

## MASTER

### Practical assessment of business and IT requirements for offshoring

Amrita, A.; Sree Kumar, A.

*Award date:*  
2009

[Link to publication](#)

#### **Disclaimer**

This document contains a student thesis (bachelor's or master's), as authored by a student at Eindhoven University of Technology. Student theses are made available in the TU/e repository upon obtaining the required degree. The grade received is not published on the document as presented in the repository. The required complexity or quality of research of student theses may vary by program, and the required minimum study period may vary in duration.

#### **General rights**

Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain

# **Practical assessment of business and IT requirements for offshoring**

Amrita  
Anjali Sree Kumar

Master's thesis August 18, 2009  
Department of Mathematics & Computer  
Science  
Software Engineering & Technology Group  
Eindhoven University of Technology

## Practical assessment of business and IT requirements for offshoring

Amrita

Anjali Sree Kumar

Department of Mathematics & Computer Science  
Software Engineering & Technology Group  
P.O. Box 513, 5600 MB Eindhoven, The Netherlands.

Master's thesis

Abstract

Aiming at lowering software development costs or improving the use of time and effort, current software industry practice consists in offshoring the design and development of the project. Offshored projects are known to often exceed the limits of planning in terms of time or costs or to produce systems that do not meet the functionality requirements imposed by the customer. In this project we consider quality of requirements.

To assess quality of requirements three approaches were used: the **LaQuSo Software Product Certification Model (LSPCM)** (1) developed by the Laboratory for Quality Software (LaQuSo) for certifying the product quality; **Structured Expert Method for Business Analysis (SEMBA)** developed by Capgemini for gathering and elicitation of business requirements; and **Integrated Requirement Management Approach (IRMA)** developed by Capgemini for gathering and elicitation of IT requirements. The research question we address is “How can we improve the quality of requirements in order to be successful in offshoring?”. For this we check the suitability of the methods used for requirement gathering (*i.e.* SEMBA and IRMA) and requirement quality assessment model (*i.e.* LSPCM) for offshoring.

Thus, in this project our focus is to study and critically analyze LSPCM, SEMBA and IRMA by conducting detailed case study on offshore projects undertaken by Capgemini. We focus on three software artifacts pertaining to business and IT requirements: context description, user requirements and high level design.

During the case studies we identified the limitations of LSPCM, SEMBA and IRMA for offshore projects. In order to address these limitations we have documented the set of checks which have to be incorporated in LSPCM and in guidelines of SEMBA and IRMA.

## Acknowledgements

This document presents our master's thesis for the Computer Science & Engineering program at the Eindhoven University of Technology (**EUT**). The research was done between February 2009 and August 2009 under the Software engineering technology (SET) group of the Computer Science Department. The work was carried out at LaQuSo (Laboratory for Quality Software), Eindhoven University of Technology and Project center of Capgemini B.V., Utrecht.

First of all, we would like to thank Manipal University (**MU**) and EUT for providing us the opportunity for pursuing dual degree master program. We sincerely extend our gratitude to our supervisors from EUT dr. Alexander Serebrenik and ir. Martijn Klabbers. We enjoyed discussing the technical problems we encountered and we are grateful for all the suggestion and guidance they gave us. We are grateful to our supervisor from MU, Dr.(Prof.) Manohara Pai M.M. for providing necessary guidance.

We are thankful to our supervisors at Capgemini, Ms. Cornelly Spier and Ms. Marion Verwey. During our internship they helped us when and where needed. We furthermore appreciate the efforts of Rob Ista, Peter Kuijhoven and Jan Klabbers of Capgemini, for making this internship possible with their practical input.

We would like to thank prof. dr. Mark van den Brand, dr. Alexander Serebrenik, prof.dr. Kees van Hee, Ms. Tonny Wildvank and Ms. Cornelly Spier for being the members of our assessment committee.

Many people from LaQuSo as well as from Capgemini have helped us to successfully complete this project, and we are greatly thankful to all of them.

We would also like to thank all our dear friends for their support during our work on this master's project.

# Contents

1. Introduction .....	5
1.1. Background information .....	5
1.2. Offshoring .....	8
1.3. Context .....	11
1.4. Research question.....	12
1.5. Approach.....	13
1.6. Document Overview .....	13
2. The techniques used and related work.....	15
2.1. LaQuSo Software Product Certification Model (LSPCM) .....	15
2.2. Structured Expert Method for Business Analysis (SEMBA).....	17
2.3. Integrated Requirements Management Approach (IRMA) .....	21
2.4. Related work .....	28
2.5. Introduction to RUP .....	32
3. Case studies .....	34
3.1. Case study 1: Application software for an insurance company.....	35
3.2. Case study2: Unified ERP for an organization.....	55
4. Reflection .....	67
4.1. Reflection on LSPCM .....	67
4.2. Reflection on SEMBA .....	71
4.3. Reflection on IRMA .....	75
5. Conclusion.....	79
5.1. Problem & Solution.....	79
5.2. Main result.....	80
5.3. Future work.....	80
Bibliography .....	82

# 1. Introduction

Software industry like any other industry mainly focuses on providing better solutions for ever evolving problems in the world. Better solutions demand improvement of the quality of delivered products. The quality of any product depends on the quality of the raw materials used for its manufacturing. In the case of software the quality of end product depends on the quality of the requirements. This is because, it is always the requirements that reflect a need for change and become the initial ingredient of a software project. So it is essential to have a good requirements document in order to get the right end product.

Requirements can be classified into different categories according to the kind of information they deal with. In this work we focus on business and IT requirements. Though in-depth studies have been carried out in this area, it has been observed that in practice at various software companies things go wrong from the stage of requirements gathering. These wrong practices lead to a system which does not conform to customer needs. Here, we attempt to improve the quality of requirements by suggesting improvements for requirements gathering methods and requirements quality assessment model. The scope of this research is confined to offshored projects.

## 1.1. *Background information*

### *Software Project Failure*

A software project is concluded as failed in the following scenario:

- The software did not meet the customer requirements.
- The software did not meet its delivery deadline.
- The actual cost of software is significantly higher than its estimated cost.
- The software had too many errors.

**Statistical facts:**

- The Standish Group reported in 2003 (2 p. 22) that only 34% of computer projects succeed, while 27% experienced cost overruns of over 20%, and 15% of projects failed altogether. On average, projects overran their original schedules by 82%, but delivered only 52% of required features and functionality.
- Few remarkable software failures in the history (2 pp. 22-23):

<b>System failures</b>	<b>Reason for failure</b>	<b>Loss in term of money</b>
<b>Failure of FBI Terror and Crime System - 2005</b>	Mismanagement, poor planning and design changes	\$170 million
<b>Failure of power monitoring and management system - April 2004</b>	A software bug	\$6 billion
<b>Failure of U.K. Pension Systems - November 2004</b>	A software bug	\$863 million
<b>Failure of student loan interest calculation system - April 2003</b>	A software bug in calculating the monthly payments on 800,000 loans.	\$8 million in interest.
<b>Failure of new ERP system of FoxMeyer pharmaceutical company – 1998</b>	A software bug	\$5 billion

- Gartner, Inc. a leading information technology research and advisory company assessed the frequency of software project failure rates in the year 2008. According to the 2008 Gartner survey (3), projects considered failures occurred at rates of:
  - 14% for small projects
  - 20% for medium projects
  - 22% for large projects
  - Averaging about 20% for all types of projects.



The reasons for these failures have been widely discussed in the literature (2 p. 24) (4) (5 p. 88) (6) (7). For example, the 2008 Gartner survey (3) says that the **25% of projects failed due to functionality issues**, 15% of projects failed because of high cost variance, 20% were canceled during delivery, and 18% were unsuccessful because they were substantially late.

In order to address these problems, (3) suggests that the project team should have following issues in place:

- Get the requirement **right**.
- Keep scope in control.
- Estimate the effort and schedule correctly.

This means, we need to assess the quality of captured requirements.

### *Requirements Quality*

Before defining the quality of requirements we must understand the term *requirement* itself. For this project we consider two types of requirements:

- **Business requirements:** Business requirements reflect the business needs of the customer. Business requirements are the outcome of business analysis.
- **IT requirements:** IT requirements define what an IT system should do for the customer. IT requirements are the outcome of system analysis.

American society for quality (8) defines quality as “A subjective term which each person has his or her own definition. In technical usage, quality can have two meanings:

- The characteristics of a product or service that bear on its ability to satisfy stated or implied needs;
- A product or service free of deficiencies.”

This definition makes it difficult to quantify quality. How to assess whether a product is free from deficiencies? How do we measure that the collected business and IT requirements are free from deficiencies?

## 1.2. *Offshoring*

There is no official definition for “offshoring”. The term “offshoring” has been used for several other types of business activities such as foreign investment activities (9). The U.S. Government Accountability Office, Bureau of economic analysis, World trade organization and many other organizations have devised their own definitions for offshoring.

For this research work we confine ourselves to the following definition of IT offshoring:

“IT offshoring is a provision of assigning the software development to an external supplier positioned in a country that is geographically remote from the client enterprise.”

Different companies use different parlance for “offshoring” such as farshoring, rightshoring, etc. The decision to offshore a project is often made in order to **lower development or design costs** or to **improve use of time and effort** or to **obtain resource needed to run or extend the company IT business**.

The studies carried out by Compass Consulting, Forrester, and A. T. Kearney state that the “**59% of all outsourcing contracts fail**” (10 p. 4). This indeed shows that offshoring is unable to fulfill its objectives mentioned in the above paragraph. An empirical study shows that the predicted cost saving from offshoring is much hyped (10 p. 4).

Then what should be our next step? Should we bring a halt to offshoring or should we find out where we went wrong? A recent ACM report states that offshoring magnifies existing risks and creates new threats (11). From this we can deduce that the offshore software projects are more prone to failure than onsite projects. Hence it is a major concern for software managers constantly seeking ways to increase the success record of their development efforts. To assist these managers one may develop check lists that can be used in order to address the risks related to requirements.

Charalambos and Robbie have identified a total of 25 risk factors involved in any offshored software development (12 p. 90) which are shown in the table below.

Rank	Risk Factor	Rating *
1	Lack of top management commitment	9.2
2	Original set of requirements is miscommunicated	8.1
3	Language barriers in project communications	7.7
4	Inadequate user involvement	7.7
5	Lack of offshore project management know-how by client	7.4
6	Failure to manage end user expectations	7.3
7	Poor change controls	7.3
8	Lack of business know-how by offshore team	7.3
9	Lack of required technical know-how by offshore team	7.2
10	Failure to consider all costs	7.1
11	Telecommunications and infrastructure issues	6.8
12	Vendor viability	6.0
13	Difficulties in ongoing support and maintenance	6.0
14	Low visibility of project process	5.8
15	Cross-cultural differences	5.8
16	High turnover of vendor employees	5.8
17	Constraints due to time-zone differences	5.8
18	Lack of continuous, face-to-face interactions across team members	5.7
19	Threats to the security of information resources	5.3
20	Negative impact on employee morale	5.2
21	Unfamiliarity with international and foreign contract law	4.8
22	Differences in development methodology/processes	4.8
23	Political instability in offshore destinations	4.4
24	Negative impact on image of client organization	3.1
25	Currency fluctuations	2.8

\*The rating is given as follows: 10=very important; 7=important; 4=slightly important; and 1=unimportant

**Table 1: Risk Factors** (12 p. 90)

The rating in above table is given by a group of experts. These experts are certified project management professionals (PMP), who are senior IT executives and members of the Project Management Institute (PMI). On average, each expert had over 17 years of IT-related experience and over 15 years of project management experience, and had managed 51 projects, nine of which were offshore development ones. This level of expertise suggests that the selected experts were well qualified to rate the risk factors.

As we can observe from the above table, the miscommunication of requirements has been ranked as the second most important risk factor. Hence we can clearly see the importance of requirements management in offshored projects.

As indicated in the previous discussion on software project failures software often fails to deliver the expected functionality *i.e.* it **fails due to functionality issues**. Failures due to functionality means that the functionality of software system is not conforming to customer needs, *i.e.*, requirements were miscommunicated. To identify possible sources of miscommunication we focus on the parties involved in the software development process:

1. Customer,
2. Development team(s),
3. Maintenance organization(s).

Miscommunication among any of them can lead to project failure. Clearly the chance of miscommunication between the customer and the offshored development team is the highest among all the combinations. This is mainly because of the geographical and cultural separation between these parties. We believe that in order to bridge this gap, clear and consistent specification of the requirements with elaborate explanations is necessary. Clear and consistent specification increases the chances for success of the project by delivering the system made from the good requirements. To this end we devise **relevant checks** for the LSPCM model which would help us in deciding whether the requirements documents prepared onsite have been sufficiently elaborated such that the customer's needs are effectively communicated to the offshored development team. Hence we can have a certification model constituting set of

checks that will help in carrying out the quality assessment. The proposed certification model must be applicable for business and IT requirements.

### **1.3. Context**

To gain a clear theoretical knowledge on this research subject, we took the assistance of university research material which included the study of a product certification model (LSPCM) developed by LaQuSo. Since our focus is mainly on the practical implications of such theoretical concepts and findings, we obtained industrial project documentations from Capgemini in order to carry out the case studies.

Hence this project was carried out at Capgemini and LaQuSo. Capgemini was our industrial partner for providing essential information related to requirements engineering methods followed at Capgemini and current industrial software projects which had an offshoring component. The experts at LaQuSo supervised us throughout our work and analyzed critically the findings with respect to previous work on related areas.

#### **1.3.1. LaQuSo**

The Laboratory for Quality Software (LaQuSo) is an activity of the Department of Mathematics and Computer Science of EUT and, simultaneously, a sub department within the Department of Science, Mathematics and Computing Science of Radboud University Nijmegen. It is involved in verification and validation of software systems and their intermediate artifacts. LaQuSo takes state-of-the-art methods, techniques, and a tool on software product quality analysis developed during more fundamental/strategic research at Eindhoven University of Technology and Radboud University Nijmegen and applies, validates, and deploys these in industry and government.

#### **1.3.2. Capgemini**

Capgemini is a major French company. It is one of the world's largest information technology, transformation and management consulting, outsourcing and professional services companies.

Capgemini helps clients to deal with changing business and technology issues. It is a 3-tier solution provider:

- Capgemini consulting,
- Outsourcing services,
- And technology services.

Being a service provider Capgemini has seen a wide range of changes in the ever evolving software industries and adapted itself to suit the market needs. This has made research inevitable within this organization.

**Research and development:** The Project Center is the home to technology services and a repository of all the projects of Capgemini. The primary goals of the Project Center are:

- Increase of margins and revenues of the projects;
- Improvement of the image of Capgemini as the leading supplier of complex projects, based on proven project management, predictability and transparency.

#### ***1.4. Research question***

The goal of our master project is to formulate suggestions on how an assessment model can be improved for business and IT requirements of offshored projects so as to minimize the miscommunication of requirements. This in turn would lead to suggestions for improvement on the approaches followed for gathering these requirements. In order to achieve this goal our first step is to study and critically analyze the certification model of LaQuSo (i.e. LSPCM) and Capgemini methods for business and system analysis (SEMBA and IRMA, respectively). The second step is dedicated to the assessment of projects undertaken by Capgemini using LSPCM, SEMBA and IRMA. At the third step, based on the assessment results we formulate suggestions on how the quality assessment approaches for business and IT requirements can be improved.

To assess business and IT requirements we have considered three software product areas as categorized by LSPCM, which are:

- Context description,

- User requirements,
- And high-level design.

The “*context description*” focuses on the checks for business requirement of a project and the “*user requirements*” together with the “*high-level design*” focuses on the assessment of IT requirements. Hence we have chosen these product areas for this study.

“*How can we improve the quality of requirements in order to be successful in offshoring?*”. For this we check the suitability of the methods used for requirement gathering (*i.e.* SEMBA and IRMA) and requirement quality assessment model (*i.e.* LSPCM) for offshoring (see Section 3.1 and 3.2).

### 1.5. **Approach**

In order to accomplish the defined objectives we have undertaken a threefold approach:

- The first fold focuses on the study of the available certification model for the requirements *i.e.* LSPCM and the approaches used in the industry for requirement gathering and elicitation *i.e.* SEMBA and IRMA.
- The second fold focuses on the application of LSPCM, SEMBA (as a checklist) and IRMA (as a checklist) on two offshore projects recently undertaken by Capgemini. We have chosen a RUP-based and a SAP-package based project in order to analyze both software development and package implementation life cycles.
- The third fold focuses on the analysis of the results obtained in second fold. In this fold we identify the limitations of LSPCM, SEMBA and IRMA and we define the rule set which have to be incorporated in to LSPCM and in guidelines of SEMBA and IRMA.

### 1.6. **Document Overview**

Chapter 2 reviews the related work and identifies specific industrial techniques used for assessing requirements and ensuring their quality. Chapter 3 describes the case studies carried out in order to accomplish our goal. Chapter 4 reflects on the major findings from the case studies with respect to the model and methods studied. Chapter 5 concludes the research work

with problem & solution, main result and future work. Sections related to SEMBA are written by Anjali, sections related to IRMA are written by Amrita and rest of the report is our combined effort.



## 2. The techniques used and related work

The starting point of our study is a comparative assessment of state-of-the-art and state-of-the-practice ways on how software quality is being specified and evaluated. In order to gather relevant information regarding the current state-of-art of this research area we analyzed related work in industry and research. This accumulated information has contributed to formulating the research goal above. The Sections 2.1, 2.2 and 2.3 elaborate on LSPCM, SEMBA and IRMA, respectively and related work is discussed in Section 2.4.

### *2.1. LaQuSo Software Product Certification Model (LSPCM)*

The Laboratory for Quality Software (LaQuSo) has developed the Software Product Certification Model (LSPCM) (1). LSPCM is a rule-based product certification model which identifies the major deliverables of software development process, such as user requirements or code, as product areas (1 pp. 5-6). There are different artifacts in each product area and these artifacts are called elements in LSPCM. For instance, glossary is an element belonging to the user requirements product area. The elements of one product area can be interrelated to the element of another product area.

LSPCM focuses on three certification criteria (1 p. 7):

- Formalness - All the obligatory elements of product area should be present.
- Uniformity - The style of elements in a product area should be standardized.
- Conformance – All the elements should conform to the properties that are subject of certification.

These certification criteria hold for all product areas. Thus all the elements belonging to the product area must be complete, uniform, and conformant to the certification properties.

As suggested by the preceding discussion two types of inputs are required for certification: one or more software artifacts and one or more properties of these artifacts that are to be certified. The properties can be of one of the following categories: (1 p. 3)

- Consistency: Do the different (parts of) software artifacts conform to each other?
- Functionality: Does input to the system produce the expected output?
- Behavior: Does the system meet general safety and progress properties like absence of deadlocks or are constraints on the specific states of the system met?
- Quality: Do the artifacts fulfill non-functional requirements in the areas of for example performance, security, and usability?
- Compliance: Do the artifacts conform to standards, guidelines or legislation?

The model has five certification levels for each product area, but only four are relevant for the certification process outcome. The first is only used for creating a baseline that the certification analysis can start. (1 p. 10)

- Initial – This level is given when all required elements of the product area are present and uniform. (1 p. 10) This is not an official certification level, but just a baseline from where the conformance checks (the heart of the assessment and certification) will be started.
- Manually verified – This level is attained when all elements, relationships and properties have been manually verified. (1 p. 10)
- Automated verified - This level is achieved when elements, relationships and properties have been verified using automated tools. (1 p. 10)
- Model verified - This level is achieved when a model representing elements, relationship and properties has been constructed by an assessor and verified with mathematical methods. (1 p. 10)
- Formally verified - This level is achieved when elements, relationship and properties have been verified with mathematical methods without the explicit model construction step. (1 p. 10)

LSPCM does not include specific checks related to requirement documents of offshored projects. So our work aims at this perspective to the existing version of LSPCM model.

## 2.2. *Structured Expert Method for Business Analysis (SEMBA)*

SEMBA (Structured expert method for business analysis) is a Capgemini method for carrying out business analysis for a project. SEMBA entails the best practices from the industry created as a result of years of experience. This method is at its infancy in terms of its practice within Capgemini. It is expected to ensure clarity of process followed within Capgemini. This level of clarity ensures a deterministic traceability from a design model back to its requirement specification. SEMBA experts suggest using SEMBA in conjunction with the Unified Project Management Method. This is to provide the details for project management processes. SEMBA contains activities from the rational unified disciplines '*Business modeling*' and '*Requirements*'. According to SEMBA, business analysis is concerned with renewing, improving and adapting business processes either with or without the use of IT. Business analysis is done by the business analyst. SEMBA serves as a helping hand to an experienced business analyst by giving guidance to find out the kind of information to be gathered and as a result the documents to be prepared. Therefore, SEMBA has a set of deliverables each of which has a template and a guideline. The templates prescribe the format and structure of a deliverable whereas the guidelines dictate the kind of content to be filled in the structure.

As suggested by SEMBA, given below are the 18 deliverables which can be documented as part of the business analysis for any proposed software system. The priority of each of these deliverables related to its importance for a particular project is determined during the business analysis phase by the analyst. All these documents are not mandatory. This decision is also made while preparing the approach document which is mandatory in SEMBA.

### **APPROACH**

Important goal of the approach is to describe in detail how the Business Analysis (BA) part of the engagement will be delivered. The purpose of this document is to document the agreement on the approach. The approach is also input for the Engagement manager to create the Project Governance Plan. The document is first created and then discussed with the client to reach an agreement on the Approach. The approach does not contain an implementation strategy as that is the main deliverable of the complete BA engagement. The approach does give clarity on

how the engagement will be set about.

#### **AS-IS UNDERSTANDING**

The purpose of this document is to create a common understanding of how the business works today as the basis for future improvements. The Business Context, Business Processes, Information and Applications of the current situation are described coherently. Only that level of detail is described which is needed to design the future situation and how to migrate (refer "*Migration strategy* document") from the current to that future situation (refer "*To-be design* document").

#### **BUSINESS INITIATIVE**

The Client's Initiative and Analysis contain the description of the entire Business Initiative and the analysis that has been done to identify the true problem which has initiated this project and/or core desires of the client. In addition to describing the Business Context and the view on possible solutions, it is also necessary to identify the relevant issues and exclusions or constraints.

#### **BUSINESS REQUIREMENTS**

The purpose of this document is to capture all business requirements regarding the Business Context, Business Processes, Information and Application Landscape. The business requirements are documented according to the Requirements Management Plan.

#### **CHANGE DRIVERS**

Describe the change drivers relevant for the project. The change driver is the primary reason or cause of why the client wants to or needs to change. It also indicates why this change driver is important.

#### **CLIENT OBJECTIVES AND EXPECTATIONS**

The purpose of this document is to define and capture the client's objectives and expectations specific for the engagement. The purpose of the document is to show the client how the vision and objectives of the client's business are understood by the service provider.

#### **CONSOLIDATED AND VALIDATED INTAKE**

The purpose of this document is to capture versions and status of all input documents on a timely basis and consolidate this input in terms of relevancy and value. For example: The client indicates which input is relevant and indicates the value of these relevant inputs. Documents that (according to the client) are not relevant or have no or low value, may better be ignored than seriously or extensively analyzed.

#### **GAP ANALYSIS**

The purpose of this document is to list all gaps between the As-Is situation and the To-Be situation.

#### **GLOSSARY**

The purpose of the Glossary is to obtain a common vocabulary among all the stakeholders, primarily within the project. But care should be taken to tune the definitions with the Business departments.

#### **MIGRATION STRATEGY**

The purpose of this document is to define a Migration Strategy. The strategy gives direction what type of Migration Scenarios to develop and supports the selection of the scenario to be selected. The Migration Strategy is based on the Migration Risks and Migration Requirements & Constraints.

#### **OPPORTUNITY REPORT**

The purpose of the document Opportunity Report is to define all opportunities and options that meet the objectives in addition to the options already defined by the client. It is in this document that the possible additional features for the project can be documented and if the client wants some of these additional properties to be included within the project scope, then it is removed from this document and put in the "*Scope* document".

#### **OVERALL MIGRATION DESIGN**

The purpose of this document is to describe a specific Migration Scenario in detail to enable the

readers of this document to base a project plan on this document. It describes the Migration Scenario in general terms and also highlights findings encountered during the detailing of this Migration Scenario.

#### **PARTICIPANT ANALYSIS REPORT**

The purpose of this document is to identify the participants and their contribution to the project. It identifies a list of participants and their contribution in the different SEMBA phases.

#### **REFERENCE MODEL**

The purpose of this document is to describe a reference model chosen and the reasons why the reference model is chosen.

#### **REQUIREMENTS MANAGEMENT PLAN**

The purpose of the Requirements Management Plan is to specify the information to be collected and control mechanisms to be used for measuring, reporting, and controlling changes to the requirements. This is done by defining:

- the requirements management organization;
- the requirements management process;
- the role of requirements management in gathering, communication and detailing of requirements;
- the different types of requirements and how they are documented;
- traceability of requirements;
- Quality criteria of requirements and how they are reviewed.

Choosing the appropriate documentation and traceability for requirements will assist the business analyst to:

- assess the project impact of a change in a requirement;
- assess the impact of a failure of a test on requirements;
- manage the scope of the project;

- verify that all requirements are fulfilled by the implementation;
- Manage change.

All decisions regarding requirements documentation, traceability, guidelines and strategies for Requirements Management are documented in this Requirements Management Plan.

#### **SCOPE**

The purpose of the document is to define the boundaries of the engagement. The scope also clarifies on which deliverables are needed to be realized by other entities in order to achieve the objectives. The scope definition is leading in times of discussion that may occur during the assignment. These discussions may concern whether some tasks and/or deliverables should or should not be executed, respectively delivered.

#### **SCOPED REFERENCE MODEL**

The purpose of this document is to elaborate what parts of the indicated Reference Models in the “reference model” document are used in the To-Be Design.

#### **TO-BE DESIGN**

The purpose of this document is to define the envisaged To-Be situation based on the analysis performed during the preparation of the As-Is Understanding document, the gathered business requirements, and the Change Drivers. The Business Context, Business Processes, Information and Applications of the To-Be situation are described coherently. Focus must be on the changes in comparison to the as-is situation.

### ***2.3. Integrated Requirements Management Approach (IRMA)***

The Integrated Requirements Management Approach (IRMA) is an initiative of a group of consultants of Capgemini working in the field of requirements. IRMA supports system analyst in gathering and elicitation of system requirement from the business requirements. For this purpose the IRMA provides a set of templates and guidelines.

The business analysis is done in order to identify the need of business and to determine the solution for business problem. If the solution of the business problem is development of software system then business analysis is followed by system analysis. SEMBA is a method which is used by business analyst for business analysis and IRMA is approach used by system analyst for system analysis.

From last 50 years software industries are providing solutions to many business problems. In practice it is not always the case that the business analysis is done prior to system analysis. When there is change in business process or a new business process is introduced then business analysis is essential but there are cases when software project starts from system analysis phase. For example, upgrade in existing software system because of technology change or amalgamation of existing software systems.

IRMA is developed by reusing and integrating standards and methods such as Volere (13), international standard for the evaluation of software quality (ISO 9126) (14) , Rational unified process (RUP) (15) and DeliverSAP<sup>1</sup>. It combines different theories and best practices used by consultants who have already worked on various assignments in the field of requirements gathering. IRMA offers a set of ready to use deliverables.

IRMA focuses on gathering and elicitation of IT requirements. It is used by the system analysts who liaisons between the client's business and implementation professionals. Thus, IRMA takes business requirements as input and transforms these to system requirements. IRMA connects with SEMBA and software engineering approaches like RUP and DeliverSAP.

IRMA focuses both on the requirements management techniques and on the quality of the requirements. It provides a detailed quality procedure document containing checkpoints to ensure that all the necessary steps have been taken. The quality assurance procedure of IRMA is supported by the **Requirement Management Assessment (RMA)** tool. The RMA tool is used to create an overview of the status of the project in regard with the quality aspects of

---

<sup>1</sup> DeliverSAP is an approach developed by Capgemini for the requirement gathering of package implementation, specially focusing on SAP projects.



requirements management. The tool can be used to reflect the status of the project at any stage of the project development life cycle. Thus, RMA can be used to monitor and to improve the quality of requirements management.

**Elementary Requirements Structure:** Capgemini develops information systems. Hence, Capgemini defines requirement as a specific property of an information system. Based on the system properties or system aspects Capgemini classify the requirements. Figure 1 displays the classification of information system properties.

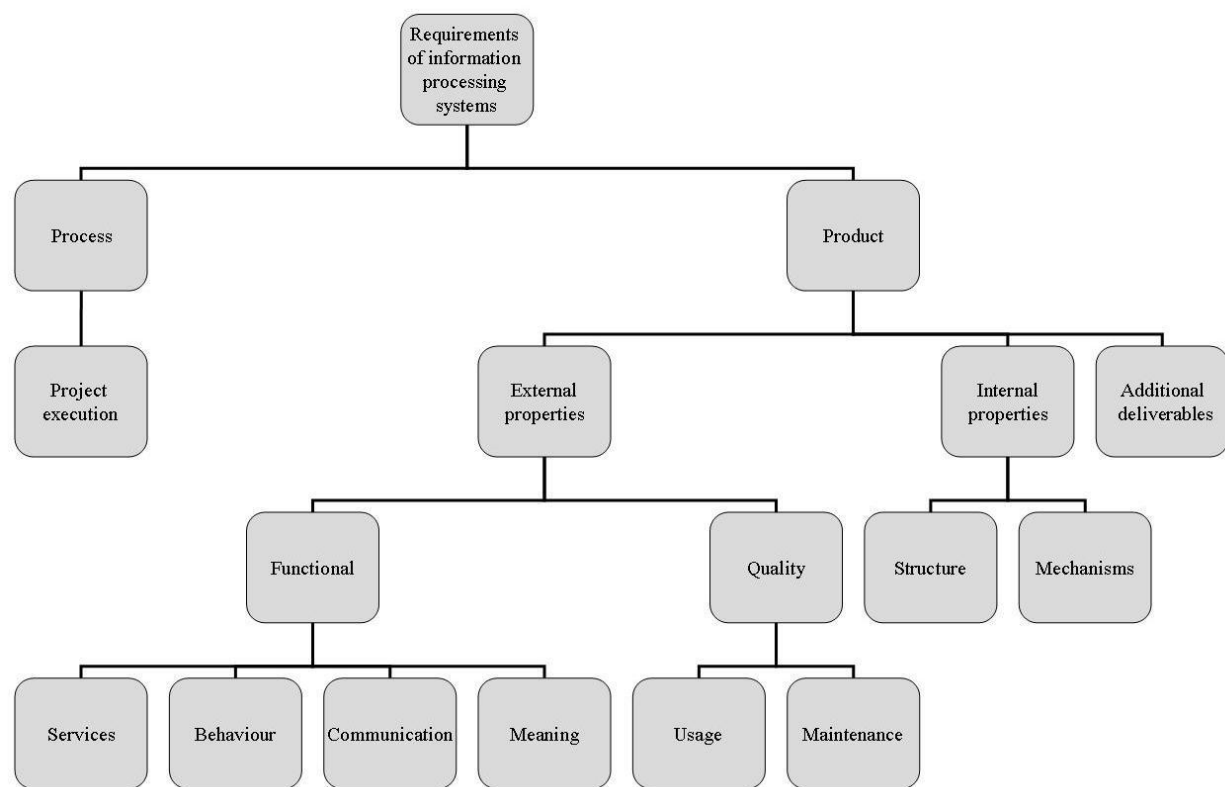


Figure 1: Classification of information system properties

The nodes of the tree represent aspects of an information system. The tree starts with a distinction between process and the product resulting from this process. Here the product is a software system. The top-level distinction of product properties is between external and

internal properties. The requirements of a system can be defined more specifically by distinguishing a system into internal properties and external properties.

Internal properties are defined as the parts or elements from which a system is composed of, also referred as the construction of a system. The external properties on the other hand describe a system from an outside point of view. This describes the functionality of a system, the behavior of a system within its environment. The leaf nodes of the product sub-tree are the elementary requirement types identified by Capgemini for information system.

The rational unified process (RUP) standard has been followed by Capgemini in from-scratch development information system. In the case of package implementation of information system, DeliverSAP is the standard. Hence, IRMA has been developed for both RUP and DeliverSAP standards.

- I. ***IRMA for RUP***: A series of templates and guidelines dealing with specific elementary requirement types. There are 13 deliverables for this approach and these deliverables are called artifacts in IRMA. IRMA divides these artifacts in following to two sets of artifacts:
  - **Must artifacts**: The must artifacts are those artifacts which are mandatory in every software development projects.
  - **May artifacts**: The may artifacts are those which are not obligatory. These artifacts are prepared in order to enhance the understandability of projects or to support requirement management or to understand the services of software system.
1. **Vision**: It captures the essence of the solution in the form of IT requirements and design constraints that give the reader an overview of the system to be developed from a business requirements perspective. This artifact is a must artifact.
2. **Supplementary specifications**: This artifact contains the supplementary requirements categorized under specific subjects. IRMA template for supplementary specification identifies the three specific subjects: usage, maintenance and design constraints. Under each of these subjects IRMA dictates various supplementary requirements. For example, the usage subject of supplementary specification demands five

supplementary requirements: user group, usability, reliability, usage efficiency and document usage. This artifact is a must artifact.

3. Requirements management plan: This document describes gathering, identification and management of requirements and the process to be followed in case of changes in requirements. This artifact is a may artifact.
4. Software development plan: It contains details about the software development process i.e. details of organizational environment of the undertaken software project, purpose of project, project statement, phase plan for the project and other details related to development process. It also describes software project releases. It does not contain any business or IT requirement related to the system to be developed. This artifact is a must artifact.
5. Domain model: This describes entities, their attributes and responsibilities, necessary to enable the analysis and description of the functionality of the system. This artifact is a must artifact.
6. Glossary: It is used to define terminology specific to the problem domain or automated system, explaining terms which may be unfamiliar to the reader of the documents. This artifact is a must artifact.
7. Use case model: It is a structured set of use cases and actors described using one or more use case diagrams. Use case model represent the complete system in terms of actors and use cases. This artifact is a must artifact.
8. Use case specification: The relationship between the use cases is graphically shown in the use case model. The specification of a single use case is described in a use case specification. For every use case of use case model there exist a use case specification artifact. This artifact is a must artifact.

9. System rules: A system rule is a statement of policy or conditions or constraints that must be satisfied and which must be enforced by the system. Example of such system rules are as follows:

- Statement of policy: An orphans' pension will only be granted to children younger than 18 years.
- Conditional rules: A bank account can only be activated after the solvency is verified.
- Constraint on data: A postal code has four numbers and two alphabetic characters.

All the system rules are enlisted in System rules artifact. The System rule artifact is a must artifact.

10. Service definition: The service definition describes the services<sup>2</sup> offered by the system to be developed to its external systems. The external system could be any other software system or the business unit of the client. This artifact is a may artifact.

11. Interface mapping: Interface mapping describes the use of services in supplier and consumer perspective. For each service enlisted in service definition the supplier software system and consumer software systems are indentified. This artifact is a may artifact.

12. Storyboard: Storyboard describes the interactions within a use case. This is not to prototype or test the look and feel of the user interface of system to be build. This artifact is a may artifact.

13. Navigation map: This document describes the structure of the user-interface elements in the system, along with their potential navigation pathways. This artifact is a may artifact.

---

<sup>2</sup> A service of a software system can be defined as the expected output of software for known input. For example: premium calculation or generate proposal for car loan.

II. **IRMA for DeliverSAP:** A series of templates and guidelines supporting the process of capturing requirements in package implementation with SAP. There are 9 deliverables for this approach and these deliverables are called artifacts in IRMA. Many artifacts of IRMA for DeliverSAP have been directly inherited from the DeliverSAP standards. All the artifacts are mandatory.

1. Business blueprint - business process design: This document is similar to the vision artifact for IRMA-RUP. It captures the solution in the form of high-level requirements and design constraints.
2. Business blueprint - data object design: This document describes all the data information objects of the system. The system or functional requirements are identified in this artifact but elaborated in functional specification artifact.
3. Business blueprint - organizational structure: This document reflects the organizational entities and scope. Examples of organizational entities are: company code or sales organization or plant or purchase organization or purchasing group.
4. Data management migration: Data management migration describes the strategies used for the data migration from the legacy system to the new system.
5. Functional specification: This artifact is designed to identify requirements that cannot be accomplished by standard content or platform features.
6. End-user learning strategy: End-user learning strategy describes the learning strategies used. The learning strategies are the methodologies used to prepare and support the affected workforce to understand, learn and embrace the solution.
7. Future system map: Future system map describes the requirements that the to-be architected SAP technical infrastructure has to meet. Sometimes this document is referred to as the technical system design.
8. Roles and authorization specification: Role and authorization specification describes roles along with their authorization restrictions, and relevant business and legal requirements.

9. Testing protocol and strategy: This document describes risk-based test strategy, test objective testing methodology, and test automation to be applied while testing deliverables, as well as the test management reporting.

#### **2.4. Related work**

1. One way to guarantee software quality improvement is through imposing certification. **SCfM\_Prod** model (16) has been developed based on end-product quality approach of certification. SCfM\_*prod* model consists of six main components: pragmatic quality factor (PQF) as the quality certification, guidelines and standard, product criteria, certification specification, certification representation method, repository and certification team (16). This model is similar to LSPCM since this too assesses the quality of the product not the process. Overall their obligations were four fold which also included a model for process evaluation:
  - a) To identify software certification requirements from the environment and industry
  - b) To construct a software certification model based on process development quality approach
  - c) To construct a software certification model based on product quality approach
  - d) To validate and evaluate the models through case studies involving organizations and industries in Malaysia.

For their first task in building a software product certification model they tried to identify the requirements through empirical studies. They took two background studies via surveys which were conducted to investigate and identify the requirements of best quality software and certification in the industry. To construct a software certification model based on product quality approach they identified the certification levels and characterized them in four distinct levels: excellent, good, basic and acceptable, and poor. The certification level of product is determined by comparing the score value obtained in the certification exercise. In LSPCM there are definite levels of certification identified as a result of the measure of completeness, uniformity and conformance. SCfM\_*prod* model is a

general product certification model which specifically does not identify requirements documents for offshored projects separately. Whereas, with our research we attempt to make LSPCM capable enough to assess the quality of offshored requirements documents. This makes LSPCM more usable compared to SCfM\_prod model when it comes to offshored project requirements document.

Furthermore this model differs from LSPCM in the following ways:

- a) The specific checks in LSPCM measure quality in terms of completeness, uniformity and conformance which were selected as a result of continuous research in this field from theory and practice. Whereas in SCfM\_prod model the quality attributes defined in ISO 9126 model form the baseline of the assessment metrics.
  - b) Unlike LSPCM this model defines two sets of attributes, which are by means of the behavioral and the impact attributes. The behavioral attributes consist of high level software quality characteristics which include usability, efficiency, functionality, maintainability, portability, integrity and reliability. Integrity is not included in ISO 9126 model but is included in this model because of the requirement from literature and empirical studies they have performed. The impact attributes indicates the conformance in user requirements, expectations and perception.
2. The Product Assurance and Safety department at the European Space Research and Technology Centre (ESTEC/TOS-Q) is responsible for product Assurance and Safety of ESA space systems. In this function, ESTEC/TOS-Q is involved in the specification, development and verification/validation for all ESA space systems. The proportion of sub-systems containing software is increasing. More and more critical functions are implemented using software. As a consequence, developers and users of space software systems want to reassure themselves about the quality of the software involved. Yet, specification and evaluation of software quality is rather immature. Definition, specification and selection of software quality requirements for a specific project and their implementation are not simple, straightforward activities. In addition, one prerequisite for the verification/evaluation of these quality requirements is the quantitative specification (e.g.

metrication) tailored to its criticality and type. TOS-QQS initiated the study **Software Product Evaluation and Certification (SPEC)** to investigate how software quality can effectively be specified, evaluated and eventually certified for space projects (17). The major difference between SPEC and LSPCM is in the ability of these models to be a generic model for product certification. LSPCM is suitable for a wide range of software's like governmental software's which has more of business requirements, space research software's which has more to do with core technical requirements and dependencies etc. Whereas SPEC can handle certification of products from scientific projects alone. We say so because, for example in a usual information systems related project the key business issues becomes very relevant in terms of the reasons for initiating such a project. This is not the case if it is a scientific project which has no business issues to be concerned about. Hence such a scientific product assessment model will not be sufficient to address the assessment of business requirements in an information systems related project.

3. A software product certification model LSPCM (LaQuSo Software Product Certification Model) (1). LSPCM provides set of rule for the assessment of major deliverables of software life cycle (see Section 2.1). Since the complete documentations for LSPCM model were available, we have chosen this as our baseline model and proceeded with further research (see Section 2.1).

Since we study the quality assessment of business and IT requirements, it is also essential to have background information on the methods of requirements gathering and maintenance. These methods are handled by requirements management. Below given are a few project management and requirements management approaches and tools available till date.

4. In (10) Warren Reid has described a software project management model called Cooperative Project Recovery (CPR) model. The CPR model is a four-tier, 13 step model which focuses on **the actions** that must be taken to turn around, overcome, and compensate for deficiencies in project management, methodology and technology. The model emphasizes on the process to be followed during the system development. Hence it



includes the requirements gathering process. The business and IT requirements come under the scope of this thesis, therefore the steps of CPR model which are related to the process of requirement gathering are described below:

- Step 2 of CPR model: This step is intended to uncover, discover, dissect, and assess the root causes of the problems and to identify the stakeholders. The approach for the system to be build is defined in this step.
- Step 3 of CPR model: This step is intended to estimate cost and budget, to define the scope of functionality to be build, to enlist the stakeholder expectation and acceptable risk, and to define the quality of the system to be build.
- Step 4 of CPR model: This step intended to identify the feasibility of the system to be build.
- Step 5 of CPR model: This step intended to define the development plan and the kind of deliverable.
- Step 6 of CPR model: This step intended to identify the priorities of the requirements.

SEMBA and IRMA (see Sections 2.2 and 2.3) are the methods to support the process of business and system requirement gathering and the above described steps of CPR model focuses on the actions that must be taken in requirement gathering. Unlike SEMBA and IRMA, the CPR model gives very short description of the actions that must be taken. SEMBA and IRMA have set of artifacts and these artifacts have predefine sections and subsection. Hence SEMBA and IRMA can used as check list for the assessment of business and IT requirements respectively. Whereas CPR model cannot be used for this purpose because of very short descriptions of the action to be taken.

5. Significant body of research is available in the field of **requirements management**. The purpose of requirements management is to assure that the organization meets the expectations of its customers and internal or external stakeholders. It includes objectives and constraints of the company. Tools for managing systems requirements help keeping

specifications consistent, up-to-date and accessible. Hence there are a large number of requirements management tools available in the market. Lists of various tools are found in the INCOSE Requirements Management Tools Survey (18). In order to analyze the Requirement Management Assessment (RMA) tool of IRMA (see Section 2.3) it was necessary to understand other requirement management tools available in the market. We used the **Automated Requirement Management (ARM)** (19 p. 10) tool for understanding the working of automated tool used for requirements management.

6. SEMBA and IRMA (see Sections 2.2 and 2.3) are the standard methods at Capgemini for carrying out business analysis and system analysis respectively. We have focused mainly on these two methods for our research purpose in order to analyze the offshoring trends in the industry and their effects. With these findings we intend to formulate suggestions on how the quality assessment approaches like LSPCM can be improved.

### ***2.5. Introduction to RUP***

The Rational Unified Process (**RUP**) is an iterative software development process framework created by the Rational Software Corporation. One of the offshored industrial projects provided by Capgemini has followed the RUP standards for the development process, it was necessary for us to have knowledge about RUP.

RUP divides one development cycle in four consecutive phases. (15)

- Inception phase
- Elaboration phase
- Construction phase
- Transition phase

Each phase is concluded with a well-defined *milestone*, a point in time at which certain critical decisions must be made and therefore key goals must have been achieved.

- Inception phase (15 p. 4): In this phase the business case which includes business context, success factors (expected revenue, market recognition, etc), and financial forecast is established. To complement the business case, a basic use case model, project plan, initial risk assessment and project description (the core project requirements, constraints and key features are generated).
- Elaboration phase (15 pp. 4-5): The purpose of the elaboration phase is to analyze the problem domain, establish a sound architectural foundation, develop the project plan, and eliminate the highest risk elements of the project.
- Construction phase (15 p. 5): The primary objective of construction phase is to build the software system. In this phase, the main focus goes to the development of components and other features of the system being designed.
- Transition phase (15 p. 6): The primary objective is to 'transition' the system from the development into production, making it available to and understood by the end user.

### 3. Case studies

To answer to our research question (as stated in Section 1.4) we apply LSPCM, SEMBA and IRMA to a number of industrial cases of offshored development. Based on the application results we formulate suggestions on how these quality assessment approaches can be improved.

After studying the quality assessment methods a number of discussions were carried out with the experts at the industry as well as at the university. From these discussions we formulated the following way of assessing and analyzing the requirements document of this case study.

1. Using LSPCM: Since LSPCM is designed as a set of rules which is in the form of specific checks, LSPCM can be directly applied on the requirements document. This is done by checking whether the specific checks specified in LSPCM are satisfied by the document under assessment. Therefore, the result of this assessment is a defect report indicating the checks violated and the reason for the violation. For example: “**SC3.1 (e)** Every requirement is uniquely identified. This was applicable but not satisfied since redundant and useless business/system rules were found”.
2. Using SEMBA and IRMA: SEMBA and IRMA are industrial methods for business analysis and requirements management respectively. They describe the detailed content of a requirements document that should be prepared after the business and system analysis. This description of the expected content can, thus, be used as a checklist assisting in detecting omissions from a requirements document under assessment. Finally the result for this kind of assessment is a document indicating the omissions relevant for that particular project.
3. Analysis: Results of application of LSPCM, on one hand, and of SEMBA and IRMA, on the other hand, will be the input for analyzing the findings. Here we compare the results in the following way:

- i. Check whether some defects will have been removed if some missing content would have been present.
  - ii. Check whether, even if all the missing contents would have been present, still some of the defects would not have been solved. Collect all such defects.
4. The knowledge gained from the analysis will serve as a feedback to LSPCM and SEMBA & IRMA developers.

### ***3.1. Case study 1: Application software for an insurance company***

Our first case study is an offshored RUP-based industrial project carried out by Capgemini. In this project the inception phase of RUP was completely carried out onsite and the elaboration was carried out partly at onsite and partly at offshore site. The construction phase was completely carried out at the offshored site. During the case study we carefully analyze the requirements documents prepared for this project and forming the major reference point for the offshored end. The assessment is detailed in section 3.1.2 and the suggestions become the conclusion of this case study.

#### **3.1.1. Case description**

This project was started in order to develop a new system to sustain the new work processes to be implemented by integrating two existing systems. This new system will completely replace the existing systems. Neither of the existing systems includes functionality to sustain the new work processes of the new required integrated system.

#### **3.1.2. Result of assessment**

##### ***Results of the LSPCM assessment:***

LSPCM is applied on the requirement document of case study and a defect list is obtained from this assessment. We categorized the listed defects in three categories: minor inconsistencies, major inconsistencies and omissions.

1. **Minor inconsistencies** – By minor inconsistencies we understand the inconsistencies which do not affect the system functionality. These inconsistencies should be removed from the requirement documents in order to improve the readability and understandability of the requirement documents. As minor inconsistencies we consider the spelling mistakes, structural errors<sup>3</sup>, ambiguous notations, errors caused by translation (Dutch to English), and the difference in the names and terms in the documents prepared by the client and project team. These differences in the document prepared by project team and the client would have been addressed if the changes in the requirements would have been documented.

The following has been observed:

- a. In the sample of 20 pages from the total of 146 pages in the given requirement documents on average one spelling error is present per page.
- b. In the sample of 41 pages from the total of 146 pages in the given requirement documents on average three structural errors are present per page.
- c. Redundant business/system rules. This doesn't affect the functionality of the system to be built but the redundancy in business/system rule cause difficulty in the management of these business/system rules. There were 3 redundant business/system rule found in total of 92 business/system rule.
- d. Useless business/system rule. The scope of the project was changed after the sign-off between the client and the project team. Hence, a number of functionalities were omitted from the use case description but the business/system rule referring to these omitted functionality were present in the list of business/system rule. There were 11 useless business/system rule found in total of 92 business/system rule.
- e. The abbreviations and their full forms were not always consistent. For example: In the activity diagram abbreviation FB (this could be abbreviation for *functioneel beheerder*) is used but in the textual description in the same context the

---

<sup>3</sup> Structural errors are the grammatical errors, business/system rule-IDs omissions or omission of non-functional requirement IDs.

abbreviation ME (Maintenance executive) is used. Correspondence of the Dutch and the English terms is not obvious for offshore development team.

- f. The names of use cases in the document prepared by the client and in the document prepared by project team are not the same. For example: The name of use case for the use case ID “UC003” is **reschedule appointment** in the client document but in the document prepared by the project team the use case ID “UC003” is **rearrange appointment**.

2. **Major inconsistencies** – In this category we summarize the inconsistencies related to the expected functionality of the system to be developed. Failing to correct them could lead to wrong interpretation of the document.

- a. The business/system rules are not sufficiently elaborated. For example: The description of one of the business/system rule is “4 digits”. It is not obvious for offshore development team that this rule is referring to 4 digits of zip code.
- b. Same rule ID is used for different descriptions of business/system rule.
- c. The description of business/system rule is not consistent. For example: One of the business/system rule is described as “The user interface is build out of three (or more) screens” but in the later section the same rule is described as “The application is composed of three interacting panes”.
- d. The business/system rules contradict each other. For example: One of the business/system rule demands that in case of *Anders* all four fields (location, city, address and zip code) are mandatory but another business/system rule demands that in this case only location and city are mandatory. Zip code and address are optional.
- e. The roles and the numbers of actors in the use case diagrams are not same as the roles and numbers of actors in use case specifications.

3. **Omissions** – In this category we summarize the omitted parts of the functional requirements, non-functional requirements and glossary.

- a. There are seven cases where the client documents demand for a specific functionality but such a functionality was missing from the document prepared by the project team.
- b. The roles of actors are insufficiently elaborated. For example: The use case diagram demands that “advisor” is the actor for the use case “create appointment”. The business/system rule referring to the use case “create appointment” demands that “call center or maintenance employee or advisor” are the actor. Whereas the documents don’t describe that all call center and maintenance employee are advisors.
- c. The design constraints section of non-functional requirements demands the data model and analysis model but these models are missing.
- d. The maintainability section of non-functional requirements demands the database design document but this document is missing.
- e. The values of parameters such as mean time to repair/replace and mean time to failure for reliability non-functional requirements are missing.
- f. The design constraints section of non-functional requirement demands availability of the interface documents but this document was missing.
- g. Many business-related terms are not described in the glossary. For example: manager, sales manager, sales executive, branch manager, and sales support. These terms are used as names of actors. Since these terms are not defined the roles and the authorities of these actors may be unclear to the offshore developers.
- h. Similarly, many project-related terms are not described in the glossary. For example: PoC, and client CDS directory.
- i. The acronyms were not expanded.

***Results of the SEMBA assessment:***

During assessment using SEMBA, we use the guidelines for the SEMBA deliverables as a checklist in order to map the content of each of these deliverables to the content of the requirements document of the offshored project under study. From the discussions with the



project manager of the project under study we derived certain concerns over the business issues which the development team had trouble resolving during the project execution. These concerns were documented and after the assessment we investigated whether these concerns would have been addressed if the missing content of the SEMBA deliverables were actually present. The table below indicates to what extent the contents of each SEMBA deliverable was present in the requirements documents of the project under study. Since the priority and measure of relevance for each of the deliverables are decided during the preparation of the Approach document, the priority and degree of relevance depend highly upon the stakeholders' interest. The elaboration on the *Approach* document in section 2.3 of chapter 2 indicates these facts. Furthermore, the project layout is decided in the *Approach* document. For this case study all the SEMBA deliverables were assumed to be applicable. Since the specific roadmaps for SEMBA-RUP was not available, this decision was convenient to work with. The decision on whether the deliverable is **optional** or **mandatory** is decided by the kind of software development method. There can be only three possible values for status information in the case study document. Either the content of the deliverable is completely found (all the sections are present) or it is partially found (few sections are present) or it is not found at all.

Since in this case study a RUP way of software development is followed, it is useful to mention here that there were certain deliverables of SEMBA which is a mandatory document and is equivalent to mandatory RUP deliverable. Mostly SEMBA is followed by IRMA i.e. business analysis followed by system analysis. IRMA also has deliverables as in SEMBA but is known as artifacts. The IRMA approach dives more to the system requirements which must conform with the business requirements gathered at the business analysis phase. Since SEMBA is followed by IRMA the SEMBA deliverables must be useful deliverables for creating the IRMA artifacts. This would thus contribute towards avoiding clones of deliverables being prepared while gathering requirements. Hence the following deliverables of SEMBA are used for creating the corresponding artifacts in IRMA as shown in the table and thereby makes it mandatory for a RUP based project. More on IRMA was discussed in the section 2.3.

SEMBA deliverable	Mapping on IRMA artifact	Explanation
<b>Change drivers</b>	Vision	Input to Chapter ' <i>positioning</i> ' of the IRMA ' <i>Vision</i> ' document.
<b>Opportunity report</b>	Vision	Input to sub section ' <i>stakeholders and end-user needs</i> ' of the IRMA ' <i>Vision</i> ' document.
<b>As-is understanding</b>	Glossary	SEMBA will be the base for ' <i>Business definitions and abbreviations</i> ' of the IRMA ' <i>Glossary</i> ' document.
	Use case model	A set of transaction diagrams that describe how a business transaction is processed through a number of business processes is described in the ' <i>Application landscape as-is</i> ' of SEMBA ' <i>As-is understanding</i> ' document. This is used in the definition of actors and use cases in the IRMA ' <i>Use case model</i> ' document.
	Domain model	The information area map in the ' <i>Information as-is</i> ' of the SEMBA ' <i>As-is understanding</i> ' document is used in the overview diagram, grouping of entities and information objects in the IRMA ' <i>Domain model</i> ' document.
<b>Business initiative</b>	Vision	The business context in the SEMBA ' <i>Business initiative</i> ' document is used in the business context of the IRMA ' <i>Vision</i> ' document.
<b>Business requirements</b>	Vision	The business requirements catalogue in the SEMBA ' <i>Business requirements</i> ' document becomes the summary of capabilities and product features in the IRMA ' <i>Vision</i> ' document.
	Supplementary specification	The migration requirements in the SEMBA ' <i>Business requirements</i> ' is used to describe the generic functionality, usage, maintenance and design

		constraints in the IRMA ' <i>Supplementary specification</i> ' document. The business rules are regarded as the rules that can be transformed into system rules in the IRMA document.
<b>Client objectives and expectations</b>	Vision	The client business vision and objectives in the SEMBA document is used for stakeholder and end-user needs in the IRMA ' <i>Vision</i> ' document.
<b>Participant analysis report</b>	Vision	The list of participants and the participant analysis report of the SEMBA deliverable is used in the stakeholder summary and end-user summary of the IRMA ' <i>Vision</i> ' document.
<b>Requirements management plan</b>	Vision	The stakeholders in scope of the SEMBA document becomes the stakeholder summary in the IRMA ' <i>Vision</i> ' document.
	Requirements management plan	The requirements management plan in the SEMBA deliverable describes the level of governance, operational instructions for the requirements elicitation and documentation and change management of requirements. This is used in the requirements management policy, requirements review process, and requirements change management and requirements types in the IRMA ' <i>Requirements management plan</i> ' document.
<b>Scope</b>	Vision	The business scope, streams and task, deliverables and level of details are used in the business context of the IRMA ' <i>Vision</i> ' document.
<b>To-be design</b>	Supplementary specification	The to-be application model in SEMBA is used in the IRMA ' <i>Supplementary specification</i> ' document.
	Service definition	The to-be interface model of SEMBA is used in the

	identification of service definition in scope of the IRMA 'Service definition' document.
Vision	The product and service definition in the to-be business context model of the SEMBA 'To-be design' document is used in the business context of the IRMA 'Vision' document.
Use case model	The to-be implementation process model in the SEMBA document is used in the process models in the IRMA 'Use case model' document.

The details of the assessment table below can be found in the assessment report (20 p. 40).

<u>SEMBA deliverables</u>	<u>For case study 1</u>	
<b>Approach</b>	Applicable and optional	Partially found
<b>As-Is Understanding</b>	Applicable and mandatory	Not found
<b>Business Initiative</b>	Applicable and mandatory	Not found
<b>Business Requirements</b>	Applicable and mandatory	Partially found
<b>Change Drivers</b>	Applicable and mandatory	Not found
<b>Client Objectives and Expectations</b>	Applicable and mandatory	Found
<b>Consolidated And Validated Intake</b>	Applicable and optional	Found
<b>Gap Analysis</b>	Applicable and optional	Not found
<b>Glossary</b>	Applicable and optional	Found
<b>Migration Strategy</b>	Applicable and optional	Not found
<b>Opportunity Report</b>	Applicable and mandatory	Not found
<b>Overall Migration Design</b>	Applicable and optional	Partially found
<b>Participant analysis report</b>	Applicable and mandatory	Partially found
<b>Reference Model</b>	Applicable and optional	Not found
<b>Requirements Management Plan</b>	Applicable and mandatory	Partially found

<b>Scope</b>	Applicable and mandatory	Not found
<b>Scoped Reference Model</b>	Applicable and optional	Not found
<b>To-be design</b>	Applicable and mandatory	Partially found

From the assessment results we obtained the content from SEMBA which were not present in the considered documents. During the assessment, we had an interview with the project manager for this case study from which we inferred a few concerns related to the business requirements which were critical for the success of the project under study. Then we checked whether these concerns would have been solved if the missing SEMBA deliverables were present. Below shown is a table which identifies the concern and the SEMBA deliverable which would have addressed the concern.

<b>CONCERNS</b>	<b>How is it relevant?</b>	<b>The documents which would have addressed the concern</b>
<p><b>Concern-1: Insufficient information on the client business and the information usage in this new project initiative. The following questions were unanswered:</b></p> <ul style="list-style-type: none"> <li>• Should the database systems be upgraded to newer technology so as to be compatible with the new system being build?</li> <li>• Should the intermediate deliverables at various milestones of the project be tested on the client</li> </ul>	<p>When the product was implemented at the client site the software had a long response time which was due to the delay in database activities. This was because the information usage details were not elaborated in the documentations.</p>	<p><b><u>SEMBA deliverable</u></b> – As-is understanding, Business initiative documents</p>

databases?		
<p><b>Concern-2: The internal and external application systems that enable information provision were missing. The application systems are other functional software units which is the part of the existing system or will be a part of the new system. For example in a “Student_Loan_Sanction_System” in a bank the loan cashing system inside the bank is an internal application system whereas the “Student_Loan_Sanction_System” might use data from the student registration system of the university or even the citizenship registration system of the municipality. These are external application systems to which the “Student_Loan_Sanction_System” has an interface.</b></p>	<p>In this case study there was a need to identify how certain application systems were interfaced to each other. For example the Microsoft outlook express was used to facilitate a functionality of the new system. Hence such a knowledge of the type of the application interface which says whether it is a real time or a file interface becomes essential.</p>	<p><b><u>SEMBA deliverable</u></b> – Section “Application landscape As-is” in the As-is understanding document</p>
<p><b>Concern-3: Exploration of opportunities in the added value of the new system was missing. This was a concern since there were ideological disagreement between the client and the</b></p>	<p>When the requirements are gathered it might be the case that every time new requirements gets added due to change of opinion within the client.</p>	<p><b><u>SEMBA deliverable</u></b> – Opportunity report document</p>

<p>service provider which later lead to removal of several features from the proposed system. If the opportunity for every feature in the system was explored the service providers would have had evidence to convince the clients about its addition or removal.</p>	<p>To avoid such frequent changes it is better to have an opportunity report documented which highlights other options that meet the client objective. It is up to the client to put these new options within the scope of the project.</p>	
<p><b>Concern-4: Ensuring that the client objectives are met and their expectations are fulfilled. This is also checked by verifying whether all the requirements are fulfilled by the implementation and by consistent way of management of the agreed upon requirements by both the client and Capgemini</b></p>	<p>In this case study we found that the requirements were prone to different implementation strategies at the design and development phase. This is undesirable when the product delivery is scheduled for a fixed date. This can be solved by following an agreed upon requirements management process which conforms to client's objectives and expectations.</p>	<p><b><u>SEMBA deliverable</u></b> – Requirements management plan document and client's objectives and expectations document.</p>

### ***Result from the IRMA assessment:***

The given case study falls under the RUP scope of IRMA. Hence, the artifacts of IRMA for RUP are used for assessment. Each artifact of IRMA has a template and IRMA proposes guidelines to fill in these templates. IRMA is used as checklist to evaluate the requirement documents of the Case study 1. Specifically, we looked for the content of each artifact in the requirement documents of Case study 1. For example: The template of vision artifact of IRMA demands the following sections to be present: introduction, description stakeholders and end-users, product overview and product feature. For each of these sections the guideline of vision artifact of IRMA dictates the kind of information to be filled in. We check whether this information is present in the given requirement documents, and we conclude that the artifact is complete if all the information required is found . If only part of the information is found, we state that the artifact is partially complete.

The following table shows the results of the evaluation. The first column of the table lists the artifacts of IRMA for RUP, second column denotes the whether the corresponding artifact must or may be present for a project in general as demanded by IRMA, the third column denotes whether the artifact must or may be present for the current case study, and the fourth column illustrates the status of corresponding artifact in current case study. (20 p. 58)

<b>Name of artifact</b>	<b>Presence</b>	<b>Applicable in case study-1</b>	<b>Found in case study-1</b>
<b>1. Vision</b>	Must	Must	Partially (20 pp. 62-64)
<b>2. Supplementary specification</b>	Must	Must	Partially (20 pp. 66-68)
<b>3. Requirement management plan</b>	May	Must	Partially (20 pp. 70-71)
<b>4. Software development plan</b>	Must	Must	Partially (20 pp. 73-76)
<b>5. Domain model</b>	Must	Must	Partially (20 p. 77)
<b>6. Glossary</b>	Must	Must	Partially (20 p. 78)
<b>7. Use case model</b>	Must	Must	Partially (20 p. 80)
<b>8. Use case specification</b>	Must	Must	Partially (20 pp. 82-83)
<b>9. System rule</b>	Must	Must	Partially (20 p. 85)
<b>10. Service definition</b>	May	Optional	Not found



<b>11. Interface mapping</b>	May	Optional	Not found
<b>12. Story board</b>	May	Not applicable.	Not found
<b>13. Navigation map</b>	May	Not applicable	Not found

**Table: IRMA Results**

### 3.1.3. Conclusion

The assessment of the requirements document of case study-1 using the LSPCM model, SEMBA (as check list) and IRMA (as check list). We have described the result of assessment in Section 3.1.2. Based on our experience with this case study we present the suggestion for the case study, LSPCM model, SEMBA and IRMA in Section 3.1.3.1, 3.1.3.2, 3.1.3.3 and 3.1.3.4 respectively.

#### 3.1.3.1. Case study -1: Application software for an insurance company

Following are the recommendation for the case study:

1. The major inconsistencies and the omissions must be solved before handing over the documents for development at the offshore end.
2. To improve the readability and the understandability of the documents among all the stakeholders and offshore development team it is essential to remove all the minor inconsistencies.
3. There should have been a change document maintained for each document of the requirements document. This would have addressed many inconsistencies.
4. The project “application software for insurance company” has been initiated because of merger of business processes. Hence in this case business analysis is essential and it is essential to understand the relation between the business and system requirements. The business requirements should have been elaborated, and the relation between business and system requirement should have been detailed. This would have addressed the conflicts and differences between the document prepared by client and the documents prepared by project team.

### 3.1.3.2. LSPCM

In this section we enlist the suggestions for the model LSPCM from business and IT perspective. Hence suggestion for LSPCM from SEMBA and IRMA.

#### ***For LSPCM from SEMBA:***

LSPCM recognizes the business context of a project in the context description product area. For offshored projects it becomes essential to look at the effects of offshoring for different business issues. Sometimes there might be a merge of organizations else it might be a change in governmental policy which became the business reason for the new project. The business reasons are the business issues. This was inferred from the assessment of the case study 1 requirements since it was specifically a case of organizational merge. The major concern while offshoring is the communication and understanding of the documents that travel across different geographical locations. The three specific criteria of LSPCM sweep superficially over the business aspect of any offshore project. Based on this knowledge we propose the following suggestions for LSPCM with respect to case study 1.

1. In this case study the business requirements were documented by the client. These were the initial set of requirements which will eventually evolve into a complete and consistent set at a later point in time. For this case study the use case documents were handed-over to the service provider without a documentation on the description of the business issue. We had to assess such documents using LSPCM taking into consideration the business requirements of an offshored project. Here LSPCM seems to have a coarse definition of rules under each of the specific criteria. Therefore in the *“formalness”* specific criteria of the product area *“context description”* the following sentences could be included. The reason for such a suggestion is also described along with it.

- a. **Suggestion** – The client’s initiative and analysis must be included in the business context. This must be present in the *“Boundaries of the target system”* where the scope is mentioned.

**Reason** – for this case study the analysis of the business context was essential.

From SEMBA we learned that the *“Scope”* document is used to define the

boundaries of the project. Hence we include this suggestion with the “*Boundaries of the target system*” of LSPCM. The “*Scope*” document falls under the SEMBA phase “*Focus & direction*”.

- b. **Suggestion** – Whenever applicable the information model of the system must be present in the business context. This must be included in the *Actors* definition in the *Context description*. The phrase “*other systems*” must be explained in a way to clearly indicate the scope of these other systems which includes for example database systems.

**Reason** – in this case study we found that when the final system was delivered and implemented at the client location the system had an undesired response time. Tracing the effect back to its cause we found that the database system within the scope of this project was least explored. Hence we came up with this suggestion.

2. From SEMBA we identified that it is an expert method and the usability of this method may lead to different results depending upon the expertise level of the business analyst. If we intend to make LSPCM capable of being used by any third party for giving a certification then we must be sure that all the relevant details for each *specific check* is included in the certification model. Comparing SEMBA and LSPCM we realized that the SEMBA expert method is detailed to very fine information and specifically covers a wide scope of business analysis. Whereas LSPCM is very coarse grained and is less detailed. Therefore while certifying any requirements document using LSPCM, if a valid certification has to be assured then there should be an elaboration on the expertise level of the certification personnel. If the certifying personnel does not have knowledge about the content of the different checks in LSPCM then a wrong assessment will be generated. This is due to the coarse grained description of LSPCM checks. Hence there can be an add-on document with the LSPCM document which indicates the different levels of certifying personnel’s (as compared to the levels of business analysts in SEMBA), with respect to the levels in the subject area expertise so as to cover the specific business context understandability of any product area. This was learned from SEMBA levels of business analysts and business analysis.

***For LSPCM from IRMA:***

1. The specific criteria “Compliance with Company Standards” under “Uniform” of LSPCM demands that no elements within a product area deviates from company standards.

**Problem:** In this case study we identified mix use of three languages English (US), English (UK) and Dutch. Since, the client and the organization of project team do not have a strict official language this inconsistency remained unaddressed in current version of LSPCM. This could affect the quality of end product in case of offshoring.

**Solution:** A check can be added to check the consistency with respect to the language used at offshore end.

2. The certification criteria “Completeness” of LSPCM require only the presence of non-functional requirements/quality requirements.

**Problem:** The requirement analyst may not be able to elicit all the quality requirements for specific domain. For example: In this particular case many parameter like, fault tolerance, data integrity and usage security, related to reliability and security non-functional requirement were missing. But these problems were not identified by the LSPCM. If IRMA would have been used then these problems would have been addressed.

**Solution:** LSPCM can dictate the list of quality attributes like in IRMA for specific domain<sup>4</sup>.

3. Prioritization of non-functional requirements/quality requirements.

**Problem:** The client may demand for high quality product but it is difficult to meet all quality constraints in given time and budget. So if the priority of quality demand is not decided at requirement analysis stage then the development process might exceed the time and budget limits.

---

<sup>4</sup> Domain could be real-time software or information system or scientific software or internet application software.

**Solution:** The quality assurance procedure of IRMA demands the prioritization of quality requirements. A check can be added in rule set of LSPCM for checking priority of quality requirements.

### 3.1.3.3.SEMBA and IRMA

#### *For SEMBA from the analysis*

While assessing the requirement documents of this case study by comparing it with the artifacts of SEMBA, it was observed that the following improvements in SEMBA could have been beneficial in deciding the quality of the business requirements:

1. There should have been a set of identified business domains from practice for each of which the importance of every SEMBA artifact could have been mentioned. Even though SEMBA has scope for being flexible enough for adapting itself for any business context, the classification of business domains would be able to help the lower level analysts to make the right decision. This would be even useful if the SEMBA expert method is used for business analysis by a third party (for example a client company) that will have very less knowledge on the activity structure in SEMBA. For example see Table 1.

<u>Business Activities</u>	<u>Priority</u>	
Merger	Approach	Optional
	As-Is Understanding	Must
	.....	.....
Change in Governmental policies	.....	.....
	Business Requirements	Must

2. With reference to section 4.1 in (20 p. 42) where the 10 major principles are mentioned from literature which would help in quantifying the quality, it would have been useful to do the following:
  - a. List out all the major quality aspects from practice and indicate the most essential ones for each business domain. For example:
    - i. **Business activity** – Merger
 

**Quality aspects** – reliability, maintainability, usability, adaptability

**SEMBA deliverables** which will assure the above quality attributes – Change drivers, gap analysis etc.

Hence this will help in creating a chart as shown in point 1, where the priorities are stated.
    - ii. **Business activity** – Change in Governmental policies
 

**Quality aspects** – reliability, maintainability.. etc
  - b. Include a table which will give an indication of the effort and cost involved in creating the required deliverables for each of these business domains. This must be scaled across different levels of quality management as indicated in the 6<sup>th</sup> principle of section 4.1 in (20 p. 42). This is indeed a difficult task to be realized but still it this can be one of the suggestions for future work research.

By the above 2 suggestions we can include the quality demands with the methodology followed in the industry and hence makes it easier to be assessed.

3. In the *user requirements* product area of the LSPCM, the first check under manual check for *Conformance* states that “No two requirements or use cases contradicts each other”. Such a check emphasizes on the uniqueness of the requirements in terms of the meaning it conveys. In SEMBA in the “*Business requirements*” deliverable, the requirements are documented in way to capture the uniqueness in terms of its numbering whereas it does not eliminate the possibility of capturing two requirements one which says property “P” and the other which says property “not P”. To avoid such ambiguities, every time when a new requirement is documented in the SEMBA deliverable, an instruction for checking such contradictions can be provided in the guideline for this deliverable. This is a serious concern

when offshoring the project development. Then the contradictory requirements statements can mislead the understandability of the developer who might be at times unaware of the context of the client business process to which such requirements might be conforming. This leads to the wrong behavior of the system.

***For IRMA:***

It is known from the interview [refer section “interview report” (20 p. 7)] with project manager of case study-1 that IRMA was not used for requirement gathering and elicitation. Using IRMA would have improved the quality of requirement documents to a great extent but there are few inconsistencies which would have been remained unaddressed even after using IRMA. Those inconsistencies are as follows:

1. Inconsistencies in variable names.

**Problem:** The variable names are inconsistent, i.e. variables names are in both English and Dutch. In the description of functionalities

**Impact:** This is not obvious for the offshore team. This gives room for developer to use two different variable one with Dutch name and other with English name.

**Suggested solution:** The language for the variable names can be dictated.

2. Inconsistencies in abbreviation used for diagrams and its description.

**Problem:** The activity diagrams have Dutch abbreviations and terms for activity and flow where as in the description of the same activity diagrams these abbreviation and terms are in English or vice-versa.

**Impact:** This is not obvious for the offshore team. This gives room for developer to make assumption about such terms.

**Suggested solution:** The language for the project terms, project abbreviation, business terms and business abbreviation can be dictated.

### 3. Contradictory business/system rules.

**Problem:** The description of business/system rule is contradicted by another business/system rule. For example: One of the business/system rule demands that in case of *Anders* all 4 field (location, city, address and zip code) are mandatory but an another business/system rules demands that in case of *Anders*, it is mandatory to enter location and city. Zip code and address is optional.

**Impact:** This could be lead wrong implementation.

**Suggested solution:** This could be resolved by adding a check on rule specification. One should check while adding a new rule, that whether the new rule is conforming to existing rules or not?

### 4. Conflict in the use case diagrams and its textual descriptions.

**Problem:** The name of actors for a use case in use case model and the name of actors for same use case in use case specification are different.

**Impact:** This gives room for developer to decide whom to allow the access for that use case. For example, Developer can decide to allow all the actors those are listed in use case model and use case specification.

**Suggested solution:** This can be avoided by including a conformance check between use case model and specification.

### 5. Redundant business/system rules.

**Problem:** The same business/system rules listed twice or more with different rule ID.

**Impact:** Redundancy of business/system rule. This increases the total number of business/system rules and this will lead difficulty in maintenance of these rules.

**Suggested solution:** IRMA suggests having unique rule ID for the Business/ System rules, but it does not say about the uniqueness of rule definition. This can be included.

### 6. Incompleteness in non-functional part



**Problem:** The data model, database design and maintainability of this model were missing from the set of documents.

**Impact:** Since maintainability of data model is not documented thus it gives room for developer to decide upon whether the changes in the database tables should be added or overwritten.

**Suggested solution:** The supplementary specification of IRMA has software maintenance and system maintenance section but these sections don't cover the maintenance of database. This can be included.

### ***3.2. Case study2: Unified ERP for an organization***

#### **3.2.1. Case description**

Our second case study is an offshore SAP-based on-going industrial project at Capgemini . The client has its units in all around the globe and these units are grouped in seven different clusters based on their geographical location. The two main objectives of Unified ERP project are following:

1. Improve and harmonize the client's business processes, and
2. Implement SAP globally and thereby replacing the existing ERP systems with ONE global SAP-kernel.

After the two week workshop conducted at the client's head office, the four mega business processes of the client were identified. In April 2009, the high level descriptions of four identified mega processes were accepted by the client for realization. When we took up this case study all the requirement documents were completed.

#### **3.2.2. Assessment**

##### ***Result from LSPCM assessment:***

The required elements of the context description product area of the LSPCM model were found in requirement documents of case study-2. Hence we carried out the assessment of for context

description. For the assessment of user requirement product area; the functional requirements, non-functional requirements and glossary are the required elements for the assessment. But, for this case study we did not received the non-functional requirements hence we were unable to proceed further for the assessment.

From the interview with software quality analyst (SQA) of case study-2 we understood that the non-functional requirements for the SAP projects are prepared at the later stage. Hence in order to carry out the assessment and to check the consistency of functional requirement we decided to apply the checks of LSPCM on the functional requirements and glossary. The detailed assessment for both context description and user requirement product area is presented in (21). Following issues were found in the requirement document of this case study:

1. The descriptions of symbols used in the flowchart were missing. Since the process flowcharts were not identified under any global standards. For example: UML diagrams or Petri-net process models.
2. Inconsistency in textual description and its pictorial form. For example: the textual description demands that the three input are required for the execution of a process but in the diagram only two inputs have been shown.
3. Inconsistency in the style of process flowcharts. For example: In some of the processes flowcharts the optional inputs are marked “optional” but in some of the process flowchart optional inputs are not shown in the diagram.
4. The conditional flows are not shown in the flow charts of the process diagram.
5. Few business-related terms are not described in the glossary. For example: production plan or production planning, demand plan.

***Result from SEMBA assessment:***

SEMBA method uses the concept of roadmaps to integrate with other methods available in practice. Roadmaps are plug-ins to SEMBA method. They enable efficient use of the method as they define a specific sequence of activities that should be completed in order to deliver the necessary products for achieving the goal that was set at the beginning of the project. The SEMBA-SAP roadmap is still under its way to be delivered. This case study was on a package

implementation project which has a different way of doing business analysis. The package used in this case study is SAP. SAP already supports a large class of business process implementations which has its own data handling support and communication logic. SAP enables users to customize these existing SAP implementations in order to meet their requirements.

In conventional business analysis there is always a decision to be made on how to solve the problem. But if we are already decided that SAP is the solution then this decision making is no more valid. Hence the approach and business initiation reasons need not be discussed in an SAP project. The gathering of business requirements form the major component of doing business analysis in an SAP project. For this case study we received the business blueprints for all the mega business processes involved, the requirements trace matrix document and the sign-off sheet for the project. The blueprints are documented as a result of gap analysis which is to bridge the gap between the as-is situation to the desired to-be situation. This case study involves analyzing the gap for 4 mega business processes. The blueprints indirectly indicate the way to customize the SAP process to meet the new requirements.

Since the SEMBA-SAP roadmap is yet to be delivered by the Capgemini we have interviewed the experts at Capgemini for figuring out the necessary deliverables for an SAP project. Whereas the IRMA for SAP roadmap is already available and thus it was effective to check which deliverable of SEMBA would be useful for those IRMA artifacts and hence scale down the deliverables of SEMBA only to these deliverables. This was indeed a good choice for proceeding with the assessment of SAP project because otherwise, we must either have to assume that all the SEMBA deliverable for RUP are equally important for SAP or none of the SEMBA-RUP deliverables are applicable. In either case there is no certainty over the correctness of the result of assessment since it is not proved by any relevant facts from literature or practice. Hence it is better to understand the artifacts of IRMA for SAP and sort the essential content from the existing SEMBA deliverables which can be reused for IRMA in case SEMBA is followed by IRMA. This is summarized in the table below:

SEMBA deliverable	Mapping on IRMA for sap artifacts	Explanation
As-is understanding	Business blueprint - business process design	The as-is SAP organizational process has to be identified in order to formulate a design that helps in a smooth migration to the expected to-be situation of the business processes. This is documented in the business process As-is of this SEMBA deliverable. Hence it can be an input for the IRMA artifact.
	Business blueprint - data object design	The SAP legacy system will have a predefined data handling technique which needs to be adapted to suit the desired data framework. The information As-is in the SEMBA deliverable documents the as-is data model and can be a valuable input for the IRMA artifact.
Business initiative	Functional specification	The relevant issues, exclusions and constraints are documented in the business initiative document of SEMBA. This can be used to identify the requirements which cannot be accomplished by standard content or platform features. The details on such out-of-scope requirements are documented in the indicated IRMA artifact.
Business requirements	Business blueprint - business process design	The SEMBA deliverable captures all business requirements regarding the Business Context, Business Processes, Information and Application Landscape which can be used for documenting the business process design blueprint document of IRMA.
	Business blueprint - data object design	The business requirements related to the "Information"(databases) content in the project is a valuable input for documenting the data object design in

		the IRMA blueprints.
Client objectives and expectations	Functional specification	The client expectations in terms of the major functional requirements conforming to the business needs documented in the SEMBA deliverable are used for identifying the functional specifications in the IRMA artifact. These set of functional requirements help in knowing whether the clients requirements and needs are correctly understood by the service providers so that the correct functional specifications with a clear indication of out of scope requirements are documented in the IRMA artifact.
Gap analysis	Functional specification	The list of identified gaps in the SEMBA deliverable is a valid input for documenting the functional specification in IRMA.
	Business blueprint - business process design	The gap between the SAP legacy systems and the required state of the SAP systems is bridged during the documentation of the process design blueprint for which the identified gaps from SEMBA can be a valid input.
	Business blueprint - data object design	The gap list from SEMBA allows to check whether all the gaps identified have been resolved by the design in the blueprint documentation of IRMA.
Overall migration design	Business blueprint documents	The SEMBA document sets the plan for changing from the as-is situation to the to-be situation which is exactly an input for documenting the blueprint artifacts of IRMA.
To-be design	Business blueprint documents	The IRMA blueprint documents bridge the gap in-between the SAP as-is structure and the expected customized SAP organizational process structure. The

		to-be design form SEMBA can be a useful input for this IRMA artifact.
Future map	system	Future system map describes the requirements that the to-be architected SAP technical infrastructure has to meet. The to-be design from SEMBA will help to identify all the to-be requirements in various levels such as business processes, application landscape and information analysis.

Once the SEMBA deliverables have been identified we proceed with the assessment of the requirements document for this case study. We use only the above selected SEMBA artifacts for the assessment and obtain the following results.

SEMBA deliverables	For case study 2	
<b>As-Is Understanding</b>	APPLICABLE	PARTIALLY FOUND
<b>Business Initiative</b>	APPLICABLE	PARTIALLY FOUND
<b>Business Requirements</b>	APPLICABLE	FOUND
<b>Client Objectives and Expectations</b>	APPLICABLE	FOUND
<b>Gap Analysis</b>	APPLICABLE	FOUND
<b>Overall Migration Design</b>	APPLICABLE	NOT FOUND
<b>To-be design</b>	APPLICABLE	PARTIALLY FOUND

### ***Result from IRMA assessment:***

The given case study falls under the SAP scope of IRMA. Hence, the artifacts of IRMA for SAP are used for assessment. Each artifact of IRMA has a template and IRMA proposes guidelines to fill in these templates. As discussed before IRMA is used as checklist to evaluate the

requirement documents of the case study 2. Specifically, we looked for the content of each artifacts in the requirement documents of Case study 2.

Unlike from IRMA for RUP, IRMA for SAP identifies all the artifacts as mandatory. The following table shows the results of the evaluation. The first column of the table lists the artifacts of IRMA for SAP, second column illustrates the status of corresponding artifact in current case study 2 (21 p. 19).

Name of artifact	Found in case study-2
1. Business blueprint - business process design	Complete (21 pp. 20-21)
2. Business blueprint - data object design	Not found
3. Business blueprint - organizational structure	Partially complete (21 p. 22)
4. Data management migration	Not found
5. Functional specification	Partially completed (21 pp. 23-25)
6. End-user learning strategy	Not found
7. Future system map	Not found
8. Roles and authorization specification	Not found
9. Testing protocol and strategy	Not found

**Table: IRMA Results for Unified ERP**

### 3.2.3. Conclusion

The assessment of the requirements document of case study-2 using the LSPCM model, SEMBA (as check list) and IRMA (as check list). We have described the result of assessment in Section 3.2.2. Based on our experience with this case study we present the suggestion for the case study, LSPCM model, SEMBA and IRMA in Section 3.2.3.1, 3.2.3.2 and 3.2.3.3 respectively.

#### 3.2.3.1. Case study -2: Unified ERP for an organization

Following are the recommendation for the case study Unified ERP:

1. All the symbols used in the flowchart of the processes should be described. Since there isn't any global standard for creating flowcharts. Thus the symbols used in the flowcharts of processes can be misinterpreted by the offshore team.

2. All the inconsistencies between the diagrams and its corresponding text should have been removed in order to avoid the wrong implementation.
3. Inconsistency in the style of process flowcharts must have been removed. For example: In some of the processes flowcharts the optional inputs are marked “optional” but in some of the process flowchart optional inputs are not shown in the diagram.
4. All the business-related terms must be described in the glossary. For example: the terms *production plan*, *production planning* and *demand plan* were not explained in the glossary.

### 3.2.3.2. LSPCM

LSPCM does not have any specific checks for handling the requirements which are specific for SAP system development. The SAP software development method is different since here the designs and specifications are meant to adapt the legacy SAP systems to meet the client business needs. Under such circumstances the development team will be well versed with the SAP package implementation and they document functional specifications which will be used to fill-in the gaps in-between the default SAP organizational process structure to the expected business process of the client. Hence every requirement statement is confined with core SAP implementation and technical constraints which describe the SAP architecture for the selected business context. These facts make LSPCM vague enough to be able to capture the errors in the requirements document. Hence one of the suggestions for improvement in the LSPCM document is in the inclusion of some specific checks which can relate the criteria for formalness, uniformity and conformance to SAP or any other package implementation projects. The following enlists all the specifics which are applicable for package implementation projects but does not have a check to measure its behaviour in terms of formalness, uniformity and conformance.

1. **Problem:** LSPCM requires functional, non functional requirements and glossary in order to start the assessment. But in SAP projects non-functional requirements are documented only during the later phase of the project development life cycle.

**Solution:** There should be an exception in the required elements of the user requirements assessment of LSPCM, when dealing with a package implementation project. This is because



for this case study the client was already convinced with having SAP as the solution for their problem. This means that the client is aware of the different quality requirements assured by SAP which led them to choose SAP as their solution. Then such a documentation of non-functional requirements at the user requirements stage may not be required. The non-functional requirements if any can be explored during the high-level design phase. Hence in LSPCM the non functional requirements may not be applicable in the case of package implementation projects like SAP projects at the user requirements stage.

2. **Problem:** In LSPCM under the uniformity checks of the user requirements product area there is a check for compliance with industrial standard. But in case of package implementation there are no globally recognized standards to support the package implementation life cycle.

**Solution:** This check in LSPCM should not be applicable if it is a package implementation project.

3. **Problem:** LSCPM identifies the process and its corresponding flowchart representations. In a SAP implementation the functionalities are broken down into processes, and these processes are further broken down into sub processes. These sub processes directly map to the functional requirements in an SAP project. LSPCM does not provide checks for checking the uniqueness in name and elaboration (conditions, assumptions, traceability to the business needs) of these sub processes.

**Solution:** Whenever an assessment for a package implementation project is to be carried out the checks for use cases in LSPCM should change into checks for lowest level of processes which is atomic and directly conforms to a functional requirement. For example: One of the checks in the user requirements product area of the LSPCM model states: *“Each use case has a unique name.”* This can be changed to *“Each lowest level process/use case has a unique name.”*

### 3.2.3.3. SEMBA and IRMA

#### ***For SEMBA***

As mentioned before SEMBA does not have a roadmap for an SAP project. Therefore this assessment was carried out in such a way so as to check whether all the relevant details as suggested by the experts, were effectively captured in the given requirement documents. Since the SEMBA-SAP roadmap is still in its development stage we have the following suggestions with regard to the kind of deliverable that can be a part of the roadmap.

1. Since the case study was an SAP project the underlying documents were the business blueprint documents for the SAP implementation. The As-is SAP ERP organizational structure for the mega business processes needs to be elaborated for such package implementation in order to fit the requirements of the new expected business process situation. This is possible only if it known that SAP is the solution. Hence a “*Package selection*” or an “*Approach*” deliverable is feasible if the decision is not made on to which package is to be used as a solution for the desired business context.
2. A deliverable which describes the process variant, conditions, dependencies and assumptions can be a part of the roadmap so that if SEMBA is used as business analysis for package implementation projects which is expected to be followed by IRMA then this can avoid a lot of rework on the information sought by IRMA artifacts.
3. The “*Business requirements*” document is an important deliverable which can be used for the preparation of the blueprint documents in such a way that it conforms to clients objectives and expectations, since mostly such requirements are gathered by conducting workshops which involve all the major stakeholders.
4. A “*Gap analysis*” deliverable is also desirable in order to develop the functional design of the desired SAP system.
5. Requirements document set can have a complete “*migration design*” document which will check whether the identified gaps have been neatly filled-in by the design plan.

6. The “*business processes to-be*” must be described in the to-be process definitions and can be the part of SEMBA deliverable.

**For IRMA:**

It is known from the interview with software quality analyst (SQA) of case study-2 that IRMA for SAP was not used for requirement gathering and elicitation; instead the standard DeliverSAP was used for requirement gathering and elicitation. Many artifacts of IRMA for SAP has been directly inherited from the DeliverSAP standards, hence the requirement document of case study-2 is partially in accordance with guideline and template of IRMA for SAP.

As it is describes earlier in Section 2.3, IRMA for SAP has nine mandatory artifacts. Every artifact has some mandatory and some optional sections. During the assessment of requirement documents of case study-2 following issues were observed in the guidelines and templates of IRMA for SAP:

1. Redundant section:
  - a. In the guidelines of functional specification artifact of IRMA for SAP the Section 7 and Section 10 has the same name and same content.
  - b. In the template of functional specification artifact of IRMA for SAP the Section 6 and Section 9 has the same name and same content.
2. Missing section: The table of content of guideline for functional specification artifact of IRMA for SAP demands Section 2 to be logical data design but in the description this Section is missing.
3. The sections of guidelines should always correspond to the same section in templates. But the guidelines of functional specification artifact of IRMA for SAP the Section 7 as data design but in the template of functional specification Section 7 is transaction design.
4. The guidelines for all the nine artifacts of IRMA for SAP specify few sections as mandatory and few sections as optional, but there are sections for which it is not prescribed neither as mandatory nor as optional.
5. The subsection Performance criteria of functional specification artifact of IRMA for SAP demands the conditions which will affect the performance. Unlikely from IRMA for RUP this

does not detailed on the performance criteria, priority of these performance criteria and the impact on the overall system if these criteria are not met.

6. In the subsection SAP global setting of business blueprint – SAP organizational structure artifacts of IRMA for SAP demands the details for SAP configuration. For example name of the countries, the regions inside a country and currencies *etc.* But in none of the artifacts prescribe anything about language of SAP GUI screens.
7. Unlikely from IRMA for RUP, in IRMA for SAP all the artifacts are not prepared by system analyst. Since the different roles are involve in preparing different artifacts it is necessary to identify an order in which these artifacts should be are prepared because one can't prepare test plan untill the requirement which has to be tested is fixed.

## 4. Reflection

This chapter discusses how the **causes** discovered from the **effects during the case study** will reflect on the certification model and the industrial methods studied during this research. We are focused only on the business and IT requirements gathered with an intention of offshoring the development of the proposed system. From the previous chapter on case studies we derive all the relevant suggestions for improving the studied techniques with respect to the particular case study. Here, we will elaborate on the overall improvements suggested for the specific checks in LSPCM and in the guidelines for the method artifacts which has been learned from the case studies.

### 4.1. Reflection on LSPCM

The current version of LSPCM was not able to identify a number of the issues in the requirement documents of offshored projects. Following are the issues which will lead to improvements in the LSPCM model:

1. In practice business requirements are gathered first which is then transformed into IT requirements. Therefore it is important to assess the quality of business and IT requirements before offshoring the development. In LSPCM the business context and IT requirements of a project is covered under the product areas "*Context description*", "*User requirements*" and "*High level design*". The certification types for the product areas "*Context description*" and "*High-level design*" is missing in the LSPCM document. Hence the complete assessment of business and IT requirements cannot be done with the current version of LSPCM.
2. A company "X" makes all its project documents complying with its own standards. Now company "X" wishes to offshore its development to a company "Y" which has its own standards. This means that the two companies may have different standards which might not have a common ground over which both can comply. Hence in the case of offshoring it is essential to check whether the prepared documents comply with an industry standard.

But in LSPCM the compliance with industry standards under the ***uniformity checks*** is never considered for deciding the certification levels.

3. From the second case study we understood that the client was already convinced with having SAP as the solution for their problem. This means that the client is aware of the different the quality assured by SAP which led them to choose SAP as their solution. Then documentation of non-functional requirements at the early stage of requirements elicitation may not be essential. The non-functional requirements if any can be explored at a later phase of project development. But in LSPCM we cannot proceed with the assessment unless the non functional requirement documents are present.
4. In LSPCM there are only checks for checking the consistency of use cases and their corresponding diagrams. For package implementation projects there are no use cases and instead there are processes and process diagrams whose consistencies must be checked.
5. If the diagrams in the requirements document do not follow any industry standard then there is no common understanding for the symbols used in these diagrams. There is a possibility for wrong interpretation when the documents are send to the offshored end. LSPCM checks that the requirement documents are compliant with industry standards. But result of the check “compliance with industry standards” never considered for deciding the achievement level.
6. In the first case study we observed that there were inconsistencies in the use of language for specifying the business and project related terms because the GUI was expected to be build in Dutch whereas the offshored team language is English. For example in the GUI illustration it is “*Stagiair scherm*” and in its description it is “Intern screen”. This was an example of a mix in the languages Dutch and English. The similarity is not obvious unless the developers are proficient in both the languages. LSPCM does not provide any checks for identifying the same.
7. During the SEMBA/IRMA assessment of the case studies we identified incomplete non-functional requirements. These incomplete parts were not identified by LSPCM.

8. During the SEMBA/IRMA assessment of the case studies we identified that the non-functional requirements were not prioritized. LSPCM does not have any check for checking whether the non-functional requirements were prioritized.
9. From the SEMBA/IRMA assessment of the case studies we identified that the “*Client objectives and expectations*” were not found. This was important for offshoring since the offshored team does not have a direct interaction with the client and will be aware of the client expectations only through the documentations. LSPCM does not have a check in the **required elements** of *Context description* product area, which will check the availability of such a “*Project initiation document*” which will document the client’s vision, objective and expectations.

Following are the reflections on LSPCM model in order to assess the requirement documents of offshore projects effectively:

1. The certificate types for the product area “Context description” and “High-level design” should be included in the model. For example: For each certification criteria in LSPCM there are number of specific criteria. LSPCM does not specify which specific criteria should be satisfied in order to achieve any certification level for “Context description” and “High-level design” product area. In this research we focus on the business and IT requirement and these requirements are mapped in “Context description”, “User requirement” and “High-level design” product area of LSPCM. The certification types of the “User requirement” product area are readily available. In similar way LSPCM should include the certification types for the product area “Context description” and “High-level design”.
2. The compliance with industry standards under the uniformity checks of LSPCM is necessary to be considered for deciding the certification levels.
3. There should be an exception in the required elements under the specific criteria “Formalness” of the user requirements assessment of LSPCM, when dealing with a package implementation project. The non-functional requirements should be **not applicable** in the case of package implementation projects.

4. Whenever an assessment for a package implementation project is to be carried out the checks for use cases in LSPCM should change into checks for lowest level of processes which is atomic and directly conforms to a functional requirement.
5. In the specific criteria “Formalness” of any product area, the required elements should also include description of all the symbols used in the pictorial representation of process models or data models or flowcharts. This check should be mandatory if industrial standards are not used for making these models.
6. In the specific criteria “Formalness” of high level design product, the translated-terms dictionary should be a required element. The translated-term dictionary should contain all the business and project environment related terms from other language. For example: if requirements document is written in English and it contains a number of business or project terms in Dutch and offshore team do not understand Dutch. In this case all the Dutch terms with its translation should be present in translated-terms dictionary.
7. In the specific criteria “Conformance” of high level design product a check should be added to check that all other language terms are present in the translated-terms dictionary.
8. The certification criteria “Formalness” of user requirement product area require presence of non-functional requirements. The non-functional requirement should be more detailed (a list of quality attribute for specific domain<sup>5</sup> can be added) and more checks for should be included in the “Conformance” specific criteria to check the consistency of non-functional requirement.
9. Check for prioritization of non-functional requirements should be added in the “Conformance” specific criteria of user requirements product area.
10. There should be a check in the required elements for client vision, objectives and expectations under the specific criteria “Formalness” of the context description product area of LSPCM.
11. LSPCM has a check for all the product area “The certification property is relevant and feasible”. The definition of the certification property should be elaborated more with example in order to improve the usability of the LSPCM model.

---

<sup>5</sup> Domain could be real-time software or information system or scientific software or internet application software.



12. The check for compliance with industry standard in LSPCM should not be applicable if it is a package implementation project.

#### ***4.2. Reflection on SEMBA***

This research work was organized for a duration within which we were conveniently able to accommodate two different industrial case studies. Though this is limited for a topic like offshored projects, the results obtained were valuable enough for a small step towards a change. In this research work we focused on both process quality approach and product quality approach. SEMBA was one among the process quality approach followed in the industry. Since SEMBA focuses on the business analysis part of a project, it results in delivering the key business requirements for the system to be build. In order to summarize our major findings on this research with respect to the quality of business requirements for offshored projects we suggest the following recommendations for improvement in the SEMBA method:

1. SEMBA being an expert method requires some level of subject matter expertise for following the activities and rendering the required deliverable. Though the method allows significant opportunity for a flexible business analysis, it hinders the possibility of SEMBA being used by an external third party with low or negligible knowledge over business analysis. A standard method must lean more towards its ease of use and convenient value for the expended effort and time. The effort includes the required man power and the cost involved. If we require an expert to use the expert method then probably we would have to choose one of them in order to be effective in terms of cost and time. We suggest the following:
  - a. Categorize all the projects from where the best practices were learned and documented in SEMBA. This can be according to the kind of project, technology used and priority of quality factors.
  - b. Identify the mandatory and optional deliverables for each of the above categories.
  - c. Construct an activity plan for each of these categories which would indicate the sequence in which the deliverables must be prepared.

Using the outcome of the above suggestions for SEMBA, we can ensure that the standard expert method would result in generating the same results for business analysis irrespective of the analyst using the standard.

2. SEMBA in itself is a project which would require time and budget to make it realizable. A table indicating the cost and effort required for each of the project category can be useful information for a third party user of the expert method.
3. SEMBA focuses on the process quality of business analysis. Process quality can be assured only if SEMBA is actually followed in the right way. The process trails can be audited with respect to the guideline information for each of the deliverables.
4. In order to ensure that none of the requirements are miscommunicated while offshoring the development part of the project it is essential to have SEMBA carried out as an onsite project and later translated into a document understood by the offshore team. This reduces the risk of misunderstanding and ambiguities in the business requirements set.
5. SEMBA can be carried out in two ways, either in an iterative way or in a linear way. By iterative we mean that the deliverables are susceptible to changes during the course of the requirements gathering process without changing the scope of the project. And by linear we mean to say that there will not be any changes once the deliverables of SEMBA are finalized and any changes in the linear approach may lead to re-documenting the entire SEMBA deliverables pertaining to the particular project. This would expend a considerable amount of time if a change management process is not formulated. This also applies to the iterative way of doing SEMBA. If the approved set of deliverables are considered final in both the way of SEMBA practice then this issue subsides. But if there is still scope for a change then the SEMBA method will not be supportive. This makes SEMBA less flexible. Hence, a way to improve the flexibility of SEMBA is to introduce a change management document within SEMBA.

Since the SEMBA-SAP roadmap is still on its way to be delivered, there were a few suggestions made during the second case study which can be used as an input for choosing the kind of

deliverables and formulating the set of guidelines in order to make a smooth transition from SEMBA to SAP artifacts. This is done to improve the reusability of the deliverables from SEMBA.

The SEMBA method would have been a good method to be followed for doing business analysis. But when we analyzed the industrial offshored projects with respect to SEMBA we found that some of the deliverable guidelines would still allow for documenting requirements which might be miscommunicated while being offshored. Such defects were identified by LSPCM. As a result we came up with the above suggestions which is reflected back into the SEMBA method so as to help it to document requirements without giving any scope for being miscommunicated. The details of the findings reflecting back into the SEMBA method is as shown below:

1. **Finding:** SEMBA suggests the business analyst using SEMBA to have the following capabilities:

- subject matter expertise e.g. mortgages, pension etc
- knowledgeable on the commonly used system development processes and package implementation processes
- understanding the relationship between IT and non-IT aspects of a project
- identify steps and event to be able to validate the solution against business requirements and system requirements

The problem with identifying above mentioned qualities for a business analyst who will be using SEMBA, would give an impression that SEMBA can be used only by an expert “*Business analyst*” though it is by itself an expert method. This implies that if we attempt to use SEMBA we must allot budget for carrying out SEMBA as a project by itself and also meet the additional expenses for hiring an expert business analyst.

**Impact on offshoring:** The cost encountered by practicing SEMBA might be high in such a way that it covers the cost effectiveness achieved by offshoring the development of the project.

**Reflection on SEMBA:** SEMBA can be extended and elaborated in such a way that the level of business analyst who will be using the method will not be an issue which will affect the quality of the SEMBA documents delivered.

2. **Finding:** If SEMBA is expected to be followed by IRMA then the deliverables produced by SEMBA must be a useful input for preparing the IRMA artifacts in order to reduce the amount of rework. Though there exists a SEMBA-IRMA roadmap, there is no clear indication of mandatory and optional deliverables of SEMBA in case of different projects.

**Impact on offshoring:** From the case studies it was observed that deliverables like “*As-is understanding*” and “*Opportunity report*” documents were missing in the requirements documents which led to miscommunication of requirements as identified in Chapter 3.

**Reflection on SEMBA:** SEMBA states that the “*Scope*” and the “*Approach*” documents of the SEMBA deliverables determine whether or not a particular deliverable will or will not be created. But since SEMBA is an expert method such decisions can also be handled by the expert method itself rather than depending on the level of expertise of the business analyst involved. Hence we can categorize all the projects from where the best practices were learned and documented in SEMBA. This can be according to the kind of project, technology used and priority of quality factors. The mandatory and optional deliverables for each of these categories can be made certain.

3. **Finding:** SEMBA cannot identify contradictory requirements statements. If a requirement states that property **P** is required then it might be the case that there is a requirement which states that property **not P** is required.

**Impact on offshoring:** Incorrect behavior of the system delivered by the offshore team.

**Reflection on SEMBA:** A sentence in the guidelines of the “*Business Requirements*” document indicating that “**no two requirements must contradict each other**”.

4. **Finding:** SEMBA is not flexible with changing requirements. If once the requirements are finalized there is no scope for a change. This means that if there is an essential change then this would result in a chain of changes along all the prepared SEMBA deliverables.

**Impact on offshoring:** In the case studies we found that the requirements and even the solution approach were changing in between when the project development was

progressing. This can lead to changes in the design documents which will eventually lead to a change in development. It is indeed an undesired situation but still it happened. We can either prevent such a situation or manage the situation. We have made an attempt to think in the ways to manage the situation.

**Reflection on SEMBA:** A way to improve the flexibility of SEMBA is to introduce a change management document within SEMBA which must be able to track precisely the changes in the set of delivered SEMBA documents in case of a change. SEMBA says that it seamlessly integrate with other standard change management approaches like “*Unified project management*” and “*Planned change management*”. In this case a roadmap indicating this integration might be desirable.

5. **Finding:** The SEMBA-SAP roadmap is not available for use.

**Impact on offshoring:** The staged delivery plan of the SAP system was not documented. A document indicating the available time and budget for the project with the distribution of work over the available time was missing. This might lead to missing the deadlines for the project.

**Reflection on SEMBA:** The SEMBA-SAP roadmap must contain a deliverable indicating the agreement between the client and the service provider on the staged delivery plan of the project.

#### **4.3. Reflection on IRMA**

In order to adequately answer the research problem (see Section 1.4), we carried out two case study and as a result of this assessment we identified the limitations of the approaches IRMA for RUP and IRMA for SAP.

Following are the identified limitations of the IRMA for RUP approach:

1. As we observe in the requirement document of the case studies that the number of business terms, project terms and acronyms were in Dutch. For example: In use case diagrams the actors name were given in Dutch where as in use case specification the name of actors were given only in English. These kind of inconsistency will remain in the

requirement documents even after using IRMA for requirement elicitation because in the current version of IRMA does not dictate the language for terms used.

2. The guideline of system rule artifact of IRMA for RUP demands that each system rule should have a unique ruleID but it does not demand the uniqueness in the definition of system rules. Hence there can be redundant system rule even after using IRMA for requirement elicitation.
3. The requirements document may contain contradictory system rules even after using IRMA for requirements elicitation because IRMA demands that each system rule should have a unique ruleID. Additionally IRMA also recommends to “SMARTify<sup>6</sup> requirements”. These instructions of IRMA do not assure that system rules are not contradictory.
4. The guidelines of use case model and use case specification artifacts of IRMA for RUP do not provide any rule which checks the conformance of use case model with use case specifications. Hence IRMA does not assure the consistency between the use case models and use case specification.
5. The current version of IRMA for RUP does not assure that all the use case specification are traceable to high-level requirements (e.g. business needs) because in the guidelines of use case specification artifact of IRMA for RUP there exist no rule which demands the traceability of the use case specification to high-level requirements.

Following are the reflections on the guideline of the artifacts of IRMA for RUP in order to address above mentioned limitations:

1. The glossary should also include the translated-terms dictionary when the GUI of system to be build is going to be in a language which different then offshore development team language. This can be added in the guidelines of glossary artifacts of IRMA for RUP.

---

<sup>6</sup> IRMA defines **SMART** as:

**Specific** (unambiguous): A requirement must be stated in such a way that there is only one way to interpret it.

**Measurable**: A requirement must be stated in such a way that one can measure whether the system conforms to the requirement.

**Acceptable**: A requirement must be stated in a way that the client is able to determine whether the requirement is met.

**Realizable/Realistic**: Requirement must be realizable within given time and budget.

**Traceable**: Requirement must be traceable to high-level requirements (e.g. business needs).

.

2. A check can be added on specification of new system rule. One should check while adding a new rule, that whether the new rule is conforming to existing rules or not? This can be added in guideline of system rule artifact of IRMA for RUP.
3. There should be unique ID and unique definition of system rule. This instruction can be included in guideline of system rule artifact of IRMA for RUP.
4. There should be a conformance check between use case model and specification. This can be added in the guideline of use case specification artifact of IRMA for RUP.
5. IRMA for SAP demands the traceability of each functionality or test plan or performance requirement to business need. This can be added in the use case specification artifact of IRMA for RUP, where once can write the traceability of use case to business needs.

The second case study comes under the SAP scope of IRMA. Hence the guideline and templates of artifacts of IRMA for SAP were used for the assessment. As mentioned in Section 3.2.1, the requirement documents were not complete when assessment was carried out. Hence only three artifacts business blueprint – business process design, business blueprint – SAP organizational structure and functional specification of IRMA for SAP were found in the requirement documents of case study 2. During the course of assessment a number of issues were observed in the guidelines and templates of the artifacts of IRMA for SAP and we have documented these issues in the Section 3.2.3.3. In order to address the issues documented in Section 3.2.3.3 we present following suggestions for the guidelines and templates of IRMA for SAP:

1. Redundant sections should be removed from the guideline and templates of functional specification artifact of IRMA for SAP.
2. Either the section logical data design should be removed from table of content of guideline for functional specification artifact of IRMA for SAP or the description of section logical data design should be added.
3. The sections of guidelines should always correspond to the same section in templates.

4. In all the artifacts of IRMA for SAP few sections are identified as “mandatory” and few sections are identified as “optional”, but there are sections for which are not identified neither as mandatory nor as optional. These sections can be specified as either group.
5. The subsection performance criteria of functional specification artifact of IRMA for SAP demands the conditions which will affect the performance. Unlikely from IRMA for RUP this does not detailed on the performance criteria, priority of these performance criteria and the impact on the overall system if these criteria are not met.
6. In the subsection SAP global setting of business blueprint – SAP organizational structure artifacts of IRMA for SAP demands the details for SAP configuration. For example name of the countries, the regions inside a country and currencies *etc.* But in none of the artifacts prescribe anything about language of SAP GUI screens. This can be added.
7. Unlikely from IRMA for RUP, in IRMA for SAP all the artifacts are not preprepared by system analyst. Since the different roles are involve in preparing different artifacts it is necessary to identify an order in which these artifacts should be are prepared because one can’t preprepare test plan untill the requirement which has to be tested is fixed. Hence the order of in which artifacts should be preapred can be dictated by IRMA.



## 5. Conclusion

The presented work is aiming towards finding a complete quality certification model for assessment of requirements from business and IT perspectives for offshoring. We have tried to improve existing LSPCM model and SEMBA/IRMA methods for offshoring. This chapter discusses the main results of our project. First, we will briefly mention what the problem was and how it was solved. Subsequently, we will analyze the achieved goals. After that, suggestions for possible directions of future work are given.

### 5.1. Problem & Solution

Project failures form an issue faced by many software service provider companies. The surveys (6) showed that this issue has gradually evolved into a major industrial concern. Research has been done in order to extract the real risk factors (12) involved in such software failures. Offshoring the development part has significantly lowered the development cost involved but the quality of the product in terms of the customer satisfaction was compromised (5). This indicated that we had the following problem:

*“Miscommunication of requirements contributes significantly for offshored project failure (as explained in Section 1.1. Are the requirements quality certification models effective to capture all those defects which lead to miscommunication? Do the company standards for gathering business and IT requirements for offshored projects; need some improvement in order to minimize the defects leading to miscommunication?”*

To solve the above problem we came up with the following strategy:

*“Apply the certification model to industrial projects and list the defects which contributed to miscommunication of requirements. Then map these defects to the process which led to the gathering of defective requirements. Suggest improvements in the methods followed by the industry, in order to trace down the requirements which would never be defective. Next suggest improvements in the certification model by including new checks which would decipher the previously un-deciphered defects in the requirements learned from the industrial projects.”*

### **5.2. *Main result***

The main result of this research is the suggested improvements for the certification model LSPCM and the industrial methods SEMBA and IRMA. This work included only the requirements documents and therefore was successful in identifying the major issues related to requirements, which led to failure of offshored projects. As a result the suggestions would help in minimizing the risks due to miscommunicated requirements for offshored projects both in terms of the product delivered and the process followed. Hence this work emphasizes on the continuous improvement at two levels; one at the process (requirements gathering) level and second at the product (requirements document) level. We conclude improved guidance and procedures for the process involved and better assessment and evaluation for the product considered.

### **5.3. *Future work***

To assess the quality of requirement is a broad area of research that can evolve in various directions. In this research work we focused on the business and IT requirements of offshore projects. During this research we observed several opportunities to further extend our research.

In this research we analyzed the LSPCM model in comparison with SEMBA/IRMA company standards. This can be further extended by comparing LSPCM model with other company standards leading to improvements in both LSPCM and the company standards. By doing this we will be able to explore different risk factors for offshoring addressed by these new company methods. The identified strengths of the newly analyzed company standards will be used to improve the checks in LSPCM.

Since we were working on this research in collaboration with Capgemini, we carried out the case studies on the requirement documents of information system projects. Furthermore, the LSPCM can be applied to other offshore projects (e.g. real-time projects or scientific projects) in order to identify the specific check related to other domains. There can be different issues and

risk related to each domain of the projects and these issues and risk can be unique for each domain. The impact of offshoring on the projects of different domain may differ from the one studied in this research work. It might be interesting to identify new ways of addressing such risks during offshoring of these projects.

Another appealing area of research could be to assess the quality of design documents for offshoring. In our research we focused on the case studies where the requirements are gathered onsite and the design and development phases were offshored. It will be interesting to assess the requirements as well as the design documents of a project where only development phase is offshored. One can analyze the LSPCM model in comparison with the company standards for design documents (e.g. Capgemini Accelerator for Software Architecture (**CASA**)). Since the certification checks and criteria for detailed design product area is readily available in current version of LSPCM model, hence this can be applied on the design documents of offshored projects. The result of assessment of LSPCM and company standards can be examined to discover the limitations and advantages of the model and company standard.

## Bibliography

1. *A Software Product Certification Model*. **Petra Heck, Martijn Klabbers , and Marko van Eekelen**. s.l. : Springer US, 2009, Software Quality Journal.
2. *Computer & Systems Failure Litigation*. **Warren S. Reid**. October 2005, The Journal of Quality Assurance Institute, pp. 22-27.
3. **McGraw, Bruce**. Why Software Projects Fail. *Fear No Project – A Project Management Blog*. [Online] Gartner, Inc, May 1, 2009. [Cited: June 1, 2009.] <http://fearnoproject.com/2009/05/01/why-software-projects-fail/>.
4. *Chaos Report*. s.l. : Standish Group, 1994.
5. *A Replicated Survey of IT Software Project Failures*. **Khaled El Emam, A. Gunes Koru**. s.l. : IEEE Computer Society, 2008, IEEE Software.
6. *IT Failure Rates--70 Percent or 10-15 Percent?* **Glass, Robert L**. 3, s.l. : IEEE Computer Society, 2005, IEEE Software, Vol. 22, pp. 110-112.
7. *Why Software Fails*. **Charette, Robert N**. s.l. : IEEE, 2009, IEEE Spectrum.
8. **Quality, American Society for**. Glossary - Entry: Quality. *American Society for Quality* . [Online] ASQ. [Cited: June 5, 2009.] <http://www.asq.org/glossary/q.html>.
9. *"Offshoring" of IT services: The impact on US economy*. **Chakraborty, K. and Remington, W**. 2005. Central Plains Conference. pp. 112-125.
10. **Warren S. Reid**. *Reviving the Drowning Large-Scale IT Project*. California, USA : WSR Consulting Group, LLC, 2007.
11. **Aspray, W., Mayadas, F., and Vardi, M.Y**. *Globalization and Offshoring for Software*. NY : ACM, 2006. A Report of the ACM Job Migration Task Force.
12. *A Risk Profile of Offshore-Outshored Development Projects*. **Nakatsu, Charalambos L. Iacovou and Robbie**. 6, s.l. : ACM, June 2008, ACM communication, Vol. 51, pp. 89-94.
13. **James Robertson, Suzanne Robertson**. *Requirements Specification Template*. s.l. : Volere, 2007.
14. **4, ISO/IEC 9126 -**. *Software engineering -- Product quality . ISO/IEC 9126 - 4* 2004.
15. **Rational Software**. *Rational Unified Process*. s.l. : Rational, The software development company, 2001.

16. *A Software Product Certification Model*. **Yahaya, Jamaiah Haji, Deraman, Aziz and Hamdan, Abdul Razak**. s.l. : ICTTA 2008, April 7-11, 2008, Information and Communication Technologies: From Theory to Applications, pp. 1-6.
17. *Software Product Evaluation and Certification*. s.l. : Final Presentation and Training, conference proceedings at ESA(European Space Agency), 2001.
18. **Tools Database Working Group-INCOSE**. INCOSE Requirements Management Tools Survey . [www.incose.org](http://www.incose.org). [Online] INCOSE, March 17, 2009. [Cited: June 5, 2009.] <http://www.incose.org/ProductsPubs/products/rmsurvey.aspx>.
19. *Requirements for requirements management tools*. **Hoffmann, M. Kuhn, N. Weber, M. Bittner, M.DaimlerChrysler AG Res. & Technol., Ulm, Germany**. s.l. : IEEE International, September 6-11, 2004, Requirements Engineering Conference, 2004., pp. 301-308.
20. **Amrita, A. and Sree Kumar, A.** *Case study-1*. Computer Science & Engineering, TU/e. 2009.
21. **Amrita, A. Sree Kumar, A.** *Case study2*. 2009.
22. *The Impact of IT Outsourcing Practice on Employment Culture: An IT Professional's Perspective*. **Juntiwasarakij, Suwan**. Charlottesville, Virginia, USA : ACM, April, 2008. SIGMIS-CPR'08. pp. 164-167.
23. *Automated requirements management*. **Hammer, T. and Huffman, L.** Colorado Springs, CO, USA : IEEE, 1998. Requirements Engineering, 1998. Proceedings. 1998 . pp. 34-40. 5912307.