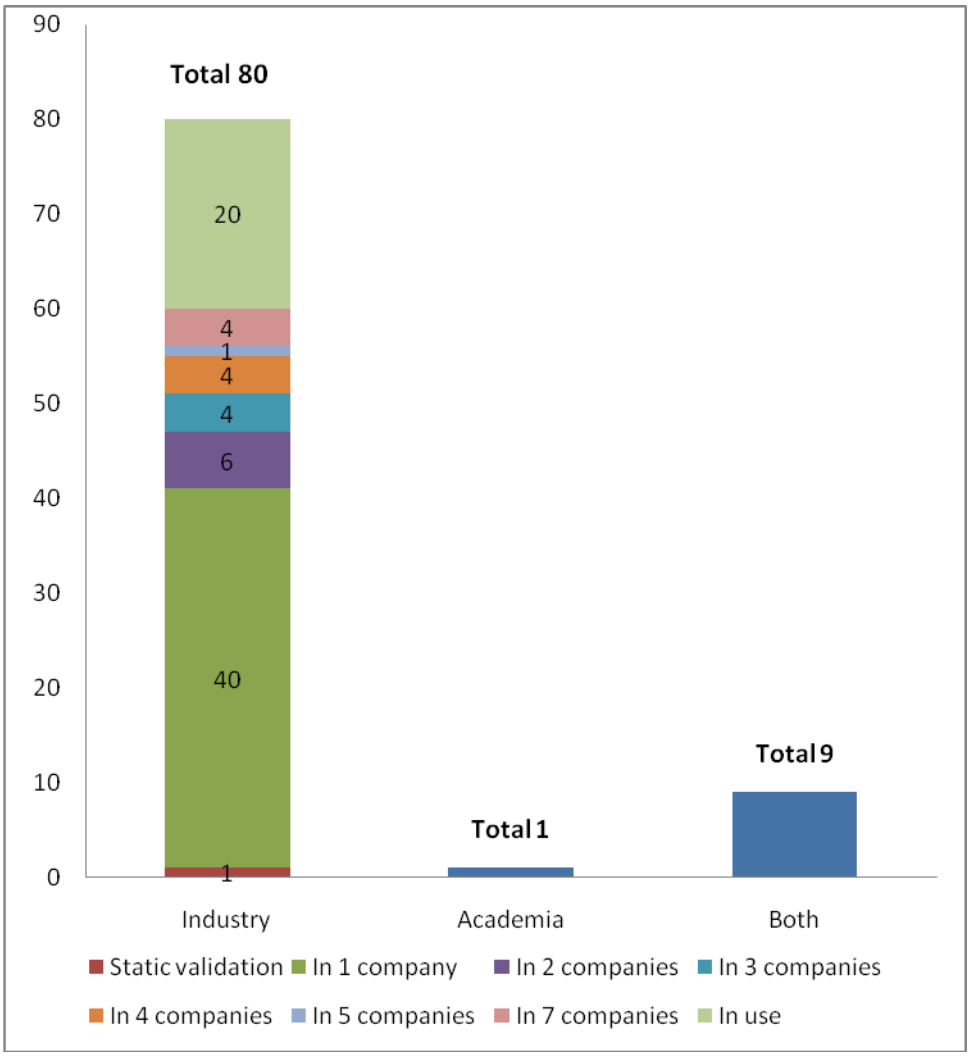

Figure 11. Distribution of practices according to validation context

b. Summary

Overall, the researchers analyzed and extracted 125 "good practices" from 48 papers in the relevant research area. They were considered to be "good practices" given the reason that they were either in-use or validated, or well-motivated by empirical data or rationale. They presented a highly transferable set of actions that might benefit practitioners and contribute promising initiatives for the analysis of this study in the next steps. 21 practices having no support were excluded from this study. They might be good practices but at the moment, their credibility was yet proven; hence they were not be used for developing the new model.

3.4.3 Mapping the indentified practices to the acknowledged challenges in MDRE

3.4.3.1 No fixed customer

In MDRE, the customer role is usually not clearly determined [17]. This poses a significant challenge to organizations which adopt market-driven development approach since it is not easy to define from which sources requirements can be elicited and how to eliminate the vagueness of the requirements. Several practices found in this systematic review aim to support practitioners to overcome this challenge.

- Use business concerns to drive requirements elicitation
- Involvement through Incident Reports, Idea Feedback, Suggestions

- Use prototyping to animate requirements

Firstly, it is necessary to consider different sources of requirements in the case of MDRE since there is no sole right answer from a specific source. Apart from surveys carried out on customers, it is advantageous to consider other sources such as Incident Reports, Idea Feedback and other Suggestions from domain experts etc. In addition, considering that sources of requirements in MDRE are diverse, using business concerns as a driven-factor will help the practitioners to focus appropriate efforts on the necessary sources of requirements. It is also beneficial to use prototype as another mean to realize requirements in a visible form to get more requirements from feedbacks.

3.4.3.2 Requirements Specification

Several problems for capturing and documenting requirements using the traditional document in MDRE have been reported [61]:

- It is challenging to store the attributes of the requirements in MDRE
- It is very hard for the diverse requirements sources to post their requirements/expectations
- Keeping track on requirements status is extremely challenging considering the inclusion and exclusion steps in release planning.

Based on the systematic review results, the following activities are suggested to overcome the aforementioned issues.

- Collect/extract requirements and save to repository
- Ensure requirement and project information are online accessible and traceable
- Define a Standard Document Structure
- Define requirement attributes: e.g. priority, needed effort
- Record rejected requirements
- Use standard templates for describing requirements
- Describe the environment from which the requirement originates (source of requirement)
- Document the rationale for the requirements.
- Specify requirement creation date
- Specify requirement due date
- Specify requirement last change date
- Specify requirement version
- Specify requirements manager
- Specify requirements owner
- Specify Requirements resource
- Note Requirement state

By using tool support and repository, it is easier for involved members to access the information of requirements from a synchronized database. Storing information on source, date time and state of requirements helps practitioners to keep track on different versions of requirements and enable then to trace backward and forward when change occurs. Moreover, by recording rejected requirements together with the rejection rationale, it is possible to handle different sets of requirements during release planning.

3.4.3.3 Continuous requirement flow

Making use of the requirements repository, the MDRE development faces another challenge which is overloading due to the continuous requirement flow from a vast number of stakeholders [11][62]. It is challenging to manage this repository in order to maintain the balance state of it so as it could be used effectively in prioritization step.

The diversity of the stakeholders and the continuous requirement flow also may introduce another challenge which is the duplication in requirements [63][64].

In addition, it is a fact that working in an environment with many stakeholders would be much harder than working with a specific customer since they introduce more changes [11]. [11] showed that there are a number of factors impacting the requirements in MDRE such as changing market demands, rougher competition and key customers that are usually not clear of their expectation. These factors are not available in Bespoke as the customer and their requirements are often fixed or lightly changed. This raises a challenge in MDRE which is how to deal with these volatilities of the market.

Suggested actions:

- Perform requirements triage
- Structuring the requirements into a hierarchy, abstraction levels
- Align requirements with the organization's product strategies
- Use a tool to manage requirements
- Collect/extract requirements and save to repository
- Continuously manage change throughout design, implementation and verification

Early dismissing the raw requirements would help practitioners to eliminate the huge income requirement flow hence effectively focus effort on the "right" requirements. Besides, by structuring the requirements into hierarchical levels, the abstract requirements which need to be broken down into details can be detected hence the volatile of requirements will be significantly reduced.

3.4.3.4 Gap between Marketing and Development

A study of Karlsson [11] has identified a gap between marketing and developing department in MDRE. The gap existed due to the different perspectives and understandings in defining a "good" requirement [11]. In the study, the marketing department considered "good" based on the financial benefit whereas the developers understood based on some criteria such as: understandability and clearness for implementation. The study mainly showed that this gap would be a collaborative challenge for MDRE development.

The following actions are suggested to be performed in order to overcome these issues.

- Ensure requirement and project information are online accessible and traceable
- Assure dependable portfolio visibility and release implementation
- Maintain the communication of requirement rationale in organization of the priority rationales through the organization
- Establish direct link between stakeholders and the developers
- Obtain common understanding of requirements among different involving teams
- Emphasize whole-product thinking

By assuring that the requirement information is online accessible, the involved members are always able to retrieve necessary information which is centralized and synchronized. It is also important that the members are aware of the business concerns of the product and communication between different stakeholders and involved teams are maintained. Based on that, the common understanding on requirements will be established and gaps between marketing and development could be reduced.

3.4.3.5 Release Planning

In fact, release planning relies mostly on estimates [17]. However, due to the diversity of stakeholders and the volatility nature, it is challenging to perform accurate

estimates for the project. Both underestimation and overestimation result in failures in development and may propagate in next releases

In addition, due to the characteristic of the MDRE in which the requirements are continuously growing, it is very difficult for prioritization [65]. This challenge is even burdened by the fact that the requirements volume in MDRE is usually very large. Selecting the optimal subset among a vast number of requirements requires making use of an effective technique. Besides, the diversity of the stakeholders also makes the prioritization more challenging according to the different importance of the stakeholders [66][67].

- Make use of must- and wish-lists
- Publish launching preparation package
- Release definition
- Release validation
- Involve Stakeholder by Voting
- Perform Pre and post traceability
- Post requirements selection evaluation
- Explicate planning levels and time horizons
- Plan for more than 1 release ahead
- Plan open-endedly with a pre-defined rhythm
- Requirements selection
- Perform prioritization
- Review all the requirements including those that weren't selected for the last release

3.5 Conclusion

This chapter presents the synthesis of a systematic review on market-driven RE practices. During the review, together with the explored practices, all information related to such practices including rationale, validation methodologies and context, and so on were also analyzed. The synthesized data provides a state-of-the-art of all proposed activities in the area. The main findings of this review contain:

- A collection of 137 practices which are explicitly suggested as beneficial to be performed in MDRE has been assembled. These activities are distributed broadly in various sub-areas of RE (e.g. Requirements Management, Requirements Elicitation, Requirements Analysis, etc).
- A collection of 16 practices which are generalized from different tools and supporting artifacts has also been identified. These activities are not explicitly mentioned in research; however, supporting artifacts are the evidences to convince the applicability of such practices.
- A detailed picture of the applicability of MDRE practices was also depicted giving a good view for practitioners about MDRE process. Based on that, a list of 125 "good practices" for MDRE which are sufficiently encouraged by empirical validation or rationale was derived. This list then served as the input for creating Uni-REPM in the chapter 5.
- A list of solutions for identified challenges in MDRE was also provided based on the captured "good practices". This list is a potential starting for researchers to dig deeper into the area.

In addition, it is also interesting to see that a large portion of the obtained practices are very up-to-date. Furthermore, most of them were validated in industrial settings and proposed on the basis of empirical context. This improves the applicability of the practices given the reason that their needs are triggered from practical work. Moreover, a number of models, techniques and frameworks were constructed to support practitioners in RE. This shows a highly concern about the transferability of the research work in this area.

Besides, it was a great experience for the researcher to process this study from planning to execution and reporting. The designated plan has shown its validity given the result that the systematic review could successfully uncover a significant number of "good practices" in literature. The piloting studies were proven to be useful in the review. The discussions during the piloting studies were very helpful for the two researchers to align their understandings in the review process.

4 LITERATURE REVIEW

4.1 The need of Literature review

This part presents the literature review results performed on CMMI_DEV [21], TickIT [22] and REPM model version 1.0 [23] in order to ensure that all the up-to-date good practices in bespoke RE were captured for the construction of Uni-REPM model. This is due to the fact that both CMMI_DEV and TickIt focus on the traditional customer-supplier relationship. By identifying new practices in CMMI_DEV and TickIt that were not covered in REPM model version 1.0 as well as updating existing actions with new information, the review process provided an additional set of actions that complements the current one. As CMMI_DEV and TickIt are the huge models that operate at organizational level, the researcher anticipated to obtain more practices in Requirements Process Management area to enhance it.

The literature review performed on CMMI_DEV focused on the two directly related process areas "Requirements Development" and "Requirements Management" as well as on the indirect area "Configuration Management". On the other hand, the literature review on TickIt was performed by studying several sections such as "Requirements Management", "Configuration Management" under "Supplier" perspective.

4.2 Literature review results

The literature review was conducted by extracting practices from CMMI_DEV and TickIt thereafter comparing those with the 68 practices extracted from REPM. While reviewing the practices in CMMI_DEV and TickIt, the researcher revealed that some of them were mentioned in both of the models. Therefore, a new action that represented both was created. Moreover, for those practices that have similar goals, they were combined into a single action. The decision whether it is possible to combine those practices was made on a case by case basis, taking into account the suitability of the actions to the REPM model version 1.0. The new actions were placed under the related Main Process Area (MPA) and assigned certain maturity levels to ensure the level homogeneity. On the other hand, updated actions were kept at the same level and same Main Process Area but their name were modified to reflect the new information. The following sections showed the list of new and updated actions.

4.2.1 Additional actions

There are 16 newly created actions added into the REPM model version 1.0. One action is under Elicitation, 2 are under Analysis and the rest belong to Requirements Process Management. This result reflected an agreement with the anticipation above. Table XV shows the list of these actions stemmed from the practices found in CMMI_DEV and TickIt.

TABLE XV. Additional actions

CMMI_DEV practice	TickIt practice	New action	Level	MPA
Engage relevant stakeholders using methods for eliciting needs, expectations, constraints, and external interfaces.		Elicit stakeholders' requirements using different methods	2	Elicitation
Analyze stakeholder needs, expectations, constraints, and external interfaces to remove conflicts and to organize into related subjects.		Analyze requirements for consistency	1	Analysis
Analyze requirements to ensure that they are complete, feasible, realizable, and verifiable.	Analyze the requirements in term of correctness and testability	Analyze requirements for completeness, correctness, feasibility and testability	1	Analysis
Establish and maintain relationships between requirements for consideration during change management and requirements allocation.		Create and maintain requirement's relation for traceability	2	Requirements Process Management
Explore the adequacy and completeness of requirements by obtaining feedback about them from relevant stakeholders.		Validate requirement's completeness and adequacy with stakeholders	3	Requirements Process Management
Assign responsibility and authority for performing the process, developing the work products, and providing the services of the requirements ***development*** process.		Assign responsibilities in requirements development and management processes	3	Requirements Process Management
Assign responsibility and authority for performing the process, developing the work products, and providing the services of the requirements ***management*** process.				
Train the people performing or supporting the requirements ***development*** process as needed.		Train people involving in requirements development and management process	4	Requirements Process Management
Train the people performing or supporting the requirements ***management*** process as needed.				

Develop an understanding with the requirements providers on the meaning of the requirements.	Establish effective communication with customers to agree upon the interpretation of the sys spec	Ensure having the same interpretation of the requirements as the requirement providers	2	Requirements Process Management
	Ensure requirements consistency exists between requirements elicitation and analysis	Establish a common understanding of requirements between different stages	4	Requirements Process Management
Manage changes to the requirements as they evolve during the project.		Install and maintain a process for requirements change management	3	Requirements Process Management
Establish and maintain a configuration management and change management system for controlling work products.	Operate a change control mechanism for sys requirements			
Identify and involve the relevant stakeholders of the requirements ***development*** process as planned.		Ensure relevant stakeholder involvement in requirements development and management processes	3	Requirements Process Management
Identify and involve the relevant stakeholders of the requirements ***management*** process as planned.				
Establish and maintain the description of a defined requirements ***development*** process.	Establish procedures for eliciting and developing system requirements spec to agreed baseline	Establish and maintain defined processes for developing and managing requirements	3	Requirements Process Management
Establish and maintain the description of a defined requirements ***management*** process.				
Track the status of change requests to closure.	Have a mechanism to ensure customers can easily and accurately determine the status and disposition of their change requests	Define a mechanism to track change request	3	Requirements Process Management
Document the set of configuration items that are contained in a baseline.		Create a baseline	3	Requirements Process Management
Maintain bidirectional traceability among the requirements and work products.		Consider change impact between requirements and other work products	2	Requirements Process Management

Specify the important characteristics of each configuration item		Specify the requirement's important characteristics	2	Requirements Process Management

4.2.2 Updated actions

Besides the newly created actions, there are three actions that were updated from the literature review process, namely “Scenario Elicitation”, ”Requirements Classifications” and “Prioritizing Requirements” . Table XVI presents the details of these actions.

TABLE XVI. Updated actions in REPM

Existing action	CMMI_DEV	Updated action
Scenario Elicitation	Analyze operational concepts and scenarios to refine the customer needs, constraints, and interfaces and to discover new requirements.	Use scenario to elicit and analyze requirements
Requirements Classification	Partition requirements into groups, based on established criteria (e.g., similar functionality, performance, or coupling), to facilitate and focus the requirements analysis.	Classify requirements into groups based on defined criteria to enhance analysis
Prioritizing Requirements	Identify key requirements that have a strong influence on cost, schedule, functionality, risk, or performance.	Prioritizing Requirements based on cost, functionality, risk or performance

4.3 Summary

The literature review process yielded a set of 84 practices consists of 68 practices derived from REPM with 3 updated ones and 16 additional actions. By complementing the original REMP model with the actions from CMMI-DEV and TickIt, the researcher is certain that a well-rounded and up-to-date set of good practices in bespoke RE has been captured to base the construction of the universal light-weight RE process maturity assessment model on.

5 UNI-REPM MODEL CONSTRUCTION

5.1 Construction methodology and objectives

The aim of Uni-REPM is to serve as a universal light-weight model presenting the maturity of RE process through sets of necessary activities. Besides the assessment purpose, this Uni-REPM model is also expected to function as a guideline giving organizations a recommended improvement path toward a better RE process from basic to advance level.

The approach of this construction is lightly driven by the technology transfer framework created by Gorschek [25] in which the candidate solution is created through the study of state of the art. Therefore, the new RE process assessing model is developed based on the RE good practices extracted from a literature view on bespoke RE and those discovered from an extensive systematic review on market-driven RE researches.

Figure 12 illustrates the process to create the new Uni-REPM model.

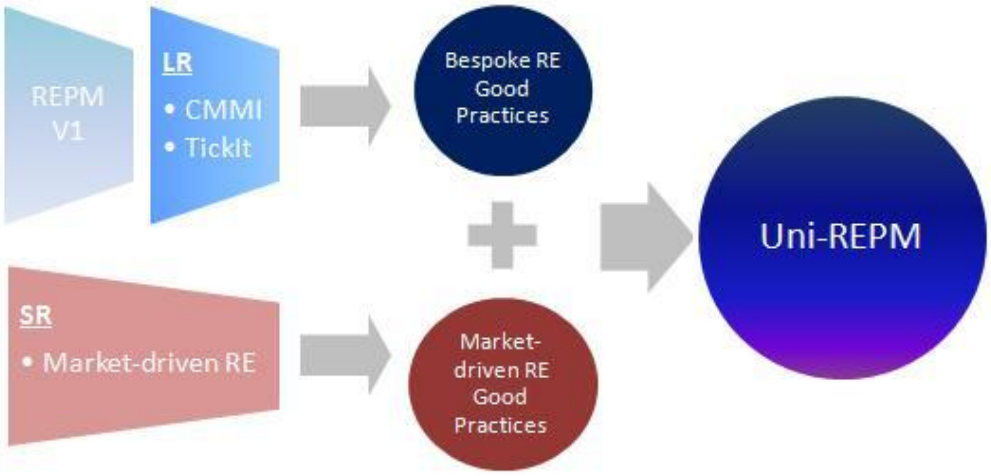

Figure 12. Uni-REPM creation process

There are several objectives which were kept in mind during the model construction stage:

- ***The feasibility of the activities in the model***: In order to ensure the applicability and the usefulness of the model in industry, the feasibility of the captured actions was highly regarded. Actions retrieved from the REPM and Literature review on CMMI_DEV/TickIt were considered as applicable given that the three models are validated in industrial settings. Actions extracted from the systematic review were strictly judged regarding their validation and rationale provided in the research.
- ***The universality of the activities in the model***: During the model creation process, all the possible combinations from two aforementioned reviews were performed in order to maximize the universality of the actions in the new model.
- ***The light-weight characteristic of the model***: the light-weight characteristic of Uni-REPM was realized by the following implementations:
    - The model checklist: This checklist is a transformation of Uni-REPM in the form of questionnaire where practitioners can easily answer and summarize to calculate the results. It is expected to provide industrial organizations a quick, simple and low-cost method to identify the strengths and weaknesses in their process.

- The model structure: the actions in the model were classified into groups and layers according to their purposes so as the practitioners can locate problems without difficulty.
- The model content: all the actions in the model were provided with their objectives and detailed description. When necessary, referenced links to supporting artifacts were also attached in the content of the actions. Hence, Uni-REPM is deemed to require minor time to learn and to use.

- ***The twofold purpose of the model***: In order to achieve this goal of the researcher, Uni-REPM was structured in two views, namely Process Area view and Level View, and equipped with both maturity level and implementation description.

5.2 Maturity Measurement

Uni-REPM makes use of an ordinal scale to assess the maturity of the RE process. This scale is inspired by the REPM [23] and GPG [43] models; and the book "Software requirements" [61]. Concerning the fact that RE Process applied in industry is usually a small-sized and ad-hoc one (compare to the others) [23][61][68], three levels of maturity has been defined, namely Departure, Inter-mediate and Destination. The reason for changing from five levels as in REPM model [23] to three levels is to make the RE process significantly better after completing each level. Hence, the benefits gained from moving from one level to another level are more visible as well. These levels will present how mature the evaluated process is. It is, however, not applicable to the whole organization maturity since the model scope only resides on RE Process. Nevertheless, it is possible to compare two processes in term of maturity using the evaluation results from the model.

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

Level 1 – Departure

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

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

Level 2 – Intermediate

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

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

Level 3 – Destination

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

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

5.3 Model structure

The construction of Uni-REPM structure was based on the original model REPM given the reason that it was validated in industrial settings. The model hierarchy has three levels, namely ***Main process area*** (MPA), ***Sub-process area*** (SPA) and ***Action***. On the top level of the model, there are seven MPAs corresponding to RE main activities. Each MPA is further broken down into several SPAs which cover closely related Actions. On the bottom level, Action denotes a certain activity that should be done or a certain item that should be present. A MPA may also have action(s) attached directly to it. An Action must attach to one MPA or SPA and there is no MPA or SPA residing under an action. Figure 13 illustrates the hierarchical structure of the model.

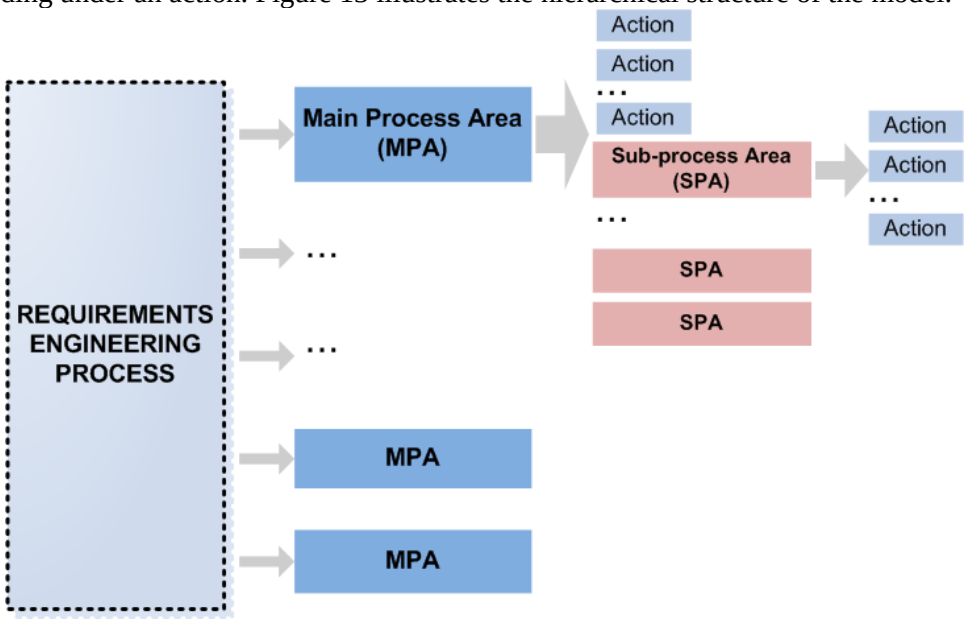

Figure 13. Uni-REPM model structure

5.3.1 Main Process Area (MPA)

On the top level of the model, a Main Process Area (MPA) represents a cluster of related practices in RE.

There are 7 MPAs in the model, namely Organizational Support, Requirements Process Management, Requirements Elicitation, Requirements Analysis (and Negotiation), Release Planning, Documentation and Requirements Specification, and Quality Assurance. The details of these MPAs are described in section 4 (Model Content).

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

5.3.2 Sub-Process Area (SPA)

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

5.3.3 Action

The smallest unit in the model is called "Action" showing a specific good practice. By performing the Actions, the organization can improve their process and gain certain benefits. For example, an action "Create a product-wide glossary of terms" once implemented will enable readers from different backgrounds to understand the technical jargons used in a requirements document.

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

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

Recommendation(s) and Supporting Action(s)

Within the description of each Action, there can be Recommendation(s) and Supporting Action(s). The idea of Recommendation(s) is to give practitioners suggestions on proven techniques or supporting tools. This information aims to help the practitioners with implementing the action. In addition, the Supporting Action(s) provided links to other Actions which will benefit the practitioners when implementing them together.

Optional Group Actions

In some cases, there are several ways to achieve the same goal/benefits but to different extent. Hence, actions that represent these different approaches can be grouped into an optional group, denoted by "OG". The optional group actions can be identified by "OG" followed by the group number and their position in the group. For example "OG1.a1" points to the option one in the first optional group.

In order for a project's RE process to qualify for a certain REPM Level, at least one action having the same level in the group has to be satisfied. In this case, there are four actions in the optional group, namely ***QA.a1.Use Checklist to Ensure Quality of Requirements (OG1.a1)*** at Level 1, ***QA.a3.Review Requirements (OG1.a2)*** at Level 2, ***QA.a4.Create Preliminary Artifacts for Quality Assurance (OG1.a3)*** at Level 3 and ***QA.a5.Organize Inspections to Ensure Quality Requirements (OG1.a4)*** at Level 3. In order for the process to complete Level 3, at least one action OG1.a3 or OG1.a4 has to be fulfilled. It does not matter if OG1.a1 or OG1.a2 is done or not. If the process wants to reach level 2, at least OG1.a2 has to be fulfilled.

5.4 Model content

The content of the model was constructed by the following steps:

- Determine the Main Process Areas of the model
- Determine the Actions of the model
- Specify levels for Actions in the model
- Gather closely related Actions into SPA(s).

5.4.1 Determine the MPAs of the model

The seven following Main Process Areas are proposed in the model based the literature review on [17]. The rationale under this structure is to ensure the model covering adequately all the activities in RE as well as to improve its understandability.

- **Organizational Support**: This main process area covers supporting activities given to RE process from the surrounding organization.
- **Requirements Process Management**: The requirements process management covers all the activities to manage, control requirements change as well as to ensure the organization of the process and coherence among team members.
- **Requirements Elicitation:** Elicitation consists of activities those are for discovering, understanding, anticipating and forecasting the needs and wants of the potential stakeholders in order to convey this information to the system developers.
- **Requirements Analysis (and Negotiation)**: Requirements after gather from different sources need to be analyzed to detect incomplete or incorrect ones as

well as to estimate necessary information for later activities (eg. risk, priorities…).

- **Release Planning:** Release planning covers crucial activities aiming to determine the optimal set of requirements for a certain release to be implemented at a defined/estimated time and cost to achieve some goals.
- **Documentation and Requirements Specification:** This MPA deals with how a company organizes requirements and other knowledge gathered during RE process into consistent, accessible and reviewable documents.
- **Quality Assurance:** This process involves activities checking the requirements against defined quality standards and the real needs of various stakeholders. It ensures that the documented requirements are complete, correct, consistent, and unambiguous.

5.4.2 Determine the Actions of the model

This step aims to analyze and combine the practices from previous study to propose Actions for the new model. RE practices which were derived from the Systematic Review and extracted from the updated REPM were categorized into the seven aforementioned MPAs. The investigation and combination were then performed in each MPA following the three steps presented in Figure 14.

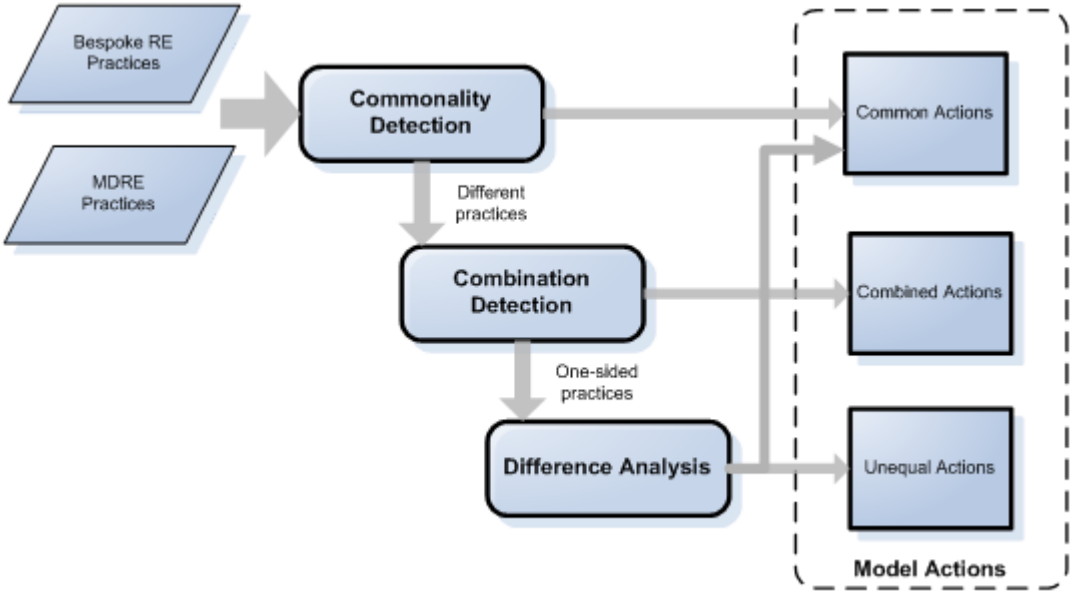

Figure 14. Practices combination process

- **Commonality Detection**: Firstly, the practices were analyzed to detect those belong to both bespoke and market-driven RE and perform identical activities or tasks. These practices are obviously applicable in both cases, hence should be put in the new model.
- **Combination Detsection**: Remaining different practices were then analyzed here in order to detect the possible combinations. Practices from the two lists which had similar goals were grouped together. Among those practices, some could complement each other to achieve a certain goal. Others might present different approaches to complete the same task. Hence, for each group, the practices were checked against the following criteria for combination detection:
    - CONFORM: Actions might differ from each other but together they conformed to certain goals. These practices were merged together to introduce a new action in the model.
    - GENERALIZE: Actions described various approaches to perform certain task. They were then generalized to a common action which covered all cases.

    - SUPPORT: Actions aimed to specify other actions. They were added to description of the new actions in the new model
- **Difference Analysis**: Practices that were found only in one environment either in bespoke or market-driven would be considered as one-sided practices. They might be applicable in both cases but were not confirmed in research. They might also be necessary activities that were applicable solely in one case. Hence, these practices were investigated by the following criteria to determine their necessity in the new model:
    - ESSENTIAL: The practices remained in the new model if they were essential for a specific case and appeared in the **Highly Applicable** group (retrieved from Systematic review result) even though they were inapplicable universally. Hence, they were included in the new model. A typical example of this category is the MPA Release Planning.
    - APPLICABLE: The practices were proposed in only one of the two environments (bespoke or market-driven RE), but they were applicable for both cases. Therefore, they were generalized to Common Actions and contributed to Actions in the new model.

One-sided practices with similar goal were also joined using the criteria in previous step (Combination Detection).

The result of this analysis including Common Actions, Combined Actions and Different Actions according to MPAs can be found in the Appendix F together with the criteria used.

5.4.3 Overlapping actions between Bespoke and Market-driven RE

The process to generate actions for Uni-REPM has also uncovered that 52% of the identified actions in the new model fell in the overlapping region between Bespoke and market-driven RE (including 14% common actions and 28% combined actions which are applicable for both approaches). This has shown that, apart from the particular activities in MDRE, a significant amount of the practices in such area was found in common with Bespoke practices during this study. This confirms the goal of the researcher which is not to add a separated part into the existing solution but to look for an integration solution for assessing RE process maturity in various situations. Figure 15 illustrates the details of the aforementioned overlap. In this figure, the common actions area denotes the number of actions which emerge explicitly in both Bespoke and market-driven RE. The combined actions area demonstrates the number of activities appearing with the same goals in both approaches. Those were then integrated into single practices. The green area of the columns presents the different activities between two approaches. In this figure, the particular area Release Planning is not calculated since it is considered specifically for MDRE.

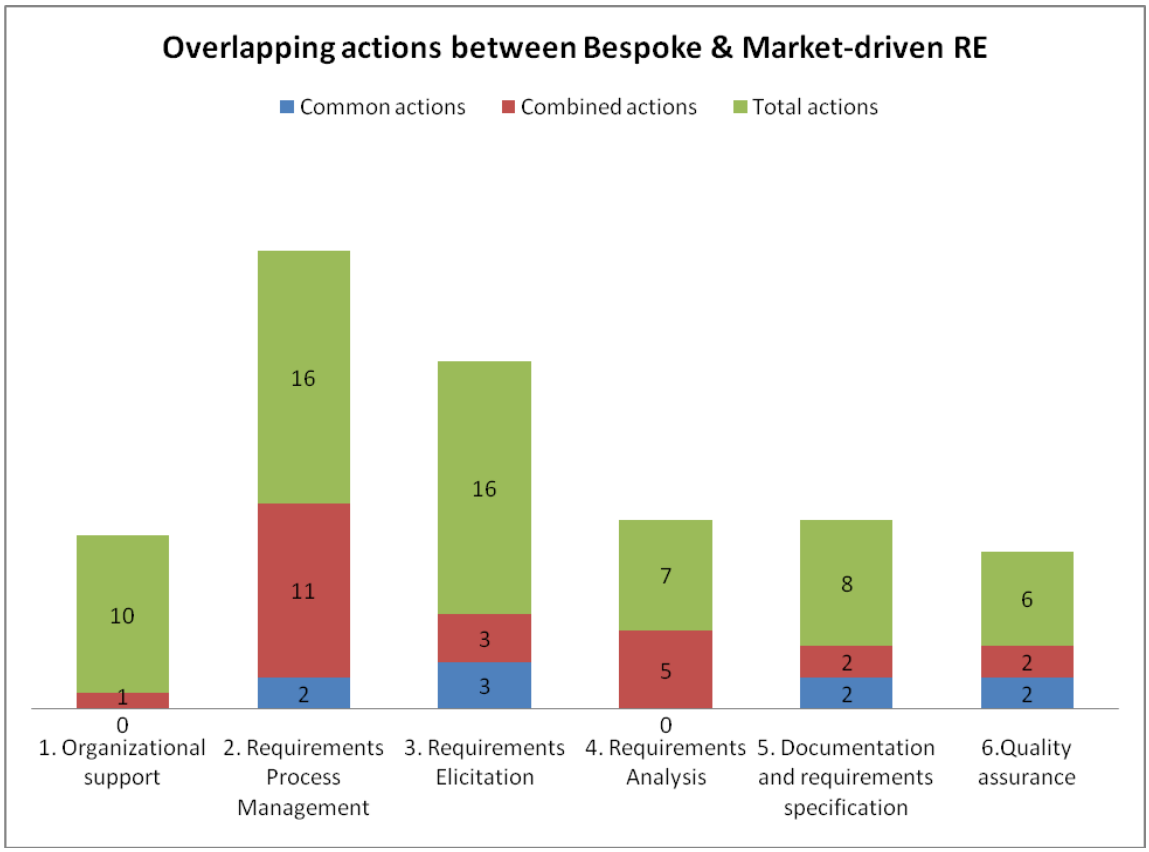

Figure 15. Overlapping actions between Bespoke and market-driven RE

5.4.4 Specify levels for Actions in the model

The generated Actions from the previous step were then analyzed to determine at which level each individual should be placed. The decision was made by examining the essentiality and complexity to perform certain Action. The levels of original actions (extracted from the REPM model) were also used as reference for this assessment. In addition, the dependencies among Actions were highly considered in order to ensure the consistency and the coherence between different levels.

5.4.5 Gather closely related Actions into SPA(s)

The closely related Actions in each MPA were then grouped into SPA(s) to give readers a better understanding of the new model. Table XVII demonstrates SPAs for each MPA together with their goals.

TABLE XVII. SPAs list according to MPAs

SPA name	SPA description
Organizational Support	
Roles and Responsibilities	The activities under this SPA aim to define the roles, and their responsibilities, that deal with different aspects of requirements engineering.
Strategic	This SPA contains practices regarding strategic decisions and/or practices influence the requirements engineering process, especially in a market-driven product.
Requirements Process Management	
Configuration Management	This SPA consists of activities for managing the configuration of them to support traceability and avoid confusion.
Requirements Communication	This SPA covers activities those are to ensure the coherence between teams and team members.
Requirements Traceability Policy	Activities under this SPA are performed in order to ensure the consistency of the system when changes occur and trace from requirements to other artifacts of the project and vice versa to apply necessary changes.
Requirements Elicitation	
Stakeholder and Requirements Source Identification	Activities under this SPA aim to identify whom the practitioner would like to listen to and which source of information he can look into in order to elicit requirements for the system.
Domain Consideration and Knowledge	This SPA consists of activities covering different types of knowledge that are necessary to be aware of during elicitation.
Elicitation Practices	This sub-process area focuses on the overall framework under which the specific elicitation practices are conducted.
Release Planning	
Selection	This SPA covers actions supporting requirements selection and release definition.
Documentation and Requirements Specification	
Documentation Deliverables	The activities under this SPA are to define the expected deliverables of RE process at the beginning of it as requirements.

5.4.6 Model checklist

The idea of a model checklist is derived from the original model REPM. It is to transfer the assessment into an easy form for practitioners. The checklist keeps the model structure with seven MPAs and SPAs and Actions. For each action, a question is composed to detect whether the corresponding action was performed in the process. For each question, the practitioners can choose one of three answers: "Complete", "Incomplete", or "Satisfied/Explained".

The reason of the option "*Satisfied/Explained*" is that, in reality, some of the actions are found out not necessary to be performed in particular situations of organizations. For example, a company has an internal glossary of terms which is explained directly to the users in the meeting. It was more effective this way than releasing it to the users as suggested in action "OS.GA.a1 Create a Product-wide Glossary of Terms (Basic Level)". In this case, this action is not useful for them. If we consider it as "Incomplete", the process may not reach the Basic level because not all actions in this level are fulfilled. This is even more unfair if all other actions in higher maturity levels are completed. Therefore, companies should not be "punished" if they

do not perform a certain nonessential action (in their point of view). In order to take into account this factor, the option "*Satisfied/Explained*" is devised. In this way, the model is more fitting to the real process and the evaluation result is less distorted. Besides, the differences between two types of development settings (bespoke and market-driven) do exist. Therefore, in some cases, the organizations may find actions inapplicable. The model checklist can be found in Appendix I.

However, it is necessary to mention that satisfying an action is about the opinion of the one being evaluated. There always exists the interpretation of the process by the practitioner during the assessment process. Therefore, on one hand, it is very important to have a person who understands the RE process in the organization deeply in order to achieve the accurate answers. On the other hand, this also introduces a threat where assessment results are influenced by subjective impressions of the appraisers.

5.5 Model Usage

5.5.1 Who will directly use the model?

Uni-REPM aims to assess the RE process maturity; hence it can be used by software practitioners who are involved in the RE process, deeply understand it and are in charge of process improvement in general. They can be (but not limited to):

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

5.5.2 How to use the model?

To assess the maturity of a RE process, the users basically perform a mapping from the actions present in the model to the activities in a real process using the checklist. They could find out one of the following situations:

- The action was deemed vital but was performed partially or not at all in this RE process. It should be marked as **"Incomplete"**
- The action was completed in this RE process. It should be marked as "**Complete"**
- The action was not necessary or possible to be performed in this process. It should be marked as **"Satisfied/Explained"**

In order to use the model as a guideline, the user can simply implement actions following the description in the model. The user may also use the ***Recommendations*** under ***Actions*** which suggest validated useful tools/techniques/methods for particular tasks. It is recommended that the process should finish one level before moving to the next level.

5.5.3 How to read the result?

After mapping all the actions present in the model as described in section 6.1.2, the following rules must be applied to assess the maturity of certain MPAs or the whole RE Process.

- All actions (within a MPA) at a certain level must be **Completed** (or **Satisfied/Explain**) in order for that MPA to achieve such level.
- For the whole process, all actions at a certain level must be **Completed** (or **Satisfied/Explain**) in order for the process to achieve such level.

An example

The result of MPA "Organizational Support" in a RE process after evaluating may look like in Table XVIII.

TABLE XVIII. Assessment result in MPA "Organizational Support"

Level	Actions in real process		Total actions in OS in Uni-REPM
	Completed	Inapplicable	
1	0	0	2
2	3	1	5
3	1	1	2

To have a better view, the result can be presented in graphs as Figure 16.

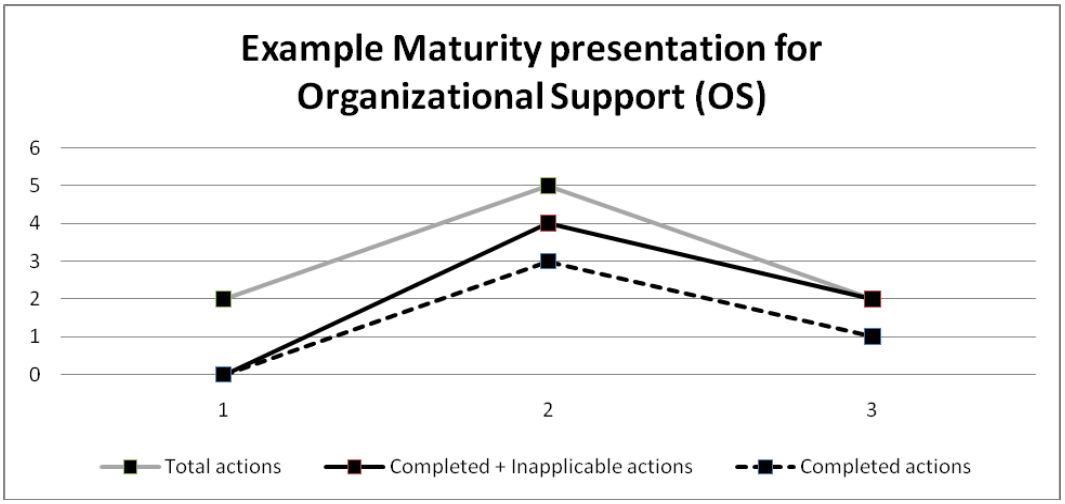

Figure 16. Graphical presentation of assessment result

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

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

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

5.6 Uni-REPM Process Area view

The overview of the Uni-REPM model based on Process area is provided in the following table. The detailed description of Uni-REPM can be found in Appendix G.

TABLE XIX. Uni-REPM Process Area view

ID	Title	Level
OS	**Organizational Support**	
OS.a1	Assign Owner of Requirements Process	1
OS.a2	Create a Product-wide Glossary of Terms	1
OS.RR	Roles and Responsibilities	
OS.RR.a1	Define Roles and Responsibilities for Requirements Engineering Process	2
OS.RR.a2	Define Roles and Responsibilities for Release Planning Activities	2
OS.RR.a3	Define Roles and Responsibilities for Change Control	2
OS.RR.a4	Define Roles and Responsibilities for Product Management Organization	3
OS.S	Strategic	
OS.S.a1	Define Product Strategies	2
OS.S.a2	Define Product Roadmaps	2
OS.S.a3	Define Organizational Strategies	3
OS.S.a4	Communicate Strategies in Organization	3
PM	**Requirements Process Management**	
PM.a1	Introduce Tool Support for Requirements Engineering	1
PM.a2	Define and Maintain a Requirements Management Process	1
PM.a3	Train personnel in Requirements Management Process and Specialty (e.g. Prioritization…)	2
PM.a4	Early connect portfolio considerations into requirements engineering process	3
PM.a5	Involve various perspectives in Requirement Engineering Process	2
PM.CM	Configuration Management	
PM.CM.a1	Manage Versions of Requirements	1
PM.CM.a2	Baseline Requirements	1
PM.CM.a3	Define a Process for Managing Change and Evolution	2
PM.CM.a4	Track change requests	2
PM.RC	Requirements Communication	
PM.RC.a1	Establish effective communication with requirements issuers	1
PM.RC.a2	Obtain common understanding of requirements among different involving teams	3
PM.RT	Requirements Traceability Policy	
PM.RT.a1	Uniquely Identify each Requirement	1
PM.RT.a2	Document Requirements' Source	1
PM.RT.a3	Document Requirements' Relation	2
PM.RT.a4	Document Impact of Requirement on Other Artifacts	2
PM.RT.a5	Define traceability policies	2
RE	**Requirements Elicitation**	
RE.SI	Stakeholder and Requirements Source Identification	
RE.SI.a1	Identify and Involve Relevant Stakeholders	1

RE.SI.a2	Distinguish between Customers, End-Users, and In-house Stakeholders	1
RE.SI.a3	Identify Other Requirements Sources	1
RE.DC	Domain Consideration and Knowledge	
RE.DC.a1	Consider System Domain Restrictions	1
RE.DC.a2	Consider System's Technical Infrastructure	1
RE.DC.a3	Consider Co-existing Business Processes	1
RE.DC.a4	Consider System's Business Process	1
RE.DC.a5	Consider System Boundaries	1
RE.DC.a6	Consider Sociopolitical Influences on Requirements Sources	2
RE.EP	Elicitation Practices	
RE.EP.a1	Adapt Elicitation Technique according to Situation	2
RE.EP.a2	Consider Quality Requirements	2
RE.EP.a3	Create Artifacts to Facilitate Elicitation and Analysis	2
RE.EP.a4	Let Business Concern/Product Strategies guide Focus of Elicitation Efforts	2
RE.EP.a5	Qualify and Quantify Quality Requirements	3
RE.EP.a6	Create Elicitation Channels for Requirements Sources	3
RE.EP.a7	Reuse Requirements	3
RA	**Requirements Analysis (and Negotiation)**	
RA.a1	Analyze for Missing, Double, Incomplete, Ambiguous Requirements	1
RA.a2	Perform Systematic Requirements Prioritization at In-project level	1
RA.a3	Perform Requirements Risk Analysis	2
RA.a4	Analyze for Requirements Functional Dependencies	2
RA.a5	Identify irrelevant requirements for early dismiss (in/out scope OR Triage)	2
RA.a6	Analyze Value-related Dependencies between Requirements	2
RA.a7	Perform refinement and abstraction of requirements	3
RP	**Release Planning**	
RP.a1	Synchronize Release Plan with Product Roadmap	2
RP.a2	Post Requirement Selection Evaluation	3
RP.a3	Plan multiple release at pre-defined interval	3
RP.a4	Involve different perspectives in release planning	2
RP.S	Requirements Selection	
RP.S.a1	Package Requirements into Releases	1
RP.S.a2	Perform Systematic Requirements Prioritization at Pre-project level based on value, cost, effort	2
RP.S.a3	Consider additional factors for prioritization	3
DS	**Documentation and Requirements Specification**	
DS.a1	Define Requirements Attributes	1
DS.a2	Establish Standardized Structure for SRS	1
DS.a3	Define Requirements States	2
DS.a4	Document Requirements Rationale	2

DS.a5	Record Rationale for Rejected Requirements		3
DS.DD	Documentation Deliverables		
DS.DD.a1	Define User Documentation Deliverables		2
DS.DD.a2	Define System Documentation Deliverables		2
DS.DD.a3	Define Management Documentation Deliverables		3
QA	**Quality Assurance**		
QA.a1	Use Checklist to Ensure Quality of Requirements	(OG1.a1)	1
QA.a2	Validate requirements with relevant stakeholders		1
QA.a3	Review Requirements	(OG1.a2)	2
QA.a4	Create Preliminary Artifacts for Quality Assurance	(OG1.a3)	3
QA.a5	Organize Inspections to Ensure Quality of Requirements	(OG1.a4)	3
QA.a6	Use System Model Paraphrasing for QA	(OG1.a5)	3

5.7 Conclusion

In this part of the thesis, the universal RE process maturity model has been introduced together with its creation process. The model is expected to be significantly improved from the grounded model REPM and advanced compared to other available models in assessing RE process maturity given the following characteristics:

- Strong foundation: based on a large set of activities captured from the two previous reviews in RE, the knowledge present in model is expected to cover RE adequately. Hence, the model could be applied extensively in various development environments. Besides, from the view of a guideline tool, the profound basis of the model will aid practitioners in effectively implementing the RE process.
- Light-weight method: As mentioned in section 5.1, during the construction process, this characteristic was greatly considered. Hence, Uni-REPM presents a highly descriptive and structural method for assessing purpose. The model itself is described in different views so as practitioners could easily find the next step to implement. In addition, the model checklist provides a quick access to the measurement result. Hence, Uni-REPM is expected to be easy to use, simple to learn and consume low cost for practitioners.
- Specific solution: based on a profound background, Uni-REPM covers intensively all activities areas in RE. This allows the model to detect even smallest problems in any specific area of RE. Furthermore, it provides practitioners concrete improvement solution by showing what to do to bring the process up to next maturity level.

The mentioned findings have shown a promising Uni-REPM which accomplished the main goal of the researcher: a universal instrument and a concrete guideline in RE.

6 THREATS TO VALIDITY

The main threat to this study is connected to the validity of the proposed Uni-REPM model. Although most of the activities in the model were derived from an extensive systematical review and an intensive literature review which supply strong empirical evidences for the model, the applicability and usefulness of Uni-REPM is yet validated in real work. Hence, the capacity of the model is still unconfirmed. Apart from that, other threats related with the research methodologies and the systematic review implementation are also identified.

6.1 Threats to validity in study methodologies

Traditional literature review was used in this thesis work to explore bespoke RE practices in research. However, reviewing on three chosen sources is not robust since there are considerably more available relevant researches which are not cover. Due to the time limitation, it was not possible for the researcher to perform a systematic review. In the attempt to limit this threat, concern regarding sources of this literature review was carefully analyzed. All the three sources of this review namely CMMI, ISO and REPM were chosen given the reason that they are known and validated in industry. This decision, to some extent, ensures the credibility of the review outcomes.

6.2 Publication bias

Publication bias is the common threat in systematic review in which positive findings tend to be published more than negative ones [15]. In this study, since the validation context and results are main factors for judging a practice, publication bias could leave a significant impact on the study. In order to lessen this threat, the researcher attempted to synthesize validation findings and conclusions from different studies. This was realized by the analysis process described in the conducted systematic review. The credibility of one practice was judged based on the aggregation of studies and not on solely one individual study. Using the analysis results, the researcher expected to overcome the subjective opinion of single study.

6.3 Threats to data selection and extraction consistency

As mentioned in Section 1, the researcher received support from an independent colleague to perform the systematic review in this thesis work. Although the two researchers were aware of the study consistency, it was not possible to perform the data selection and extraction parallelly and cross-check due to the huge amount of identified researches and the limited time frame. However, a two-step process containing piloting and actual steps has been developed in order to eliminate this threat. By performing study pilot, the two researchers could conduct study test and evaluate the test results and the agreement level based on which we could discuss and align our decisions. Although this strategy cannot ensure the consistency as absolutely as the cross-check method, it is an effective and efficient method for achieving both goals: consistency and schedule. Moreover, the two researchers have cooperated in all the assignments and projects in their courses in university. Hence, it is believed that they have built quite good mutual understandings and the gaps between their knowledge and skills are fairly small.

7 CONCLUSION

This section is to summarize the main findings of this thesis work and to list the potential work for the model in the future. It contains three sub-sections. The first sub-section will give an overview on the findings regarding the stipulated research questions. The second sub-section will provide a summary of the contribution of this thesis in the research area. The final part will name the potential work that could be continued from this study in the future.

7.1 Research questions and findings

7.1.1 Research question 1: What are good practices for MDRE?

In order to accurately identify the answer, the question RQ1.1 was broken down in to 3 sub-questions and an extensive systematic review was performed in five known database. The systematic review has shown its success in detecting the good practices for MDRE using defined strategies and process. The review focused on exploring two sets of practices: those are explicitly suggested in MDRE and those could be generalized from supporting artifacts such as models, frameworks, and techniques and so on. In the first set, the review addressed 137 practices proposed from research. Most of the practices were from industrial context. In the second set, 16 practices were derived and most of the supports were validated in industries. Apart from that, since the outcomes of this review served as basis for the new model, an in-depth analysis was performed to evaluate the applicability of detected practices in industries. This investigation aimed to aggregate the validity of the captured practices based on their validation results and a set of predefined criteria. In the end, the whole practices together with their evaluation results were synthesized. 125 practices which were highly justified by empirical data and rationale were identified as "good practices" for MDRE. These practices spread broadly in all process areas. In addition a large portion of these practices were found highly applicable given the reason that they were validated in industrial settings.

7.1.2 Research question 2: What are good practices for bespoke RE?

A traditional literature review was conducted in three chosen sources in order to uncover all the available good practices for bespoke RE. Giving the validation, the success and the popularity, CMMI and ISO TickIT convinced the researcher as the credible sources for this review. REPM on the other hand was specifically tailored for bespoke RE and validated in industrial settings, hence conforms to the goal of the review. In this study, the researcher made use of the actions presented in REPM as the basic set and compared those with practices in Requirements Development and Requirements Management process areas in CMMI and "Requirements Management" and "Configuration Management" under "Supplier" perspective in TickIT. The review returned a set of 94 practices including 68 practices originally retrieved from REPM with 3 updated ones and 16 additional ones.

7.1.3 Research question 3: Based on the answers for RQ1 and RQ2, what major practices could be used to assess the maturity of the RE process?

The answer for this question was formulated in the form of an assessment model namely Uni-REPM. The structure of Uni-REPM is mainly based on REPM with three layers of Main Process Area, Process areas and Sub-process area. The maturity assessment was however modified from five to three levels namely Departure, Intermediate and Destination. In order to compose the content of Uni_REPM, the

knowledge elicited from the two previous reviews was analyzed and synthesized in a practical way in the new model. Practices in Uni-REPM were classified in different process areas and described clearly what to do, what benefits could be achieved and a reference link could be found. During the construction, the researcher attempted to integrate all possible practices belonging to two different development approaches so as the model can address the maturity of RE processes in various scenarios regardless development environment and organization characteristics. In conclusion, Uni-REPM has reflected a detailed picture of available good practices for RE in all development approaches.

7.2 Contribution summary

The main contribution of this study is the universal instrument Uni-REPM for evaluating the RE process maturity in industrial organizations. Based on its profound background and structured construction, the model is deemed to be extensively applicable and useful in industries. The usage of the model is expected significantly advanced compared to existing models such as CMMI and ISO given the fact that Uni-REPM is a quick, simple and low-cost solution for assessment purpose. From inspectors' view, all they have to do is to answer the question list which was generated from Uni-REPM, summarize the assessment results of actions in each MPA and draw the result. This simple process can reduce the huge expense required for appraisers and training. Moreover, the content of the model is systematically constructed based on broad reviews on published researches, hence it convinces the researcher about its intensive coverage in the area. Apart from that, Uni-REPM also functions as a guideline providing practitioners instruction on what to do for certain MPAs in the RE process. From users' view, all they have to do is to read the description of each practice and follow referenced link to perform such action. As a guidance tool, Uni-REPM is believed to effectively transfer the explored knowledge in RE to industrial practitioners hence to some extent reduce the gaps existing between the theoretical and practical worlds.

Apart from the obtained model, the study also provided the whole picture of RE activities through the systematic review and literature review.

7.3 Future work

The study results a promising instrument for assessing the maturity of RE process intensively. However, due to time frame limitation, the question for the applicability and capability of Uni-REPM is still opened. Hence, the following tasks are identified to continue in future:

- Extend the systematic review on a wider range of research (e.g. perform the review on more databases) so that more practices could be identified to improve the model.
- Currently, Uni-REPM is mainly based on theoretical work. This could result in a theoretically perfect model which may lack activities performed in real work. Therefore, it is beneficial to extend the model basis by getting input from industrial practitioners.
- Validate the applicability and usefulness of Uni-REPM in industrial settings. Although it was conducted on a strong empirical study basis, it is advantageous to make use of the model in real work to confirm its purpose. It is worth noticing that this validation work has been started and one can refer to Nguyen's study for further reading [69].
- Develop an automatic tool to support practitioners assessing their RE processes using Uni-REPM. The tool can quickly summarize the appraising results and provide graphical presentation on the maturity of the RE process to the practitioners.

REFERENCES

[1] Sommerville I., Ransom J., "An empirical study of industrial requirements engineering process assessment and improvement", *ACM Transactions on Software Engineering and Methodology (TOSEM)*, vol.14 no.1, pp.85-117, 2005.

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

[3] Hall T., Beecham S., and Rainer A,, "Requirements problems in twelve software companies: an empirical analysis", *Software IEE Proceedings*, vol.149, no.5, pp.153-160, 2002.

[4] Leffingwell D., "Calculating the Return on Investment from More Effective Requirements Management", *American Programmer*, vol.10, no.4, pp.13–1, 1997.

[5] Boehm B.W., Papaccio P.N., "Understanding and controlling software costs", *IEEE transactions on software engineering*, vol.14, no.10, pp.1462-1477, 1988.

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

[7] Nikula U., Sajeniemi J., and Kalvianen H., "A State-of-the-Practice Survey on Requirements Eng. in Small- and Medium-Sized Enterprises", Techical report, Telecom Business Research Centre, Lappeenrata University of Technology, 2000.

[8] Niazi, M. , "An empirical study for the improvement of requirements engineering process", *17th International Conference on Software Engineering and Knowledge Engineering*, Taipei, Taiwan, Republic of China, pp. 396–399, July 14 to 16, 2005

[9] S. Adam, J.Doerr, M.Eisenbarth, "Lessons learned from best practice-oriented process improvement in Requirements Engineering – A glance into current industrial RE application", *Proceedings of the 4th International Workshop on Requirements Engineering Education and Training*, Atlanta, GA, 2009, ISBN 978-1-4244-7695-4.

[10] Neill, C. J., & Laplante, P. A., "Requirements engineering: State of the practice", *IEEE Software*,18(4), 40–45, 2003.

[11] Karlsson L. et al., "Requirements engineering challenges in market-driven software development – An interview study with practitioners", *Information and Software Technology*, Vol. 49, 6, pp. 588-604, 2007.

[12] Gorschek T., Garre P., Larsson S.B.M., and Wohlin C., "Industry evaluation of the requirements abstraction model", *Requirements Engineering*, vol. 12, pp. 163-190, 2007.

[13] Villalon C., Agustin C., Gilabert S., Seco D., Sanchez G., Cota P., "Experiences in the Application of Software Process Improvement in SMES", *Software Quality Journal*, Vol. 10, pp. 261 – 273, 2002.

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

[15] Kitchenham B., Charters S., "Guidelines for performing Systematic Literature Reviews in Software Engineering", Version 2.3, Keele University, *Reviews in Software Engineering Technical Report*, EBSE-2007-01, 2007.

[16] Sawyer P., "Packaged Software: Challenges for RE", *Proceedings of the Sixth International Workshop on Requirements Engineering: Foundation for Software Quality*, Stockholm, Sweden, 2000.

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

[18] Babara P., Craig M., "Innovations for Requirement Analysis. From Stakeholders' Needs to Formal Designs", *14th Monterey Workshop 2007*, Monterey, CA, USA, September 10-13, 2007.

[19] Brian B.,Daniel J. P.,Juergen K.,Arnold R., "*Software & Systems Requirements Engineering: In Practice*" , McGraw-Hill Osborne Media, 2009

[20] Ralph R. Y., "*The Requirements Engineering Handbook (Artech House Technology Management and Professional Development Library)*", Artech Print on Demand, 2003

[21] CMMI for Development, Version 1.2, CMMI-DEV v1.2, CMU/SEI-2006-TR-008, Technical Report, Software Engineering Institute, August 2006,

URL: http://www.sei.cmu.edu/pub/documents/06.reports/pdf/06tr008.pdf.
[22] The TickIT Guide – Using ISO 9001:2000 for Software Quality Management System, Construction, Certification and Continual Improvement, Issue 5.0, 2001.
[23] Gorschek T., Tejle K., "A Method for Assessing Requirements Engineering Process Maturity in Software Projects", Blekinge Institute of Technology, Master Thesis Computer Science no. MSC-2002:2, 2002.
[24] Gomes A., Pettersson A., "market-driven Requirements Engineering Process Model – MDREPM", Master Thesis Computer Science, 2007.
[25] Gorschek T., Garre P., Larsson S., Wohlin C., "A Model for Technology Transfer in Practice," *IEEE Software*, vol. 23, no. 6, pp. 88-95, 2006.
[26] Deming W.E., "*Out of the Crisis*", MIT Press International, Cambridge, Mass, W. E. 1982.
[27] Humphrey W. S., "*Managing the Software Process*", Addison-Wesley Longman Publishing Co., Inc., 1989.
[28] Paulk M.C., Curtis B., Chrissis M.B., Weber C.V.,"Capability Maturity Model TM for Software Version 1.1", 1993.
[29] Kuvaja P., Bicego A., "BOOTSTRAP: Europe's assessment method", *IEEE Software*, v.10 n.3, p.93-95, May 1993
[30] Paulk M.C., Konrad M.D., "An Overview of ISO's SPICE Project", *American Programmer*, Vol. 7, No. 2, February 1994, pp. 16-20.
[31] Redzic C., Baik J., "Six sigma approach in software quality improvement", *4th International Conference on Software Engineering Research*, Management and Applications, SERA 2006.
[32] Canfora G., Garcia F., Piattini M., Ruiz F., and Visaggio C., "Applying a framework for the improvement of software process maturity", *Software - Practice and Experience*, vol. 36, pp. 283-304, 2006.
[33] Regnell, B., Brinkkemper, J., "Market-Driven Requirements Engineering for Software Products", In: Aurum, A., Wohlin, C. (eds.) Engineering and Managing Software Requirements, pp. 287-308, *Springer*, Heidelberg , 2005.
[34] Sawyer P., Sommerville I. and Kotonya G., "Improving Market-Driven RE Processes", Lancaster, Lancaster University, 1999.
[35] Carlshamre P., "Release Planning in Market-Driven Software Product Development: Provoking and Understanding", *Requirements Engineering*, vol. 7:139–151, 2002.
[36] Jurison J., "Software Project Management: The Manager's View", *Communications of the Association for Information Systems*, Vol. 2, Article 17, 1999.
[37] Potts C., "Invented requirements and imagined customers: requirements engineering for off-the-shelf software", *Second IEEE International Symposium on Requirements Engineering (RE'95)*, pp.128, 1995.
[38] Abran A., Moore J. W., "Guide to the Software Engineering Body of Knowledge—SWEBOK", *Los Alamitos: IEEE* 2004 Edition, 2004
[39] Maiden N., Gizikis A., "Where Do Requirements Come From?", *IEEE Software.* 2001, Vol. 18, 5, pp. 10-12.
[40] Curtis B., Krasner H., Iscoe N., "A field study of the software design process for large systems", *Communications of the ACM*, vol.31, pp.1268-1287, 1998.
[41] Khurum M., Aslam K., and Gorschek T., "A method for early requirements triage and selection utilizing product strategies", *Proceedings of 14th Asia-Pacific Software Engineering Conference (APSEC'07)*, Piscataway, NJ, USA, pp. 97-104, 2008.
[42] Sommerville I., Kotonya G., "*Requirements Engineering*", Wiley & Sons, Chichester, 1997.
[43] Sommerville I., Sawyer P., "Requirements engineering: a good practice guide", *John Willey & Sons*, 1997.
[44] Kotonya G., Sommerville I., "*Requirements Engineering – Process and Techniques*", John Wiley, NewYork NY, 1998.

[45] Davis A., "*Just Enough Requirements Management: Where Software Development Meets Marketing*", Dorset House Publishing. New York, NY, 2005.
[46] Kasse T., "*Practical Insights into CMMI*", Artech House, Boston, 2004.
[47] Smith C., "Using a Quality Model Framework to Strengthen the Requirements Bridge", *icre*, Third International Conference on Requirements Engineering (ICRE'98), pp.0118, 1998
[48] Kauppinen M., Aaltio T., Kujala S., "Lessons Learned from Applying the Requirements Engineering Good Practice Guide for Process Improvement", *Software Quality - ECSQ 2002 : 7th International Conference*, Helsinki, Finland, 2002.
[49] Niazi M. and Shastry S., "Role of Requirements Engineering in Software development Process", *IEEE International Multi-Topic Conference (INMIC03)*, Islamabad, pp 402-407, 8-10 December, 2003.
[50] Brereton P., Kitchenham B., Budgen D., Turner M., and Khalil M., "Lessons from applying the systematic literature review process within the software engineering domain", *Journal of System and Software*, vol. 80, no.4, pp.571-583, 2007.
[51] http://www.ei.org/compendex
[52] http://www.ei.org/inspec_inspecachieve
[53] http://info.scopus.com/scopus-in-detail/facts/
[54] Lubars M. et al., "A Review of the State of the Practice in Requirements Modeling", *Proc. IEEE Int'l Symp. Requirements Eng.*, IEEE Computer Soc. Press, Los Alamitos, Calif., 1993, pp. 2-14
[55] Carmel E., Becker S., "A process model for packaged software development", *IEEE Transactions on Engineering Management*, vol. 41, no.5, pp.50-61, 1995.
[56] Hutchings A., Knox S., "Creating Products Customers Demand", *Communications of the ACM*, vol.38, no.5, 1995.
[57] Regnell B., Beremark P., Eklundh O.: "Requirements Engineering for Packaged Software - A Baseline Process in a Situated Improvement Programme", *Proc. Conference on European Industrial Requirements Engineering (CEIRE'98)*, Hammersmith, UK, 1998.
[58] Cohen, J., "*A coefficient of agreement for nominal scales*", Educational and Psychological Measurement, vol. 20, 37-46, 1960.
[59] Barney S., Aurum A., and Wohlin C., "A product management challenge: Creating software product value through requirements selection," *Journal of Systems Architecture*, vol. 54, pp. 576-593, 2008.
[60] Maiden N.A.M., Rugg G., "ACRE: Selecting methods for requirements acquisition," *Software Engineering Journal*, vol.11, pp. 183-192, 1996.
[61] Wiegers K., "*Software Requirements"*, Microsoft Press, Redmond, Washington, 2003.
[62] Natt och Dag J. et al, "A Linguistic-Engineering Approach to Large-Scale Requirements Management", *IEEE Software. 1*, 2005, Vol. 22, pp. 32-39.
[63] Natt och Dag J., Thelin T., Regnell B., "An Experiment on Linguistic Tool Support for Consolidation of Requirements from Multiple Sources in Market-Driven Product Development", *Empirical Software Engineering*, Vol. 11, 2, pp. 303-329, 2006,
[64] Novorita R., Grube G., "Benefits of Structured Requirements Methods for Market-Based Enterprises", *Proc. International Council on Systems Engineering (INCOSE) Sixth Annual International Symposium "Systems Engineering: Practices and Tools"*, Boston, Massachusetts, 1996.
[65] Regnell B., et al., "Visualization of Agreement and Satisfaction in Distributed Prioritization of Market Requirements", 2001.
[66] Ruhe G., Greer D., "Quantitative Studies in Software Release Planning under Risk and Resource Constraints", *IEEE*, pp. 262-271, 2003.
[67] Berander P., "Evolving Prioritization for Software Product Management", Karlskrona : Blekinge Institute of Technology, 2007
[68]Alexander I., Stevens R.,"Writing better requirements", Addison-Wesley, 2002.
[69] Nguyen M., "Requirements Engineering Process Maturity model: empirical evaluation", Blekinge Institute of Technology, Master Thesis Computer Science no. MSC-2010:28, 2010.

APPENDIX A: LIST OF IDENTIFIED PRACTICES

Note: the grey background denotes overlapped practices in compare to generalized practices (from RQ.1.2).

No.	Practice Name	Activity Area	Ack. by
1	Consider different method for eliciting different knowledge and requirements	Elicitation	[5]
2	Receive customer feedback in all phases of the software life cycle	Elicitation	[10]
3	Give customers feedback afterwards about their suggested requirements.	Elicitation	[10]
4	Involvement through Incident Reports, Idea Feedback, Suggestions	Elicitation	[10]
5	Train and motivate technical support staff	Elicitation	[14]
6	Use business concerns to drive requirements elicitation	Elicitation	[14]
7	Define procedures for receiving, analyzing and documenting requirements derived from users' in-service experience	Elicitation	[14]
8	Identify and consult system stakeholders	Elicitation	[14]
9	Record open questions when they occur	Elicitation	[19]
10	Perform Scenario analysis	Elicitation	[19]
11	Domain knowledge	Elicitation	[19]
12	Iteratively refine the non-functional requirements	Elicitation	[21]
13	Monitor source of problem	Elicitation	[25][30]
14	Collect/extract requirements and save to repository	Elicitation	[30]
15	Assign tracking and control info	Elicitation	[30]
16	Describe the environment from which the requirement originates (source of requirement)	Elicitation	[30]
17	Perform elicitation on different sources	Elicitation	[46]
18	Define validation checklists	Requirements validation	[14]
19	Use standard templates for describing requirements	Requirements validation	[14]
20	Organise formal requirements inspections for each release	Requirements validation	[14]
21	Use prototyping to animate requirements	Requirements validation	[14]
22	Propose requirements test cases	Requirements validation	[14]
23	Performing systematically requirements validation and verification is critical.	Requirements validation	[21]
24	Organize structured process validation walkthroughs (with or without any tool support)	Requirements validation	[21]
25	Release validation	Requirements validation	[25]
26	Validate the problem statement with the source to check that it is an accurate reflection of the intent	Requirements validation	[30]
27	Perform requirements validation from external perspectives	Requirements validation	[38]
28	Evaluate requirements from various perspective	Requirements validation	[47]
29	Make use of must- and wish-lists	Release Planning	[1]
30	Estimate resources	Release Planning	[3][22]
31	align requirements with the organization's product strategies	Release Planning	[3]
32	requirements selection had to be aligned with business strategies	Release Planning	[4]
33	Perform systematic requirements prioritization	Release Planning	[1][10][14] [25][30][42]
34	Involve Stakeholder by Voting	Release Planning	[22]
35	Plan open-endedly with a pre-defined rhythm	Release Planning	[23][35]
36	Theme identification	Release Planning	[25]
37	Publish launching preparation package	Release Planning	[25]
38	Release definition	Release Planning	[25]
39	Requirements Selection	Release Planning	[25][32]

40	Plan for more than 1 release ahead	Release Planning	[23][32]
41	Consider strength of the interdependencies other than types	Release Planning	[32]
42	Consider the comparison of requirements, group of requirements and releases	Release Planning	[32]
43	Perform Requirements bundling	Release Planning	[36]
44	Explicate planning levels and time-horizons	Release Planning	[23][35]
45	Consider when adding new features for the next release package	Release Planning	[40]
46	Keep the release package action not too often and too many out at a time	Release Planning	[40]
47	Use work-around when necessary for new release	Release Planning	[40]
48	Consider Interdependencies among Quality requirements	Requirements Analysis	[29]
49	Integrate a reuse measurement process into the RE process	Requirements Analysis	[21][44]
50	Identify interdependencies between requirements	Requirements Analysis	[8][13]
51	Focus on value-related interdependencies in a product development situation.	Requirements Analysis	[8]
52	Analyze similarity, completeness, ambiguity, complexity and feasibility	Requirements Analysis	[10]
53	Assure validity, consistency, stability, importance of req specs	Requirements Analysis	[11]
54	Structuring the requirements into a hierarchy, abstraction levels	Requirements Analysis	[13]
55	evaluate the cost of meeting each requirement	Requirements Analysis	[14]
56	review all the requirements including those that weren't selected for the last release	Requirements Analysis	[14]
57	Supplement natural language with other descriptions of requirements	Requirements Analysis	[14]
58	Develop complementary system models	Requirements Analysis	[14]
59	Model the system architecture	Requirements Analysis	[14]
60	Perform requirements triage	Requirements Analysis	[16]
61	Reuse core asset identification	Requirements Analysis	[25]
62	Convince stakeholders	Requirements Analysis	[27]
63	Train stakeholders in Prioritization techniques	Requirements Analysis	[27]
64	Publish the priorities	Requirements Analysis	[27]
65	Estimate effort for requirements implementation	Requirements Analysis	[1][27]
66	Consider different dimensions in Requirements prioritization	Requirements Analysis	[26][27][37]
67	Schedule development.	Requirements Analysis	[27]
68	Re-link the abstraction level when there are requirements changes	Requirements Analysis	[33]
69	Specify Requirements resource	Requirements Analysis	[33]
70	Describe problem scenario	Requirements Analysis	[30]
71	Categorize each problem statement (usability, availability, etc.) and correlate (duplicates, contradictions, etc.) with all other problem state	Requirements Analysis	[30]
72	Analyze the problems in terms of market and competitive posture	Requirements Analysis	[30]
73	Assess the value	Requirements Analysis (and Negotiation)	[30]
74	"Maintain" the communication of requirements rationale in organization of the priority rationales through the organization	Requirements Analysis	[42]
75	Identify Domain analysis team for Requirements reuse	Requirements Analysis	[44]
76	Identify Domain analysis work plan for Requirements reuse	Requirements Analysis	[44]
77	Identify Involvement of domain experts	Requirements Analysis	[44]
78	Domain selection	Requirements Analysis	[44]
79	Perform Reuse assessment	Requirements Analysis	[44]
80	Set Reuse Target	Requirements Analysis	[44]
81	Write detailed requirements specification	Requirements Process Management	[1][34]
82	Define a Standard Document Structure	Requirements Process Management	[14]

83	Baseline the high-level requirements at a fixed cut-off time	Requirements Management Process	[14]
84	At the product concept stage, document the business goals and user requirements	Requirements Management] Process	[14]
85	Make use of attributes for the specification of requirements on all levels of abstraction.	Requirements Management Process	[16]
86	Document the rationale for the requirements.	Requirements Management Process	[21]
87	Requirements Exclusion Rationale	Requirements Management Process	[32]
88	Note Requirement state	Requirements Management Process	[33][36]
89	Specify requirement due date	Requirements Management Process	[33]
90	Specify requirement version	Requirements Management Process	[33]
91	Specify requirement creation date	Requirements Management Process	[33]
92	Specify requirement last change date	Requirements Management Process	[33]
93	Define requirement attributes: e.g. priority, needed effort…	Requirements Management Process	[36]
94	Note Requirement state	Requirements Management Process	[36]
95	Ensure requirements and project information are online accessible and traceable	Requirements Management Process	[38]
96	Portfolio mgt	Requirements Management Process	[25]
97	Define product roadmap	Requirements Management Process	[17][23][25]
98	Specify requirements owner	Requirements Management Process	[33]
99	Identify quality indicators	Requirements Management Process	[43]
100	Specify product strategy	Requirements Management Process	[3]
101	Specify product-technology roadmap.	Requirements Management Process	[3]
102	Disseminating the roadmapping knowledge	Requirements Management Process	[9]
103	Specify three distinct roles that appeared to be necessary in a roadmapping context	Requirements Management Process	[9]
104	Introduce strategies for managing business risk associated with requirements	Requirements Management Process	[20]
105	Create win-win partnerships among process owners, external consultants and internal IT-staff.	Requirements Management Process	[21]
106	Separate the planning of products' business goals from R&D resource allocation	Requirements Management Process	[23]
107	Emphasize whole-product thinking	Requirements Management Process	[23]
108	Specify requirements manager	Requirements Management Process	[33]
109	Assign responsibilities for analysis and validation.	Requirements Management Process	[30]
110	Specify roles and responsibilities in detail	Requirements Management Process	[36]
111	Define the core team with empowered stakeholders such as product manager, a marketing manager and a technical project manager.	Requirements Management Process	[38]
112	Perform more detail study on specific idea (features, etc.) while planning roadmap	Requirements Management Process	[39]
113	Install an effective core team for each product release	Requirements Management Process	[47]

114	Continuously manage change throughout design, implementation and verification	Requirements Management Process	[1]
115	Establish direct link between stakeholders and the developers	Requirements Management Process	[34]
116	Perform Pre and post traceability	Requirements Management Process	[11]
117	Uniquely identify each requirement	Requirements Management Process	[14]
118	Define change management policies	Requirements Management Process	[14]
119	Use a tool to manage requirements	Requirements Management Process	[14]
120	Define policies for requirements management	Requirements Management Process	[14]
121	Record rejected requirements	Requirements Management Process	[14]
122	Consider dependencies between the RE process and the support tools it comes with	Requirements Management Process	[21]
123	Enforce traceability policies.	Requirements Management Process	[21]
124	Use the RE process to prevent requirements leakages	Requirements Management Process	[21]
125	Get a data architect involved at least on a part-time basis.	Requirements Management Process	[21]
126	Install a process for change impact analysis	Requirements Management Process	[21]
127	Consider internal and external stakeholders	Requirements Management Process	[25]
128	Establish Requirements organizing	Requirements Management Process	[25]
129	Consider requirement tools/methods for Elicitation to be aligned with used management tools.	Requirements Management Process	[26]
130	Maintain priority information of requirements	Requirements Management Process	[27]
131	Product life-cycle with early gate reviews	Requirements Management Process	[38]
132	Train the stakeholders involved in roadmapping	Requirements Management Process	[39]
133	Maintain domain requirements repository	Requirements Management Process	[44]
134	Adapt before making use of agile methods in product management	Requirements Management Process	[45]
135	Focus the product life cycle on upstream gate reviews	Requirements Management Process	[47]
136	Using online checklist for meetings	Requirements Management Process	[47]
137	Assure dependable portfolio visibility and release implementation	Requirements Management Process	[47]

APPENDIX B: LIST OF IDENTIFIED MODELS AND FRAMEWORKS

Name	Type	Purpose	Activity Area	Ack. by
ACRE	Framework	Assist requirements engineer to choose method for requirements elicitation	Elicitation	[5]
Customer involvement factory	Model	Identify the needs of the customer in a manner that really involves the customers	Elicitation	[10]
Lightweight replanning process model	Model	select the most promising features to accommodate changing market driven product demands	Release Planning	[22]
Abstraction Model	Model	Handle large quantities of requirements of varying degrees of detail and offers a structure and process for the work-up of these requirements	Requirements Analysis	[33]
requirements abstraction model	Model	Handle large quantities of requirements of varying degrees of detail and offers a structure and process for the work-up of these requirements	Requirements Analysis	[16]
FSM (functional size measurement)	Model	Complex model, could be used to estimate size and cost of requirements (both Functional and Non-functional)	Requirements Analysis	[37]
REPEAT	Model	RE process	Requirements Process Management	[1]
goal-oriented requirements communication	Model	efficiency and effectiveness of requirements communication can be increased	Requirements Process Management	[11]
Software Product Management process	Model	able to offer a capable product support service for solving the customers' problems and collecting some valuable feedback data to be analyzed later and used for designing the future product versions.	Requirements Process Management	[15]
Software Product Management Workbench	Framework	RE framework	Requirements Process Management	[25]
REQUEST	Model	Manage RE process	Requirements Process Management	[30]
Traceability model	Model	the system meets the current	Requirements Process Management	[17]

APPENDIX C: LIST OF IDENTIFIED TECHNIQUES AND METHODS

Name	Type	Purpose	PA	Ack. by
Customer Participation Sessions (CPSs)	Method	Involve customers in the requirements gathering process.	Elicitation	[10]
Enhanced requirements elicitation and mobile system construction	Method	Wide Audience Requirements Elicitation and Rapid Prototyping they need.	Elicitation	[2]
Observation	Technique	Method for eliciting requirements	Elicitation	[5]
Unstructured Interview	Technique	Method for eliciting requirements	Elicitation	[5]
Structured Interview	Technique	Method for eliciting requirements	Elicitation	[5]
Protocol Analysis	Technique	Method for eliciting requirements	Elicitation	[5]
Card sorting	Technique	Method for eliciting requirements	Elicitation	[5]
Laddering	Technique	Method for eliciting requirements	Elicitation	[5]
Repertory Grids	Technique	Method for eliciting requirements	Elicitation	[5]
Brainstorming	Technique	Method for eliciting requirements	Elicitation	[5]
Rapid Prototyping	Technique	Method for eliciting requirements	Elicitation	[5]
Scenario Analysis	Technique	Method for eliciting requirements	Elicitation	[5]
Rapid Application Development (RAD) Workshop	Technique	Method for eliciting requirements	Elicitation	[5]
Ethnographic Methods	Technique	Method for eliciting requirements	Elicitation	[5]
Agile roadmapping	Method	to define roadmap	Organizational Support	[39]
Quper	Technique	Roadmapping for Quality Requirements	Organizational Support	[43]
Roadmapping	Technique	explore and communicate the dynamic linkages between markets, products and technologies over time	Organizational Support	[24]
AHP	Technique	Prioritization technique	Release Planning	[12]
AHP	Technique	Prioritization technique	Release Planning	[13]
AHP	Technique	Prioritization technique	Release Planning	[18]
Analytical Hierarchy Process (AHP).	Technique	Prioritization technique	Release Planning	[22]
Cost-value approach	Method	Select requirements	Release Planning	[13]

Cummulative Voting	Technique	Prioritization technique	Release Planning	[18]
Cumulative Voting	Technique	Prioritization technique	Release Planning	[12]
Hierarchical Cumulative Voting	Technique	quantify the importance of different requirements	Release Planning	[12]
Planning game	Technique	Prioritization technique	Release Planning	[18]
Post-release Analysis of Requirements Selection Quality	Method	finding process improvement proposals for the release planning activity	Release Planning	[18]
Visualization of interdependencies	Technique	identification of singular requirements, clusters of interdependent requirements, as well as highly dependent requirements at a quick glance	Release Planning	[8]
A Method for Early Requirements Triage and Selection	Method	a stepwise guide to creating product strategies taking both strategic and technical views into account, perform requirements triage	Requirements Analysis	[3]
Analytic Hierarchy Process	Technique	Prioritization technique	Requirements Analysis	[37]
Binary search tree	Technique	Prioritization technique	Requirements Analysis	[37]
Bubblesort	Technique	Prioritization technique	Requirements Analysis	[37]
Cost benefit analysis	Technique	Prioritization technique	Requirements Analysis	[37]
Cumulative voting	Technique	Prioritization technique	Requirements Analysis	[37]
Hierarchical Cumulative Voting	Technique	Prioritization technique	Requirements Analysis	[37]
Hierarchy AHP	Technique	Prioritization technique	Requirements Analysis	[37]
Minimal spanning tree matrix	Technique	Prioritization technique	Requirements Analysis	[37]
Multi-Attribute Utility Theory	Technique	Prioritization technique	Requirements Analysis	[37]
Numeral assignment	Technique	Prioritization technique	Requirements Analysis	[37]
Outranking	Technique	Prioritization technique	Requirements Analysis	[37]
Pair-wise comparison	Technique	Prioritization technique	Requirements Analysis	[42]
Pair-wise comparisons	Technique	Prioritization technique	Requirements Analysis	[26]
Planning game	Technique	Prioritization technique	Requirements Analysis	[26]
Planning Game	Technique	Prioritization technique	Requirements Analysis	[37]
Priority groups	Technique	Prioritization technique	Requirements Analysis	[37]
Tool-supported pair-wise	Technique	Prioritization technique	Requirements Analysis	[26]
Top 10 Requirements	Technique	Prioritization technique	Requirements Analysis	[37]

Value Point estimation	Method	To evaluate the gained value (covering financial and non-financial value) of the software product, useful for the release planning decision making	Requirements Analysis	[38]
Weighting Method	Technique	Prioritization technique	Requirements Analysis	[37]
Wiegers	Technique	Prioritization technique	Requirements Analysis	[42]

APPENDIX D: LIST OF IDENTIFIED TOOLS

Name	Type	Note	PA	Ack. by
Cost-value tool support	Tool	prioritization tool using AHP	Release Planning	[13]
PARSEQ tool	Tool	tool spport for PARSEQ	Release Planning	[18]
ReqSimile	Tool	finding and linking similar requirements	Requirements Analysis	[6]
A market Driven REQuirements Management Tool (MDREQ)	Tool	Capability: Four-stage Requirements Process Support, Administrative Authorization Functions, Database to Database Communication, Remote Access, Miscellaneous Usability features, Prioritization sub-process (Full Spin Management)	Requirements Process Management	[30]
Caliber RM - Borland	Tool	Lifecycle oriented, for large systems, traceability	Requirements Process Management	[31]
Requisite Pro - IBM Rational	Tool	Change management, traceability, XML support; work with MS Word, Rational Rose, TeamTest, MS Project	Requirements Process Management	[31]
ReqMan	Tool	Repository system	Requirements Process Management	[9]
RM Trak - RM Trak	Tool	Requirement management entry level	Requirements Process Management	[31]
RTM system	Tool	Provide repository for requirements management	Requirements Process Management	[28]
Vital link - Compliance Automation	Tool	Database-centric	Requirements Process Management	[31]

REFERENCES INCLUDED IN THE SYSTEMATIC REVIEW

[1] Regnell B., Beremark P., and Eklundh O., "A market-driven Requirements Engineering Process - Results from an Industrial Process Improvement Programme", *Springer*, pp. 121-129, 1998.

[2] Rossi M., and Tuunanen T., "A method and tool for wide audience requirements elicitation and rapid prototyping for mobile systems", Springer-Verlag, Berlin, Germany, pp. 629-40, 2004.

[3] Khurum M., Aslam K., and Gorschek T., "A method for early requirements triage and selection utilizing product strategies", *Proceedings of 14th Asia-Pacific Software Engineering Conference (APSEC'07)*, Piscataway, NJ, USA, pp. 97-104, 2008.

[4] Barney S., Aurum A., and Wohlin C., "A product management challenge: Creating software product value through requirements selection," *Journal of Systems Architecture*, vol. 54, pp. 576-593, 2008.

[5] Maiden N.A.M., and Rugg G., "ACRE: Selecting methods for requirements acquisition," *Software Engineering Journal*, vol.11, pp. 183-192, 1996.

[6] Natt och Dag J., Thelin T., and Regnell B., "An experiment on linguistic tool support for consolidation of requirements from multiple sources in market-driven product development", vol. 11, pp. 303-29, 2006.

[7 Regnell B., Martin H., Natt och Dag J., Beremark P., and Hjelm T., "An industrial case study on distributed prioritization in market-driven requirements engineering for packaged software," *Requirements Engineering*, vol. 6, pp. 51-62, 2001.

[8] Carlshamre P., Sandahl K., Lindvall M., Regnell B., and Natt och Dag J., "An industrial survey of requirements interdependencies in software product release planning", *IEEE Comput. Soc*, Los Alamitos, CA, USA, pp. 84-91, 2000.

[9] Jantunen S. and Smolander K., "Challenges of Knowledge and Collaboration in Roadmapping," *Proceedings of the international Workshop on Software Product Management, IWSPM, IEEE Computer Society*, pp. 19-26, 2006.

[10] Kabbedijk J., Brinkkemper S., Jansen S., and van der Veldt B., "Customer involvement in requirements management: lessons from mass market software development", *17th IEEE International Requirements Engineering Conference*, Piscataway, NJ, USA, pp. 281-6, 2009.

[11] Fricker S., Gorschek T., and Glinz M., "Goal-oriented requirements communication in new product development", *Proceedings of the 2008 Second international Workshop on Software Product Management* , IWSPM, IEEE Computer Society, Washington, DC, 27-34, 2008.

[12] Berander P., and Jonsson P., "Hierarchical Cumulative Voting (HCV) prioritization of requirements in hierarchies," vol. 16, pp. 819-849, 2006.

[13] Karlsson J., Olsson S., and Ryan K., "Improved practical support for large-scale requirements prioritizing," vol. 2, pp. 51-60, 1997.

[14] Sawyer P., Sommerville I., and Kotonya G., "Improving market-driven RE processes", Espoo, Finland, Tech. Res. Centre Finland, pp. 222-36, 1999.

[15] Kilpi T., "Improving software product management process: implementation of a product support system", IEEE Comput. Soc., Los Alamitos, CA, USA, pp. 3-12, 1998.

[16] Gorschek T., Garre P., Larsson S.B.M., and Wohlin C., "Industry evaluation of the requirements abstraction model," *Requirements Engineering*, vol. 12, pp. 163-190, 2007.

[17] Ramesh B., Dwiggins D., DeVries G., and Edwards M., "Towards requirements traceability models", *Proceedings of the 1995 International Symposium and Workshop on Systems Engineering of Computer-Based Systems, IEEE Computer Society Press*, Tucson, AZ, USA, pp. 229-232, 1995.

[18] Karlsson L. and Regnell B., "Introducing tool support for retrospective analysis of release planning decisions," *Springer-Verlag*, Berlin, Germany, pp. 19-33, 2006.

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

[20] Fogelstrom N.D., Svahnberg M., and Gorschek T., "Investigating impact of business risk on requirements selection decisions", *Proceedings of the 2009 35th Euromicro Conference on Software Engineering and Advanced Applications*, Piscataway, NJ, USA, pp. 217-23, 2009.
[21] Daneva M., "Lessons learnt from five years of experience in ERP requirements engineering", IEEE Comput. Soc., Los Alamitos, CA, USA, pp. 45-54, 2003.
[22] AlBourae T., Ruhe G., and Moussavi M., "Lightweight replanning of software product releases," *Proceedings of the First International Workshop on Software Product Management, IWSPM'06*, Minneapolis, MN, pp. 27-34, 2006.
[23] Lehtola L., Kauppinen M., Vahaniitty J., and Komssi M., "Linking business and requirements engineering: Is solution planning a missing activity in software product companies?", *Springer London Ltd*, The Guildway, Old Portsmouth Road, Artington, Guildford, GU3 1LP, United Kingdom, pp. 113-128, 2009.
[24] Lehtola L., Kauppinen M., and Kujala S., "Linking the business view to requirements engineering: Long-term product planning by roadmapping,", *Proceedings of the 13th IEEE international Conference on Requirements Engineering, IEEE Computer Society*, Paris, pp. 439-443, 2005.
[25] Van De Weerd I., Brinkkemper S., Nieuwenhuis R., Versendaal J., and Bijlsma L., "On the creation of a reference framework for software product management: Validation and tool support", *Inst. of Elec. and Elec. Eng. Computer Society*, Minneapolis, MN, United states, pp. 3-11, 2006.
[26] Karlsson L., Thelin T., Regnell B., Berander P., and Wohlin C., "Pair-wise comparisons versus planning game partitioning-experiments on requirements prioritization techniques", *Empirical Softw. Engg.*, vol.12, no.1, pp. 3-33, 2007.
[27] Firesmith D., "Prioritizing requirements," *Journal of Object Technology*, vol. 3, pp. 35-47, 2004.
[28] Grynberg A., and Goldin L., "Product management in telecom industry-using requirements management process", *IEEE Comput. Soc.*, Los Alamitos, CA, USA, pp. 63-70, 2003.
[29] Svensson R.B., Gorschek T., and Regnell B., "Quality requirements in practice: An interview study in requirements engineering for embedded system", *Springer Verlag*, Amsterdam, Netherlands, pp. 218-232, 2009.
[30] Yeh A.C., "REQUirements Engineering Support Technique (REQUEST) A market Driven Requirements Management Process", *Proc. of Second Symposium of Quality Software Development Tools*, pp. 211-223, 1992.
[31] Wieringa R. and Ebert C., "RE'03: Practical requirements engineering solutions," *IEEE Software*, vol.21, pp. 16-17, 2004.
[32] Carlshamre P., "Release planning in market-driven software product development: provoking an understanding", *Requirements Engineering Journal*, vol. 7, no.3, pp. 139-51, 2002.
[33] Gorschek T. and Wohlin C., "Requirements abstraction model," *Requirements Engineering*, vol.11, pp. 79-101, 2006.
[34] Karlsson L., Dahlstedt ÅG, Regnell B., Natt och Dag J., and Persson A., "Requirements engineering challenges in market-driven software development - An interview study with practitioners," *Information and Software Technology*, vol. 49, pp. 588-604, 2007.
[35] Lehtola L., Kauppinen M., and Vahaniitty J., "Strengthening the link between business decisions and RE:
Long-term product planning in software product companies", *Proceedings of the 15th IEEE International Requirements Engineering Conference*, pp. 153-162, 2007.
[36] Carlshamre P., and Regnell B., "Requirements lifecycle management and release planning in market-driven requirements engineering processes", IEEE Comput. Soc., Los Alamitos, CA, USA, pp. 961-5, 2000.
[37] Herrmann A., and Daneva M., "Requirements prioritization based on benefit and cost prediction: An agenda for future research", *Inst. of Elec. and Elec. Eng. Computer Society*, Barcelona, Catalunya, Spain, pp. 125-134, 2008.

[38] Ebert C., and De Man J., "Requirements uncertainty: Influencing factor and concrete improvements", Association for Computing Machinery , St. Louis, MO, United states, pp. 553-560, 2005.
[39] Wilby D., "Roadmap transformation: from obstacle to catalyst", *Agile Conference*. Piscataway, NJ, USA, pp. 229-34, 2009.
[40] Jansen S., and Brinkkemper S., "Ten Misconceptions about Product Software Release Management explained using Update Cost/Value Functions", *First International Workshop on Software Product Management*, pp. 44-50, 2006.
[41] Van den Akker M., Brinkkemper S., Diepen G., and Versendaal J., "Software product release planning through optimization and what-if analysis", *Inf. Softw. Technol.*, vol. 50, pp. 101-111, 2008.
[42] Lehtola L., and Kauppinen M., "Suitability of requirements prioritization methods for market-driven software product development", *Software Process: Improvement and Practice*, vol. 11, no.1, pp. 7-19, 2006.
[43] Regnell B., Svensson R.B., and Olsson T., "Supporting roadmapping of quality requirements", *IEEE Softw.*, vol. 25, no.2, pp. 42-7, 2008.
[44] Lam W., Jones S., and Britton C., "Technology Transfer for Reuse: A Management Model and Process Improvement Framework", IEEE Comp Soc, Anon, Ed., Colorado Springs, CO, USA, pp. 233-240, 1998.
[45] Fogelström N.D., Svahnberg M., and Olsson P., "The impact of agile principles on market-driven software product development," *Journal of Software Maintenance and Evolution*, vol. 22, pp. 53-80, 2010.
[46] Jantunen S., and Smolander K., "Towards Global market-Driven Software Development Processes: An Industrial Case Study", *Proceedings of the 2006 international Workshop on Global Software Development For the Practitioner*, Shanghai, China, pp. 94-100, 2006.
[47] Ebert C., "Understanding the product life cycle: four key requirements engineering techniques", *IEEE Software*, vol. 23, no. 3, pp. 19-25, 2006.
[48] Mohamed S.I. and Wahba A.M., "Value estimation for software product management", *IEEE International Conference on Industrial Engineering and Engineering Management*, Singapore, pp.2196-2200, 2008.

APPENDIX E: LIST OF GOOD PRACTICES

1. List of highly applicable "good practices"

Practice Name
"Maintain" the communication of requirements rationale in organization of the priority rationales through the organization
Align requirements with the organization's product strategies
Analyze similarity, completeness, ambiguity, complexity and feasibility
Analyze the problems in terms of market and competitive posture
Assess the value for requirements
Assign responsibilities for analysis and validation.
Assign tracking and control info
Assure dependable portfolio visibility and release implementation
Assure validity, consistency, stability, importance of requirements specifications
Categorize each problem statement (usability, availability, etc.) and correlate (duplicates, contradictions, etc.) with all other problem state.
Collect/extract requirements and save to repository
Consider dependencies between the RE process and the support tools it comes with
Consider different dimensions in Requirements prioritization
Consider different method for eliciting different knowledge and requirements
Consider Interdependencies among Quality requirements
Consider internal and external stakeholders
Consider requirement tools/methods
Consider strength of the interdependencies other than types
Consider the comparison of requirements, group of requirements and releases
Construct roadmap
Continuously manage change throughout design, implementation and verification
Create win-win partnerships among process owners, external consultants and internal IT-staff.
Define the core team with empowered stakeholders such as product manager, a marketing manager and a technical project manager.
Describe problem scenario
Describe the environment from which the requirement originates (source of requirement)
Establish direct link between stakeholders and the developers
Disseminating the roadmapping knowledge
Document the rationale for the requirements.
Emphasize whole-product thinking
Enforce traceability policies.
Ensure requirements and project information are online accessible and traceable
Estimate effort for requirements implementation
Estimate resources
Evaluate requirements from various perspective
Explicate planning levels and time horizons
Focus on value-related interdependencies in a product development situation.
Focus the product life cycle on upstream gate reviews
Get a data architect involved at least on a part-time basis.
Give customers feedback afterwards about their suggested requirements.
Identify interdependencies between requirements
Install a process for change impact analysis
Install an effective core team for each product release
Integrate a reuse measurement process into the RE process
introducing strategies for managing business risk associated with requirements
Involve Stakeholder by Voting

Involvement through Incident Reports, Idea Feedback, Suggestions
iteratively refine the non-functional requirements
Make us of must- and wish-lists
Monitor source of problem
Note Requirement state
Organize structured process validation walkthroughs (with or without any tool support)
Perform elicitation on different sources
Perform more detail study on specific idea (features, etc.) while planning roadmap
Perform Pre and post traceability
Perform prioritization
Perform requirements validation from external perspectives
Perform triage
Performing systematically requirements validation and verification is critical.
Plan for more than 1 release ahead
Plan open-endedly with a pre-defined rhythm
Portfolio management
Publish launching preparation package
Receive customer feedback in all phases of the software life cycle
Release definition
Release validation
Re-link the abstraction level when there are requirements changes
Requirements organizing
Requirements Exclusion Rationale
Requirements selection
requirements selection had to be aligned with business strategies
Reuse core asset identification
Separate the planning of product's business goals from R&D resource allocation
Specify product strategy
Specify product-technology roadmap.
Specify requirement creation date
Specify requirement due date
Specify requirement last change date
Specify requirement version
Specify requirements manager
Specify requirements owner
Specify Requirements resource
Specify roles and responsibilities in detail
Specify three distinct roles that appeared to be necessary in a roadmapping context
Structuring the requirements into a hierarchy, abstraction levels
Theme identification
Train the stakeholders involved in roadmapping
Make use of attributes for the specification of requirements on all levels of abstraction.
Use the RE process to prevent requirements leakages
Validate the problem statement with the source to check that its accuracy
Write detailed requirements specification
Define and maintain a Requirements management process
Introduce tool support for RE
Define traceability policy
Obtain common understanding of requirements among different involving teams
Create artifacts to facilitate elicitation and analysis
Perform refinement and abstraction
Perform requirements triage
Analyze for double requirements

Post requirements selection evaluation

2. List of "good practices"

Practice Name
At the product concept stage, document the business goals and user requirements
Baseline the high-level requirements at a fixed cut-off time
Define a Standard Document Structure
Define change management policies
Define policies for requirements management
Define procedures for receiving, analyzing and documenting requirements derived from users' in-service experience
Define requirement attributes: e.g. priority, needed effort…
Define validation checklists
Develop complementary system models
Domain knowledge
evaluate the cost of meeting each requirement
Identify and consult system stakeholders
Model the system architecture
Organize formal requirements inspections for each release
Propose requirements test cases
Record open questions when they occur
Record rejected requirements
review all the requirements including those that weren't selected for the last release
Scenario analysis
Supplement natural language with other descriptions of requirements
Train and motivate technical support staff
Uniquely identify each requirement
Use a tool to manage requirements
Use business concerns to drive requirements elicitation
Use prototyping to animate requirements
Use standard templates for describing requirements

3. List of Discarded Practices

Adapt before making use of agile methods in product management
Consider when adding new features for the next release package
Convince stakeholders
Cost-benefit study
Domain REQUIREMENTS referencing
Domain selection
Identify Domain analysis team for Requirements reuse
Identify Domain analysis work plan for Requirements reuse
Identify Involvement of domain experts
Identify quality indicators
Keep the release package action not too often and too many out at a time
Maintain domain requirements repository
Maintain the requirements priorities
Perform Requirements bundling
Publish the priorities
Reuse assessment
Reuse Target setting
Schedule development.
Train stakeholders in Prioritization techniques

Use work-around when necessary for new release
Change analysis for requirements reuse

APPENDIX F: RESULTS OF COMBINATION ANALYSIS

I. Organizational Support

1. Common Actions: None
2. Combined Actions

bespoke RE Practice	MDRE Practice	New Action	Rationale
Assign responsibilities in requirements development and management processes	Assign responsibilities for analysis and validation. Specify roles and responsibilities in detail	Define Roles and Responsibilities for Requirements Engineering Process	CONFORM

3. Different Actions

bespoke RE Practice	MDRE Practice	New Action	Rationale
	Assign process owner who contributes the necessary line know-how, design new processes and operational procedures, provides the project with the appropriate authority and resources.	Assign Owner of Requirements Process	APPLICABLE
Term Definition		Create a Product-wide Glossary of Terms	APPLICABLE
	Install an effective core team for each product release	Define Roles and Responsibilities for Release Planning Activities	ESSENTIAL
	Define group to responsible for requirements change management	Define Roles and Responsibilities for Change Control	APPLICABLE
	Specify three distinct roles that appeared to be necessary in a roadmapping context Define the core team with empowered stakeholders such as product manager, a marketing manager and a technical project manager.	Define Roles and Responsibilities for Product Management Organization	ESSENTIAL
	Specify product strategy Introducing strategies for managing business risk associated with requirements	Define Product Strategies	ESSENTIAL
	Specify product-technology roadmap. Perform more detail study on specific idea (features, etc.) while planning roadmap Construct roadmap Emphasize whole-product thinking Separate the planning of products' business goals from R&D resource allocation Use Roadmap to link market, product and technology over time Perform more detail study on specific idea (features, etc.) while planning roadmap	Define Product Roadmaps	ESSENTIAL
	Portfolio management	Define Organizational Strategies	ESSENTIAL
	Disseminating the roadmapping knowledge Assure dependable portfolio visibility and release implementation	Communicate Strategies in Organization	ESSENTIAL

II. Requirements Process Management

1. Common Actions

bespoke RE Practice	MDRE Practice	New Action
Requirements Identification	Uniquely identify each requirement	Uniquely Identify each Requirement
Requirements Origin Specification	Specify Requirements resource	Document Requirements' Source

2. Combined Actions

bespoke RE Practice	MDRE Practice	New Action	Rationale
Information Interchange Through CARE Information handling Through CARE	Collect/extract requirements and save to repository, Consider dependencies between the RE process and the support tools it comes with Consider requirement tools or methods Ensure requirements and project information are online accessible and traceable Use a tool to manage requirements Consider Tool Support for RE process (Abstract)	Introduce Tool Support for Requirements Engineering	CONFORM
Version traceability	Specify requirement version	Manage Versions of Requirements	
Re-prioritization – New Requirements Re-prioritization – New Releases Re-prioritization due to Change Re-prioritization with Regularity Install and maintain a process for requirements change management	Continuously manage change throughout design, implementation and verification Define change management policies Install a process for change impact analysis	Define a Process for Managing Change and Evolution	CONFORM
Ensure having the same interpretation of the requirements as the requirement providers	Validate the problem statement with the source to check that it is an accurate reflection of the intent Give customers feedback afterwards about their suggested Requirements	Establish effective communication with requirements issuers	CONFORM
Establish a common understanding of requirements between different stages	Create win-win partnerships among process owners, external consultants and internal IT-staff, Establish direct link between stakeholders and the developers, Get a data architect involved at least on a part-time basis, "Maintain" the communication of requirements priorities rationale in organization, Establish communication among different involving teams	Obtain common understanding of requirements among different involving teams	CONFORM
Interaction Matrice Create and maintain requirement's relation for traceability	Document Requirements Dependencies (Abstract) Consider the comparison and dependencies among requirements, group of requirements in release planning	Document Requirements' Relation	GENERALIZED
Backward-from traceability, Backward-to traceability, Forward-from traceability, Forward-to traceability	Define traceability policy, Perform Pre and post traceability, Enforce traceability policies.	Define traceability policies	

Train people involving in requirements development and management process	Train the stakeholders involved in roadmapping Train and motivate technical support staff	Train personnel in Requirements Management Process and specialty (e.g. Prioritization…)	CONFORM
Establish and maintain defined processes for developing and managing requirements	Define procedures for receiving, analyzing and documenting requirements derived from users' in-service experience Use the RE process to prevent requirements leakages Define policies for requirements management Define and maintain a RE management process	Define and Maintain a Requirements Management Process	CONFORM
Keep relevant stakeholders involve in requirements development and management processes	Evaluate requirements from various perspectives	Involve various perspectives in Requirement Engineering Process	
Create a baseline	Baseline the high-level requirements at a fixed cut-off time Define stopping criteria so that team members can determine what is good or stable enough in continuous change situation	Baseline Requirements	CONFORM

3. Different

bespoke RE Practice	MDRE Practice	New Action	Rationale
	Early connect portfolio considerations into requirements engineering process	Early connect portfolio considerations into requirements engineering process	ESSENTIAL
Define a mechanism to track change request		Track change requests	APPLICABLE
Consider change impact between requirements and other work products		Document Impact of Requirement on Other Artifacts	

III. Requirements Elicitation

1. Common Actions

bespoke RE Practice	MDRE Practice	New Action
System Domain Consideration	Domain knowledge Describe the environment from which the requirement originates (source of requirement) Record open questions when they occur	Consider System Domain Restrictions
Requirements Reuse	Integrate a reuse measurement process into the RE process Reuse core asset identification	Reuse Requirements
Quantitative Requirements Description	Iteratively refine the non-functional requirements	Qualify and Quantify Quality Requirements

2. Combined

bespoke RE Practice	MDRE Practice	New Action	Rationale
Stakeholder Identification: Ask Executive Stakeholders, Research Stakeholders Stakeholder Consulting: Executive Stakeholders, General Stakeholders, In-house Stakeholders	Identify and consult system stakeholders Consider internal and external stakeholders	Identify and Involve Relevant Stakeholders	CONFORM + GENERALIZED
Use scenario to elicit and analyze requirements Prototyping System Models Environmental Models Architectural Models	Scenario analysis Use prototype or scenario to facilitate Elicitation and Analysis Use prototyping to animate requirements Describe problem scenario Develop complementary system models Model the system architecture Supplement natural language with other descriptions of requirements	Create Artifacts to Facilitate Elicitation and Analysis	GENERALIZED
Elicit stakeholders' requirements using different methods	Consider different method for eliciting different knowledge and requirements Use suitable techniques to elicit requirements Monitor sources of the requirements such as User Groups, Customer Councils, competitive analysis and marketing groups	Adapt Elicitation Technique according to Situation	GENERALIZED + SUPPORT

3. Different

bespoke RE Practice	MDRE Practice	New Action	Rationale
Stakeholder Consulting: Executive Stakeholders, General Stakeholders, In-house Stakeholders		Distinguish between Customers, End-Users, and In-house Stakeholders	APPLICABLE
	Perform elicitation on different sources	Identify Other Requirements Sources	APPLICABLE
Technical Domain Consideration		Consider System's Technical Infrastructure	APPLICABLE
Operational Domain Consideration		Consider Co-existing Business Processes	APPLICABLE
Business Domain Consideration		Consider System's Business Process	APPLICABLE
Boundary definition through categorization		Consider System Boundaries	APPLICABLE
Human Domain Consideration		Consider Sociopolitical Influences on Requirements Sources	APPLICABLE
		Consider Quality Requirements	
	Use business concerns to drive requirements elicitation	Let Business Concern/Product Strategies guide Focus of	ESSENTIAL

		Elicitation Efforts	
	Establish Customer Involvement through Incident Reports, Idea Feedback, Suggestions Motivate technical support staff to identify requirements from customers' feedback Receive customer feedback in all phases of the software life cycle	Create Elicitation Channels for Requirements Sources	APPLICABLE

IV. Requirements Analysis

1. Common Actions: None
2. Combined Actions

bespoke RE Practice	MDRE Practice	New Action	Rationale
Volatile requirements Identification Ambiguous Requirements refinement Analyze requirements for consistency Analyze requirements for completeness, correctness, feasibility and testability	Analyze similarity, completeness, ambiguity, complexity and feasibility Assure validity, consistency, stability, importance of requirements specification Analyze for double requirements	Analyze for Missing, Double, Incomplete, Ambiguous Requirements	CONFORM
Prioritizing Requirements based on cost, functionality, risk or performance	Estimate effort for requirements implementation Evaluate the cost of meeting each requirement	Perform Systematic Requirements Prioritization at In-project level	SUPPORT
Interaction Matrice	Consider Interdependencies among Quality requirements, Identify interdependencies between requirements	Analyze for Requirements Functional Dependencies	SUPPORT
Boundary definition through categorization	Perform requirements triage	Identify irrelevant requirements for early dismiss (in/out scope OR Triage)	CONFORM
Classify requirements into groups based on defined criteria to enhance analysis Global System Requirements Identification	Re-link the abstraction level when there are requirements changes Structure the requirements into a hierarchy, abstraction levels Perform refinement and abstraction	Perform refinement and abstraction of requirements	CONFORM

3. Different Actions

bespoke RE Practice	MDRE Practice	New Action	Rationale
	Consider strength of the interdependencies other than types Focus on value-related interdependencies in a product development situation.	Analyze Value-related Dependencies between Requirements	APPLICABLE (Combined by CONFORM)
Risk Assessment-Selected, Risk Assessment-individual, Risk Assessment-sets		Perform Requirements Risk Analysis	APPLICABLE

V. Release planning

MDRE Practice	New Actions	Rationale
Align requirements with the organization's product strategies Requirements selection had to be aligned with business strategies	Synchronize Release Plan with Product Roadmap	ESSENTIAL

Release validation Review all the requirements including those that weren't selected for the last release Post requirements selection evaluation	Post Requirement Selection Evaluation	ESSENTIAL
Plan for more than 1 release ahead Plan open-endedly with a pre-defined rhythm,	Plan multiple release at pre-defined interval	ESSENTIAL
Make use of must- and wish-lists Release definition Requirements Selection	Package Requirements into Releases	ESSENTIAL
Assess the value Estimate effort for requirements implementation, evaluate the cost of meeting each requirement Estimate resources Perform prioritization	Perform Systematic Requirements Prioritization at Pre-project level based on value, cost, effort	ESSENTIAL
Involve Stakeholder by Voting	Involve different perspectives in release planning	ESSENTIAL
Consider different dimensions in Requirements prioritization	Consider additional factors for prioritization	ESSENTIAL

VI. Documentation and requirements specification

1. Common Actions

bespoke RE Practice	MDRE Practice	New Action
Record Requirements Rationale	Document the rationale for the requirements.	Document Requirements Rationale
Rejected Requirements Documentation	Record rejected requirements Requirements Exclusion Rationale	Record Rationale for Rejected Requirements

2. Combined Actions

bespoke RE Practice	MDRE Practice	New Action	Rationale
Document Summary Document Usage Description Business Case Requirements Description Template	Define a Standard Document Structure Use standard templates for describing requirements	Establish Standardized Structure for SRS	CONFORM + SUPPORT
Specify the requirement's important characteristics	Specify requirement creation date Specify requirement due date Specify requirement last change date Specify requirements manager Specify requirements owner Specify requirements resource Use of attributes for the specification of requirements on all levels of abstraction. Define requirement attributes: e.g. priority, needed effort… Write detailed requirements specification	Define Requirements Attributes	CONFORM + SUPPORT

3. Different Actions

bespoke RE Practice	MDRE Practice	New Action	Rationale
	Note Requirement state	Define Requirements States	APPLICABLE
User documentation		Define User Documentation Deliverables	APPLICABLE
System documentation		Define System Documentation Deliverables	APPLICABLE
Management documentation		Define Management Documentation Deliverables	APPLICABLE

VII. Quality Assurance

1. Common Actions

bespoke RE Practice	MDRE Practice	New Action
Analysis Through Checklists	Define validation checklists	Use Checklist to Ensure Quality of Requirements
Requirements Inspection	Organize formal requirements inspections for each release Organize structured process validation walkthroughs (with or without any tool support)	Organize Inspections to Ensure Quality of Requirements

2. Combined Actions

bespoke RE Practice	MDRE Practice	New Action	Rationale
Requirements Test Cases User Manual Draft	Propose requirements test cases	Create Preliminary Artifacts for Quality Assurance	GENERALIZED
Validate requirement's completeness and adequacy with stakeholders	Perform requirements validation from external perspectives	Validate requirements with relevant stakeholders	CONFORM

3. Different Actions

bespoke RE Practice	MDRE Practice	New Action	Rationale
Requirements Review		Review Requirements	APPLICABLE
System Model Paraphrasing		Perform System Model Paraphrasing	APPLICABLE

OS Organizational Support

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

OS.a1 Assign Owner of Requirements Process — Level 1

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

Supporting action(s)

- PM.a2 Define and Maintain a Requirements Management Process

OS.a2 Create a Product-wide Glossary of Terms — Level 1

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

OS.RR Roles and Responsibilities

In order to be able to produce repeatable and predictable results, it is important to define the roles, and their responsibilities, that deal with different aspects of requirements engineering.

OS.RR.a1 Define Roles and Responsibilities for Requirements Engineering Process — Level 2

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

Supporting action(s)

- PM.a2 Define and Maintain a Requirements Management Process

OS.RR.a2 Define Roles and Responsibilities for Release Planning Activities Level 2

The responsibilities in release planning include deciding which prioritization aspects to consider, how to prioritize, selecting requirements into release, just to name a few. The roles involved in release planning can be product managers, marketing managers, technical managers, experts, customers etc.

OS.RR.a3 Define Roles and Responsibilities for Change Control Level 2

As change happens all the time throughout the product lifecycle it is necessary to manage changes effectively by defining who is responsible for what in change control process. Some of the possible roles are change control board, change submitter, evaluator, modifier, and verifier. The change control board is in charge of making decisions whether to approve proposed changes. The change control board should comprise people from different perspectives e.g. project management, product management, marketing, and development in order to have well-rounded and accurate decisions. The evaluator is responsible for analyzing the impact of the requirements change. The modifier executes the approved change on affected artifacts whereas the verifier checks if the change was implemented correctly.

OS.RR.a4 Define Roles and Responsibilities for Product Management Organization Level 3

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

Supporting action(s)

- PM.a2 Define and Maintain a Requirements Management Process

OS.S Strategic

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

OS.S.a1 Define Product Strategies Level 2

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

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

Supporting action(s)

- OS.RR.a4 Define Roles and Responsibilities for Product Management Organization
- OS.S.a3 Define Organizational Strategies

OS.S.a2 Define Product Roadmaps Level 2

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

Supporting action(s)

- OS.RR.a4 Define Roles and Responsibilities for Product Management Organization
- OS.S.a1 Define Product Strategies

OS.S.a3 Define Organizational Strategies Level 3

Organizational strategies express the decision of the company board on the set of existing products, which new products are introduced and product development strategy. The organizational strategies are defined by knowing where the company is, where it should be heading and how it will get there. They are also served as the basis to define product strategies.

Supporting action(s)

- OS.RR.a4 Define Roles and Responsibilities for Product Management Organization

OS.S.a4 Communicate Strategies in Organization Level 3

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

Supporting action(s)

- OS.RR.a4 Define Roles and Responsibilities for Product Management Organization
- OS.S.a3 Define Organizational Strategies
- OS.S.a1 Define Product Strategies
- OS.S.a2 Define Product Roadmaps

PM Requirements Process Management

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

PM.a1 Introduce Tool Support for Requirements Engineering Level 1

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

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

Recommendations

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

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

Supporting action(s)

- PM.a4 Train people in requirements engineering specialty

PM.a2 Define and Maintain a Requirements Management Process — Level 1

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

Recommendations

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

Supporting action(s)

- OS.a1 Assign Owner of Requirements Process

PM.a3 Train personnel in Requirements Management Process and Speciality — Level 2

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

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

Supporting action(s)

- PM.a2 Define and Maintain a Requirements Management Process

PM.a4 Early connect portfolio considerations into requirements engineering process — Level 3

Portfolio is considered as the top driver of the requirements engineering process in MDRE case. It provides an overview of features across future releases covering the vision, market, architecture and technology. Based on portfolio considerations, the requirements team can drive individual roadmaps of the releases and projects. Hence, you should early define the connection between these two phases. You can realize this connection by specifying which

step(s) in RE the product portfolio considerations will involve, e.g. elicitation, triage, release planning, etc.

Supporting action(s)

- OS.S.a3 Define Organizational Strategies
- OS.S.a1 Define Product Strategies

PM.a6 Involve various perspectives in Requirements Engineering Process — Level 2

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

Supporting action(s)

- RE.SI Stakeholder and Requirements Source Identification

PM. CM Configuration Management

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

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

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

Recommendations

You can use CVS, Subversion to support version control.

Supporting action(s)

- PM.a1 Introduce Tool Support for Requirements Engineering

PM.CM.a2 Baseline Requirements — Level 1

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

when release planning is done.

Supporting action(s)

- PM.a1 Introduce Tool Support for Requirements Engineering
- PM.CM.a1 Manage Versions of Requirements

PM.CM.a3 Define a Process for Managing Change and Evolution Level 2

Change has been agreed to be the nature of requirements. Since there is no way to avoid it, you had better define a process to control it. The process is recommended to include the following considerations:

- Roles in the process: you should clearly define the involved stakeholders to process change request when it occurs at the beginning of the requirements process. The roles may include customer(s) if exist, engineer, project manager, etc...Especially, the roles for verification and approval should be specified.
- Procedure: the procedure of the change process should also be defined. They can cover certain steps a change request must follow and requirements mentioned in the request should be considered. This can also specify which factors and which technique to use during re-analyzing process.

Recommendations

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

Supporting action(s)

- OS.RR.a3 Define Roles and Responsibilities for Change Control
- PM.a1 Introduce Tool Support for Requirements Engineering

PM.CM.a4 Track change requests Level 2

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

Recommendations

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

Supporting action(s)

- PM.CM.a3 Define a Process for Managing Change and Evolution

PM.RT Requirements Traceability Policy

Along the requirements process, you will mostly apply number of changes on requirements. In order to ensure the consistency of the system, it is important that you prepare for these cases so that you can

always trace from requirements to other artifacts of the project and vice versa to apply necessary changes.

PM.TR.a1 1. Uniquely Identify each Requirement Level 1

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

Recommendations

Repository can automatically assign ID for requirements for you.

PM.TR.a2 Document Requirements' Source Level 1

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

Supporting action(s)

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

PM.TR.a3 Document Requirements' Relations Level 2

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

Supporting action(s)

- RA.a4 Analyze Requirements Functional Dependencies
- RA.a6 Identify irrelevant requirements for early dismissal (in/out scope OR Triage)

PM.TR.a4 Document Impact of Requirement on Other Artifacts Level 2

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

Supporting action(s)

- PM.a1 Introduce Tool Support for Requirements Engineering
- DS.a1 Define Requirements Attributes

PM.TR.a5 Define Traceability policies — Level 2

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

Supporting action(s)

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

PM.RC Requirements communication

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

PM.RC.a1 Establish effective communication with requirements issuers — Level 1

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

Supporting action(s)

- PM.RC.a2 Document requirements source

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

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

Supporting action(s)

- RE.SI.a1 Identify and Involve Relevant Stakeholders

RE Requirements Elicitation

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

RE.SI Stakeholder and Requirements Source Identification

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

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

Explicitly identify all potential stakeholders, who can be customers, end-users, marketing personnel, managers, developers, testers etc, and consult the relevant ones. The stakeholders will provide requirements or impose constraints on the system. This ensures that all the concern of affected people will be taken into account.

RE.SI.a2 Distinguish between Customers, End-Users, and In-house Stakeholders — Level 1

Customers are the people who have the authority to purchase/order the system whereas end-users are the ones actually using the system in their work. In some cases, customers may be end-users as well. As customers and end-users can have different interests and expectations in the system, it is important to distinguish between them in order to elicit all relevant requirements. In-house stakeholders involving in the development/management of the system are often overlooked in the elicitation process.

Supporting action(s)

- RE.SI.a1 Identify and Involve Relevant Stakeholders

RE.SI.a3 Identify Other Requirements Sources — Level 1

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

RE.DC Domain Consideration and Knowledge

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

RE.DC.a1 Consider System Domain Restrictions Level 1

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

Supporting action(s)

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

RE.DC.a2 Consider System's Technical Infrastructure Level 1

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

Supporting action(s)

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

RE.DC.a3 Consider Co-existing Business Processes Level 1

The elicitation process should take into account other business processes which are supported by the system being developed in order to reveal process requirements and constraints imposing on the system.

Supporting action(s)

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

RE.DC.a4 Consider System Boundaries Level 1

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

Supporting action(s)

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

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

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

Supporting action(s)

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

RE.EP Elicitation Practices

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

RE.EP.a1 Adapt Elicitation Technique according to Situation Level 2

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

Recommendations

Some of the useful techniques you can choose to use:

- Observation
- Interview
- Scenario analysis
- Personas
- Product reviews

RE.EP.a2 Consider Quality Requirements Level 2

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

RE.EP.a3 Create Artifacts to Facilitate Elicitation and Analysis — Level 2

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

Recommendation

You can use Scenario analysis to perform this action.

RE.EP.a4 Let Business Concerns/Product Strategies guide Focus of Elicitation Efforts — Level 2

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

Supporting action(s)

- OS.S.a1 Define Product Strategies

RE.EP.a5 Qualify and Quantify Quality Requirements — Level 3

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

Supporting action(s)

- RE.EP.a2 Consider Quality Requirements

RE.EP.a6 Create Elicitation Channels for Requirements Sources — Level 3

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

RE.EP.a7 Reuse Requirements — Level 3

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

RA Requirements Analysis (And Negotiation)

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

RA.a1 Analyze for Missing, Double, Incomplete, Ambiguous Requirements — Level 1

After elicitation phase, the raw requirements need to be analyzed in term of the following aspects:

- Completeness: whether there is something missing or overlapped
- Correctness: whether the requirements are correct
- Testability: whether the requirements are clear enough

This step is to uncover the incomplete, incorrect and inadequate requirements so that a clarification could be made with the source of requirements to obtain the correct desires or expectations of the stakeholders.

RA.a2 Perform Systematic Requirements Prioritization at Project-level — Level 1

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

Recommendations

You can choose to use Prioritization techniques such as

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

- Quality Function Deployment (QFD)

Supporting Action(s)

- RP.S.a2 Perform Requirements Prioritization at Pre-project level based on value, cost and effort
- RP.S.a3 Consider additional advantageous dimensions for prioritization

RA.a3 Perform Requirements Risk Analysis — Level 2

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

RA.a4 Analyze Requirements Functional Dependencies — Level 2

It is important during the analysis to consider the relations among requirements. This is based on the fact that requirements are usually not singular. There are requirements that require the others requirements to be implemented before, or exclude the others. Hence, having an overview of these relations will help you in later phase, e.g. in release planning, change control, etc. The usual considered relations are so-called functional dependencies which include:

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

RA.a5 Identify irrelevant requirements for early dismissal (in/out scope OR Triage) — Level 2

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

Recommendations

Using Model for Early Requirements Triage and Selection (MERTS) as a tool for requirements early dismissing in case there is no specific customer.

Supporting action(s)

- RE.DC.a4 Consider system boundaries
- OS.S.a3 Define Organizational Strategies
- OS.S.a1 Define Product Strategies

RA.a6 Analyze Value-related Dependencies between Requirements Level 2

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

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

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

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

RA.a7 Perform refinement and abstraction of requirements Level 3

Requirements commonly come from many sources with different ways of thinking/expecting, hence they are usually diverse in levels of abstraction. For example, one stakeholder requires "System must look user-friendly" while others ask for something like "Waiting time does not exceed 5 seconds". By performing refinement/abstraction to synchronize their abstraction levels, it will be easier for understanding, managing and further activities such as prioritization. It is also recommended to keep multiple levels of abstraction, for instance: high level which consists of more general and goal-like requirements, and low level which contains more specific requirements for implementation.

Recommendations

You can make use of Requirements Abstract Model (RAM) to perform this step. This model is validated in industry and very useful for this purpose.

RP Release planning

Release planning covers crucial steps aiming to determine the optimal set of requirements for a certain release to be implemented at a defined/estimated time and cost to achieve some goals. Performing this step carelessly would lead to high risky situations or fail to achieve planned goals. For example, placing important features at a too late release would make the product miss the right moment to gain the customers' impression.

RP.a1 Synchronize Release Plan with Product Roadmap Level 2

Product roadmap is important to support the planners in determining the contents of a release. By aligning the requirements with the product plan (including strategies and time) in the roadmap, the planners could easily consider whether the requirements should be included or excluded in a certain release.

Example: If the roadmap states that the upcoming release should target at Chinese market; requirements investigated from Chinese market such as Chinese language feature, Chinese keyboard feature, etc. should be of higher priorities.

In addition, product-technology road map gives planners an overview of the relationship among product releases and their evolvement along the time axis. Hence, it is beneficial for planners to use product-technology road map along their decision process to decide which requirements need to be postponed or excluded, etc. in a certain release due to technology constraints.

Vice versa, the planners should also consider how new features will impact the existing product.

Supporting action(s)

- OS.RR.a2 Define Roles and Responsibilities for Release Planning Activities
- OS.S.a2 Define Product Roadmaps

RP.a2 Post Requirement Selection Evaluation Level 3

Post-release evaluation is the step to assess the quality of the requirements selection in the previous step. This is to uncover previous mistakes/misunderstanding, gain experiences in decision making, hence ensure the quality of decisions for future releases.

The evaluation can be done by analyzing the measurements such as customer value, market penetration, profit, and revenue etc. of different product releases after they were released. Based on this information, the planners could see whether he/she had made a correct decision at that time and further investigate the mistakes.

Recommendations

PARSEQ (Post- Release Analysis of Requirements Selection Quality) is an industrial-validated method supporting this step.

Supporting action(s)

- OS.RR.a2 Define Roles and Responsibilities for Release Planning Activities

RP.a3 Plan multiple releases at pre-defined interval Level 3

Although it is obvious that not all of the potential requirements could be included in one release, it is always recommended to show the plan to implement them in next few releases. The reason is that, requirements present customers' desires, and excluded requirements indicate that customers might be disappointed. Hence, having few releases ahead enable sale personnel to be able to show that they may be compensated in next releases. This is very important for customer-relationship development.

The planning should be undertaken at pre-defined interval since Market-driven planning is more like a trade-off between current state and future. Therefore, regularly reviewing the plans will give more chances to evaluate the decisions, hence re-plan to adapt to the current situation.

RP.a4 Involve different perspectives in release planning Level 2

As mentioned above, release planning itself is a trade-off among customer-value, financial value, developing cost, risk, etc. The nature of it clearly shows a need of involving different perspective in the process. The four perspectives: product management, marketing, development and finances preferably participate in this step. It is also recommended that the different perspectives could form a cross-functional team and work together along the product life cycle in order to achieve a mutual understanding and improve the decision making quality.

One way to perform this diverse involvement is to get different perspectives participate in prioritization and give them appropriate weights based on their importance.

Supporting action(s)

- RE.SI.a1 Identify and Involve Relevant Stakeholders
- OS.RR.a2 Define Roles and Responsibilities for Release Planning Activities

RP.S Requirements Selection

RP.S.a1 Pack requirements into release Level 1

Requirements after being prioritized will be selected to certain releases. The selection activity usually requires the involvement of different perspectives from marketing, developing and management, etc. Besides, you should also consider the interdependences when packing requirements into release. There are requirements with low priorities but mandatory for other higher priority ones, hence taking the relations into account is very important. Currently, there is no tool that fully support for this activity even it is considered a very challenging one.

Supporting action(s)

- OS.RR.a2 Define Roles and Responsibilities for Release Planning Activities
- RP.S.a2 Perform Requirements Prioritization at Pre-project level based on value, cost, effort
- RP.S.a3 Consider additional advantageous factors for prioritization

- RA.a4 Analyze Requirements Functional Dependencies
- RA.a7 Analyze Value-related Dependencies between Requirements

RP.S.a2 Perform Requirements Prioritization at Pre-project level based on value, cost and effort — Level 2

Requirements prioritization at pre-project level helps to determine the relative necessity of the requirements. With a huge number of mandatory requirements which are impossible to be implemented all at the same time, it is crucial to specify which are more critical than others.

In addition, requirements need to be prioritized along more than one dimension (related or even opposing ones). And these dimensions can be valued differently by different stakeholders. Usually, customer-value, cost and effort are considered as the basic dimensions. Customer-value present customer preference of the requirements while cost and effort present how much would be spent to implement the requirements (in finance and man month).

Recommendations

Several prioritization techniques are available and validated for the engineers to choose, namely:

- Pair-wise comparisons
- Prioritization working groups
- Scale of 1-to-10 rankings
- Voting schemes (e.g., give each stakeholder a specific number of votes to
- distribute amongst the requirements or classes of requirements being prioritized)
- Weightings (e.g., weight the votes of different stakeholders)
- Value-Based Software Engineering [Boehm 2003]
- WIN-WIN [Boehm 2001]
- Quality Function Deployment (QFD)

RP.S.a3 Consider additional advantageous dimensions for prioritization — Level 3

Apart from the aforementioned dimensions, the prioritization can also take into account additional ones such as interdependencies, business value, risk, harm avoidance, legal mandate, etc.

Usually the result of prioritization is served as an input for requirements selection. Hence, the more aspects are considered, the more carefully the selection can be performed which can result in a better decision.

DS Documentation and Requirements Specification

Documentation and Requirement specification deal with how a company organizes requirements and other knowledge gathered during requirements engineering process into consistent, accessible and reviewable documents. The software requirements specification (SRS) contains the product's detailed functional and quality requirements.

DS.a1 Define Requirements Attributes Level 1

Each requirement is specified with a number of attributes associated with it. Attributes are assigned values to reflect what is known about the requirement such as estimated cost, priority, state. Different attributes are specified and utilized depending on the various needs of the companies. The benefit of having attributes is to separate important pieces of information about a requirement from its description. Hence, companies can manage requirements more effectively and efficiently by looking at different properties of the requirements. In case requirements are stored in a database, managers can use tool support to simply query, sort or filter the requirements.

Recommendation

Some of the attributes that should be present are ID, Title, Description, Requirement Source, Status and Rationale.

Supporting action(s)

- PM.a1 Introduce Tool Support for Requirements Engineering

DS.a2 Establish Standardized Structure for SRS Level 1

Companies should define a common standard structure which reflects the best practice to organize the requirements document in the companies. The best structures vary among companies as they are influenced by the custom of companies, the type of products developed and the development processes. The common structure helps users to understand the document faster and assure high quality of documents.

DS.a3 Define Requirements States Level 2

The states of the requirements represent their refinement levels in the progress towards release. Some possible states are New (requirement is issued), Selected (requirement is analyzed and selected for implementation), Implemented (requirement is successfully realized), Rejected (requirement is excluded). Tracking requirements states help to monitor the requirements and project progress more accurately.

Supporting action(s)

- DS.a1 Define Requirements Attributes
- PM.a1 Introduce Tool Support for Requirements Engineering

DS.a4 Document Requirements Rationale Level 2

The reason why a requirement is included should be recorded in order to for the readers to understand the requirements. It is extremely useful in case those who initially defined the requirements have left the company. Moreover, it will help problem expert to check if the requirements are consistent with the problem being solved.

Supporting action(s)

- DS.a1 Define Requirements Attributes
- RE.DC Domain Consideration and Knowledge

DS.a5 Record Rationale for Rejected Requirements Level 3

When requirements are rejected after analysis or negotiation, the reason for rejection and who rejected it should be recorded immediately to avoid being forgotten. This information will be helpful for future reference when dismissed requirements resurface as they can be checked without spending effort in re-analysis.

Supporting action(s)

- DS.a1 Define Requirements Attributes
- DS.a3 Define Requirements States
- RA.a6 Identify irrelevant requirements for early dismiss (in/out scope OR Triage)
- RP.S.a1 Pack requirements into release

DS.DD Documentation Deliverables

The documentation deliverables imply all deliverables the requirements engineering process supposes to produce. You should clearly define the expected deliverables of RE process at the beginning of it as requirements for the project itself. This activity can also be done together with members from other phases that will use the information later.

DS.DD.a1 Define User Documentation Deliverables Level 2

The user deliverables may consist of user manual, user dictionary, etc. This aims to describe the system from user points of view and how to use it.

DS.DD.a2 Define System Documentation Deliverables Level 2

Management deliverables cover all the necessary documents for managing the system such as: system design, technical specification, etc.

DS.DD.a3 Define Management Documentation Deliverables Level 3

Management deliverables cover all the necessary documents for managing the system such as maintenance, administrative manual, etc.

QA Quality Assurance

This process involves checking the requirements against defined quality standards and the real needs of various stakeholders. It ensures that the documented requirements are complete, correct, consistent, and unambiguous.

QA.a1 Use Checklist to Ensure Quality of Requirements OG1.a1 Level 1

Checklist draws the participants' attention to the certain aspects of the requirements document as well as the frequently encountered problems. Checklist should not be too long to prevent people from referring to the list too often.

QA.a2 Validate requirements with relevant stakeholders Level 1

Requirements must be validated with the relevant stakeholders in order to ensure their consistency, completeness and adequacy. Moreover, the intent and interpretation of the requirements can also be verified.

QA.a3 Review Requirements OG1.a2 Level 2

Reviewing is the technique involving peers (someone other than the author) to examine the requirements and identify defects. The author is then responsible for correcting the found problems.

QA.a4 Create Preliminary Artifacts for Quality Assurance OG1.a3 Level 3

Creating possible test cases or writing a draft user manual based on the requirements document can uncover problems related to ambiguities, inconsistencies or usability. The test cases or draft user manual can be used later as a basis for actual artifacts.

QA.a5 Organize Inspections to Ensure Quality of Requirements OG1.a4 Level 3

Inspections are formal meetings in which a small team of inspectors with different perspectives (e.g. customer, analyst, developer, and tester) carefully examine the requirements, detect errors and resolve them together. This technique can identify a high percentage of requirements errors but requires all parties to be present at the same time.

QA.a6 Use system model paraphrasing for Quality Assurance OG1.a5 Level 3

Converting system models into natural language enables general stakeholders to understand these models more clearly and comment on them. In this way, additional

requirements or problems will be detected.

Supporting action(s)

- RE.EP.a4 Create Artifacts to Facilitate Elicitation and Analysis

Appendix H: Uni-REPM - Maturity Level view

In this section, the model can be viewed by maturity level. This view shows the practices from all process areas which the organization should implement in order to achieve a specific maturity level.

LEVEL 1 - DESTINATION			
OS	**1**	**Organizational Support**	
OS.a1	1	Assign Owner of Requirements Process	1
OS.a2	1	Create a Product-wide Glossary of Terms	1
OS.RR	1	Roles and Responsibilities	
OS.S	1	Strategic	
PM	**2**	**Requirements Process Management**	
PM.a1	2	Introduce Tool Support for Requirements Engineering	1
PM.a2	2	Define and Maintain a Requirements Management Process	1
PM.CM	2	Configuration Management	
PM.CM.a1	2	Manage Versions of Requirements	1
PM.CM.a2	2	Baseline Requirements	1
PM.RC	2	Requirements Communication	
PM.RC.a1	2	Establish effective communication with requirements issuers	1
PM.RT	2	Requirements Traceability Policy	
PM.RT.a1	2	Uniquely Identify each Requirement	1
PM.RT.a2	2	Document Requirements' Source	1
RE	**3**	**Requirements Elicitation**	
RE.SI	3	Stakeholder and Requirements Source Identification	
RE.SI.a1	3	Identify and Involve Relevant Stakeholders	1
RE.SI.a2	3	Distinguish between Customers, End-Users, and In-house Stakeholders	1
RE.SI.a3	3	Identify Other Requirements Sources	1
RE.DC	3	Domain Consideration and Knowledge	
RE.DC.a1	3	Consider System Domain Restrictions	1
RE.DC.a2	3	Consider System's Technical Infrastructure	1
RE.DC.a3	3	Consider Co-existing Business Processes	1
RE.DC.a4	3	Consider System's Business Process	1
RE.DC.a5	3	Consider System Boundaries	1
RE.EP	3	Elicitation Practices	
RA	**4**	**Requirements Analysis (and Negotiation)**	
RA.a1	4	Analyze for Missing, Double, Incomplete, Ambiguous Requirements	1
RA.a2	4	Perform Systematic Requirements Prioritization at In-project level	1
RP	**5**	**Release Planning**	
RP.S	5	Requirements Selection	
RP.S.a1	5	Package Requirements into Releases	1
DS	**6**	**Documentation and Requirements Specification**	
DS.a1	6	Define Requirements Attributes	1
DS.a2	6	Establish Standardized Structure for SRS	1

DS.DD	6	Documentation Deliverables	
QA	**7**	**Quality Assurance**	
QA.a1	7	Use Checklist to Ensure Quality of Requirements	1
QA.a2	7	Validate requirements with relevant stakeholders	1

LEVEL 2 – INTER-MEDIATE			
OS	**1**	**Organizational Support**	
OS.RR	1	Roles and Responsibilities	
OS.RR.a1	1	Define Roles and Responsibilities for Requirements Engineering Process	2
OS.RR.a2	1	Define Roles and Responsibilities for Release Planning Activities	2
OS.RR.a3	1	Define Roles and Responsibilities for Change Control	2
OS.S	1	Strategic	
OS.S.a1	1	Define Product Strategies	2
OS.S.a2	1	Define Product Roadmaps	2
PM	**2**	**Requirements Process Management**	
PM.a3	2	Train personnel in Requirements Management Process and Specialty (e.g. Prioritization…)	2
PM.a5	2	Involve various perspectives in Requirement Engineering Process	2
PM.CM	2	Configuration Management	
PM.CM.a3	2	Define a Process for Managing Change and Evolution	2
PM.CM.a4	2	Track change requests	2
PM.RC	2	Requirements Communication	
PM.RT	2	Requirements Traceability Policy	
PM.RT.a3	2	Document Requirements' Relation	2
PM.RT.a4	2	Document Impact of Requirement on Other Artifacts	2
PM.RT.a5	2	Define traceability policies	2
RE	**3**	**Requirements Elicitation**	
RE.SI	3	Stakeholder and Requirements Source Identification	
RE.DC	3	Domain Consideration and Knowledge	
RE.DC.a6	3	Consider Sociopolitical Influences on Requirements Sources	2
RE.EP	3	Elicitation Practices	
RE.EP.a1	3	Adapt Elicitation Technique according to Situation	2
RE.EP.a2	3	Consider Quality Requirements	2
RE.EP.a3	3	Create Artifacts to Facilitate Elicitation and Analysis	2
RE.EP.a4	3	Let Business Concern/Product Strategies guide Focus of Elicitation Efforts	2
RA	**4**	**Requirements Analysis (and Negotiation)**	
RA.a3	4	Perform Requirements Risk Analysis	2
RA.a4	4	Analyze for Requirements Functional Dependencies	2
RA.a5	4	Identify irrelevant requirements for early dismiss (in/out scope OR Triage)	2
RA.a6	4	Analyze Value-related Dependencies between Requirements	2
RP	**5**	**Release Planning**	
RP.a1	5	Synchronize Release Plan with Product Roadmap	2
RP.a4	5	Involve different perspectives in release planning	2

RP.S	5	Requirements Selection	
RP.S.a2	5	Perform Systematic Requirements Prioritization at Pre-project level based on value, cost, effort	2
DS	**6**	**Documentation and Requirements Specification**	
DS.a3	6	Define Requirements States	2
DS.a4	6	Document Requirements Rationale	2
DS.DD	6	Documentation Deliverables	
DS.DD.a1	6	Define User Documentation Deliverables	2
DS.DD.a2	6	Define System Documentation Deliverables	2
QA	**7**	**Quality Assurance**	
QA.a3	7	Review Requirements	2

LEVEL 3 - DESTINATION			
OS	**1**	**Organizational Support**	
OS.RR	1	Roles and Responsibilities	
OS.RR.a4	1	Define Roles and Responsibilities for Product Management Organization	3
OS.S	1	Strategic	
OS.S.a3	1	Define Organizational Strategies	3
OS.S.a4	1	Communicate Strategies in Organization	3
PM	**2**	**Requirements Process Management**	
PM.a4	2	Early connect portfolio considerations into requirements engineering process	3
PM.CM	2	Configuration Management	
PM.RC	2	Requirements Communication	
PM.RC.a2	2	Obtain common understanding of requirements among different involving teams	3
PM.RT	2	Requirements Traceability Policy	
RE	**3**	**Requirements Elicitation**	
RE.SI	3	Stakeholder and Requirements Source Identification	
RE.DC	3	Domain Consideration and Knowledge	
RE.EP	3	Elicitation Practices	
RE.EP.a5	3	Qualify and Quantify Quality Requirements	3
RE.EP.a6	3	Create Elicitation Channels for Requirements Sources	3
RE.EP.a7	3	Reuse Requirements	3
RA	**4**	**Requirements Analysis (and Negotiation)**	
RA.a7	4	Perform refinement and abstraction of requirements	3
RP	**5**	**Release Planning**	
RP.a2	5	Post Requirement Selection Evaluation	3
RP.a3	5	Plan multiple release at pre-defined interval	3
RP.S	5	Requirements Selection	
RP.S.a3	5	Consider additional factors for prioritization	3
DS	**6**	**Documentation and Requirements Specification**	
DS.a5	6	Record Rationale for Rejected Requirements	3
DS.DD	6	Documentation Deliverables	
DS.DD.a3	6	Define Management Documentation Deliverables	3

QA	7	Quality Assurance	
QA.a4	7	Create Preliminary Artifacts for Quality Assurance	3
QA.a5	7	Organize Inspections to Ensure Quality of Requirements	3
QA.a6	7	Use System Model Paraphrasing for QA	3

Appendix I: Uni-REPM Checklist

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

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

RE.SI.a1	Do you identify and involve different stakeholders in elicitation? For example, customers, end users, testers, developers, marketing etc.				
RE.SI.a2	Different types of stakeholders have their own demands and needs. Do you distinguish between customers (who pay for the system), end-user (who use the system) and in-house stakeholders (who support or create the system)?				
RE.SI.a3	Besides the above stakeholders, do you elicit requirements from other sources (e.g. partners, distributors, company standards, regulations etc)?				
RE.DC	Domain Consideration and Knowledge				
RE.DC.a1	Do you systematically elicit information about restrictions or possibilities that the domain may impose on your product?				
RE.DC.a2	Do you consider the technical infrastructure when developing the system?				
RE.DC.a3	When eliciting requirements, do you consider how your system will contribute to the business process in customer's organization?				
RE.DC.a4	Do you take into account the co-existing business processes which the system should support?				
RE.DC.a5	Do you elicit the information about what is part of your system and what is outside of the scope? Do you use this information to enable yourself to focus on what is within the system boundaries?				
RE.EP	Elicitation Practices				
RE.EP.a1	Do you determine which quality aspects of the system to focus on (such as performance, usability, reliability etc) and explicitly elicit requirements about them?				
RE.EP.a2	Do you describe quality requirements in details such as max, min, average value?				
RE.EP.a3	Do you use the business objectives to guide how you conduct your elicitation efforts?				
RE.EP.a4	Depending on each situation, certain elicitation techniques are more appropriate to use than others. Do you consider different techniques to suit each case?				
RE.EP.a5	When appropriate, do you use additional artifacts such as prototyping or scenario to aid in the elicitation and analysis process?				
RE.EP.a6	Do you create different channels to capture all forms of requirements from various sources?				

RE.EP.a7	Do you have a systematic process to reuse requirements from other systems developed in the same application area?				
RE.DC.a6	Are you aware of and have ways to deal with the political or organizational influence on the requirements sources when eliciting requirements?				
RA	**Requirements Analysis**				
RA.a1	Do you systematically analyze whether you have double requirements or whether there are requirements missing? Do you systematically analyze whether your requirements are ambiguous? Do you systematically analyze whether your requirements are incorrect? Do you systematically analyze whether your requirements are testable?				
RA.a2	Do you have a systematic way of prioritizing the requirements so that you know which to focus on? Do you priorities based on several viewpoints (e.g. cost, value, risk, penalty)? Do you priorities with the help of your stakeholders? Do you weigh the importance of different stakeholders?				
RA.a3	Do you systematically assess the risks of individual requirements or set of requirements?				
RA.a4	Do you systematically estimate whether there are any dependencies or relations between requirements?				
RA.a5	Do you have a process for selecting, at an early stage, which requirements to focus on and which to discard immediately, so that you spend your resources on the right requirements?				
RA.a6	Do you estimate and document how much requirements may impact or increase or decrease the value of other requirements?				
RA.a7	Do you classify and group requirements in to different categories based on their goals or levels of abstraction?				
RP	**Release Planning**				
RP.a1	Is your release plan in line with your product roadmap? Are there areas in your product roadmap where you have no or insufficient requirements? Do you also propose to change this part of the roadmap?				
RP.a2	Do you evaluate whether you actually selected the right requirements for certain release?				
RP.a43	How far ahead does your planning stretch? Are you able to foresee and start preparing for subsequent releases already now?				
RP.a4	Do you consult your stakeholders in prioritization and decision making (RP)? Do you weigh the importance of different stakeholders?				

RP.S	Requirements Selection				
RP.S.a1	Do you clearly define a release with all the necessary information before passing it to the next development stage?				
RP.S.a2	Do you priorities based on several viewpoints (e.g. cost, value)?				
RP.S.a3	Do you consider several additional factors for prioritization (eg. risk, interdependencies, penalty and so on)?				
DS	**Documentation and Requirements Specification**				
DS.a1	Do you follow a standardized structure of how the System Requirements Specification should be written, or generated?				
DS.a2	Do you define attributes for each requirements such as ID,title, descriptions, author etc?				
DS.a3	Do you define and store states that requirements should follow during their lifetime? For example, new, assigned, rejected, implemented, tested, delivered.				
DS.a4	Do you document the reason why the requirement is specified and what function the requirement has?				
DS.a5	Do you record which requirements were rejected and why so as to avoid re-analysis if the same requirements reappear later?				
DS.DD	Documentation Deliverables				
DS.DD.a1	Do you define what user manuals and other user documentation that shall be delivered together with your product?				
DS.DD.a2	Do you define system documentation that shall be delivered together with your product?				
DS.DD.a3	Do you define management documentation that shall be delivered together with your product?				
RV	**Requirements Validation**				
RV.a1	Do you have and systematically use a checklist for ensuring the quality of your requirements?				
RV.a2	Do you validate requirements with relevant stakeholders to ensure that they reflect the correct intent?				
RV.a3	Do you conduct requirements reviews to ensure their quality?				
RV.a4	Do you create artifacts (e.g. user manuals or test cases) to assist you in ensuring the quality of your requirements?				
RV.a5	Do you organize inspections to review the requirements with other stakeholders?				

RV.a6	Do you convert system models into natural language in order to detect requirements errors?				