Requirements engineering in market-facing projects: a case study

I'LL NEED TO KNOW YOUR REQUIREMENTS BEFORE I START TO DESIGN THE SOFTWARE.
E-mail: SCOTTADAMS@AOL.COM
FIRST OF ALL, WHAT ARE YOU TRYING TO ACCOMPLISH?
I'M TRYING TO MAKE YOU DESIGN MY SOFTWARE.
© 2006 Scott Adams, Inc./Dist. by UFS, Inc.
I MEAN WHAT ARE YOU TRYING TO ACCOMPLISH WITH THE SOFTWARE?
I WON'T KNOW WHAT I CAN ACCOMPLISH UNTIL YOU TELL ME WHAT THE SOFTWARE CAN DO.
TRY TO GET THIS CONCEPT THROUGH YOUR THICK SKULL: THE SOFTWARE CAN DO WHATEVER I DESIGN IT TO DO!
www.dilbert.com
CAN YOU DESIGN IT TO TELL YOU MY REQUIREMENTS?

© Scott Adams, Inc./Dist. by UFS, Inc.

Author: Martin Wesselink

Requirements engineering in market-facing projects: a case study

Date
October 3, 2011

Author
Martin Wesselink
Business Information Technology,
School of Management and Governance
Student number 0196614
E-mail m.g.wesselink@student.utwente.nl

Graduation committee
Dr.ir. Hans Moonen
Department of Information Systems and Change Management
University of Twente, School of Management and Governance
E-mail hans.moonen@utwente.nl

Dr. ir. Maya Daneva
Department of Information Systems
University of Twente, School of Computer Science
E-mail m.daneva@utwente.nl

Geoffrey Davitt
Department Chubb Insurance Group, IT
E-mail gdavitt@chubb.com

George White
Department Chubb Insurance Group, IT
E-mail ghwhite@chubb.com

Executive summary

Chubb wants to improve their business processes by automating parts of their communication channels between them and their brokers. Therefore, Chubb is developing market-facing systems that address these business processes. However, the current approach that Chubb is using for developing systems is mainly suitable for developing systems based on existing systems. Due to that market-facing systems are systems that are not based on existing systems; the current approach is not optimal for the development of market-facing systems. Therefore, Chubb asked us to do a research on gathering, negotiating, validating and prioritizing requirements of market-facing projects from a multi-stakeholder perspective. What brings us to the central question of this research, namely:

How to gather, negotiate, validate, and prioritize requirements of market-facing projects from a multi-stakeholder perspective?

In order to answer this question, this research is divided into five sub questions. Namely;

1. What methods/approaches for similar projects and companies are suggested in scientific literature?
2. How does Chubb currently gather, negotiate, validate and prioritize the requirements of market-facing projects?
3. What can be improved in the way Chubb gathers, negotiates, validates and prioritizes requirements reviewing the current practice and comparing against scientific literature?
4. What would be a better way for Chubb to carry out requirements engineering for market-facing projects?
5. How to evaluate the proposed framework/methods?

Literature study
The first sub question is answered based on a scientific literature study. The papers found and used in this study are selected based on certain inclusion and exclusion criteria. The overall requirement engineering approaches found in this literature study are mainly based on plan-based and agile strategies. Plan-based strategies consist of an overall implementation plan and agile strategies consist of small iterations where stakeholders have to work intensively with each other to achieve the highest satisfaction. The techniques found in the literature study to gather requirements are; grouping diverse stakeholders, interviewing experts, the use of case scenarios, the use of default templates and documenting requirements and decisions. To enhance the negotiation process, the literature suggest to keep consistency by the use of an ontology, to negotiate iteratively to solve conflicting demands, to consider the emotional factors to gain a better understanding of stakeholders and their requirements and to use a collaboration system to document the requirements and improve the accessibility of it. For the validation of requirements, the literature suggests to review viewpoints by multiple and different stakeholders, to use cognitive profiles in order to gain a better understanding of the requirements by the stakeholders and to use automated validation mechanisms in case of vast amounts of requirements to speed up the process. Further, mechanisms that are suggested by the literature to prioritize requirements are cost and benefit mechanisms and cumulative voting. In the former, stakeholders collaborative determine what the cost is of implementing each potential requirement and determine how much value the requirement will gain. In the latter, each stakeholder is given an amount of 100 points that he or she has to use to vote on his or her most important requirements. The requirements that have the most votes are marked as important and requirements that have the least votes are marked as not necessary or will be dropped out of the project.

Case study
The second sub question is answered through an exploratory case study that is conducted within Chubb. In this case study, experts were interviewed, available documentation was analyzed and direct observations were done to collect data. Findings from this case study suggested that the overall requirements engineering within market-facing projects in Chubb are done based on an iterative approach where in prototypes are used to negotiate, validate and prioritize requirements and to gather additional requirements. In this process a business specification document is created where in all the requirements, specifications and decisions are logged. The stakeholders in market-facing projects are chosen among three aspects, namely; their expertise, their financial sponsorship and their delivery responsibilities. Techniques that are used to gather requirements are; market-research, workshops, case scenarios and face-to-face meetings. In the negotiation process, Chubb is using visualization tools, face-to-face meetings and a collaboration system to increase the understanding of the

requirements by the stakeholders and to facilitate discussions. If there are conflicting requirements, the requirements engineer reflects these concerns to the scope of the project to determine if there are inline with it. Besides this, the requirement engineer validates the concerns by making everyone aware of the conflict and by retrieving their opinions about it. Further on, the requirement engineer also estimates how much time and money it will cost to realize these conflicting requirements. Based on these aspects, the requirement engineer decides whether to drop or to add the conflicting requirements to the development of the system. The prioritization of the requirements in market-facing projects in Chubb is mainly done through two techniques, namely reflecting the requirements back to the scope of the project and by determining the return of investment on the requirements. When a requirement delivers a high return on investment and is in scope of the project, the probability that it will gain a high priority is most likely. When a requirement delivers a low return on investment and is not inline with the scope of the project, the probability that it will gain a low priority is most likely.

Recommendations

The third question is answered based on the findings of previous research questions. Elements from the literature and the current approach are analyzed to improve the current approach. The overall approach that is currently used can be improved by using the approach proposed by Sen and Hemachandra in [1]. With this approach, Chubb could reduce the chance that requirements may not be recognized or recognized too late. For the gathering of requirements, Chubb could use the cognitive tool proposed in [2] to gain a better understanding of the issue by the business experts. In this way, the business experts would find it easier to express their thoughts on the new system. Further, case scenarios can be used to capture the purpose of the system and the reason why a given design will meet that purpose. Finally, templates can be used in the gathering process to speed up the requirements validations and prioritization process. To enhance the negotiation process, ontology can be used to reduce the chance of misinterpretations of the requirements by the stakeholders and a collaboration system could be used to improve the cost effectiveness, user satisfaction and the final outcomes of the project. The validation process could be improved by using the model suggested by Ahmad in [3]. This model addresses the problem of ambiguous and conflicting requirements by negotiating and validating the requirements in iterations that consists of pre-defined steps. In the validation process, Chubb could use the cognitive method of Carod and Cechich in [2], previously mentioned. With this method, they can improve the sign off technique that is currently used by presenting the requirements in a way that the stakeholders feel most comfortable with. Through this, the stakeholders will better understand what he or she needs to sign off on. Finally, the prioritization process in market-facing project could be improved by using the model of Recheva et al. [4] and the cumulative voting method proposed by Leffingwell and Widrig in [5]. The model of Recheva et al. in [4] could be used as guideline to make sure that the stakeholders consider all important aspects in the prioritization process and the cumulative voting method proposed by Leffingwell and Widrig in [5] could be used to reduce the gap of human subjectivity in the prioritization process.

A framework is made based on these recommendations to answer the fourth research question (see page 59); “What would be a better way for Chubb to carry out requirements engineering for market-facing projects?” This framework consists of small iterations where each iteration consists of a pre-sprint, sprint and post-sprint process. In the pre-sprint process, requirements are elicited through the following techniques;

(i) Market research
(ii) Stakeholder selection based on expertise of the product, sponsors and deliverables
(iii) Interviewing based on cognitive profiles
(iv) Existing system analysis
(v) Case scenarios

In the sprint process, the requirements elicited in the pre-sprint process are negotiated, validated and prioritized. In this process, ontology should be used to reduce misunderstandings and notes should be stored in a collaboration system to understand later on why decisions are made. To solve conflicting goals or requirements, the spiral model of in [3] should be used to iteratively solve those conflicts. Further on, the goals and requirements should be validated through multiple sources and by the sign off technique. To prioritize requirements, the aspects mentioned in [4] should be addressed. In the post-sprint process, the requirements and goals should be documented in the business specification document.

Validation

The last research question: “How to evaluate the proposed framework?” is through some articles found in scientific literature. Winbladh mentioned in [6] 6 guidelines on where a requirements engineering framework needs to consists of. These guidelines are used to evaluate the proposed framework and from this it turned out that the proposed framework covers all guidelines to some extend. Further on, the metrics of Costello and Lui in [7] could be used to measure the impact of the framework in practice. However, these metrics measure only quantitative data. To fully evaluate the framework, also qualitative data need to be considered. In order to retrieve qualitative data, interviews need to be conducted with stakeholders that are familiar with the existing approach and the proposed framework in this thesis.

The implementation of the new framework needs to be done after or simultaneously with a project that is using the existing approach where in the metrics of Costello and Lui [7] are used. In this way, data from both projects can be compared and evaluated.

Limitations and implications for further research

- Due to the fact that this research is only conducted within Chubb, it has its limitations in generalizing the results for scientific field and other practices. To generalize the findings from this study for the field of science and other practices, this study also needs to be conducted in other companies that are developing market-facing systems with multiple stakeholders.
- The researcher did an extensive literature study and found articles based on predefined inclusion and exclusion criteria to reduce the chance of missing important literature. However, during this study the researcher found a vast amount of literature and due to time limitations it could be that there is more literature that could be relevant for this research. Therefore, more research needs to be done to find elements that can contribute to the requirements engineering process in market-facing projects.
- Some interviewees (experts) mentioned that developers do not always understand the business. However, other sources in this study did not indicate this. To find out if developers understand the business, more research needs to be done.
- According to the business experts, the selection of stakeholders is done among three aspects (sponsorship, expertise and deliverables). Nevertheless, other sources in the case study only showed that these aspects are present among the stakeholders but do not provide evidence that the selection of stakeholders is really done among these aspects. More research is required to find out, how stakeholders are selected in projects. Knowing how stakeholders are selected in project could contribute in understanding the requirements among the stakeholders.
- The new framework is based on the framework proposed by Sen & Hemachandra in [1]. However, in their paper they mentioned that their framework is only tested on the correctness and not yet on the completeness of requirements. Therefore, more research is required to test the new framework on the completeness of the requirements.
- The cognitive tool suggested by Carod and Chechic in [2] is only tested in a controlled environment. In order to adapt this tool, it should also be tested in practice. Therefore more research on this is required.

Preface

Seven months ago I started this research project in Chubb Insurance Group Australia to complete my study Business and IT at the University of Twente. I was very interested in doing my research aboard to improve my personal skills and to face the challenge to step out of my comfort zone by going alone. Chubb Insurance Group Australia gave me the opportunity to do a research project in their office in Sydney. This research project is about the requirements engineering in market-facing technology projects that involves multiple stakeholders and is described in this thesis. In this project, a literature study is done to find the best elements in the field of science that can be used for market-facing projects. An exploratory study is done to understand the current situation of how the requirements engineering is done in market-facing projects. Recommendations are drawn and an improved framework is proposed for the requirements engineering based on the findings from both studies. Finally, suggestions are made to evaluate the proposed framework.

During this research project I learnt a lot about the insurance business in the way how they conduct business and their terminology. Besides this, I also gained knowledge from practice and the field of science on requirements engineering, project management, conducting case studies and writing this thesis.

However, without the help of other people I would probably not be able to accomplish this thesis. Therefore I would like to thank everyone who supported me in conducting this research. My special thanks go out to my supervisors at Chubb and at the University of Twente; Geoffrey Davitt, my supervisor at Chubb, who was always willing to help me and gave me the opportunity to do my research project within Chubb Australia. George White, my second supervisor at Chubb, for giving me advice when I was in doubt and his experience as a good project manager. Hans Moonen, my supervisor at the University of Twente, who is always willing to help me, and for his clear feedback and in arranging my internship in Chubb. Maya Deneva, my second supervisor at the University of Twente, for pointing out the research directions when I was in doubt and for giving me clear, structured and straightforward feedback.

Further on, I would like to thank all the interviewees for helping me by participating in my research, all my friends and the all the people that I met in Australia. All of them really helped me in motivating me to finish this thesis. Last but not least, I would like to I thank my family who always supported me in my decisions and are always willing to help me.

Martin Wesselink
Enschede, 03-10-2011

Contents

1. Introduction

Businesses are continuously trying to optimize their business processes in order to gain more benefits. One of the ways to optimize their processes is to automate them by IT. According to [8], the IT quality influences the organisation. The higher the quality of systems, information and service, the better the business will perform. Therefore, businesses are constantly improving existing systems and developing new systems. In order to improve existing systems or to develop new systems, requirements need to be gathered, negotiated and validated.

Currently, the Chubb insurance group company is transubstantiating this and want to improve some of their business processes by automating parts of their communication channels between them and their brokers. Therefore, they are developing a market-facing technology (MFT) system called iClose that will automate a range of business processes and will provide a complete solution to address all aspects of the broker-insurer relationship. At the moment of writing, Chubb realized a beta version of iClose and this beta version needs to be modified against business requirements that need to be gathered and validated. However, due to the fact that this system involves multiple stakeholders from different organizations, Chubb want to know how the requirements can be gathered, validated and negotiated most properly. Therefore, the main goal of this research is to determine how Chubb can improve the gathering, validation and negotiation of the requirements in a complex multi-stakeholder system project.

1.1 Chubb insurance company

Chubb insurance company is an insurance company that was formed in 1967 and is listed in the New York Stock Exchange since 1984. Originally they are an American company who were specialists in marine underwriting business in the seaport district of New York City. Nowadays their head office is located in Warren, New Jersey, United States and it is one of the largest publicly traded insurance organisations in the world.

Chubb provides insurance in all kinds of areas to business and personal customers all over the world. Currently, Chubb has more than 10,000 employees and over 100 branches in North America, Europe, South America and the Pacific Rim.

The organisation is divided into two areas, namely; production departments and service departments. The production departments consist of the following departments;
Chubb Commercial Insurance, Chubb specialty Insurance, Accident and health and Personal lines.
The service departments consist of; Administration, Accounts, Claims, Operations Services Division, Human resources, Information Technology, Marketing, Loss Control Services.

In Australia, Chubb is located in Sydney, Melbourne, Perth and Brisbane and offers commercial insurance products, accident and health insurance products, construction and commercial surety bonds. One of the commercial insurance products is the ForeFront package. This package consists of multiple policies and will be digitalized through the iClose system.

1.2 Market-facing Technology and the iClose project

According to the dictionary of marketing, market-facing is the point of contact between the supplier and their customers [9]. A market-facing enterprise, is an organisation that is sensitive to the needs of its markets and customers [9] and therefore it adapts itself to these needs [10]. In terms of the project, market-facing points are between Chubb, her brokers and customers. Chubb as a market-facing enterprise using the iClose project, want to adapt to the needs of the Small and Medium Enterprises by;

- Addressing the gaps of the current manual approach to Small and Medium Enterprises.
  Currently, quotes and policy for small and medium enterprises are manually produced, which takes too much time and effort. One of the objectives of the iClose system is to digitalize these processes in order to reduce the time and effort to quote and bind. In this way, Chubb can offer Small and Medium Enterprises more cost effective and less time consuming quotes and policies.
- Increasing current and new distribution channels.
  With iClose system, brokers will be able to request quotes and policies digitally and as mentioned before, through the digitalization it will reduce time and costs in the current distribution channels.
- Providing free underwriting and support resources.
  The iClose system will automatically quote, rate and bind policies. This means that underwriters do not directly have to be involved in the process and therefore brokers have the ability to quote, rate and bind whenever they want.
- Improving distribution management information.
  More data will be captured in the iClose system than in the manual process. Through this, the management would be able to gain more statistical information about the quote, rate and bind processes of Small and Medium Enterprises.

By means of these strategic objectives, and as already mentioned, Chubb wants to digitalize a range of business processes by the iClose project to provide a complete solution that addresses all aspects of the broker-insurer relationship. iClose will contain a series of modules and interfaces that can be implemented individually or in any combination to fill the gaps in the current market-facing situation. It will contain a flexible architecture to expand it with other modules as the need arises. The current addressed functionality is represented in Figure 1.

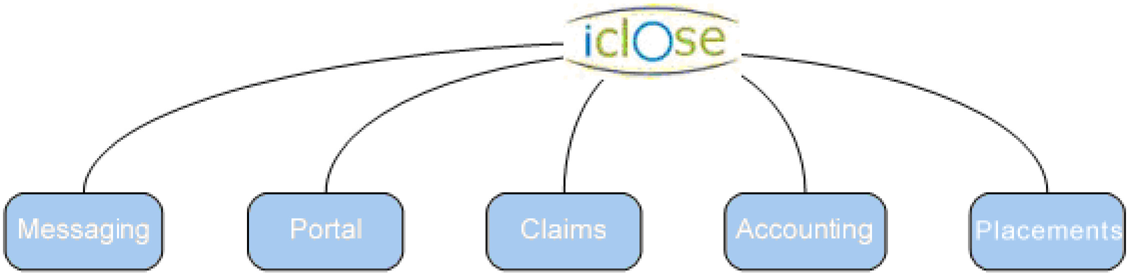


Figure 1: Functionality in the iClose system

iClose Messaging

iClose Messaging is a facility that transfers data between parties in a reliable and secure environment. All the modules use iClose Messaging as the security and transport layer. Features that are included in the iClose Messaging system are;

- Support for four different security profiles
- Synchronous and asynchronous data exchange
- Support attachments
- Message receipt acknowledgement

Parties that use iClose require an iClose Messaging Server to communicate with each other.

iClose Webportal
The iClose Webportal is a complete framework for developing, rating and deploying e-commerce insurance products. It empowers insurers, giving the ability to design, configure and deploy new revised insurance products quickly and cost effectively using the following utilities:

- iDesign - used to build online insurance projects
- eRate – a rating engine. For example to calculate premiums,
- iWrite – used to create documentation that will be returned to the broker,
- iRefer – supports referral for transactions that cannot be rated.

iClose Claims
iClose Claims automates the process of claim handling in the broker/client relationship. With iClose Claims, claims are initially registered through facilities in the broking system and synchronized with the insurer's claim system. This reduces duplication, ensures data in both systems is up to date, provides brokers with a more comprehensive claims profile for their clients and facilitates claims analysis and reporting. iClose claims can be implemented independently and linked to other processing facilities to provide an end to end solution.

iClose Accounting
The accounting functions between the broker and insurers will be automated due to the iClose accounting module. The first accounting process that will be automated is the settlement (payment and allocating money to the outstanding premium) process, where iClose accounting automatically produces settlements files and forwards this to the Underwriter's* system. When a broker pays the underwriter by Electronic Funds Transfer, it triggers the upload of the settlement file into the underwriter's system and matches the policies with the payments.

iClose Placements
The iClose Placements module is used to simplify the quote and bind phases within the process of creating / renewing policies (attached in Appendix A), where a quote is the commercial offer and the bind is the agreement before the policy has been officially issued. This module provides the broker with the ability to initiate a quote request, to submit the required risk details to multiple insurers and to monitor and manage quote responses. The module provides insurers with multiple options for responding to the quote request and to convert the quotes to policies. The process of iClose placements is depicted in the Figure 2.

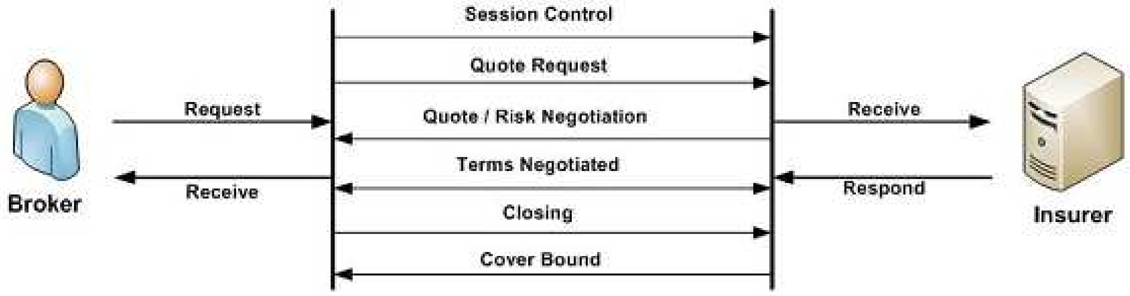


Figure 2: The process of iClose placements.

Overall iClose process
The overall process of iClose product is depicted below in Figure 3. In this diagram the broker access the system and request a quote of an insurance package. The insurers provide the broker a quote and based on the best quote the broker request the chosen insurer to bind the quote. The insurer then bind the quote (policy) and create an invoice for the broker. All steps in the diagram are marked with a number and each number is explained below;

1) Broker accesses the iClose placements via broking system.
2) Broker initiates a quote request within iClose Placements.
3) The broker chooses the insurers to request a quotation.
4) Quote submitted to the insurer via iClose messaging. Insurer receives email advising quotation request received.

* A professional that evaluates the risk of insuring a particular person or asset and uses that information to set premium pricing of insurance policies.

5) Insurer accesses quotation request (multiple channels).
6) Negotiations between broker and insurer.
7) Insurer offers terms via iClose Placements, Broker receives email advising terms have been provided.
8) Broker accesses iClose to advise whether cover is required or quotation lost. A request to bind cover is then sent to Insurer and Insurer receives email advising cover request.
9) Insurer then confirms cover bound via iClose Placements, Broker receives email advising cover bound, then uses broker system to upgrade quote to a policy with terms and policy coverage summary.

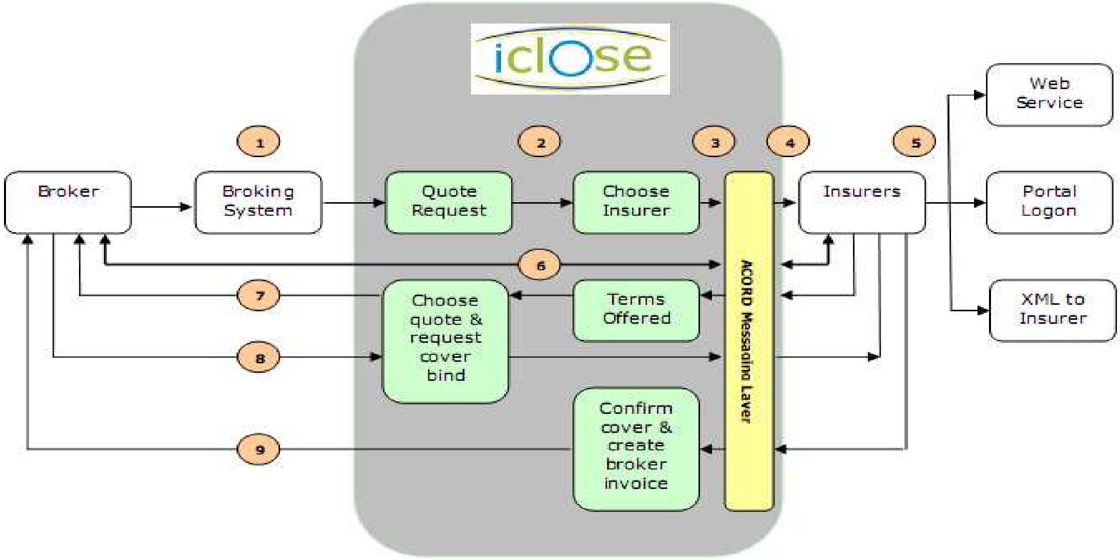


Figure 3: Overall process of the iClose system.

However, as mentioned before in the introduction, the iClose project is still under development and the first product that will be digitalized through the iClose project is the ForeFront insurance package. The reason that this package is chosen is because it has the lowest revenue when underwriting it manually and requires the least effort in digitalizing it because of its standard liability products.

1.3 Forefront Portfolio package

The forefront portfolio package is an insurance package that is designed for private owned companies that cope with many of the most dangerous threats to their financial well-being. For example the risk in the relationships dealing with;

- Employees
- Vendors
- Investors
- Customers
- Competitors
- Government Agencies
- Suppliers
- Creditors

The forefront Portfolio contains eight coverage sections that formulate a comprehensive insurance solution designed to be flexible, able to be tailored to the customers' needs. These eight coverage sections are described in Table 1.

Table 1: Insurance coverage sections of Forefront package

Forefront Portfolio Coverage Section:	The Risk:
Employment practice Liability Insurance	Employees and former employees can sue a firm, its board members and its officers for discrimination, harassment and other illegal employment practices.
Director and Officers Liability Insurance	Investors, customers, clients, government regulators, and Insurance competitors can sue a firm's board members and officers over their actions or decisions.
Trustees Liability Insurance	Retirees, former employees, and employees can sue the firm and its plan fiduciaries for alleged mismanagement in administering benefits.
Crime Insurance	A trusted employee can embezzle funds, steal inventory, or commit fraud over a long period of time.
Miscellaneous Professional Liability Insurance	Customers can sue a firm for alleged errors and omissions committed during the delivery of a professional service.
Statutory Liability	A regulatory/Government body may impose a fine upon the company for breach of their statutory requirements
Kidnap/Ransom and Extortion Insurance	An employee can be kidnapped while travelling overseas, or a criminal can attempt extortion against the firm by threatening its employees or products
Internet Liability Insurance	The company can be sued for copyright infringement or defamation over content it posts on its website.

1.4 Problem description

As already mentioned in Section 1.0 and 1.2, Chubb is developing a system that is called iClose. With this system they want to digitalize a range of business processes and provide a complete solution that addresses all aspects of the broker-insurer relationship. The first product that Chubb wants to digitalize is the Forefront insurance packages. In order to do that, this package has to be analyzed and the demands and restrictions from the insurance and brokers have to be understood completely. However, in previous situations, most of the system developments by Chubb were based on old existing (legacy) systems that never considered market-facing technologies. The traditional method that is used in these cases is the System Development Life Cycle (SDLC) where the requirements are gathered from existing systems through a pre-defined framework. After the requirements engineering, the new system was developed, tested and delivered. Due to that, the iClose system is a completely new system that involves a new way of doing business (market-facing), the current SDLC requirements engineering framework does not fit. In this project, there are no existing systems that can be analyzed for the complete development. Requirements have to be gathered, negotiated, validated and prioritized through multiple stakeholders which is a complex process because all stakeholders have different perspectives, requirements and priorities. Therefore, Chubb want to have a requirements engineering framework that can be used in market-facing projects like iClose.

According to [11], Requirements Engineering (RE) is the process of discovering the degree to which the software system meets the purpose for which it was intended. This is done by identifying stakeholders and their needs, documenting these needs in a form that is suitable for analysis, communication and implementation. However, this process encounters a number of difficulties. There are multiple distributed stakeholders that have their own views and goals. These views and goals may vary and conflict with other stakeholders. Stakeholders can have problems making these goals explicit which can constrain their satisfaction due to multiple factors outside their control [11]. Over the last decades several researchers have developed various approaches in order to address this gap, such as Goal-Oriented requirement engineering [12-15], Agile based visualization techniques [16, 17], Agent-Oriented requirements engineering [18] and Rapid Application Design [19]. According to [20] most of these techniques run into trouble when practitioners get lost in the complexity of the methodologies, or analysts lose sight of the goals by becoming fixated on generating deliverables, or the techniques become unmanageable when the number of stakeholders is large. Therefore, currently and over the last several decades researchers have been developing techniques to negotiate and validate the requirements among the stakeholders. In [21], Balfagih & Hassan present a quality model that classifies non-functional characteristics based on multiple stakeholders' requirements. Arnold et al., propose in [22] a framework that supports the modelling and automated validation of requirements based on a model driven approach. However, this framework is only applicable in the .NET platform by garbage-collected programming language such as C#. There are many other techniques available in academic literature among the validation and negotiation of requirements. In order to create a requirements engineering framework for Chubb that considers market-facing technology and addresses the issue to gather, negotiate, validate and prioritize requirements among multiple stakeholders, the following question needs to be answered;

How to gather, negotiate, validate and prioritize the requirements of market-facing projects from a complex multi-stakeholder perspective?

To answer this main research question, I formulated several sub questions. These sub questions are;

1. What methods/approaches for similar projects and companies are suggested in scientific literature?

2. How does Chubb currently gather, negotiate, validate and prioritize the requirements of market-facing projects?

3. What can be improved in the way Chubb gathers, negotiates, validates and prioritizes requirements reviewing the current practice and comparing against scientific literature?

4. What would be a better way for Chubb to carry out requirements engineering for market-facing projects?

5. How to evaluate the proposed framework/methods?

First, these sub questions need to be answered in order to answer the main question. A literature review needs to be done to find the best elements of how to gather, validate and negotiate the requirements for market-facing systems. After this, the current situation within Chubb about how requirements engineering is done in market-facing projects need to be investigated through an exploratory case study. Based on the findings of this case study and the literature study, recommendations and a framework will be created on the gathering, negotiation, prioritization and validation of requirements that can be used to analyze other insurance packages for the iClose project and other market-facing projects. Finally, limitations will be discussed and suggestions will be made to test and evaluate the new framework.

1.4.1 Practical significance in general

According to Evens [23], Mastering End-to-End Business Processes and Customer Engagement are in the top 10 concerns of Chief Information Officers (CIO). These concerns all involve requirements engineering. Mastering End-to-End business processes is about understanding and gathering the right requirements that are needed to enhance these End-to-End business processes. Customer Engagement also effects requirement engineering, by listening to customers, system and Business Analysts will understand what needs to be changed to get optimal customer satisfaction. According to a survey conducted by Duvall in 2009 [24] the following concerns are all within the Top 10 concerns of CIO's;
(i) IT and business alignment
(ii) Business agility and speed to market
(iii) Business process re-engineering
(iv) IT reliability and efficiency.
All these concerns are affected by requirements engineering. Improving the process of requirements engineering results in a improved Business and IT alignment, business agility and speed to market, Business process re-engineering and in an improved IT reliability and efficiency. Also Luftman and Kempaiah stated in a survey conducted [25] in 2007 that one of the Top 5 concerns is to improve IT quality which is also effected by requirements engineering. Therefore, from a practical point of view it can be concluded that proper requirements engineering is very crucial. Gartner recognized this and

published a recent article [26] about Wiki systems that can be used as a collaboration medium to gather, negotiate and validate the requirements of multiple stakeholders. Besides this, Gartner also [27] mentioned that Project Managers should collaborate on requirements with stakeholders, including executive sponsors, to determine what the primary business driver of the project is (schedule, budget or functionality). Based on these recent publications it can be concluded that they are still looking for improvements on requirements engineering in the field of practise.

1.4.2 Practical significance for Chubb

A requirement engineering framework special for market-facing projects will reduce time, effort and complexity in the requirement engineering process. By reducing the complexity, Chubb would be able to identify issues in the requirement engineering process earlier and better. In the end, all these aspects can decrease the costs in de overall development of market-facing projects. Besides this, having a formalized approach of requirements engineering in market-facing projects also enriches the insight of the market-facing project. Through that the stakeholders get acquainted with the standardized approach, they will be able to understand the status of the project.

1.4.3 Theoretical significance

As can be observed from the introduction of this section, the gathering, validation and negotiation of requirements is also very popular in the field of research. In previous studies lots of methods, frameworks and approaches are described [2, 3, 22, 28-30]. Some of these papers mention that aspects such as cognitive profile mapping [2, 28, 29], tools like Unified Modelling Language (UML) and Object Constraint Language (OCL) [22], decision logging and systematic negotiations [13, 27] will contribute to the requirements engineering. However, as [31] already mentioned, empirical testing is needed to validate these methods, frameworks and approaches. This research can contribute to the discipline of Requirements Engineering by initiating existing models in practice to provide empirical evidence.

1.5 Structure and approach

This research is divided into four parts which are used incrementally. The first chapter is the research approach. This part consists of an orientation study of the research topic to gain insights on the issues that need to be addressed, an overview of the company and project, a problem description, the research objectives, the research questions and research methodology are formulated and described. In the second chapter, the first sub question of the research is addressed by an in-dept scientific literature research to gain best elements mentioned in the field of academics. In the third chapter of this thesis is explained how an exploratory case study is conducted within Chubb and findings from this study are described to answer the second sub question about how Chubb is currently gather, negotiate, validate and prioritize the requirements of market-facing projects. In the fourth chapter;

(i) Recommendations are given based on the findings from the previous studies to answer the research question: “What can be improved in the way Chubb gathers, negotiates, validates and prioritizes requirements reviewing the current practice and comparing against scientific literature?”

(ii) A new framework for requirement engineering in market-facing projects is presented drawn from the recommendations and previous studies in order to answer the research questions: “What would be a better way for Chubb to carry out requirements engineering for market-facing projects?”

(iii) Suggestions are given to evaluate the new framework to answer the last research question: “How to evaluate the proposed framework/methods?”

In the final chapter, conclusions are drawn and limitations and implications for further research are given based on the findings of this research. A schematic structure of the chapters is depicted in Figure 4.

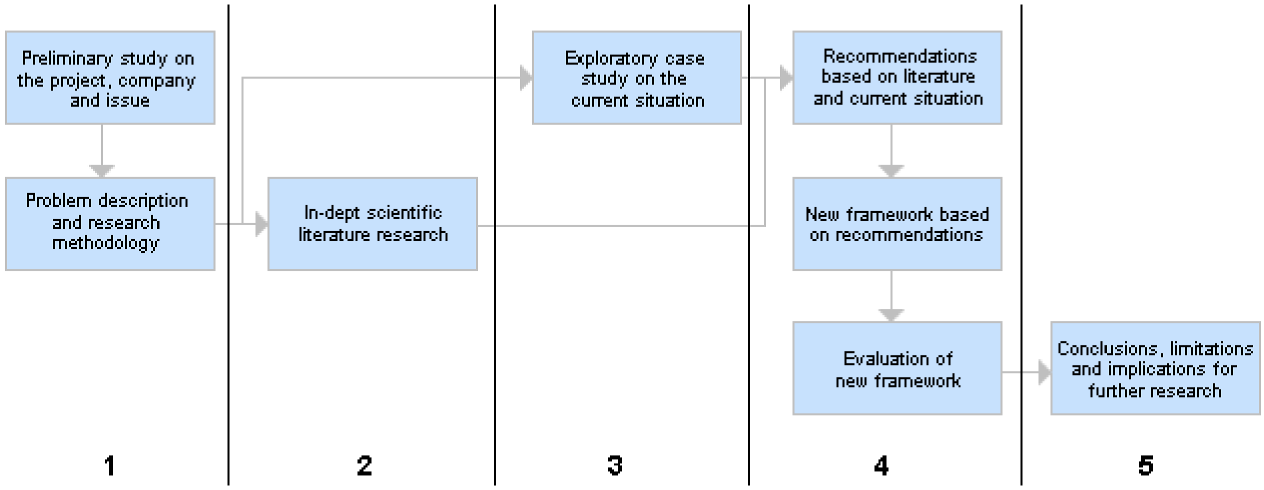

Preliminary study on the project, company and issue
Problem description and research methodology
In-dept scientific literature research
Exploratory case study on the current situation
Recommendations based on literature and current situation
New framework based on recommendations
Evaluation of new framework
Conclusions, limitations and implications for further research
1
2
3
4
5

Figure 4: Schematic framework of research structure.

2. Literature study

In this chapter, the elements found in academic literature are described. This chapter is divided into multiple sections, starting with a first section where the literature search methodology is described. Based on that, in the second section, the requirements engineering techniques found in the literature are described. The following three sections consider the four aspects of the main research question, namely; how to gather requirements in a multi stakeholder project, the negotiation with stakeholders about the requirements in a multi stakeholder project and the validation and prioritization of requirements in a multi stakeholder project. The last section presents the conclusion of our findings and provides an answer to the first research question.

2.1 Literature search methodology

The central research question that is addressed by this literature research is the first research question mentioned in the previous chapter, namely: "What methods/approaches for similar projects and companies are suggested in scientific literature?" In order to answer this question, sub questions are formulated. Answering these sub questions will finally answer the concerned research question of this chapter with the focus on the central research question on this thesis. These sub questions are;

- What methods/approaches are used in requirements engineering for similar projects?
- What techniques/methods are used to gather requirements from multiple stakeholders?
- What techniques/methods are used to negotiate requirements among the stakeholders?
- What techniques/methods are used to validate and prioritize the requirements from multiple stakeholders?

All these questions will be addressed through a systematic literature review according to the guidelines of [32]. Relevant papers of the first question will be identified by requirement engineering approach/methods. For the second question, relevant papers will be identified by requirements gathering techniques and methods. For the third question, relevant papers will be identified by requirements negotiation techniques and methods. Relevant papers for the forth question will be identified by requirements prioritization and validation techniques and methods. In the conclusion, the central research question of this chapter will be answered based on the findings from the sub questions mentioned above.

2.1.1 Search strategy

The intention of a systematic literature review is to ensure that a relative consensus of relevant literature is collected [32]. To do this, multiple scientific search citation databases are used that consult multiple scientific libraries and databases. The search databases that were chosen are Scopus and Web of Science. For searching through these databases, the following search strings were used;

For the first question the search strings are:

(i) E-insurance
(ii) insurance system requirements
(iii) requirements multi-stakeholders
(iv) customer facing internet system requirements
(v) market-facing requirements
(vi) requirements engineering insurance.

For the second question the search strings are;

(i) requirements gathering
(ii) requirements multi-stakeholders
(iii) negotiation in the requirements elicitation and analysis process.

The search strings for the third question are;

(i) negotiate requirements stakeholders
(ii) requirements negotiation
(iii) negotiation in the requirements elicitation and analysis process.

For the final sub question the search strings are used;

(i) validating requirements stakeholders
(ii) requirements validation prioritization
(iii) requirements prioritization.

The composition of these search strings are based on a small manual study on requirements engineering. From this manual study, search strings were elicited and iteratively used to get the composition that provided the most relevant results. During this process, varieties of combinations of these strings were used. The reason that Scopus and Web of Science were chosen, is because Scopus addresses multiple other well know databases, like IEEE Xplore, ACM Digital library and Science Direct, and Web of Science is chosen due to the fact that it coverage's the most Top 10 IS journals and Top 25 IS journals [33]. Further on, the search included scientific conference proceedings, journals, magazines and workshop proceedings and was conducted by searching in the title, abstract and keywords. Based on the findings of this search, two other search strategies were conducted, namely a back- and forward search. In the backward search, citations of previously found papers are reviewed, which resulted in new relevant papers. These found papers were screened on their title, abstract and keywords with the same inclusion and exclusion criteria used in the previous search. In the forward search articles were identified that cited previously found papers. These articles were also screen on their title, abstract and keywords with the inclusion and exclusion criteria that was used in the previous search. Further on, for both search strategies the search engine Scopus and Web of Science were used because these engines cover the most conference and journal papers.

2.1.2 Selection of the studies

Figure 5 depicts the selection of the papers based on the search strategy mentioned in the previous section.

The search was performed between March 3 and March 10, 2011 and resulted in 3287 potentially relevant studies, 1234 from the search engine Scopus and 1524 from the search engine Web of Science. These studies were screened on the title and abstract-based inclusion criteria.

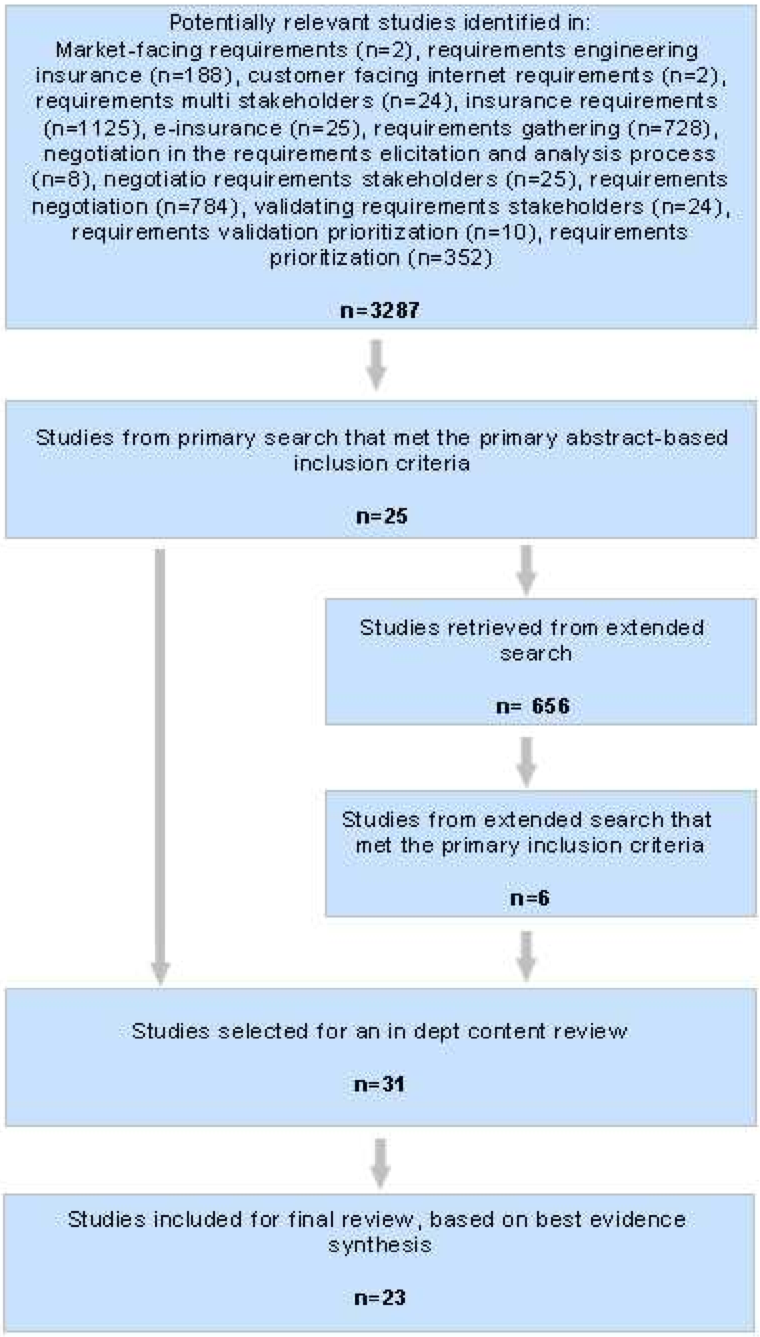

Potentially relevant studies identified in:
Market-facing requirements (n=2), requirements engineering insurance (n=188), customer facing internet requirements (n=2), requirements multi stakeholders (n=24), insurance requirements (n=1125), e-insurance (n=25), requirements gathering (n=728), negotiation in the requirements elicitation and analysis process (n=8), negotiatio requirements stakeholders (n=25), requirements negotiation (n=784), validating requirements stakeholders (n=24), requirements validation prioritization (n=10), requirements prioritization (n=352)
n=3287
Studies from primary search that met the primary abstract-based inclusion criteria
n=25
Studies retrieved from extended search
n= 656
Studies from extended search that met the primary inclusion criteria
n=6
Studies selected for an in dept content review
n=31
Studies included for final review, based on best evidence synthesis
n=23

Figure 5: Selection of scientific studies

From this screening, there were 25 studies found that were relevant. These 25 studies were further investigated by doing a forward and backward search on the citations in the studies. During this search another 6 studies were found that met the primary inclusion criteria. After that, these 31 articles were reviewed more in dept on the content based on the inclusion and exclusion criteria. This resulted in a total of 23 relevant studies. These studies were selected for a final review and analysis of this literature study. These studies are;

- From Scopus:
  [11] [20] [22] [2] [28] [3] [34] [35] [36] [37] [38] [34] [35]

- From Web of Science:
  [29] [39] [40] [41] [42] [43] [44]

- Extended search:
  [45] [4] [46]

During the final review, concepts that are consistent will be identified in the found studies and compared with each other. Besides this, studies were checked if they are duplicated in the found studies or if they used the same data.

2.1.3 Inclusion and exclusion criteria

As mentioned before the found papers were analyzed based on inclusion and exclusion criteria. The inclusion criteria are the criteria that the papers should meet and the exclusion criteria are the criteria that the papers should not meet. If the paper met one of the exclusion criteria, it will be excluded from the research. If the paper met one of the inclusion criteria and did not met one the exclusion criteria it will be included in the research. The inclusion criteria of this literature study are;

I1 – The paper discusses a RE approach for multiple stakeholders.

I2 – The paper discusses aspects of requirement negotiation, validation and prioritization.

I3 – The paper mentioned aspects on requirements gathering methods and techniques.

I4 – The paper discusses RE aspects and approaches for market-facing and insurance projects.

I5 – The paper is original, no duplications in the found literature and use unique data sets.

The exclusion criteria of the literature study are;
E1 – The paper discusses market-facing or insurance project aspects that are not specifically applicable
to RE.

E2 – The paper discusses RE that is not applicable in market-facing or multi-stakeholder projects.

E3 – If papers have the same results. One of them will be excluded.

E4 – The paper only reviews other approaches.

In the last review, 8 papers were excluded. Some of them did not meet the inclusion criteria and others meet the exclusion criteria. The papers that were excluded from the study are listed in Appendix B. Two papers did not meet the inclusion criteria and met the exclusion criteria E1. Four papers were excluded because they meet the exclusion criteria E2. There were two duplicated papers, where one was excluded due to exclusion criteria E3. One paper gave an overview of known methods in scientific literature and was excluded due to E4.

2.1.4 Concept matrix

A literature review should be concept-centric. Therefore, concepts determine the organizing framework of a review [32]. As for the literature study of this thesis, concepts were identified in the last review of the found papers, where the full text was retrieved and analyzed. These concepts are summarized in the concept matrix presented in Table 3. The concepts are organized by the gathering of requirements, the negotiation of requirements and the validation and prioritization of requirements. In the table each concept is given a number and described in Table 2.

Table 2: Description of concepts

Concept description	
1 = Requirements approaches	8 = Iterative
2 = Diverse stakeholders & Pairing staff	9 = Emotions
3 = Interviewing	10 = Viewpoint stakeholders
4 = Logging	11 = Cognitive selection
5 = Scenarios & Stakeholder positions	12 = Automated validation mechanisms
6 = Collaboration system & Logging	13 = Cost Benefit agile prioritization
7 = Consistency	14 = Cumulative Voting

Table 3: Concept matrix

Articles	Concepts													
		Gathering				Negotiation				Validation & prioritization				
	1	2	3	4	5	6	7	8	9	10	11	12	13	14
Nuseibeh and Easterbrook [11]					x					x			x	
Fruhling et al. [20]	x	x			x									
Arnold et al. [22]												x		
Carod and Cechich [2]			x								x			
Kof [28]	x		x									x		
Ahmad [3]	x		x		x			x		x				
Cafer and Misra [29]					x			x			x			
Nejati and Chechik [34]							x	x						
Liu et al. [35]	x									x				
Peng et al. [36]	x			x		x						x		
Laurent et al. [45]		x					x					x		
Gaur et al. [37]					x									
Jiang and Yang [38]							x				x			
Molina [39]	x						x				x			
Wu et al. [40]	x					x		x						
Port and Bui [41]	x	x						x		x				
Kelly [42]	x			x	x			x						
Gorschek and Wolin [43]				x				x						
Racheva et al. [4]													x	
Colomo-Palacios et al. [44]									x					
Chatzipetrou et al. [47]														x
Sen and Hemachandran [35]	x													
Ramos et al. [46]									x					

Concept 1 addresses the approaches in requirements engineering. Concepts 2 to 5 are mainly used in the gathering process of requirements. The concepts 6 till 9 are categorized in the requirements negotiations process and the concepts 10 to 14 are categorized in the requirements validation and prioritization process. Although these concepts are divided into categories, they are overlapping to a certain extent. For example iteration is not only considered during the requirements negotiation, but also during the requirements gathering, validation and prioritization. To begin with, the first concept, requirements engineering approaches, is explained and described in the next section.

2.2 Requirements engineering approaches

Many different approaches are proposed in the found literature. However, most of these approaches have similarities with each other. The approaches that differs the most from each other are explained in this section. Starting first with the goal oriented approach. In the goal oriented approach, the focus is on the goals of the stakeholders. According to [48] a goal in requirements engineering is; “an objective that the system under consideration should achieve”. In general, in the goal oriented approach, the first step is to identify the goals of the stakeholders where different stakeholders have different goals which may conflict with each other. There are several approaches proposed to analyse which goals are important and to determine what the main goal of the system is and how to solve goal conflicts [28] [39] [35].

Some other approaches are based on collaboration engineering [36] [20] [40]. According to [49] collaboration engineering is an approach to design re-usable collaboration processes and technologies that are meant to produce predictable success among practitioners of recurring mission-critical collaborative tasks [49]. According to [20], the purpose of collaboration engineering is to quickly and repeatedly create an environment that promotes creativity and communication between participants and consists of the processes described in Table 4.

Table 4: Processes in collaboration engineering [20]

Pattern	**Description**
Generate	Move from having fewer concepts to having more concepts.
Clarify	Move from less to more shared meaning for the concepts under consideration.
Reduce	Move from have many concepts to a focus on fewer concepts deemed worthy of further attention.
Organize	Move from less to more understanding of the utility of priority of concepts toward goal attainment.
Build consensus	Move from having more disagreement to having less disagreement on course of action.

Relating this to requirements engineering, the first process “generate” is to collaboratively generate as many requirements as possible by the stakeholders (gather requirements). The second and third process “clarify” & “reduce” relates to the negotiation of requirements where stakeholders have to discuss which requirements are important and which are not. The final processes “organize and build consensus” relate to the validation and prioritization of the requirements where stakeholders have to validate and prioritize the requirements in order to get a shared understanding of the system. Studies that consider Wiki systems as a collaboration tool to gather, negotiate, validate and prioritize requirements use this kind of approach.

The approaches described above are based on plan based strategies. These strategies have an overall implementation plan. However, in reality, it seems that requirements can change over time which can change the whole implementation plan. Therefore another strategy is developed; this strategy is called Agile where approaches use small iterations where stakeholders have to work intensively with each other to review each other to achieve the highest satisfaction. Found studies that consider this strategy are [41] [42] [3]. However, in this strategy there is a chance that important requirements may go unrecognized or recognized too late and get not implemented [41]. For example, Sen and Hemachandran tried to address this aspect and proposed in [35] an agile technique that extracts goals from stakeholders. Where they decompose high level goals into lower level or sub goals. They validated their methodology through a case study and found that the correctness of the elicited goals was obtained by the approval of the stakeholders, but the issue of completeness could not be verified.

Reflecting this back to the first sub question, “What methods/approaches are used in requirements engineering for similar projects?” some approaches that are used are goal oriented approaches, collaboration engineering and several other agile approaches. However, all these approaches have their pros and cons. Therefore the next sections describe the elements found in the literature that are important to take into consideration, starting with the gathering of requirements.

2.3 The gathering of requirements in multi stakeholder projects

In the found literature, multiple elements can be used to improve the requirements gathering process. These elements are addressed below, grounded by the related literature.

2.3.1 Diverse group of stakeholders and pairing

As already mentioned in the introduction, a lot of attention has been paid to examine the requirements. As for the requirements gathering process, [20] [41] mentioned that a group of diverse stakeholders is critical. Fruhling et al., evaluated in [20] the collaboration engineering process that would facilitate system requirements validation and elicit new requirements from multiple stakeholders who had different needs. They conducted two sessions where they applied several collaboration techniques. In the second session, Fruhling et al. made more diverse groups of stakeholders by assigning fixed seats to diminish influencing group dynamics such as ranks, personality and organizational background. Each group of stakeholders had technical staff and functional users. As a result, this second session generated twice as many requirements per participant than the first session. Also Port and Bui mentioned, in [41], that the group of stakeholders must be diverse. They suggest that each group of stakeholders must include at least one customer representative and one development manager. In which the customer representative can be seen as the functional user and the development manager as technical staff. However, Laurent et al. criticise in [45] that the requirements generated by joint requirements gathering can lead to increasingly large volumes of stakeholders' requests. This can lead to a total project failure because the amount of requests cannot be realized in the available amount of time.

2.3.2 Interviewing

Besides the diverse group of stakeholders, another technique that can be used to gather proper requirements is interviewing [2] [28] [3]. Ahmad proposes a new negotiation spiral model in [3] that also considers requirements elicitation. According to Ahmad [3], requirements elicitation is the process to extract and identify the requirements systematically from a combination of human stakeholders. She mentioned that one of the techniques that can be used to identify requirements is interviewing. Also Carod and Cechich mentioned in [2] that interviewing should be used for requirements elicitation, especially in new domains. In [2] they did a case study where they map stakeholders to cognitive profiles. There findings showed that the elicitation techniques, like interviewing, should be carefully selected according to the cognitive skill of the stakeholder to improve the requirements engineering process. However, these are only results from a controlled experiment, which have their limitation, as they cannot be generalized to every situation. In [28], Kof suggests that it is necessary to know the rationale that lead to particular requirements in order to solve contradictions in the requirements. By rationale, he means the goal of a particular requirement that is suggested by a stakeholder. In order to identify goals from stakeholders, he mentioned that two key questions could be applied, namely: why and how [28]. An answer to a how-question gives a possible refinement of the goal. Moreover, an answer to a why-question identifies its superior goals. During interviewing, these techniques can be used to identify the goals of the requirements and stakeholders properly.

2.3.3 Logging

Another aspect that needs to be considered is logging. By the use of logging, stakeholders will better understand the requirements. In [43], Gorschek and Wohlin developed a model for market-driven requirements engineering that allows the placement of requirements on different levels and supports abstraction or a break down of the requirements to make them comparable with each other. One of the initial steps in this model, Gorschek and Wohlin mentioned [43] to create an overview of the requirements to the extent of being understood by the Product Manager. This overview of the requirements should at least per requirement contain; a description, the reason, benefit and rationale, the restrictions and risks and a title. By having this overview, the requirement engineer will be able to compare and validate these requirements. Also Peng et al., addresses in [36] the aspect of logging in the sense of a Wiki system. They recognize that in projects with a large scale of stakeholders, the focus of attention becomes a problem. Therefore, Peng et al. suggest in [36] the use of a Wiki system, which has the ability to offer distributed requirements elicitation and documentation. In a Wiki system, the behaviour of the stakeholders among the requirements can be monitored. Based on that, the requirements can be validated. I.e. every stakeholder can browse through the existing requirements. The longer a requirement version the more times it is browsed, indicates it is more stable [36]. However, Kelly proposes in [42] a framework that can be used to improve the facilitation of requirements elicitation. As part of this framework, she proposes the use of user stories documented on story cards, where stakeholders have to document their story from an user perspective. These stories are then used to identify a high-level plan of the project and to provoke an in dept discussion between developers and stakeholders in order to draw up proper requirements for the system under development.

2.3.4 Scenarios and templates

The user stories technique that Kelly mentioned in [42] can also been seen as a scenario technique, where every story card is a different scenario. The use of different scenarios can contribute to the requirements gathering process where the same aspect will be treated from multiple viewpoints, which will rise in more requirements. Nuseibeh and Easterbrook mentioned, in [11], that the only way to capture the system's purpose and the reason about whether a given design will meet this purpose is by describing the environment and expressing what the new system must achieve in that environment. Nuseibeh and Easterbrook suggest in [11] to do this by modelling stakeholders' goals and by using scenarios that illustrate how goals can be achieved. Gaur et al., propose in [37] a Multi-Person Decision-Making Model to negotiate and integrate the requirements of various stakeholders. In this model, they use user stories as a scenario technique to acquire requirements from the users. In order to retrieve these user stories, they made a prescribed template as follows; "As a <user type>, I want to <goal> So that <reason>". They same kind of technique is also used in [20] where Fruhling et al. used a template of how a requirement should be stated. This helped them to speed up the process of categorizing and validating the collected requirements. Another paper where templates are used to collect the information from users is [29]. Cafer and Misra proposed in [29] a cognitive requirement specification model that is based on the cognitive classification of customers. For the collection of the requirements, they proposed two different types of models based on the cognitive capabilities of the users. Each model contains a template that has to be used to gather the requirements from the users. Also Ahmad addresses the concept of scenarios in her paper [3]. Besides interviewing, she also mentioned the use of use-cases to gather requirements. A use case is a description of potential series of iterations between the system and its environment [50] that can be seen as a description of several scenarios in which the system interacts.

2.3.5 Conclusion

In summary, according to the found literature, the aspects and techniques that need to be considered to gather the requirements properly are; the use of a diverse group of stakeholders where each group at least should have one person from the development staff and one person that is actually going to use the system. Stakeholders should be interviewed in order to get a complete set of requirements. Questions should be made based on cognitive profiles, including why and how questions to refine goals and identify superior goals. The elicited requirements should be logged in a central system in order to compare and validate the requirements among the stakeholders. Scenarios should be used to gather multiple viewpoints from stakeholders and the environment, which improves the requirements elicitation process. Templates should be used to speed up the categorization and validation process.

2.4 The negotiation of requirements in multi stakeholder project

The negotiation process in requirements engineering has been extensively researched. Many papers have been published on this topic. However, due to the limitations of time, only those papers that are potentially interesting for the situation of Chubb are addressed in this research. Elements found in these papers are described below.

2.4.1 Collaboration system and logging

As already mentioned in the section above about the gathering of requirements in a multi stakeholder projects, Peng et al. suggest in [36] the use of a Wiki system to document and validate requirements. However, they suggest that the Wiki system is also sufficient for requirements negotiation. By the use of a Wiki system, stakeholders can modify the requirements by giving a mandatory motivation. Based on that, other stakeholders can mark the current requirements version according to their own preference that will indicate the degree of satisfaction of that specific requirement [36]. Once the requirement reaches the required degree of satisfaction, it will become mature and suitable for further development. Due to this technique, distributed stakeholders can negotiate with each other and all the discussions are recorded which could be convenient to know later on why certain decisions are made. Besides Peng et al. also Wu et al, suggested the use of Wiki system in [40]. In [40] Wu et al. tested 32 real-client, graduate level team projects using a wiki-based requirements negotiation support system called WikiWinWin. This system is based on the WinWin equilibrium theory of [51]. The WinWin equilibrium theory links win, conditions, issues options and agreements. It establishes a win-win equilibrium where all win conditions are covered by agreements and all issues are resolved by options [40]. The results of the test showed that; 1) better project outcomes were correlated with the frequency of use. The more the Wiki system was used, the better the requirements results. 2) The users found the Wiki system beneficial to the requirements engineering process. 3) The tool showed improvements on cost-effectiveness. However, the usability for non-experts is still challenging and it is not sure whether the frequency of use is the real cause of the improved results. It could be that these stakeholders had more expertise in general.

2.4.2 Consistency

Many software projects fail due to incomplete or incorrect requirements. In order to get the right requirements, information needs to be gathered from stakeholders. This information then needs to be interpreted, analyzed and validated. However, due to the fact that most customers could hardly articulate their requirements it can easily be misunderstood by other stakeholders [38]. Therefore, Jiang and Yang propose a model in [38] to elicit performance requirements from customers for financial information systems based on ontology. By using an ontology model they want to bridge the cap of inconsistency. The ontology describes the requirements in a formal and a unified language to avoid misunderstanding between different stakeholders [38]. Luarent et al., recognize this in [45] and stated in their conclusion that consistently using a project glossary throughout the software development lifecycle can also significantly improve performance. In [45] they propose an approach for automating a significant part of the requirements prioritization process due to a non-functional requirement classifier. This classifier is a data-mining tool that classifies a broad variety of non function requirements types related to attributes such as security, performance or usability. In [39], Molina et al. propose a measurable requirements metamodel that offers support to the elicitation of measurable requirements. This measurable requirements metamodel provides users with the abstract syntax for the specification of measurable requirements. By using an abstract syntax, Molina et al., addresses the gap of inconsistency due to abstraction that describes the way how requirements should be formulated. This would benefit the negotiations of the requirements among the stakeholders. Nejati and Chechik present in their work [34] a formal framework for merging and conflict resolution in the negotiation process. They stated that different stakeholders use different requirement models. To solve this conflict, these models need to be merged. This merging can only be done when these models are consistent. According to Nejati and Chechik stakeholders have to negotiate in order to solve these inconsistencies. Once these inconsistencies are resolved, the stakeholders can use the framework presented in [34] to merge their requirement models and to align their requirements with each other.

2.4.3 Iterative

According to many studies [34] [3] [29] [30] [42] [41] [43] [40] the requirements negotiation process needs to be done iteratively in order to get the best requirements for the system under development. As mentioned before, in [3] Ahmad introduces a spiral model that consists of elements that support the requirements elicitation and analysis process. In the analysis process of this model, negotiations amongst the stakeholders have to occur in order to get an overall agreement on the requirements. According to Ahmad this is an iterative process and therefore her model allows renegotiation. She describes that the requirements negotiation process exists of; 1) Identify conflicts. 2) Develop alternatives solutions. 3) Elaborate solutions. 4) Judgement and trade-off. 5) Evaluate and analyse agreement. If stakeholders still have disagreements after “evaluate and analyse agreement”, then this has to go into another spiral beginning with the first step: Identify conflicts [3], see Figure 6. Wu et al., state in [40] that; “Stakeholder collaboration is the key success factor for requirements negotiation. During the negotiation the stakeholders have to collaboratively and incrementally found out what has to be build.” The process of collaboratively and incrementally negotiating the requirements is an iterative process where stakeholders have to share knowledge, understand each other and to resolve conflicts of interests [40]. These activities are similar to the negotiation activities described in [3]. As mentioned before in Section 6.1, in [42], Kelly proposes a framework to improve the facilitation of agile requirements elicitation. According to her, multiple diverse stakeholders have to continuously negotiate with each other to clarify requirements. This remains a difficult aspect in the requirements engineering and her framework tries to address this issue. The results of her study indicate that writing user stories achieved in an agile (iterative and incremental development) manner gives positive results on exploring and classifying requirements [42].

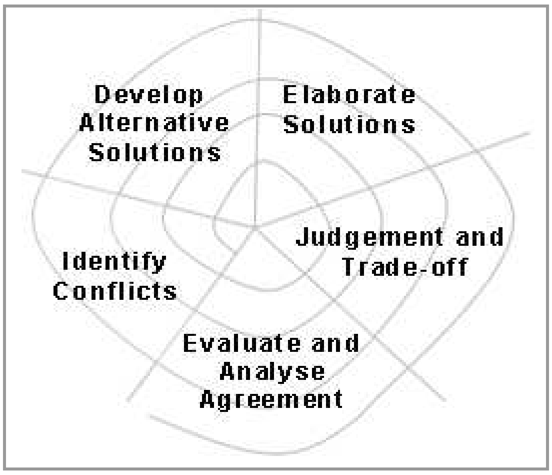


Figure 6: Requirements negotiation process explained in [3]

2.4.4 Emotions

An element in the requirements engineering that did not have much attention is the emotional factor of humans that are involved during the process [44]. The requirements engineering process is knowledge intensive and because of that it also is human intensive. Therefore, one of the key elements that play an important role during the negotiation and acceptance activities of the requirements is the emotional factor. According to Ramos and Barry [46], considering the emotional factor in requirements engineering will contribute to the transformation process that involves the use of a new system to users and in defining the requirements in ways that are beneficial for the stakeholders. Colomo-Palacios et al. [44] proposed in their paper a tool for analyzing and evaluating emotions in order to create a better understanding of the requirements and for the categorization of the requirements. The tool that they propose is the Affect Grid psychological tool that is created by Russel et al. [52]. During the requirements engineering process, stakeholders have to express the emotion that requirements raise for them. After each iteration, the requirements and emotions change and will be registered. The collected data is then encoded together with the requirements to ensure the pursued emotional traceability. Based on that, the requirement engineer will be able to know the stakeholder's emotions that involve understanding the stability and reliability of the definition of the requirements. By knowing this, it helps the requirements engineer or manager to create an environment that is capable of combating the effects of bad emotions.

2.4.5 Conclusion

In summary, the elements found on the negotiation of requirements in a multi stakeholder's project are; 1) the use of a collaboration system where all the decisions made are logged. This will help later on to analyze and understand why decisions are made, which reduces time and effort. 2) During the negotiations, definitions and requirements need to be consistent. This avoids misunderstanding between stakeholders and improves the performance of the negotiation process. 3) Iterative negotiations about the requirements, where stakeholders have to participate to get an overall agreement among all stakeholders. 4) Emotional factors should be considered to understand the stability and reliability of the definitions of the requirements.

2.5 The validation and prioritization of requirements in a multi stakeholder's project

In this section, the elements of validation and prioritization of requirements found in the literature are presented and discussed. The addressed papers are selected on the relevance of this study. The first element that is addressed is the viewpoint of the stakeholders.

2.5.1 Viewpoint of stakeholders (source validation)

Stakeholders have their own perception on the software development process. However, most of these perceptions have similar elements and relate to some extend to each other. This means that most stakeholders often have similar requirements but formulated it in different ways with different priorities. Therefore these requirements need to be validated and prioritized. Liu et al. presented and tested in [35] a Correlation-Based Priority Assessment framework, which prioritizes the requirements gathered from multiple stakeholders by incorporating their overlapping perspective on relationships of requirements. They found that by using this framework, software process requirements with a longer and stronger impact on other requirements from multiple perspectives receive higher priorities [35]. Also [3] takes the viewpoints of stakeholders into consideration in her spiral model. In the elaboration step, stakeholders have to discuss the solution alternatives where everyone can put in their perspective to promote a better understanding among all the stakeholders. Due to the discussion, the stakeholders are automatically addressing the next step: to validate the requirement by evaluating and analyzing them [3]. Nuseibeh and Easterbrook present in [11] an overview of the field of software systems requirements engineering. In their work they suggest that only modelling information flows and system states of the environment and expressing what the new system must achieve is not enough. According to Nuseibeh and Easterbrook the viewpoints in the sense of goals and scenarios from stakeholders that illustrate how goals can be achieved have to be taken into consideration [11]. In [41], Port and Bui mixed two primary strategies for requirements prioritization. Their purpose is to get the best from both strategies without having the drawback. The strategies that they mixed are plan-based strategy and agile strategy. The new strategy contains seven process steps. The second process step in their strategy is: Generate a candidate list of ‘base requirements’ from the current collection of stories, where stories are the viewpoints of the stakeholders. Then in the third step a cost and value ranking technique will be used to asses the collected requirements based on the stories of the stakeholders.

2.5.2 Cognitive selection

Another aspect that can effect the validation and prioritization of requirements are the cognitive skills of the stakeholders. Stakeholders have different skills and have different preferences to analyze requirements. According to Felder and Silverman, these preferences have the following dimensions [53] [54]; sensing/intuitive, visual/verbal, active/reflective, sequential/global. As already mentioned in Section 6.2, Carod and Cechich did a case study in [2] where they map stakeholders to cognitive profiles. In this case study, they tested visual and non-visual specifications based on the cognitive profile of the stakeholders. They found that 81.8% of the respondents with strong visual preferences agreed on feeling more comfortable with visual specifications and 36.4% of the respondents with non-visual preferences felt more comfortable with visual specifications. Based on that, they concluded that stakeholders perform better when their cognitive profiles are aligned with the notation of the software requirements specification [2]. As mentioned before stakeholders vary in their technical knowledge and therefore it may not be suitable to apply one general technique to every stakeholder. Therefore, [29] proposed a cognitive requirement classification model where they classify stakeholders as professional or amateur. Both classes have their own model for validating requirements. However, this model is only suitable in small projects because it is not going too much into details and lacks thorough analysis. The performance requirements model proposed in [38] is also based on a cognitive approach. This model uses different performance metrics for different stakeholder’s roles to validate performance requirements. According to [39], in goal-oriented requirements engineering, the use of visual models is more promoted than textual descriptions about the system. This probably implicates that stakeholders find it easier to understand and to analyze the goals of the system within a graphical presentation. For example in a goal tree, stakeholders can directly see what the main goal (root) is and what the sub goals are to achieve the main goal. This technique can contribute to the validation and prioritization process of requirements among multiple stakeholders.

2.5.3 Automated validation mechanisms

Large projects with many stakeholders often have many potentially conflicting requests and requirements. In these projects, the requirements from stakeholders can rise up to over a thousand requirements. Analyzing these requirements by hand can lead to missed deadlines, disorganized development efforts and late discovery of architectural requirements. Therefore, several researchers tried to automate the process of requirements prioritization. Laurent et al., proposed an approach for automating a significant part of the requirements prioritization process in [45]. The method they propose uses a probabilistic traceability model combined with a standard hierarchical clustering algorithm to cluster incoming stakeholder requests into hierarchical feature sets. They did a case study to test this model and their findings were that due to the probabilistic nature, the model did not perfectly detect and categorize all the requirements. However, this model could still be applied for low level and exhausting categorization tasks. This allows stakeholders to work at a higher level of abstraction, which reduces their workloads in prioritizing requirements. In [36], Peng et al. proposes a new requirements maturity concept. This concept is applied in a Wiki system where multiple stakeholders collaborate in the requirements negotiation process. In this Wiki system, the requirements maturity concept determines, based on the degree of modification, satisfaction of the stakeholders and the duration, if the requirement is become mature or not. Therefore stakeholders can only pay attention to those mature requirements and analyze them. This could reduce a significant part of their workload as requirement analysts [36]. As mentioned before in the introduction, in [22], Arnold et al. created a validation framework that supports modelling and automated validation of a set of functional and non-functional requirements. This framework is based on the idea of expressing requirements in terms of responsibilities and scenarios and organizing these responsibilities and scenarios in contracts that can be transformed to components of an actual system. Their findings were that this framework offers a novel solution for automated requirements validation. However, this is very specific to .NET platforms and languages such as C# [22]. In [28], Kof uses natural language processing techniques to automatically identify goals in dialogs of stakeholders. This technique contributes to the Win-Win negotiation approach to resolve conflicts among stakeholders in such a way that the goal of every stakeholder is satisfied. This technique helps to identify conflicts early in the development process.

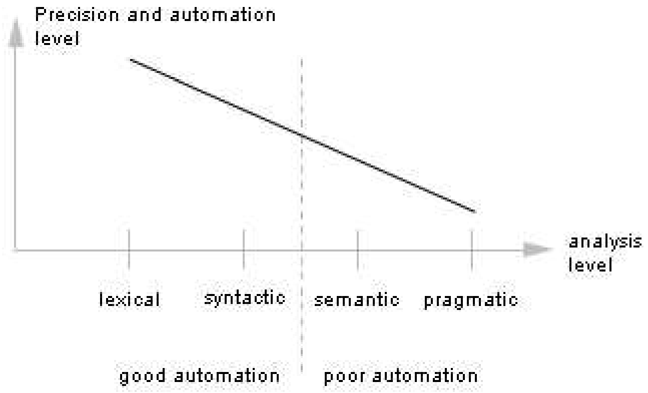


Figure 7: Levels of analysis on goal identification

However, he stated that there are different levels of analysis on goal identification. These levels are; lexical, syntactic, semantic and pragmatic. The best results on goal identification can only be achieved in the first two levels [28]. The precision of the different analysis levels is depicted in the Figure 7.

2.5.4 Cost and benefit prioritization

In a cost and benefit prioritization stakeholders collaboratively determine for each potential requirement what the cost of implementing it would be and how much value the requirement will contribute to the system [11]. As mentioned earlier in this section, Nuseibeh and Easterbrook proposed in their paper [11] a requirements prioritization framework that can be used for agile and plan-based driven software development. Their proposed framework is based on a cost and benefit prediction model and uses two approaches. The first approach is mainly an agile approach and adds a bit of plan based approach by including cost assessment and Pareto ordering in the requirements prioritization. The second approach takes primarily a plan based approach but adds frequent reprioritization (iteratively). They tested their new prioritization approaches through simulations and found that the new approaches produced better results than pure agile or plan-based approaches. However, they only tested their new approach on a limited amount of variations in agile and plan-based software development. Since there are many variations in agile and plan based software development in practise, their results do not represent all these variations. Racheva et al. [4] did a multiple case study on agile requirements prioritization methods and yielded a conceptual model for understanding the prioritization process. The result of the case study suggested that there are five aspects that clients consider in making decisions on requirement priorities. These aspects are;

Business Value, Effort Estimation/Size Measurement, Learning Experience, Input from the developers and External Change. Reflecting this to the cost benefit prioritization approach, the Business Value can be seen as the benefit and Effort Estimation/Size measurement as the cost. However, as the findings of their case study suggested, Learning Experience, Input from the developers and External Change also impact the decision on prioritization and should be taken into consideration in the requirements prioritization process.

2.5.5 Cumulative Voting

Another prioritization mechanism is the Cumulative Voting method. This method is a simple and straightforward voting schema where each stakeholder is given a constant amount of 100 points that the individual stakeholder can use for voting on his or her most important requirements. In this way the amount that the stakeholder assigned to the requirement represents the relative preferences (priorities) he or she prefers among the requirements [47]. However, when multiple stakeholders are involved in the project, this method faces some challenges because not all stakeholders have the same knowledge and influence on the project. Therefore Chatzipetrou et al. proposed a framework in [47] for analysing data obtained from Cumulative Voting prioritization studies. This framework studies the correlation structure of the data, grouping of variables and respondents and also detects the outliers, which contributes to the process of cumulative voting from a mutli-stakeholder perspective.

2.5.6 Conclusion

In summary, the important elements found in the literature to validate and prioritize the requirements in a multi stakeholder's project are; viewpoint of stakeholders, cognitive selection, automated validation mechanisms, cost and benefit prioritization and cumulative voting. Due to the viewpoints of the stakeholders, the requirement analyst will be able to better understand the importance of the value of the requirements. Based on the viewpoints, the requirement analyst can determine which requirements are most important and which are less important. Secondly, stakeholders perform better in validating requirements when their cognitive profiles are aligned with the notation of the requirements specification. Therefore, cognitive selection is an important concept in the validation process of requirements. Using automated validation mechanisms can reduce requirement validation workload in large projects where many stakeholders are involved that together can have thousands of requirements. Automated validation mechanisms automatically categorize requirements, which makes it easier for requirement engineers to do the validation and prioritization. Cost and benefit prioritization techniques allow stakeholders to collaboratively determine for each potential requirement what the cost and benefit of implementing it will be and how much the requirement will contribute to the system. Finally, Cumulative Voting is a simple method consisting of a voting schema where each stakeholder is given a constant amount that he or she can use for voting on the most important requirements. In this way the most important requirements among the stakeholders can be discovered and prioritized.

2.6 Conclusion – Answer to Research Question 1

The found elements from this study can contribute to the thesis by creating an understanding of the issues that occur during the requirements engineering process. The concepts address issues that occur during the requirements engineering process. The found literature in this study proposed approaches, models and frameworks that address these issues and answers the first research question; “What methods/approaches for similar projects and companies are suggested in scientific literature?”. In summary, the overall requirement engineering approaches that are found in the literature are; plan based approaches (i.e. goal oriented and collaboration engineering approaches) and agile approaches that consists of small iterations to address changing requirements. The techniques found in the literature to gather requirements are; grouping diverse stakeholders, interviewing, documenting, and the use of case scenarios and standard templates. For negotiation, the found literature suggest; to use collaboration systems, to document decisions, to be consistent for example by the use of a ontology, to do negotiations iteratively and to consider the emotional factor of the stakeholders. Techniques that can be considered to validate requirements are; reviewing different viewpoints from different stakeholders, making use of cognitive tools to gain an better understanding and to use automated validations mechanisms when there is a vast amount of requirements. Finally, the techniques found in the literature that can be used for prioritizing the requirements are; prioritizing based on costs & benefits and by the to do cumulative voting among the stakeholders.

After understanding the current situation and the issues that occur within Chubb, the found techniques, approaches and methods in this chapter can be used to improve the current process of requirements engineering in market-facing projects within Chubb. Based on the found approaches, mentioned in Section 2.2, and the current situation at Chubb, the most suitable approach can be gained (goal oriented, collaboration engineer and plan-based or agile). Gathering techniques found in this literature study can be used to optimize Chubb's requirements gathering process. During the requirements negotiations, techniques for negotiation found from the literature can be used to improve the negotiations and the understanding of the requirements among the stakeholders. The found validation techniques can be used to optimize the correctness of the requirements and prioritization techniques can contribute to the prioritization process by determining the importance of the requirements in a systematic way.

However, this literature study is not exhaustive meaning that not all the elements discussed in the scientific literature are addressed. Due to time limitations and the vast amount of available literature on requirements engineering, it was not possible to address all elements. Only the elements that were important, according to the thoughts of the researcher, are addressed. Further research can be conducted to gain more elements that can be used during the requirements engineering. The next chapter describes a case study within Chubb in order to gain an understanding of the current situation of the requirements engineering process.

3. Case study – The current situation

In the first section of this chapter, the setup of the case study research is described. Findings from the case study research are described in the second section. In the final section of this chapter, the research question 2 is answered based on the findings from the previous section.

3.1 Case study research methodology

As mentioned before, this study uses a case study as one of its primary research method. The reason for this is that this study tries to understand and improve the phenomena in practice, this requires a qualitative research to understand all variables. Action research would have been an applicable research as well, however this costs more time and would not be suitable within the time available for this thesis.

The research question that needs to be answered in this case study is: “How does Chubb currently gather, negotiate, validate and prioritize the requirements of the iClose system?” which is presented in Section 1.4. Through the fact that this question is a “how” question, this case study will investigate a process.
In this question, the process that needs to be explored is the gathering, negotiation, prioritization and validation of the requirements currently at Chubb in market-facing technology projects. Through that there is little known about the current requirements engineering in market-facing projects within Chubb, the most appropriate design for this research is an exploratory case study [55]. The purpose of this study is to collect data and to make knowledge explicit about concepts that can be improved in the requirements engineering process of market-facing projects at Chubb. During this exploratory case study, the following units of observations were observed.

- Stakeholders
    - Role
    - Behaviour
    - Meeting attendance
    - Expertise
- Meetings
- Iterations
- Available Documentation

3.1.1 Case study research questions

In order to answer the research question:” How does Chubb currently gather, negotiate, validate and prioritize the requirements of the iClose system?”, sub questions are formulated and presented in Table 5. Data that is collected through these questions is both qualitative and quantitative. The nature of the data is specified for each question. Qualitative data is collected from interviews with stakeholders and quantitative data is collected by direct observations due to counting the iterations, meetings, requirements and attendance of the stakeholders.

Overall requirement engineering process		
SRQ 2.1	Does the project have iterations in terms of requirements engineering?	Quantitative / Qualitative
SRQ 2.2	Are stakeholders using a standard methodology or templates during the requirements engineering?	Quantitative / Qualitative
SRQ 2.3	What templates of methodologies are stakeholders using during the requirements engineering	Qualitative

Requirements gathering		
SRQ 2.4	How are requirements gathered?	Qualitative
SRQ 2.5	How many requirements are collected?	Quantitative

Requirements negotiation		
SRQ 2.6	How many stakeholders are involved during the requirements engineering?	Quantitative
SRQ 2.7	How many conflicting demands occurred during the requirements engineering?	Quantitative
SRQ 2.8	How are these conflicting demands addressed?	Qualitative
SRQ 2.9	How are requirements communicated to stakeholders?	Qualitative
SRQ 2.10	Do the developer and the other stakeholders understand each other?	Qualitative/ Quantitative

Requirements validation		
SRQ 2.11	How are requirements tested/validated?	Qualitative

Requirements prioritization		
SRQ 2.12	How are requirements prioritized?	Qualitative

Table 5: Sub questions of research question 2

3.1.2 Case selection and generalization

For this investigation, two cases are selected. The first case is the investigation of the requirements engineering of the iClose project which is explained in Section 1.2. The second case is the investigation of the requirements engineering in a similar project that involves multiple stakeholders and addresses market-facing technology in order to be able to generalize the findings and limitations of this study. This project is called the ARCH project.

At the moment, Chubb Asia Pacific is using Kaizen tools to create, rate and issue insurance policies. These kaizen tools are Word and Excel templates that calculate the premium and generate the wording of the policies. In the ARCH project, these tools will be replaced by a system that is called Avantage. This software package is a custom made package for Chubb and developed in Chubb Canada. With Avantage, Chubb is trying to improve the process of creating, rating and issuing policies in order to save resources and in the future to outsource some parts of this process due to market-facing technology. Therefore, during the development of this system, the market-facing technology aspect has to be taken into consideration and multiple stakeholders have to be involved to improve the process. However, in the ARCH case there are limitations, these are;

- The market-facing technology is very limited and therefore is not the main focus of the project, which is in contrast to the iClose project where the main focus is on market-facing technology.
- The ARCH project is based on existing systems that will be reengineered and improved, therefore the requirements engineering deviates from the requirements engineering approach in the iClose project.

Due to these limitations the main focus will be on the iClose project and the ARCH project will be used to ground the findings in the iClose project.

The generalization of the results will be limited to Chubb since the particular cases only occur internally in Chubb. The effort of this study is mainly focussed on particularization [56] and specification [57] because this research is an in-dept examination of the current requirements engineering situation. Further, due to this examination, concepts can be developed that can be implemented and used to do analyses for future research on requirements engineering within Chubb.

3.1.3 Data collection

The data of the cases will be gathered through multiple sources. These sources are; interviews, available documents and direct observations. A detailed description of each source is described below, starting with interviews.

Interviews
Interviews with users in different functional areas and management levels were held to determine how requirements gathering, negotiation, prioritization and validation is done at Chubb in both projects. These interviews were held at the sites and by telephone during the period February through August 2011. Findings from these interviews are mainly qualitative. A standard interview template is used to

guide the interview; this template is attached in Appendix C and explained here. The first couple of questions are used to identify the interviewees' role and position in the company and the project. The section 'definitions of requirements` is used to extract the overall methodology of how the interviewee is doing requirements engineering. The third section of the interview template `gathering requirements` is used to understand the way of how the interviewee is gathering requirements. The fourth section of the template is `requirements negotiation` and is used to understand how the interviewee solves conflicts and negotiates among the stakeholders in the projects. The fifth section of the interview template is 'prioritization and validation of requirements` and is used to gain knowledge about how the interviewee prioritizes and validates the requirements. The last two sections of the interview template, namely 'general requirements questions' and `case scenarios` are used to identify and understand the issues that the interviewee addresses during the requirement engineering process in market-facing technology projects and in multi stakeholder projects. As evidence and with permission of the interviewee and the company, these interviews are recorded on tape and transcribed for analysis purposes.

Available documents
Documents that are available within Chubb are analyzed to validate the findings from the interviews and to find other concepts of the current requirements engineering methodology. The documents that are analyzed are;

- Plan of approach documents, to gain knowledge on how the requirements engineering is done.
- Meeting notes, to gather information about the negotiation among the stakeholders.
- Other documents, it could be that other documents are used during the requirements engineering, which are not included in the plan of approach. Therefore, during the interview sessions, stakeholders will be asked which document are available and used in market-facing technology projects and multi stakeholders projects.

However, the available documents are analyzed critically since these documents can deviate from how the requirements engineering process is done in the real world at Chubb.

Direct observations
Direct observations are done to observe how the requirements engineering process is currently done in the iClose project. However, no observation were be done in the ARCH project because the observer did not had the permission for this. Findings from the observations are documented through a template that is used during the observations and addresses the following aspects;

- The overall requirements engineering process.
  Are they using a plan-based approach or an agile approach?

- The negotiation among the stakeholders.
  How are conflicting requests of requirements from stakeholders solved?
  How does the requirement engineer communicates the requirements to the stakeholders?
  What techniques does the requirement engineer use to gain an understanding of the requirements by the stakeholders?

- The prioritization process.
  Analyzing the methodology that is used to prioritize the requirements

- The gathering of the requirements.
  Which techniques does the requirement engineer use to gather the requirements?

- Number of iterations
- Total number of stakeholders
- Number of stakeholders in meetings
- Number of meetings
- Number of addressed requirement concerns per meeting
- Total number of addressed requirement concerns by stakeholders

Total number of iterations, total number of meetings, number of addressed issues per meeting and total number of requirements are collected to indicate the complexity. The total number of stakeholders, number of stakeholders in meeting, number of addressed issues per meeting are collected to indicate the understanding of the stakeholders. The template used for the direct observation is attached in Appendix D & E. Further on, during the observations the following aspects are considered;

(i) A requirement concern can have multiple requirements
(ii) Iteration is the amount of loops per requirement concern during the observation
(iii) Total iteration is the amount of meetings about the requirements where the current prototype is discussed during the observation period.

3.1.4 Data analysis

As mentioned in the Section 3.1.2, there are two types of data collected, namely qualitative data and qualitative data. These different types of data have different types of analyses. The analysis of the quantitative data, which is collected through direct observations and available documentation, is done by descriptive statistics e.g., frequency, mean, mode, standard deviation. The results of this analysis are used to support the findings from the qualitative data.

In the case of the qualitative date two analysis approaches were used, namely a within-case analysis and a cross-case analysis. In the within-case analysis approach, the concepts from one source are compared to another source in the same case. Differences and similarities between the sources are described based on this analysis. In the cross-case analysis, concepts found in one case are compared to the concepts found in the other case. Differences and similarities of the concepts between the cases are described based in this analysis. For these analyses, the transcripts that are produced from the interviews and direct observations, and available documentation are used as a data source. These transcripts are analyzed by the use of coding techniques due to a specialized qualitative data analysis software package that is called NVivo. This software package helps to classify, sort and arrange information, to examine relationships in the data and combine analysis. From these analyses, themes are produced, described and explained.

3.1.5 Internal validation

As already mentioned in the section above, there are multiple data sources selected for this case study that are used for analysis. The reason for this is to internally validate the findings of the case study. By having this, multiple sources will eliminate the chance of alternative causes of the results. As mentioned before, these sources are; interviews, available documents and direct observation. Interviews will be recorded, available documents will be copied and transcripts will be made of the direct observations. With this data, other researchers will have the possibility to reproduce the findings found in this research. However, depending on the sensitivity of the data, it could be that interviewees may choose to stay anonymous and parts of the available documents and produced transcripts could be concealed.

3.1.6 External validation

The external validation of the case study is about the generalization of the results. This means that the findings from this case study should also hold in different organizations that are in the same situation. Therefore, this case study should actually be done in two different organizations but due to the fact that this research is only dedicated to Chubb; it would not be possible to investigate another organization. However, two case studies are chosen within Chubb to generalize the research to a certain extent. One case study investigates the requirement engineering in the iClose project that is a multi stakeholder market-facing technology project and the other investigates the requirements engineering in another similar project called ARCH that also involve multiple stakeholders and market-facing technology. As mentioned before, the findings from these cases are compared by a cross-case analysis to find similarities. For example, to gather external valid information, finding X in project (or case) A must also be true in project (or case) B and the other way around.

3.2 Findings from the current situation

As mentioned before in the previous section, interviews are conducted, available documentation is analyzed and direct observations are done in order to retrieve data on how Chubb currently engineers the requirements in market-facing project. The experts that were interviewed are given in Table 6.

Table 6: Interviewed experts

iClose project		
Role	**Reference**	
Project Manager	ICPM1	Appendix F1
CFO	CFO1	Appendix F2
IT Manager (interview iClose)	ITM1	Appendix F3

ARCH project		
Role	**Reference**	
Project Manager	ARPM1	Appendix F4
Business Analyst	ARBA1	Appendix F5
IT Manager (interview ARCH)	ITM2	Appendix F6

Market-facing projects in general		
Role	**Reference**	
Business Partner Service Manager	BPSM1	Appendix F7

3.2.1 Overall process of requirements engineering in MFT projects

During the research period, the overall requirement engineering process is analyzed in order to define which major methodology is used in market-facing projects that involve multiple stakeholders. For this, several questions are defined in Section 3.1.2 that are used as a guideline to gain an understanding of this process. Starting with the first question;

SRQ 2.1: “Does the project have iterations in terms of requirements engineering?”

<u>In the iClose project;</u>
The Project Manager in the iClose project stated in the interview that the requirements engineering is more a back and forward process to elaborate the business requirements. By this, the back and forward process can be seen as iterations, in which the requirements iteratively are elaborated and once it is mature enough it will be signed off;

"It is more of a back and forward process where someone from the business will elaborate the business requirements. The Business Analysts will write it down put it into a document and than back and forward, so it will get refined. Until it is ready to be signed off. [ICPM1]"

During the direct observation, all meetings were addressing requirements concerns and most of these concerns were addressed in multiple meetings (Appendix G). This implicates that the requirements concerns are going through multiple iterations formulate the final requirements.

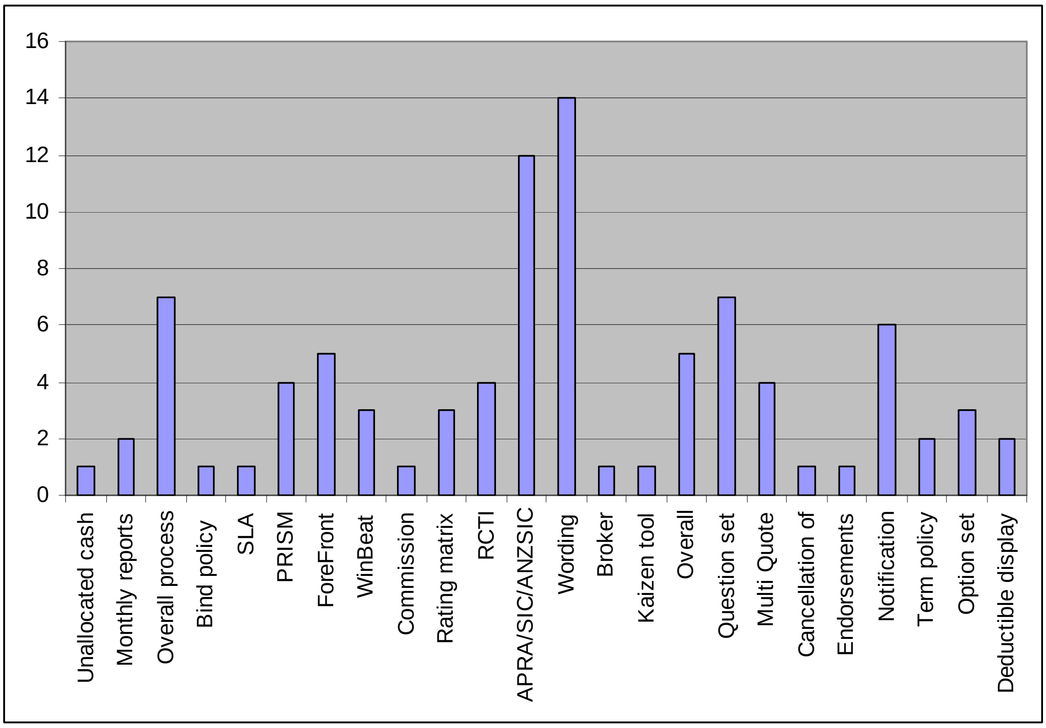


Figure 8: Number of iterations per requirement concern

As depicted in Figure 8, in the observation period, 24 concerns were addressed where 8 concerns had the lowest amount of 1 iteration, 2 concerns had the highest amount of 14 and 12 iterations and the average of amount of iterations per concern is 3,21. Also in the notes that were created after the meetings stated requirement concerns multiple times, which implicated that these are multiple times addressed in different meetings. I.e. in the notes of 24-05-2011 and in the notes of 22-06-2011 (Appendix H) where the occupations code problem is addressed, which means that this is going through multiple iterations in order to generate the final requirements from the concern.

The ARCH project

Two of the four participants that were interviewed from the ARCH project explicitly stated that iterations are used during the development of the project. In the interview with the IT Manager [ITM1] about the ARCH project, he stated that;

"What we are planning to do, and we have done this twice, is to demonstrate it in an iterative approach of the development lifecycle. So basically we develop something of the system and we showed it to make sure that they agree with the changes that we made. [ITM2]"

Further on, the Project Manager of the ARCH project suggested that they are using the agile approach for some parts of the project;

"You are familiar with the kaizen tools that we have here? Yes. Ok, this is the agile portion; the Canadian team have taken the kaizen tools and reverse engineered them into Advantage. [ARPM1]"

This is in line with what the IT Manager says about the iterations because the agile approach consists of multiple iterations. Besides this, there is also a document that consists of a registration of attendance of the participants during the iterations (Appendix I). In this document, there are two iterations where the attendances of the participants are recorded and a third iteration is made that will be used to record the attendance of the participants for the next iteration. This document is an excel document with the following tabs: 14-04-2011 (Prototype Review), 15-04-2011 (INT Demo) and 18-04-2011 (INT Demo). Each tab represents an iteration and consists of a table of attendance, which grounds the statements made by the Project Manager and IT managers.

Findings from question SRQ 2.2 and SRQ 2.3: "Are stakeholders using a standard methodology or templates during the requirements engineering?" and "What templates of methodologies are stakeholders using during the requirements engineering?"

The iClose project
According to the IT manager and the found documentation on the intranet of Chubb (Appendix J) the general methodology that is used for IT projects within Chubb is SDLC;

"*Basically we are following the standard SDLC (software development life cycle). This is a set of phases where we need to go through including testing and releasing, migration testing*" [ITM1].

However, as the CFO and the IT Manager suggested, that the iClose project is different from traditional projects in terms of requirements engineering and development;

"*The iClose project is probably a little bit different from how it normally go about gathering requirements for a IT project*" [CFO1]

"*If you look at the methodology used for the iClose product, it is actually quite a different methodology to what we used for other projects*" [ITM1].

Several sources suggested that the approach used in the iClose project is more a prototyping approach, in which they monitor, elaborate and validate the development of the project on a periodically bases, for example every week. During these iterations the business see how the new system is going to work and based on that they give new requirements and validate the current requirements, the Project Manager of iClose stated that;

"*The other thing is it may minimize the risk is because we are doing prototyping. We don't have people that go a way and build something for 6 months and then come back. We have to see something every week. Because of that we know that some things aren't working and we can change the approach*" [ICPM1].

Findings from the direct observations show that there are several (6 observed) meetings in which the prototype of the iClose system is discussed among multiple stakeholders (Appendix K). During these meetings, stakeholders validated the current system, requirements concerns and suggested new requirements concerns. These findings are in line with the findings from the available documents. For example the outstanding issues from the meeting of 16-02-11 and 22-02-11show that issues of 16-02-11 are discussed and a new requirement concern "Document generation and email to Broker and CICA" is added in the meeting of 22-02-11 (Appendix L).
In the interview with the Business Process Sevice Manager, who was involved in another market-facing project in another country, he stated that this other project also used prototyping as the development and requirements engineering method;

"*That is called PeopleSure, and we are rolling it out in Australia at the moment. In terms of the requirements for that project, it was done in Agile, so more in a prototype type affair*" [BPSM1].

The documentation that is used in that project is a Business specification documented in which the requirements are documented. He said that this document is generally used in these kinds of projects;

"*Yeah, there generally is a kind of a business spec (business specification document)*" [BPSM1].

In terms of the iClose project, this statement is confirmed from several sources. The CFO stated that there is a business specification document that is used to document the requirements. With this documentation, the developer and other stakeholders will be able to understand the requirements that are formulated by the business;

"*There has been a business requirement document writing which have been come out of the meetings with the business. It more a guide or a reference point then a formal document is used by developers to program*" [CFO1].

From the available documentation analysis, this document is found and attached in Appendix M.

Further on, during documentation analysis, another template was found. This template is used in meetings to document and monitor the requirement concerns and consists of a table with the following columns; no, priority, requirement concern, comments from Chubb, comments from the developer, complexity and effort. Examples of the template are presented in Appendix L. Besides this, it seems to be that not all the available documentation and templates within Chubb for software development and requirements formulation are used, according to the CFO of Chubb Australia;

"*We are not using, or following all the documentation that is out there in Chubb. But certainly the important documentation*" [CFO1]

According to the Project Manager this is because the project does not consider very critical processes and because of time limitations it would not be realistic to use all the available documents and templates;

"*I would not be surprised if that is a very detailed framework. We use some of that, but how much we use is definable. Because the other thing is; it depends on the project. If you spending 20 million dollars, you make sure that the requirements are right. If you spent 20 million dollars and the system isn't used. You will get into trouble. On the flip side; if you are spending 20 000 dollars, you do something fast. You think it is going to work; it might work of it might not work, but you got a greater leap up there because you aren't losing that much money. For the iClose project we are spending a couple of 100 000 dollars maybe more, but we have a greater latitude to accept risk*" [ICPM1]

The ARCH project

In the ARCH project, the requirements engineering approach is also mainly based on prototyping. The Business Analyst and the Project Manager of the ARCH project and the CFO mentioned that the major approach for requirements engineering and software development used in the ARCH project is done by prototyping;

"*This is the agile portion; the Canadian team have taken the kaizen tools and reverse engineered them into Advantage*" [ARPM1]

"*They build the product requirement into the prototype and then they discuss the prototype*" [CFO1]

"*When prototype is build we give a presentation. That is how we give the business the chance to show them how the prototype is working. Then they try to use the prototype and they suggest; if it is working the way they want or if they want to change something or if they want to remove something, that kind of stuff*" [ARBA1].

As the Business Analyst already suggested, once a prototype is ready the stakeholders will sit down and validate the requirements and negotiate what they want to change, add or remove. However, some parts of the development and requirements engineering in the ARCH project is also done due to the "old" methodology. The Project Manager stated that if there is functionality required that is currently not supported by the old existing system then the Business Analyst will use this methodology to gather requirements from the experts;

"*But there are certain requirements that are needed in Advantage that kaizen do not support. For those, we have a Business Analyst, taken the old methodology.*" [ARPM1]

However, the Business Analyst clarified that when there are some parts of the requirements not clear, this will be gathered from multiple sources instead of extracting it from the system;

"*You still need to gather the requirements because it is not clear enough or just not right*" [ARBA1]

In the ARCH project, there are several templates used for requirements engineering. The Project Manager mentioned that there are many temples available within Chubb, but, as similar to the iClose project, only the templates that are critical are used;

"*We have standard templates. That are templates that Chubb is using for gathering requirements. Although it is very broad, it kind of makes it what you need to make it*" [ARPM1].

During the documentation analysis, several templates were found. These templates are attached in Appendix N. The first template in Appendix N is used to formulate the business specification document (which is also found in the iClose project). Due to this template the stakeholders, requirements, goals and scope, the testing and implementation are defined and explained. The second template, in Appendix N, is used to formulate the processes that are impacted by the system under development. Once this template is filled out, it explains the impact and the workflow of the system under development.

Conclusions from questions SRQ 2.1, SRQ 2.2 and SRQ 2.3
Based on the findings from the first question, it can be concluded that most of the requirements concerns are addressed multiple times in order to draw the final requirements. From this, it can be concluded that iterations do occur during the requirements engineering and the project development of market-facing projects that involve multiple stakeholders. However, some requirements concerns are addressed only once. This can be through that these requirement concerns are very clear and obvious to everyone and therefore the stakeholders agreed on the concern directly in the first iteration.
Further on, the methodology that is used for market-facing projects that involve multiple stakeholders is generally based on prototyping. This approach consists of periodically meetings wherein the current software prototype of the system is discussed and validated. In this approach the development and requirements engineering are done simultaneously, which has the benefit that requirements can be validated in the next prototype meeting.
Besides this, there seems to be one general template that is used in several kind of market-facing projects. This template is the business specification template (Appendix N) wherein the requirements are documented and elaborated according to the feedback of the stakeholders.

3.2.2 Current situation of requirements gathering in MFT projects

Several aspects are analysed to determine how the requirement gathering is done at Chubb in the iClose and the ARCH project. These aspects involve how stakeholders are selected in the requirements engineering process, how they gather requirements, the techniques and methods they use and how they document their requirements. Based on the findings of these aspects, the questions SRQ 2.4 and SRQ 2.5 are answered.

Selection of stakeholders
Requirements are gathered from several sources. One of these sources is the stakeholders who have certain expertise in which they can contribute to the gathering, validation, prioritization and negotiation process of the requirements. Therefore, the process of how the stakeholders are involved in the project is analyzed. Findings from several sources in the iClose and ARCH project suggest that the stakeholders are selected based on the following aspects;

Expertise of the product
Several sources suggested that the people who have the knowledge of how the product works should be involved during the project as stakeholders [CFO1, ARBA1, BPSM1]. This could be; the end users as well as managers that understands how the product works or should work. For example in the ARCH project the managers are chosen to be the stakeholders because they know how the product works or should work. If they do not know it, they will go to their experts or end-users and asked how it works or should work;

"*Yes but we don't get back to the persons, I get back to the SBU managers. And they need to answer the questions. If they don't know, they can go back to their team and ask the questions*" [ARBA1].

Sponsorship
Another aspect that is mentioned by several sources in the selection of the stakeholders is; sponsorship [ICPM1, BPSM1]. The existence of a project is because there are people who finance the project to achieve a certain goal. These people are chosen as stakeholders in the project because they can translate their goal into requirements. For example as the Project Manager of the iClose project suggested that one of the sponsor stakeholders could be the end customer who pays for the project;

"*There are people they need to be involved because they got to deliver stuff and sign off and the end customer, because they pay for it*" [ICPM1].

Deliverables
A third aspect in the stakeholder selection is suggested by the IT Manager, namely; There several stakeholders chosen by the fact that they have to deliver the final product [ITM1]. Often these people are the developers. These people have expertise in the feasibility of the system under development. They are able to determine what is possible, what not and how much time it will cost.

During the direct observations in the iClose project, the involvement of the stakeholders was registered in order to validate the suggestions by the interviewees. If these factors are considered during the stakeholder's selection then at least one stakeholder of each factor should be involved in the project. From the 24 meetings that were observed, the involvement of the different types of stakeholders is (Appendix O);

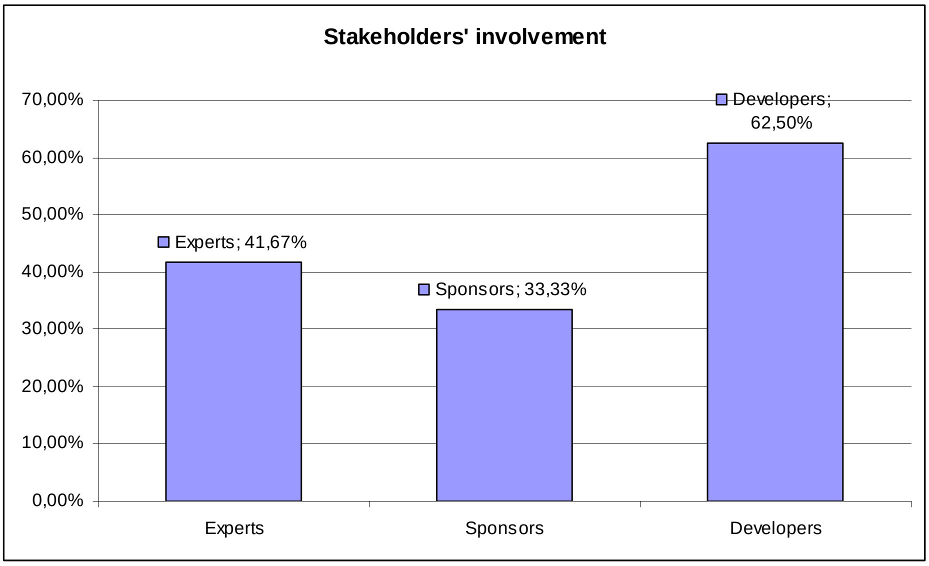


Figure 9: Involvement of different types of stakeholders

As can be seen from Figure 9, all the types of stakeholders are involved in the project. However, this does not fully prove that there is a relationship between the selection factors and the stakeholders in the project because it could also be coincidence that in this project all three stakeholders are involved. To prove that there is a real relationship, multiple projects within Chubb need be observed. However, due to time limitations and authority reasons this was not possible within this research.

Gathering process, techniques and methods
In the iClose and ARCH project the gathering of requirements is done in multiple ways. Techniques that are found during the research are; market research, existing systems, workshops, case scenarios and interviews in terms of face-to-face meetings. Each aspect is further elaborated below starting with the market research.

Market research
One of the first steps for involving market-facing technology in the project is conducting a market research. Several experts of the iClose project mentioned that a market research was done [ICPM1, CFO1, ITM1]. For example the Project Manager stated that they did a market research in order to understand what the customer wants;

"*We also did some market research. So we said to the brokers. What do you think you want? You want this or this. If you have a choice you will get A or B? What is the most important thing for you? How would this fit in your process? So we have done a lot of that*" [ICPM1].

However, in the ARCH project they did not conducted a market research jet but as the Project Manager of the ARCH project suggested during the interview, this is because they did not started with the market-facing technology aspect. She mentioned that when they start with the market-facing technology aspect, they will do a market research;

"*I think the market research comes in when you want to add more addition functionality like the market-facing. But that is the second step*" [ARPM1].

As can be seen from the statements of the Project Managers, in both projects they conduct a market research in which they ask the market several questions to extract their requirements for the new system. A template that is used during the market research of the iClose project was found during the search on available documents. This template is attached in Appendix P.

Existing systems

Another method that Chubb use in market-facing technology projects to gather requirements is by looking at existing systems. These could be systems that are used by competitors as well as systems that are used internally [ITM1, BPSM1, CFO1]. The requirement engineer analyzes these systems and documents what should be considered in the new system. The reason to look at competitor systems, suggested by the IT Manager of the iClose project, is to identify critical success factors through analyzing the indistinctness's of their system;

"*We would look at the competitor to see what they have done, because we don't want to make the same mistakes*" *[ITM1].*

Several documents are found that contain screenshots of systems from competitors, which is in line with the findings from the interviews. Some of these screenshots are attached in Appendix Q.

However, in the ARCH project, the requirements are not gathered through existing systems of competitors but through systems that are used internally [CFO1, ITM2]. For example the CFO, who is one of the sponsor stakeholders in the project, stated that;

"*For the ARCH project it is done through the existing kaizen tools. By analyzing the kaizen tools you are doing the requirements gathering*" *[CFO1].*

Screenshots of these kaizen tools are made and passed through to the developers. Based on these screenshots they understand what functionality has to be in the new system. From this it can be concluded that existing systems within Chubb as well as existing systems from competitors are analyzed in order to the gather requirements.

Workshops

Multiple interviewees from the iClose project stated that they using workshops to extract requirements for the system under development [ICPM1, ITM1, BPSM1]. According to the Project Manager; during these workshops they have a round-table discussion with representatives from the business and IT. In these discussions they brainstorm and use process maps to elaborate their ideas into requirements. The Business Analyst then documents these requirements into a business specification document. However, none of the interviewees of the ARCH project stated that they did a workshop to gather requirements. A possible reason for this could that they did not jet started with building the market-facing technology in the project. On the other side the Business Partner Service Manager, who is involved in multiple market-facing project around the globe at Chubb, stated the following;

"*For the stakeholders' involvement it is all about creating relationships to those people to understanding their business. In that case it is more about; just talking to them, starting of with a workshop and from that workshop making sure you engage all the stakeholders regularly. Market-facing technology, in the example*[†] *that I gave earlier, it was more about Chubb providing something to the market that was easy to use, very easy to use and very quick and efficient. So it was not only about going to the market, going to the brokers and asking; what do you want? It was more about thinking what are we going to launch to the market and what are the objectives. So the requirements were still performed internally and then we came into testing it*" *[BPSM1].*

From his experience and the experience from experts in the iClose project it can be concluded that workshops are used in the market-facing projects within Chubb.

[†] The example that BPSM1 gave is about a market-facing project that is called Peoplesure and was developed in the United Kingdom.

Case scenarios

A technique that is used in the ARCH project to gather requirements is by the use of case scenarios. According to the Project Manager of the ARCH project, they created case scenarios based on the documents that were generated by the existing systems. She stated the following;

"*For documents we do case scenarios, we are getting examples of documents that they already issued because they bring out additional issues that the generator might not think of. So you say; give me your most complex policy and then you look through that and you will see exceptions*". *[ARPM1]*

However, no other sources suggested in the ARCH that case scenarios are used and neither in the iClose project. Probably case scenarios are used but not explicitly, for example during workshops. Therefore, this is also considered as a gathering technique that is probably used in Chubb during the requirements engineering in market-facing technology projects.

Face-to-face meetings

The final requirements gathering method that was found during the research is face-to-face meetings. In face-to-face meetings, the requirement engineer sits down with a specific stakeholder that has a certain expertise and tries to extract the expertise from the specific stakeholder and transforms this into requirements. In the iClose project, the Project Manager mentioned that someone from the business will elaborate the business requirements and the Business Analyst will write this down (see below), which means that they are sitting together to elaborate the requirements.

"It is more of a back and forward process where someone from the business will elaborate the business requirements. The Business Analysts will write it down put it into a document and than back and forward, so it will get refined". [ICPM1]

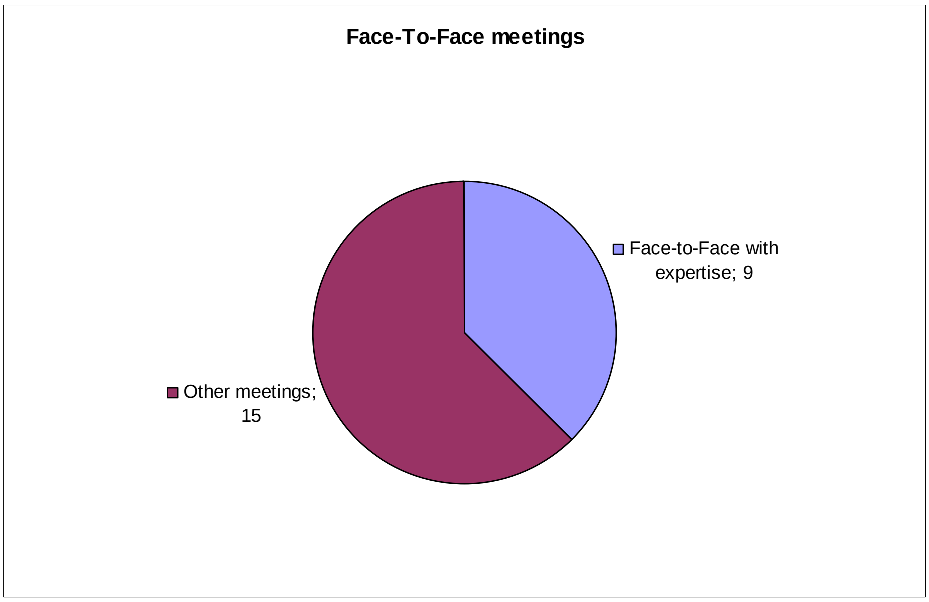


Figure 10: Observed amount of face-to-face meetings

As depicted in the Figure 10, from the observation period, there were 9 of the 24 meetings observed in the iClose project (Appendix R) wherein a business specialist sat down with the Business Analyst or Project Manager to explain and elaborate the requirements. From these meetings, several notes are made. Some of these notes are attached in Appendix S.

Besides the iClose project, the Project Manager of the ARCH project also mentioned that they are using face-to-face (interviews) meetings to gather requirements from experts [ARPM1]. From this it can be concluded that face-to-face meetings are held to gather requirements in market-facing project that involve multiple stakeholders.

Documentation

During the gathering process, in both projects, the requirements that are elicited are documented in a business specification document. Several experts mentioned this in the interviews [ICPM1, ARPM1, CFO1]. These statements are justified with the findings from available documentation in both project and the direct observation in the iClose project. From the available documentation, both projects used a business specification document (Appendix M & N[‡])

In the observation period in the iClose project, several (24) requirements concerns were addressed. These requirements concerns were discussed during meetings and from these discussions the requirements were derived and stored in the business specification document. I.e. as can be seen in the observation analysis, the requirements concern "APRA/SIC/Occupation list" is addressed multiple times. This means that this concerns is discussed in multiple meetings and from these discussions, the result is documented in the business specification document. In the beginning they wanted to use a SIC list consisting of +/- 1000 options and mapping it to ARPA codes. After several discussions they found that the ARPA mapping is not required and that there are to many SIC codes which is opposed to the user friendliness. Therefore, they decided to use an occupation category list that has a limited amount of options to choose from. This list is finally stored in the business specification document and passed through to the developer as a requirement that this functionality has to be in the system (see Appendix M). Further on, no other documentation or templates were found that are used during the requirements gathering process.

Conclusion

In this paragraph, the research questions SRQ 2.4 and SRQ 2.5 are answered based on the findings presented in the previous subsections. Starting with question SRQ 2.4:

Findings from question SRQ 2.4: "How are requirements gathered?"

In summary, before gathering the requirements, stakeholders are selected. To select these stakeholders for the project, the requirements engineer considers three aspects, namely;

- Expertise of the product. Based on the expertise of a stakeholder on a product requirements can be drawn, validated and prioritized.
- Sponsors. People who finance the project are considered in the selection of the stakeholder because they are the persons who define the ultimate goals of the project.
- Deliverables. Based on what needs to be delivered, the people who are able to build it need to be involved in the project. Therefore, deliverables is one of the concerns that is addressed in selecting the stakeholders.

Once the stakeholders are defined and the scope of the project is clear, the requirement engineer starts gathering the requirements. In this process he or she uses the following techniques; doing a market research, analyzing existing systems, organizing workshops, making case scenarios and conducting face-to-face meetings to extract requirements from experts. When the requirements are gathered, they are documented in a business specification document, which is passed through to the developer.

Findings from questions SRQ 2.5: "How many requirements are collected?"

During the observation period in the iClose project it turned out that the exact number of requirements is very hard to measure because the requirements are defined in multiple ways. For example, the requirements are defined through images, tables and examples wherein multiple requirements are addressed. On the other side, during the observations it was possible to measure the requirements concerns that were addressed in meetings. This measurement can also give an indication about the size of the project, if there are other numbers of requirement concerns available from other projects. However, due to that the iClose project is the only project where the number of concerns is available; it was not possible to determine the size of the project in terms of the number of requirements. Therefore, more research is needed within Chubb. The number of requirements concerns that were addressed during the observation period is 24. Due to that the observation is only done in a limited time frame of the project and not during the whole development of the project, it could be that this number not includes all concerns that are addressed.

[‡] Due to authority restrictions, the researcher was only able to retrieve a concealed document of both projects

3.2.3 Current situation of requirements negotiation in MFT projects

Unfortunately, not all requirements are inline and could conflict with each other. During the observations in the iClose project, there were 2 of the 22 requirement concerns that are outliners (Appendix G) in terms of iterations compared to the other concerns, this is depicted in Figure 11 and for simplicity duplicated below.

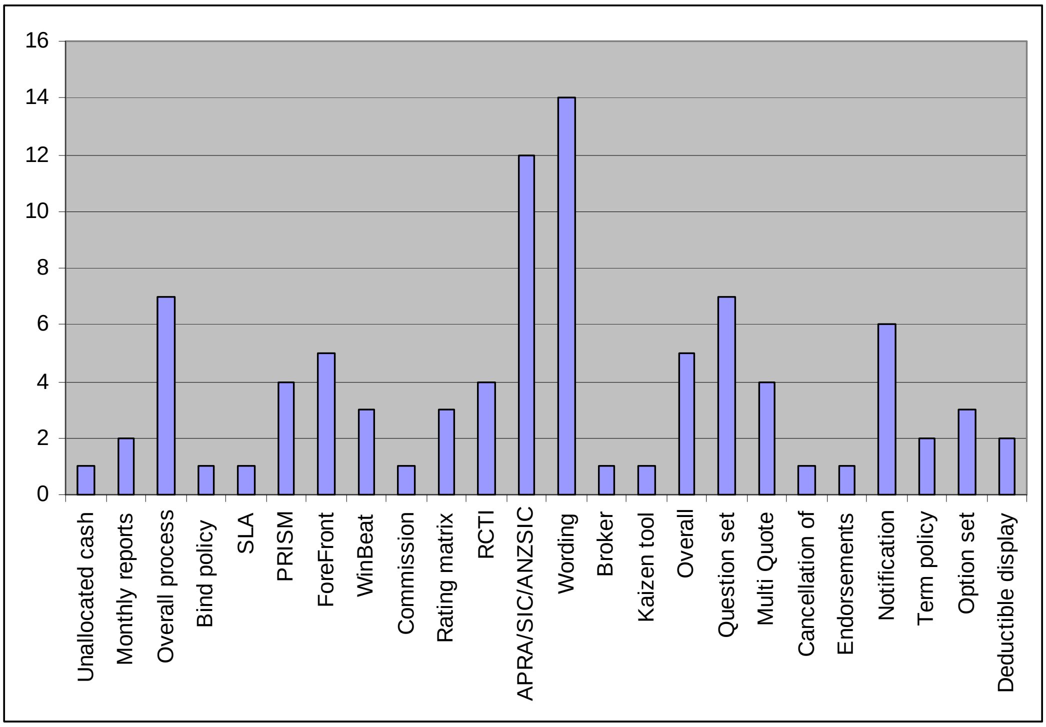


Figure 11: Number of iterations per requirement concern

From this finding, there is a high probability that conflicts occurred in these requirement concerns. In order to solve these conflicts, stakeholders have to negotiate with each other to get to an overall agreement about what they want and what they need. According to the requirement experts, this is done through several methods, where the overall process that incorporates these methods is to solve the conflicts is as follows;
Basically, there are three factors addressed when solving conflicting requirements in market-facing technology projects within Chubb, namely; scope, quality and time & cost. The first thing that is addressed is the Scope. If a conflict occurs in the requirements engineering, the requirement engineer will look at the scope of the project to see if it the requirements are inline with that;

"*You go back to scope. When you pull back to scope, it usually works. In terms of conflicting requirements*" [ICPM1]

If it is not, then the requirements will be dropped. If it is, then the quality of the requirements will be assessed. The requirement engineer will validate the quality of the conflicting requirements by making everyone aware of the essence of the conflict and based on that a decision will be made;

"*Basically you are hooking at the triple constrained and try to negotiate with the stakeholders to explain the position. In IT we can't tell the business what they should or shouldn't have. The business is always the driver of the development process.*" [ITM1]

The third factor that influence the negotiation process is time and cost. The conflicting requirements are measured on how much time it will take to realize it and how much it will cost. Based on that, the requirement engineer decides to extend the time of the project or the costs in order to incorporate these requirements. Several sources suggested that these factors are the main drivers in the negotiation process on requirements engineering [ICPM1, ITM1, ARPM1]. I.e. as the Project Manager of the iClose project suggested;

"*In a project you got three things you can manage and that is the time frame, scope or requirements and the amount of money you spent on the project. The easy way to negotiate out these requirements is like; you can put that requirement into the systems, by either increase how much you spent and put more resources in the project or you need to let the same amount on the team and extend the time frame. So we can do that, but it will take longer or cost more or we can reduce the requirement. There is a sort of area matrix that you can work through because usually you can't say we can do more with the same amount of money or we can do that with the same amount of time. You need to be able to manage it like that. That is a very good way.*" [ICPM1]

To establish a good negotiation, all the stakeholders that are involved have to understand the requirements. Fortunately, the requirement engineers in market-facing projects at Chubb are aware of this. For example, the IT Manager stated that this avoids scope creep and makes sure that all the stakeholders know what they are talking about:

"*It is obviously very important to for the stakeholders to understand the requirements. And I guess from another perspective, it avoids surprises and scope creep. Having the stakeholder to understand what we are going to deliver, makes sure that everyone is on the same page and there are no surprises.*" [ITM1]

Therefore, they use several techniques to create an understanding of the requirements among the stakeholders. According to multiple sources [ITM1, ICPM, BPSM1, ARPM], these techniques are;

- Visualisation is used to demonstrate how things work or look like. The requirement engineers create process maps to demonstrate the processes that the system under development takes care of (see Appendix A) and they create images about how the system under development is going to look like to demonstrate the functionality.

- Face-to-face meetings to demonstrate the system. In both projects, these meetings are held to allow the stakeholders to see how the system is going to work. Due to this, the stakeholders will better understand the purpose of the system. Several sources stated that these meetings are held in market-facing projects;

"*It is really a kind of demonstrating the system, come with a prototype or document in the requirements and you kind of give a face-to-face explanation*" [BPSM1]

"*Now the videoconferencing allows us to demonstrate the system. They can actually see the person that is speaking and this helps a lot. You got a face-to-face meeting*" [ITM2]

- The use of a collaboration system. This system is only used in the ARCH project but several sources from the iClose project suggested that the use of a collaboration system would be useful in the requirements engineering process;

-

"*Although I know it is very useful, it could be fantastic, especially at Chubb, because we are multi-sided*." [ICPM1]

"*To be honest we haven't. That is generally not our mode. But on the other side, it can be fruitful because we are widespread and there are a lot of tools that will allow us to collaborate effectively*." [ITM1]

In the ARCH project the requirement engineers and the developers are communicating through a collaboration system that is called SharePoint. In this system, the requirements from the requirement engineers are passed through to the developers and discussed. This allows them to understand what is going on, as the Project Manager of the ARCH project mentioned during the interview;

"*It is a place to store the documentation and then there is a section where you can record an issue and you can respond to it, so you can see the trail of communications. We are trying to use that* *instead of email, because a lot of time in the email a lot of things get lost or not everyone is copied in*." [ARPM1]

However, the Project Manager also mentioned that depending on the level of involvement of the stakeholder it is not necessary that all stakeholders have to know exactly what is going on. According to her, some stakeholders do not have to understand the requirements at a very detailed level but they have to understand the key issues [ARPM1]. The reason for is that some stakeholders do not want to be aware of all the details in the development of the project. Their biggest concern is to make sure that the goal of the project will be achieved, i.e. the sponsors and business experts of the project. For example in the iClose project (Appendix O), there are 5 stakeholders of the 11 that participated in less than 5 meetings (Figure 12). These stakeholders probably do not care about all the requirements in details because they are not actively involved in the development of the project.

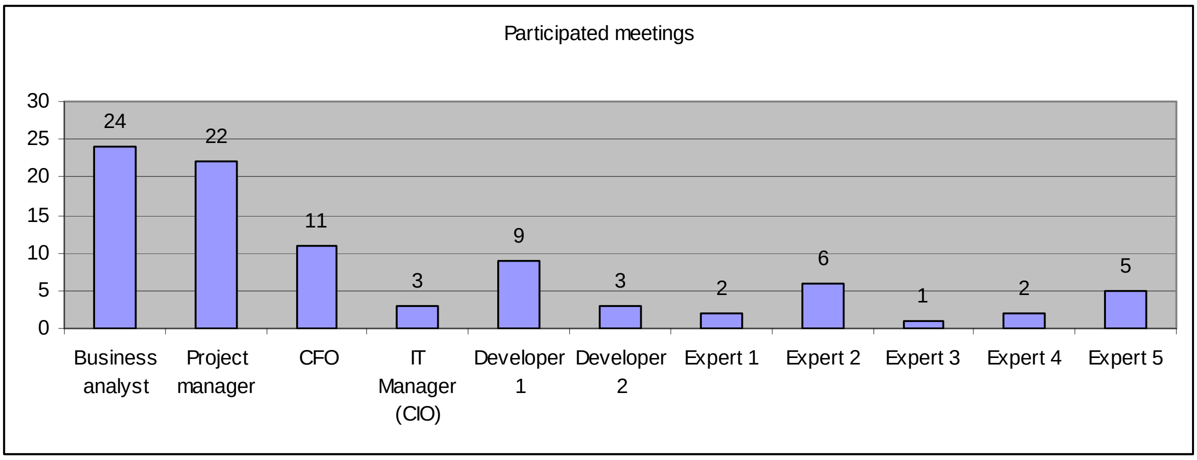


Figure 12: Amount of participated meetings per stakeholder

Although there are several techniques used for improving the understanding, sometimes it happens that the developers do not fully understand the requirements. The Project Manager of the ARCH project stated this occurs and therefore they have to be explicit in formulating the requirements;

"*Yes the developers don't understand the insurance business and that is why the requirements are so important. What we are trying to do is to tell them from a business perspective why something is important*." [ARPM1]

From the observations in the iClose project, the amounts of iterations per requirement concern discussed with the developer and the business are recorded and analyzed (Appendix S). From this analysis, there were no outliners between the business (stakeholders) and the developers among any of the requirement concerns. As can be seen in Figure 13, the number of iterations of developer 1 and 2 are in line with the iterations of the rest of the business (stakeholders). This means that the developers did not have any extra meetings with the Project Manager or Business Analyst to discuss specific requirement concerns, which could indicate that they understood the requirements that were presented by the business. However, due to that not all the communication is done through meetings, it could be that the developers discussed requirement concerns with the Project Manager through an alternative communication channels. For example; through a direct phone call.

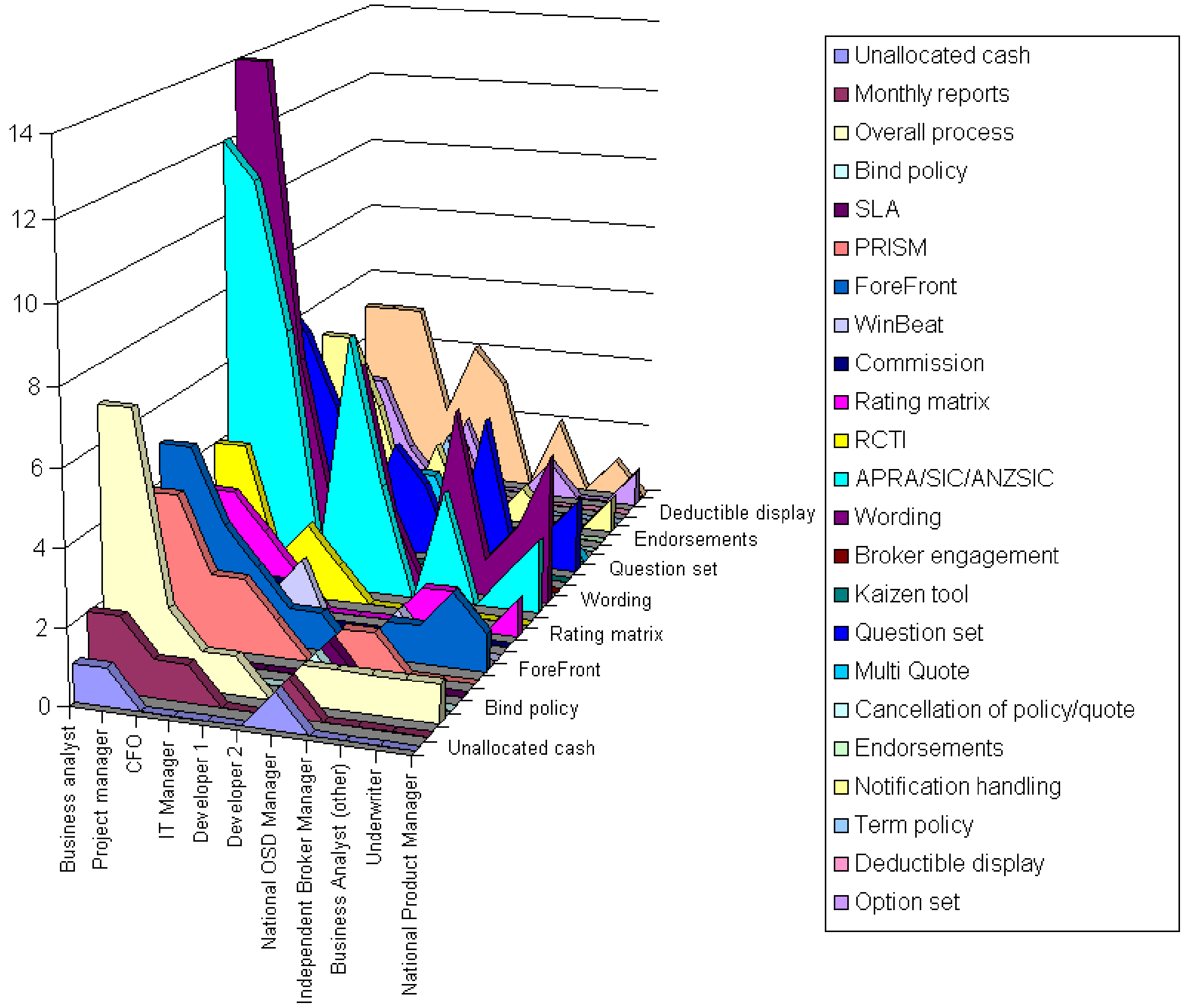


Figure 13: The number of iterations per stakeholder per requirement concern

Conclusion

Reflecting the findings from the case study back to the research questions, during the observations in the iClose project, 11 stakeholders were counted that are involved during the requirements engineering. This answers the first research question about requirements negotiation: SRQ 2.6 “How many stakeholders are involved during the requirements engineering?” Besides this, also the amount of meetings wherein the stakeholders participated was also counted. From this, there are several stakeholders that participated in the meetings less than 5 times and therefore they probably do not understand all the requirements in detail as also is suggested by the Project Manager of the ARCH project. However, for this, more research is required but due to time limitations, this was not possible to carry out in this project.

Further on, also the amount of requirement concerns and the quantity of the times that these concerns are addressed during the meetings are counted in the iClose project in the observation period. There were 22 requirement concerns addressed during the meetings and from these concerns, 2 of them were outliners that were addressed more than 10 times. A reason for this could be that there were possible conflicts and therefore these concerns were addressed many times in order to get to a solution. Also from the notes that were made during the observation, there seems to be some conflicts in these requirement concerns. Reflecting this to the research question SRQ 2.7 “How many conflicting demands occurred during the requirements engineering?” of this case study is 2.

According to several sources, the factors that are addressed to solve these conflicts are (research question SRQ 2.8);

1) Reflecting the conflicting concerns to the scope of the project to determine if it is inline with the scope
2) Validate the quality of the requirements that conflicts by making everyone aware of the essence of the conflict and based on that a decision will be made.
3) Measure the conflicting requirements on how much time it will take to realize it and how much it will cost, based on that it will be decided to extend the time frame of the project or the cost of the project to incorporate and solve these conflicting requirements.

These three factors are addressed during a negotiation process with the stakeholders. In order to start this negotiation process the stakeholders have to understand the requirements. Therefore, the requirement engineers are using several techniques to communicate it to the stakeholders (research question SRQ 2.9 & SRQ 2.10). The techniques that they are using are;

1) Visualisation, like process maps and images to demonstrate the functionality and how the system is going to look like.
2) Face-to-face meetings, were the requirement engineer explains in a face-to-face meeting how the system is going to work.
3) Collaboration systems, that stores the trail of communications in order to increase the understanding of the stakeholders about how and why decisions are made.

Further on, the Project Manager of the ARCH project suggested that the requirements are not always understood by the developers. However, this is not in line with the findings from the observations in the iClose project. The findings from the analysis of the observation in the iClose project did not show any deviations in the iterations per requirements concern between the business and the developers. This could mean that the developer understands the requirements or that this is communicated through other communication channels than normal meetings.

3.2.4 Current situation of requirements validation in MFT projects

The elicited requirements need to be validated in order to make sure that they are correct. Requirement engineers at Chubb are using several techniques or methods to do this. This paragraph describes the validation techniques that are used in the market-facing projects within Chubb, starting with the first, namely; multiple sources.

Multiple sources
The most frequent used method within market-facing projects at Chubb to validate requirements is the use of multiple sources. During the requirements engineering process the requirement engineer uses multiple sources to check if the elicited requirements are right. Multiple requirements experts mentioned this technique during the interviews, although mostly in different fashions. For example the CFO and the Project Manager of the ARCH project gave an example about that the elicited requirements from the stakeholders are validated through the existing systems and documents;

"*The IT Manager did the mapping of the information from the project that is required for Prism. You want be able to get that booking, unless you have that information, so when the underwriters coming up with the requirements, the IT Manager will cross this back to Prism to make sure it is complete.*" [CFO1]

"*What I do is; I go to the people who actually use the system today and ask: how do you do that and what do you actually see? I look at data in the system to see how it actually sits up there to validate about what is going on. It is getting the information from different sources just to validate.*" [ARPM1]

As the Project Manager already suggested, it is about getting the information from different sources. Besides documents and existing systems, the Business Analyst in the ARCH project also gives another example. She mentioned that she is going to multiple experts to verify if the elicited requirement is correct;

"*We decided not to go to one people but multiple. You have to go to two persons that know everything.*" [ARBA1]

These findings are inline with the findings from the observations. During the observations in the iClose project, the Business Analyst and the Project Manager organized multiple meetings with experts to verify requirements, for example; (Appendix U) observation no. “13 – Meeting about the question set with the Independent Broker Manager” and observation no. “24 – iClose outstanding issues” where in both meetings the question set was discussed with other stakeholders. During the meeting in observation no. 13 the question set was validated with the Independent Broker Manager and during the meeting in observation no. 24 this questions set was verified by other experts, in this case the CFO and National Product Manager.

Sign off

Another technique that is mentioned by several requirement engineering experts is to sign off the requirement by the stakeholders [ITM1, ICPM1, BPSM1]. Once the requirement engineer elicited the requirement from the stakeholders, they go back to the stakeholders or their manager and asked them to sign it off. If they sign off, the specific stakeholder will take the responsibility that according to him or her, the requirements are correct and that the requirement engineer can pass these requirements through to the developer. Most of the time, the specific stakeholder then realizes that he or she have to be sure that the requirements have to be correct and therefore he or she will often involve other stakeholders to verify these requirements;

“*Because you can get all the requirements from someone that can’t sign off and he sends you to Rodney and Rodney goes: Oh these requirements are not right, I can’t sign this off. And you think you collected 80% of them, so then you are in the wrong hole. So the way you setup these questions is very useful. It sort of works you through the process.*” [ICPM1]

In the business specification document (Appendix M and Appendix N), a final chapter that is called “Sign off” or “Approval” is included wherein several stakeholders have to sign off the document. When they sign off, they all agree that the business requirements presented in the document are correct. The Business Partner Service Manager, who is involved in multiple market-facing projects around the world within Chubb, mentioned that this technique is generally used in software development projects within Chubb;

“*A lot of this comes down to a good Business Analyst skill, experience, but also making sure that the person who has given you the requirements has the accountability to sign it off. In other words if they changes their mind all the time, then you get to a state in the project where you say; right this is the requirements, this is writing down in black in white, sign that off.*” [BPSM1]

Changing requirements

However, some requirements are constantly changing, and therefore, these requirements are hard to validate. Once when the requirement is elicited and validated it can be that the requirement is already changed. According to several requirement experts, this concern should be addressed during the requirements gathering. During the gathering of the requirements the requirement engineer must validate with the source if requirements that are gathered can change over time [ICPM1,ARBA1]. If these requirements change over time, and decisions about these requirements are made later on, then, according to the Business Partner Service Manager, this will be considered in the Agile development where in every iteration the requirement will be verified if it is changed or not;

“*really you need an IT team that is kind of prepare for that and that is what Agile does, because you can have a set of a kind of changes that roll in through each iteration*” [BPSM1]

An example of this occurred during the observation of the iClose project. In several meetings, the wording issue was addressed because the responsible stakeholder mentioned that the wordings will change over time. Therefore, the Project Manager validated the wording requirements in several iterations to make sure that they are still correct and flexible enough for future changes;

“*Like now, we are trying to build the system and we are using the current wording policy. We know that within three weeks they will change the wording policy and we know that they do it again three months after that*.” [ICPM1]

Conclusion

In summary, the answer of the question that concerns the validation of requirements “SRQ 2.11: How are requirements tested/validated?” is that in market-facing projects two major techniques are used namely the validation by the use of multiple sources and the sign off principle. By the use of multiple sources, the requirements engineer consults multiple sources to elicit the requirements. These sources include; experts, documentation and existing systems. In terms of the sign off principle, the requirement engineer asked the stakeholder to sign off the requirements. When the stakeholder signed off the requirement, he agreed that these are the final requirements and that these requirements can be passed through to the developer in order to be realized.

However, some requirements are constantly changing and these are hard to validate if they are not gathered properly. In Chubb, the requirement engineers asked the stakeholders if the requirements, that they suggest, will change over time. If this is the case, then the requirement engineer validates these requirements during the iterations in order to make sure that the concerning requirements are still correct and flexible enough to address changes.

3.2.5 Current situation of requirements prioritization in MFT projects

Unfortunately, often not all elicited requirements can be realized in the system under development due to time and cost constraints. Therefore, requirements have to be prioritized to determine which requirements are crucial to be in the system and which are not. At Chubb, this is also done in the requirements engineering process in market-facing technology projects. In this process, the requirement engineers are mainly using two aspects to determine which requirements are crucial and which are not. These two aspects are; scope of the project and return of investment.

Scope

One of the initial steps in setting up a project in Chubb is determining the scope of the project, wherein the goal and the basic elements are defined (see Appendix V). This scope is more or less the red line through the project and is used to determine which requirements have to be in the system and which have to be left out. When a requirement is elicited, the requirement engineer analyzes it to verify if the requirement is inline with the scope of the project [ICPM1, ARBA1, ITM1, ARPM1]. If the requirement is inline with the scope of the project, then it will be taken into account. If not, the requirement will be dropped out of the project. The Business Analyst in the ARCH project gave an example of this during the interview;

"*You can delay on risk analysis, but that is for your information, (but rating and etc. is helping to calculate the premium) and thus that is the document that goes outside. You need to focus on the goal of the project. I just give you the very big example*" [ARBA1]

In this statement, she mentioned that rating is one of the elements of the scope of the project and the risk analysis is not. Therefore, the risk analysis document will not directly be taken into consideration in the project. This is also inline with what the Project Manager of the iClose project stated about how he prioritizes requirements;

"*In terms of prioritizing; you can go through a process of saying what the objectives are and what is the bare of requirements to fulfil that. What are the dependencies across that to work and how to sequence them and how do we actually prioritize the development and the delivery of it?*" [ICPM1]

In this statement, he mentioned that he is looking at the objectives of the project and based on that he determines if the requirements fulfil these objectives. These objectives are defined in the scope of the project. If the requirements do not fulfil these objectives, they would not be taken into consideration in the project.

Return on investment

Besides the scope of the project, the majority of the requirements engineers also suggested that they take the return on investment into consideration to determine the priority of the requirements [ARBA1, ARPM1, CFO1, ITM1, BPSM1]. If the considered requirement does not support the return on investment, it will gain a low priority or it will be dropped out of the project. As what the Project Manager of the ARCH project stated;

"*It is the same thing as, if you leave something out; you have to know what the impact is. What is the impact of the business by leaving it out? What does it cost to leave it outside of the system? That is what drives the priority.*" [ARPM1]

With this statement, she explained how she analyzes the requirements if it would not be in the system. From that perspective, she tries to find the return on investment and decides to include or exclude the requirements.

During the observation period in the iClose project, 31% of the observations addressed prioritization. In the meetings where requirements were prioritized, in 40% of the cases the accent of the prioritization was on the scope of the project and 60% on the return on investment (Appendix W). Nonetheless, most of the time both mechanisms were used but with one as the main prioritization mechanism and the other as a supporting prioritization mechanism. Figure 14 depicts the percentage of the meetings where in prioritization was done and in Figure 15 the percentage of the main prioritization mechanisms are depicted.

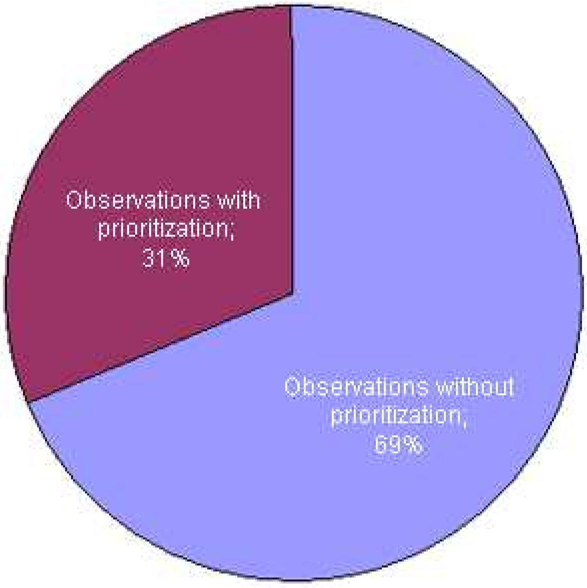


Figure 14: Prioritization during meetings

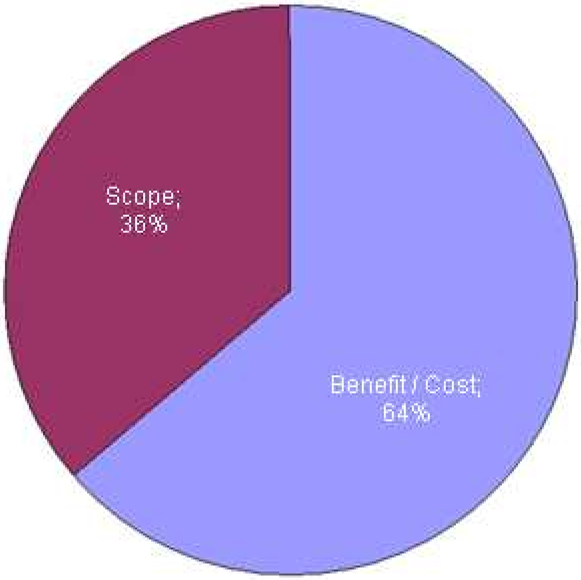


Figure 15: Applied prioritization mechanisms

Other aspects
However, return on investment and the scope of the project are not the only aspects that are considered in prioritizing requirements in market-facing projects. In the statement of the Project Manager of the ARCH project above, she explains that she is also trying to analyze the impact of the requirement in the business processes. Besides this, the IT Manager mentioned another technique that is used for prioritization. He stated that the complexity of the product is also considered as a factor to prioritize the requirement;

"*When you prioritize you looking at things like; is it going to make the product complex? Does one requirement has a greater return on investment?*" [ITM1]

If the requirement increases the complexity of the product, it will have a lower priority or probably dropped out of the project. If the requirement increases simplicity, it will be taken into consideration to realize it.

Requirement prioritization management techniques
Once the priorities of the requirements are clear, they need to be organized and documented. The Project Manager in the iClose project organizes and documents the requirements into three categories, namely; critical requirements that must be in the system, requirements that should be in the system but is not strictly necessary in order to functioning properly and requirements that are not necessary at all but would be nice to have in the system, like requirements on the layout;

"*You can just say; you tell me what you are going to do: look and feel or functionality? They usually choose functionality. And what you than can do is once the systems start working and when you get money coming in we can use some of the money to enhance it so we prioritize the requirements in* *release 1 2 and 3. Critical requirements come into release one. Need to have but not critical comes into release 2 and 3 is all the nice things*." [ICPM1]

However, no other documents or sources suggested this organizing technique. A possible explanation for this could be that the Project Manager is the only person who is organizing this and therefore the existing documents and other sources of evidence are possibly only stored on his personal account. Due to restrictions, it was not possible to retrieve these documents.

In the ARCH project, another technique is used to organize the requirements. According to the Project Manager, once the priorities of the requirements are clear, they are organized in a priority list wherein requirements are prioritized on a scale from one to twenty;

"*No, we have various priority lists. Some of them are saying which insurance products you are going to put in. All the insurance products here in Australia are written and prioritized from one to twenty. And then we have additional functionality they are trying to put into the system, and again that is prioritized based on the business need. We work trough those lists and then it is a priority list that goes into a plan.*" [ARPM1]

Similar to the previous mentioned technique, the Project Manager of the ARCH project is the only source that mentioned this. A possible explanation for this, could be the same reason as the previous mentioned technique, namely that the Project Manager is the only person who manage this and therefore no other sources of evidence exists.

Conclusion
In summary, prioritization (SRQ 2.12) is mainly done based on two aspects namely; Scope of the project and return on investment. For example; when a requirement deliver a high return on investment and is inline with the goal and scope of the project, it will gain a high priority. When a requirement has a low return on investment and is not inline with the goal and scope of the project, it will gain a low priority or possibly dropped out of the project. There are also other aspects that are used to prioritize, but these aspects are mostly used in combination with the scope of the project and / or the return on investment. These aspects are; the impact of the requirement in the business processes and the complexity of it.

The organization and documentation of these prioritized requirements are done in different ways. In the iClose project, the organization and documentation is done based on three categories. After prioritizing, the requirements are categorized into; 1) requirements that are critical and must be in the system. 2) Requirements that should be in the system but are not critical. 3) Requirements that would be nice to have it in the system. In the ARCH project, another method is used. In this project, the requirements are listed and classified on a scale from one to twenty.

3.3 Conclusion – Answer to Research Question 2

In conclusion, the second research question, namely: "How does Chubb currently gather, negotiate, validate and prioritize the requirements of projects like the iClose project" of this thesis is answered.

At the moment, the requirements engineering for market-facing projects within Chubb is done based on an iterative approach were requirement concerns are assessed multiple times in order to form the right requirements. In this approach they demonstrate prototypes wherein they negotiate prioritize and validate the requirements. By this, the requirement engineering and development is done simultaneously and therefore, the requirements gathering, negotiation, validation and prioritization are also done simultaneously. During this process, mainly one overall document is used wherein all the requirements, business logic and scope and goals are defined. This document is called a business specification document.

To gather requirements, the requirements engineers have to choose stakeholders to gain information. Therefore, the requirements engineers consider three aspects to select stakeholders, these aspects are; expertise, sponsorship and delivery responsibilities. Once the stakeholders are clear and the scope of the project is defined, the requirement engineer starts gathering the requirements and therefore he/she uses the following techniques; market-research, existing systems, workshops, case scenarios and face-to-face meetings (interviews).

During the negotiation process, the stakeholders have to understand each others concerns. To gain an overall understanding, the requirement engineer uses techniques. These techniques are; visualisation (process maps and images), face-to-face meetings to explain the system and collaboration systems to discuss the status quo of the project and to register the communication. If there are conflicting requirements, the requirement engineer tries to reflect the conflicting concerns to the scope of the project to determine if it is inline with it. Besides this, he or she also validate the quality of the requirement that conflicts, by making everyone aware of the essence of the conflict and based on their opinions the requirement engineer will make a decision. Further on, the requirement engineer also measures the conflicting requirement on how much time it will take to realize it and how much it will cost. Based on this, he or she will make a decision to extend the time frame or to increase the budget of the project in order to solve the conflict.

After the requirements are gathered, they have to be validated. The validation of the requirements in market-facing projects is mainly done due to two techniques, namely; the use multiple sources to validate the requirements and a sign off technique. The sign off technique is used to make sure that the stakeholder, who is responsible of specific requirement concerns, agreed on the formulated requirements from those concerns.

Finally, the prioritization of the requirements is done based on the scope of the project and the return on investment. When a requirement delivers a high return on investment and it is in the scope of the project, the requirement will gain a high priority. When the requirement delivers a low return on investment and is not inline with the scope, it will gain a low priority or it will be dropped out of the project. The next chapter describes the assessment of these findings from the case study by the findings from the literature study and based on that recommendations are drawn.

4. Recommendations on the current situation

Findings from the case study and literature study are analyzed. Based on this analysis, recommendations for market-facing technology projects within Chubb are drawn and described in this section. The first section of this chapter describes the recommendations on the overall process, the second section describes recommendations on the gathering process, the third section on the negotiation process, the fourth section on the validation process, the fifth section on the prioritization process and in the final section an answer is given on the third research question by summarizing the findings from the previous sections.

4.1 Recommendations on the overall requirement engineering process

As mentioned in Section 1.2, a market-facing enterprise is an organization that is sensitive to the needs of the market and customers. These needs are constantly changing. In order to accommodate these needs with a system, the development has to be done rapidly and the highest satisfactions of the market and customer have to be achieved. From the findings of the literature study in Section 2.2, the most suitable development approach for this issue is an AGILE approach. In this approach small iterations are used wherein stakeholders have to work intensively with each other to achieve a high satisfaction on the needs. As the findings from the case study suggested, this approach is already used in the both projects. They use small iterations wherein multiple stakeholders are sitting together to elicit requirements and review current requirements. A drawback of this approach is that there is a chance that important requirements may go unrecognized or recognized too late and got not implemented [41]. However, during the observations in the iClose project there were no important requirements recognized too late. Due to the fact that the project was still in development after the observations, it could not be determined if requirements were not recognized at all.

An approach that can be used to address this gap, is the approach proposed by Sen and Hemachandra in [1]. In this approach, the goals of the stakeholders need to be extracted. Once these goals are extracted, they need to be decomposed from high level goals to lower level or sub goals. From this, the goals can be reviewed in order to see if any goals are missing. The approach revolves around 1 to 4 week cycles that are called Sprints (iteration). Each Sprint consists of three main activities, namely pre-sprint, the sprint it self and post-sprint which refer to the activities before, during and after the elicitation of goals. The information inputs and outputs of each activity are presented in Table 7.

Table 7: Inputs and outputs of activities [1]

Activity		Inputs	Outputs
Pre-Sprint	Examine	▸ Interview Transcripts ▸ Corporate Policies ▸ List of Requirements ▸ Work Flow Diagram ▸ Mission Statement	■ Initial List of Goals
	Count	▸ Stakeholder participating ▸ Initial List of Goals	■ Total number of Stakeholders ■ Total number of Initial Goals
	Data entry	▸ Total number of Stakeholders ▸ Total number of Initial Goals ▸ Name of Initial Goals	■ Software in Execution
	Distribute	▸ Initial List Of Goals to stakeholders	■ Goal Preference Model
	Data Entry	▸ Priority Value of a goal	■ First Compilation Table
Sprint 1…N	● Develop	▸ First Compilation Table ▸ Activity Card Computer Output-1 to Activity Cards Compiler Output-(N-1)	■ Activity Card
	● Wrap		■ Activity card in tabular format
	● Review		■ Activity Card Compiler Output-1 to Activity Card Compiler Output-N
	● Adjust		
Post-Sprint		Activity Card Compiler Output-N	Software Requirements Document

During the pre-sprint activity, information is gathered from the stakeholders and formulated into goals. These goals are presented into a first compilation table, shown in Table 8.

Table 8: Format of a First Compilation Table [1]

Name of Initial Goals (without predecessor)	**Priorities assigned by stakeholder** (Sh-1 to Sh-M)				**Priority Of Initial Goals by Sorting in Ascending Order of T.PV** $T.PD=\sum_{PV=1}^{PV=M}$
	Sh-1	**Sh-2**	**.....**	**Sh-M**	
A0	PV_{11}	PV_{12}		PV_{1m}	$T.PV_{a0}$
A1	PV_{21}	PV_{22}		PV_{2m}	$T.PV_{a1}$
A2	PV_{31}	PV_{32}		PV_{3m}	$T.PV_{a2}$
.. ..	.. ..	.. ..	.. ..	.. ..	
An	PV_{n1}	PV_{n2}		PV_{nm}	$T.PV_{ax}$

This table is used in the sprint activity to refine the goals including identifying sub goals and responsibilities. Then these goals are wrapped up into a goal tree and reviewed. In the review process, each goal is separately discussed (negotiated, validated and prioritized) among all the stakeholders and finally adjustments are made based on their feedback. In the post-sprint activity, the requirement engineers document the specifications based on the identified goals and sub goals [1]. However, Sen and Hemachandra only validated their approach on the correctness of the goals and did not verify the completeness of the goals.

4.2 Recommendations on the requirements gathering process

One of the techniques found in the case study to elicit requirements is conducting face-to-face meetings with stakeholders that are experts of certain business processes. These face-to-face meetings are similar to what is suggested by the literature as “interviewing experts”. This technique seems to be well suitable for requirements elicitation in new domains, like market-facing technology within Chubb. Therefore, these meetings can be more elaborated to improve the overall requirements engineering process. For example cognitive tools can be used. If experts, that are involved in face-to-face meetings, are assessed on their cognitive profiles they would be able to better understand and prioritize the requirements [2]. Carod and Cechich proposed a method that use elicitation techniques based on the characteristics of the stakeholders in [2]. The characteristics wherein stakeholders could be classified are; sensing, intuitive, visual, verbal, active, reflective, sequential and global. Elicitation techniques that can be used based on the classification are; interviewing, questionnaires (in combination with other technique) and (visual) models. In order to assess the stakeholders, the Index of Learning Styles should be used (ILS) in form of a questionnaire [58]. Based on the result of ILS questionnaire, the elicitation technique can be chosen wherein the stakeholder feels the most comfortable with. Findings from their experiment demonstrated that 72,7% of the conflicts appeared when software requirements specifications were not inline with the preferences of the stakeholders. However, this was only tested in a controlled environment and does not suggest that this would also be the case in practice [2].

Besides this, also case scenarios can be used. Currently, this is already used in one of the projects at Chubb. By the use of case scenarios, the stakeholders would be able to understand and to express the environment wherein the new system is going to work. Through this, the requirement engineers can capture the system's purpose and the reason about whether a given design will meet that purpose [11]. Case scenarios are cases wherein the goals are highlighted due to the appearance and behavior of the (new) system. These cases explains what people are trying to do with the system, what procedures are adopted and carried out by the system and the interpretations are by the people about the system [59]

Another technique that could be used to speed up the current requirements validation and prioritization process in market-facing projects is to use standard templates on how a requirement should be stated. Fruhling et al. found in [20], that preparing a requirements' template and suggesting how requirements should be stated were important improvements to speed up the validation process. For example, a statement could be something like "The system should…." [20].

4.3 Recommendations on the requirements negotiation process

Wu et al. stated in [40] that during the negotiation, stakeholders have to collaboratively and incrementally find out what has to be build. During the case study, in both projects, stakeholders work together to determine what the requirements are by discussing requirement concerns multiple times. This means that in both projects the stakeholders are collaboratively and incrementally working on the requirements, which is inline with the statement of Wu et al. mentioned earlier. Wu et al. mentioned that the activities in this process are sharing knowledge, understanding each other, resolving conflict of interest, co-authoring software requirements [40]. In order to enhance these activities, a collaboration system can be used. During the case study it turned out that this is partly done. In the ARCH project a collaboration system is used for requirements engineering but no such system is used in the iClose project. To enhance these activities, the system that is proposed by Wu et al. in [40] could be used. This system is based on the win-win equilibrium theory which links win conditions, issues, options and agreements. Based on these aspects it creates a win-win equilibrium where all the win conditions are covered by agreements and all issues are resolved by options that are covered by agreements and it shows how this is reached. They collected data from projects over a two year period wherein this system is used. Findings from this research showed improvements in cost effectiveness, user satisfaction, and in combination with complementary practices, such as front-end teambuilding, site-visits, mutual learning and prototyping helped the project teams to achieve better outcomes. A drawback of collaboration systems is that non-experts could face some major challenges in understanding these systems [40].

Besides this, also consistency is important during the negotiation process. Especially in the development of systems that involves the knowledge from business experts. Often this knowledge has to be interpreted by multiple stakeholders and this can easily be misunderstood by them. In terms of the market-facing projects in Chubb it could be that some stakeholders do not fully understand the requirements or misinterpreted them. For example as the Project Manager of the ARCH project stated; that sometimes the developers do not fully understand the insurance business. Therefore, the developers can easily misinterpret the requirements from the business. In order to address this gap Jiang and Yang proposed a model in [38] that can be used to elicited requirements based on ontology.

However, this model has its limitations for the use in market-facing projects, it only addresses performance requirements and its purpose is to elicited requirements from customers. Nonetheless, the idea to use ontology for requirement engineering can contribute to reduce the issue of misinterpretations among the stakeholders. In an ontology, explicit formal descriptions are given to concepts in a particular domain [38]. By using these formal descriptions of concepts, everyone knows what the concepts mean, which reduces the gap of misunderstanding these concepts. For example in the iClose project, the concept “bind” can be described as: the process of transforming <computing> a quote <input> into a policy <output>.

4.4 Recommendations on the requirements validation process

During the case study it turned out that requirements are validated by multiple sources. Once the requirements are elicited, the requirement engineer verifies the requirements by multiple stakeholders, documentation and by information extracted from existing systems. However, this can cause some issues because stakeholders often formulate brief, ambiguous and conflicting requirements which are hard to validate. To address this issue, the proposed spiral model of Ahmad [3] can be used (Figure 6).

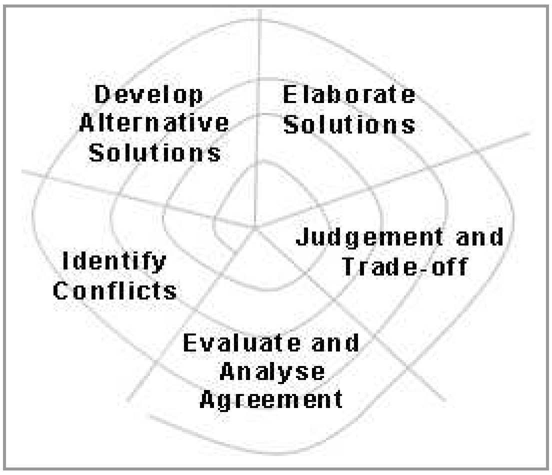


Figure 6: Requirements negotiation process

The first step in this model is to identify conflicts in the list of candidate requirements that are elicited by the stakeholders. Then alternative solutions that address these conflicts have to be developed. After that, the solutions have to be elaborated by the stakeholders to gain a better understanding. Next, they judge and trade off the requirements based on the defined criteria in the previous step. If all the conflicts are not resolved yet then it has to go trough the next iteration, starting by the first step until all conflicts are resolved and requirements are validated by all stakeholders.

Another method that is used to validate requirements in market-facing projects in Chubb is to sign off the requirements by the stakeholders once these are elicited and negotiated by other stakeholders. The stakeholder who has the responsibility of the requirement will need to sign off the final requirements. When he or she signed off the requirements, he or she agrees that these requirements are final and right. Often these stakeholders will do a final verification before they sign off. In order to improve this process the requirements engineer can use cognitive tools to determine the cognitive profiles of the stakeholders. By determining these profiles, the requirement engineer can adjust the presentation of the requirements. This will increase the understanding of the requirements and goals by the specific stakeholder which will speed up the sign off process. As mentioned earlier in paragraph 4.1.2, Carod and Cechich proposed a method that use elicitation techniques based on the characteristics of the stakeholders in [2]. Parts of this method can be used to improve the sign off process. Stakeholders can be classified in the characteristics (sensing, intuitive, visual, verbal, active, reflective, sequential and global) due to the questionnaire on [58]. Once the characteristics of the stakeholders are know, the requirements can be presented in a way that is most comfortable for them (visual or non-visual, sequential or global, etc.).

4.5 Recommendations on the requirements prioritization process

From the case study it turned out that requirements are prioritized through considering the scope of the project, the business value (return on investment), complexity and impact, where the scope and the business value are the dominant aspects. These findings are to a certain extent inline with the findings from Racheva et al. [4]. As mentioned earlier in paragraph 2.5.4, they did a case study on requirements prioritization methods in agile development approaches. From this case study they found that the aspects that are considered when making decisions on requirement priorities are: business value, effort estimation/size measurement, learning experience, input from the developers and external change. Further on, they also found that the project context has a significant impact on the prioritization criteria and that the business value is a dominating requirements criterion. In reflection of these findings with the findings from the case study at Chubb, in both case studies it turned out that "business value" is a dominating factor and some other aspects have similarities with each other. The aspect "complexity" found in the case study at Chubb is similar to the "effort estimation/size measurement" found by Racheva et al. [4] and the "impact" aspect found in the case study at Chubb is similar to the "external change" aspect found by Racheva et al. [4]. However, Racheva et al. [4] did not explicitly mentioned anything about the scope or goal of the project as a prioritization criteria. They mentioned a criterion called "negative value" in which they assessed the requirements on the necessity to support the main usage scenario. This is to some extend similar to the assessment of scope of the project found in the case study. Although the model that Racheva et al. yielded is a descriptive model, it can be used to optimize the prioritization process in market-facing technology projects at Chubb. Due to the fact that the model describes the prioritization process, it can be used as a guideline in order to make sure that all aspects are considered and the overall process is structurized which can reduce time. This model is depicted in Figure 16.

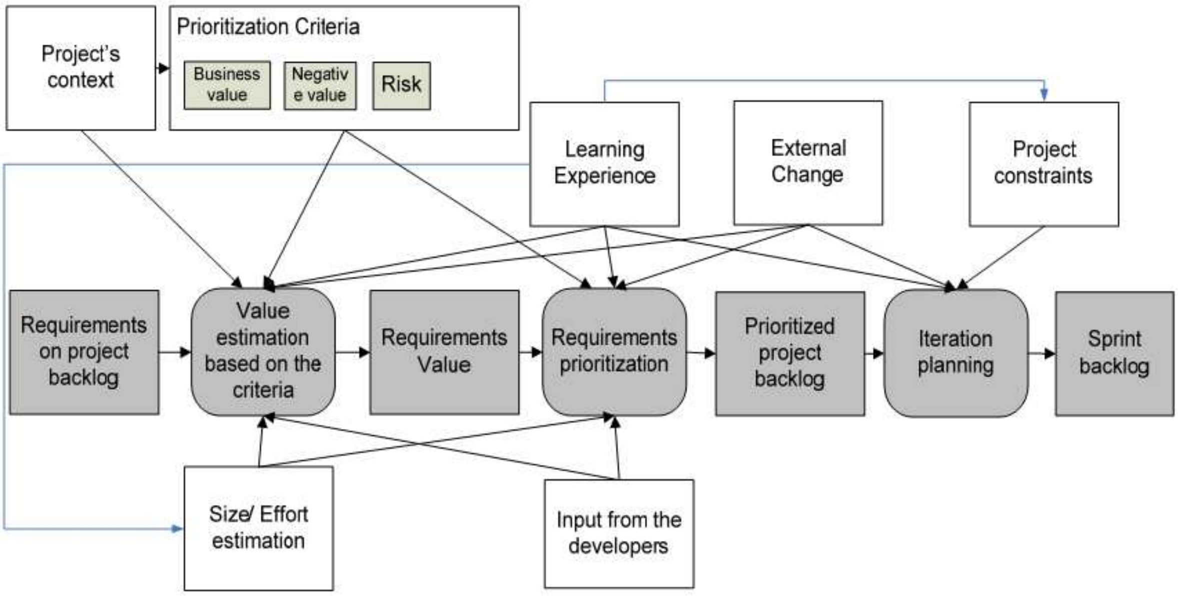


Figure 16: Conceptual model of [4]

The grey boxes in the model are the actual prioritization process and the white boxes around it are the aspects that influence the prioritization activities. The first activity is estimating the value of the requirements based on the criteria. In this activity all the aspects are considered. After estimating the value, the requirements get prioritized. Also in this activity all aspects are considered. Once the requirements are prioritized, the next iteration of the agile approach can be planned wherein the aspects "learning experience", "external change" and "project constraints" are considered. However, in this process the human subjectivity can varied the prioritization because people try to prioritize the requirements independently. To address this gap, the requirement engineer can also use the cumulative voting method described by Leffingwell and Widrig [5]. In this method, each stakeholder can assign in total 100 points to the requirements. These points can be distributed in any way that the stakeholder desires, where requirement with the highest amount is the most important and the lowest amount is the least important. After the stakeholders divided their point among the requirements, the results need to be combined and from that the relative preferences of the requirements can be extracted in a systematic and controlled manner.

4.6 Conclusion - Answer to Research Question 3

In this section, aspects that can be used to improve the requirement engineering process in market-facing projects within Chubb are analyzed and described based on the findings from the case study and the literature study. In relation to the research question 3: “What can be improved in the way Chubb gathers, negotiates, validates and prioritize requirements reviewing the current practice and comparing against scientific literature?”, these aspects answers this research question. In summery, the answer of this research question is;

Improvements on the overall approach

- Current approach can by improved by using the approach proposed by Sen and Hemachandra [1] to reduce the chance that important requirements are unrecognized or are recognized too late. However, this approach is not verified on the completeness of the goals.

Improvements in the gathering process

- Face-to-face meetings can be improved by the use of cognitive tools to gain a better understanding by the business expert about the system and the requirement concerns. Therefore, the method proposed by Carod and Cechich in [2] can be used. In this method, the characteristics of the stakeholders are determined. Based on the characteristics of the stakeholder, a elicitation technique can be chosen wherein the stakeholder feels the most comfortable with. However, this method is only tested in controlled environments.
- Case scenarios can be used in market-facing technology projects to capture the systems’ purpose and the reason about whether a given design will meet that purpose [11].
- Standard templates to state requirements could be used in the gathering process to speed up the requirements validation and prioritization process.

Improvements in the negotiation process

- Negotiation process can be improved by using a collaboration system in each market-facing project. For example the system proposed by Wu et al. in [40]. In [40] they stated that through this system they achieved improvements in cost effectiveness, user satisfaction and outcomes of the project. However, non-experts could face some challenges in understanding this system [40].

- Ontology can be used to reduce the issue of misinterpretations of requirements by stakeholders. In an ontology, explicit formal descriptions are given of concepts in a particular domain [38]. With these descriptions, the stakeholders will gain a better understanding of what is meant by specific concepts in the project.

Improvements in the validation process

- Requirements formulated by stakeholders can be brief, ambiguous and conflicting. By the use of multiple sources, these requirements can be hard to validate. In order to address this issue, the model proposed by Ahmad in [3] can be used. This model consists of multiple iterations wherein requirements are negotiated and validated by all stakeholders.
- By signing off the requirements, the requirement engineer could use the cognitive method proposed Carod and Cechich in [2] too. By presenting the requirements in a way where in the specific stakeholder feels most comfortable with, he of she will better understand what he or she need to sign off on. This reduces effort and time of understanding the requirements by the specific stakeholder.

Improvements in the prioritization process

- To reduce time in the prioritization process, the model of Racheva et al. in [4] can be used as a guideline. By using this model, the requirement engineer makes sure that he or she considers the important aspects in the prioritization process which reduces the chance of wrongly prioritized requirements.

- Sometimes requirements are prioritized by the requirement engineer independently and because of that, human subjectivity can disturb a proper prioritization. To address this issue, the cumulative voting method proposed by Leffingwell and Widrig in [5] can be used, where all stakeholders vote on the requirements to prioritize them. However, when using this technique, the requirement engineer needs to be aware of that all votes are not equal. For example the vote of the developer can have a higher influence then that of a business expert.

5. Recommended situation

In this chapter the Research Question 4: "What would be a better way for Chubb to carry out requirements engineering for market-facing projects?" is answered. The improved approach that Chubb can use to carry out requirement engineering in market-facing project is based on the findings of the case study in Section 3.2 and the recommendations made in Chapter 4. Elements from both sections are adopted in a framework that is given in Table 9. This framework is an elaboration on the framework of Sen and Hemachandran in [1]. As mentioned earlier in Section 2.2, this framework is an Agile framework that consist of small iterations. These iterations are called sprint and each revolves around a 1 to 4 week cycle. In each sprint there are three processes, namely pre-sprint, sprint and post-sprint that refers to the activities before, during and after elicitations of goals and requirements.

Pre-sprint

In the pre-sprint, initial information is examined through different techniques used in market-facing projects within Chubb. These techniques are to some extend elaborated by the suggestions from the literature. The techniques that can be used in this process are;

(i) Market research
(ii) Stakeholder selection based on expertise of product, sponsors and deliverables
(iii) Interviewing based on cognitive profiles
(iv) Existing system analysis
(v) Case scenarios

Market research can be done the gain important information about the current needs of the market. Stakeholders can be selected based on their expertise, sponsorship and responsibilities. Once stakeholders are identified, the stakeholders that have certain expertise can be interviewed to extract information from them. Before the interview, the specific stakeholder can be assessed on their cognitive characteristics and based on that, tools can be used to demonstrate the concerns. Existing systems can be analyzed to extract important information that is required by the business and case scenarios can be made to gain additional information. After the examination, an initial list of goals is produced and counted together with the total number of stakeholders. Then, this information can be transformed in a set of goals and compiled into a first compilation table (given in Table 10). After that, this table is stored in a collaboration system and distributed among the stakeholders. Information that is used in this process consists of; interview transcripts, corporate policies, workflow diagrams, mission statement, existing systems, market information, a default template to state requirements, stakeholders information and the initial list of goals compiled into a first compilation table.

Sprint

This process exists mainly of negotiating, validating and prioritizing the goals and requirements. Once the first compilation table is created and distributed, the goals and requirements in this table will further elaborated in the development, wrap, review and adjustment activities. The stakeholders discuss the initial goals and refine them into sub goals. To facilitate this discussion, an ontology that is described in paragraph 4.1.3 can be used to reduce misunderstandings. Notes can be stored in the collaboration system in order to understand later on why decisions are made. In terms of conflicting goals or requirements, the spiral model of [3] can be used to iteratively and collaboratively solve the conflicts. The goals can be validated through multiple sources and signed off (by presenting the requirements based on the cognitive characteristics) by the specific stakeholders that are responsible for the goals and requirements. Then every stakeholder can prioritize the goals and requirements individually by voting based on business value, negative value, size/effort estimation, input from developers, external change, learning experience, and project constraints. After this, the requirement engineer can merge the results and determine the overall priorities of the goals. Each main goal with their sub-goals is then transformed into an activity card (depicted in Figure 17). All these activities can be executed simultaneously.

Table 10: Format of a First Compilation Table [1]

Name of Initial Goals (without predecessor)	**Priorities assigned by stakeholder** (Sh-1 to Sh-M)				**Priority Of Initial Goals by Sorting in Ascending Order of T.PV** $T.PD=\sum_{PV=1}^{PV=M}$
	Sh-1	**Sh-2**	**.....**	**Sh-M**	
A0	PV*11*	PV*12*		PV*1m*	T.PV*a0*
A1	PV*21*	PV*22*		PV*2m*	T.PV*a1*
A2	PV*31*	PV*32*		PV*3m*	T.PV*a2*
.. ..	.. ..	.. ..	.. ..	.. ..	
An	PV*n1*	PV*n2*		PV*nm*	T.PV*ax*

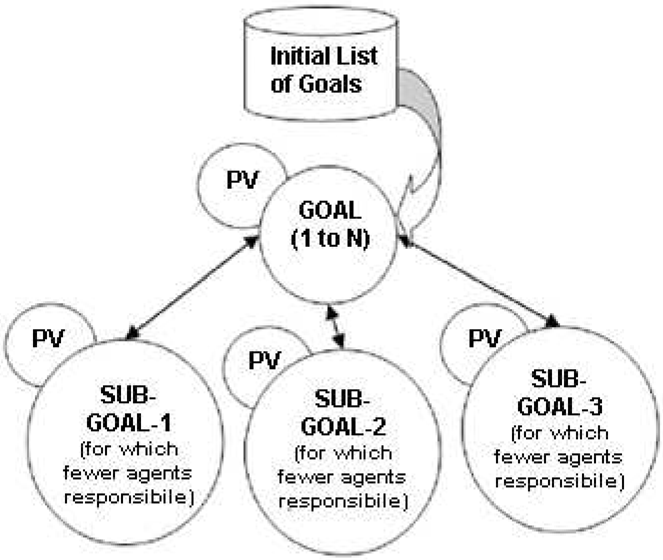

Initial List of Goals
PV
GOAL (1 to N)
PV
SUB-GOAL-1 (for which fewer agents responsibile)
PV
SUB-GOAL-2 (for which fewer agents responsibile)
PV
SUB-GOAL-3 (for which fewer agents responsibile)

Figure 17: Goal activity card

Post-sprint
In this process, all the activity compiler cards are transformed into requirements and documented in the overall software requirements documentation that is already used within Chubb for market-facing projects. A template of this document is attached in Appendix N.

Table 9: Improved framework for requirements engineering based on [1]

Activity		Inputs	Outputs	Techniques
Pre-sprint	Examine	Interview transcripts Corporate policies Workflow diagrams Mission statement Existing systems Market information Requirement template	Initial list of goals	Interviewing based on cognitive profiles Market research Existing system analysis Case scenarios Stakeholder selection based on expertise of product, sponsors and deliverables
	Count	Stakeholders participating Initial list of goals	Total number of stakeholders Total number of initial goals	
	Data Entry	Total number of stakeholders Total number of initial goals Name of initial goals		Collaboration system
	Distribute	Initial list of goals to stakeholders	First compilation table	
Sprint	Develop Wrap Review Adjust	First compilation table Activity card compiler output -1 to activity card compiler output N-1	Activity card Activity card compiler output-1 to activity card compiler output-N Identified changing requirements	Ontology Collaboration system Spiral model of [24] for conflicting requirements; identify conflicts, alternatives, judge and trade off Multiple sources Requirement value based on business value, negative value, size/effort estimation, input from developers, external change, learning experience, project constraints Cumulative voting Sign off by the use of cognitive profiles
Post-sprint		Activity card compiler output	Software requirements documentation	Documenting

In summary, this framework addresses all the concerns that are mentioned in Section 4.1. It uses the overall approach proposed by Sen and Hemachandran in [1] to ;

- Develop the requirements rapidly, due to 1 to 4 weeks sprints
- Achieve the highest satisfaction of the market and costumers' need by actively involving people that know the market
- Addresses the issue that important requirements may go unrecognized or recognized too late due to multiple information source, the actively involvement of all the stakeholders in the discussion about requirements and multiple validation techniques.

As mentioned before, in the pre-sprint, initial requirements are gathered. One of the gathering techniques used in this process is interviewing specific stakeholders. During these interviews, the cognitive tool of Carod and Cechich in [2] can be used to improve the understanding of the requirements and goals by the stakeholders. Further on, the use of templates are addressed in the framework to speed up the validation process by simplifying the formulation of the requirements and case scenario's are used to create a better understanding of the goals and constraints of the system. In the sprint process the following techniques are taken into account;

- A collaboration system to enhance knowledge sharing, understanding of each other and resolving conflicts of interests.
- The use of an ontology in order to keep consistency during the overall development
- The spiral model of [3] to resolve brief, ambiguous and conflicting requirements
- Multiple sources, to validate the requirements
- Requirement value assignment based on the descriptive model of [4] and the prioritization aspect from the case study to make sure that all priority aspects are considered
- Cumulative voting for prioritizing the requirement to avoid human subjectivity
- Sign off based on cognitive profiles to enhance the understanding of the requirements by the stakeholders that have to sign off.

Final, in the post-sprint process, the business specification template found in the case study is used to document the requirements.

6. Evaluation of the recommended situation

This chapter describes how the framework is validated based on the guidelines given by Winbladh et al. in [60]. Further on, it explains how the framework can be evaluated in practise based on metrics and the constraints of the framework are described. In the final section conclusions are drawn and an answer is given on Research Question 5.

6.1 Validation of the proposed framework

The study in [60] showed that many companies do not use textbook requirements engineering processes. According to [60] there are many contributing factors why these approaches are neglected for example through the stakeholders' preferences, habits and the application domain. In order to enhance the adaptability of formal requirement engineering approaches, Winbladh et al. conducted a survey in [6]. In this survey, they used a set of questions that are based on pre-defined characteristics of requirements engineering and needed to be answered by the participants who have experience in software development. They used standard statistics as metrics to evaluate the results of the research. Findings from this survey suggested that the guidelines described in Table 11 should be used to enhance the adaptability of requirements engineering approaches [6].

Table 11: Guidelines to enhance the adaptability of requirements engineering approaches [6]

Guideline 1	**Strive for narrative structures** Requirements with narrative structures, e.g. use cases and stories, describe operational requirements and can capture informal conversations with customers. Narrative requirements are typically easy to understand, could be the starting point for test scenarios, and message sequence charts.
Guideline 2	**Strive for domain descriptions** Consistently referencing defined terms in a domain model can decrease ambiguity. Cleary defined terms can also increase understand-ability. A domain description aids testability as abstract test data can be selected from the model. A domain model can also be a useful starting point to module design.
Guideline 3	**Strive for modularity** Keeping narrative and domain models separate and decomposing these into related functionalities and concepts, is a useful strategy to provide higher levels of maintainability and traceability
Guideline 4	**Strive for relations among requirements** Relations between domain and narrative structures can improve un-ambiguity and understanding of the requirements. Relations can provide guidance for change impact analysis during maintenance. Syntactic and semantic relations could also be the basis for a useful requirements engineering tool
Guideline 5	**Strive for a formal internal model** Providing a formal description of the requirements could increase the level of un-ambiguity and could be used as a basis for automated test support.
Guideline 6	**Strive for a tool support** Tool support could make use of the guidelines above to provide functionality including searching, re-factoring and showing relations. The tool should provide an easy-to-use interface that shields the user from the complexity of internal formal representations.

To evaluate the framework proposed in Section 4.2, these guidelines are used as a metric to determine the usability of it;

Guideline 1 - Strive for narrative structures
In the pre-sprint of the framework, case scenarios and interviews are used to describe the operational requirements that can be formulated into narrative requirements which are easy to understand by the stakeholders.

Guideline 2 - Strive for domain descriptions
In the framework, the use of ontology is considered. By the use of this, specific terms are pre-defined which will increase the understand-ability and decrease the ambiguity among the stakeholders by keeping the terms consistent.

Guideline 3 - Strive for modularity
All decisions and retrieved information are registered in a collaboration system. Because of that, requirements can be traced back to the information on where the requirement is elicit from. This can be from narrative models (case scenarios, interviews, etc.) or domain models (market research, existing systems, etc.). Further on, the proposed collaboration system of [37] in the framework provides the ability to easily maintain requirements through to a Wiki mechanism.

Guideline 4 Strive for relations among requirements
In the sprint process, the requirements are collaboratively engineered based on the information gathered in the pre-sprint process. The information that is used in this process consists of domain and narrative data. This information is combined and registered during the negotiations in the sprint process to indicate syntactic and semantic relations among it that is needed to elicit and maintain requirements. These relations should be used as guidance for change impact analysis.

Guideline 5 Strive for a formal internal model
The Activity card that is used in the proposed framework in Section 4.3 is a formal model to describe requirements that should increase the level of un-ambiguity and could be used as a basis for automated test support.

Guideline 6 Strive for a tool support
The use of a collaboration system in the proposed framework in Section 4.3 provides functionality including searching, re-factoring and showing relations.

However, to measure the real impact of the framework, metrics should be used to gain insights, control and improvements of the requirements engineering process. Therefore, Costello and Lui proposed a set of metrics in [7]. These metrics are presented in Table 12.

Table 12: Metrics for requirements engineering approaches [7]

Metric	**Overview purpose**
Requirements volatility	Indicates changes (additions, deletions, modifications) and reasons for changes to requirements. Provides insight into system maturity and stability. Aids in predicting future requirements, design, and code volatility. Essential in interpreting other metrics.
Requirements traceability	Indicates degree to which development organization maintains accountability for meeting requirements at each stage of life cycle via a requirements traceability matrix. Provides quantitative means for determining whether all required relationships/dependencies are addressed. Assists in exposing incompletely specified, overly specified and complex areas of system. Essential in interpreting other metrics.
Requirements completeness	Indicates completeness of all sections of requirements specifications, whether all allocated higher level requirements are addressed, and the degree of decomposition of allocated higher level requirements.

	Assists in determining readiness to proceed to design.
Requirements defect density	Indicates number of requirement defects that are initially detected during an inspection or walkthrough. Classified by type, critically, and source. Provides early insight into quality, assists in cost/schedule estimation, and indicates effectiveness and extent of inspection/walkthrough process. Useful in predicting product/process volatility. Essential in interpreting other metrics.
Requirements fault density	Indicates number of requirements faults that are initially detected during test execution or posttest analysis. Classified by type, critically, and source. Assists in determining effectiveness of software process and the extent and effectiveness of testing. Useful in predicting product/process volatility and quality. Essential in interpreting other metrics.
Requirements interface consistency	Indicates consistency and completeness of interface information at each level of specification.
Requirements problem report/action item issue	Indicate number of requirement problems detected/issues raised via any process; as such include requirement defects and faults. Characterize problems by source, product, type of problem, finding activity severity, critically, age, and primary reason for closure. Number and status of requirements problem reports indicate quality of requirements engineering and inspection processes. Essential in interpreting other metrics.
Requirements integrated progress	Indicates overall requirements progress. Encompasses measures of volatility, traceability, completeness, defect density, fault density, test coverage, and problem report action/items as appropriate for phase under consideration

The first metric consists of counts of requirements that are changed in each iteration and is classified by the reason of change [7]. When the framework is properly applied, the volatility is probably high in the beginning and decreases over time in the requirements engineering process.

The second metric, documents the set of linkages of requirements in a specification to their origins and descendants due to a requirement traceability matrix [7]. This matrix indicate how well the framework maintain accountability for meeting requirements throughout the requirements specification activities by determining if all required relationships and dependencies are addressed and expose incomplete specified or overly complex areas of the system.

The third metric, quantifies the level of decomposition of higher level requirements allocated to given specifications and indicate the number of “to be completed’ items and the number allocated requirements that trace to items in the lower level specification [7]. If there is a low degree of decomposition, it may mean that;

- The requirement is simple or specifies a very limited function or behavior
- The requirement’s decomposition spans several specifications
- The high level requirement is documented at a too low level of detail
- The analysis of the requirement was superficial and the requirement needs to be decomposed further.

If there is a high degree of decomposition, it may mean that;

- It represents a complex function and should be monitored throughout the life cycle
- The level of requirement itself may be too general and should be spilt into several high-level requirements before going to the next iteration.

In general measuring the degree of decomposition and the number of allocated requirements indicates if all the higher level requirements have been addressed in the lower specifications. If this is the case, then there is a high degree of completeness if not, the requirements are probably incomplete.

The fourth and fifth metric count the number of requirement defects and faults where a fault and a defect differ only in the means by which each is detected. A defect is detected via inspection of the requirements and a fault is detected via testing. In both cases the number of defects/faults found is divided by the size of the item inspected or the size of the item under test to obtain a density measure. Based on this, the framework can be validated among the number of defect or fault requirements. If this is high, it could indicate that the requirements elicitation needs to be improved.

The sixth metric count the number of requirements those are consistent with the design and code in each iteration. This indicates the amount of requirements that are truly realized and therefore are valid requirements.

The seventh metric counts all the problems that occurred during the overall process. This indicates the quality of the requirements engineering.

The final metric involves all the previous mentioned metrics to indicate the overall requirements progress of the framework.

In order to measure the quality of the frameworks, these metrics have to be used in multiple requirement engineering approaches or models that currently are used in market-facing projects within Chubb. Findings from these approaches can be used to benchmark the new framework. Therefore, the proposed framework in Section 4.2 should be implemented in a new market-facing project after or simultaneously with another market-facing project. However, these metrics only measure quantitative data from approaches. In order to evaluate the new framework thoroughly, also qualitative data need to be gathered. Qualitative data can be gained through interviews with stakeholders that are involved in multiple market-facing project including projects wherein the new framework is used. Nevertheless, the fact that every project differs in size and complexity should be taken into consideration during the evaluation.

6.2 Constraints of the proposed framework

The aim of the proposed framework is to improve the requirements engineering process in market-facing projects within Chubb. These projects are in a new domain of technology that is being developed within Chubb. Therefore, most of the knowledge that is needed during these projects are gathered through the business and is not mainly based on existing systems. This implicates that elicitation techniques are used to gather knowledge of the business that is not already addressed in other systems. Because of that, this framework is probably only suitable for new domains within Chubb Australia.

In addition, as the needs of the market are changing rapidly, the market-facing technology has to be developed rapidly too in order to achieve those needs. The proposed framework is addressing this by developing the system iteratively in sprints. Although the proposed framework tries to capture all the goals it could be that some goals still go unrecognized due to time constraints and therefore this framework would probably only suitable for projects that do not address critical business processes.

Another constraint is that the proposed framework is specific designed for Chubb Australia. This framework is elaborated on the existing requirement engineering approaches that are used in market-facing projects within Chubb Australia. The reason that is chosen to elaborate on existing approaches is to enhance the adaptability by the stakeholders. Therefore, implementing this framework in another context could be challenging.

Further on, in the proposed framework the involvement of multiple stakeholders have to consider. For example in the sprint process the requirements and goals have to be discussed and validated by multiple stakeholders in order to enhance the correctness of it. If this is done by one person, there is a chance that the requirements are subjective and do not meet the goal of the project.

Finally, in the proposed framework also the use of a collaboration system is suggested. However, special knowledge is required by stakeholders in order to use a collaboration system properly. Therefore, stakeholders should already be familiar with similar systems or should need some special training on using these systems.

6.3 Conclusion - Answer to Research Question 5

This chapter answers the last research question: "How to evaluate the proposed framework/methods?". In the beginning of this chapter, the guidelines mentioned by [6] are given and reflected to the new framework to measure the adaptability. From this reflection it turned out that every guideline is covered in the framework. However, these guidelines do not suggests any improvements in practice but increase the likelihood of it. In order to measure and evaluate the new framework in practice, the metrics of [7] can be used. These metrics collects quantitative data about the requirements engineering. Nonetheless, only quantitative data is not enough to evaluate the new framework because every project differs in size and complexity. To evaluate the new framework thoroughly, also qualitative data need to be gathered. This can be gathered through interviewing stakeholders that are involved in projects wherein the new and existing framework is used. Therefore, the best moment to put the framework into use is after or simultaneously with another market-facing project that consider the existing method and is measured with the metrics suggested in [7]. Further on, the overall constraints of the framework that should be taken into account are;

(i) This framework is probably only suitable for new domains within Chubb Australia.
(ii) This framework is specifically made for market-facing projects, and deploying it in another context can be challenging.
(iii) This framework is specifically designed on the involvement of multiple stakeholders
(iv) Stakeholders need some expertise to work with the collaboration system mentioned in the framework

7. Discussion

The purpose of this section is to point out the limitations of this research, new research directions and the recommendations for Chubb. In the first section of this chapter, a summery is given where in all the research questions are answered. Based on this, recommendations are made for Chubb in the second section. Unknown artefacts found in the case study where little information is available on are mentioned in the third section. Finally, the limitations and implications for further research are described in the last section.

5.1 Answers to the research questions

The main research question of this thesis is; “How to gather, negotiate, validate and prioritize the requirements of market-facing projects from a complex multi-stakeholder perspective?” and is answered by five sub research questions. These research questions are answered in the previous chapters and summarized here. The first research question is “What methods/approaches for similar projects and companies are suggested in scientific literature?” and is answered based on a scientific literature study. Articles from the scientific field are found through multiple search engines and selected based on pre-defined inclusion and exclusion search criteria.

The overall requirement engineering approaches found during this literature study are mainly based on plan based and agile strategies. The former strategy generally consists of an overall implementation plan and the latter of small iterations where stakeholders have to work intensively with each other to achieve the highest satisfaction. Techniques found in the literature to gather requirements or to improve the gathering process are; grouping diverse stakeholders, interviewing experts, the use of case scenarios, the use of default templates and documenting requirements and decisions. For the negotiation process, the literature suggests to keep consistency by the use of an ontology, to negotiate iteratively to solve conflicting demands, to consider emotional factors to gain an better understanding of the stakeholders and by the stakeholders and to use a collaboration system to improve the accessibility of the requirements and to document discussions. Techniques that are suggested by the literature to validate requirements are; reviewing viewpoints from different stakeholders, making use of cognitive profiles to gain a better understanding by the stakeholders and the use of automated validations mechanisms in case of vast amounts of requirements to speed up the process. Further on, mechanisms that are suggested for the prioritization process are cost and benefits mechanisms and cumulative voting. Stakeholders collaboratively determine what the cost is of implementing each potential requirement and how much value this requirement will gain in cost and benefit mechanisms. With cumulative voting, each stakeholder is given an amount of 100 points that he or she has to use to vote on his or her most important requirements. The requirements that have the most votes are marked as important and requirements that have the least votes are marked as not necessary or dropped out of the project.

The answer of the second question: “How does Chubb currently gather, negotiate, validate and prioritize the requirements in market-facing projects?” is answered based on an exploratory case study on the iClose and the ARCH project within Chubb. Several experts were interviewed, available documentation was analyzed of both projects and direct observations were done in the iClose project. Findings from this study suggested that the overall requirements engineering in market-facing projects is done based on an iterative approach where in prototypes are used to negotiate, validate and prioritize requirements. Further on, the requirements engineering and development of the system are done simultaneously and a business specification document is used to store the requirements and decisions. Stakeholders in the projects are chosen among three aspects, namely based on their expertise, their (financial) sponsorship and delivery responsibilities. Techniques that are used to gather requirements in market-facing projects are;

market-research to determine if and what need there is for market-facing technology, existing systems (competitor systems and legacy systems) to gain usability and best practices, workshops to brainstorm new ideas, case scenarios to retrieve additional requirements and face to face meetings to extract knowledge for the system from business experts. During the negotiation process, the stakeholders have to understand each other to establish an effective discussion. Therefore, Chubb uses visualization tools (process maps and images), face-to-face meetings and a collaboration system to enhance the understanding by the stakeholders. If there are conflicting requirement concerns; the requirement engineer will reflect these concerns to the scope of the project to determine if they are inline with the project. He or she will validate the quality of the conflicting requirement concerns by making everyone aware of the conflict and based on their opinions he or she will make a decision and the requirement engineer will estimate how much time and money it will cost to realize the conflicting requirements. Based on these aspects he will determine whether to drop the requirement or to add the requirement to the development of the system. The validation of requirements in market-facing technology projects is mainly done through two techniques, namely; the use of multiple sources and a sign off technique. In the former technique multiple sources (existing information and experts) are used to check if the requirement is correct. The latter technique is used to make sure that the stakeholder, who is responsible of specific requirement concerns, agreed on the formulated requirements of those concerns. Final, the prioritization of the requirements in market-facing technology projects is mainly done based on the scope of the project and the return on investment of the requirement. When a requirement delivers a high return on investment and it is in the scope of the project, the probability that it will gain a high priority is most likely. When the requirement delivers a low return on investment and is not inline with the scope of the project, the probability that it will gain a low priority is most likely.

The answer of third research question: “What can be improved in the way Chubb gathers, negotiates, validates and prioritizes requirements reviewing the current practice and comparing against scientific literature?” is;
The overall approach that is currently used in market-facing projects within Chubb can be improved by the approach proposed by Sen and Hemachandra in [1]. Their approach is similar to the approach that Chubb currently is using. By using this approach, Chubb will reduce the chance that important requirements are unrecognized or are recognized too late. In the requirements gathering process, the requirement engineers at Chubb could use cognitive tools to gain a better understanding by the business experts about the system and requirement concerns. Further on, case scenarios can be used to capture the systems’ purpose and reason about whether a given design will meet that purpose. Standard templates could be used in the elicitation process to speed up the requirements validation and prioritization process. In the negotiation process, ontology can be used to reduce the issue of misinterpretations of requirements by the stakeholders and a collaboration system should be used in all projects to improve the cost effectiveness, user satisfaction and outcomes of the project. In the validation process, the model proposed by Ahmad in [3] can be used. This model addresses the problem of ambiguous and conflicting requirements by negotiating and validating the requirements in iterations that consists of pre-defined steps. The signing off technique that is used at Chubb to validate the requirements can be improved by using the cognitive method proposed Carod and Cechich in [2]. With this method, requirements are presented to the stakeholders in a way that the specific stakeholders feel most comfortable with. Therefore, the specific stakeholders will better understand what he or she needs to sign off on. In the prioritization process, the model of Racheva et al. in [4] can used as a guideline to make sure that the stakeholders consider all important requirement aspects in the prioritization process. Further on, the cumulative voting method proposed by Leffingwell and Widrig in [5] can also be used to reduce to gap of human subjectivity in the prioritization process.

The fourth research question is “What would be a better way for Chubb to carry out requirements engineering for market-facing projects?” and is answered based on the findings and recommendations described above. A framework is made that answers this question. This framework is presented on page 59 and is an agile framework that consists of small iterations.

The processes that are considered in these iterations are; pre-sprint, sprint and post-sprint. Requirements are first elicited in the pre-sprint processes due to the following techniques;

(i) Market research
(ii) Stakeholder selection based on expertise of product, sponsors and deliverables
(iii) Interviewing based on cognitive profiles
(iv) Existing system analysis
(v) Case scenarios

After that, in the sprint process the requirements are negotiated, validated and prioritized. To facilitate this discussion, ontology can be used to reduce misunderstandings. Notes can be stored in the collaboration system to understand later on why decisions are made. In terms of conflicting goals or requirements, the spiral model of [24] can be used to iteratively and collaboratively solve the conflicts. The goals can be validated through multiple sources and signed off by the specific stakeholders. Then the goals and requirements can be prioritized based on business value, negative value, size/effort estimation, input from developers, external change, learning experience, project constraints. After this, in the post-sprint process, the requirements and goals of the iteration need to be documented in the business specification document.

The fifth research question: “How to evaluate the proposed framework?” is answered by validating the framework against the guidelines given by Winbladh et al. in [6]. Further on, the framework can be evaluated in practise by the metrics that are proposed by Costello and Lui in [7]. However, these metrics produce only quantitative data on the framework. To evaluate the framework thoroughly, also qualitative data should be considered. Therefore interviews need to be conducted with stakeholders that are familiar with the existing approach and the proposed framework in this thesis.

The implementation of this framework should be done after or simultaneously with a project that use the existing approach wherein the metrics of Costello and Lui [7] are used. In this way, data from both frameworks can be evaluated. However, this framework is not applicable in every situation and therefore it has some constraints. These constraints are;

(i) This framework is probably only suitable for new domains within Chubb Australia.
(ii) This framework is specifically made for market-facing projects, and deploying it in another context can be challenging.
(iii) This framework is specifically designed on the involvement of multiple stakeholders
(iv) Stakeholders need some expertise to work with the collaboration system mentioned in the framework

5.2 Recommendations for Chubb

In summery, Chubb could use the framework that is proposed in this thesis to gather, negotiate, validate and prioritize requirements in a structured way. This could reduce time and could increase the outcomes of the requirements engineering process. Next to this, a collaboration system could be used at Chubb in all market-facing projects. With the use of a collaboration system the communication between the stakeholders are logged. Through this, the requirement engineer has the ability to understand why certain decisions are made or still have to be made. This could increase the decision-making process in validating and prioritizing requirements. Another aspect found in this thesis is that the process of solving requirements in market-facing projects within Chubb could be improved by the use of the spiral model that is proposed by Ahmad in [3]. This model, demonstrates how conflicts can collaboratively be solved in pre-defined activities. By using this model, Chubb can reduce the amount meetings and time that is needed for solving these conflicts in the traditional way.

Further on, findings in this thesis also suggests that Chubb could use cognitive profiles during face-to-face meetings and in validating requirements. By the use of these profiles, requirements can be presented to the stakeholders in a way wherein they feel most comfortable with. This could increase the understanding of the requirements and requirement concerns by the stakeholders.

Finally, in this thesis the use of ontology is suggested for market-facing projects within Chubb. In ontology, definitions of terms from the business and IT are given. This increases the consistency in the project and the understanding of the terms by the stakeholders and can contribute in gathering, negotiating, validating and prioritizing requirements.

5.3 Lessons learnt from Chubb

During the scientific literature study, no information was found on the scope of the project in the requirements prioritization process. For example the descriptive model of Racheva et al. in [4] and the cumulative voting of method Leffingwell and Widrig in [5] do not specifically suggest that the scope of the project is one of the aspects where on the stakeholders prioritize the requirements. Although this aspect was not found in the literature, in the case study it turned out that the scope of the project is one of the major aspects that is used in prioritizing requirements in market-facing projects within Chubb. In the field of science, researchers could investigate if the scope of the project is also considered in other projects and what the outcomes are of taken this aspect into consideration in the prioritization process.

Further on, in the found scientific literature, not much attention is given on singing off requirements. From the case study it turned out that signing off the requirements is one of the major techniques that is used to make sure that requirements are validated. Specific stakeholders have to sign off the requirements and by this they have to take the responsibility that the requirements he or she checked are ok. Researchers could investigate what the influences are of using this technique in projects and how comfortable stakeholders are with this technique.

Finally, during the case study it turned out that stakeholders are selected in the project by considering three aspects, namely; sponsorship (people who found the project), expertise (people who bring in certain expertise) and deliverables (people who deliver the final product). By using these aspects, the Project Manager makes sure that he or she involves the right stakeholders in the project and reduces the change of wrongly gathered, negotiated, validated and prioritized requirements. Researchers could investigate what the effects are in projects when considering these aspects in selecting stakeholders.

5.4 Limitations and implications for further research

The goal of this study is to explore how to gather, negotiate, validate and prioritize the requirements of market-facing projects from a complex multi-stakeholder perspective. Due to the fact that the goal of the research is dedicated to Chubb, it has its limitations in generalizing the results for the field of science and in practice in other companies. The explorative case study is only conducted in Chubb and reflected with the findings from the literature study described in chapter 2. To generalize the findings and recommendations for the field of science and other practices, this study also needs to be conducted in other companies that are developing market-facing systems with multiple stakeholders.

Further on, the researcher did an extensive literature study and found articles based on a systematic way to reduce the chance of missing important literature on requirements engineering in market-facing projects from a multi-stakeholder perspective. However, due to the vast amount of literature and time limitations it could be that there are other elements in the scientific literature that can be used to improve the proposed framework and recommendations. For this, further research is required.

Several experts in the case study suggested that developers do not always understand the demands from the business. However, quantitative data from the case study showed that developers do not have intensively contact with requirements engineers among certain requirements concerns. This could mean that developers understand the requirements in the iClose project or that there was extensively contact among requirements concerns between the developer and requirement engineer where in the observer was not aware of. Therefore, further research is required to determine if developers understand the business and their requirements.

According to several experts, the selection stakeholders for a project are based on three aspects (sponsorship, expertise and deliverables). However, quantitative data showed that these three aspects are presented in the project but it did not prove that these aspects are used for selecting stakeholders. Knowing how stakeholders are selected can contribute to the understanding of the requirements by the stakeholders in market-facing projects. Therefore, further research is required on how stakeholders are selected in market-facing projects.

The framework that is proposed in this thesis is based on the existing method in Chubb and the framework that is proposed by Sen & Hemachandra [1]. With their framework they try to address the gap of that requirements may go unrecognized or recognized too late. However, they only tested their framework on the correctness of requirements and not yet on the completeness. Therefore, further research is required to investigate the completeness of requirements in the proposed framework.

Finally, the cognitive method of Carod and Chechic [2] that is included in the proposed framework to gain a better understanding of the requirements by the stakeholders based on cognitive profiles is only tested in a controlled environments. Because of this, further research is required to test this method in practice.

References

1. A.M. Sen and K. Hemachandran, “Elicitation of goals in requirements engineering using agile methods,” *Proc. Proceedings - International Computer Software and Applications Conference*, 2011, pp. 263-268.

2. N.M. Carod and A. Cechich, “Cognitive-driven requirements prioritization: A case study,” *Proc. Proceedings of the 9th IEEE International Conference on Cognitive Informatics, ICCI 2010*, 2010, pp. 75-82.

3. S. Ahmad, “Negotiation in the requirements elicitation and analysis process,” *Proc. Proceedings of the Australian Software Engineering Conference, ASWEC*, 2008, pp. 683-689.

4. Z. Racheva, et al., “A conceptual model of client-driven agile requirements prioritization: Results of a case study,” *Proc. ESEM 2010 - Proceedings of the 2010 ACM-IEEE International Symposium on Empirical Software Engineering and Measurement*, 2010.

5. D. Leffingwell and D. Widrig, *Managing software requirements : a use case approach*, Addison-Wesley, 2003.

6. K. Winbladh, et al., “Eliciting required characteristics for usable requirements engineering approaches,” *Proc. Proceedings of the ACM Symposium on Applied Computing*, 2009, pp. 360-364.

7. R.J. Costello and D.-B. Liu, “Metrics for requirements engineering,” *Journal of Systems and Software*, vol. 29, no. 1, 1995, pp. 39-63.

8. N. Gorla, et al., “Organizational impact of system quality, information quality, and service quality,” *Journal of Strategic Information Systems*, vol. 19, no. 3, 2010, pp. 207-228.

9. A. Ivanovic, *Dictionary of Marketing*, Peter Collin Publishing, Ltd., 1998, p. 244

10. CommerceDictionary.com, “CommerceDictionary.com,” 2011; http://www.commercedictionary.com/.

11. B. Nuseibeh and S. Easterbrook, “Requirements engineering: a roadmap,” *Book Requirements engineering: a roadmap*, Series Requirements engineering: a roadmap, ed., Editor ed.^eds., ACM, 2000, pp. 35-46.

12. G. Regev and A. Wegmann, “Where do goals come from: The underlying principles of goal-oriented requirements engineering,” *Proc. Proceedings of the IEEE International Conference on Requirements Engineering*, 2005, pp. 353-362.

13. M. Shibaoka, et al., “GOORE: Goal-oriented and ontology driven requirements elicitation method,” *Book GOORE: Goal-oriented and ontology driven requirements elicitation method*, Series GOORE: Goal-oriented and ontology driven requirements elicitation method 4802 LNCS, ed., Editor ed.^eds., 2007, pp. 225-234.

14. H. Kaiya and M. Saeki, “Using domain ontology as domain knowledge for requirements elicitation,” *Proc. Proceedings of the IEEE International Conference on Requirements Engineering*, 2006, pp. 186-195.

15. H. Kaiya, et al., “AGORA: attributed goal-oriented requirements analysis method,” *Proc. Requirements Engineering, 2002. Proceedings. IEEE Joint International Conference on*, 2002, pp. 13-22.

16. A.M. Sen and S.K. Jain, “An Agile Technique for Agent Based Goal Refinement to Elicit Soft Goals in Goal Oriented Requirements Engineering,” *Proc. Advanced Computing and Communications, 2007. ADCOM 2007. International Conference on*, 2007, pp. 41-47.

17. A. Sen and S.K. Jain, “A Visualization Technique for Agent Based Goal Refinement to Elicit Soft Goals in Goal Oriented Requirements Engineering,” *Proc. Requirements Engineering Visualization, 2007. REV 2007. Second International Workshop on*, 2007, pp. 2-2.

18. L. Marcio Cysneiros and E. Yu, “Requirements engineering for large-scale multi-agent systems,” *Software engineering for large-scale multi-agent systems*, 2003, pp. 77-148.

19. J. Martin, *Rapid application development*, Macmillan Publishing Co., Inc., 1991.

20. A. Fruhling, et al., “Designing and evaluating collaborative processes for requirements elicitation and validation,” *Proc. Proceedings of the Annual Hawaii International Conference on System Sciences*, 2007.

21. Z. Balfagih and M.F. Hassan, “Quality model for web services from multi-stakeholders' perspective,” *Proc. Proceedings - 2009 International Conference on Information Management and Engineering, ICIME 2009*, 2009, pp. 287-291.
22. D. Arnold, et al., “Modeling and validating requirements using executable contracts and scenarios,” *Proc. 8th ACIS International Conference on Software Engineering Research, Management and Applications, SERA 2010*, 2010, pp. 311-320.
23. B. Evans, “Global CIO: The Top 10 CIO Issues For 2010,” 2009 http://www.informationweek.com/news/global-cio/security/showArticle.jhtml?articleID=222002799.
24. M. Duvall, “Top 10 Concerns Of CIOs,” 2009; http://www.information-management.com/news/-10016150-1.html.
25. J. Luftman, “Key issues for IT executives 2007,” *MIS Quarterly Executive*, vol. 7, no. 2, 2008, pp. 269-286.
26. Gartner, “Data Modeling Without Meetings: Wiki over Whiteboard?,” 2010; http://www.burtongroup.com/Research/PublicDocument.aspx?cid=1987.
27. Gartner, “New Delivery Models and Requirements Drive the Evolution of Finance,” 2010; http://www.gartner.com/DisplayDocument?doc_cd=209156.
28. L. Kof, “On the identification of goals in stakeholders' dialogs,” *Book On the identification of goals in stakeholders' dialogs*, Series On the identification of goals in stakeholders' dialogs 5320 LNCS, ed., Editor ed.^eds., 2008, pp. 161-181.
29. F. Cafer and S. Misra, “A Cognitive Requirement Specification Model,” *2009 24th International Symposium on Computer and Information Sciences*, 2009, pp. 517-520.
30. D. Port, et al., “Using simulation to investigate requirements prioritization strategies,” *Proc. ASE 2008 - 23rd IEEE/ACM International Conference on Automated Software Engineering, Proceedings*, 2008, pp. 268-277.
31. M. Daneva and A. Herrmann, “Requirements prioritization based on benefit and cost prediction: A method classification framework,” *Proc. EUROMICRO 2008 - Proceedings of the 34th EUROMICRO Conference on Software Engineering and Advanced Applications, SEAA 2008*, 2008, pp. 240-247.
32. J. Webster and R.T. Watson, “Analyzing the past to prepare for the future: Writing a literature review,” *Management Information Systems Quarterly*, vol. 26, no. 2, 2002, pp. 3.
33. R.B. Schwartz and M.C. Russo, “How to quickly find articles in the top IS journals,” *Communications of the ACM*, vol. 47, no. 2, 2004, pp. 98-101.
34. S. Nejati and M. Chechik, “Let's agree to disagree,” *Proc. 20th IEEE/ACM International Conference on Automated Software Engineering, ASE 2005*, 2005, pp. 287-290.
35. X. Liu, et al., “Priority assessment of software process requirements from multiple perspectives,” *Journal of Systems and Software*, vol. 79, no. 11, 2006, pp. 1649-1660.
36. R. Peng, et al., “A Requirements Maturity Measurement Approach based on SKLSEWiki,” *Proc. Proceedings - International Computer Software and Applications Conference*, 2010, pp. 251-254.
37. V. Gaur, et al., “An application of multi-person decision-making model for negotiating and prioritizing requirements in agent-oriented paradigm,” *Proc. DSDE 2010 - International Conference on Data Storage and Data Engineering*, 2010, pp. 164-168.
38. H. Jiang and X. Yang, “Performance requirement elicitation for financial information system based on ontology,” *Proc. IEEE Region 10 Annual International Conference, Proceedings/TENCON*, 2009.
39. F. Molina, et al., “An MDE Modeling Framework for Measurable Goal-Oriented Requirements,” *International Journal of Intelligent Systems*, vol. 25, no. 8, 2010, pp. 757-783; DOI 10.1002/int.20430.
40. D. Wu, et al., “Finding Success in Rapid Collaborative Requirements Negotiation Using Wiki and Shaper,” *43rd Hawaii International Conference on Systems Sciences Vols 1-5 (Hicss 2010)*, 2010, pp. 663-672.
41. D. Port and T. Bui, “Simulating mixed agile and plan-based requirements prioritization strategies: proof-of-concept and practical implications,” *European Journal of Information Systems*, vol. 18, no. 4, 2009, pp. 317-331; DOI 10.1057/ejis.2009.19.

42. S. Kelly, “Towards an Evolutionary Framework for Agile Requirements Elicitation,” *Eics 2010: Proceedings of the 2010 Acm Sigchi Symposium on Engineering Interactive Computing Systems*, 2010, pp. 349-352.

43. T. Gorschek and C. Wohlin, “Requirements abstraction model,” *Requirements Engineering*, vol. 11, no. 1, 2006, pp. 79-101; DOI 10.1007/s00766-005-0020-7.

44. R. Colomo-Palacios, et al., “A study of emotions in requirements engineering,” *Book A study of emotions in requirements engineering*, Series A study of emotions in requirements engineering 112 CCIS, ed., Editor ed.^eds., 2010, pp. 1-7.

45. P. Laurent, et al., “Towards automated requirements triage,” *Proc. Proceedings - 15th IEEE International Requirements Engineering Conference, RE 2007*, 2007, pp. 131-140.

46. I. Ramos, et al., “Requirements engineering for organizational transformation,” *Information and Software Technology*, vol. 47, no. 7, 2005, pp. 479-495.

47. P. Chatzipetrou, et al., “Prioritization of issues and requirements by cumulative voting: A compositional data analysis framework,” *Proc. Proceedings - 36th EUROMICRO Conference on Software Engineering and Advanced Applications, SEAA 2010*, 2010, pp. 361-370.

48. A. van Lamsweerde, “Goal-oriented requirements engineering: a guided tour,” *Proc. Requirements Engineering, 2001. Proceedings. Fifth IEEE International Symposium on*, 2001, pp. 249-262.

49. R.O. Briggs, et al., “Collaboration engineering with ThinkLets to pursue sustained success with group support systems,” *Journal of Management Information Systems*, vol. 19, no. 4, 2003, pp. 31-64.

50. I. Jacobson, et al., *Object-oriented software engineering: a use case driven approach*, Addison-Wesley, 1992.

51. M. Lee, “Foundations of the WinWin Requirements Negotiation System,” University of Southern California, 1996.

52. J.A. Russell, et al., “Affect Grid: A single-item scale of pleasure and arousal,” *Journal of personality and social psychology*, vol. 57, no. 3, 1989, pp. 493.

53. R.M. Felder, “Matters of style,” *Asee Prism*, vol. 6, no. 4, 1996, pp. 18-23.

54. R.M. Felder and L.K. Silverman, “Learning and teaching styles in engineering education,” *Engineering education*, vol. 78, no. 7, 1988, pp. 674-681.

55. R.K. Yin, *Applications of case study research*, Sage Publications, Inc, 2003.

56. R.E. Stake, *The art of case study research*, Sage Publications, Inc, 1995.

57. M.Q. Patton, *Qualitative evaluation and research methods*, Sage Publications, Inc, 1990.

58. B. Soloman and R. Felder, “Index of Learning Styles Questionnaire,” 2003; http://www.engr.ncsu.edu/learningstyles/ilsweb.html.

59. J.M. Carroll, *Making use: scenario-based design of human-computer interactions*, MIT Press, 2000, p. 46.

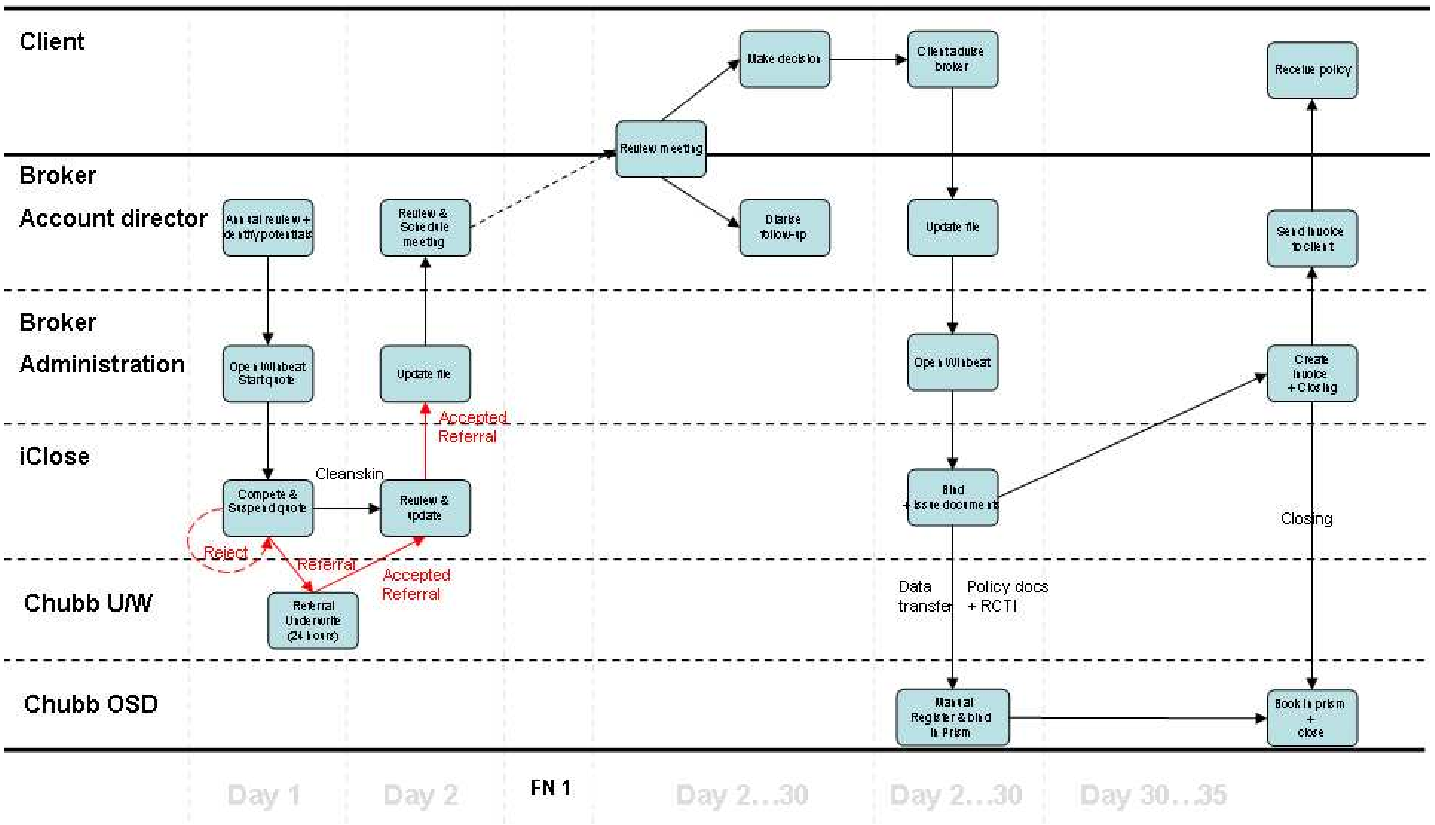
60. J. Aranda, et al., “Requirements in the wild: How small companies do it,” *Proc. Proceedings - 15th IEEE International Requirements Engineering Conference, RE 2007*, 2007, pp. 39-48.

Appendices

Appendix A: Process diagram of client request

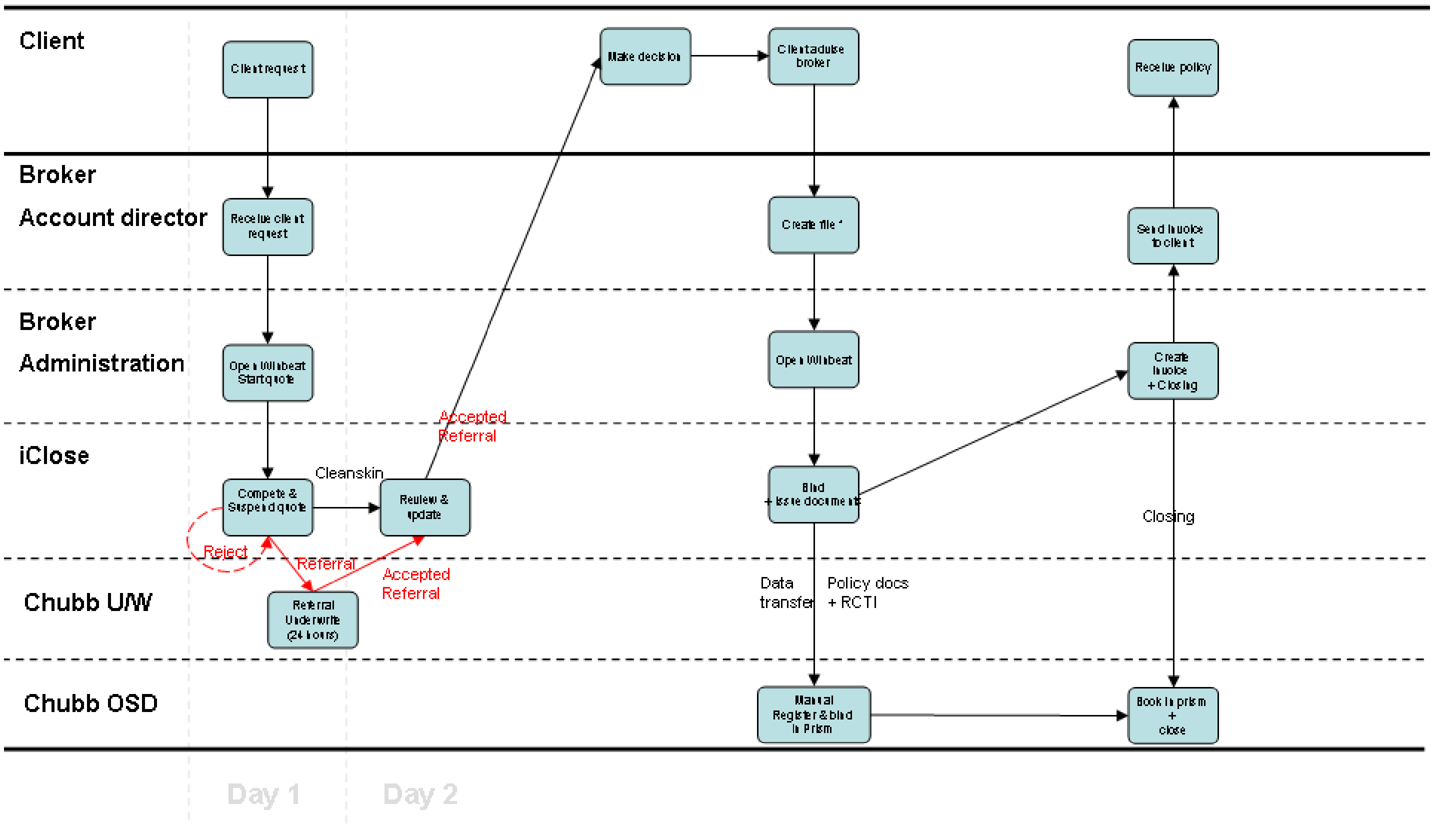
New Business - Annual Review

Assumptions
1. Only Prism gets updated
2. No submission step
3. Mid term adjustments or cancellations through existing manual process
4. CICA gets a soft copy of iClose policy
5. No future effective dates



New Business - Client Request

Assumptions
1. Only Prism gets updated
2. No submission step
3. Mid term adjustments or cancellations through existing manual process
4. CICA gets a soft copy of iClose policy
5. No future effective dates



Client

Broker Account director

Broker Administration

iClose

Chubb U/W

Chubb OSD

Client request

Receive client request

Open Winbeat Start quote

Compete & Suspend quote

Reject

Referral

Referral Underwrite (24 hours)

Accepted Referral

Cleanskin

Review & update

Accepted Referral

Make decision

Client advise broker

Create file *

Open Winbeat

Bind + issue documents

Data transfer

Policy docs + RCTI

Manual Register & bind in Prism

Create invoice + Closing

Send invoice to client

Receive policy

Closing

Book in prism + close

Day 1

Day 2

Appendix B: List of excluded studies with rationale for exclusion

Reason for excl.	Authors	Title	Year	Source
E1	Dasgupta et al.	E-Commerce in the Indian Insurance Industry: Prospects and Future	2002	Electronic Commerce Research
E1	Taylor et al.	Technology readiness in the e-insurance industry: an exploratory investigation and development of an agent technology e-consumption model	2002	Journal of Insurance Issues
E2	Balfagih, Z., Hassan, M. F.	Quality model for web services from multi-stakeholders' perspective	2009	ICIME 2009
E2	Qureshi, N. A., Perini, A.	Requirements engineering for adaptive Service Based Applications	2010	RE2010
E2	Wen et al.	Stakeholders-driven requirements semantics acquisition for networked software systems	2010	COMPSAC W 2010
E2	Verlaine et al.	Towards automated alignment of web services to requirements	2010	WeRE 2010
E3	Carod, N. M., Cechich, A	Cognitive influences in prioritizing software requirements	2010	ICSOFT 2010
E4	Daneva, M., Herrmann, A	Requirements prioritization based on benefit and cost prediction: A method classification framework	2008	EUROMICR O 2008

Appendix C: Interview template

Before interview

- Explain objective
- For the purpose of my master thesis, this interview will be recorded.
- This interview will approximately take 30 minutes

Objective

The objective of this interview is to analyse and understand how requirement engineering is currently done in Chubb. With requirements engineering in the terms of: requirements gathering, requirements negotiations, requirements validation and prioritization.

Questions

1. Organizational

1.1	Can you tell me what kind of company Chubb is? (Number of employees, age, business area etc…)
1.2	Can you tell me something about the projects / services the company provide in terms of time on the market, typical customers/ end-users, projects?
1.3	What is your position in the company? (Role, daily tasks, responsibilities, etc)

2. Definition of requirements

2.1	What is a “requirement” to you?
2.2	What is a “good requirement” to you? And to the company? Is the quality of the requirements assessed? How?
2.3	How is Chubb gathering requirements for a project like iClose?
2.4	Is there a static methodology/framework that you are using for requirements engineering?
2.5	Are you using tools/templates for requirements engineering? If yes: What tools/templates are you using? If no: Did you considered using tools/templates? Why (not)?

3. Gathering of requirements

3.1	Which techniques are you using to gather information from these stakeholders? (Interviews, case scenarios, joint applications development etc…)
3.2	Which other sources and techniques do you use to gather information from? (Available documentation, competitor analysis, market research)
3.3	What do you document and what not?
3.4	How are requirements documented? What information and attributes are documented about the requirements?
3.5	How do you choose the stakeholders for a project?

4. Requirements negotiations

4.1	How do you involve the stakeholders during the project?
4.2	How do you solve conflicting requirements?
4.3	How do you determine if the suggested requirements/input from the stakeholder is right?
4.4	Do you think it is important that all the stakeholders understand the requirements? Why?

5. Prioritization and validation of requirements

5.1	How do you determine which requirements are critical and which are not?
5.2	How do you validate the information from the stakeholders?
5.3	How is it decided what to include in the product? How are the requirements prioritized? What is difficult when deciding what to include in the product?
5.4	What kinds of dependencies between the requirements have you come across? Are dependencies documented? Are dependencies actively looked for?

6. General requirements questions

6.1	How many requirements are handled in a typical project? Who suggests the requirements?
6.2	What is the most time consuming part in the requirements process?
6.3	Is it possible to make decisions too late? What can be the effect in that case?
6.4	Have you ever had that project deadlines were delayed? What were the main reasons for that?
6.5	Have you ever considered using a collaboration system for requirements engineering? Why (not)?

7. Case scenarios

7.1	Sometimes there are “wicked problems” in projects, which means that requirements are constantly changing. How would you address this problem?
7.2	What was the worst project that you had to do to gather requirements? What did you do?

Appendix D: Direct observation – Field Evaluation

Location: ______________________________
Report date: ______________________________
Observation period: From: ____- __ - __ To: ____- __ - __
Project ______________________________
Participants: ______________________________
____

1. Are decision documented?
   □ yes □ no

2. Are requirements requests from stakeholders documented?
   □ yes □ no

3. Do stakeholders use templates to formulate their requirements?
   □ yes □ no

4. Do stakeholders use a collaboration system?
   □ yes □ no

5. Do stakeholders use cognitive tools to generate an overall understanding of the requirements and the system?
   □ yes □ no

6. Is the group of stakeholders diverse?
   □ yes □ no

7. Does the requirements engineer use prioritization mechanisms?
   □ yes □ no

8. Does the requirements engineer use automated validation mechanisms?
   □ yes □ no

9. Are the stakeholders consistent in explaining and defining their requirements?
   □ yes □ no

10. Doe the requirements engineer use interviews to gather the requirements from the stakeholders?
    □ yes □ no

11. Is the requirements engineering done iteratively?
    □ yes □ no

12. Number of stakeholders: ___

13. Number of requirement concerns: ___

Requirement concerns:

____
____
____

To what extend do stakeholders participate during the requirements engineering process?
Not at all A great
deal
1 2 3 4 5

Actions that the requirements engineer took to solve conflicting demands:

Actions that the requirements engineer took to gather the requirements:

Actions that the requirements engineer took to prioritization and validate the requirements:

Actions of the stakeholders during the negotiation process of the requirements:

Other observations:

Appendix E: Direct observation – Overall field Evaluation

Report date: ____
Observation period: From: ____- __ - __ To: ____- __ - __
Project ____

1. Total number of stakeholders: ___

2. Total number of requirement concerns: ___

3. Total number of iterations: ___

4. Total number of meetings: ___

5. Total number of iterations per requirement concern:

No.	Requirement concern	Iterations
1		
2		
3		
4		
5		
6		
7		
8		
9		
10		
11		
12		
13		
14		
15		

6. Stakeholders participating in meetings

Stakeholder	Stakeholder function	Meetings
1		
2		
3		
4		
5		
6		
7		
8		
9		
10		
11		

12		
13		
14		
15		

Appendix F: Interviews

Appendix F1: Interview with the Project Manager of the iClose project about requirements engineering at Chubb

Date	24-03-2011
Time	11:00 – 12:00
Location	Chubb – Meeting room 3
Medium	On site (Sydney)
Project	iClose

Can you first tell me about the company? What it does?
Some of this you need to check on the net because globally I'm not sure about the numbers. But in Australia, we are talking about a general insurance company and it is high end stuff. So it is special insurance around marine cargo, public liability, building liability, cooperate travel, products like that. Meanly products for big cooperate companies.
In terms of numbers of employees is about 200 in Australia. Spilt in two big offices in Sydney and Melbourne and smaller in Brisbane and Perth.

In terms of age it is an old company. It is over a 150 years old I think globally. In Australia it is about 30 years old.

What is actually a requirement from your point of view?
A requirement for me is what the business needs to get the job done. Have an objective in mind. The requirements is what the project team (we) need to build to get the objective.

How do you see what a good requirement is?
A good requirement is a clear and concise. They are pragmatic, so they are at the right level. And they need to be outcome based so the requirements are what the business need to do rather than how they think they need to do it.
Requirements should be: This is the objective we are trying to get to as apposed the way we do the work.
They should give the project team their requirements. The project team will then work out what the best way is to deliver their requirements.

3:36 How is the requirements engineering done here at Chubb?
At the moment the way we do it is;

Gathering requirements is done by a number of ways, it is through analyzing existing systems because we often enhancing or replacing an existing system. We are doing it now and what do we need to improve on. So do some analysis on the existing process. We start workshops so we get people around to work for. So how we think how it is going to work. What to think we need. So by doing process maps, brainstorming, stuff like that.
We also put business analysts in a team to actually sit down and document the requirements for them. It is more of a back and forward process where someone from the business will elaborate the business requirements. The Business Analysts will write it down put it into a document and than back and forward, so it will get refined. Until it is ready to be signed off.

So you do not have a static framework which you apply?
I think there is a corporate framework. The issue with that is you can use that if there are lots of resources and lots of planning. Because the framework usually is robust and it will get you a great outcome. But sometimes the smaller projects either don't have the time or the money the appetite

to do that in detail. When you talk to the IT manager. He would be able to point you towards where that framework is. I would not be surprised if that is a very detailed framework. We use some of that, but how much we use is definable. Because the other thing is; it depends on the project. If you spending 20 million dollars, you make sure that the requirements are right. If you spent 20 million dollars and the system isn't used. You will get into trouble. On the flip side; if you are spending 20 000 dollars, you do something fast. You think it is going to work; it might work of it might not work, but you got a greater leap up there because you aren't losing that much money. For the iClose project we are spending a couple of 100 000 dollars maybe more, but we have a greater latitude to accept risk. The other thing is it may minimize the risk is because we are doing prototyping. We don't have people that go a way and build something for 6 months and then come back. We have to see something every week. Because of that we know that some things aren't working and we can change the approach. So there are different approaches that we can use. Basically these are methodologies / frameworks.

We also do prototyping, which is a type of gathering requirements. We don't do the JAD, that is a formal methodology but we don't use that. Again it will be nice to but it is a … issue at the moment for us.

Do competitor analysis. One of the things we done is; What is everyone else doing what are the competitors are doing. It is quite easy to see and a lot of that is what comes out of the requirements.

We actually wanted to meet of what other competitors are doing. We don't want to go in front of them because that is expensive. We are conservative company. We rather follow than lead. They making all the mistakes and we will deal with these mistakes and jump a head.
We also did some market research. So we said to the brokers. What do you think you want? You want this or this. If you have a choice you will get A or B? What is the most important thing for you? How would this fit in your process? So we have done a lot of that.

8:49 How we chosen stakeholders of the project?
There are people they need to be involved because they got to deliver stuff and sign off and the end customer, because they pay for it. Other stakeholders that you need to involve are IT, because they deliver. So the business you need to involve because they pay for it. External stakeholder because they have the technology we need.

What about the end-users?
Say that the business is the end-users. Another important stakeholder we don't choose is risk management and compliance. So are working in a highly regulated environment. Obviously anything we build need to forward in to our privacy legislation, trade practise legislation, insurance legislation. Austalia has a really regulatory regime. It is one of the greatest in the world. Because of that we always have the risk and compliance in the project, because they have to sign off.

Are you using standard techniques or frameworks for the negotiation process? Because you have stakeholders who probably have conflicting demands. How do you deal with those conflicting demands?
Number of ways. There are always conflicting requirements and we always need to negotiate that out. Because the business typically; everything they wants, they lists. But we don't have enough time or money to deliver that. So part of that is the scope. So parts are A B or C and everything that deals with D, we are not building. So you always have to test your requirements back to the scope. And that is a good way of saying that is not in scope what we are doing so it is not in the current requirements so we do it later.

11:25 So the requirements are more determined by the BA and based on that he will gave feedback to the stakeholders and say this is not a real crucial requirement to right now in the project?

The business analyst involved will more come down to the project manager. IT is a high level of negotiation. Because the BA is going to be working with the end user, at that stage he is not going to say; we are not doing that. He is just documentation and that wants to negotiate that at a team level.

In a project you got three things you can manage and that is the time frame, scope or requirements and the amount of money you spent on the project. The easy way to negotiate out these requirements is like; you can put that requirement into the systems, by either increase how much you spent and put more resources in the project or you need to let the same amount on the team and extend the time frame. So we can do that, but it will take longer or cost more or we can reduce the requirement. There is a sort of area matrix that you can work through because usually you can't say we can do more with the same amount of money or we can do that with the same amount of time. You need to be able to manage it like that. That is a very good way, there are tools around that.

<u>13:21 Is Chubb using some of those tools?</u>
Well the tool is actually having someone that facilitates that discussion, someone like me. So that is the workshop where you go trough. So it is not a template, it is a human tool. So yes we do use them. It is simple saying you can't have it unless you give me some money or another two years.

Conflicting requirements. You go back to scope. When you pull back to scope, it usually works. In terms of conflicting requirements, when you get people involved (stakeholders). They usually have a role and responsibility and you can take their requirement to their roles and responsibility. So product person will be saying this is what the product requirements are and they have the ownership on that. Risk and compliance will say this is your risk and compliance requirements and they own that. If they disagree, then usually you will say, product person you don't run the risk and compliance requirements they do. So that is the way you negotiate it out. They won't tell you what to do and you won't tell them what to do. Also bring it back to scope first then bring it back to role and responsibilities, you can also bring it back to costs and time if required. So there are a number of step where you go through. If the requirements is in scope, do we have enough money to do it, do we have enough time to do it. If there is a cross against any one of them it will fall out. If it is yes yes yes then if everyone agrees on the requirement yes, again, it is almost a checklist where you go through, if it is hurting anyone's time then you need to discuss it. No we can't do it in time, do we want to extend the time? Yes or no . No we don't have enough money? Do we want to extend the money? If everyone agree? If you don't agree, why not? Is it your role to disagree on that?

But sometimes you get is a sponsor to solve that. The people that are working as a team can't decide and you escalated it to through the senior member of the team and he makes that decision for you. But it is not a great outcome. You want people to negotiate it out by having a discussion you don't want to tell them what to do, because that is going to be a bit cranky.

<u>Sometimes requirement are depending on each other? Are there certain ways how you document those dependencies?</u>
We definitely look for the dependencies in the requirements because what it usually means is that you got to build one thing before you can build the other thing. And you need to understand those dependencies that comes with the sequence stuff. And you may actually put them into strengths of work so it comes up a bit to a critical. Sometimes you say critical staff and dependencies. What is critical and what is not critical? Critical is the bare minimum to get a result against your objective. Can you go back to your scope and say: What is the bare minimum that I need to do to deliver my objective? You also say where are the dependencies? While sometimes that is depended on some non critical scats somewhere else. You still need to build the non-critical stuff. You might have to prepare a document in the system that is critical there is some requirements that other people have need to be done before it happened, like risk and compliance, sometimes we say it is not critical. But it is. It needs to be done before. That comes into sequencing. In terms

of prioritizing; you can go through a process of saying what the objectives are and what is the bare of requirements to fulfil that. What are the dependencies across that to work and how to sequence them and how do we actually prioritize the development and the delivery of it. So you get valuable outcomes. But in terms of tools you use, it is more a process where you go through and you are always going back to the scope and objective saying. If we got a question on how important it is, or how to sequence it. How do we deliver on business benefits, because you may reprioritize things, because people might want them. But it might not provide a tolerate impact. Someone want to look the systems really good. People always want the systems to look really good, like webpage and stuff, but realistically the end-users sometimes don't care. They just wanted to be able to process the new business. So in terms of the ForeFront stuff we got as an example, we have some requirements about the look and feel of the systems and then we have some requirements about what it does, like rating, binding and issuing business. Now, when I prioritize stuff, I will say to them do you want a systems that looks really good but doesn't work or do you want a system that will do the job but doesn't look good, but will look good later and I can guarantee you that is an easy way to go: Oh hang on if it looks good but it doesn't work as well, then I make less money. So I reprioritize that, while real requirements came into the meeting room a lot of people were saying one that looks like that, a competitors one. Well we can look like that but it might not getting you to the business objectives which is increase new businesses. So that is a sort of steps you can work through. And that is usually a sort of way you can prioritizing it. You can just say; you tell me what you are going to do: look and feel or functionality? They usually choose functionality. And what you than can do is once the systems start working and when you get money coming in we can use some of the money to enhance it so we prioritize the requirements in release 1 2 and 3. Critical requirements come into release one. Need to have but not critical comes into release 2 and 3 is all the nice things. And you know what happened? 6 months later, they had forgotten what they wanted to have: orange or blue or funny buttons to press on. What they really wanted is; is it working, do my clients like it and does it write new business for me.

There might be tools, people like to use tools and so long the way I doing things that work for me. I maybe done the tool that is usually a cam corporate tool set. I'm sure you can find it. I'm encourage to do a bit a research, which you've already done. There are thousand ways of prioritizing. Again some of them really rigid and robust. But the other thing is, if you are only doing a small piece of work. "how big is the process you are running through" Like if you crack a walnut with a sledgehammer. If to much process of prioritization around it, you spent so much time on meeting. It is all about finding the right amount of stuff.

If you got small medium and high complexity project. For a small project it is likely you will have a chat with a few people and we will agree on it in a meeting, because it is a small project. No one wants to put that effort into it. As it become more complex, you might to have specific meetings, run workshops, a little bit of negotiation, use some simple tools. But in a high complexity context, you will be using templates, excel spread sheets, you will be writing the motive of how complex this requirement is (high medium or low). How much benefits it has (high, medium or low). Can we quantify the benefit? Can we quantify the requirement? You might go through a process of some weeks, sometimes even months. And people will sign it off and there will be negotiations. We will have meetings just about one requirement or one screen. But for projects that are close to three months, you look for simple. To twelve it might be medium, Anything above twelve it is going to be complex, people come in the project, people leave the project. So you need to go through the trouble of documenting, using tools. So there is a sort of a scale that you go alone and I'm sure you will find it in your research. We're in between small and medium (iclose), so I don't know how it is going to work. We will see soon.

What is the most time consuming part in the requirements engineering process?
You go from having no requirements to 50% of them or even 80% of them quite quickly. So if you say it will takes of 4 week to gather the requirements, you will have 60% of the requirements by the end of the first week. You might have 85% almost 90% by the end of the second week. So you are half way through your time frame. It will at least take you two weeks to get the other 10%

of the requirements. If you know what I mean. So it is very easy to say; this is what I need, getting the overview of the requirements. Getting them to refine it, so you can actually build something, takes you 50% of the time in that last 20% of getting the requirements. Does that make sense? So easy to get it and writing down. Refining it, answering it, negotiating out the complexities and getting the business to sign off on it. So to say it is a 100% of the requirements .If you build that is, I'm happy, is very difficult. What they say is: Can you just build it now and have a look? You say: No I can't, I want you to tell me: This is what I want and then go build it. So that is the reason why we are trying to use some prototyping. Whatever it is a month or 6 months. 50 % of the requirements you will get very early and to last 10% or 20% where you are getting it done is really hard. This is not: This are my requirements… it is more like refining them, signing off on them and understanding the real impact.
We are not going to sit here and say to you: My requirements of the system are write, issue and bind (iClose). I want it online for 100 brokers, I wanted integrated with their backend systems. There you go, that is 50%. You know what I need, when you walk out of the room. But in the high level, we need to say: Which product? Which processes? Which platform? How much do you want to spent on this? You know there are a lot of other questions. So you go: Ooh that is complicated, so you go to another level of detail. And then you go; well I needed to sign off on this. To build it for you. And if we build where you ask for that was wrong, that is your problem. And then they go: Ohh wait a minute I have a other look then.. So that is how it is. And that is really difficult. That is the challenge of business requirements. And this all links in because that 20% work of getting to sign it off is why you need to choose the right stakeholders. Because you can get all the requirements from someone that can't sign off and he sends you to Rodney and Rodney goes: Oh these requirements are not right, I can't sign this off. And you think you collected 80% of them, so then you are in the wrong hole. So the way you setup these questions is very useful. It sort of works you through the process.

Have ever considered some kind of a collaboration system, like a wiki system?
The answer is: No I haven't considered it at Chubb, because business is not mature enough in understanding how to use it.

In order to work well, you need people to understand how to collaborate and in use the system like that and being involved in doing it. I don't think that would work here at the moment, because people do not have the time for it. You do not have the system; we have to set it up for them and have to train them on doing it. Although I know it is very useful, it could be fantastic, especially at Chubb, because we are multi-sided. So a collaboration tool would be really useful, for example: Melbourne, Sydney, Singapore and the UK are all stakeholders using it would be fantastic. It just not goanna work at the moment, because Chubb already got their framework, and using another framework would means you have to go to someone and say: Can we put this in the framework? And that would be so difficult that it wouldn't be worth the effort at the moment. So the answer is: I haven't considered here, and the reason is because I don't want be training up on people doing it. I rather insist on someone else its project and how to do it.

A wiki would be great. However there is some collaboration technique. From email, people are using direct communication, like telephone conferencing, face-to-face meetings and instant messenger, which is a step into the right direction. And maybe we can do it, it is worth rising. We can have a look. I did not actively thought about that. So the answer is No, why not? Just I think it could be difficult at the moment. Anything that is difficult I rather avoid, because it is going to be hard for the project.

30:10 Have you ever had a project you where you had wicked problems? Where requirements are changing all the time?
Broadly a lot of requirements are like this, because things are always changing. Like now, we are trying to build the system and we are using the current wording policy. We know that within three weeks they will change the wording policy and we know that they do it again three months after that.

In managing you just have to be aware of that, you need to try to engage the right people. So that they tell you that they changing staff early enough for incorporating the requirements. But a lot of the time one of the solutions is that you need to build a solution, whether it is a IT-solutions, a Business-solution, process-solution of whatever solution it is, you need to go and try to build it that is flexible going fore worth. So you might build it in modules. So rather than building it all lock down, so you got to change the whole thing. Every time you have a change in the requirements, and you build it in modules, you can pull a module out, change it and stick it back in, which will reduce the time it takes to get the change done. And the complexity and also you just need to recognize it up front. Are you working in a changing environment and be comfortable with it and accept it, so that it is not a surprise. As long as people expecting it, they usually saying: Ohh we though it will change and it has, rather then surprise six weeks work has been a waste of time, someone changed the requirement. It depends on how the project team approaches it. They will be aware of it, keeping their eyes open, see what you are gathering.
By gathering requirements you need to be saying; are these static or do they change often? Can you tell me when do you think they change? How do you now when they are going to be changed? So what are the requirements now and what could it be tomorrow. And then again it loops back to prioritization. The critical requirements are today's requirements and nice to have are for tomorrow. So we do it for today, and nice to have we do it for tomorrow.

33:00 What is the worst project that you ever had in terms of requirements engineering?
The project that I worked on it, was in another company. It was a complex project ran for three years. And what we found was, it takes 6 months to gather the requirements and there was a lot of conflicting requirements, lot of different stakeholders, lot of changing requirements. And what that meant is, it takes 6 months time to get the requirements and 12 months to build it. So 18 months after you started, you say here is the system. And everything is changed, we got new requirements. And we go: We just worked on this for 18 months, and they go like; yeah we don't use that anymore. But you have been developing it for 12 months so we've forgotten it. So you need to break it down and do the prototyping so that 3 months after you started they can have a look at it so that they can say: Yes or No. And when you showing up with the requirements after six months they go like: I have seen the prototype you are already working on it and I know what you are doing and hopefully to repeat that after 9 months. That was a really crap project, we lost a 20 million dollar. So the way you gather requirements is really critical. You need to use the right approach for the right project.

34:50 One final question: I forgot to ask for your role/task/position in the company.
I am a project manager, I'm a contract project manager. My role is to manage the end-to-end project. So that should be managing the resources allocating to the project to achieve the objective. Getting the stakeholder together, business analyst together to get the requirements, getting the IT services involved and so on. That is what it should be, but my responsibility is end-to-end. I'm making sure that the solution is delivered and specific to this project, I have a hype of projects. Which means that some projects are busier at sometimes and some aren't. So this project is behind, but we changing that in the next couple of weeks. Because the other project that I'm working on is now hopefully slowing down. But broadly, my role is to manage other people's activity. I don't actually doing work, I'm managing the work. If I am doing the work, then something is wrong. It is a great job, you don't actually doing anything. You tell people what to do, and that is what I like.

Appendix F2: Interview with the CFO about requirements engineering at Chubb.

Date	10-05-2011
Time	10:00 – 10:45
Location	Chubb – Office CFO
Medium	On site (Sydney)
Project	iClose

00:58 Can you tell me something about the company? What Chubb does and etc..
In terms of about the Chubb Company in Australia it is about 250 employees in multiple branches, Sydney Melbourne Brisbane. In terms of the products we sell, they are commercial and personal. The biggest proportional is commercial, property and casualty and also especially products like professional ingenuity and then on the personal side it is high network liability insurance. We also have a health division which sells corporate travel, but we are also getting into the consumer market. But at the moment we are only dealing through brokers. All the requests are coming in through brokers, which are dealing with the public. But we are setting up a division that is dealing direct with the public. By taking affinity groups, like super firms or associations. We come up with a product for the association that we think that they would like.

02:57 Can you tell me something about your position in your company? You roll and responsibility?
My main role is Chief Financial Officer of Australia and I'm also a director/member of a board. My traditional responsibilities includes also IT, although I'm not the IT manager. The IT manager reports to me from an admin perspective. It also include risk and compliance, in some way. Also the Risk and Compliance expert reports to me in a way from an admin perspective. In sense of that strategy, we have an executable community that is called OZCOMM. That is a board where you discuss the business, like productions and the support. With the strategy we have got we mark and split the strategy to allocate it to different people. My district is to look after is the operational process design and operational specialty. Things like setting up the centralized renewals, anything to do with process work. Also the business transformation manager reports to me as well.

4:43 What is a requirement according to you?
It is a business need, supported by a business case.

5:02 What do you think is a good requirement?
Yeah, I thought there are two levels of requirements. The high level is, where you are trying to define the business needs. In terms of that requirements, a good one is that can generate some money for the company. We shouldn't spending time and money and energy on something that isn't deliver some positive return. There should be a business case up front, obviously a business need and then a business case and that covers the high level requirement. But obviously, if you talk to the IT, requirements are, the detailed data gathering to information that they need to get to build solution. A good requirements on that is; The one that is accurate. Getting the business representative involved in the project so you don't get through people in between (for example in the iClose project with the developers and experts). You get the information directly from the business, so they can design the solution.

06:48 How is Chubb gathering the requirements?
The iClose project is probably a little bit different from how it normally go about gathering requirements for a IT project. Because you have an IT resource (developers from external

company) on the project and we haven't followed the IT structure in terms of gathering requirements. Which would mean, that the business analyst, is getting this and is writing it into detail. In this project it is less structured then that. The business that is directly involved with the developing company and explaining to him and to the project manager and the business analyst and feeding those requirements directly through to the developers company so they can directly build the system and then I suppose can directly to go the business and show them what we have done to verify if it, if it is right. To be honest it is like real time process which is better then through a lot of documentation.

09:51 Is there a static framework/methodology for the requirements engineering? Are there some templates of standard tools which you are using?
Chubb got standard documentation you are suppose to following in terms of IT projects. The project manager used some of those. Because of the size of the project, it is not a huge project. We are not using, or following all the documentation that is out there in Chubb. But certainly the important documentation. But certainly the important documentation, the project manager filled that in, and it goes through internally process at Chubb to get approved and as a software solution. We also need to get it through what is called a build permit. We did a questionnaire that the project manager made, to see the robustness of the solution and if companies will buy the solution. That is more a security if it is successful for Chubb. That more the basic requirements.

11:34 So the build permit is created here at Chubb and then it has to sent trough to the developers' company or is it something internally?
No it is not with the developers' company, it is internal in Chubb, you need to get information from the developers' company in terms of what there security reasons are, their business continuity plan and things like that. Then the IT (home office) will sit down and review that whole permit and they will asked some questions and they will approve it. The questions will go through a next meeting and more information needed, or they don't fit in the Chubb's structural requirements and they will not approve it. The iClose project got approved.

13:10 Which techniques are you using for gathering the requirements?
What we are doing now with the developers is a workshop process. Where we get business people in the same time as with developers. And out of these meetings, the project manager is following it up, in terms of design information required or some data analysis to support direction where we are going. There has been a business requirement document writing which have been come out of the meetings with the business. It more a guide or a reference point then a formal document is used by developers to program. The programming is done by developers after the workshops and discussions they know what we want and how we do it. Then it is all good and we move on, it is not that we change it. There isn't a script driving program or IT does his own testing. It is more interactive.

15:05 Are you also using some other sources to get information that you are using for requirements gathering? Like doing a competitor analysis, market research etc..
We did that as part of the original business case to get approval to move on with the developers. We made a paper which set out how to market will look like in terms of e-commerce. Nearly the way how insurers are interacting with brokers and that shows that we are behind with our competitors in terms of market-facing technology. It was also a case of something that we needed to do to compete. And brokers wanted us to reduce the transaction costs. That was really the business case to get approval.

16:13 What do you document and what do you not document? For example how are you telling the developers the requirements?
It is informal, is mainly through discussion and email. It is not through formal specific documentation.

16:55 How do you choose the stakeholders in a project like the iClose project?
The first decision that we had to make was to choose a product as a pilot. And all those products

were relatively complex. So we needed a lite touch type market-facing technology environment. So the four products were marine cargo product, INT specialist product, TLD and the forefront product. Out of those, we decided to choose forefront, because it was the least complex and best suited for the technology and also had the best support from the business. So it was really the demand, the business support and the ability of the product. And after that the stakeholder fill in the place, because you have someone who knows the product, which is now an underwriter who sells that product. Originally the independent broker manager was looking after the forefront product, but know he / she is moved to a distribution roll. The process manager setup the broker system. So it is an underwriter, the independent broker manager and …. Who is in charge of the broker team and the IT manager from an IT perspective is involved. And we have local stakeholders, like brokers and… the developers are stakeholders

19:31 Do you also involve brokers?
Yeah , for example in an other market facing technlogy IT project, ARCH, and the project manager is now working on both of those. Big part of the project, he is going to brokers, he is getting the information of what works for them. So he is getting that information for that project, might be doing that with the iClose project as well.

20:13 How do you solve conflicting requirements?
Because it is a single product, we did not really have conflicts between stakeholders. The tension what we need to solve in terms of the iClose project is, the underwriters need for information against the process need for a simple solution. What happened is being compromise on the underwriting information that is required. So in a manually environment you have a proposal form they are required to complete, to get all the information they need so they can sell the product to the customer. It about 20 / 30 questions, they had to cut that down to a hand full of questions. So the broker can use the system, he/she would be able to create quote and bind business in a short period of time if they want. So that is the only conflict, tension we have had. It is not between stakeholders, it is between information and the need for a simple solution.

21:55 And how was that solved? The underwriter had to commit to it?
Yeah that is right, they need something simple to get to the brokers. Because the full system is there and if the are not going to use it, the brokers will go to the competitors and they know that, so they have to work within their underwriting guidelines and to cut down the question set.

22:26 How do you determine if the suggested requirements from stakeholders are really the right requirements?
Well the test in the end of the day will be the pilot, whether the broker will use the system or not. You will get feedback from brokers during the pilot, naturally so then it will be tested. If they don't like it, they will tell us why.

23:16 If brokers have other demands on their requirements, for example I want this, but other brokers are saying I don't want this. How do/are you want to solve that?
Well if that situation occurs we come up with a standard solution for all of the brokers and obviously we tell the broker that we go for the simplicity, that it has to be a simple standard solution across all brokers. So as much as possible to deal it clear. To be honest, the forefront product is standard to all brokers, so we don't use different products to different brokers. It should mean that different people are asking for different functionality, which is delivering the product.

24:26 How do you determine which requirements are critical and which are not?
The first thing is making sure which information is needed and secondly the very basic information; the underwriters need to be able to write and bind cover and that is based on revenue and assets. After that, if the underwriters will still need the information, the way we decide is practically collect that information, which means it take longer for the broker to complete the transaction and we will cut it out.

25:36 How do you validate the information from the stakeholders? How do you make sure that the information from the stakeholder is formulated correctly (it is the intended information from the stakeholder)?
Well in terms what we have done for data requirements is. The IT Manager did the mapping of the information from the project that is required for Prism. You want be able to get that booking, unless you have that information, so when the underwriters coming up with the requirements, the IT Manager will cross this back to Prism to make sure it is complete. Also collecting information as much as you need. And then we worked based on the existing proposal form of the underwriters. In terms of functionality, we ask him to change functionality and what it will cost. We need to balance between what the brokers want and the costs. Actually there was one change we made to the base functionality in iClose project and that was to provide multi quote functionality side by side. The underwriters felt that the brokers wouldn't use it if we don't have that. That was something like a surprise and we went build it.

28:17 How do you consider the dependencies on the requirements? For example; the iClose system has to be consistent with the data in Prism that means that there has to be some kind of a link between prism and iClose. And a dependency could be that for example a message has to be by XML.
So that is a example, it is not for the pilot, building a link between Prism and iClose. So we felt that wasn't important enough to build it into the pilot because it was only worth building XML if you have the volume of business for the cost of building that. The pilot will just to get to a certain level of volume which keeps us going to do it manually, but we do that when we go live after the pilot. So that is a good example, I can't think of any other.

30:47 How many requirement are handled in this typical project? Can you tell me an amount of requirements? Is it a lot of requirements compared to other projects?
No it is quite a simple project, in terms of the number of requirements. The iClose product is already build. So all what we are doing is, identifying the changed required to the functionality to be successful so for example the multi quote. And then after that we just configuring the product so writing metrics in it, and our wording and our question set and that is it. It sound very simple but it will take us a while to get to where we are. It is relatively a straight forward project; we are not building it from scratch.

31:51 And in other similar project?
It is different within the other market-facing system that we are rolling out. It is already been build in the UK. What we had to do was bringing it across here and tailored for our product. So in terms of technology requirements it was straight forward. The difficulty with that project is, because it is our first market-facing technology project, to get the process around that. To really understand the broker needs, the work-a-rounds, manuals and processes internally by the brokers.

33:13 In bigger projects, are you using some automated validation technology for validating requirements?
No we don't have that. The only other big IT project that we have running is the ARCH project. Which is bringing in Chubb's Canada system in Australia. It is straight forward in terms of bringing in systems that is already build and operating in Canada. It is just telling the straight needs and the way they are doing the requirements gathering. For the ARCH project it is done through the existing kaizen tools. By analyzing the kaizen tools you are doing the requirements gathering. It is not just gathering the requirements and document them, you are actually gathering the requirements and building a prototype. What the IT is doing, it taken the kaizen tools and reverse engineer them so they have the information they need and then get together and tailoring the advantage of the Canada system.

35:09 What does the Kaizen tool exactly do?
Well it is like a simple version of a system. It is a excel spreadsheet document. But the concept is the same as a system. You got one entry point which feed you through to rate the premium and it generates all the documentation. So it is the same concept of what a proper system would do, but

now it is excel and word.

36:10 Is Chubb using a collaboration system for requirements engineering?
I don’t think so. By collaboration system you mean like the wiki example which you gave earlier?
Yes, or for example netmeeting or messaging systems
We are using netmeetings with the developers for demonstrations.
But also something like a forum?
Not really. To discuss the requirements, going back to the kaizen tool, of the ARCH project, once they insert the kaizen tools, they build the product requirement into the prototype and then they discuss the prototype. And then they give the thumbs up or they concerns the issues. That is the method we are using the validate the requirements in that project.

37:32 Sometimes you have wicked problems. And wicked problems are requirements that are changing all the time. Have you ever had that problem?
Yes that happens all the time.
How are you addressing those problems?
Well in the iClose project because of the validation of what we are doing. The project manager is doing a good job of getting the stakeholder together, in terms of the functionality and the product. We are pretty much stuck to those principles (scope) all the way through the project. There has been some issues with how we are dealing with it. But by having the principles (scope) up-front it will drive you through the process without having to many major changes in requirements. Anytime you have a discussion it is also good to bring up those principles, which reduce complexity.

38:57 What is the worst project that you had to do to gather requirements? What did you do?
I think it is the OPRAH project, the project before ARCH.
What went wrong?
We used a vendor that didn’t had a solution. So we were building a system with them. We provided the business expertise and they provided the IT expertise. And they were building a system that we could use and they could sell the system to customers. So we underestimate the time of involvedness and complexity. We really underestimate the risk in the whole strategy. So we are using the strategy of getting the technology that is already working somewhere else. The other thing that was that they have tried to run the project from Singapore, we couldn’t communicate the business with them.

Appendix F3: Interview with the IT Manager about requirements engineering at Chubb (iClose)

Date	29-03-2011
Time	15:00 – 16:00
Location	Chubb – Meeting room 3
Medium	Telephone (Melbourne)
Project	iClose

00:37 Can you tell me something about the company?
Chubb is established in 1892 so it is an old company. It started as a small family business, selling marine insurance. As the time past the company became much larger. Currently, the number of employees worldwide, is around 10.800. The age of the organization, it is over 125 years old and basically it branched out to many different types of insurance. We effectively have three SBU's (Service Business Units) and they cover personal insurance, commercial insurance and the financial services.

01:48 What is your position in the company?
I recently promoted in the IT managers role and previously I was a business system analyst manager and I was looking after all of the applications owned by Chubb. Now my role is expended to the full included construction for the Australian religion.

02:24 What is a requirement?
A requirement is a requirement for a system to perform a function. Now, you can have a single requirement or you can have multiple requirements. And you have technical requirements and function requirements.

02:55 How do you see what a good requirements is?
A good requirement makes a business request and obviously provides return on investment. So basically we spending money, to build the requirement we need obviously return on investment.

03:25 How are you gathering requirements in a project like iClose?
iClose is a sort of Chubb first adventure into market facing technology. Basically the way the requirements are working; We have a product ForeFront, that is a very standard product. And obviously the type of product is suitable to market facing technology. You don't really want a complicated rate, booking, issuing process. MFT is a good tool for a standard simple products where you don't need any training to complete. You can actually go online and put it in through a portal. In terms of a wider gathering of requirements for a project like iClose. The requirements gathering process is really depending on the type of technology and also what we are trying to achieve. For iClose, because of it is a standard product, it has a standard rating matrix, the wording is quite standard. So there is not a lot of decision making that goes into rating in iClose or rating a Forefront policy. So basically the way we are doing it is essentially meeting with the business on a regular basis (once a week). To gather these requirements we have a business representative and this representative is responsible for passing on the requirements to us. We already have completed our requirements document and this document contains the overall framework for the product and it could be build into the iClose platform.

05:36 Is there a framework for gathering requirements?
Again it depends on what you are trying to achieve. If you look at the methodology used for the iClose product, it is actually quite a different methodology to what we used for other projects.

If we are looking to other strategic pilot projects we have in the pipeline, it are replacements of older booking systems and have a very different framework for that. We use excel, word, kaizen tool and basically we reverse engineering the kaizen-tool to the other platform. We don't have that for the ForeFront or the iClose. So we actually started from the requirements gathering process first and building these requirements into the iClose platform.

06:46 What kind of techniques are you using for gathering requirements?
We use existing systems. We do a quite of a bit of work-shopping. We have a round-table discussion where the representatives from the business, IT and operations responsible for the processing of transactions are sitting together. So sit around in the workshop and basically brainstorm in terms of what we want the system to do, what are our objectives are, and how we building this for market facing technology, or are we building this robust processing or building it for efficiency. So there are number of decision that we make through that workshop. But the idea of the workshop is to actually flash out all the requirements. So we do see someone sitting over there taking all the notes and documenting what the business is actually looking for.

08:15 Do you also look at competitor, or are you looking at what the market is doing**?**
We will look at the market, simple because this is our first product for market-facing technology. We would look at the competitor to see what they have done, because we don't want to make the same mistakes. If they have a market-facing tool that is not user friendly, we won't go down that same track. Let them build the tool and then we would leverage them off what they have actually done. Rather then just make the same mistakes.

09:10 During a brainstorm session, for example, what is documented and what is not documented?
We wouldn't try to come up with solutions in the workshop. Obviously, when I am talking about solutions, I am talking about technical requirements. In a workshop we really are trying to flush out the business requirements. So we are looking for the functional requirements, and the technical requirements come later because we don't want to burden the business people with the technical requirements. So we have the workshops to flush out the function requirements, what does the business want that the system actually do for them. That decides the framework for the application. And again if it is very simple product then we don't want to over complicated the development framework for the product.

10:26 For a product like iClose, how do you choose the stakeholder?
We have a SBU manager, who is responsible for the region (like the Australian region). He is also the key stakeholder, he is to make sure that the application we are building; A. meets the requirements, B. provides return on investment, C is usable by the product indemnity. So he is one of the key stakeholders, the sponsor for the project. The other stakeholders are obviously people from operations, underwriter stakeholders, people who going out to the broker to sell the product, finance to make sure that we can actually cover it in the policy. So there are a number of key stakeholders.

11:35 How do you solve conflicting requirements?
When you build an application, you got an amount of money to spend. If you do run into conflict, the first thing to do is negotiation, so you sell your position to the stakeholders to explain why you do something. Then determine that is something that must be in the system. If it is in the system then you……..fund the amount of money to spend. Basically you need to prioritize. When you prioritize you looking at things like; is it going to make the product complex? Does one requirement has a greater return on investment? You look at thinks like that and then you try to convince the other stakeholders. There is a thing in project management, which is called the triple constrained. Basically it is quality, time and costs. Each impacts the other. Basically you are hooking at the triple constrained and try to negotiate with the stakeholders to explain the position. In IT we can't tell the business what they should or shouldn't have. The business is always the driver of the development process. But what we can say is: There is a amount of money we got,

this is the time for the project and we got resources constraints and thinks like that. So basically it is trying to sell the position and let them make the decision. In IT you don't want to tell the business what we are going to deliver you should deliver what the business wants.

14:25 Different stakeholders have different viewpoint on the requirements. They stakeholder probably have the same requirements but tell them in different ways. Are you using techniques to solve this issues? For example by the use of visualisation: Process maps etc…
We use process maps. We also use various processing documents, like RADs (Roll activity diagrams). Basically that allows us to map the processing, map out the interaction between the system and the user. And move the process around to try to find the most efficient way of processing.

15:24 Once you created those process maps. Do you see an improvement in the understanding of the stakeholders?
Visualisation is always better then trying to express the process verbally. Everyone understands it. They say a picture can tell a thousand words. You can look at a picture and that is what document diagrams are. Look at the process diagrams, and the user can automatically understand the processes because of the lines, because of interaction points. If you try doing that just verbally, it doesn't make a lot of sense. By actually seeing it, it makes a lot more sense.

16:16 So you think it is important that stakeholders understand the requirements?
The reason why it is important is because; obviously the stakeholders are the sponsors or the key people in the project. So they need to understand what they are going to get. Is the recommended solution going to provide business value to my external customers? It is obviously very important to for the stakeholders to understand the requirements. And I guess from another perspective, it avoids surprises and scope creep. Having the stakeholder to understand what we are going to deliver, makes sure that everyone is on the same page and there are no surprises and scope creep. Having the stakeholder to understand what we are going to deliver, makes sure that everyone is on the same page and there are no surprises. The worst thing what you can do is withhold information or withhold things from the stakeholders, because then you are going to create a disconnection between the stakeholder and IT.

17:31 How do you determine which requirements are critical and which are not?
It depends on what deliverables you are trying to provide with the system. For example if you are doing a write, book and issue. If you are making a write, book and issue system. There are three key things there. You want to write a policy. You want to book a policy and you want to be able to generate the documents. These are the three obvious critical requirements of the system. And then you also look at like; Is it build for efficiency? You know the critical things that the system must provide. And really your requirements are determined by the business representative. The business representative is closest to the business product. So obviously they would say I need these three things. It is the responsibility of IT to look at those three things and say; Ok we can do this and this but we can't this. The reason why we can't do this is because of these reasons. If that third thing is critical then it is better to look at a way of how to do it. How we would deliver on that requirements, but still meet the requirements of IT, for example.

19:24 How do you validate the requirements of the stakeholders? How do you know if what the stakeholder says is true?
There are a couple of techniques. You can paraphrase. Basically, during the workshop, you may be talking to the business and a lot of ideas are coming out. You need to validate this by paraphrasing. So for example; If I understand correctly… and then you paraphrase what the user said to get conformation. The other way to do it, is by having a formal document that outlines what you are trying to achieve. It maybe a project charter that basically outlines all of the functions/things that are within or out of scope. And then you need to sign it off from the stakeholders or from the sponsor to say: Now our understanding of what you are trying to achieve

is this… and we are not trying to build this…. So that everything is quite clear. There is a lot of transparency. If you don't put it in the out of scope items, the user that they actually are going to get it. And you don't want to come to the end of the project and start asking what about this functionality…. Well we identified that in the beginning of project and as in the agreement this was out of scope. So there could be no surprises. The scope is quite critical, because that determines how long your project is going to be, it determines the costs because building a lot of functions into the systems that is going to impact your costs. If the system is very complex it impacts the quality if the build is not good.

21:41 How do you determine how big the scope is?
It is again a negotiation with the business. The whole process of the functional requirements is to identify what the system is suppose to do. Those things are decided upfront, and again are governed by, for example the business will give us a list of things what we will have to do. We will then determine what the costs of these requirements are. What will it costs to deliver this? You give a budget to deliver a project, so obviously if the budget is less then it the cost to deliver it then negotiation process has to take place, because you cannot build something if you don't have the money to support it. You need to avoid scope creep and that sort of things.

22:53 What is actually the most time consuming part in the requirements process?
Probably the very first; Building the requirements. Unfortunately we often find that the business don't really know…. They are very good sales people, but not when it comes to the processing site. We generally find that we are going to a little bit of a pain to make sure that we have all the functional requirements documented and it is clear to everybody what we actually are going to deliver. That is from my perspective. And partially if you are not transparent with business about what we are going to deliver the next pain point is the surprise that come in the end. The business believe that they are getting one thing and finally they will getting something else.

24:08 Is it possible to make decisions later on in the project?
Yes it is possible. If there is a decision that is going to be made later on in the project, we then need to consider if we are still on track. We need to look at, if we do this, have we got the money, how is it going to impact the project timeline? Are there any quality issues? You know those sorts of things. It maybe a negotiation process, because we may have defined a date for the project and if the business is rating additional functionality then we may have to drop other things to still make the deadline. So yeah there will be some negotiation as well.

25:00 What happens if a project gets delayed?
There is potential financial costs or reputation costs. But generally if we agreed on a date and if there is a delay, there is slipperage in the project, we make things transparent with the business. So that the business is aware of the delay and its impact on the project. The last thing that you want to do is trying to hold that stuff from the business. Because again you get to the end and you are not transparent, you are not demonstration that you are a partner with the business. You have to make sure of a strong communication.

26:03 Have you ever considers of using a collaboration system to gather requirements? Like a wiki tool?
To be honest we haven't. That is generally not our mode. But on the other side, it can be fruitful because we are widespread and there are a lot of tools that will allow us to collaborate effectively. If we look at the other strategic projects, another system that has been developed with our office in Canada, we are using SharePoint to communicate the requirements between the branches and that is essentially working quite well.

27:09 Have you ever had project where you had "wicket problems" where requirements are changing all the time.
Yes many times. There was one project were a lot of money was spent because we didn't mile down the requirements, we didn't prepare a scope document, we didn't prepare a project charter. We didn't followed the standard project methodology where we look down the business. So we

got to a reactive mode, we were trying to deal with every single request that was coming in. This is very difficult because you can not measure the return on investment. It just keep dragging out to you can not take it into the market place. The way how you can deal with that is to make pretty hard decisions and saying this is what we are going to deliver. If it is not going to provide any business value, then decide whether you want to continue with the project. And unfortunately we had to sold in that situation. We cancelled the project, and a lot of money was spent unfortunately.

29:15 Is there a standard project methodology at Chubb?
Basically we are following the standard SDLC (software development life cycle). This is a set of phases where we need to go through including testing and releasing, migration testing. That is the methodology we use.

29:45 Are there some document about this at Chubb?
I think if you jump on the Chubb net and have a look at the IT pages. You will find a lot of information. Get to the homepage and type in IT, and it will take you to the IT homepage.

30:23 What was the worst project that you had to do to gather requirements?
The one I just mentioned. It was a very stressful project. It was quite de-motivating. Nobody liked it to get into a reactive mode, because things were get to out of control very quickly. That was the worst project. A lot of time and a lot of money was spent. If you look at it objectively, the right decision was made because this wasn't going to deliver the business value that was hoped. When you are developing an application you need to make decisions like this.

Appendix F4: Interview with the Project Manager of the ARCH project about requirements engineering at Chubb.

Date	20-06-2011
Time	10:00 – 11:00
Location	Chubb – Office IT manager ARCH project
Medium	On site (Sydney)
Project	ARCH project

00:25 Can you tell me what kind of a company Chubb is?
Chubb is a general insurance company that involves commercial, personal and small business. We have 10.000 employees world wide in Chubb. We are an old company, established in 1882 first to write marine business. That was the first business that we wrote, and then we abandon it to other lines.

01:20 Can you tell me what kind of project Chubb is doing? Related to IT?
The IT project is constantly going on worldwide and they are divided up by regions or zone where Chubb operates. The majority of the projects are in the US, because we are an US company and that is the biggest portion of our business. Out here in Asia Pacific region, the ARCH project is the largest project that is going on. Probably for the next coming three to four years. It is very large, we are staring in Australia and then when we finished here in the end of this year, we move on to the next country which should be Singapore and after that Hong Kong, we are going throughout the whole region.

02:35 What is your position in the project and company?
My position changes based on the project, I have been in Chubb for 33 years next week. I started out in underwriting; I was there for maybe two years. Then I move on to IT. In IT I moved, from the ground up, so I started as a programmer, developer, and then a business analyst for a long time, then I became a project manager/program manager for different projects. It depends on the project, on what aspect I actually take on. In ARCH I'm a project manager. I am managing the business aspects of it, for Australia right now, and then for the other countries as we go forward. I'm not managing the development; it is more business requirements and then implementation.

03:40 Can you tell me something more about the ARCH project?
What we are doing is; there is a need to replace the technology that we have. There is something that is called the PRISM system. That is used to support the underwriting, claims and finance. It used to be a single package system for those three. Five years ago, we took the package finance out and replaced it with better technology that is globally at Chubb, it is called GFS. Claims right now are no plans for, but ARCH should actually replace the underwriting aspect of the system and not just replacing it but also providing a lot more functionality that they have today. The idea is; is that it is not just a technology, it is also that the business is growing and the cost of underwriting right now with the old technology is always the same but the premium is getting much less for the policies. Thus we need to find a way to service these policies with much less money and ARCH is actually providing this to do that.

05:15 That is probably also the reason why market facing technology is involved in the project?
Yes, that is another aspect / another feature to get the costs down.

05:29 According to, what is a good requirement? What makes a good requirement?
ARCH is a multi-national team, we have people here, we have people in Canada, we have people in the US and we have people in Singapore. And everybody needs to be able to read the

requirements, to understand it in the same way. So a good requirements is where the writing is good, people do understand it, lots of pictures and you state the obvious and addition to things that may not be obvious. Just to make sure that everybody understands it the same way. In the end of the day, good requirements are; you know what is a good requirement is if what is delivered from development is what you need. You only know that in the end, so you kind of have to wait but I do look for certain features.

06:36 What kind of approach in terms of requirements engineering is used in the ARCH project?
We have two types. You are familiar with the kaizen tools that we have here? Yes. Ok, This is the agile portion; the Canadian team have taken the kaizen tools and reverse engineered them into Advantage. So they are reverse engineering the data captured for the risk analysis, all the rating and all the policy documents that we have in there. But there are certain requirements that are needed in Advantage that kaizen do not support. For those, we have a Business Analyst, taken the old methodology. She is going to interview business people about things of; how do you make a decision of when to attach certain policy documents. It is mostly around policy documents and providing additional requirements on those.

07:59 That is more or less the plan based approach? By gathering it and then implementing it. So it is not like an iterative approach?
The prototyping idea is done with the kaizen tools, reverse engineering. What the Business Analyst is doing that should be final. But I think what we find is; Advantage is very structured, and it is structured in a way that we not always understand. So we come up with what we need and they come up with questions that we never anticipated. In that sense it is iterative, we have to manage the questions and answer process.

09:02 Are you using a standard methodology or a standard framework for this?
For the work that Business Analyst is doing, we have standard templates. That are templates that Chubb is using for gathering requirements. Although it is very broad, it kind of makes it what you need to make it.

09:35 Which techniques are you using to gather information from the stakeholders?
Yes we do use interviews that we do with the underwriters and also operational staff. For documents we do case scenarios, we are getting examples of documents that they already issued because they bring out additional issues that the generator might not think of. So you say; give me your most complex policy and then you look through that and you will see exceptions.

10:08 Do you also use other techniques to retrieve information? For example in the iClose project they did a market research about what the competitors are doing**.**
Not for this aspect of the project. This is basically taken the kaizen tools; how you operate today, and you have it as a base and I think the market research comes in when you want to add more addition functionality like the market-facing. But that is the second step.

10:50 What is actually documented in the project?
We have something called a project charter which is your project definition. We have things like issues logs, we have status reports we meet on a weekly bases as project team worldwide. The status reporters log the issues and action logs, everything is documented that way. We have budgets that we have to meet. There is budget documentation, feature enhancement lists, so things that cannot be addressed in this stage. In the implementation we put on a log for future development. There is lots of documentation in that world a lot of artefacts.

12:00 In that sense you can see that the ARCH project is definitely bigger than the iClose project.
Yes, it is bigger and I think due to that it is international have to be very careful that everybody understands the same thing.

12:15 So keeping consistency really matters in this project?
Yes it does. Chubb has a standard SDLC system, they have a standard process and they tell you

which documents you should produce for all those different stages and we are trying to do all of those things.

12:47 In the project the SDLC methodology that is provided by Chubb is really used?
Yes it is really used. We will be measured against it because it is a large project. And the money for this project came from the highest level within Chubb. So it is not a local thing, it came from the upper executive committee and they gave us millions of dollars and you manage that now and we will come back to us and make sure that you are delivering what you said you will deliver. Everybody is making sure that we are following the steps and if there are any concerns or issues, they are flagged immediately so there are no disappointments in the end.

13:52 How are the stakeholders chosen in the ARCH project?
We have different committees. We have an executive governments committee, that are the executive leaders of both regions, so the head of Canada and the head of Asia Pacific. Then you have a government committee that meets once a month and deals with any issues. And there you have the business leads, the underwriting leads for the zone. As well as the underwriting leads for Australia, so you got those people. And these are all stakeholders and then the people that we work with, again that is the lead underwriters here in Australia, we go to them first, and if they do not have the expertise to answer my questions or something like that, we will assign someone from there staff to work with. So it depends of the class of business that we are looking at in any point in time to how you need to get.

15:15 So it depends on the problem, the time and etc**…**
Yes, so if I'm doing property than I need to know who is your property specialist and the same for causality, or INT.

15:31 Do you also consider the involvement of the developers and expertise sitting together?
It is difficult because all the developers are in Canada and due to the time difference we have issues. But it does happen, and that is when we show prototypes. They are showing a prototype for the first time, then we sit with the business and the developers are on the other end, by web-ex, and have a conversation about what they see. The developers are sitting there and there are some difficult functions that we are trying to originate in Advantage. For that, we are going to have the developers sitting in on that discussion with the business. So there are times when you put them together but not for every discussion. A lot of the discussion is done by me and the Business Analyst by talking to the business people here and then we talk to Canada separately.

17:10 Sometimes there are conflicting requirements. How do you address these requirements?
Conflicting requirements, we are trying to solve here locally. I get the lead underwriter/managers involved and say; you can only have one way of doing this and you have to agree. That is how I managed it locally. Sometimes you have conflicts that you come up with requirements that nobody ever considers, that are brand new, and in our project definitions it is said; this is the scope of our project. And we had a number of instances that somebody identified something that was never discussed before. Right now we are trying to see if we can fit it in. If we can't fit it in then the discussion goes up to the government committee and they say, I can do this but it means we have to draft off some of the other things. So it is a negotiation in that way. It is all handled through a government process. We are not sitting here in an individual level arguing with our self's. It have to be that it either fits in the plan or to drop that date that is must be done. If you do you get extra work, then you have to move out the date.

18:52 Do you think it is important that all the stakeholders understand the requirements?
It depends, because not all the stakeholders are going to understand the requirements at a very detailed level. It is important that they understand the key issues. But not how the new aspect works, they are not going to understand that. I think that most people have to understand; this is the function that I need, this is why it is important, for example I have 20% of the book fallen into that so you need to support it. You need to bring it up to a higher level, so all the stakeholders can agree on that it is important or not without understanding the detailed functionality. Hopefully

they understand insurance enough that they know what you are talking about.

19:57 For developers it sometimes is hard to understand insurance isn't?
Yes the developers don't understand the insurance business and that is why the requirements are so important. What we are trying to do is to tell them from a business perspective why something is important. How the process around it works, all the things that they can't see. Let them come up with a technical solution how it is going to operate and then we have to validate it into in the picture of the users about how it should work. Yes you right, the developers do not understand the business.

20:55 How do you determine which requirements are critical and which are not?
It is based on a couple of things, like how many of your policies are impacted. How much time it will take you to make a work around if something doesn't exists? What is the impact of having to do something manually? That makes it critical. I think that that are the two big things that I could think of.

21:40 How do you validate the information from the stakeholders?
There are different things. We are trying to look at from different angles through different people. The managers are not detailed people so they tell something has to work a certain way. What I do is; I go to the people who actually use the system today and ask: how do you do that and what do you actually see? I look at data in the system to see how it actually sits up there to validate about what is going on. It is getting the information from different sources just to validate.

23:17 How do you decide what has to be in the product/solution and what can be left?
There was an agreement at the start of the project to what functionality will be there. We accepted the solution, following the operated in Canada; we accepted it with the understanding that it provided certain functionality and we expect those things to be there. If we are trying to add something to it then we have to; prioritize, like I said for example the number of policies that impacts, the workarounds, the efficiency and things like that. It is the same thing as, if you leave something out; you have to know what the impact is. What is the impact of the business by leaving it out? What does it cost to leave it outside of the system? That is what drives the priority.

24:35 Are you using special priority mechanisms for that?
No, we have various priority lists. Some of them are saying which insurance products you are going to put in. All the insurance products here in Australia are written and prioritized from one to twenty. And then we have additional functionality they are trying to put into the system, and again that is prioritized based on the business need. We work to those lists and then it is the priority list that goes into a plan.

25:30 For example a mechanism that is addressed in the literature is based on a WIKI system. By counting the modifications, views etc… it determines how important the requirement is.
No, the requirements in this project are too complex for that. They are very big requirements and different types of people of the business which have different views. So it needs to be discussed and negotiated. It is a verbal process. You can use it, for example if you have a dot.com site and you want some improvement. It can be used for something like that. But we are trying to improve the business, where different minds are coming together and figuring out the right answer.

27:32 What is actually the most time consuming part in the requirements engineering process?
Waiting for the users for an answer on the questions that I ask. But I guess; there is a lot of documentation. Getting the document right is the longest part. But a lot of that is the process of waiting for the users to get to know what you need. Nobody is around; it is a busy time etc…

28:19 Is it possible that you can make decisions late in the project?
Yeah. We have a small example of that now; where the prototype moved along after. We saw it but it moved along for two months and somebody else in the US looked at our structure and said; you are not doing something correctly in terms of how we are storing the data. It is a little late to

go back and change you data structure, it is one insurance product but we get it right as we are going forward. But it is a lot of rework and now we have the question; what is the value of reworking that one insurance product. Is it really going to hurt us in the end? So you do get things like that. People are giving their opinions late and that is an aspect of this project because there are so many people involved and they are not all working at the same page. Some of the people in the front line are very clever to get the things done quickly and the people in the US are sitting back and are looking at it in theory and they are not in charge in getting the things done. So yes, it could happen.

30:18 Did you ever had projects where the deadlines were delayed?
Nothing is delayed yet. But there is a lot to do

30:40 Are you using a collaboration system for the requirements engineering?
Not for requirements but as a place to store things we are using SharePoint. It is a place to store the documentation and then there is a section where you can record an issue and you can respond to it, so you can see the trail of communications. We are trying to use that instead of email, because a lot of time in the email a lot of things get lost or not everyone is copied in. For me as a manager, if I'm not copied in, in a low level of discussion and I would find people are talking to things with the Business Analyst, that they should have that or they are going down a long path then I wouldn't have known. It is just email and you would never have seen it. So a type of collaboration thing where you can record the conversation is quite good.

31:59 How do you address wicked problems, Problems that are changing all the time?
Yes we have that with the kaizen tools. They are constantly changing. We take them, we reverse engineer them, we go off and building the new system: Advantage. In the mean time the kaizen tools over here are making some changes, for example to the rating or new documents are being added. We try to get all these changes get over to the Business Analyst and she tries to get them into the new project. But I think it is becoming a real problem because it means you will never finish something that is constantly moving and we need a finish date. We are going to have to somehow freeze that kaizen world. You have to manage it, either you freeze it or you say that there is a period in the end of the project where you go back and do a compare to make sure you have have everything in there. That is what we will probably do with the documents and we will freeze the documents and we will certainly do a comparison in the end of the project on the wording.

33:40 What was the worst project that you ever had to do in gathering the requirements?
The worst project that I had to do in terms of requirements engineering was before I came here. It was project that I had to do in London. We tried to open up a Lloyd syndicate. In Lloyd you have a very specific way to conduct your business and also the systems you have to use are very specific. They didn't tell you which one you have to use. There are a number of systems on the market that would satisfy our needs. The problem for us was; we didn't understand how Lloyd is operating. We didn't understand what they wanted. So how do you evaluate those systems and then you have to pick one and you have the whole problem of bringing it into Chubb and integrating it with other Chubb systems. For me that was the hardest thing, nobody at Chubb had ever working in a syndicate and we didn't know what Lloyds wanted. So where do you get the requirements from? We actually had to hire someone who is really good and had managed syndicates before and managed the systems. And we brought him in as a consultant and in the end to actually run our operations in the syndicate. You get an expert, if you don't have an expert you buy one. That was expensive, but it was a right choice because we couldn't learn it quickly enough. And I'm sure that even after a couple of years of running now, they are still learning because it is such a different way of operating.

Appendix F5: Interview with the Business Analyst in the ARCH project about requirements engineering at Chubb

Date	29-03-2011
Time	15:00 – 16:00
Location	Chubb – Office Project Manager
Medium	On site (Sydney)
Project	ARCH

00:51 Can you tell me what kind of company Chubb is?
I found Chubb really friendly and using more technology than other insurance companies. I am here for two years and worked at the OPRHA project. That failed due to some reasons and than I worked on the Kaizen tool (ARCH project). There is one set finished and we have installed that and now we are going to the next set. Finally it will be rolled out practice. This project is called the ARCH project. I believe that Chubb really is technology wide. In general, the people are nice and it is a good place to work.

01:59 Can you tell me something about the projects/services the company provide in terms of time on the market, customers etc…?
Well, I'm not directly involved in the market facing technology project. But I know that some product are currently going into market facing technology. For example currency vortex, where the broker can quote or push a submission to the underwriter. That is how it works, so that we are connected with the broker on that level. Currently there is one for CSI, where the independent broker manager is working one the market facing technology and we have plans for ARCH as well to get into market facing technology. Where in the broker can request small quotes.

02:54 What is your position in the company?
I'm a business analyst. My main task is to get requirements, to identify the priority and to see if the requirements is reliable or valid. But sometimes that is not possible. You have to compromise or identify a solution for that. My task is to identify that, I also provide information to the technical team in a way that they can understand and design something. When the product come to testing, I have to arrange the meeting with the business and to provide a presentation and to do testing to is to see whatever it works or is in line with the requirements and I do the take up.

04:03 What is a take up ?
A take up is, when the system is going to start in Chubb, we need to put some data in. That is called a take up.

04:28 What is actually a requirement to you?
Think it is to do the business tasks more easily. It can help them to improve, in terms of systems, things that they do everyday and that is like taking to long for them. The requirements are aspects we can build into the system that can help to save time and give them more efficiency. I think that is called a requirements and you can build those requirements into the system. Some requirements are as well like; I don't like this colour. They are also requirements but they don't have priority.

05:46 How is Chubb gathering requirement like in the term of the ARCH project?
What we do is looking into the Kaizen tool (existing system) and that has all the basic information for the product that the kaizen tool is developed for. We sent that kaizen tool to Canada (development team), but before that we do a demo about how the kaizen tool works, like; this is the way how we used the system and this how we put information into it. And then we send the

tool with all the information to Canada. They are trying to analyze the tool by them self and they identify requirements and say OK this is how it is working and they try to find any gaps. They will document if there is a gap, and then they sent it to me. I will find the gaps and go back to the business and say I tried to do it by myself, if I know I will provide the information, if I don't know I will go back to the business and say will this is something that we find that is not right or this is defined as a gap, what is that? And they provide us information and we build a prototype based on that. When prototype is build we give a presentation. That is how we give the business the chance to show them how the prototype is working. Then they try to use the prototype and they suggest; if it is working the way they want or if they want to change something or if they want to remove something, that kind of stuff. It is based on an existing system and it is a ongoing process because when you have a prototype you still need to gather the requirements because it is not clear enough or just not right. You need to go back to the business and give the requirements again.

07:59 Do you have special tools or templates for defining the requirements?
In case of a question or a problem we don't have any kind of template but we use SharePoint. We have to create and log issues about this and that, or how to do that and then it is sent back to me and I put information provide a response about how we address that issue.

08:54 And SharePoint is something like a forum?
A kind of yes, you can create your log up there, you can upload documents and add suggestions.

09:18 How do you choose the stakeholders in the project? There is a project and you need people how do you choose the people?
It is not my responsibility/duty to choose stakeholders. Currently I know that we have the SBU managers, and the persons who know the project better (specialists of the kaizen tool).

09:50 You said something that if you get questions from Canada, you go back to the business?
Yes but we don't get back to the persons, I get back to the SBU managers. And they need to answer the questions. If they don't know, they can go back to their team and ask the questions.

10:15 Thus the SBU manager is more or less the stakeholder?
That is right, yes. We have the operational manager as the stakeholder and the IT manager.

10:35 What does the IT manager do in the project?
IT manager is the stakeholder in terms of the other process system (PRISM). He need to give the information that is demanded by PRISM.

11:05 And he is responsible for the requirements of PRISM isn't it?
Yes that is right, he is responsible for PRISM. If we need to do something with PRISM, he is responsible for that. Another stakeholder is the operational manager. We need his input to identify the workflow, how it is going to be after the system is implemented. Can we skip a step, or can we add something, for all this kind of stuff we need his input.

11:34 So it is more or less the whole overall process of it?
That is right yes.

11:40 Reflecting back to the requirements again. If you get a question back from Canada, like a questions of: I don't know how this is' and probably you can end up with a problem that stakeholders have different opinions about that. How do you solve that kind of conflict?
It happens that like if one stakeholder said he want the product to look like this and the other one said I don't want that I want it that like this. Then we are going to find that out, one step ahead. We had that, and we went to the manager (CSI) of them and said, well this is the issue; do you want it this way or do you want it that way? Currently for Australia, I think we normally follow the home office pattern. For example if they use that, then we are also going to use that. We do something similar, can we use that one and make some changes.

13:04 What is actually the home office?
The home office is in the US, it is the headquarter. Sometimes, we can't say we are following you or we are going that way, but it will give more bias and more distance. That is why you just say, if something happened, then you go to the top level manager and ask for his opinion.

13:39 How do you determine if the input from the stakeholders is correct? For example a stakeholder can say, this has to be in the system. But how do you know for sure if that is correct?
Sometimes, it happens that they say I want that, but it is not 100% sure that it is important, or it is not 100% sure that it may affect the process or something. We need to find that out and to see if the data is stable / correct. For example; Someone is saying I need to build risk analysis and everything, but without building risk analysis you are not stopping doing business. You're spending time on risk analysis, while you should spent time one documentation and rating, because if you don't have that, they will stop doing the business. You need to identify two things. You can delay on risk analysis, but that is for your information, (but rating and etc. is helping to calculate the premium) and thus that is the document that goes outside. You need to focus on the goal of the project. I just give you the very big example. Another thing happens; Someone says: I don't like this way (numerical), I want to have the list of things by alphabetically (another person is saying; I want to use it by numbers). Then you need to see what is more appropriate, and what saves time, you need to consider all of the factors (effects). Based on that you have to make a decision, I can do it this way but then the disadvantage is that and you can provide them options. By this is option 1, option 2, option 3 when they agree the advantage and disadvantage. They can say I want to go for option X.

15:55 I think it is called a benefit cost analyst isn't it? You look at what is the benefit of it and what are the costs. If this option has more benefit , that we choose that one.
Yes that is right

16:10 How do you determine which requirements is critical and which one is not?
You look at what has the biggest effect on the business. You can also think about money, time, and the resources as well. All together, time, money and resources determine if it benefits the process and if it is useful. You should also consider testing, and determine if it is inline with the scope of the project.

17:00 Sometimes you get the requirements, but the requirement is sometimes not complete. Do you have some analysis tools for that or are you validating the requirements on that aspects?
We do define requirements in that way; let's give you a real example. Someone says we should have a multiple currency function to this particular policy for creating the premium. You can have multiple options, lets say A, B and C. One is Australian, B is US currency and C is Japanese currency. Someone come to us and say we need multiple currency. And it is very hard to build this, because that you need to add external sources to define the currency etc.. So we got the requirement, we needed to go back to the business to find out how many policies we currently have in PRISM or any other source that use another currency? We found 7 policies that have multiple currency. So this is something that is not useful to make. You need to spent I think one month to put it in the system and it is not worth to do that. So it is better to do these policies manually. So we decided that requirement is ok, but how many policies do you currently have and based on that it is not worth. So we go back to and said there are only seven policies that have another currency and that is not worth to build it into the system. You can better leave it outside the system and do it manually. And then they agreed on that and said it is fine to leave it out the system. So for validating we go back to the historical data and see what is there and what is not and if it is ok, is this a valid requirement.

19:22 How long the does the ARCH project takes?
According to the current schedule, our plan is to launch all products is the end of February next year (2012). So we will come up with all the documentation and everything by the end of this year. We started the project in February, and we started with the first product INT and D&O from

CSI. We are almost done with D&O and created a prototype and we are already done with the INT prototype. I approved that one last month. Tomorrow we will do the D&O prototype and after that we will do Crime. So we are going really fast with the requirements engineering. We are still on schedule.

20:41 What kind of issues this you have in the requirements process?
The issue that I found is the lack of insurance knowledge. Because the software that we are developing is basically for the insurance industry. Therefore you need to have some knowledge about it (from the developer point of view). I found that some people don't understand how it is working and it delayed the process and you need to be very consensus and very precise to provide the requirements. You need to specify everything, you can't assume that the technical part understand what you are trying to say. Because the builders of the insurance software come for 50 percent outside the insurance industry.

21:45 So it is more like the communication between the Chubb and the developers? Because the developer don't understand the processes of Chubb and the meaning of the definitions.
Yes that is right.

21:56 How do you solve those conflict?
We decided to use SharePoint where we put the information in but sometimes it is not clear enough. Therefore we also decided to do a weekly meeting and presentation through a Netmeeting, where we sit around to and saying: well this is the way how I see it. For that we are making charters and design templates. I demonstrate them how the solutions should look like and ask them: "What do you think of the design?" That is how we work. Sometimes they come back to us and saying; Well that solution is not good, it is not able to support this or that. Do you have another option about how we can handle this?

22:58 Did you ever had project that were delayed?
Yes, I was working on another project and that one was delayed. It was delayed because of complexity. The developer didn't had enough knowledge of the insurance industry, it was an outsourcing project. They didn't understand how the current Australian industry worked. They mainly were addressing small insurance providers. So it was a big project for them to build and because of that they were not able to provide a proper solution. The system was more like a static system where you can't make any changes. So if you wanted to change something, they had to rebuild everything. I think that was the main issue and also the lack of insurance knowledge.

24:00 Did you every consider a collaboration system? You mentioned something about SharePoint which you are using as a collaboration system. Does it also have some requirements validation/templating?
I don't think that SharePoint has any kind of that functionality. In SharePoint you can prioritize the requirements by your self, saying this is high, this is medium and then you can generate reports from it. Saying for example that high requirements should be done on day one, or day two.

25:14 So it is a system that is specialized for requirements engineering?
Yes, it is more like a log system where you create a log and a table with as many column as you want, saying the priority, severity or whatever. When you create a new log you can assign a priority and then you can have a discussion about that. You can generate a report from that, that is stating the high priorities where we should work on first.

26:10 How would you address "wicked problems" / problems that are changing all the time?
Yes, we decided not to go to one people but multiple. You have to go to two persons that know everything. One person that knows it in very detail and it saying for example we need to build everything. The other person is then probably saying, we do 80% or we do something that is enough to issue a policy for the rest I don't care. If we address everything, that makes the system complex and mind that we have to go through all screens and that consumes more time and the

some aspects can be done outside the system. So there are two types of people within the department. If you go to one, he will say: ok I'm fine with that and then we go to the manager and the manager will say: ok it is done and once he said that, he will sign it off and say ok that is it. I think in this way we do minimize the dispute. The other one is like the example you said. When something is changed or it will change (the rate or something), it is more a design question that the designer should understand. For example how the requirements are specified by the business analyst. Providing the information like this is something that will change after some time. So you will have to make the system the way that it will allow that change. For example to make the rating changeable, it should be simple so that it can be changed and updated afterwards. So you need to provide that information in advance, saying that you need to build it this way. By providing that it will probably change after some time. Then they understand that the requirement will change after few months or years, and they can make a database or system that will support this change.

28:58 What was the worst project that you ever had to do in terms of gathering the requirements?
I think it will be hard to gather requirements for any specified system from a specific department if you don't have any information or knowledge on that system. Then it is really hard for you to go to gather that information. I had that problem, I wanted to get out some reports from PRISM and I did not know how PRISM look liked. It was hard for me to get the information out of PRISM. You have to contact someone who has the knowledge on that.

Appendix F6: Interview with the IT Manager about requirements engineering at Chubb (ARCH)

Date	15-06-2011
Time	15:15 – 16:00
Location	Chubb – Meeting room 3
Medium	Telephone (Melbourne)
Project	ARCH

00:25 What is your position in the company?
I'm the information technology manager for Australia

00:35 What is your position in the ARCH project?
In the ARCH project I've got a very limited responsibility. The reason for this is because we have employed some very strong project managers to lead the project from the Australian perspective. My role is now more on a consultant bases by looking at various issues and risk management. Things like that.

01:16 What is a good requirement to you for the company and the ARCH project?
It is very similar to our early interview. Where you make use of various tools. Thinks like process diagramming or role activity diagramming, descriptions about what the system is supposed to achieve. I said by diagramming, a picture can say a thousand words. A good requirement includes sustained diagramming, the use of diagramming, to get the message across and writing a thousand pages is not a good use of time.

02:28 What kind of approach are you using in the ARCH project? Is a plan-based or prototyping approach?
To be honest the ARCH project is very different to the other projects that we have done. Our strategy in this project is a list drop strategy. The system used in this project is a commercial launched product that is already used in Canada. What we are planning to do is going to use a list-drop strategy by taking the applications and role it out in Australia with as little as much customization as possible. What we want here is; we want to lent the technology that is used to support a much bigger proper business then in the Asia pack.

To extend on that, basically the requirements gathering is… we certainly are going through that process because we do have to change the systems to achieve a multiple support and to repertory the requirements for what we have in the other Asian countries. What we are planning to do, and we have done this twice, is to demonstrate it in an iterative approach of the development lifecycle. So basically we develop something of the system and we showed it to make sure that they agree with the changes that we made. By keeping in the mind that the whole strategy is a lift and drop.

04:25 Reflecting this back to the iClose project, this is more a similar approach isn't it?
That is right yes

04:37 What other similarities does this project and the iClose project have? I also saw something about market facing technology.
Yes there is market-facing technology involved. The market-facing technology is actually on the renaissance application which is the personal line application. We don't have market facing technology on the commercial lines package. There is a distinct difference there.

05:14 Are there also some differences in the approach of requirement engineering in the iClose project and in the ARCH project?

The big difference between iClose and the ARCH project is that in the ARCH project the systems already exists and in the iClose project we started from scratch. Once you got systems that does exists, you can elaborate on that to build the market-facing technology because you already got the shell there.

06:25 Reflecting this back to the interview, this is the requirement gathering technique that you are using?

There are a few other distinct differences between iClose and the ARCH project. In terms of the requirements gathering process, for the ARCH project we actually build a number of excel, fba, and word tools, and I think you would have seen these tools. These tools form the basis of 80% of the requirements, because those tools have been build around of what the business was actually looking for in terms of the rating, calculation and document generation. And what they are essentially trying to do is to reverse engineer the current tools into the new system. So they know what the business wants, it is functioning in the excel, fba, word tool and we need to build that into the new system. Therefore, most of the requirements are already done and need to be reversed engineered.

07:52 And how are the other requirements gathered?

We have identified a number of big gaps in the Coby tools, these excel, fba, Word tools. For advantage, the issues was really to legislate requirements, for example the ARPA requirement. Which is the local authority that monitors the financial institutes in terms of what business they write, and reviewing their integrity and financial positions and things like that. Obviously we need to get a report on what we are doing in terms of our products. We had to write a number of functional specs that had to pass over to the Canadian development team and they are going to develop these requirements into the system.

09:04 What is actually documented in the ARCH project? Are all the requirement documented and then pass through to Canada?

The Coby tools are weakening away as the business wants to work. They use the tools to generate the documents. So the document output is already there. Having that tool they can reverse engineer it into the new system. The tools are going the help with the change management. Because people are already used to the Coby tool, so we can replicate the functionality into the replicated system. People will easier accept the system as well as the amount of training that has been involved. The change management isn't that big because they are already used to the various screens and used to the functionality and that sort of things.

10:15 So it is more that the developers get the whole system and then they are trying to rebuild it?

Yes that is right.

10:19 How are the stakeholders chosen in the ARCH project?

The stakeholder are… essentially easily recognized. You got business managers that are interested in making sure that the system meets their requirements. Your stakeholders are easily identifiable. By the APRA project, we already knew who the major stakeholders are. On top of that, because the APRA project failed, we have now a lot more PMO oversight from the US. We basically call that office the EPMO, which means the enterprise project management office. And they make sure that we are actually following all the requirements of good project management qualities.

11:40 Did you also recognized some conflicting requirements during the development of the ARCH project?

Well, this is not an easy project because the changes that we made to the system apart from the mandatory litigator requirements that have to build into the system. Everything else that we do in terms of functionality we have to run pass through to the Canadian tech to make sure that our recommending is going to work for them as well. And to be honest they have to majority size. So if we say we need this particular function. They will look at it, and if they say we won't need this, we won't get it. The reason that they got the power to do that is because it is actually their system. So any changes that we made has to benefit both sides. So it has to benefit the Canadian side, it has to provide them efficiencies and that is also for the Asian side.

12:55 What if you requested something that has to be in the system. But in Canada they say we aren't going to build this into the system.
We are going back to the lift and drop strategy. We agreed going into this project that the strategy is to take their system and drop it into Asia Pacific. If there is something that is absolutely necessary, then we are going into negotiation and we are trying to get out our message across to get them to buy in. But if they don't agree with the change and it isn't going to provide any value at all and it is actually going to add development time then they will say that they are not prepared to build it into the system.

13:56 How do you determine which parts are critical and which part are not?
It is a little bit difficult because we know what the mandatory requirements are and to us they are critical, they got to be build into the system to make the legate report that is required for Australia. These changes are requested. It is more like; we do have a change we see as critical, we discuss it with Canada and look for an alternative solution. That means that it really comes down to the Canada team to make a decision.

14:52 So they are more the drivers of the project?
Yes they are and we are adopting a lift and drop strategy.

15:06 How do you validate the information from the stakeholders? This is done in the iterations as mentioned before isn't it?
It is yes.

15:42 What is the most time consuming part in the requirements engineering process?
I guess it is probably the reverse engineering part. Because we have got a team of developers over here. You know developing the Coby tools, the excel, FBA word tool. Somehow we have to depart that knowledge what they have build across to the team in Canada who are actually developing the system. And I think that is probably the most time consuming part, because even if you have a team in Canada who are used to insurance terminology in a global company, you still go to a process that you have to actually educate the developers on the products and while the Coby tool have actually go a long way in existing in that process there is still a lot of communication of how the system is suppose to work.

17:02 How do you educate the developer in Canada? Do you have special techniques for?
We actually got video conferencing now, that we didn't had when you and I discussed this questions last time in the interview on the iClose project. Now the videoconferencing allows us to demonstrate on the system they can actually see the person that is speaking and it helps a lot. You got a face-to-face meeting.

17:59 Are the deadlines in the project still on time or is there some delay involved?
I had a call today, for a weekly status meeting with the team in Canada, Asia pacific, US and Australia. We talked about where we are in terms of requirements gathering process and everything has the status of green. That is basically that we are on track. At this stage the project is looking quite good. From a budget perspective, we are a way under budget. I think that is because the Coby tools are already being developed and it is just passed to the developers and they can just play around with it and that sort of helps.

18:50 Do you also use some other collaboration system for the requirements engineering?
We are using a tool that is called SharePoint, that is a collaborative tool what we use to pass

information to and from Canada.

19:15 What kind of a system is it actually?
It is like a document repository. People can add documents, for example if write functional specification document you can add that into SharePoint and the developers in Canada can pickup that document and can continue to work.

19:36 Can they also put comments in that system?
Yes it is also good for questions to answer. It is a Microsoft product, where you can come up with a list of questions that you can put into it. And using to follow the SUN principle the developers can work on it and other users can respond to the questions they have. So you are not losing a lot of time.

We also got other tools, we also use Symtpon a quite of bit. That allows you to send quick messages to get quick responses. You can see that the other person is online so you can shout a quick message to them. (Instant message in Lotus Notus). It is quite good for getting a quick response.

21:03 Are you also using some default documentation or templates in this project?
We are following the EPMO recommendations in terms of the templates. They got an entire project management methodology which includes a lot of templates and things like that. But obviously we don’t need to use all of the templates. So we customize the methodology to use the templates that we need as part of the process. So for example; we have a functional specification template, and we use that consistently to relay the requirements.

21:48 Is this methodology similar to the prince II methodology?
It follows the prince 2 principles. They have actually developed; we can access all of the templates and other information online going through the EPMO portal. If you would like to take I look, I can sent you a link. I will send you the link after the call.

22:37 Did you had any problems in the project which are changing all the time?
Not so much that things changing, it is more the geographical challenge that we are trying to deal with in Canada. With the time differences and that present a number of different challenges. For us it is probably a less impact than in Canada because they deal in into the pool around half past eight to nine where we are dialling in at half past eight in the morning. Therefore it is a less impact to us. That sort of thing will always have challenges because people can’t make it. People got personal things that they need to see to and that sort of things.

And another challenge is the infrastructure that need to be good at both sides. That is also a sort of a challenge that we are trying to get through at the moment.

24:14 What kind of challenge is that exactly?
Well, it is more like, we can deliver a brilliant system but if it doesn’t have the infrastructure to support it. For example it still takes a lot of time to log on, we haven’t considered the infrastructure as a side of it. That will have a negative impact on the use of the system. One of the things that we are trying to do at the moment is that we are analyzing the infrastructure to make sure that the infrastructure we have in place can deliver the response time that the user is looking for. The services is going to be housed in Toronto, you have to consider the distance and the support required. When it is day up here it is night in Toronto, do we have continuous support to around the clock. That are the kind of things that we are trying to work out at the moment.

Appendix F7: Interview with the Business Partner Service Manager about requirements engineering in market-facing projects at Chubb.

Date	24-06-2011
Time	11:00 – 12:00
Location	Chubb – Office IT manager ARCH project
Medium	Telephone (Singapore)
Project	Market-facing in general

00:15 Can you tell me what kind of a company Chubb is?
I guess you got a lot of this from the Project Managers already? You have had an interview with the Project managers, have you? Yes that is right. Probably she got through all this stuff. Chubb is a major insurance company. Top 10 in the world, 10.000 employees I guess. It is established in 1882 and providing insurance to commercial lines, in health, all sources. And I would categorize Chubb in a medium to large organisation.

00:56 What is you position in the company?
My position is Business Partner Service Manager. Let me this explain, it is not necessary the usual Business Analyst, Project Manager role. It is really effective to the kind of IT management area. We had recently called a IT transformation project and part of that is like aligning resources in a pool. Historically different IT teams in Chubb have been aligned to active service business units. In other words they might be aligned to personal lines, aligned to health, they might be aligned to… you know the actual business units. And IT transformations is about taken the…. There is a pool of resources, you know Business analyst, technical people, QA and they aligned all together as a whole rather then a business unit. My role is kind of all business facing parts. So basically project management at the moment, sitting under me and business analyst are sitting under me. That is kind of my role. I'm in charge of the business facing parts of IT.

02:22 What is according to you a good requirement?
That is a good question I guess. When you talk about requirements, I mean you probably know that they kind of come in all kind of shapes and sizes, a good requirement for IT is something that is well defined and can be well measured. A good requirement in an IT project is a low level description that can be easily developed from technical perspective, easily tested against it and can be easily signed off. That is something that amplified the project.

03:34 Are you also familiar with the iClose project?
I'm familiar with all kinds of projects. I'm also familiar with market-facing projects and I can give you an example if you want? Yes

03:51 What kind of approaches for requirements engineering is used in the iClose project and the ARCH project?
Ok, I'm at Chubb for 10 years and I used all sorts of methodologies for requirements, like Agile, waterfall…. You name it, we have done it. For market-facing technology platform that we have developed in the UK, and I think that is probably a better example than iClose because it is actually live and it is being used in the market for the last few years. That is called Peoplesure, and we are rolling it out in Australia at the moment. In terms of the requirements for that project, it was done in Agile, so more in a prototype type affair, in other words we had someone who knew the business very well, that could consult with the major stakeholders and to come up with solutions without trying to provide any information. The answer here is I used agile and I had used plan-based that all depends on the type of project that you are involved in. It should depend on which way you are planned to get the requirements.

05:20 Do you have some standard documentation that you are using for this**?**
Yeah, there generally is a kind of a business spec (business specification document) and again this depend on the team and on the project and they are using things like; a business case, make users to talk about the existing system, combine all the data and change it, how do you measure those changes. I think depending on the project and certainly on the waterfall methodology than you usually use certain templates for business requirements. The obviously is about in agile is depending on the way you learnt how to do agile projects. It is less likely that you are going to use a template.

06:19 What kind of techniques are you using for gathering the information from the stakeholders?
Yeah that is an interesting one, it could be case scenarios' but I think it depends where your stakeholders are. In terms of market-facing technology the challenges of getting the requirements is much harder instead of other projects. Because most (99%) of the projects that I have been involved in are all internally focussed so the systems has been rolled out internally rather than to the market. And for the stakeholders' involvement it is all about creating relationships to those people to understanding their business. In that case it is more about; just talking to them, starting of with a workshop and from that workshop making sure you engage all the stakeholders regularly. Market-facing technology, in the example that I gave earlier, it was more about Chubb providing something to the market that was easy to use, very easy to use and very quick and efficient. So it was not only about going to the market, going to the brokers and asking; what do you want? It was more about thinking what are we going to launch to the market and what are the objectives. So really the requirements were still performed internally and then we came into testing it. Then need to have a section of your well trusted brokers that come and gave them that system and they have to comment back on the ease of use of the system.

08:17 For market facing technology, are you also doing some kind of a market research to see if there is a potentially for the product?
We do but it is less of an IT role. I guess that will form the prior business case before it gets to IT. When there is a need for lets say market-facing technology, then that need comes from the business. They probably see a hole in the market and that is when the market research is done. For that it gets a kind of position with IT and IT comes with a technical solution for that. So I think the market research, competitor research and all that kind of things are done before it gets to the IT department.

09:17 What do you document in a project and what do you not document in a project for market facing?
The proper answer is that you should document everything. It depends on the kind of methodology you use and obviously in agile the system becomes the documentation. In waterfall methodology you should be document everything. It all depends on the size of the project. I ran projects before, where every single requirement is documented and then labelled. Like you got here, one large requirement and that is broken down to a requirement tree where everything is documented and could be tracked during the lifetime of the project. But I think in reality, if you actually want to complete an IT project you can't do that for a large implementation since it will take forever. That is why you kind of make sure the major part is documented and the system almost documents the rest. If look at the reason for documentation, what is the reason? There are few major reasons, that is; one is to let know the development team what to develop and the other one is that the business know what they get and to sign off on it and the third one is that the QA team can test on it and sign it off. That is kind of the three main reasons for documentation. Depending on the team, depending on the kind of relationships with them, that is kind of what drives what you document.

11:02 And what is actually QA team?
Yes that is the Quality Assurance team, the quality team for the testing the actual system to make sure it works.

11:11 How do you chose the stakeholder in a project like a market facing project?
It is on experience actually. The stakeholders are the people that have got an effected interest in the success of the project. It comes from a different point of view. If you are looking at market-facing technology and actually that covers a broad range of all kinds of projects. It really depends on the type of project but usually it will be that someone as a sponsors, someone that is actually paying for the system, also someone from various departments that is actually going to use the system. It is really a kind of everyone that has got an effect interest to the success of the project and so you have got an accountability to make the decisions.

12:15 How do you involve those stakeholders during the project?
It depends on the level of the stakeholders, you've a kind of a stakeholder and a senior stakeholders group that are standing in the front so they might be involved maybe once a month. So if there are any major issues that goes on a certain threshold they made the decisions. And then all the other stakeholders, which are part of the project, it just depends on the nature of the project but obviously, they way you engage them, the more frequent, the better. You do things like setup thresholds for the decision making to make sure that you have got a framework. If you need a decision to make by one of the stakeholders, you know when to get them, and why to get them, does that make sense?

13:12 Sometimes in a project you have conflicting requirements, demands from the business which are conflicting. How do you solve those conflicts in a project?
They answer is that you need good business analyst. It is all about personality, but I guess you could use frameworks, you could look at the requirements, and you could try to measure the value to the business or the value you get from the requirements. But the reality in a real project is that you need a very good business analyst. Someone who can kind of mediate between the conflicting viewpoints and come up with a solution that either compromises one or prove that one of those requirements isn't the right thing to do. So it is all about building relationships with people

14:12 How do you determine if the suggested input from the stakeholders is right?
The answer is similar to the answer before, making sure that you have got a good business analyst, because that business analyst will have business experience and the similar things before. So I guess it is the same act, either you use experience, or you are trying to record the return on investment for that requirement. So I think it is a kind of similar to the question before.

14:54 Do you think it is important that all the stakeholders understand the requirements?
That depends on the level of the stakeholders. Senior stakeholders absolutely not. All they need to know is the direction, the business case and if anything goes wrong but obviously, the lower you down in the level of the stakeholder then yes it does become important. They need to understand which stakeholder is kind of own which requirement and obviously they need to sign those off. They need to be completely aware and understand the requirements to sign off and the make the project successful.

15:39 How do you make them understand the requirements? Do you have special techniques for that?
Yes, again it is by having a good Business Analyst and good documentation. It kind of goes back to the earlier comments; it is the holy grail of the Business Analyst. Ultimately, if you create a very detailed business specification than the likelihood of a business stakeholder reading it and understanding every word in it is pretty unlikely and that is a kind of a difference between working internally and externally. Internally when you are for example rolling out an underwriting system, you are relying on the partnership between IT and the business department to get that trust and even if it is all writing down, in the end of the day, the business trusted you because you did the right thing. To actually answer the question, It is really a kind of demonstrating the system, come with a prototype or document in the requirements and you kind of give a face-to-face explanation.

16:57 How do you determine which requirements are crucial and which are not?
Again, it is based on experience. You need to estimate how crucial it is, even for Business

Analyst with 15 years experience cannot determine what a critical requirement is and what is not. The reason for that is, ultimately you are getting the requirements from the people that are using the system, which is one set but you also need from their supervisors to give you a scene of a kind on the direction. So for instance you might be have the high level business case is around setting up the products externally or the high level business might be save people time, save activity. The person that use the system might want to have X,Y and Z in the system, and the Business Analyst might take that requirement and get the extra comment from a external perspective but it need to get a level up to look at all the requirements say well actually in the ground schema we shouldn't be spending money on here. So the answer is; you need experience but you also need someone that got a bigger picture to look up the requirements as a whole and make a decision on what we shouldn't do and what we should do based on the strategic direction on the system and the costs of putting that requirement in.

19:28 Do you have special prioritizing techniques for prioritizing the requirements or is this also based on experience?
Yes, I think it is based on experience, similar of determining about what is critical or not. As a framework it would be based on the strategic direction of that system but also on the return of investment of that system. In other words let's say; you are writing a market-facing system that sell insurance to customers, now, this system might to put off several insurance products, like travel products, personal active product. How would you prioritize against which product should go in that system? It is really information on return on investments, in other words, which product is going to sell the most? You know, what do we forecast to be the biggest money maker? That is how you prioritize the requirements from a high level by looking at the amount of money that it is going to make for the company.

20:48 What kind of dependencies on the requirements have you come across?
This is again based on experience and getting a very good Business Analyst. A very good Business Analyst does exactly that, you know; to understand the dependencies, what effects what, that is the essence of a very good skilled Business Analyst. And in terms of how you kind of document it, one example might a requirements tree where you have got the main requirements, let say; produce a market-facing technology system that involves PE insurance. The next level might be; it needs to sell new business, it needs to support midterms adjustment and it need to support renewals. And then each requirements break down and down until on the right sight of the requirement tree you have an extremely low level of requirements that need to developed against. In that way you see all the dependencies.

22:24 What is the most time consuming part of requirements engineering?
The most time consuming part in requirements process is probably what we just said in 5.4. If you look at the process depended on the methodology you use, you know you are talking to the business, you are getting the requirements from a high level, but then understanding the impact of the requirements, the interdependency of it that takes the most time because that is where the Business Analyst gear on it, thinking about if you do a change or implement on that then what does that mean to X, Y and Z? And usually it is not only X,Y,Z and often is A till Z. There is always a massive kind of dependencies between requirements that need to understand and documented.

23:28 Is it possible to make decisions later on in the project?
If you look at the historical way of rolling out systems you are talking about waterfall methodology. In other words; you get the requirements, you do the development, you do the testing and release it. Now the answer of this question is; yes you can make decision too late, but the reality is, is that if you are more then an IT team you need to be adaptive to change because that is how the world works. You get the requirements, the requirements change, what do you see in the systems? What do you understand some of the kinds in the reality that what you developed in things changed? The answer is yes, but really you need an IT team that is kind of prepare for that and that is what Agile does, because you can have a set of a kind of changes that roll in through each iteration.

25:04 Have you ever had a project where deadlines were delayed? What were the main reasons for that?
Yeah, any IT professional that says no is lying. There are always deadlines that are delayed; I mean I guess it depends on how you manage it, because there are different parts of the project that you can tweak in order to change the project. You can reduce the scope, you can reduce the quality or you can increase the resources. The main reasons are a kind of all the reason that we talked about. The requirements phase is the first critical phase of the project, to get that right is very important, usually why most projects are delayed are usually but not always due to the technical side. You know something might be more technical complicated that we originally thought, we might have infrastructure problems or there might be a fundamental issue with the requirements, the business changed their mind and you might need to go back to make sure that you understand them. That are the major reasons for delaying the project.

26:47 Have you ever used a collaboration system for requirements engineering?
Collaboration system what is that?
A collaboration system is more or less like; I know that for the ARCH project they are using SharePoint, but you can also you a WIKI system, something like that.
Yes, that is extremely important in a distributed IT team when you got developers in one country, analysts in the other and the business in a third, then yes it is absolutely crucial. The answer is yes, we could use and use WIKI, SharePoint amd Hadle, is another one that we are using. We use lotus notes to share documents. Yes, it is absolutely critical to any IT project in a distributed environment.

27:45 And that kind of systems are also used for discussions and stuff? To elaborate on the requirements?
You need to make sure that the Business Analyst is close to the business, to the people that give them the requirements. When you are working in a distributed team then that document is sent to the developers and it can be sent to a kind of a collaboration system. It kind of usually is, as long as the requirements are accessible to everyone that is kind of the most important thing.

28:35 Sometimes there are wicked problems in a project. What I define as a wicked problem is a problem that is changing all the time. For example you gathered the requirements and while you are building it, the requirements are already changed. Do have that often in a project? How are you addressing that kind of a problem?
If you are recording the requirements in a right way and you have got the right relationship with the person that is given you the requirements, then there shouldn't be the case that you have a changing requirements because if the requirement changes, then you probably didn't understand it in the first place. A lot of this comes down to a good Business Analyst skill, experience, but also making sure that the person who has given you the requirements has the accountability to sign it off. In other words if they changes their mind all the time, then you get to a state in the project where you say; right this are the requirements, this is writing down in black in white, sign that off. From that you continue.

30:11 What was the worst project that you ever had to do in terms of gathering requirements? How did you addressed that problem?
There are various one. One of the projects that didn't work here in the past is where we had a very very distributed IT team. In other words you have the business team in Australia that were giving the requirements, you had the Business Analyst in Singapore, and the developers in China, and that is a very different communication style and a very different culture. Ultimately, the requirements that were recorded for the system that was build, wasn't the right one for the business and I think what happened, in terms of the failure of the project; What you do is to make sure that the person who is given the requirements, the business, is seeing very early what you are developing and prototyping because they you bring the work , you can't just read the document and picture a working system. You need to able to see it because once you see it, then you can say; ooh actually I didn't expect that can we have that or can it be moved around or can

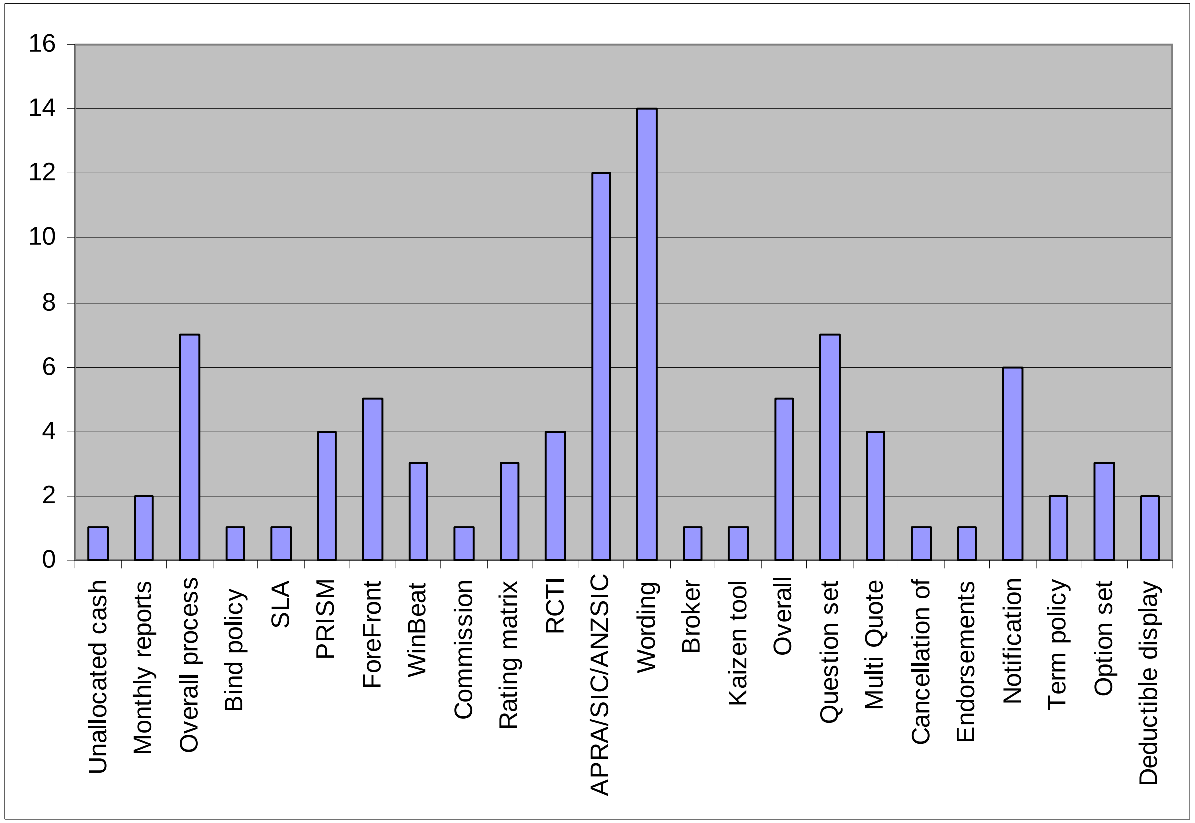
we have that instead and that. That is Agile; you know not so a word, it is usually the right way of doing things. People need to see something on the verify it.

Appendix G: Direct observations: Iterations per requirement concern

Iterations per requirement concerns:

No.	Requirement concern	Iterations
1	Unallocated cash	1
2	Monthly reports	2
3	Overall process	7
4	Bind policy	1
5	SLA	1
6	PRISM	4
7	ForeFront	5
8	WinBeat	3
9	Commission	1
10	Rating matrix	3
11	RCTI	4
12	APRA/SIC/ANZSIC	12
13	Wording	14
14	Broker engagement	1
15	Kaizen tool	1
16	Overall development (validation)	5
17	Question set	7
18	Multi Quote	4
19	Cancellation of policy/quote	1
20	Endorsements	1
21	Notification handling	6
22	Term policy	2
23	Option set	3
24	Deductible display	2

Average amount of iteration: 3,21



Addressed concerns per meeting

Meeting No	Meeting	Requirement concerns
1	25-2-2011	Unallocated cash Monthly reports Overall process Policies binding SLA
2	16-2-2011	Overall process PRISM ForeFront WinBeat Monthly reports Commission Rating matrix RCTI
3	18-2-2011	RCTI APRA/SIC WinBeat Clausing/Wording Overall process
4	21-2-2011	RCTI Clausing/Wording Overall process
5	22-2-2011	WinBeat Clausing/Wording PRISM Broker engagement
6	7-3-2011	Clausing/Wording PRISM RCTI
7	17-3-2011	PRISM
8	22-3-2011	Overall process Wording ForeFront Rating Kaizen tool
9	28-3-2011	Overall process Wording ForeFront
10	5-4-2011	Overall process Wording ForeFront

Meeting No	Meeting	Requirement concerns
13	12-5-2011	Questions
14	16-5-2011	SIC / APRA
15	17-5-2011	SIC / APRA Wording Multi quote Question set
16	24-5-2011	SIC / APRA Wording
17	2-6-2011	SIC / APRA Multi quote
18	8-6-2011	Questions Wordings Multi Quote SIC / APRA Cancellation of policy Endorsements Notification handling Declined / referral handling
19	16-6-2011	Questions set Wordings SIC / APRA Declined / referral handling Rating 10m option
20	20-6-2011	Questions set Wordings (exclusions & endorsements) Options
21	22-6-2011	Questions set Wordings (exclusions & endorsements) Term policy SIC/ANSIC (occupation categories) Referral message Deductible Option set
22	27-6-2011	Term policy SIC/ANSIC (occupation categories) Referral message

11	29-4-2011	APRA/SIC Overall development / validation
12	5-5-2011	APRA/SIC Overall development / validation Wording Questions ForeFront Multi Quote

		Deductible Option set
23	28-6-2011	SIC/ANSIC (occupation categories)

Appendix H: Meetings notes of 24-05-2011 and 22-06-2011

Notes of the meeting on 24-05-2011

Purpose: Meeting on wording

Chair: Business Analyst (BA)
Developer (DEV)
Project Manager (PM)
Expert

Date: 24-05-2011

Minutes

APRA codes are not used for ForeFront. BA has to contact CFO & IT Manager to figure out if we are only need SIC or NZSIC codes. Although, PRISM requires an APRA code, ask IT Manager or CFO if we can use a dummy code.

After the meeting, DEV will give a demonstration to BA and Expert of the current iClose system in the RC environment.

BA2 (from other project) should be contact to figure out how he did the mapping of the wordings in the Nexus tool. However, PM suggested that this should not be used for the iClose project, since the nexus is used internally and included all the endorsements. iClose will have to use a “lite” version of this.

BA will have to set up a meeting with PM and Expert on Thursday to discuss the status of the wording (11:30).

Expert will provide templates (documents) of bindings, schedules and policy wordings

The purpose is to figure out the triggers of the endorsements. BA will document these triggers and will validate and provide these to DEV.

DEV will implement the business questions this week, probably in the next two days.

Notes of the meeting on 22-06-2011

Purpose: Meeting on business requirements outstanding issues

Chair: Project Manager (PM)
Business Analyst (BA)
Developer (DEV)

Date: 22-06-2011 (13:00)

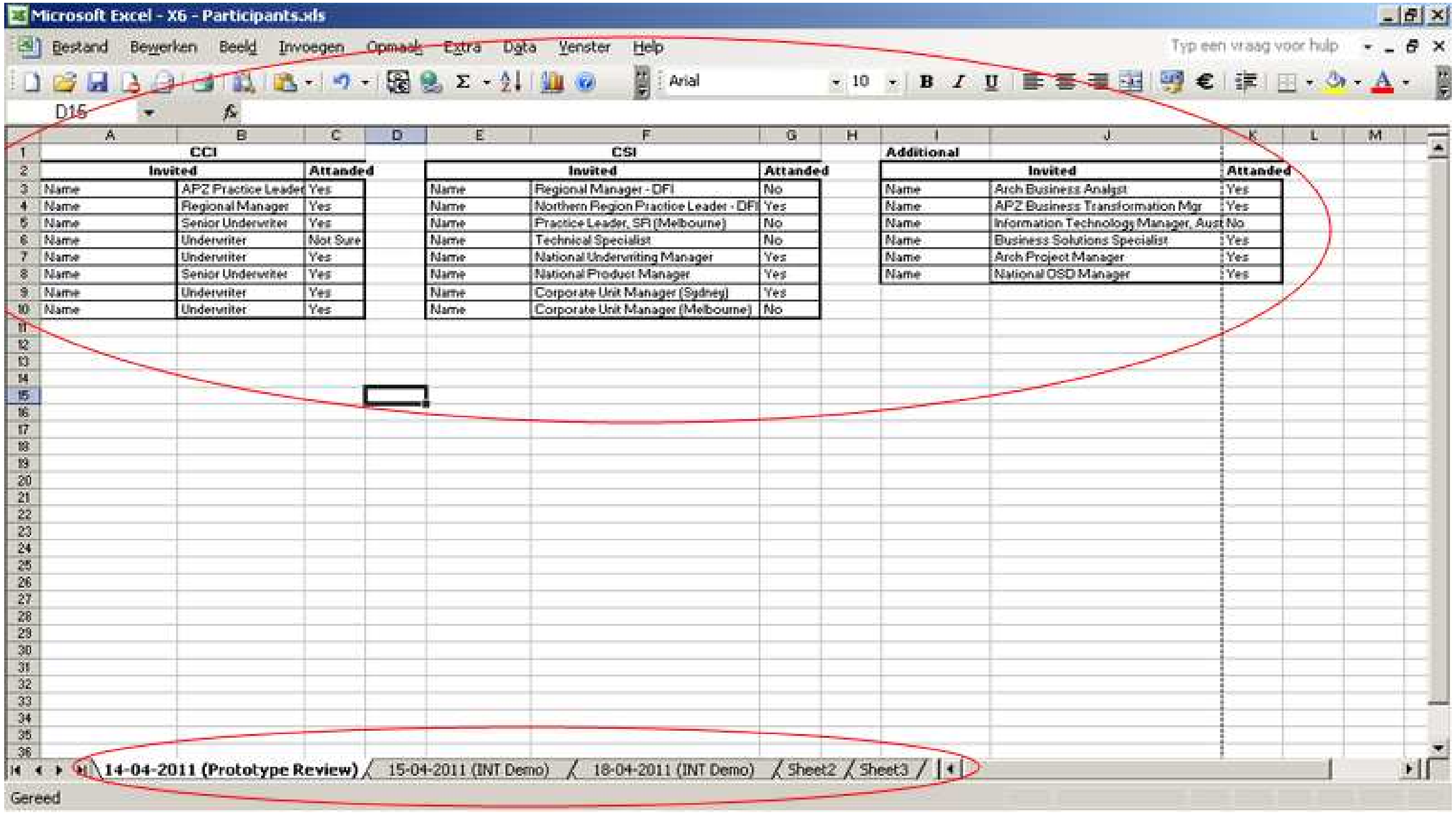
Minutes

- PM and BA will find out what is needed in terms of occupations
  - Not going to use ANZSIC
  - Use occupation category and SIC.
- BA will ask CFO if we do use a minimum policy term, and if this term is minimal 12 months.
- The referral message for the pilot will be something like; referral is required for this…. This will be discussed tomorrow, need to confirm with the business. This will be a compromised solution, because if there are more referral triggers.
- Need to confirm if we do want to show the deductible on the screen or only in the wording

- DEV will find out what effort is needed to build-in the 10M option

Appendix I: Attendance of participants during prototype meetings

Attendance of meeting 14-04-2011



CCI

Invited		Attanded
Name	APZ Practice Leader	Yes
Name	Regional Manager	Yes
Name	Senior Underwriter	Yes
Name	Underwriter	Not Sure
Name	Underwriter	Yes
Name	Senior Underwriter	Yes
Name	Underwriter	Yes
Name	Underwriter	Yes

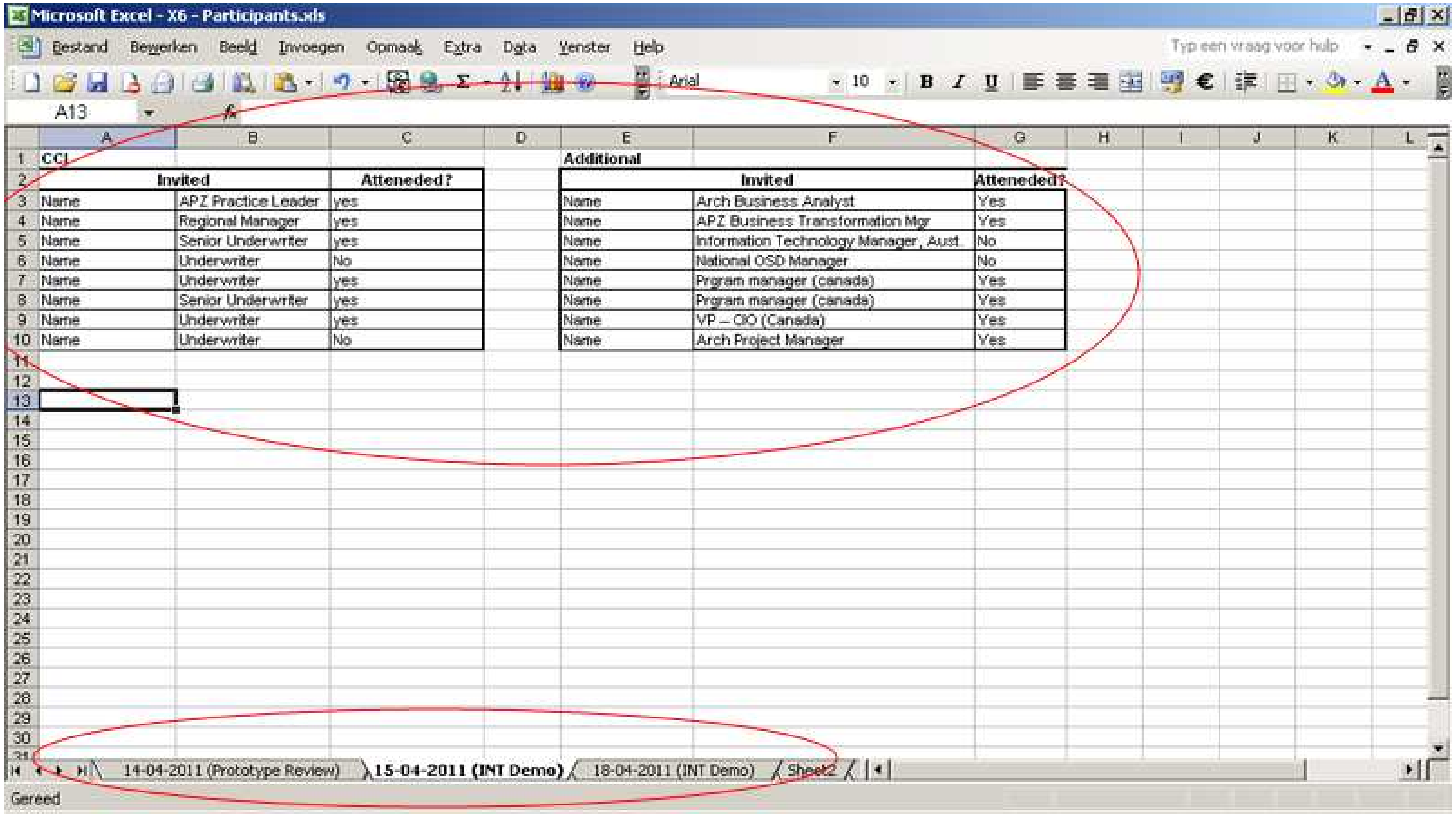
CSI

Invited		Attanded
Name	Regional Manager - DFI	No
Name	Northern Region Practice Leader - DFI	Yes
Name	Practice Leader, SR (Melbourne)	No
Name	Technical Specialist	No
Name	National Underwriting Manager	Yes
Name	National Product Manager	Yes
Name	Corporate Unit Manager (Sydney)	Yes
Name	Corporate Unit Manager (Melbourne)	No

Additional

Invited		Attanded
Name	Arch Business Analyst	Yes
Name	APZ Business Transformation Mgr	Yes
Name	Information Technology Manager, Aust	No
Name	Business Solutions Specialist	Yes
Name	Arch Project Manager	Yes
Name	National OSD Manager	Yes

Attendance of meeting 15-04-2011



CCI

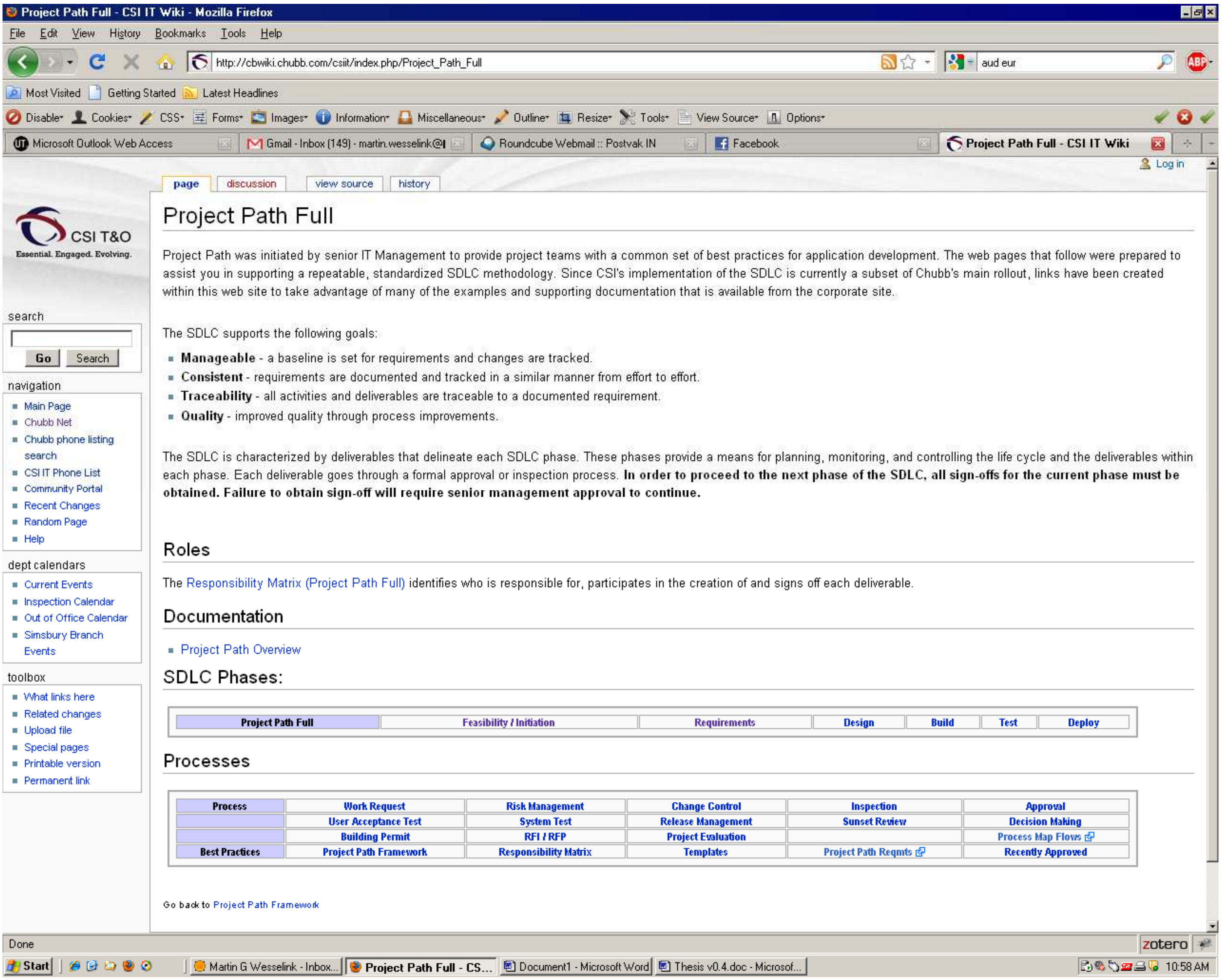
Invited		Atteneded?
Name	APZ Practice Leader	yes
Name	Regional Manager	yes
Name	Senior Underwriter	yes
Name	Underwriter	No
Name	Underwriter	yes
Name	Senior Underwriter	yes
Name	Underwriter	yes
Name	Underwriter	No

Additional

Invited		Atteneded?
Name	Arch Business Analyst	Yes
Name	APZ Business Transformation Mgr	Yes
Name	Information Technology Manager, Aust.	No
Name	National OSD Manager	No
Name	Prgram manager (canada)	Yes
Name	Prgram manager (canada)	Yes
Name	VP – CIO (Canada)	Yes
Name	Arch Project Manager	Yes

Appendix J: Software development life cycle documentation

SDLC Phase: Software development project path


Project Path Full - CSI IT Wiki - Mozilla Firefox
File Edit View History Bookmarks Tools Help
http://cbwiki.chubb.com/csiit/index.php/Project_Path_Full
aud eur
Most Visited Getting Started Latest Headlines
Disable Cookies CSS Forms Images Information Miscellaneous Outline Resize Tools View Source Options
Microsoft Outlook Web Access
Gmail - Inbox (149) - martin.wesselink@
Roundcube Webmail :: Postvak IN
Facebook
Project Path Full - CSI IT Wiki
Log in
page
discussion
view source
history
CSI T&O
Essential. Engaged. Evolving.
search
Go
Search
navigation
Main Page
Chubb Net
Chubb phone listing search
CSI IT Phone List
Community Portal
Recent Changes
Random Page
Help
dept calendars
Current Events
Inspection Calendar
Out of Office Calendar
Simsbury Branch Events
toolbox
What links here
Related changes
Upload file
Special pages
Printable version
Permanent link
Done
zotero
Start
Martin G Wesselink - Inbox...
Project Path Full - CS...
Document1 - Microsoft Word
Thesis v0.4.doc - Microsof...
10:58 AM

Project Path Full

Project Path was initiated by senior IT Management to provide project teams with a common set of best practices for application development. The web pages that follow were prepared to assist you in supporting a repeatable, standardized SDLC methodology. Since CSI's implementation of the SDLC is currently a subset of Chubb's main rollout, links have been created within this web site to take advantage of many of the examples and supporting documentation that is available from the corporate site.

The SDLC supports the following goals:

- **Manageable** - a baseline is set for requirements and changes are tracked.
- **Consistent** - requirements are documented and tracked in a similar manner from effort to effort.
- **Traceability** - all activities and deliverables are traceable to a documented requirement.
- **Quality** - improved quality through process improvements.

The SDLC is characterized by deliverables that delineate each SDLC phase. These phases provide a means for planning, monitoring, and controlling the life cycle and the deliverables within each phase. Each deliverable goes through a formal approval or inspection process. **In order to proceed to the next phase of the SDLC, all sign-offs for the current phase must be obtained. Failure to obtain sign-off will require senior management approval to continue.**

Roles

The Responsibility Matrix (Project Path Full) identifies who is responsible for, participates in the creation of and signs off each deliverable.

Documentation

- Project Path Overview

SDLC Phases:

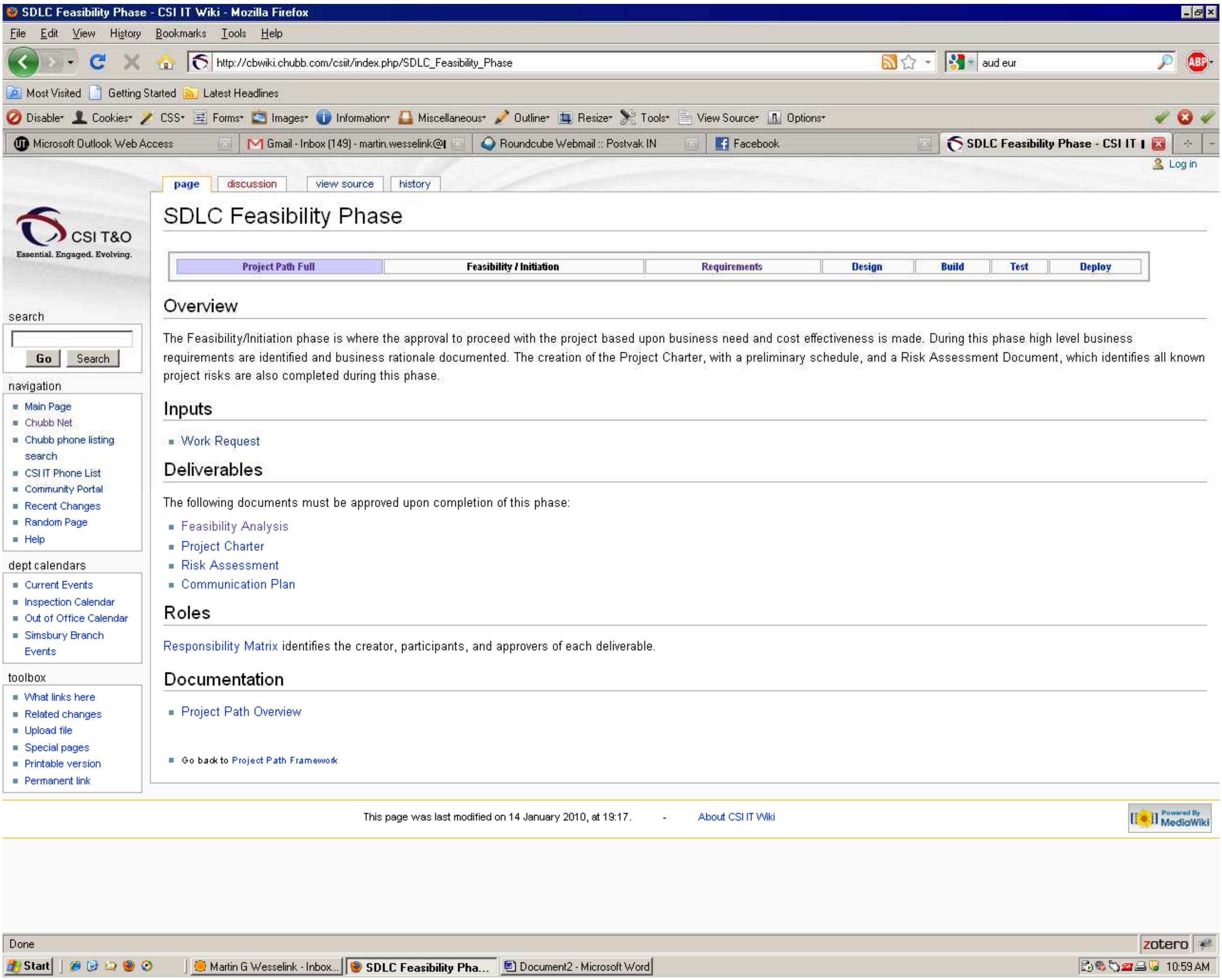
Project Path Full	Feasibility / Initiation	Requirements	Design	Build	Test	Deploy

Processes

Process	Work Request	Risk Management	Change Control	Inspection	Approval
	User Acceptance Test	System Test	Release Management	Sunset Review	Decision Making
	Building Permit	RFI / RFP	Project Evaluation		Process Map Flows
Best Practices	Project Path Framework	Responsibility Matrix	Templates	Project Path Reqmts	Recently Approved

Go back to Project Path Framework

SDLC Phase: Feasibility


SDLC Feasibility Phase - CSI IT Wiki - Mozilla Firefox
File Edit View History Bookmarks Tools Help
http://cbwiki.chubb.com/csiit/index.php/SDLC_Feasibility_Phase
aud eur
Most Visited Getting Started Latest Headlines
Disable Cookies CSS Forms Images Information Miscellaneous Outline Resize Tools View Source Options
Microsoft Outlook Web Access
Gmail - Inbox (149) - martin.wesselink@
Roundcube Webmail :: Postvak IN
Facebook
SDLC Feasibility Phase - CSI IT
Log in
page
discussion
view source
history
CSI T&O
Essential. Engaged. Evolving.
search
Go
Search
navigation
Main Page
Chubb Net
Chubb phone listing search
CSI IT Phone List
Community Portal
Recent Changes
Random Page
Help
dept calendars
Current Events
Inspection Calendar
Out of Office Calendar
Simsbury Branch Events
toolbox
What links here
Related changes
Upload file
Special pages
Printable version
Permanent link
Powered By MediaWiki
Done
zotero
Start
Martin G Wesselink - Inbox...
SDLC Feasibility Pha...
Document2 - Microsoft Word
10:59 AM

SDLC Feasibility Phase

Project Path Full	Feasibility / Initiation	Requirements	Design	Build	Test	Deploy

Overview

The Feasibility/Initiation phase is where the approval to proceed with the project based upon business need and cost effectiveness is made. During this phase high level business requirements are identified and business rationale documented. The creation of the Project Charter, with a preliminary schedule, and a Risk Assessment Document, which identifies all known project risks are also completed during this phase.

Inputs

- Work Request

Deliverables

The following documents must be approved upon completion of this phase:

- Feasibility Analysis
- Project Charter
- Risk Assessment
- Communication Plan

Roles

Responsibility Matrix identifies the creator, participants, and approvers of each deliverable.

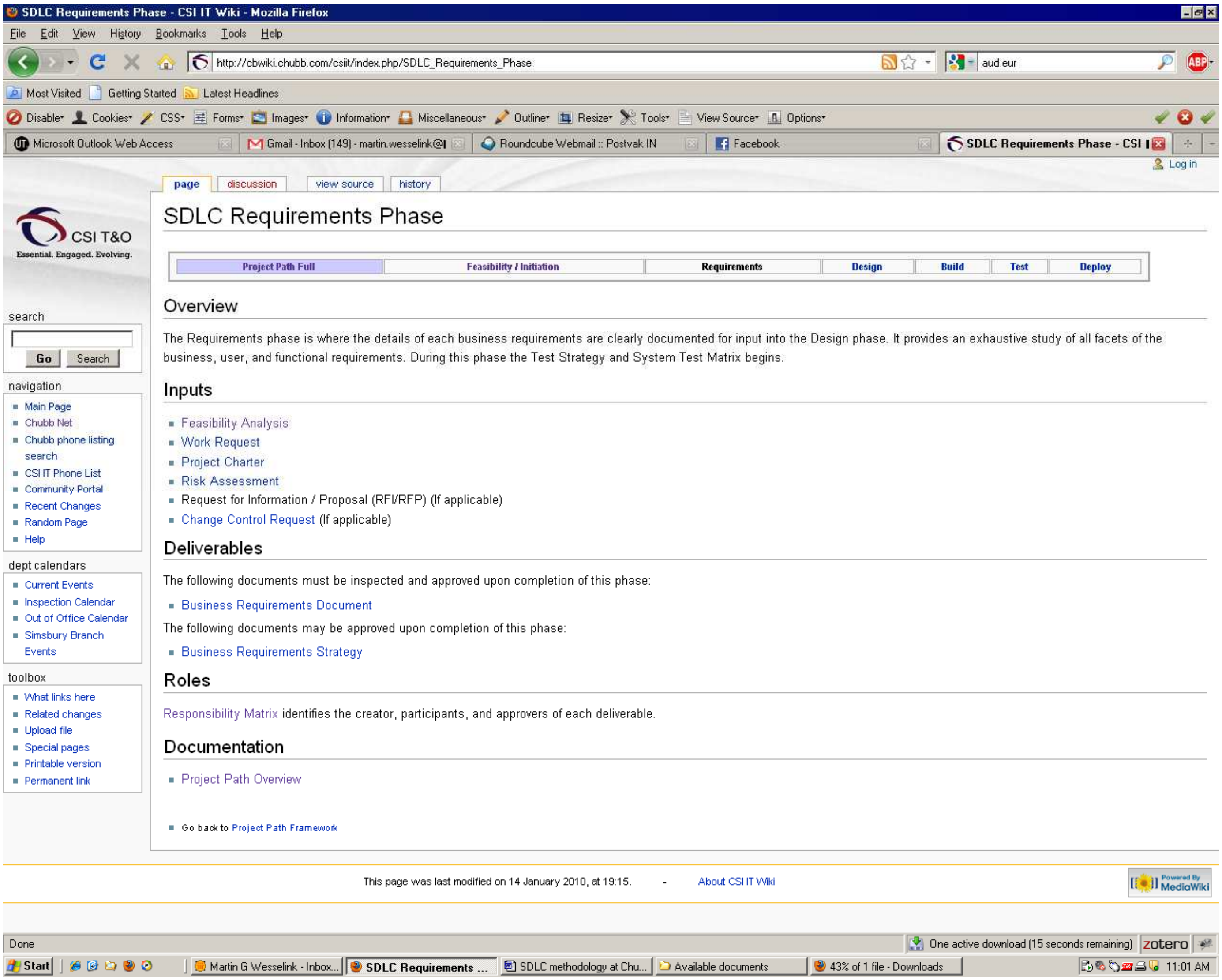
Documentation

- Project Path Overview

- Go back to Project Path Framework

This page was last modified on 14 January 2010, at 19:17. - About CSI IT Wiki

SDLC Phase: Requirements engineering



SDLC Requirements Phase - CSI IT Wiki - Mozilla Firefox
File Edit View History Bookmarks Tools Help
http://cbwiki.chubb.com/csiit/index.php/SDLC_Requirements_Phase
aud eur
Most Visited Getting Started Latest Headlines
Disable Cookies CSS Forms Images Information Miscellaneous Outline Resize Tools View Source Options
Microsoft Outlook Web Access
Gmail - Inbox (149) - martin.wesselink@
Roundcube Webmail :: Postvak IN
Facebook
SDLC Requirements Phase - CSI I
Log in
page discussion view source history
CSI T&O
Essential. Engaged. Evolving.
search
Go Search
navigation
Main Page
Chubb Net
Chubb phone listing search
CSI IT Phone List
Community Portal
Recent Changes
Random Page
Help
dept calendars
Current Events
Inspection Calendar
Out of Office Calendar
Simsbury Branch Events
toolbox
What links here
Related changes
Upload file
Special pages
Printable version
Permanent link
Project Path Full
Feasibility / Initiation
Requirements
Design
Build
Test
Deploy
Powered By MediaWiki
Done
One active download (15 seconds remaining)
zotero
Start
Martin G Wesselink - Inbox...
SDLC Requirements ...
SDLC methodology at Chu...
Available documents
43% of 1 file - Downloads
11:01 AM

SDLC Requirements Phase

Overview

The Requirements phase is where the details of each business requirements are clearly documented for input into the Design phase. It provides an exhaustive study of all facets of the business, user, and functional requirements. During this phase the Test Strategy and System Test Matrix begins.

Inputs

- Feasibility Analysis
- Work Request
- Project Charter
- Risk Assessment
- Request for Information / Proposal (RFI/RFP) (If applicable)
- Change Control Request (If applicable)

Deliverables

The following documents must be inspected and approved upon completion of this phase:

- Business Requirements Document

The following documents may be approved upon completion of this phase:

- Business Requirements Strategy

Roles

Responsibility Matrix identifies the creator, participants, and approvers of each deliverable.

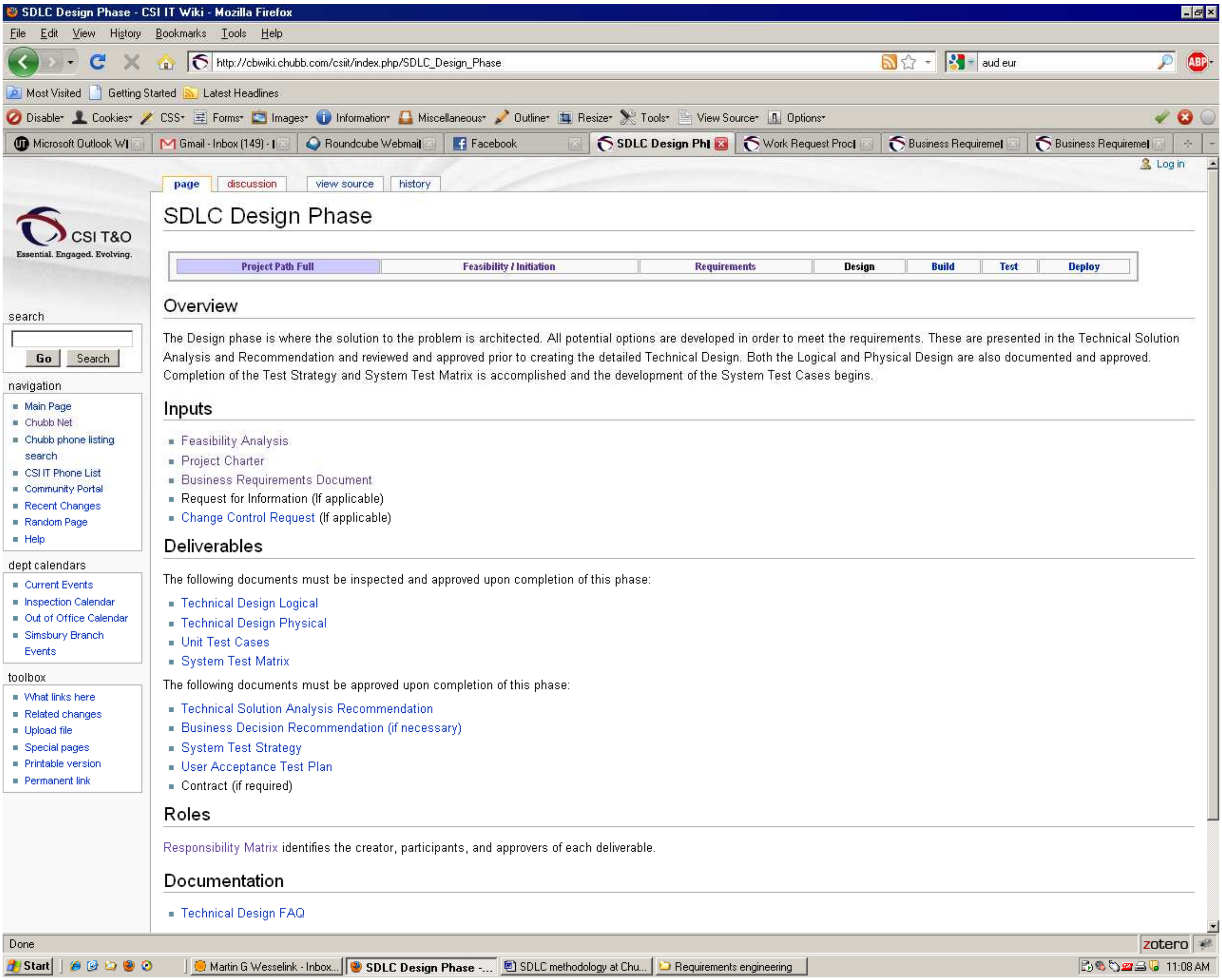
Documentation

- Project Path Overview

- Go back to Project Path Framework

This page was last modified on 14 January 2010, at 19:15. - About CSI IT Wiki

SDLC Phase: Design



SDLC Design Phase

Project Path Full	Feasibility / Initiation	Requirements	Design	Build	Test	Deploy

Overview

The Design phase is where the solution to the problem is architected. All potential options are developed in order to meet the requirements. These are presented in the Technical Solution Analysis and Recommendation and reviewed and approved prior to creating the detailed Technical Design. Both the Logical and Physical Design are also documented and approved. Completion of the Test Strategy and System Test Matrix is accomplished and the development of the System Test Cases begins.

Inputs

- Feasibility Analysis
- Project Charter
- Business Requirements Document
- Request for Information (If applicable)
- Change Control Request (If applicable)

Deliverables

The following documents must be inspected and approved upon completion of this phase:

- Technical Design Logical
- Technical Design Physical
- Unit Test Cases
- System Test Matrix

The following documents must be approved upon completion of this phase:

- Technical Solution Analysis Recommendation
- Business Decision Recommendation (if necessary)
- System Test Strategy
- User Acceptance Test Plan
- Contract (if required)

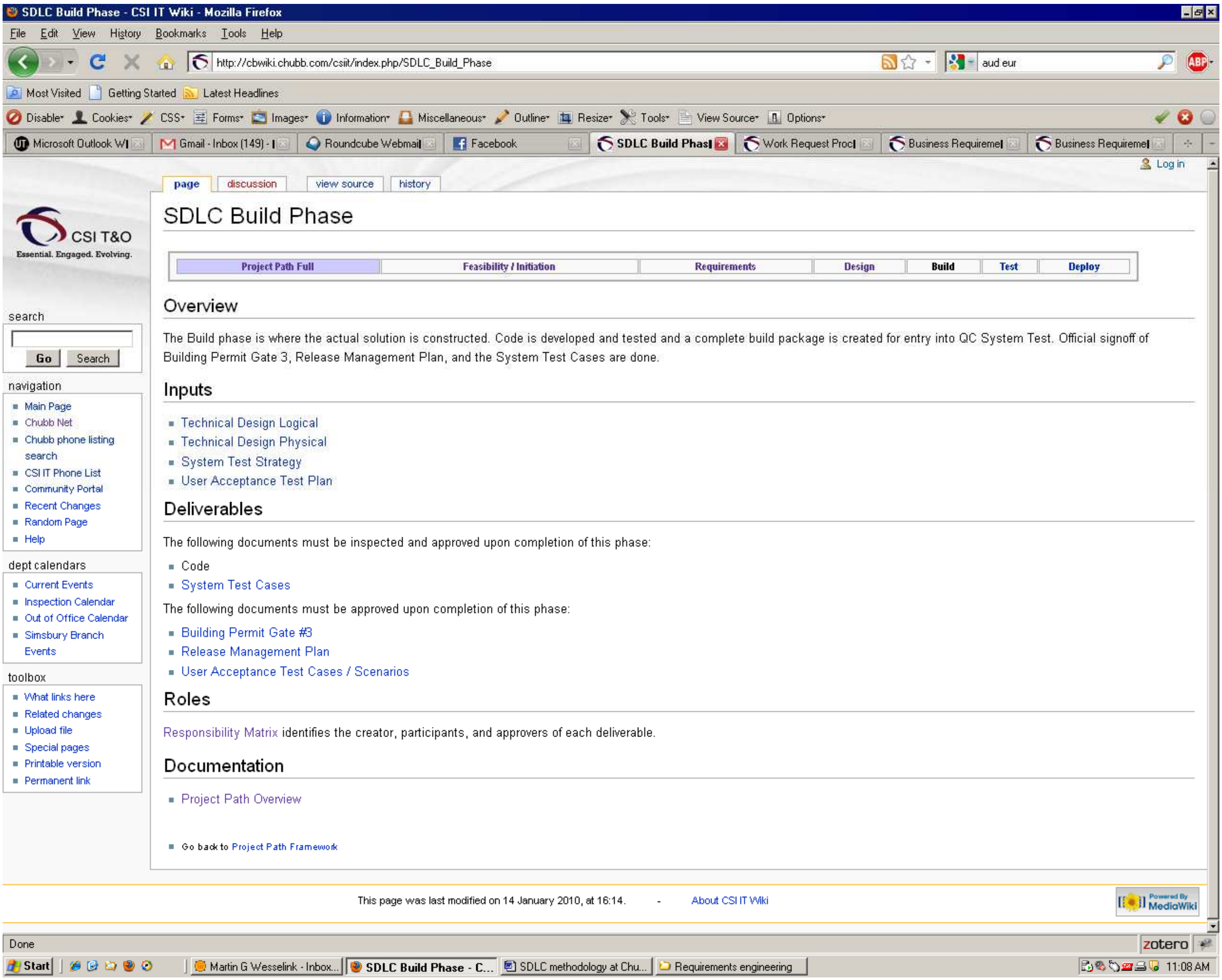
Roles

Responsibility Matrix identifies the creator, participants, and approvers of each deliverable.

Documentation

- Technical Design FAQ

SDLC Phase: Building



SDLC Build Phase

Project Path Full	Feasibility / Initiation	Requirements	Design	Build	Test	Deploy

Overview

The Build phase is where the actual solution is constructed. Code is developed and tested and a complete build package is created for entry into QC System Test. Official signoff of Building Permit Gate 3, Release Management Plan, and the System Test Cases are done.

Inputs

- Technical Design Logical
- Technical Design Physical
- System Test Strategy
- User Acceptance Test Plan

Deliverables

The following documents must be inspected and approved upon completion of this phase:

- Code
- System Test Cases

The following documents must be approved upon completion of this phase:

- Building Permit Gate #3
- Release Management Plan
- User Acceptance Test Cases / Scenarios

Roles

Responsibility Matrix identifies the creator, participants, and approvers of each deliverable.

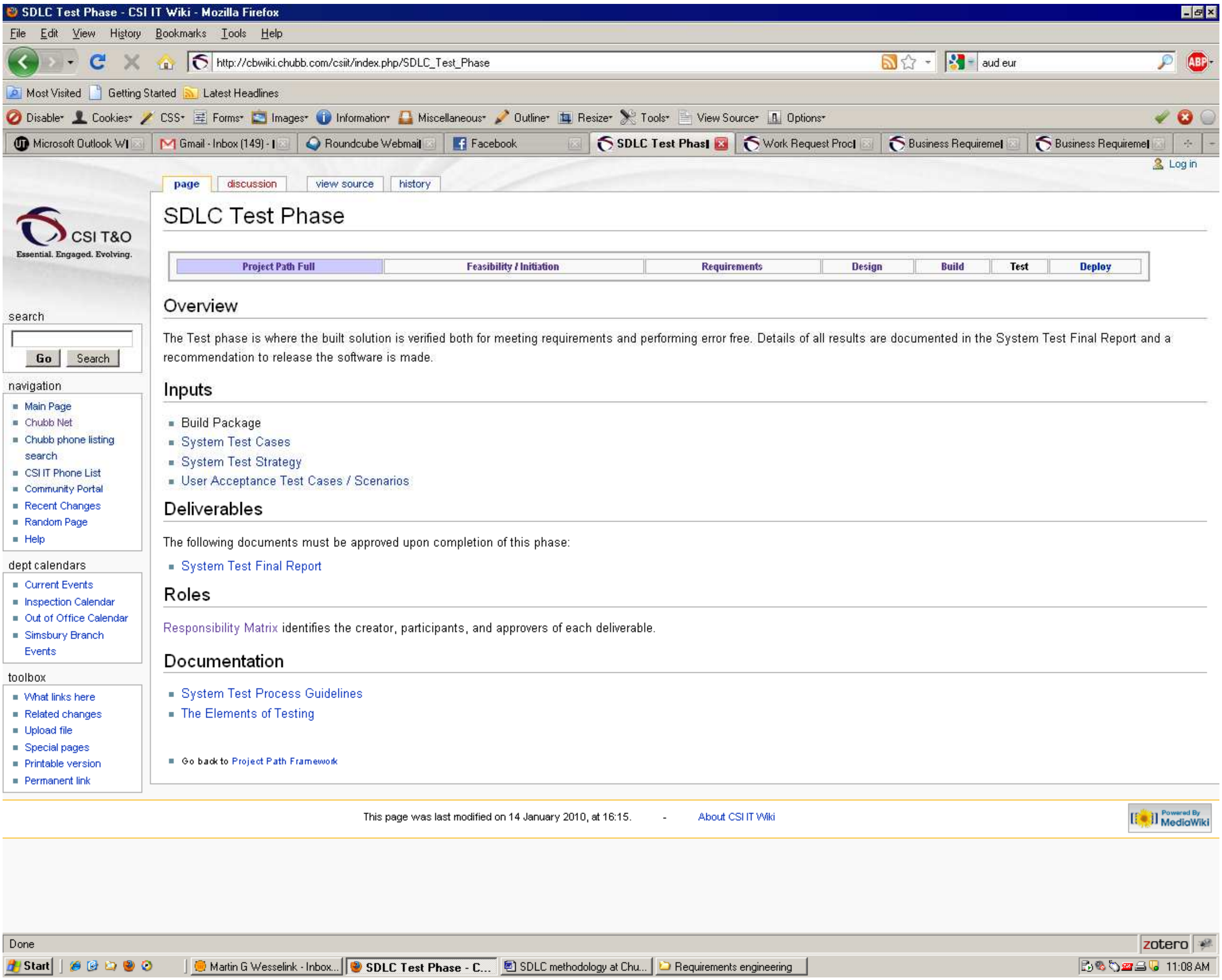
Documentation

- Project Path Overview

- Go back to Project Path Framework

This page was last modified on 14 January 2010, at 16:14. - About CSI IT Wiki

SDLC Phase: Testing



SDLC Test Phase

Project Path Full	Feasibility / Initiation	Requirements	Design	Build	Test	Deploy

Overview

The Test phase is where the built solution is verified both for meeting requirements and performing error free. Details of all results are documented in the System Test Final Report and a recommendation to release the software is made.

Inputs

- Build Package
- System Test Cases
- System Test Strategy
- User Acceptance Test Cases / Scenarios

Deliverables

The following documents must be approved upon completion of this phase:

- System Test Final Report

Roles

Responsibility Matrix identifies the creator, participants, and approvers of each deliverable.

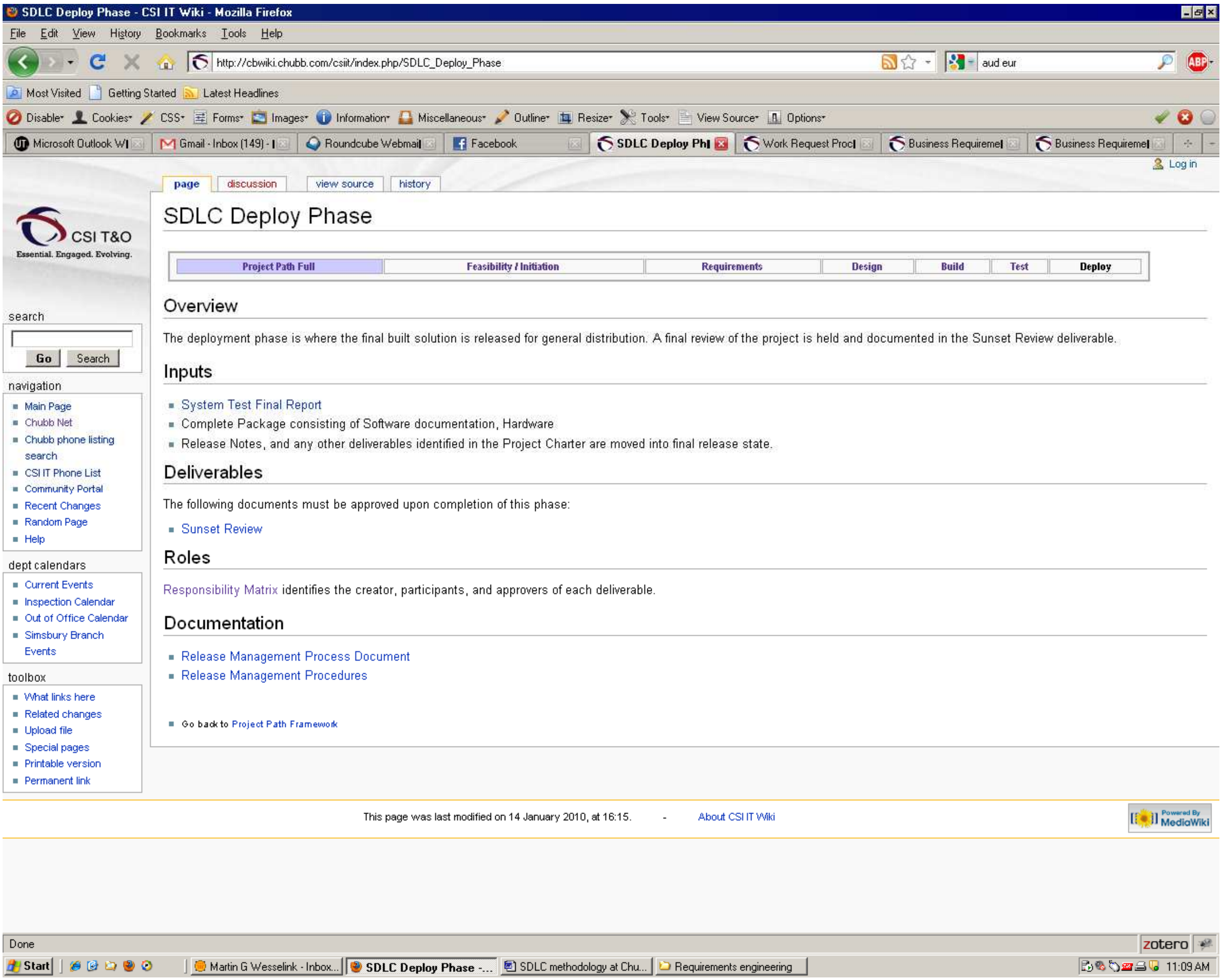
Documentation

- System Test Process Guidelines
- The Elements of Testing

- Go back to Project Path Framework

This page was last modified on 14 January 2010, at 16:15. - About CSI IT Wiki

SDLC Phase: Deploying



SDLC Deploy Phase

Project Path Full	Feasibility / Initiation	Requirements	Design	Build	Test	Deploy

Overview

The deployment phase is where the final built solution is released for general distribution. A final review of the project is held and documented in the Sunset Review deliverable.

Inputs

- System Test Final Report
- Complete Package consisting of Software documentation, Hardware
- Release Notes, and any other deliverables identified in the Project Charter are moved into final release state.

Deliverables

The following documents must be approved upon completion of this phase:

- Sunset Review

Roles

Responsibility Matrix identifies the creator, participants, and approvers of each deliverable.

Documentation

- Release Management Process Document
- Release Management Procedures

- Go back to Project Path Framework

This page was last modified on 14 January 2010, at 16:15. - About CSI IT Wiki

Appendix K: Direct observations – Overall field evaluation

Report date:	28-06-2011	
Observation period:	From: 06- 02 -2011	To: 01- 07 - 2011
Project	MFT project - iClose	

1. Total number of stakeholders: 11

2. Total number of requirement concerns: 24

3. Total number of iterations: 6
   a. Meeting 16-02-2011
   b. Meeting 22-02-2011
   c. Meeting 29-04-2011
   d. Meeting 05-05-2011
   e. Meeting 17-05-2011
   f. Meeting 16-06-2011

4. Total number of meetings: 24

5. Total number of iterations per requirement concern:

No.	Requirement concern	Iterations
1	Unallocated cash	1
2	Monthly reports	2
3	Overall process	7
4	Bind policy	1
5	SLA	1
6	PRISM	4
7	ForeFront	5
8	WinBeat	3
9	Commission	1
10	Rating matrix	3
11	RCTI	4
12	APRA/SIC/ANZSIC	13
13	Wording	15
14	Broker engagement	1
15	Kaizen tool	1
16	Overall development (validation)	5
17	Question set	8
18	Multi Quote	4
19	Cancellation of policy/quote	1
20	Endorsements	1
21	Notification handling	7
22	Term policy	2
23	Option set	3
24	Deductible display	2

Stakeholders participating in meetings

Stakeholder	**Stakeholder function**	**Meetings**
1	Business Analyst	24
2	Project Manager	22
3	CFO	11
4	IT Manager	3
5	Developer 1	9
6	Developer 2	3
7	Expert (National OSD Manager)	2
8	Expert (Independent Broker Manager)	6
9	Expert (Business Analyst of other project)	1
10	Expert (Underwriter)	2
11	Expert (National Product Manager)	5

Prioritization in meetings

Meeting no.	Type
2	Benefit/Cost
5	Benefit/Cost
11	Benefit/Cost
14	Benefit/Cost
15	Benefit/Cost
18	Benefit/Cost
19	Benefit/Cost
22	Benefit/Cost / Scope
7	Scope
12	Scope
24	Scope

Appendix L: Outstanding issues from meeting 16-02-2011 and 22-02-2011

Key outstanding issues of meeting 16-02-2011

#	Pri	High Level Requirement	Chubb Comment	Ebix Comment	Complexity	Effort
1		**RCTI** to be produced at bind and included in documents emailed to CICA and Broker.	Expert noted this was a capability in very early discussions.			
2		**APRA Occupation Codes:** iClose: Need to collect the APRA occupation code from list provided by APRA and include in information provided to CICA on Bind. Prism: Required field when entering into Prism either manually or using XML load	Would be good if we can cover this requirement with the SIC codes. However, SIC codes and APRA occupation codes would need to be mapped. May be easier to leave separate and enhance in a later phase.			
3		**SIC Codes:** We need to capture the SIC code for the insured for adding in clauses/exclusions	Issues to confirm are: 1) Are we to use SIC code or ANZIC code. Do we need to be consistent with what other insurers are using 2) What to do where the insured business covers multiple SIC codes. 3) How to capture in a user friendly way. There are ~1000 codes. Too much for a drop down list. 4) How do other online systems collect the data.			
4		**Clause numbering:** Need to confirm the CICA requirement here.	The requirement for numbering will drive complexity of iClose build.			

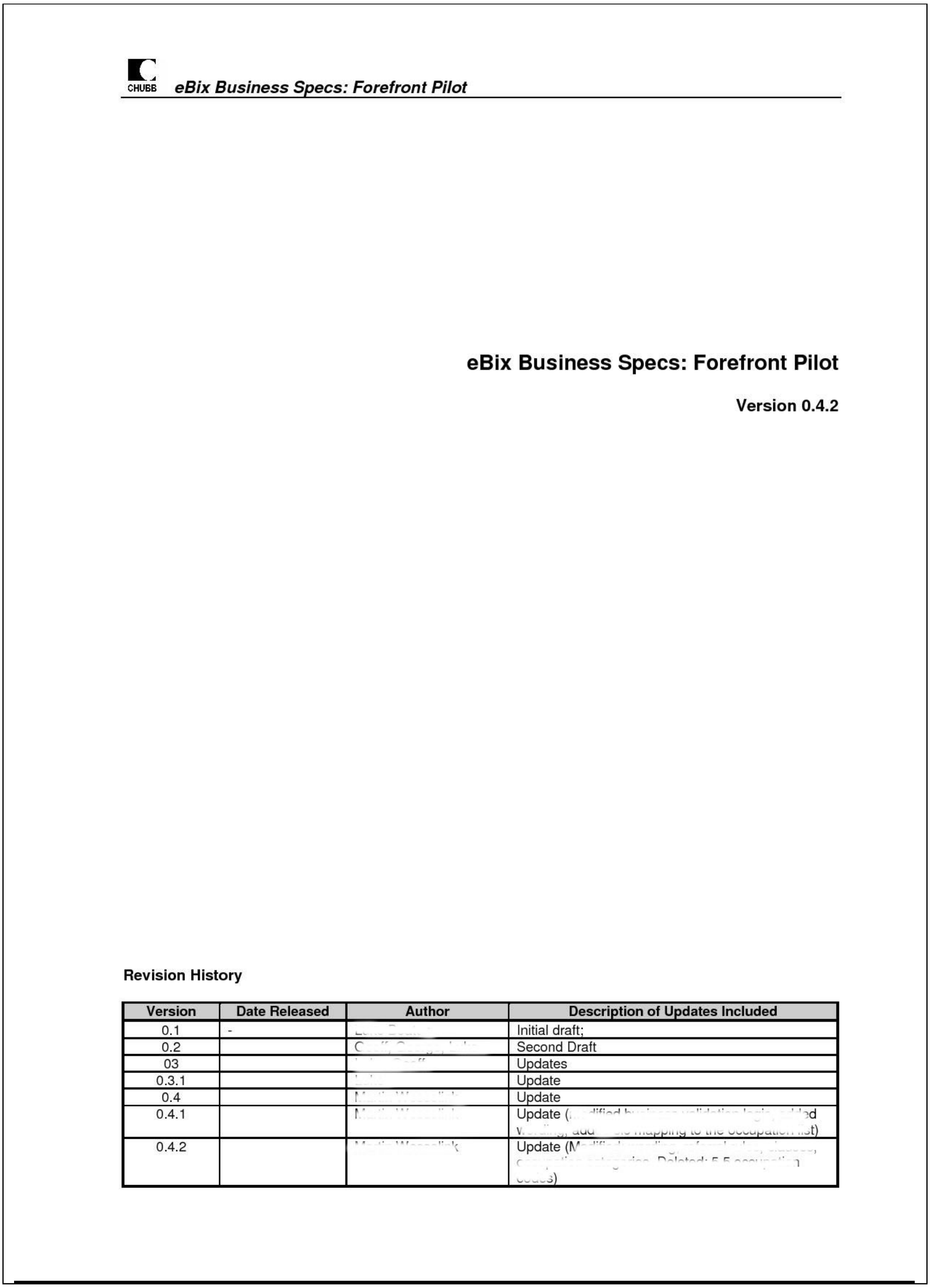
5		**Integration with Ebix Broker systems**	Need to confirm which Ebix system to integrate with in the pilot 1) Winbeat, eGlobal or CBS 2) Schedule implications 3) Reporting (Winbeat has no structured data) 4) Next Transaction			

Key outstanding issues of meeting 22-02-2011

#	Pri	High Level Requirement	Chubb Comment	Ebix Comment	Complexity	Effort
1		**RCTI** to be produced at bind and included in documents emailed to CICA and Broker.	Expert noted this was a capability in very early discussions.			
2		**APRA Occupation Codes:** iClose: Need to collect the APRA occupation code from list provided by APRA and include in information provided to CICA on Bind. Prism: Required field when entering into Prism either manually or using XML load	Would be good if we can cover this requirement with the SIC codes. However, SIC codes and APRA occupation codes would need to be mapped. May be easier to leave separate and enhance in a later phase.			
3		**SIC Codes:** We need to capture the SIC code for the insured for adding in clauses/exclusions	Issues to confirm are: 5) Are we to use SIC code or ANZIC code. Do we need to be consistent with what other insurers are using 6) What to do where the insured business covers multiple SIC codes. 7) How to capture in a user friendly way. There are ~1000 codes. Too much for a drop down list. 8) How do other online systems collect the data.			

4		**Clause numbering:** Need to confirm the CICA requirement here.	The requirement for numbering will drive complexity of iClose build.			
5		**Integration with Ebix Broker systems**	Need to confirm which Ebix system to integrate with in the pilot 5) Winbeat, eGlobal or CBS 6) Schedule implications 7) Reporting (Winbeat has no structured data) 8) Next Transaction			
6.		**Document generation and email to Broker and CICA**	Need all doco generated to go to broker and CICA email address (probably the NUC address)			

Appendix M: Business specification document



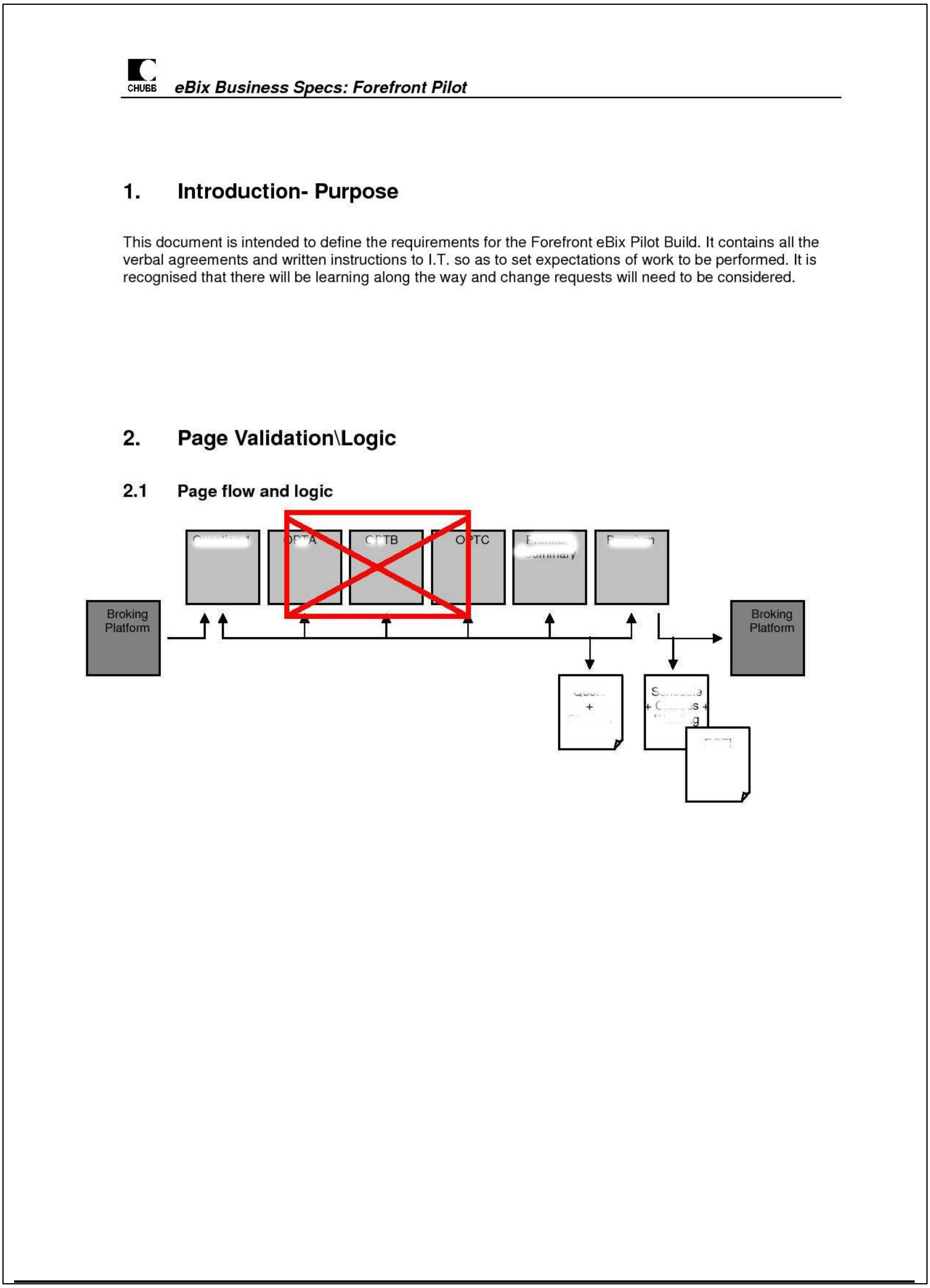
CHUBB eBix Business Specs: Forefront Pilot

eBix Business Specs: Forefront Pilot

Version 0.4.2

Revision History

Version	Date Released	Author	Description of Updates Included
0.1	-	[redacted]	Initial draft;
0.2		[redacted]	Second Draft
03		[redacted]	Updates
0.3.1		[redacted]	Update
0.4		[redacted]	Update
0.4.1		[redacted]	Update ([redacted])
0.4.2		[redacted]	Update ([redacted])



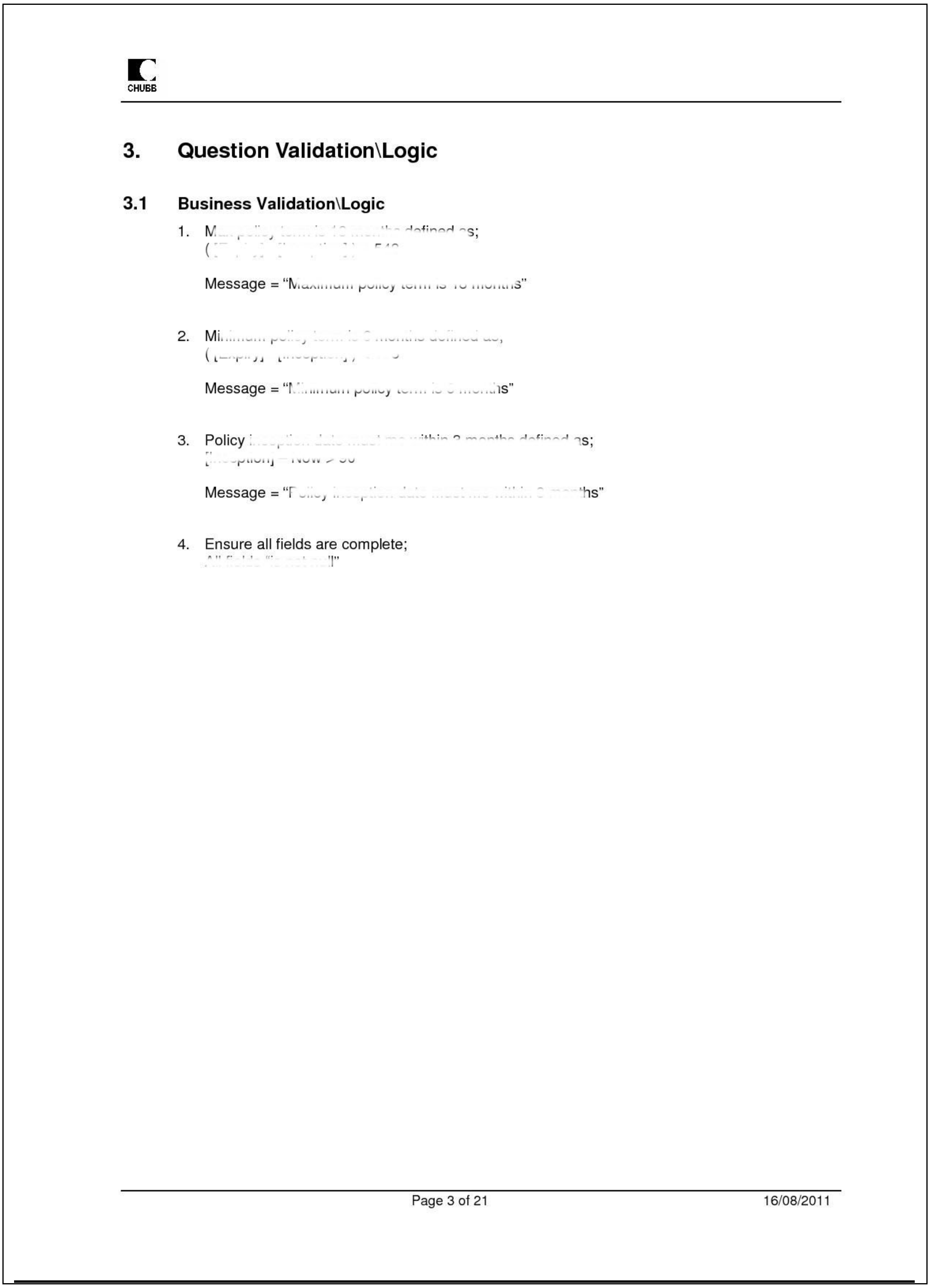
CHUBB *eBix Business Specs: Forefront Pilot*

1. Introduction- Purpose

This document is intended to define the requirements for the Forefront eBix Pilot Build. It contains all the verbal agreements and written instructions to I.T. so as to set expectations of work to be performed. It is recognised that there will be learning along the way and change requests will need to be considered.

2. Page Validation\Logic

2.1 Page flow and logic



3. Question Validation\Logic

3.1 Business Validation\Logic

1. Max policy term is 18 months defined as;
   ([Expiry] – [Inception]) > 548

   Message = "Maximum policy term is 18 months"

2. Minimum policy term is 6 months defined as;
   ([Expiry] – [Inception]) < 183

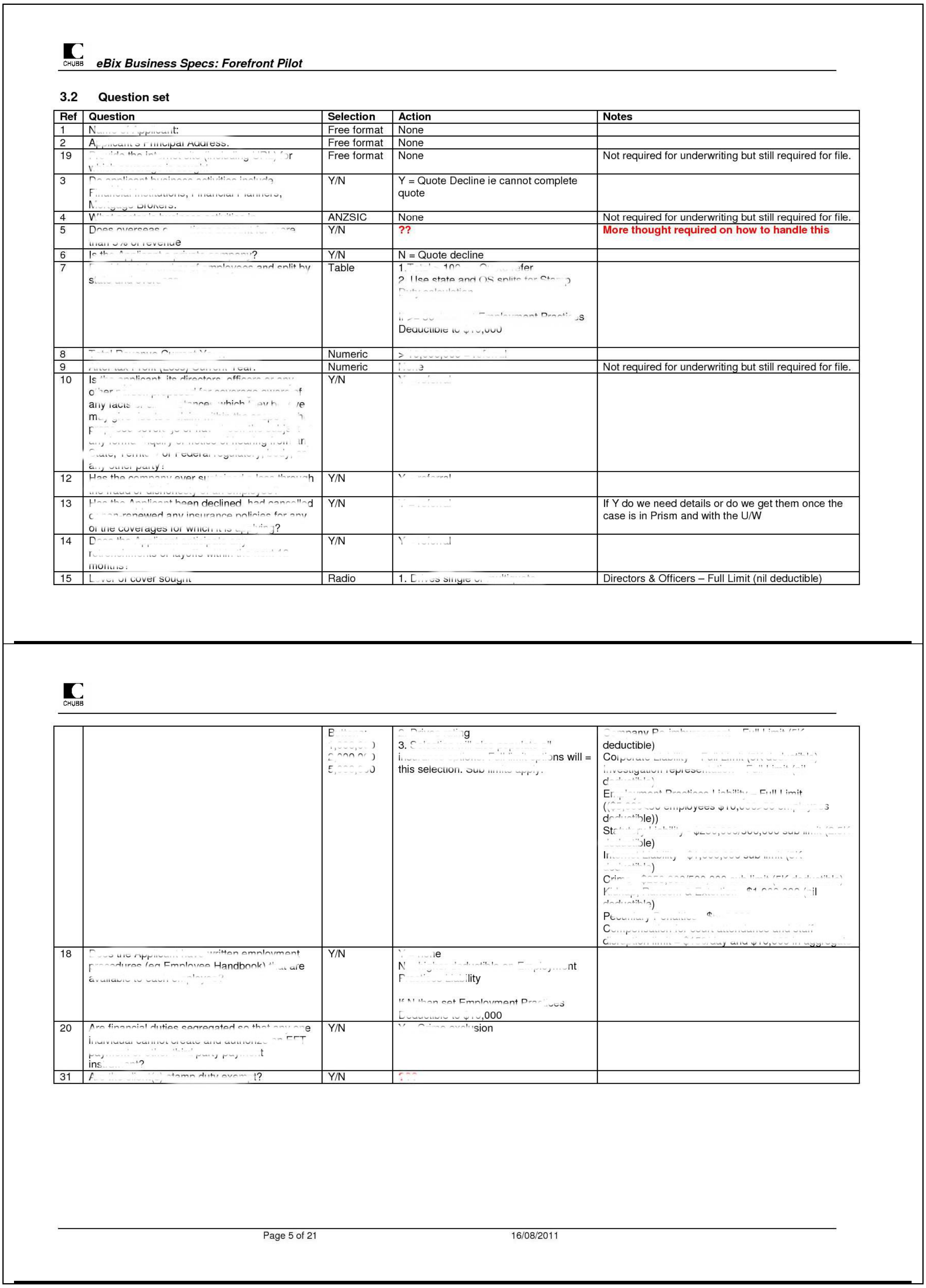
   Message = "Minimum policy term is 6 months"

3. Policy inception date must me within 3 months defined as;
   [Inception] – Now > 90

   Message = "Policy inception date must me within 3 months"

4. Ensure all fields are complete;
   All fields "is not null"

Page 3 of 21 16/08/2011



3.2 Question set

Ref	Question	Selection	Action	Notes
1	Name of Applicant:	Free format	None	
2	Applicant's Principal Address.	Free format	None	
19	Provide the internet site (including URL) for which coverage is sought	Free format	None	Not required for underwriting but still required for file.
3	Do applicant business activities include Financial Institutions, Financial Planners, Mortgage Brokers.	Y/N	Y = Quote Decline ie cannot complete quote	
4	What sector is business activities in	ANZSIC	None	Not required for underwriting but still required for file.
5	Does overseas operations account for more than 5% of revenue	Y/N	??	**More thought required on how to handle this**
6	Is the Applicant a private company?	Y/N	N = Quote decline	
7	Provide total number of employees and split by state and overseas	Table	1.Total > 100 or OS = refer 2. Use state and OS splits for Stamp Duty calculation If >= 50 then set Employment Practices Deductible to $10,000	
8	Total Revenue Current Year	Numeric	> 10,000,000 = referral	
9	After tax Profit (Loss) Current Year.	Numeric	None	Not required for underwriting but still required for file.
10	Is the applicant, its directors, officers or any other person proposed for coverage aware of any facts or circumstances which they believe may give rise to a claim within the scope of the proposed coverage or have been the subject of any formal inquiry or notice of hearing from any State, Territory or Federal regulatory body, or any other party?	Y/N	Y = referral	
12	Has the company ever sustained a loss through the fraud or dishonesty of an employee?	Y/N	Y = referral	
13	Has the Applicant been declined, had cancelled or non-renewed any insurance policies for any of the coverages for which it is applying?	Y/N	Y = referral	If Y do we need details or do we get them once the case is in Prism and with the U/W
14	Does the Applicant anticipate any retrenchments or layoffs within the next 12 months?	Y/N	Y = referral	
15	Level of cover sought	Radio Buttons: 1,000,000 2,000,000 5,000,000	1. Drives single or multiquote 2. Drives rating 3. Selection will also populate all insurance options. Full limit options will = this selection. Sub limits apply.	Directors & Officers – Full Limit (nil deductible) Company Re-imbursement – Full Limit (5K deductible) Corporate Liability – Full Limit (5K deductible) Investigation representation – Full Limit (nil deductible) Employment Practices Liability – Full Limit (($5,000<50 employees $10,000>50 employees deductible)) Statutory Liability - $250,000/500,000 sub limit (2.5K deductible) Internet Liability - $1,000,000 sub limit (5K deductible) Crime - $250,000/500,000 sub limit (5K deductible) Kidnap, Ransom & Extortion - $1,000,000 (nil deductible) Pecuniary Penalties - $[illegible] Compensation for court attendance and staff disruption limit – $150/day and $10,000 in aggregate

18	Does the Applicant have written employment procedures (eg Employee Handbook) that are available to each employee?	Y/N	Y = none
			N = higher deductible on Employment Practices Liability
			
			If N then set Employment Practices Deductible to $10,000	

20	Are financial duties segregated so that any one individual cannot create and authorize an EFT payment or other third party payment instrument?	Y/N	Y = Crime exclusion	
31	Are the client(s) stamp duty exempt?	Y/N	???	

3.2.1 Last Question Changes

- Put questions 10 & 11 together in:
  Is the applicant, its directors, officers or any other person proposed for coverage aware of any facts or circumstances which they believe may give rise to a claim within the scope of the proposed coverage or have been the subject of any formal inquiry or notice of hearing from any State, Territory or Federal regulatory body, or any other party?

- Replace question 12 with:
  Has the company ever sustained a loss through the fraud or dishonesty of an employee?

- Remove questions 15,16 & 17

- Remove questions 20, 21, 22, 23, 24, 25, 26, 27 and replace it with:
  Are financial duties segregated so that any one individual cannot create and authorize an EFT payment or other third party payment instrument?

- Remove question 28

- Remove questions 29 and 30 and replace it with ANZIC triggers.

3.3 Referral Validation\Logic\Process

The following defines what a referral is defined as and how it will be handled.

3.3.1 Referral rules

A quote is to be referred off system where if any of the following are true;

Question 7 > 100
Question 8 > 10,000,000
Question 10 = Yes
Question 12 = Yes
Question 13 = Yes
Question 14 = Yes
“Franchisee” selected from occupation list

3.3.2 How referral occurs

Where any referral is required the system will pop up the following message;

“This submission referred for [referral reason/s]and has been forwarded to [Underwriter here] for further underwriting.”

The system will grey out the button leading to the premium page. [or halt the process]

3.4 Decline Validation\Logic\Process

The following defines what a referral is defined as and how it will be handled.

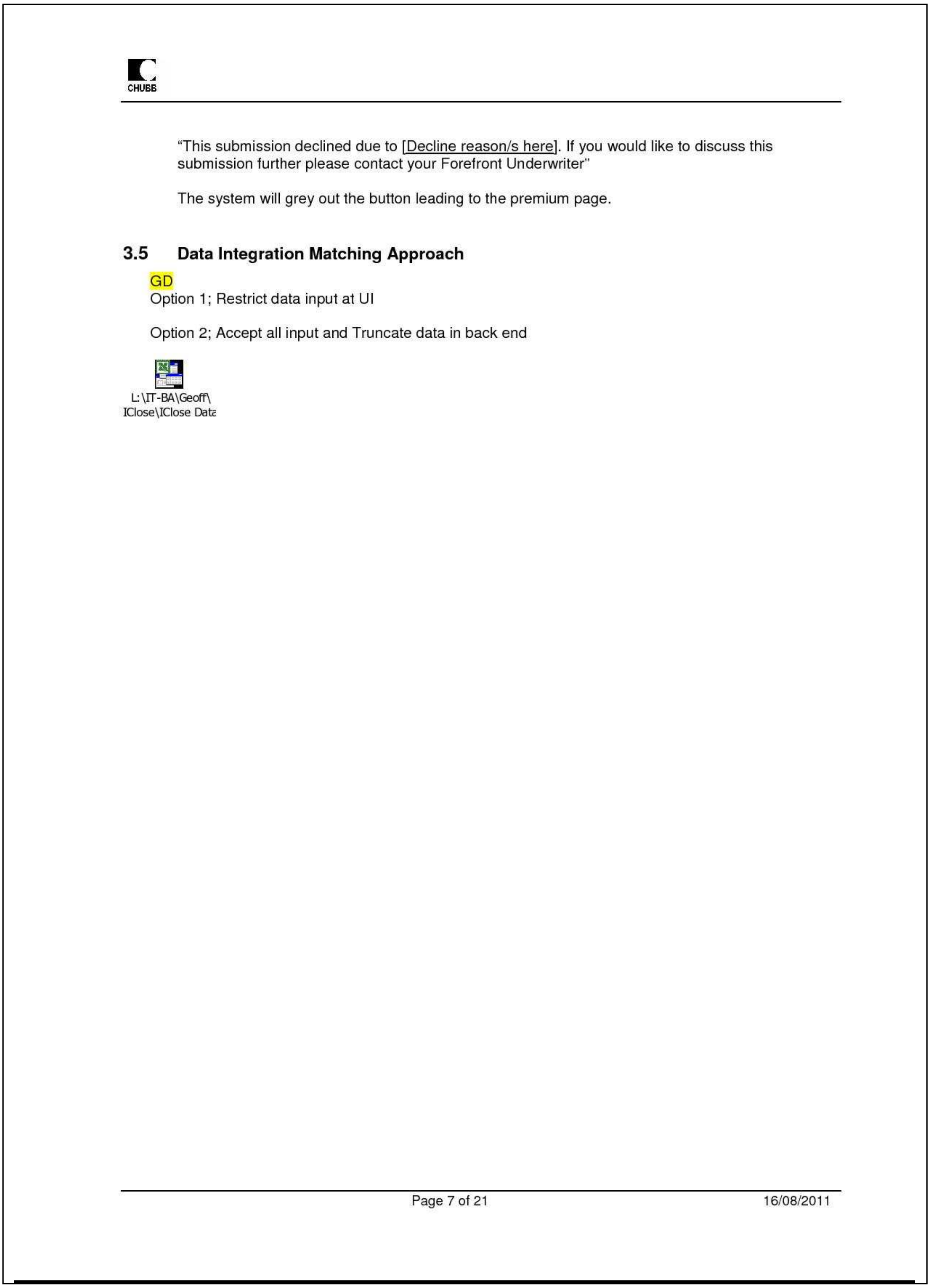
3.4.1 Decline rules

A quote is to be declined where if any of the following are true;

Question 3 = Yes
Question 6 = No

3.4.2 How decline occurs

Where any decline is required the system will pop up the following message;



"This submission declined due to [Decline reason/s here]. If you would like to discuss this submission further please contact your Forefront Underwriter"

The system will grey out the button leading to the premium page.

3.5 Data Integration Matching Approach

GD
Option 1; Restrict data input at UI

Option 2; Accept all input and Truncate data in back end

L:\IT-BA\Geoff\
IClose\IClose Data

4. Rating

The following defines how rating will be calculated;

[Ba[redacted] – [Effective Date] lookup [redacted] -
[redacted]

	Limit				
Revenue	$1,000,000	$2,000,000	$5,000,000		$10,000,000
$0 AND <=$1,000,000	[redacted]	[redacted]	[redacted]		
>$1,000,000 AND <=$3,000,000	[redacted]	[redacted]	[redacted]		
>$3,000,000 AND <=$6,000,000	[redacted]	[redacted]	[redacted]		
>$6,000,000 AND <=$8,000,000	[redacted]	[redacted]	[redacted]		
>$8,000,000 AND <=$10,000,000	[redacted]	[redacted]	[redacted]		
>$10,000,000	[redacted]	[redacted]	[redacted]		[redacted]

[redacted]

[Stamp Duty] = IF([SD Ex[redacted]
(([Base Premium]+[GST])*([NSW]/[EmpTotal]) * look up [NSW Rate]) [redacted]
(([Base [redacted]

[redacted]

*Note: All brokers receive the same commission rate, and there will never be any netting back. GST Com will [redacted]

4.1 GST (on Premium)

- [redacted] rate is charged [redacted] calculated for all Australian states as per the above calculation.

- Premium calculated f[redacted] i.e. not charged as per the above calculation.

4.2 Commission GST

- Current rate is charged[redacted] of commission calculated.
- The percentage of brokerage services performed outside of Australia would not normally attract the pro rata GST on the brokerage services (Commission GST) however it has been agreed that a fla[redacted] will be charged on commission calculated.

4.3 Stamp Duty Rates

Tax rate tables are required to be date effective with a 'current to'. Current stamp duty tax rates by state are as follows;

Australian Capital Territory

Effective Date	Expiry Date	Rate
[redacted]	Current	[redacted]

New South Wales

Effective Date	Expiry Date	Rate
[redacted]	Current	[redacted]

Northern Territory

Effective Date	Expiry Date	Rate
[redacted]	Current	[redacted]

Queensland

Effective Date	Expiry Date	Rate
[redacted]	Current	[redacted]

South Australia

Effective Date	Expiry Date	Rate
[redacted]	Current	[redacted]

Tasmania

Effective Date	Expiry Date	Rate
[redacted]	Current	[redacted]

Victoria

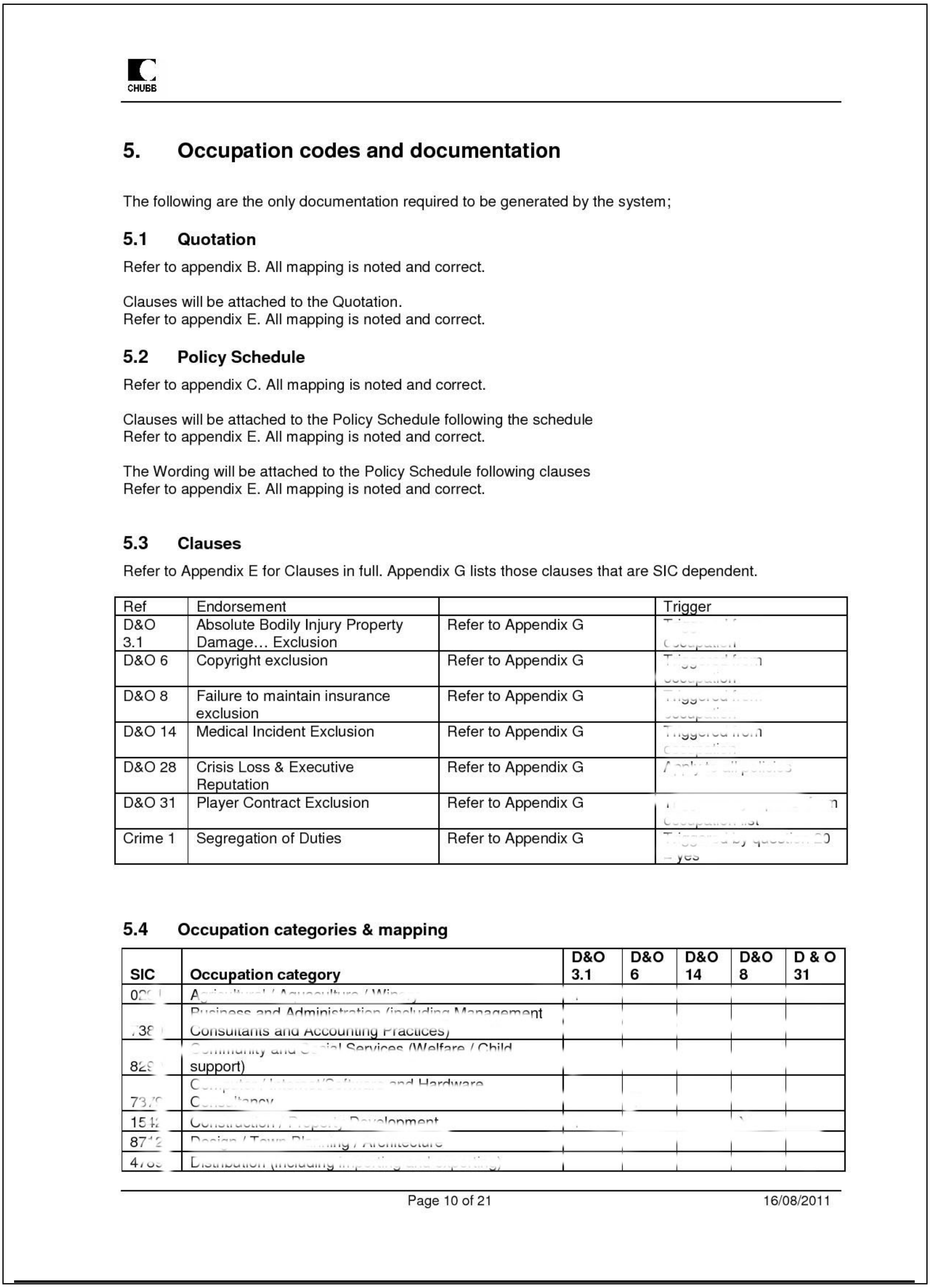
Effective Date	Expiry Date	Rate
[redacted]	Current	[redacted]

Western Australia

Effective Date	Expiry Date	Rate
[redacted]	Current	[redacted]

Overseas

Effective Date	Expiry Date	Rate
[redacted]	Current	[redacted]



5. Occupation codes and documentation

The following are the only documentation required to be generated by the system;

5.1 Quotation

Refer to appendix B. All mapping is noted and correct.

Clauses will be attached to the Quotation.
Refer to appendix E. All mapping is noted and correct.

5.2 Policy Schedule

Refer to appendix C. All mapping is noted and correct.

Clauses will be attached to the Policy Schedule following the schedule
Refer to appendix E. All mapping is noted and correct.

The Wording will be attached to the Policy Schedule following clauses
Refer to appendix E. All mapping is noted and correct.

5.3 Clauses

Refer to Appendix E for Clauses in full. Appendix G lists those clauses that are SIC dependent.

Ref	Endorsement		Trigger
D&O 3.1	Absolute Bodily Injury Property Damage… Exclusion	Refer to Appendix G	Triggered from Occupation
D&O 6	Copyright exclusion	Refer to Appendix G	Triggered from occupation
D&O 8	Failure to maintain insurance exclusion	Refer to Appendix G	Triggered from occupation
D&O 14	Medical Incident Exclusion	Refer to Appendix G	Triggered from occupation
D&O 28	Crisis Loss & Executive Reputation	Refer to Appendix G	Apply to all policies
D&O 31	Player Contract Exclusion	Refer to Appendix G	Triggered from Occupation list
Crime 1	Segregation of Duties	Refer to Appendix G	Triggered by question 20 = yes

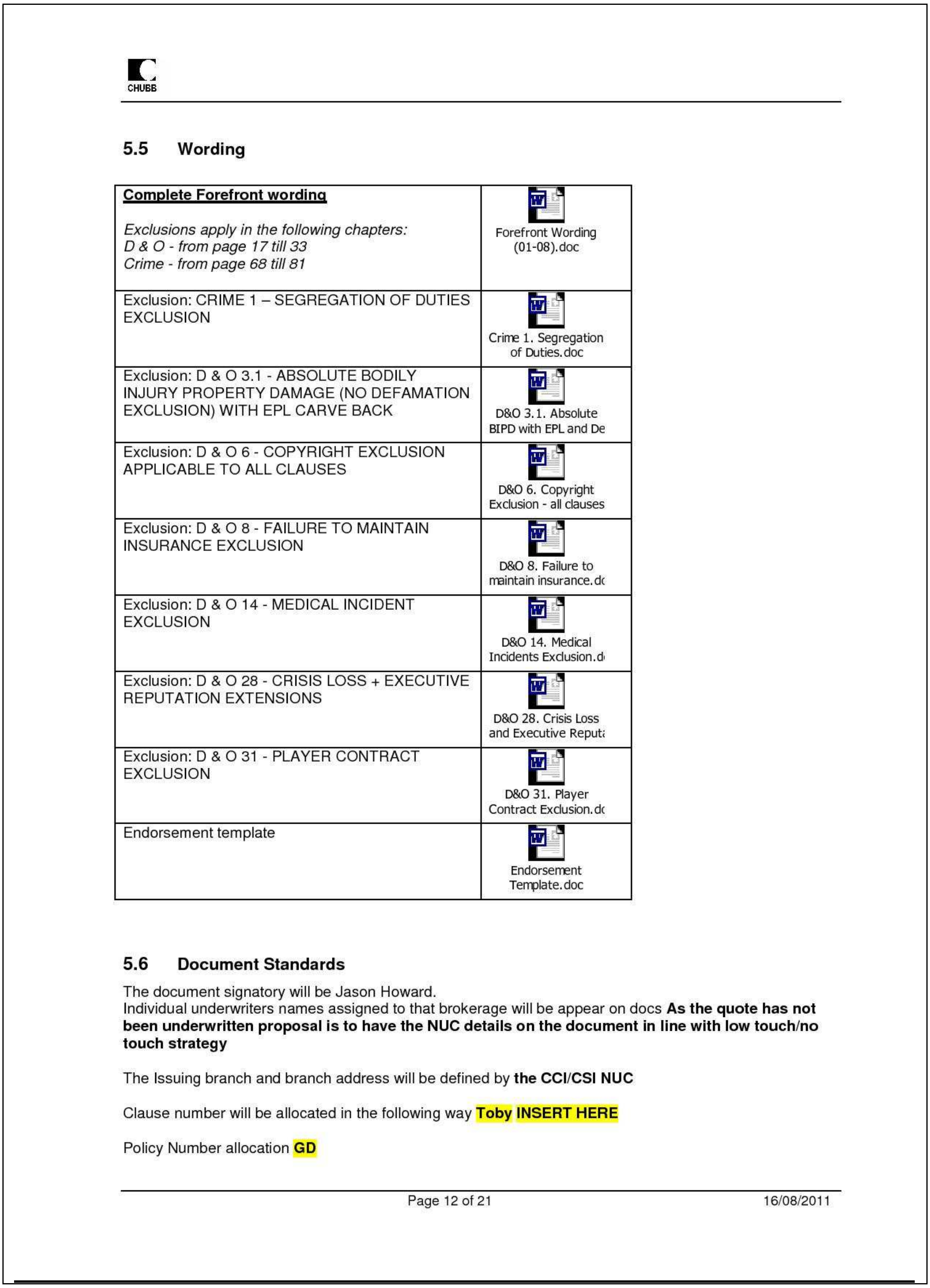
5.4 Occupation categories & mapping

SIC	Occupation category	D&O 3.1	D&O 6	D&O 14	D&O 8	D & O 31
02[illegible]	Agricultural / Aquaculture / Winery					
738[illegible]	Business and Administration (including Management Consultants and Accounting Practices)					
829[illegible]	Community and Social Services (Welfare / Child support)					
737[illegible]	Computer / Internet/Software and Hardware Consultancy					
154[illegible]	Construction / Property Development					
87[illegible]2	Design / Town Planning / Architecture					
47[illegible]	Distribution (including importing and exporting)					

8299	Education (including Day Care centres)					
8711	Engineers				Y	
0291	Farming (Agricultural and Animal)	Y				
6299	Financial Planners / Stockbrokers					
7389	Franchises					
9199	Government / Councils					
7011	Hotels					
7441	Human Resources and Recruitment					
6411	Insurance Brokers (including Finance and Mortgage Brokers)					
8111	Legal Services / Solicitors / Lawyers				Y	
7999	Leisure (including Tourism and Recreation)	Y				
3999	Manufacturing (Solely)					
4225	Manufacturing including Warehousing and Distribution / Wholesale	Y				
4899	Media (including Art, Music and Entertainment)					
8099	Medical Services					
1081	Mining					
8699	Not for Profits					
2099	Pubs / Cafes and Restaurants					
2759	Printing & Publishing Services		Y			
6531	Real Estate and Property Management				Y	
5999	Retailing					
5229	Security, Defence and Protective	Y				
7299	Service Related (including customer service, call centres and beauty therapy)					
7699	Trade Services (metal, wood, electrical and automotive)					
4739	Transport and Storage				Y	
4724	Travel Companies					
4939	Utilities					
7941	Sports					

5.4.1 Last occupation and mapping changes

- Added: Printing & Publishing Services
- Removed Sports from Leisure
- Added: Sports



5.5 Wording

Complete Forefront wording *Exclusions apply in the following chapters:* *D & O - from page 17 till 33* *Crime - from page 68 till 81*	Forefront Wording (01-08).doc
Exclusion: CRIME 1 – SEGREGATION OF DUTIES EXCLUSION	Crime 1. Segregation of Duties.doc
Exclusion: D & O 3.1 - ABSOLUTE BODILY INJURY PROPERTY DAMAGE (NO DEFAMATION EXCLUSION) WITH EPL CARVE BACK	D&O 3.1. Absolute BIPD with EPL and De
Exclusion: D & O 6 - COPYRIGHT EXCLUSION APPLICABLE TO ALL CLAUSES	D&O 6. Copyright Exclusion - all clauses
Exclusion: D & O 8 - FAILURE TO MAINTAIN INSURANCE EXCLUSION	D&O 8. Failure to maintain insurance.dc
Exclusion: D & O 14 - MEDICAL INCIDENT EXCLUSION	D&O 14. Medical Incidents Exclusion.d
Exclusion: D & O 28 - CRISIS LOSS + EXECUTIVE REPUTATION EXTENSIONS	D&O 28. Crisis Loss and Executive Reputa
Exclusion: D & O 31 - PLAYER CONTRACT EXCLUSION	D&O 31. Player Contract Exclusion.dc
Endorsement template	Endorsement Template.doc

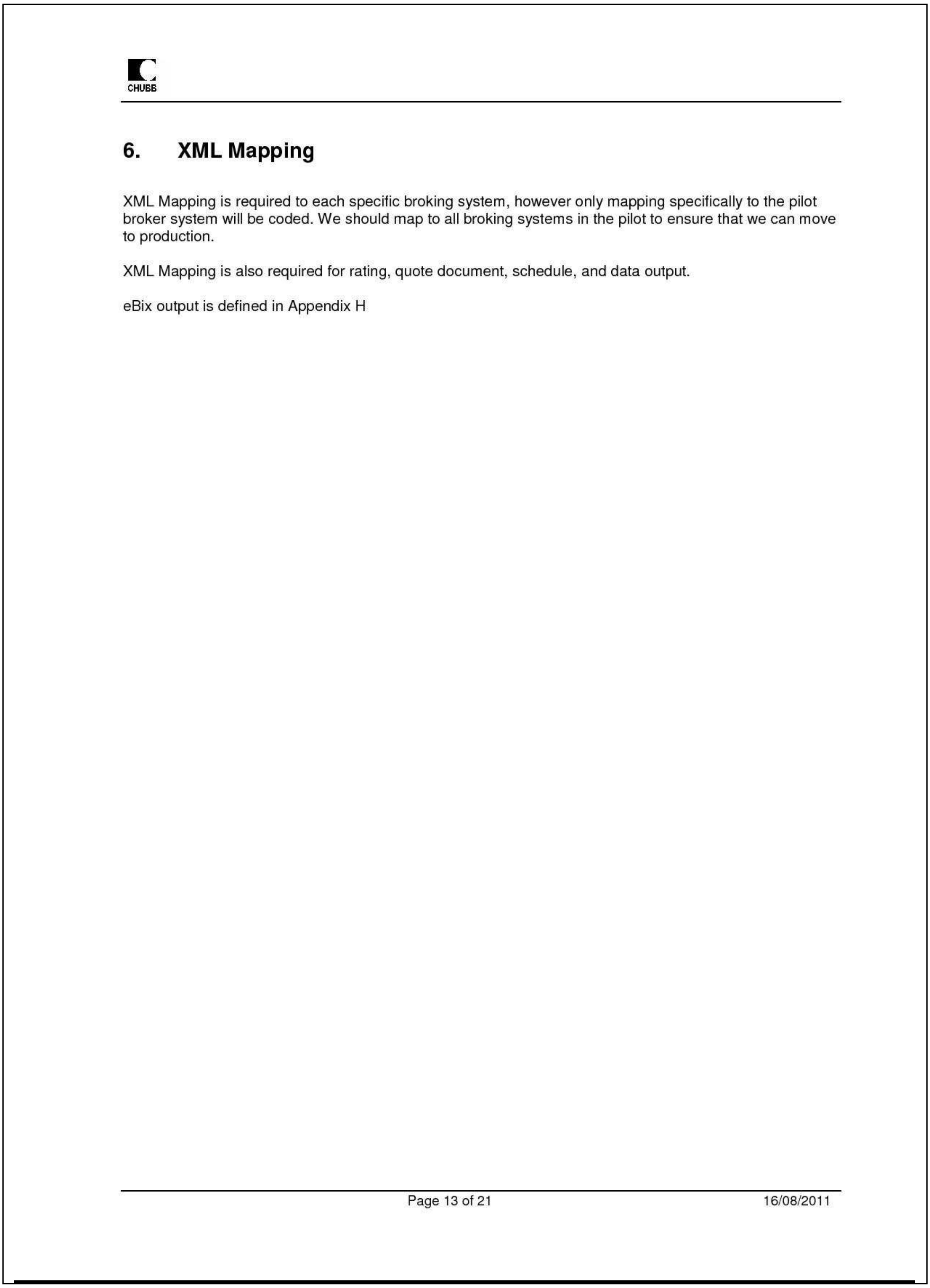
5.6 Document Standards

The document signatory will be Jason Howard.
Individual underwriters names assigned to that brokerage will be appear on docs **As the quote has not been underwritten proposal is to have the NUC details on the document in line with low touch/no touch strategy**

The Issuing branch and branch address will be defined by **the CCI/CSI NUC**

Clause number will be allocated in the following way **Toby INSERT HERE**

Policy Number allocation **GD**

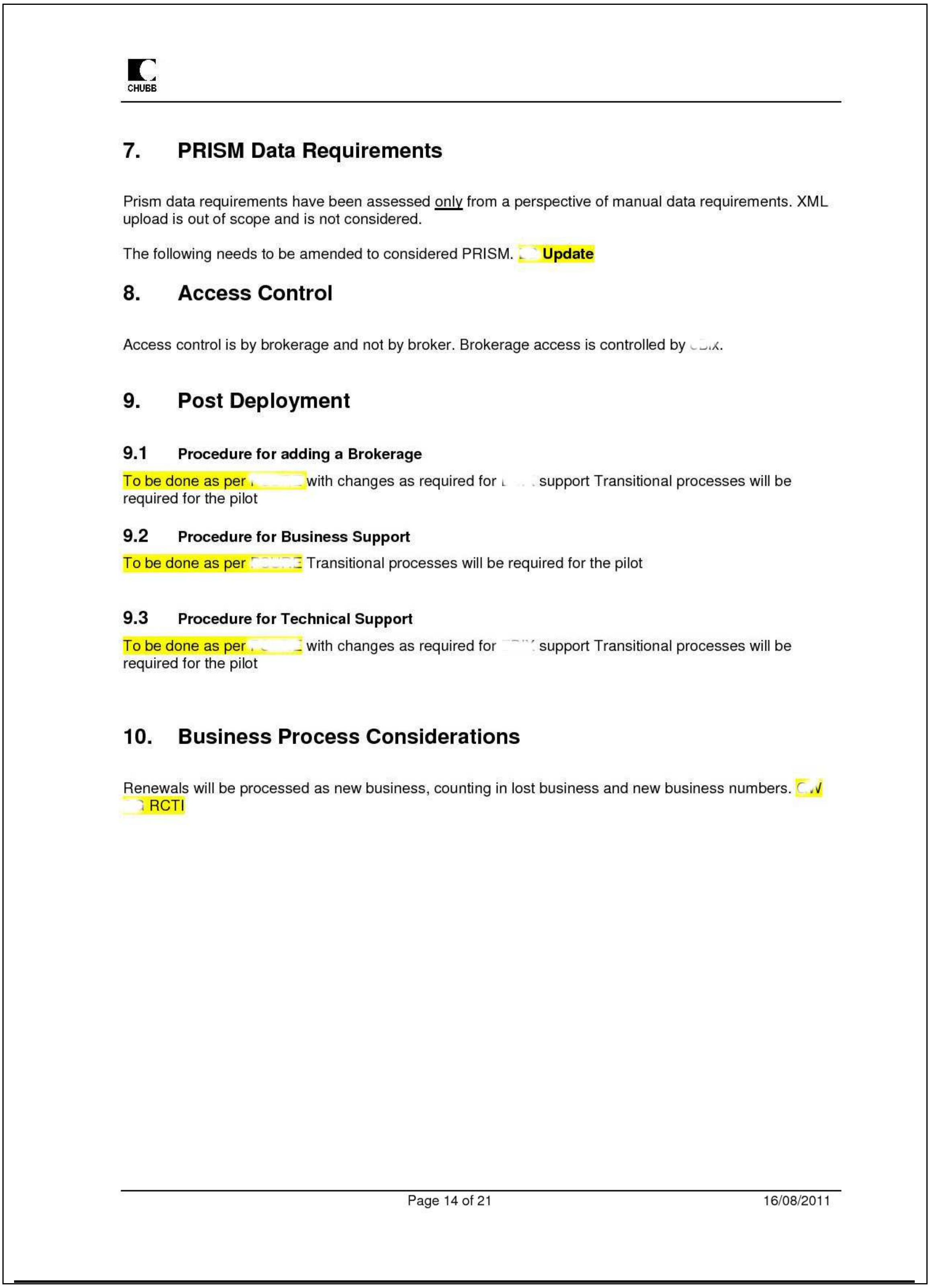


6. XML Mapping

XML Mapping is required to each specific broking system, however only mapping specifically to the pilot broker system will be coded. We should map to all broking systems in the pilot to ensure that we can move to production.

XML Mapping is also required for rating, quote document, schedule, and data output.

eBix output is defined in Appendix H



7. PRISM Data Requirements

Prism data requirements have been assessed only from a perspective of manual data requirements. XML upload is out of scope and is not considered.

The following needs to be amended to considered PRISM. [redacted] **Update**

8. Access Control

Access control is by brokerage and not by broker. Brokerage access is controlled by [redacted].

9. Post Deployment

9.1 Procedure for adding a Brokerage

To be done as per [redacted] with changes as required for [redacted] support Transitional processes will be required for the pilot

9.2 Procedure for Business Support

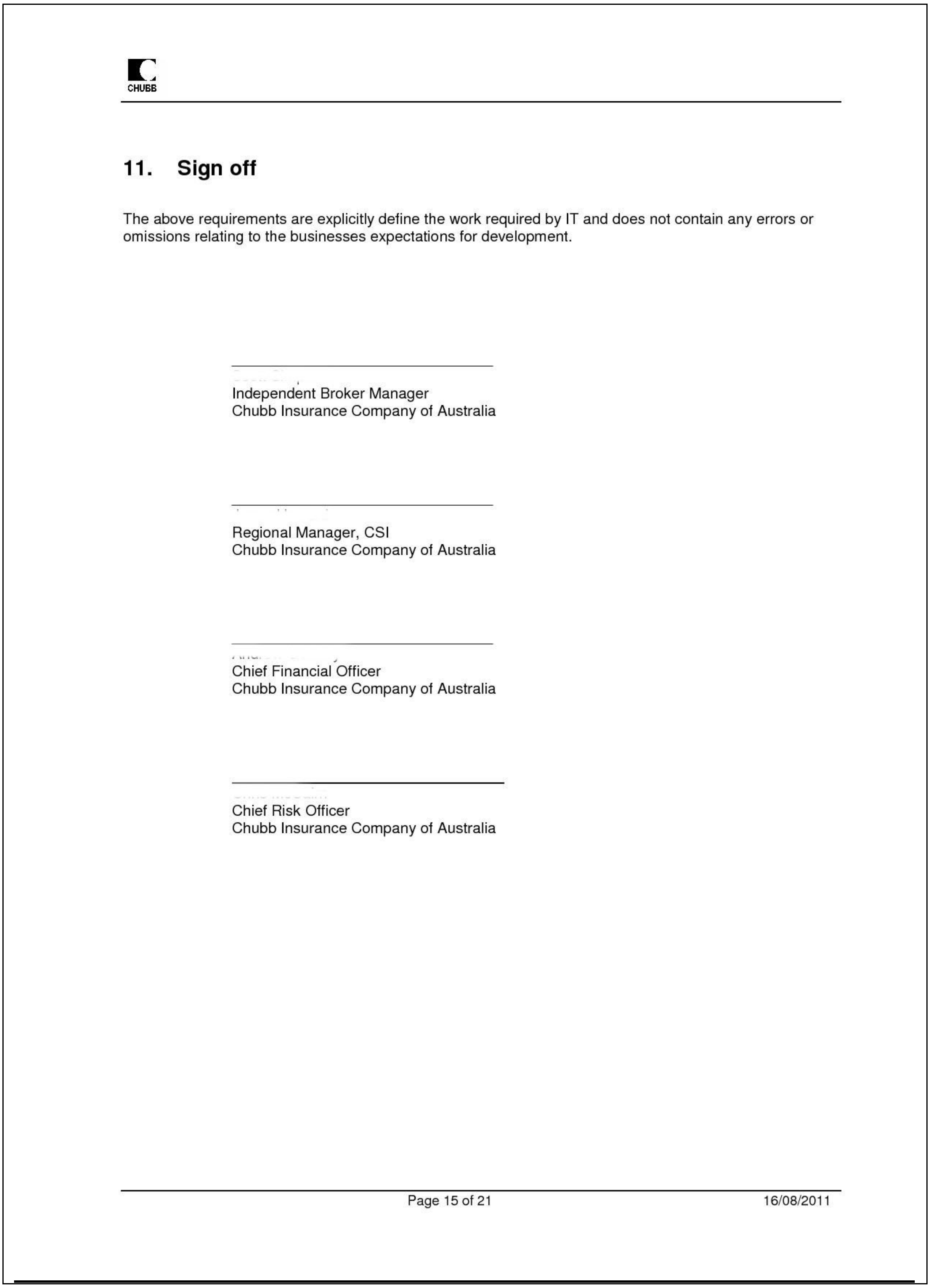
To be done as per [redacted] Transitional processes will be required for the pilot

9.3 Procedure for Technical Support

To be done as per [redacted] with changes as required for [redacted] support Transitional processes will be required for the pilot

10. Business Process Considerations

Renewals will be processed as new business, counting in lost business and new business numbers. [redacted] RCTI



11. Sign off

The above requirements are explicitly define the work required by IT and does not contain any errors or omissions relating to the businesses expectations for development.

____
Independent Broker Manager
Chubb Insurance Company of Australia

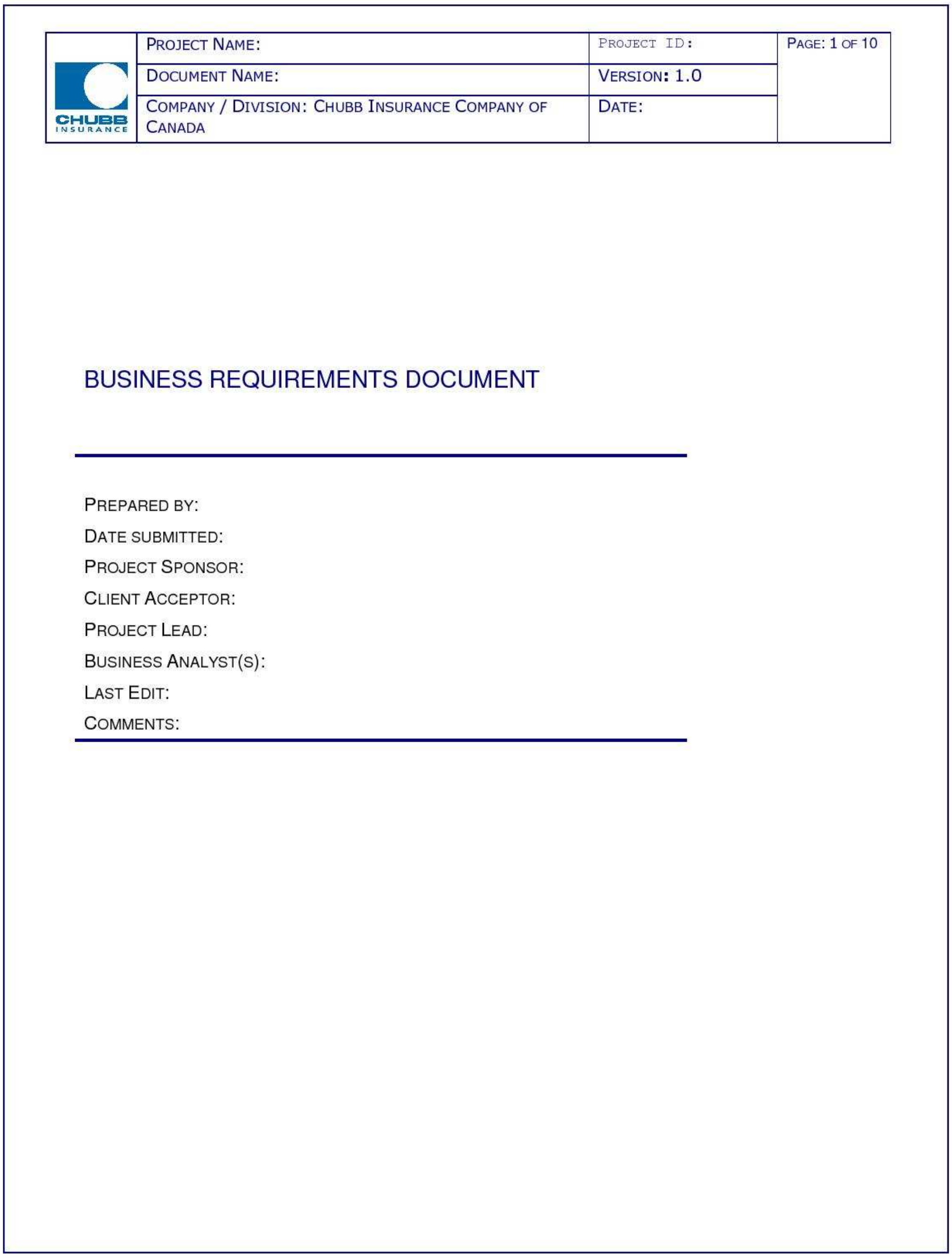
____
Regional Manager, CSI
Chubb Insurance Company of Australia

____
Chief Financial Officer
Chubb Insurance Company of Australia

____
Chief Risk Officer
Chubb Insurance Company of Australia

Appendix N: Business specification document

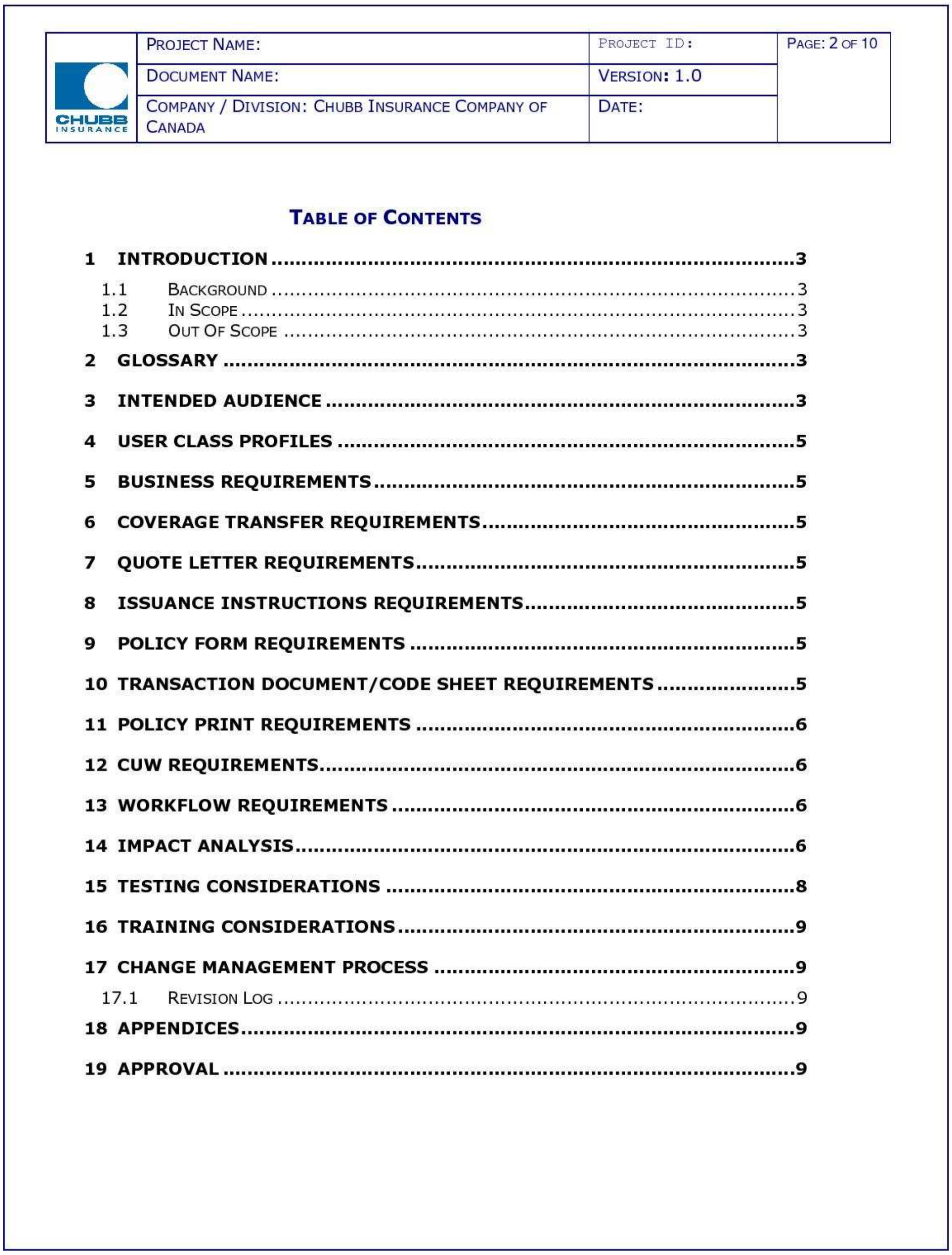
Business requirements template



CHUBB INSURANCE	PROJECT NAME:	PROJECT ID:	PAGE: 1 OF 10
	DOCUMENT NAME:	VERSION: 1.0	
	COMPANY / DIVISION: CHUBB INSURANCE COMPANY OF CANADA	DATE:	

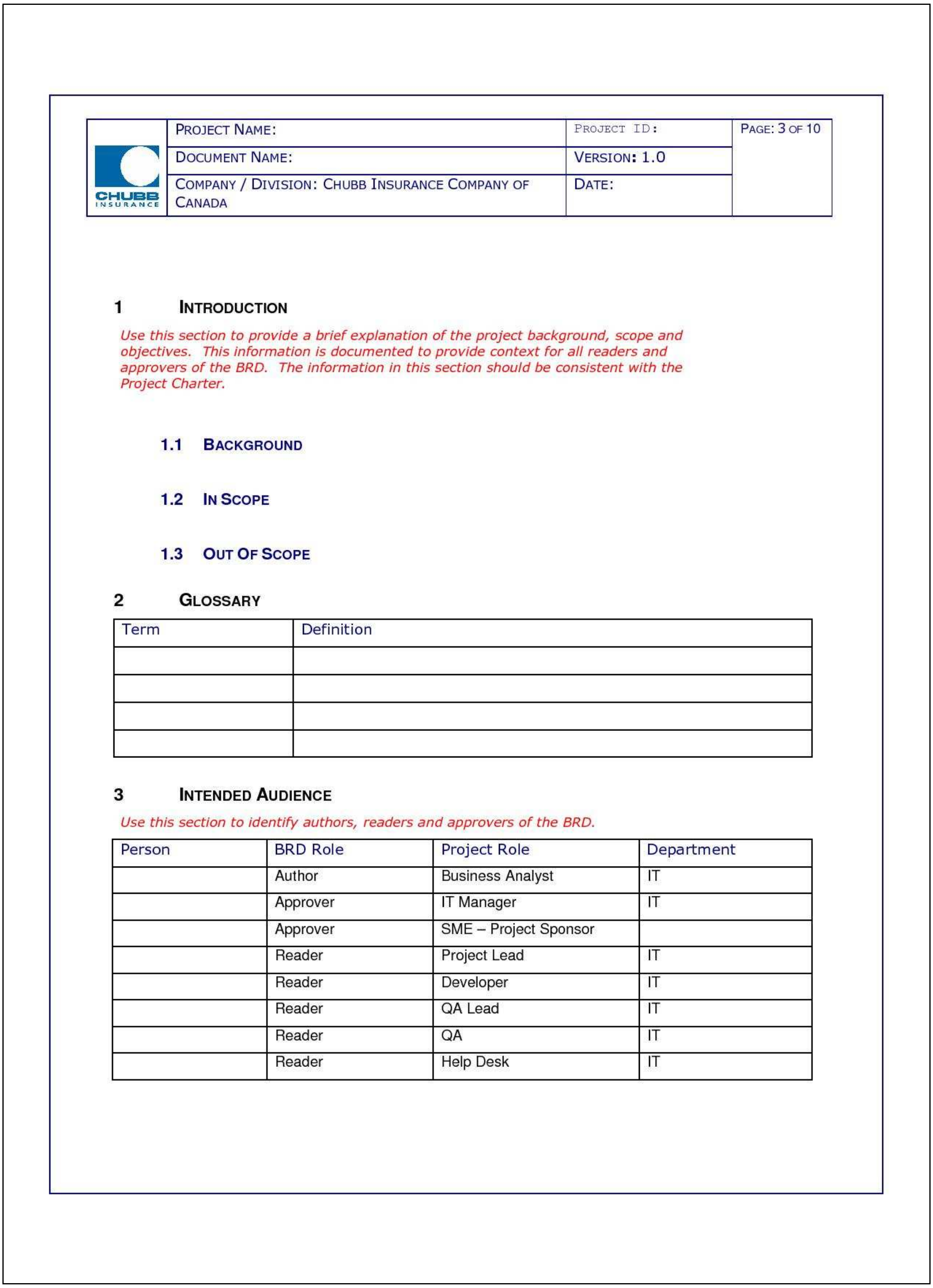
BUSINESS REQUIREMENTS DOCUMENT

PREPARED BY:
DATE SUBMITTED:
PROJECT SPONSOR:
CLIENT ACCEPTOR:
PROJECT LEAD:
BUSINESS ANALYST(S):
LAST EDIT:
COMMENTS:



CHUBB INSURANCE	Project Name:	Project ID:	Page: 2 of 10
	Document Name:	Version: 1.0	
	Company / Division: Chubb Insurance Company of Canada	Date:	

Table of Contents



CHUBB INSURANCE	PROJECT NAME:	PROJECT ID:	PAGE: 3 OF 10
	DOCUMENT NAME:	VERSION: 1.0	
	COMPANY / DIVISION: CHUBB INSURANCE COMPANY OF CANADA	DATE:	

1 INTRODUCTION

Use this section to provide a brief explanation of the project background, scope and objectives. This information is documented to provide context for all readers and approvers of the BRD. The information in this section should be consistent with the Project Charter.

1.1 BACKGROUND

1.2 IN SCOPE

1.3 OUT OF SCOPE

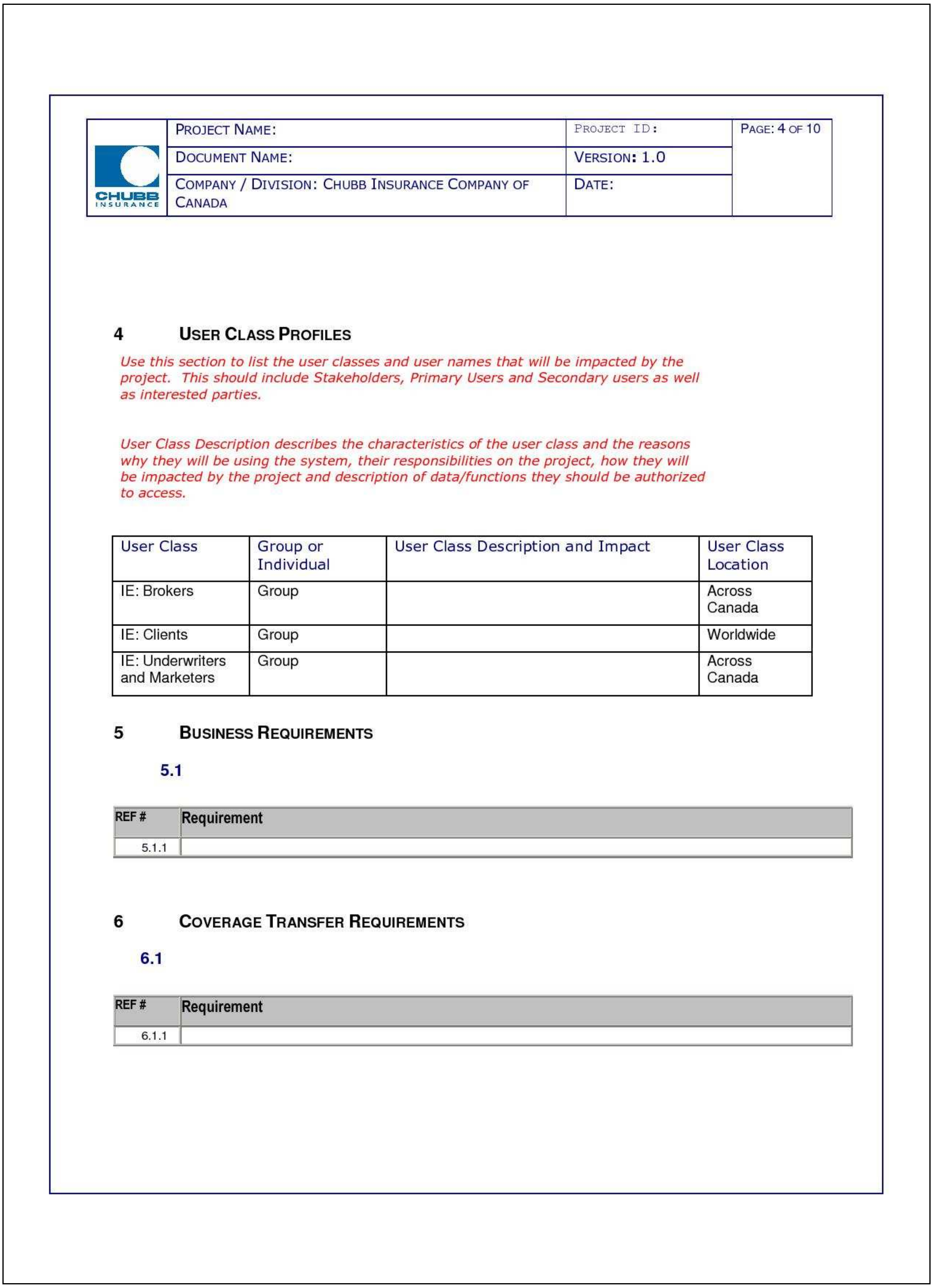
2 GLOSSARY

Term	Definition

3 INTENDED AUDIENCE

Use this section to identify authors, readers and approvers of the BRD.

Person	BRD Role	Project Role	Department
	Author	Business Analyst	IT
	Approver	IT Manager	IT
	Approver	SME – Project Sponsor	
	Reader	Project Lead	IT
	Reader	Developer	IT
	Reader	QA Lead	IT
	Reader	QA	IT
	Reader	Help Desk	IT



CHUBB INSURANCE	PROJECT NAME:	PROJECT ID:	PAGE: 4 OF 10
	DOCUMENT NAME:	VERSION: 1.0	
	COMPANY / DIVISION: CHUBB INSURANCE COMPANY OF CANADA	DATE:	

4 USER CLASS PROFILES

Use this section to list the user classes and user names that will be impacted by the project. This should include Stakeholders, Primary Users and Secondary users as well as interested parties.

User Class Description describes the characteristics of the user class and the reasons why they will be using the system, their responsibilities on the project, how they will be impacted by the project and description of data/functions they should be authorized to access.

User Class	Group or Individual	User Class Description and Impact	User Class Location
IE: Brokers	Group		Across Canada
IE: Clients	Group		Worldwide
IE: Underwriters and Marketers	Group		Across Canada

5 BUSINESS REQUIREMENTS

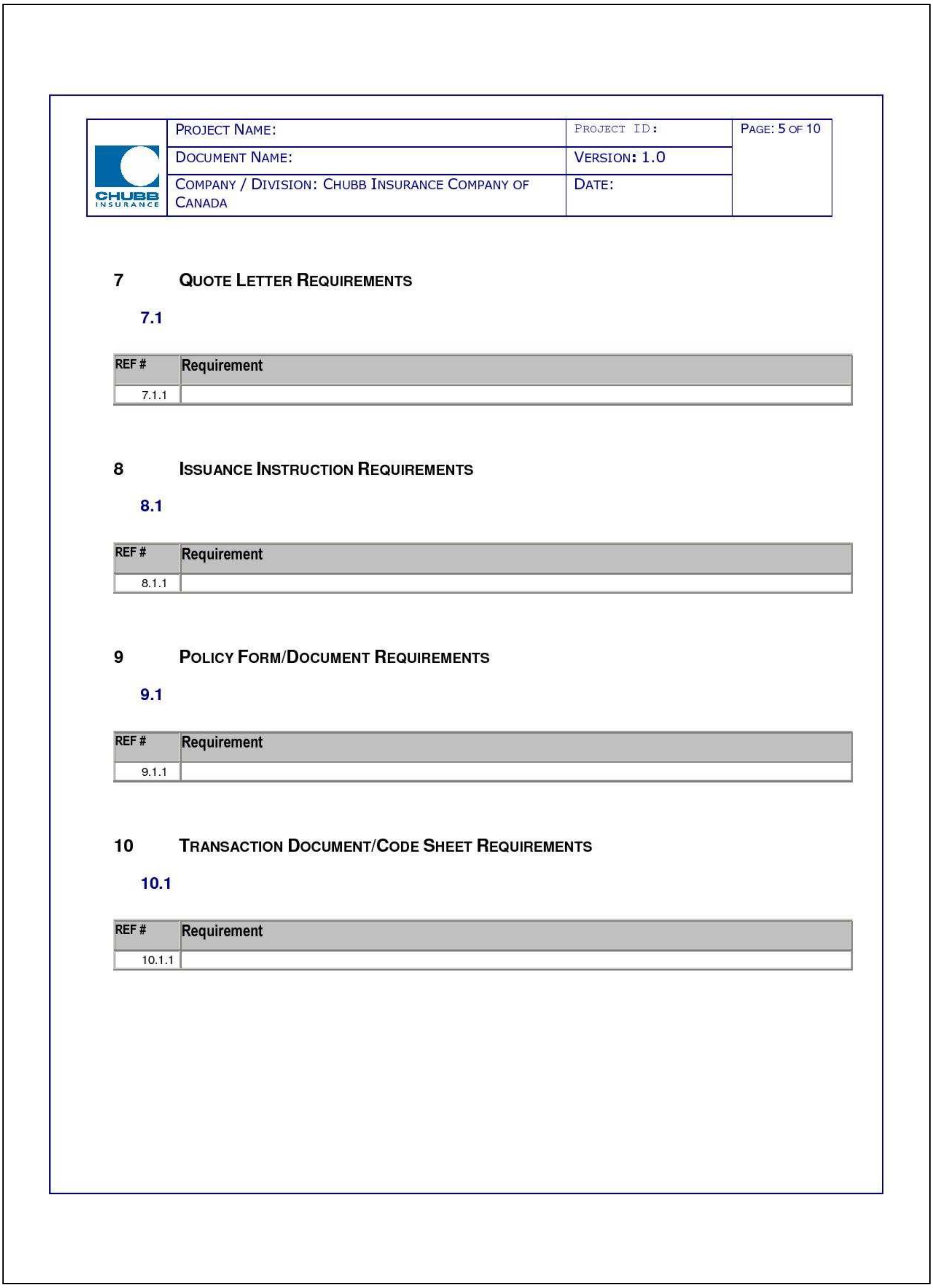
5.1

REF #	Requirement
5.1.1	

6 COVERAGE TRANSFER REQUIREMENTS

6.1

REF #	Requirement
6.1.1	



CHUBB INSURANCE	PROJECT NAME:	PROJECT ID:	PAGE: 5 OF 10
	DOCUMENT NAME:	VERSION: 1.0	
	COMPANY / DIVISION: CHUBB INSURANCE COMPANY OF CANADA	DATE:	

7 QUOTE LETTER REQUIREMENTS

7.1

REF #	Requirement
7.1.1	

8 ISSUANCE INSTRUCTION REQUIREMENTS

8.1

REF #	Requirement
8.1.1	

9 POLICY FORM/DOCUMENT REQUIREMENTS

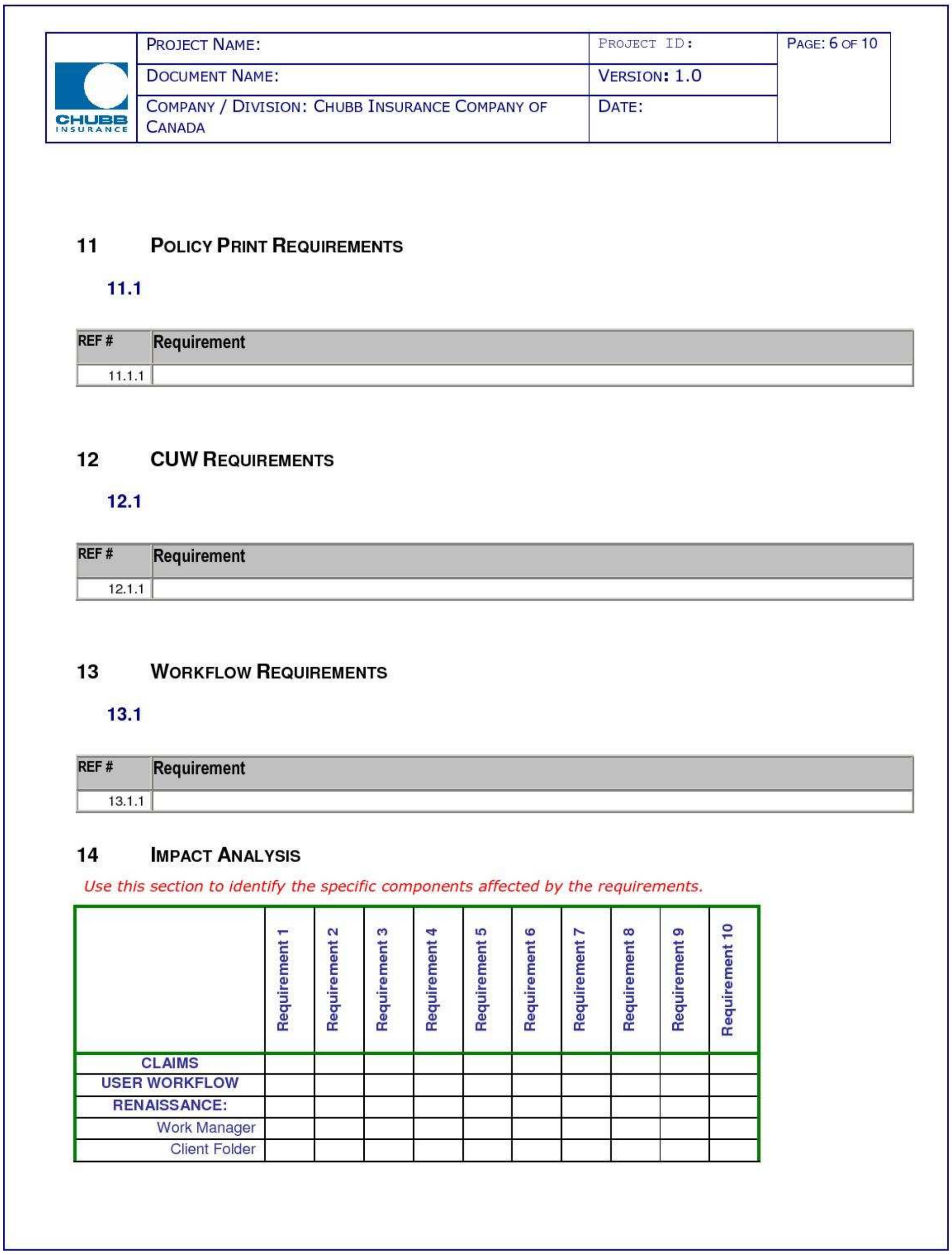
9.1

REF #	Requirement
9.1.1	

10 TRANSACTION DOCUMENT/CODE SHEET REQUIREMENTS

10.1

REF #	Requirement
10.1.1	



CHUBB INSURANCE	Project Name:	Project ID:	Page: 6 of 10
	Document Name:	Version: 1.0	
	Company / Division: Chubb Insurance Company of Canada	Date:	

11 Policy Print Requirements

11.1

REF #	Requirement
11.1.1	

12 CUW Requirements

12.1

REF #	Requirement
12.1.1	

13 Workflow Requirements

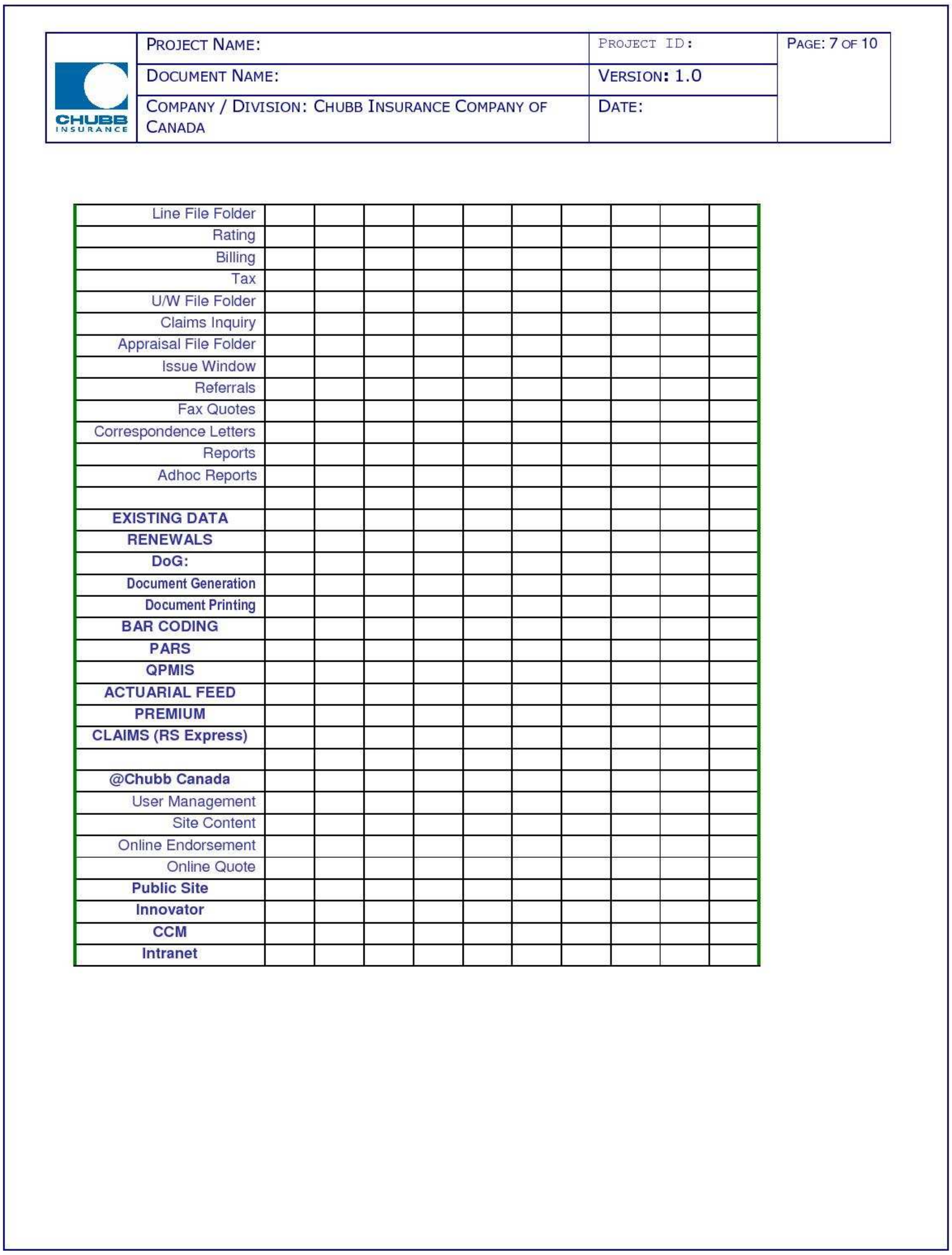
13.1

REF #	Requirement
13.1.1	

14 Impact Analysis

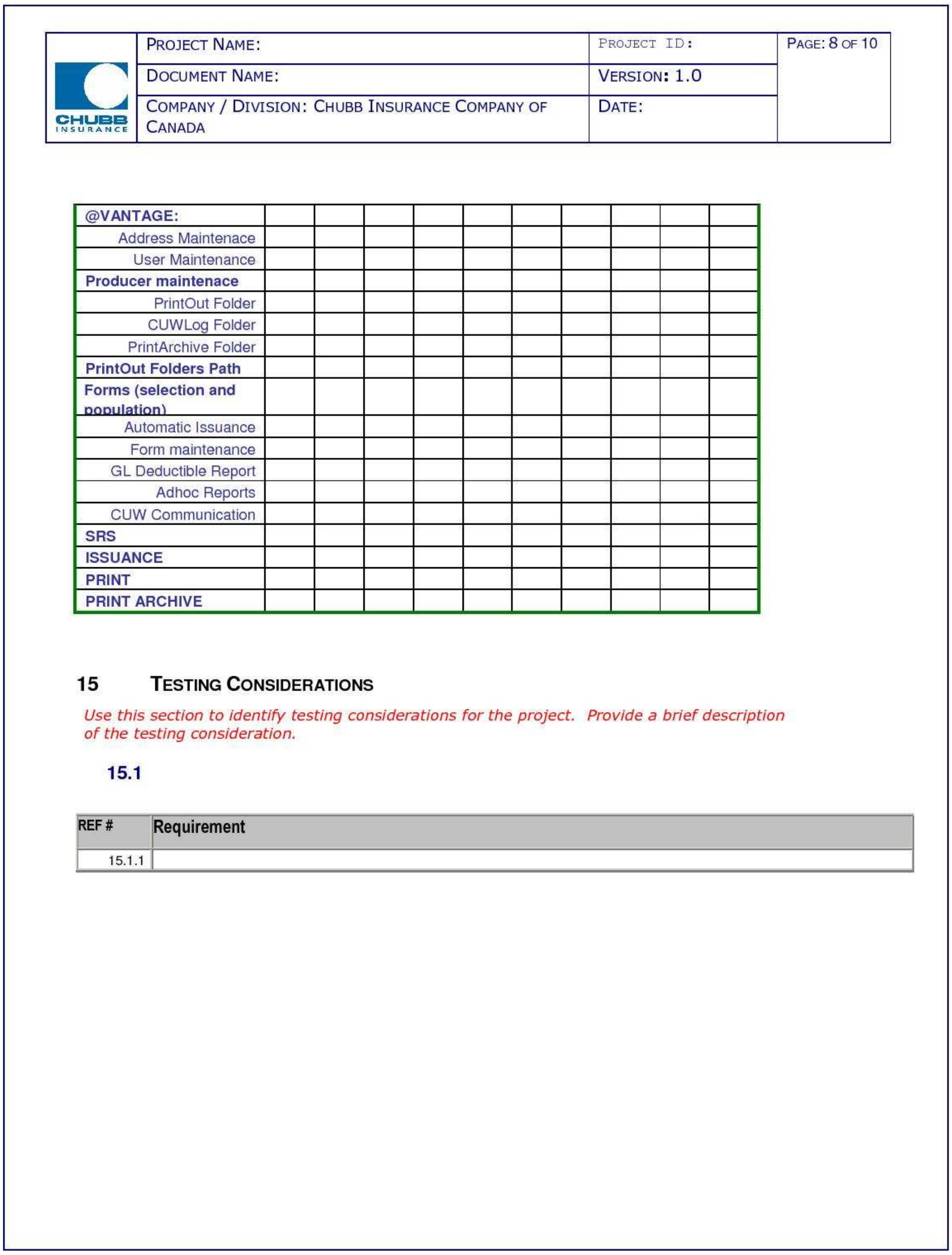
Use this section to identify the specific components affected by the requirements.

	Requirement 1	Requirement 2	Requirement 3	Requirement 4	Requirement 5	Requirement 6	Requirement 7	Requirement 8	Requirement 9	Requirement 10
CLAIMS										
USER WORKFLOW										
RENAISSANCE:										
Work Manager										
Client Folder										



CHUBB INSURANCE	PROJECT NAME:	PROJECT ID:	PAGE: 7 OF 10
	DOCUMENT NAME:	VERSION: 1.0	
	COMPANY / DIVISION: CHUBB INSURANCE COMPANY OF CANADA	DATE:	

Line File Folder										
Rating										
Billing										
Tax										
U/W File Folder										
Claims Inquiry										
Appraisal File Folder										
Issue Window										
Referrals										
Fax Quotes										
Correspondence Letters										
Reports										
Adhoc Reports										
EXISTING DATA										
RENEWALS										
DoG:										
Document Generation										
Document Printing										
BAR CODING										
PARS										
QPMIS										
ACTUARIAL FEED										
PREMIUM										
CLAIMS (RS Express)										
@Chubb Canada										
User Management										
Site Content										
Online Endorsement										
Online Quote										
Public Site										
Innovator										
CCM										
Intranet										



	PROJECT NAME:	PROJECT ID:	PAGE: 8 OF 10
	DOCUMENT NAME:	VERSION: 1.0	
	COMPANY / DIVISION: CHUBB INSURANCE COMPANY OF CANADA	DATE:	

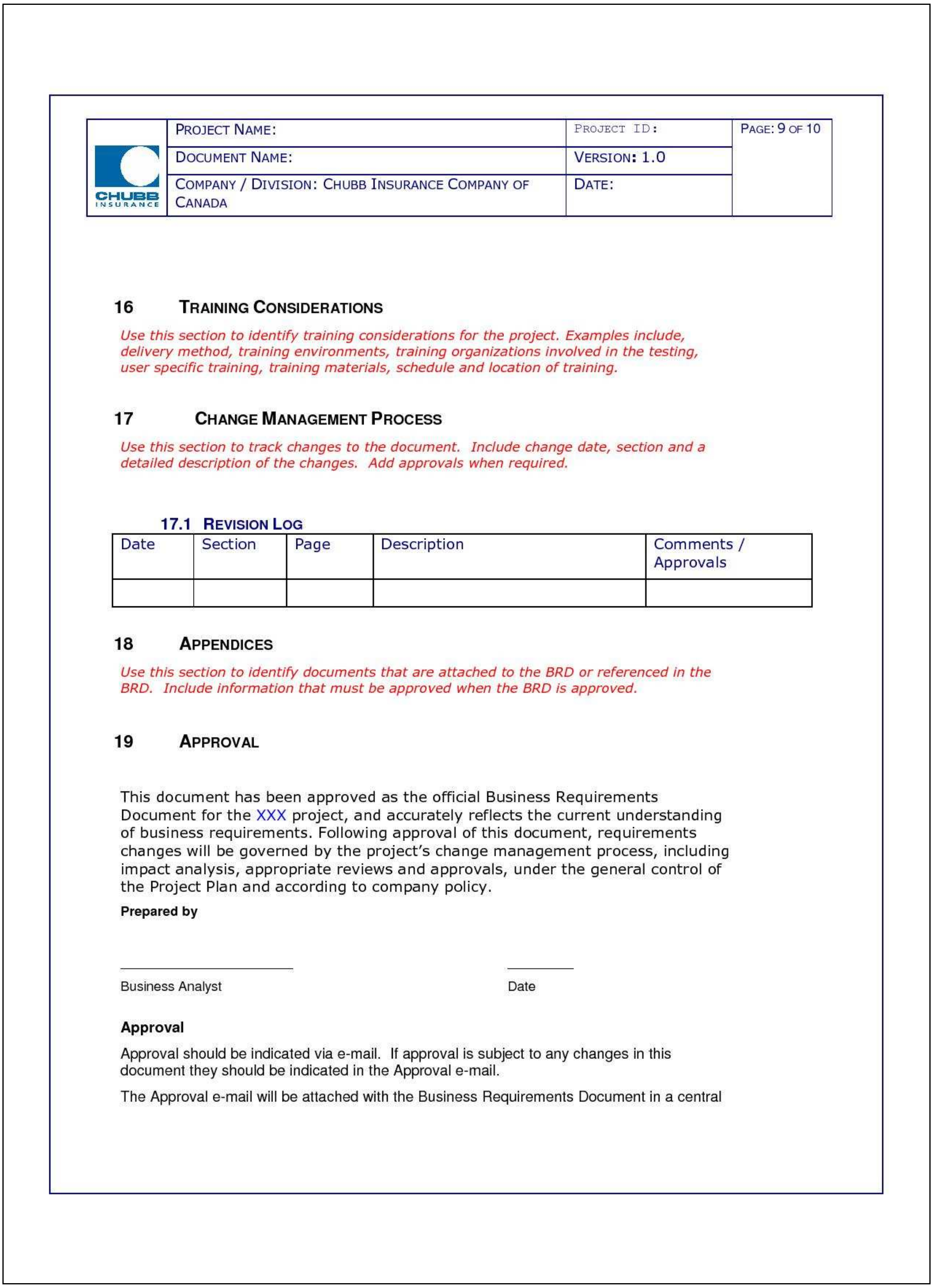
@VANTAGE:										
Address Maintenace										
User Maintenance										
Producer maintenace										
PrintOut Folder										
CUWLog Folder										
PrintArchive Folder										
PrintOut Folders Path										
Forms (selection and population)										
Automatic Issuance										
Form maintenance										
GL Deductible Report										
Adhoc Reports										
CUW Communication										
SRS										
ISSUANCE										
PRINT										
PRINT ARCHIVE										

15 TESTING CONSIDERATIONS

Use this section to identify testing considerations for the project. Provide a brief description of the testing consideration.

15.1

REF #	Requirement
15.1.1	



CHUBB INSURANCE	PROJECT NAME:	PROJECT ID:	PAGE: 9 OF 10
	DOCUMENT NAME:	VERSION: 1.0	
	COMPANY / DIVISION: CHUBB INSURANCE COMPANY OF CANADA	DATE:	

16 TRAINING CONSIDERATIONS

Use this section to identify training considerations for the project. Examples include, delivery method, training environments, training organizations involved in the testing, user specific training, training materials, schedule and location of training.

17 CHANGE MANAGEMENT PROCESS

Use this section to track changes to the document. Include change date, section and a detailed description of the changes. Add approvals when required.

17.1 REVISION LOG

Date	Section	Page	Description	Comments / Approvals

18 APPENDICES

Use this section to identify documents that are attached to the BRD or referenced in the BRD. Include information that must be approved when the BRD is approved.

19 APPROVAL

This document has been approved as the official Business Requirements Document for the XXX project, and accurately reflects the current understanding of business requirements. Following approval of this document, requirements changes will be governed by the project's change management process, including impact analysis, appropriate reviews and approvals, under the general control of the Project Plan and according to company policy.

Prepared by

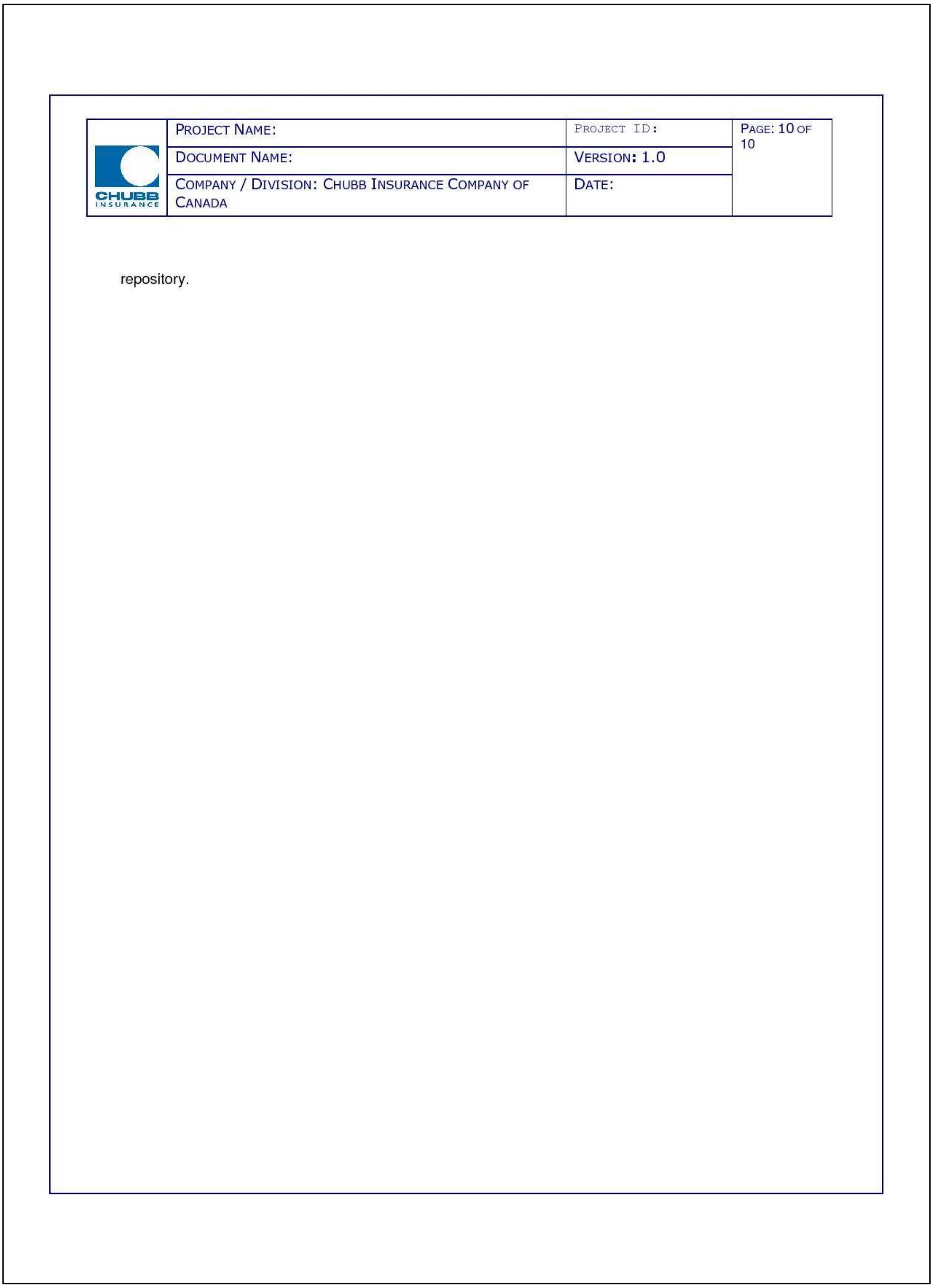
________________________ __________

Business Analyst Date

Approval

Approval should be indicated via e-mail. If approval is subject to any changes in this document they should be indicated in the Approval e-mail.

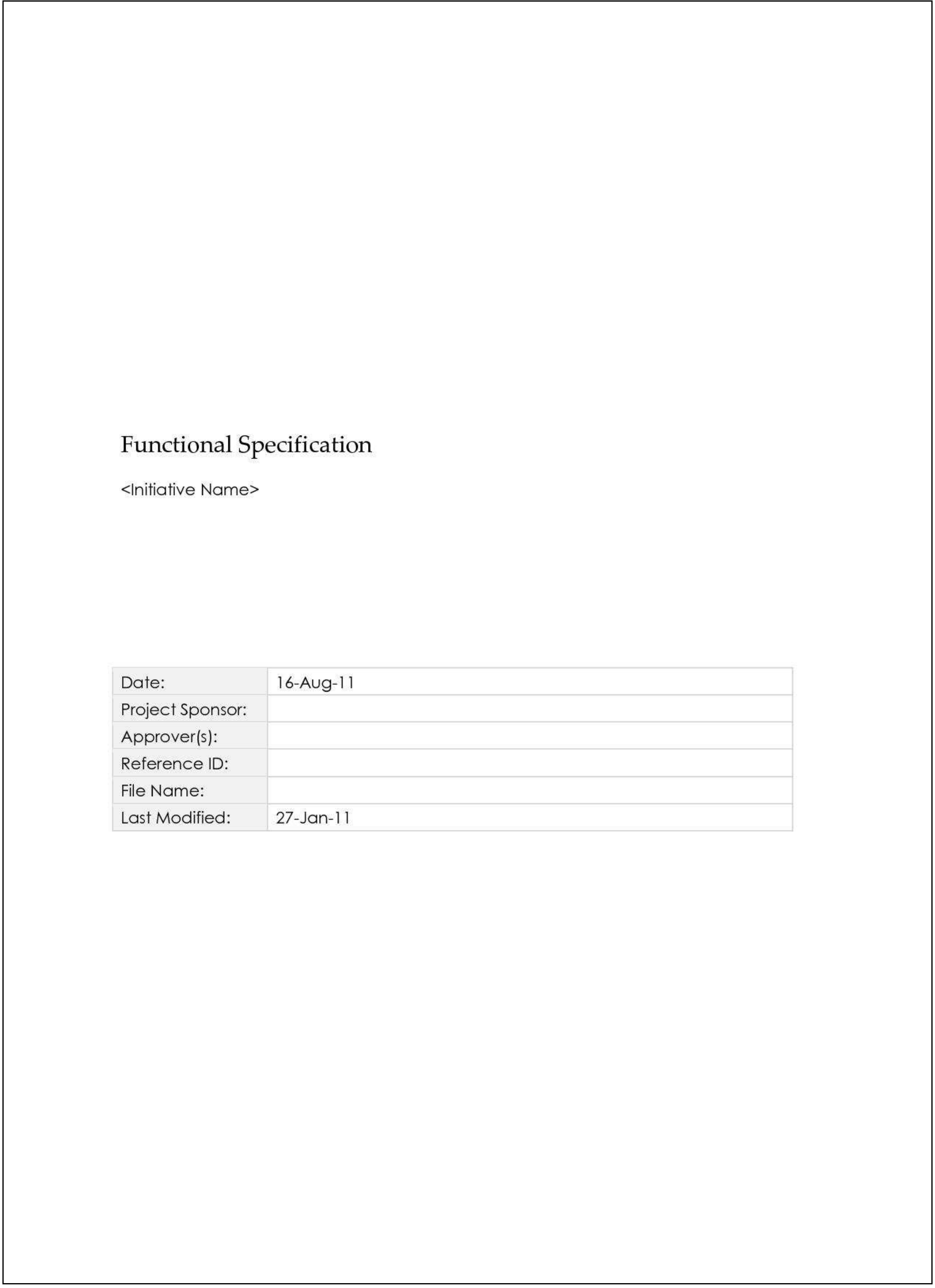
The Approval e-mail will be attached with the Business Requirements Document in a central



CHUBB INSURANCE	PROJECT NAME:	PROJECT ID:	PAGE: 10 OF 10
	DOCUMENT NAME:	VERSION: 1.0	
	COMPANY / DIVISION: CHUBB INSURANCE COMPANY OF CANADA	DATE:	

repository.

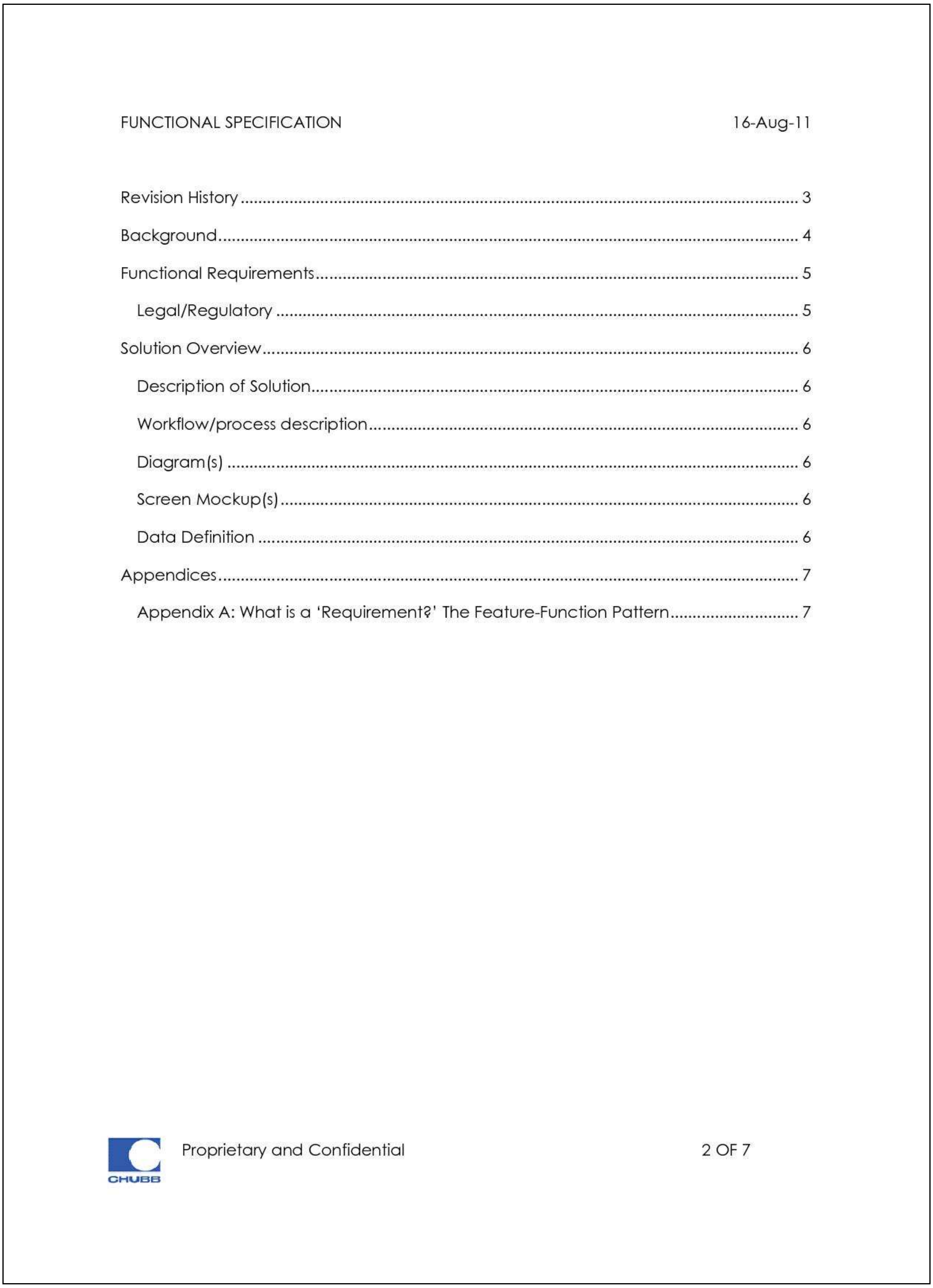
Functional specification template

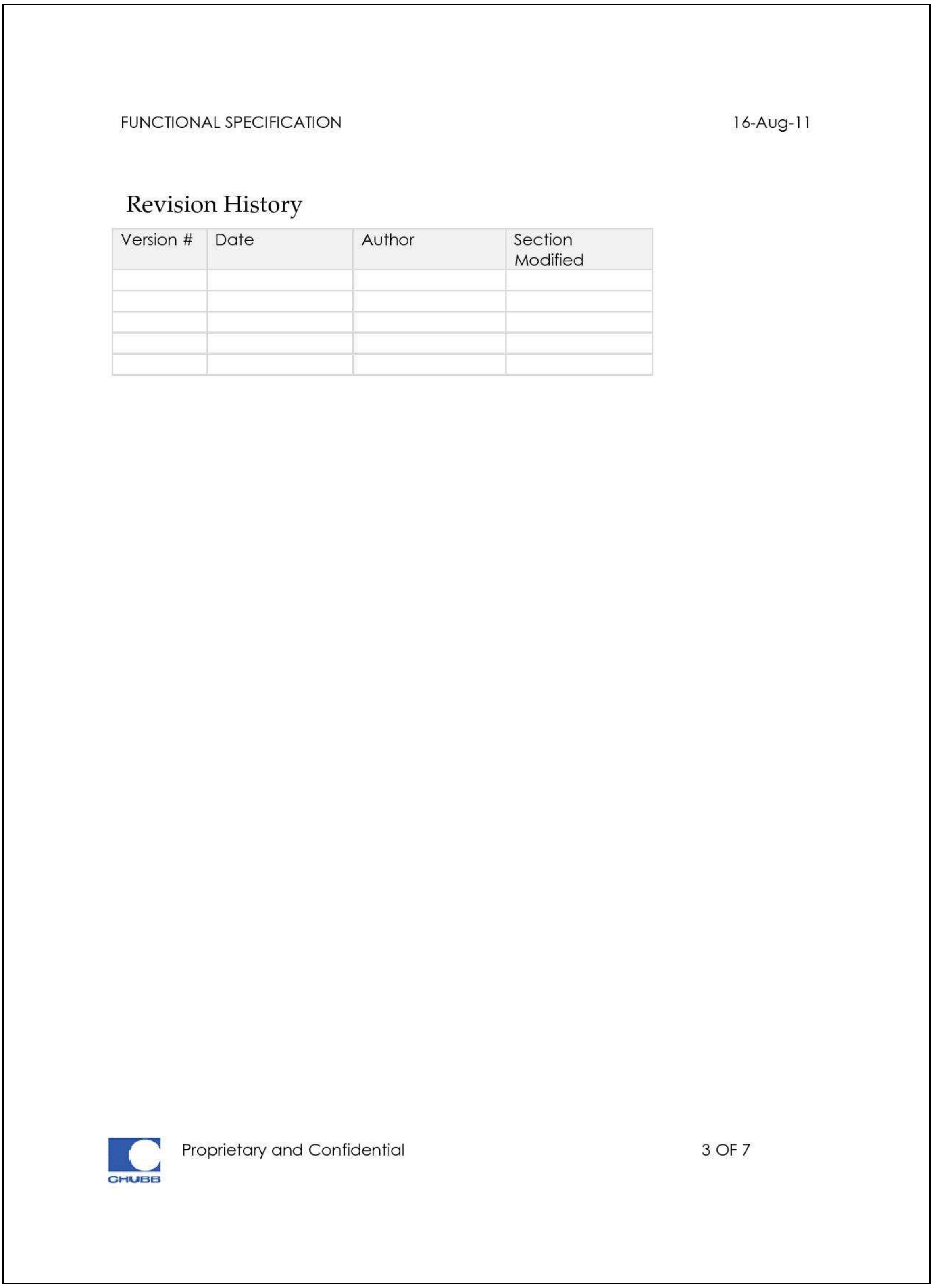


Functional Specification

<Initiative Name>

Date:	16-Aug-11
Project Sponsor:	
Approver(s):	
Reference ID:	
File Name:	
Last Modified:	27-Jan-11



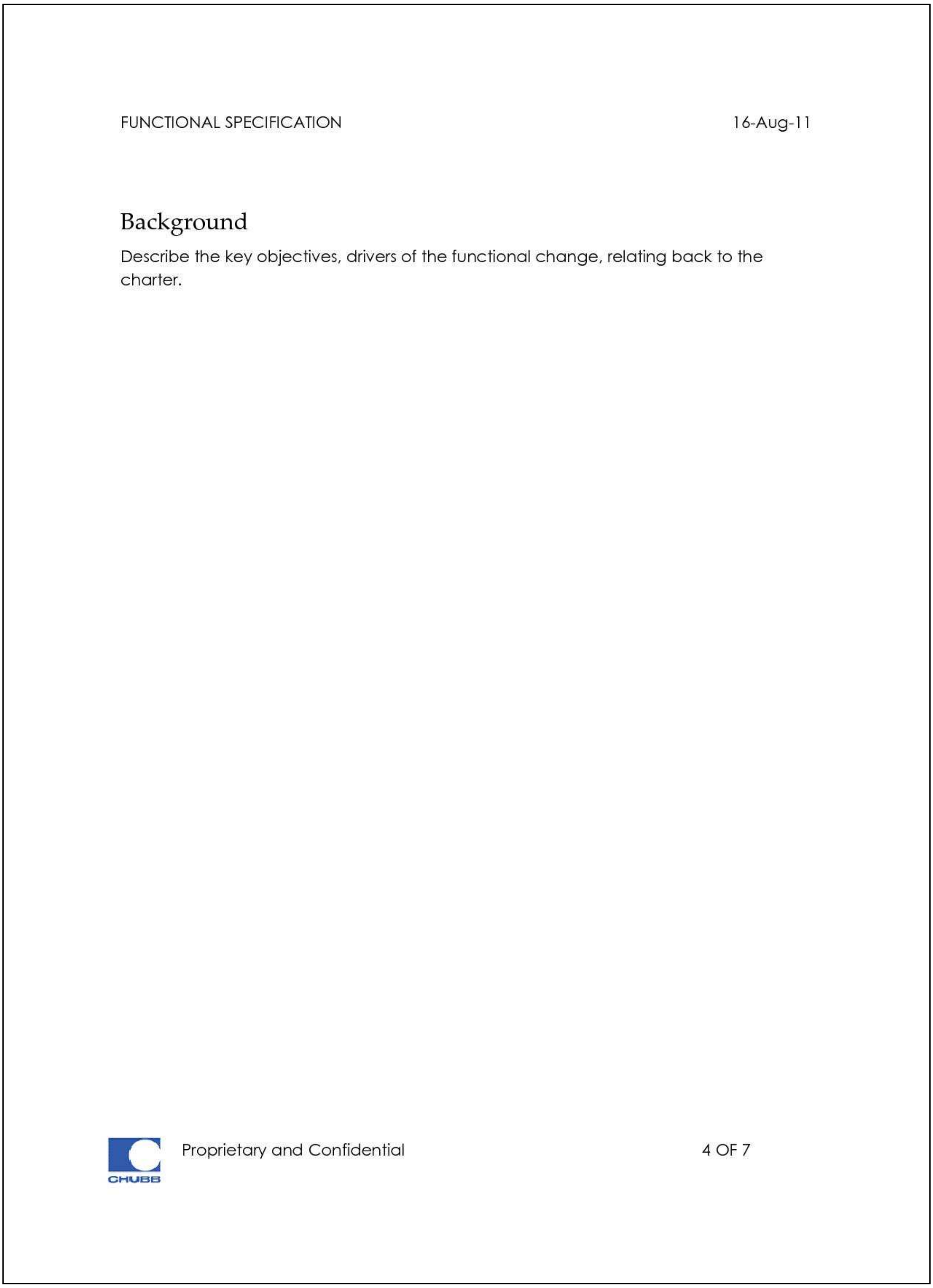


FUNCTIONAL SPECIFICATION 16-Aug-11

Revision History

Version #	Date	Author	Section Modified

Proprietary and Confidential 3 OF 7

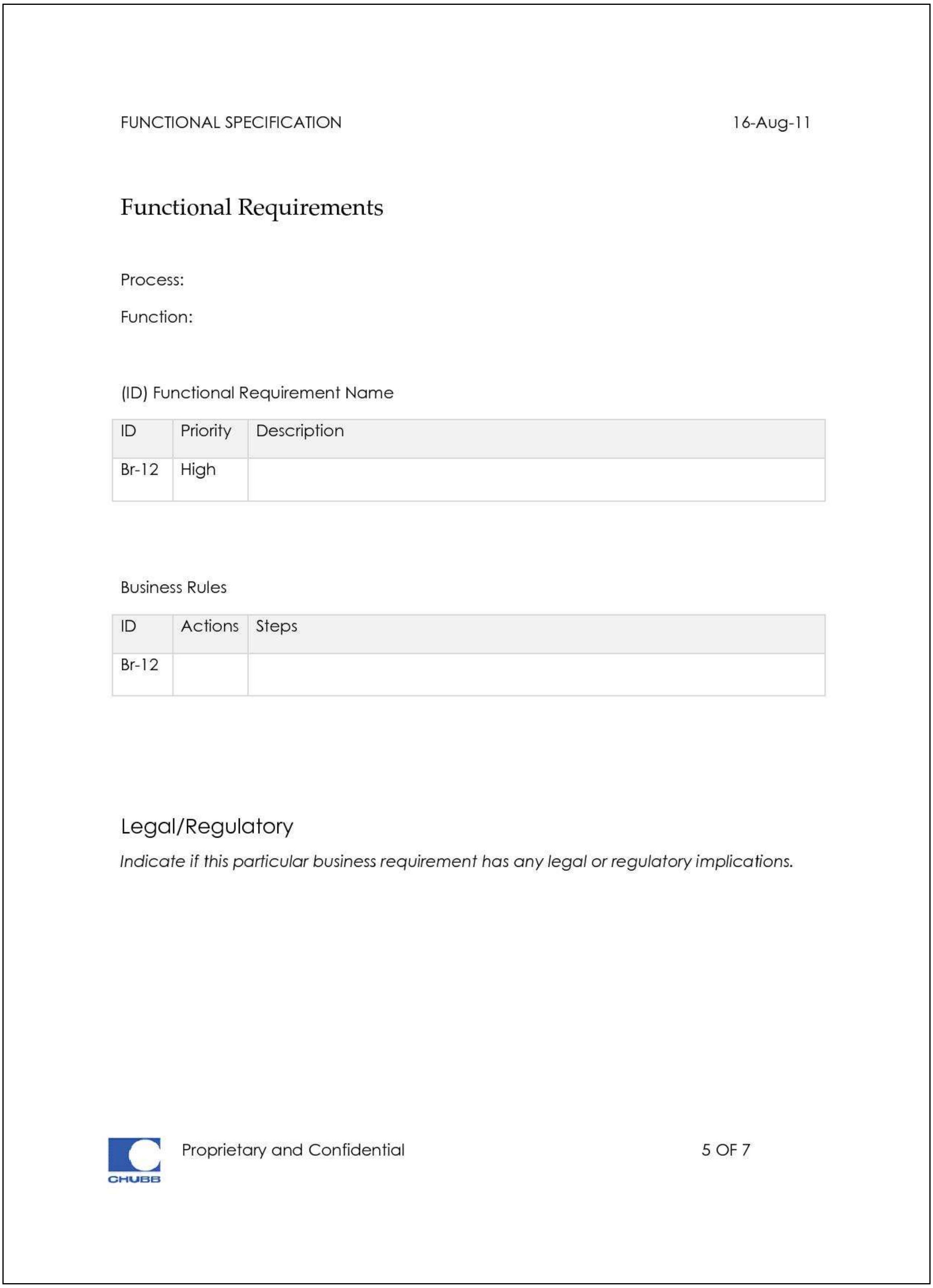


FUNCTIONAL SPECIFICATION 16-Aug-11

Background

Describe the key objectives, drivers of the functional change, relating back to the charter.

Proprietary and Confidential 4 OF 7



FUNCTIONAL SPECIFICATION 16-Aug-11

Functional Requirements

Process:

Function:

(ID) Functional Requirement Name

ID	Priority	Description
Br-12	High	

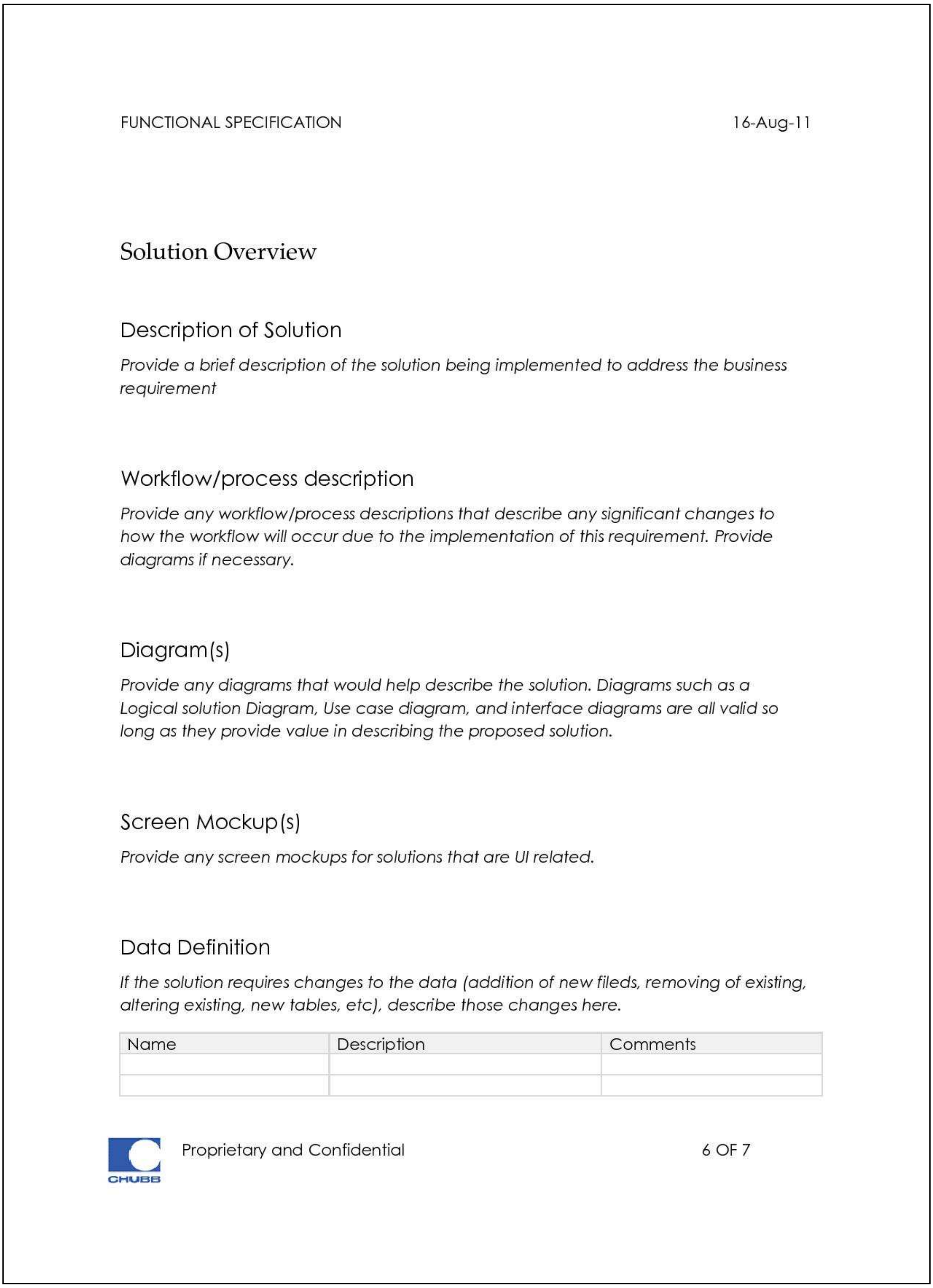
Business Rules

ID	Actions	Steps
Br-12		

Legal/Regulatory

Indicate if this particular business requirement has any legal or regulatory implications.

CHUBB Proprietary and Confidential 5 OF 7



FUNCTIONAL SPECIFICATION 16-Aug-11

Solution Overview

Description of Solution

Provide a brief description of the solution being implemented to address the business requirement

Workflow/process description

Provide any workflow/process descriptions that describe any significant changes to how the workflow will occur due to the implementation of this requirement. Provide diagrams if necessary.

Diagram(s)

Provide any diagrams that would help describe the solution. Diagrams such as a Logical solution Diagram, Use case diagram, and interface diagrams are all valid so long as they provide value in describing the proposed solution.

Screen Mockup(s)

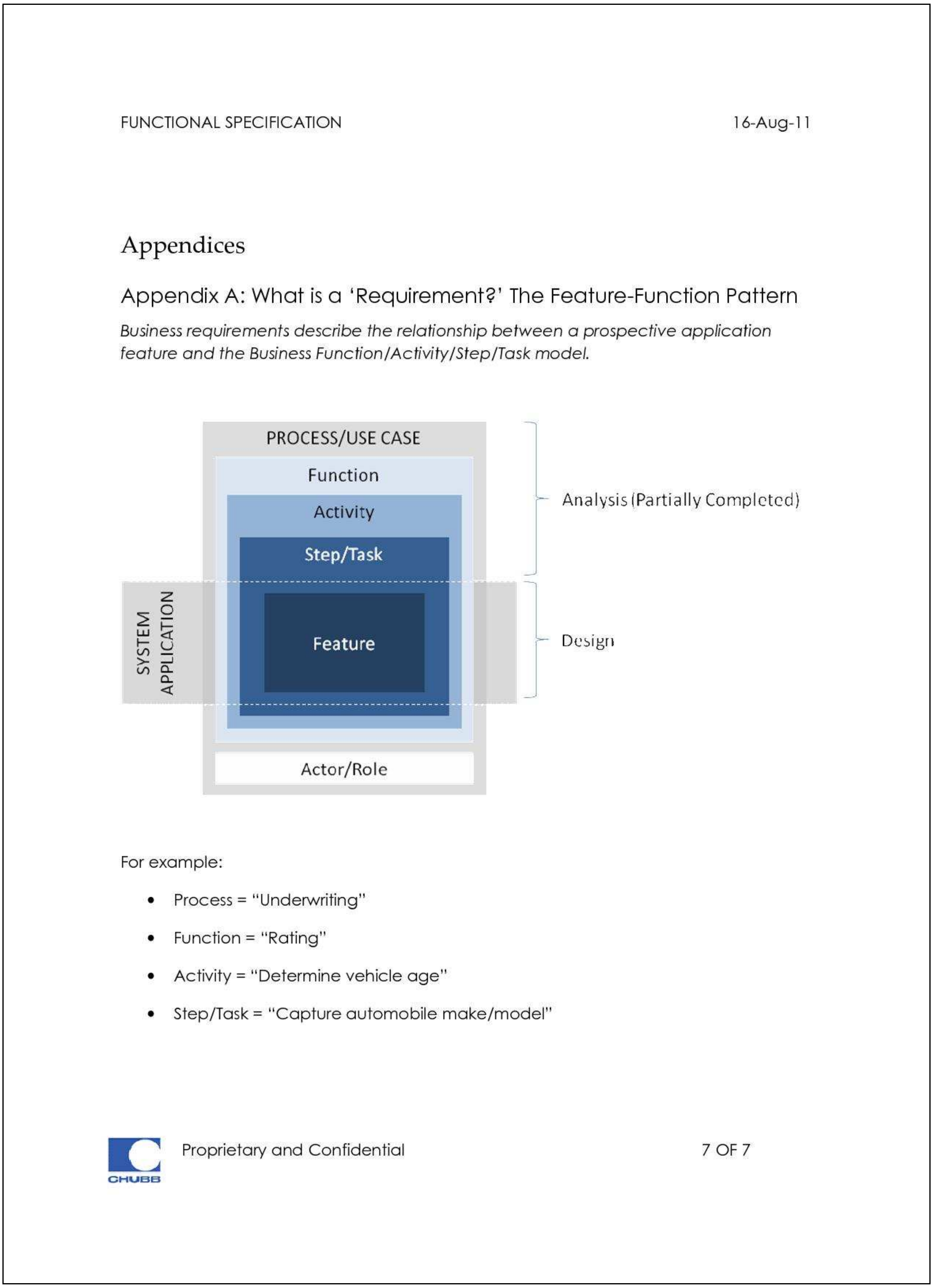
Provide any screen mockups for solutions that are UI related.

Data Definition

If the solution requires changes to the data (addition of new fileds, removing of existing, altering existing, new tables, etc), describe those changes here.

Name	Description	Comments

Proprietary and Confidential 6 OF 7



Appendices

Appendix A: What is a ‘Requirement?’ The Feature-Function Pattern

Business requirements describe the relationship between a prospective application feature and the Business Function/Activity/Step/Task model.

PROCESS/USE CASE
Function
Activity
Step/Task
Feature
SYSTEM
APPLICATION
Actor/Role
Analysis (Partially Completed)
Design

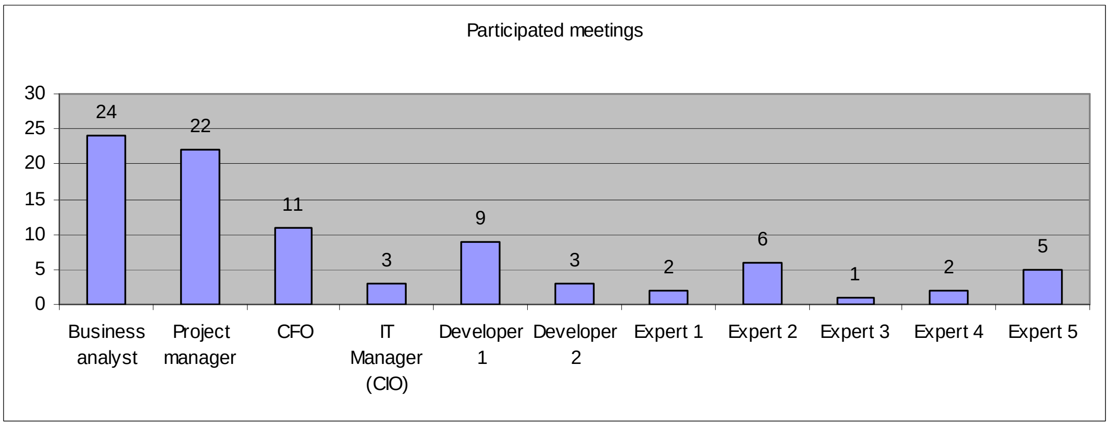
For example:

- Process = “Underwriting”
- Function = “Rating”
- Activity = “Determine vehicle age”
- Step/Task = “Capture automobile make/model”

Appendix O: Involvement of stakeholders

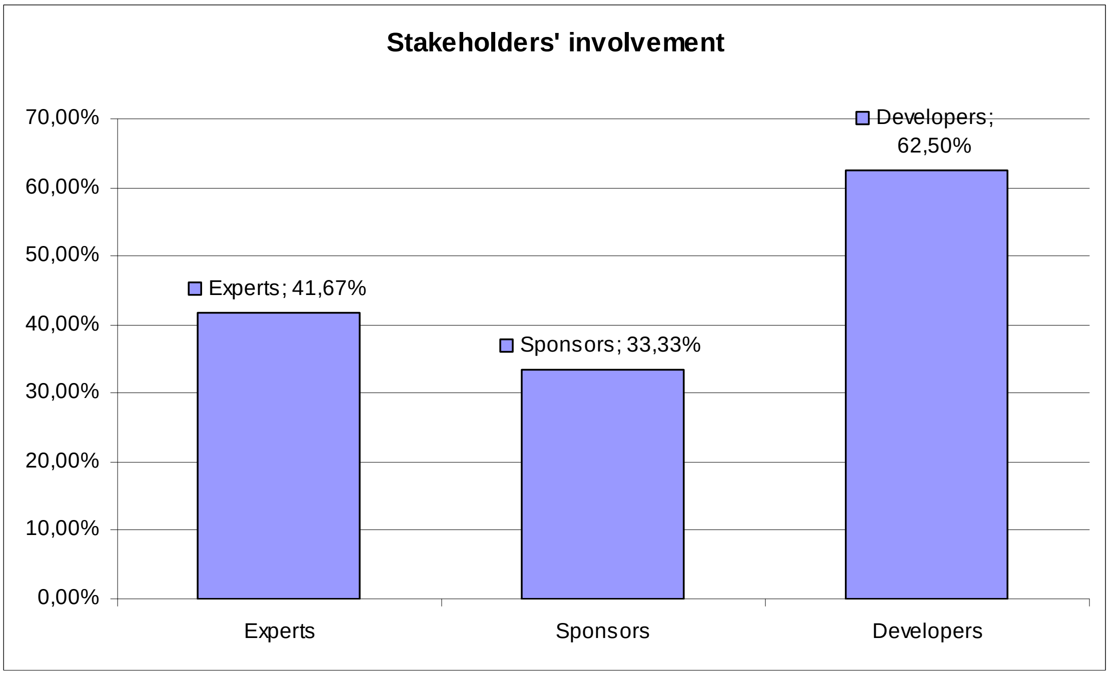
Participated meetings per stakeholder

Stakeholder	Participated meetings	Type
Business Analyst	24	Business Analyst
Project Manager	22	Project Manager
CFO	11	Sponsor
IT Manager (CIO)	3	Expert
Developer 1	9	Developer
Developer 2	3	Developer
Expert 1	2	Expert
Expert 2	6	Expert
Expert 3	1	Expert
Expert 4	2	Expert
Expert 5	5	Expert

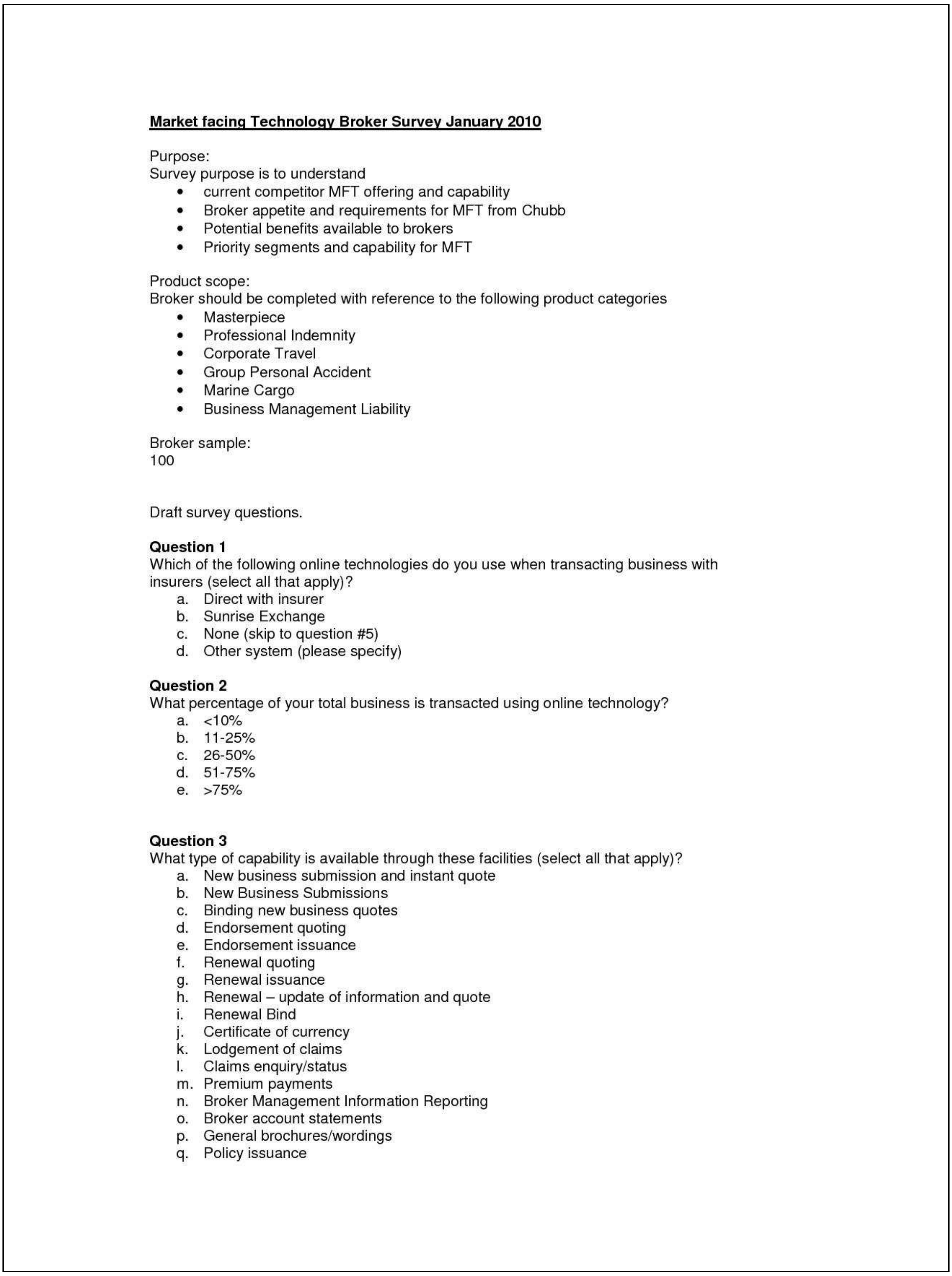


Involvement in the meetings per stakeholder type

Stakeholder type	Involvement in %	Amount of meetings involved in
Experts	41,67%	10 of 24
Sponsors	33,33%	8 of 24
Developers	62,50%	15 of 24



Appendix P: Marker research questionnaire



Market facing Technology Broker Survey January 2010

Purpose:
Survey purpose is to understand

- current competitor MFT offering and capability
- Broker appetite and requirements for MFT from Chubb
- Potential benefits available to brokers
- Priority segments and capability for MFT

Product scope:
Broker should be completed with reference to the following product categories

- Masterpiece
- Professional Indemnity
- Corporate Travel
- Group Personal Accident
- Marine Cargo
- Business Management Liability

Broker sample:
100

Draft survey questions.

Question 1
Which of the following online technologies do you use when transacting business with insurers (select all that apply)?

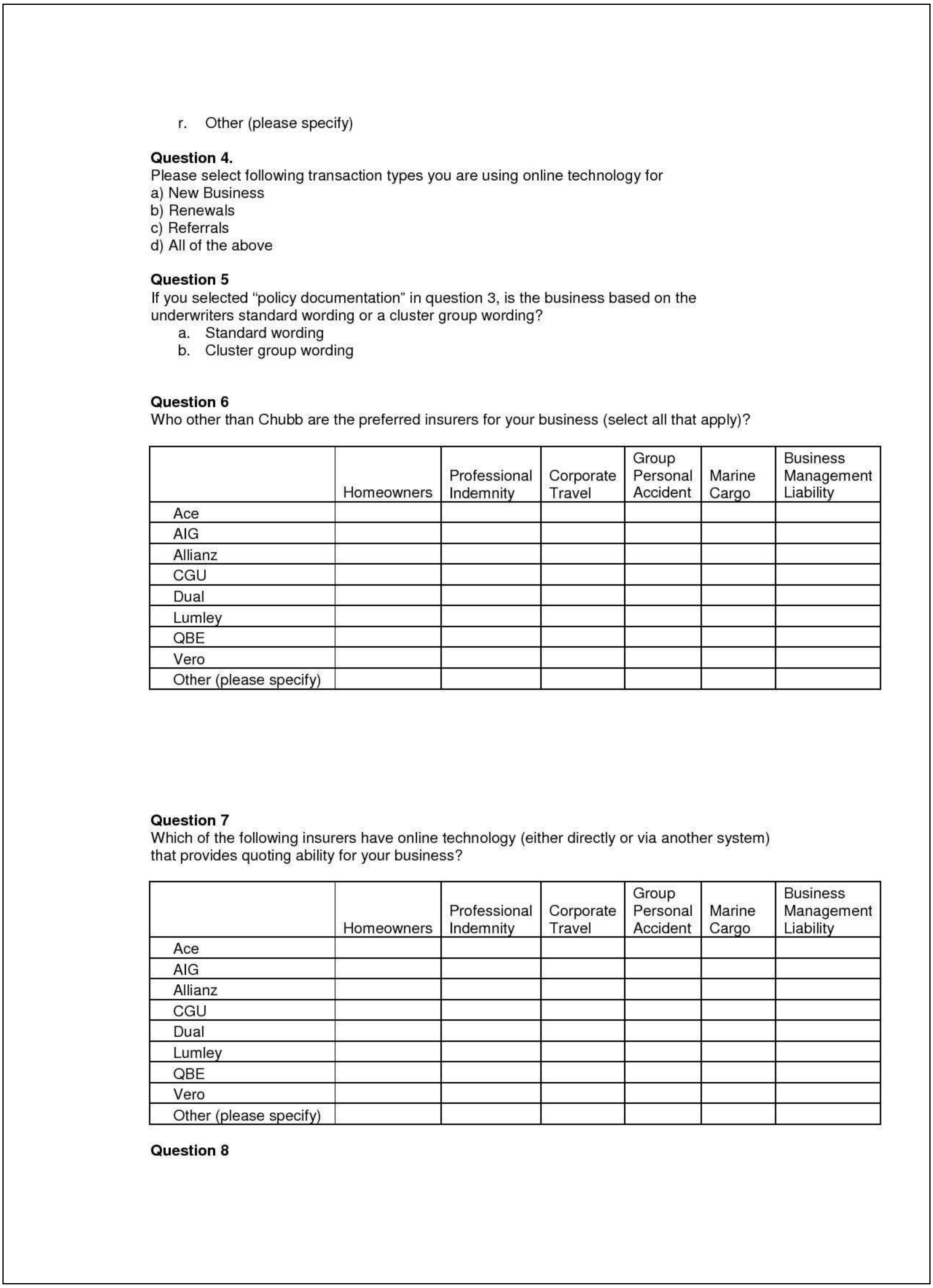
a. Direct with insurer
b. Sunrise Exchange
c. None (skip to question #5)
d. Other system (please specify)

Question 2
What percentage of your total business is transacted using online technology?

a. <10%
b. 11-25%
c. 26-50%
d. 51-75%
e. >75%

Question 3
What type of capability is available through these facilities (select all that apply)?

a. New business submission and instant quote
b. New Business Submissions
c. Binding new business quotes
d. Endorsement quoting
e. Endorsement issuance
f. Renewal quoting
g. Renewal issuance
h. Renewal – update of information and quote
i. Renewal Bind
j. Certificate of currency
k. Lodgement of claims
l. Claims enquiry/status
m. Premium payments
n. Broker Management Information Reporting
o. Broker account statements
p. General brochures/wordings
q. Policy issuance



r. Other (please specify)

Question 4.
Please select following transaction types you are using online technology for
a) New Business
b) Renewals
c) Referrals
d) All of the above

Question 5
If you selected “policy documentation” in question 3, is the business based on the underwriters standard wording or a cluster group wording?
a. Standard wording
b. Cluster group wording

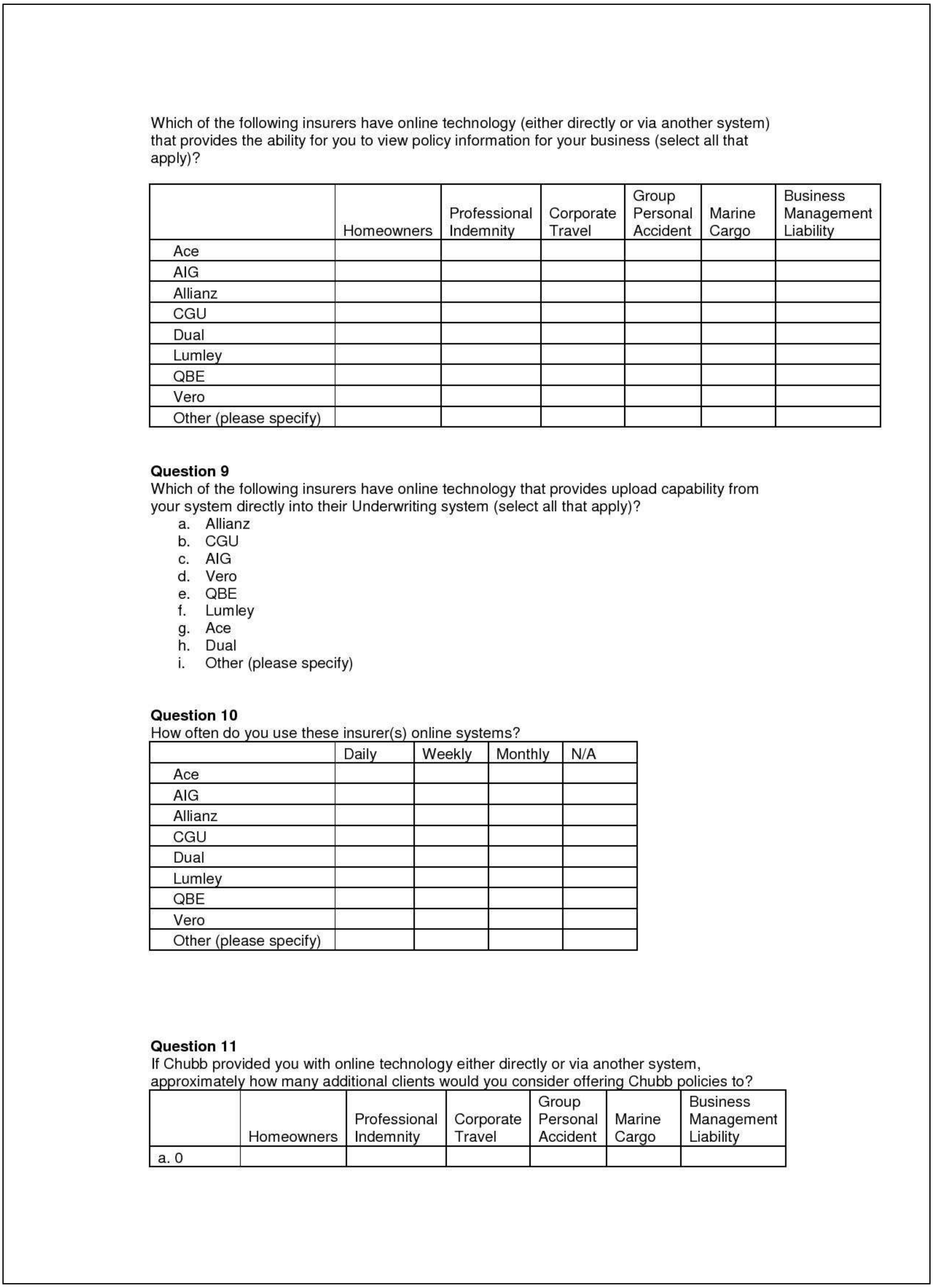
Question 6
Who other than Chubb are the preferred insurers for your business (select all that apply)?

	Homeowners	Professional Indemnity	Corporate Travel	Group Personal Accident	Marine Cargo	Business Management Liability
Ace						
AIG						
Allianz						
CGU						
Dual						
Lumley						
QBE						
Vero						
Other (please specify)						

Question 7
Which of the following insurers have online technology (either directly or via another system) that provides quoting ability for your business?

	Homeowners	Professional Indemnity	Corporate Travel	Group Personal Accident	Marine Cargo	Business Management Liability
Ace						
AIG						
Allianz						
CGU						
Dual						
Lumley						
QBE						
Vero						
Other (please specify)						

Question 8



Which of the following insurers have online technology (either directly or via another system) that provides the ability for you to view policy information for your business (select all that apply)?

	Homeowners	Professional Indemnity	Corporate Travel	Group Personal Accident	Marine Cargo	Business Management Liability
Ace						
AIG						
Allianz						
CGU						
Dual						
Lumley						
QBE						
Vero						
Other (please specify)						

Question 9

Which of the following insurers have online technology that provides upload capability from your system directly into their Underwriting system (select all that apply)?

a. Allianz
b. CGU
c. AIG
d. Vero
e. QBE
f. Lumley
g. Ace
h. Dual
i. Other (please specify)

Question 10

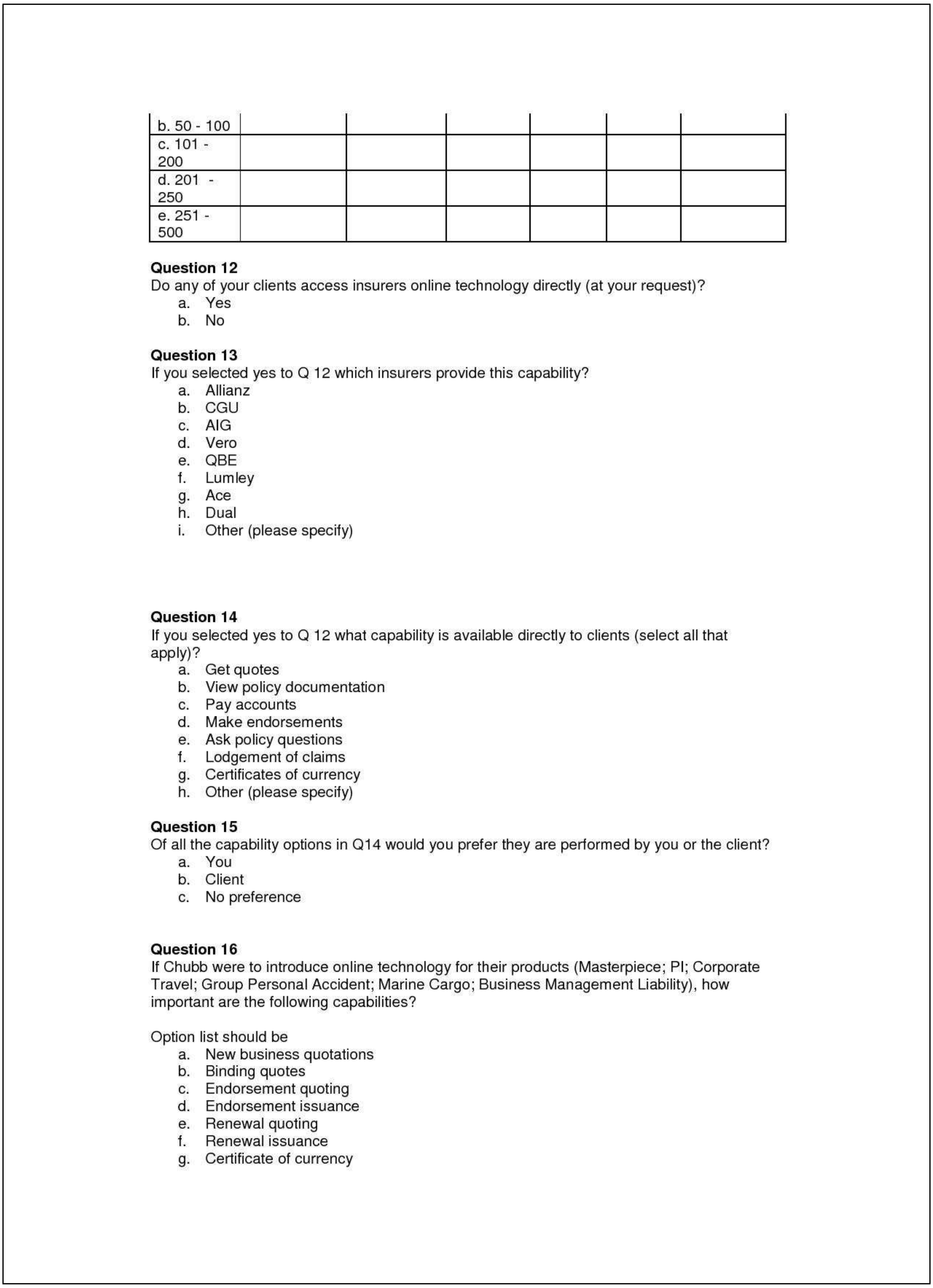
How often do you use these insurer(s) online systems?

	Daily	Weekly	Monthly	N/A
Ace				
AIG				
Allianz				
CGU				
Dual				
Lumley				
QBE				
Vero				
Other (please specify)				

Question 11

If Chubb provided you with online technology either directly or via another system, approximately how many additional clients would you consider offering Chubb policies to?

	Homeowners	Professional Indemnity	Corporate Travel	Group Personal Accident	Marine Cargo	Business Management Liability
a. 0						



b. 50 - 100						
c. 101 - 200						
d. 201 - 250						
e. 251 - 500						

Question 12
Do any of your clients access insurers online technology directly (at your request)?

a. Yes
b. No

Question 13
If you selected yes to Q 12 which insurers provide this capability?

a. Allianz
b. CGU
c. AIG
d. Vero
e. QBE
f. Lumley
g. Ace
h. Dual
i. Other (please specify)

Question 14
If you selected yes to Q 12 what capability is available directly to clients (select all that apply)?

a. Get quotes
b. View policy documentation
c. Pay accounts
d. Make endorsements
e. Ask policy questions
f. Lodgement of claims
g. Certificates of currency
h. Other (please specify)

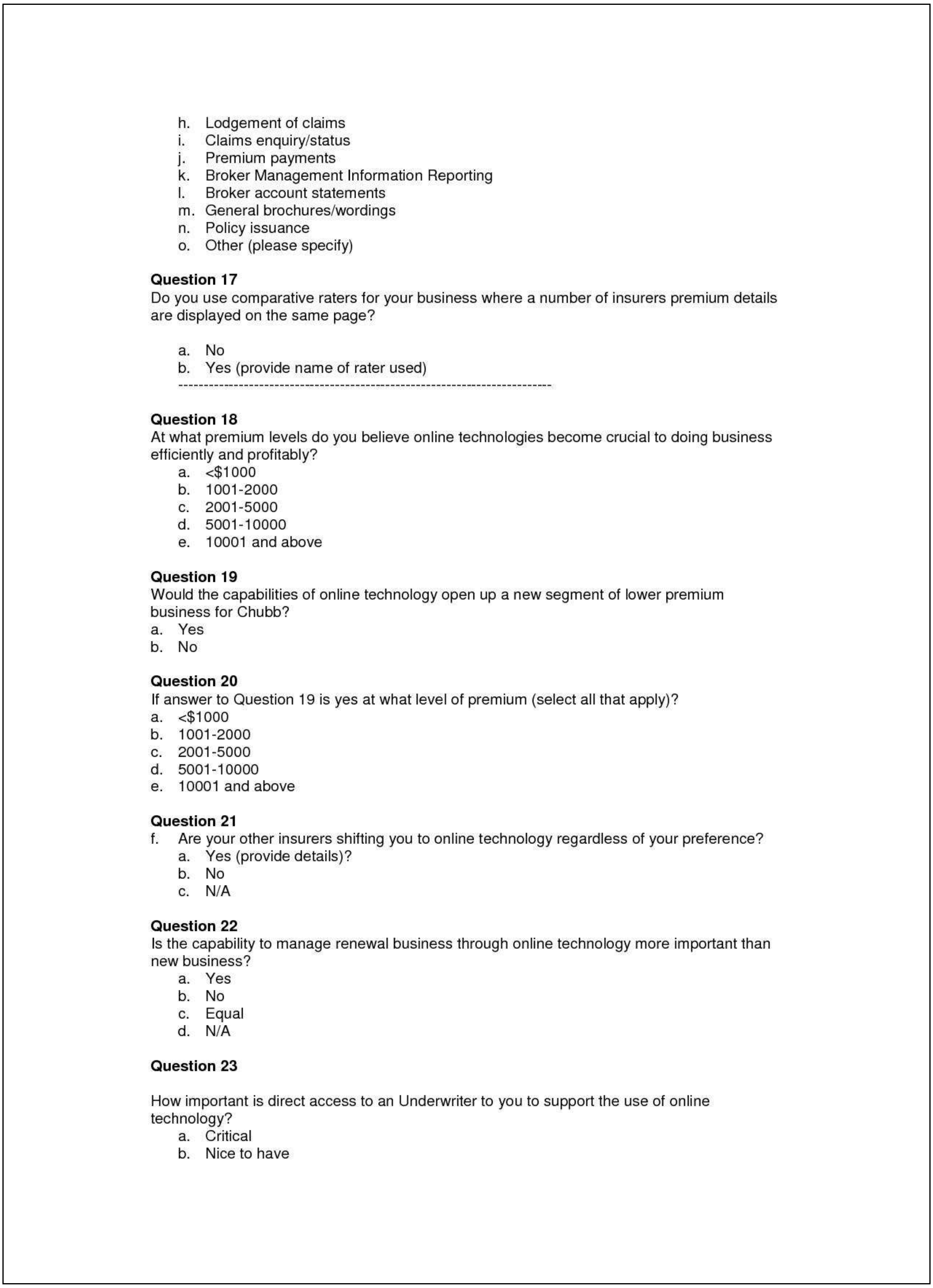
Question 15
Of all the capability options in Q14 would you prefer they are performed by you or the client?

a. You
b. Client
c. No preference

Question 16
If Chubb were to introduce online technology for their products (Masterpiece; PI; Corporate Travel; Group Personal Accident; Marine Cargo; Business Management Liability), how important are the following capabilities?

Option list should be

a. New business quotations
b. Binding quotes
c. Endorsement quoting
d. Endorsement issuance
e. Renewal quoting
f. Renewal issuance
g. Certificate of currency



h. Lodgement of claims
i. Claims enquiry/status
j. Premium payments
k. Broker Management Information Reporting
l. Broker account statements
m. General brochures/wordings
n. Policy issuance
o. Other (please specify)

Question 17
Do you use comparative raters for your business where a number of insurers premium details are displayed on the same page?

a. No
b. Yes (provide name of rater used)

Question 18
At what premium levels do you believe online technologies become crucial to doing business efficiently and profitably?

a. <$1000
b. 1001-2000
c. 2001-5000
d. 5001-10000
e. 10001 and above

Question 19
Would the capabilities of online technology open up a new segment of lower premium business for Chubb?

a. Yes
b. No

Question 20
If answer to Question 19 is yes at what level of premium (select all that apply)?

a. <$1000
b. 1001-2000
c. 2001-5000
d. 5001-10000
e. 10001 and above

Question 21
f. Are your other insurers shifting you to online technology regardless of your preference?
a. Yes (provide details)?
b. No
c. N/A

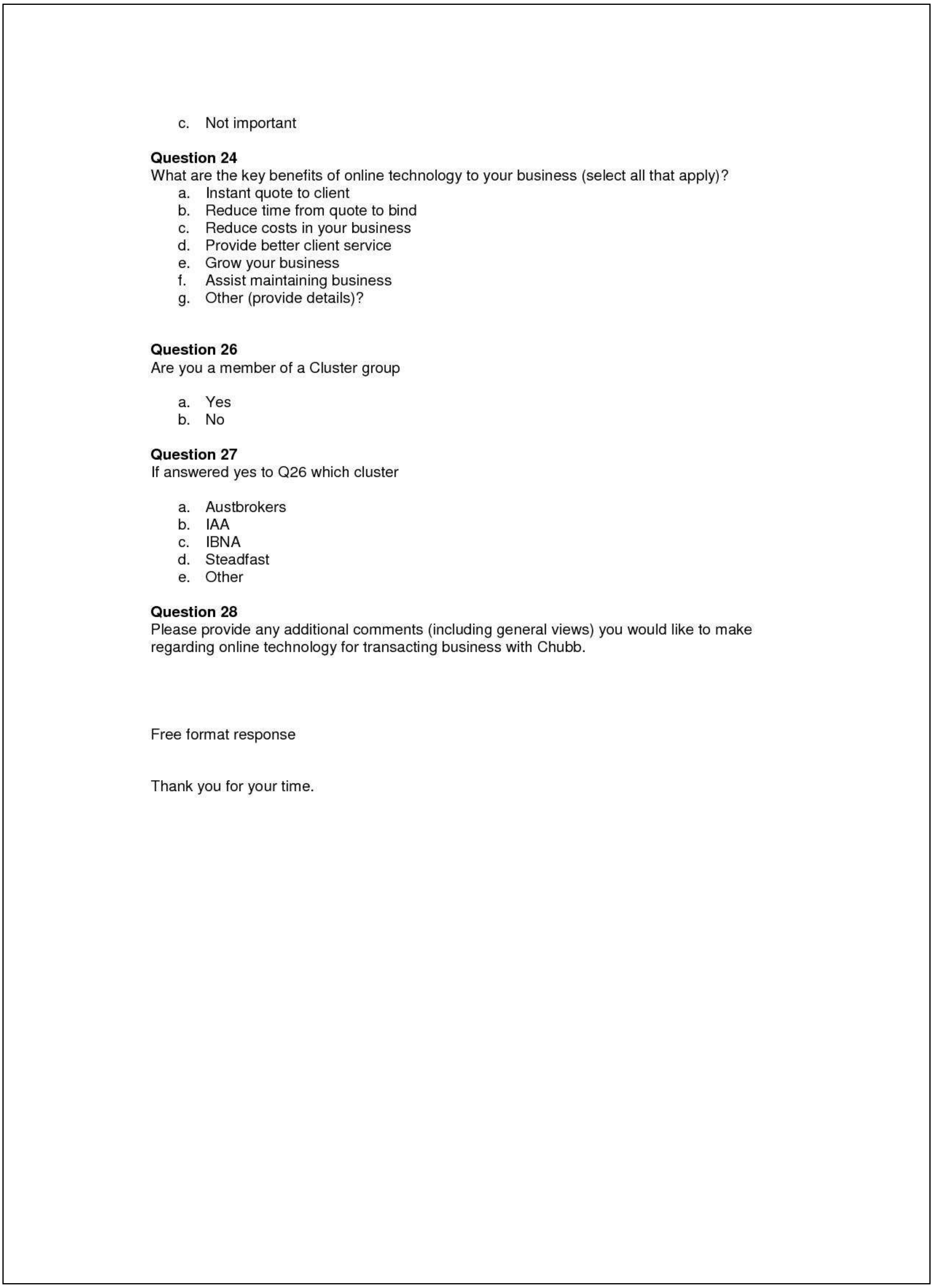
Question 22
Is the capability to manage renewal business through online technology more important than new business?

a. Yes
b. No
c. Equal
d. N/A

Question 23

How important is direct access to an Underwriter to you to support the use of online technology?

a. Critical
b. Nice to have



c. Not important

Question 24
What are the key benefits of online technology to your business (select all that apply)?

a. Instant quote to client
b. Reduce time from quote to bind
c. Reduce costs in your business
d. Provide better client service
e. Grow your business
f. Assist maintaining business
g. Other (provide details)?

Question 26
Are you a member of a Cluster group

a. Yes
b. No

Question 27
If answered yes to Q26 which cluster

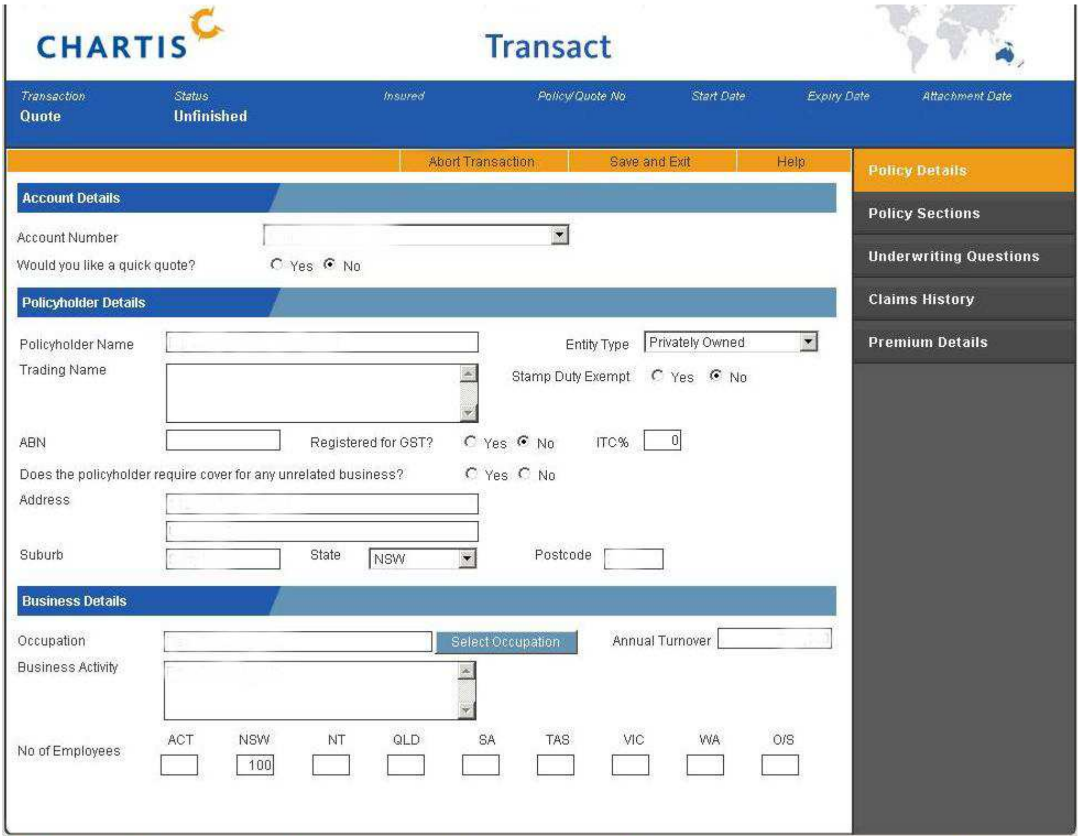
a. Austbrokers
b. IAA
c. IBNA
d. Steadfast
e. Other

Question 28
Please provide any additional comments (including general views) you would like to make regarding online technology for transacting business with Chubb.

Free format response

Thank you for your time.

Appendix Q: Screenshots of competitor's systems



Transaction	Status	Insured	Policy/Quote No	Start Date	Expiry Date	Attachment Date
Quote	Unfinished					

Account Details

Account Number

Would you like a quick quote? ○ Yes ● No

Policyholder Details

Policyholder Name

Entity Type Privately Owned

Trading Name

Stamp Duty Exempt ○ Yes ● No

ABN

Registered for GST? ○ Yes ● No

ITC% 0

Does the policyholder require cover for any unrelated business? ○ Yes ○ No

Address

Suburb

State NSW

Postcode

Business Details

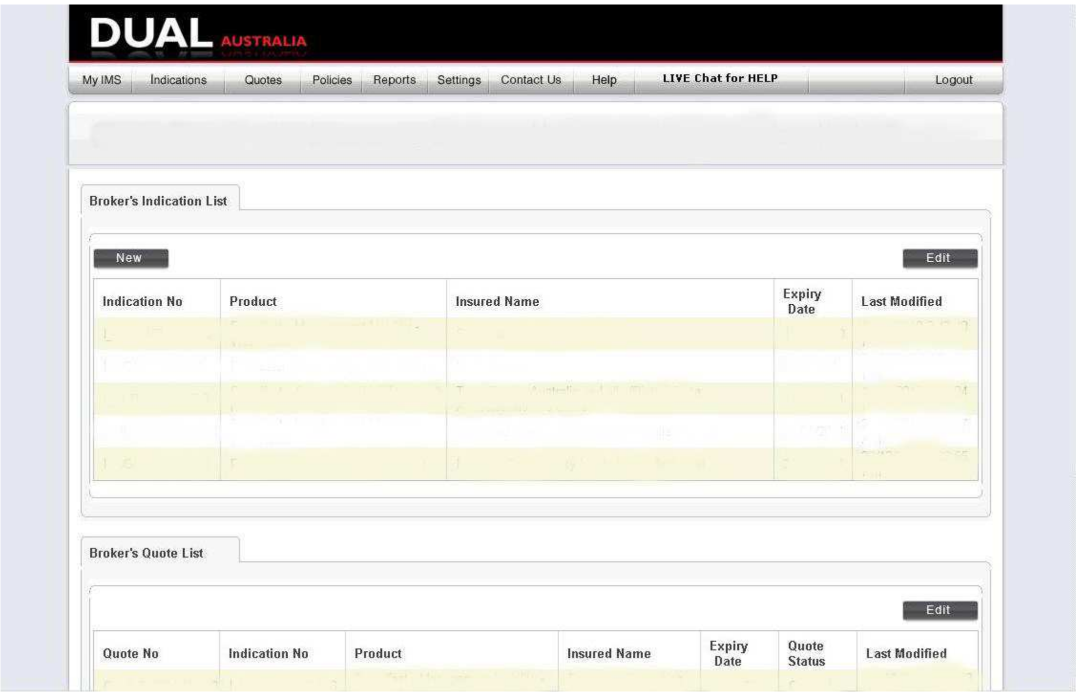
Occupation

Annual Turnover

Business Activity

No of Employees

ACT	NSW	NT	QLD	SA	TAS	VIC	WA	O/S
	100							



Broker's Indication List

Indication No	Product	Insured Name	Expiry Date	Last Modified
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]

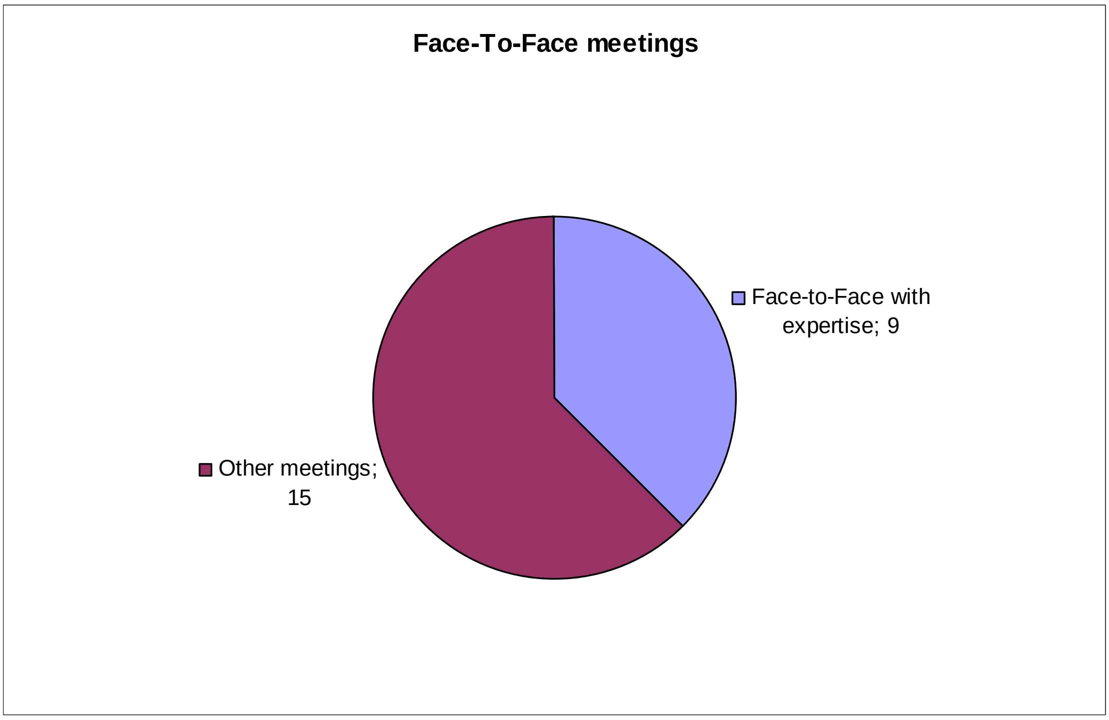
Broker's Quote List

Quote No	Indication No	Product	Insured Name	Expiry Date	Quote Status	Last Modified
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]

Appendix R: Observations on face-to-face meetings

Meetings

Amount of meetings	Face-to-Face with expertise	Other meetings
24	9	15



Appendix S: Meetings notes of face-to-face meetings with experts

Notes meeting with National Product Manager

Date: 05-04-2011
Time: 09:30 – 10:00

Wordings are not changing in the structure. Purpose is to make it simpler.

The wordings are to be changed, however it can probably take a long time when these wordings will be approved. This depends on if the wordings have to be send through the US or not. If not, the earliest is probably the end of May. The National Product Manager will have the draft of the wordings in a few weeks, and then she/he can/will send it to the Project Manager.

The Project Manager will send the National Product Manager some information about the iClose project.

Notes on meeting with National OSD Manager

Date: 15-02-2011
Time: 14:00 – 15:00

This meeting is about how the overall process works.

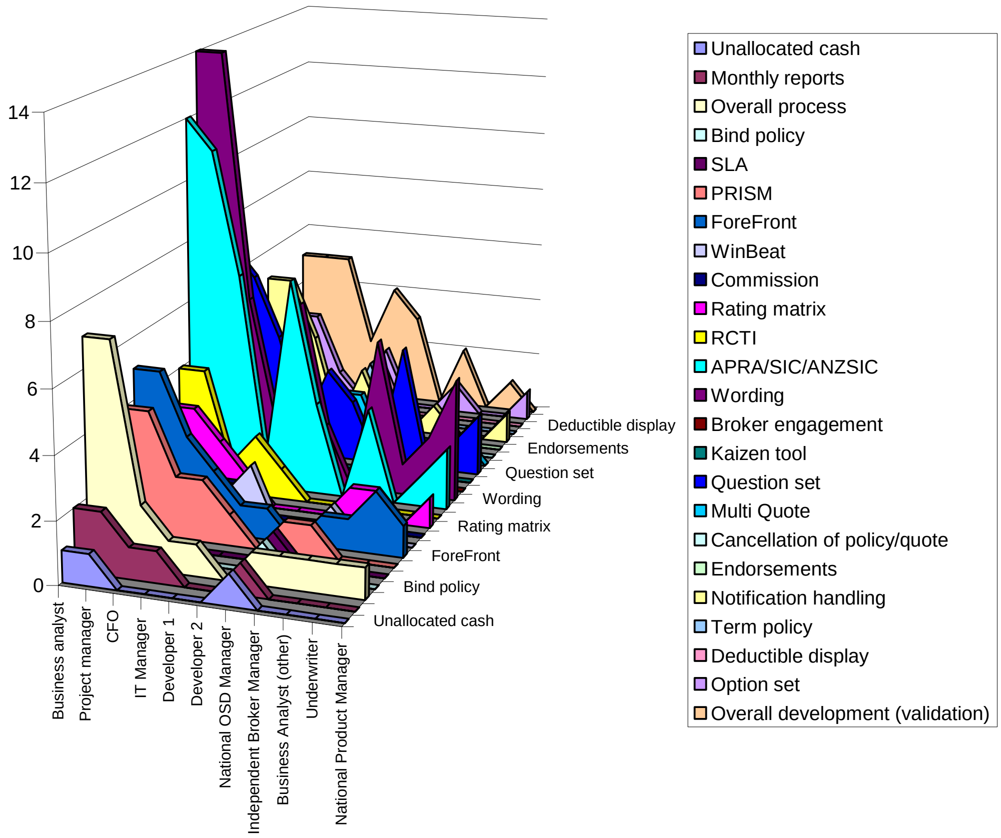
Original process
Register -> bind -> book into prism

Process
1. Iclose sends PDF (policy doc) to xxxxxx@chubb.comW
2. iClose sends data to CICA
3. (OSD manually) Sheet + pdf file on L drive
4. XML comes in check off + (register, bind, book)
    - Sheet updated
    - XML file stored on L drive

Issues
1) We need an ongoing reconciliation process of premiums being written & actual booked
    - Monthly reports of bindings by volume, premium, cover and commission
    - (OSD) Need some kind of a SLA (number of days for booking)
    - Every 15th of the month?
2) Unallocated premium cash -> How to analyze (back track) this with iClose?
3) Policies are binded and entered by pre-assigned numbers in prism. For example number 645 are the policies x,y,z...
4) Right now, this is convenient to use. Will this remain the same with iClose?

Appendix T: Number of iterations per stakeholder per requirement concern

	Business Analyst	Project Manager	CFO	IT Manager	Developer 1	Developer 2	National OSD Manager	Independent Broker Manager	Business Analyst (other)	Underwriter	National Product Manager	**Total**
Unallocated cash	1	1					1					2
Monthly reports	2	2	1	1			1					5
Overall process	7	7	2	1	1		1	1	1	1	1	16
Bind policy	1	1					1					2
SLA	1	1					1					2
PRISM	4	4	2	2	1		1	1				11
ForeFront	5	5	3	2	1	1		1	1	2	1	17
WinBeat	3	3	2	1	2			1				9
Commission	1	1	1	1								3
Rating matrix	3	3	2	1				1	1		1	9
RCTI	4	4	1	2	1							8
APRA/SIC/ANZSIC	12	11	7	1	7	3		3		1	2	35
Wording	14	14	6	2	6	2		5	1	2	4	42
Broker engagement	1	1	1		1			1				4
Kaizen tool	1	1							1			2
Question set	7	6	4	1	3	2		4		1	2	23
Multi Quote	4	4	3	1	2	2		1		1		14
Cancellation of policy/quote	1	1	1									2
Endorsements	1	1	1									2
Notification handling	5	5	3		2			1			1	12
Term policy	2	2			2							6
Deductible display	2	2			2							6
Option set	3	3	1		2			1			1	11
Overall development (validation)	5	5	5	2	4	3		2		1		22
Total	90	88	46	18	37	13	6	23	5	9	13	


14
12
10
8
6
4
2
0
Business analyst
Project manager
CFO
IT Manager
Developer 1
Developer 2
National OSD Manager
Independent Broker Manager
Business Analyst (other)
Underwriter
National Product Manager
Deductible display
Endorsements
Question set
Wording
Rating matrix
ForeFront
Bind policy
Unallocated cash
Unallocated cash
Monthly reports
Overall process
Bind policy
SLA
PRISM
ForeFront
WinBeat
Commission
Rating matrix
RCTI
APRA/SIC/ANZSIC
Wording
Broker engagement
Kaizen tool
Question set
Multi Quote
Cancellation of policy/quote
Endorsements
Notification handling
Term policy
Deductible display
Option set
Overall development (validation)

Appendix U: Observations No. 13 and 24

Direct observation No. 13 – Meeting about question set with the Independent Broker Manager

Location:	Chubb – Office Independent Broker Manager	
Report date:	16-05-2011	
Observation period:	From: 2011- 05 – 12 10:00	To: 2011- 05 – 12 10:30
Project	iClose – questions set discussion	
Participants:	Business Analyst Independent Broker Manager	

14. Are decision documented?
    Yes

15. Are requirements requests from stakeholders documented?
    Yes

16. Do stakeholders use templates to formulate their requirements?
    No

17. Do stakeholders use a collaboration system?
    No

18. Do stakeholders use cognitive tools to generate an overall understanding of the requirements and the system?
    No

19. Is the group of stakeholders diverse?
    No, only business people no developers

20. Does the requirements engineer use prioritization mechanisms?
    No

21. Does the requirements engineer use automated validation mechanisms?
    No

22. Are the stakeholders consistent in explaining and defining their requirements?
    Yes, based on the question set of the requirements document

23. Doe the requirements engineer use interviews to gather the requirements from the stakeholders?
    No

24. Is the requirements engineering done iteratively?
    Yes

25. Number of stakeholders: 2

26. Number of requirement concerns: 1

Requirement concerns:

- Questions

To what extend do stakeholders participate during the requirements engineering process?
Not at all A great
deal
1 2 3 **4** 5

Actions that the requirements engineer took to solve conflicting demands:

Other observations:
Meeting with Independent Broker Manager to discuss the questions that should be asked to the broker for generating a quota for their customers. Before the meeting the Project Manager suggested in the last meeting that the system should be simple and easy to use. Therefore the objective of this meeting is to reduce and review the question set. The Independent Broker Manager determined which questions should be in and which ones should be removed. He mentioned that several questions can be formulated as one question, other questions could be removed and some should be moved to another section. All these suggestions were documented by the Business Analyst. After the meeting these suggestions were discussed with the Project Manager and added to the requirements document. The Business Analyst then modified the questions and communicated this back to Independent Broker Manager and the Project Manager. The Independent Broker Manager confirmed this and made some last minor changes. The Business Analyst sent the questions set to the Project Manager. During the next meeting the questions set was addressed. The Business Analyst mentioned that question 29/30 can be removed if the APRA / SIC mapping is done correctly. The Project Manager confirmed that question 29 & 30 can be removed and the developer confirmed that the final question set will be implemented.

Direct observation No. 24 – iClose outstanding issues

Location:	Chubb Sydney – Meeting room 4	
Report date:	30-06-2011	
Observation period:	From: 2011- 06 – 30 14:00	To: 2011- 06 – 30 15:30
Project	iClose – Outstanding issues	
Participant:	Business Analyst	
	CFO	
	Project Manager	
	National Product Manager	
	Developer	

1. **Are decision documented?**
   Yes

2. **Are requirements requests from stakeholders documented?**
   Yes

3. **Do stakeholders use templates to formulate their requirements?**
   No

4. **Do stakeholders use a collaboration system?**
   No

5. **Do stakeholders use cognitive tools to generate an overall understanding of the requirements and the system?**
   No

6. **Is the group of stakeholders diverse?**
   Yes

7. **Does the requirements engineer use prioritization mechanisms?**
   Yes, time and budget

8. **Does the requirements engineer use validation mechanisms?**
   Yes, input from experts

9. **Are the stakeholders consistent in explaining and defining their requirements?**
   No

10. **Doe the requirements engineer use interviews to gather the requirements from the stakeholders?**
    No

11. **Is the requirements engineering done iteratively?**
    Yes

12. Number of stakeholders: 5

13. Number of requirement concerns: 4

Requirement concerns:

- SIC/ANSIC (occupation categories)
- Question set

- Wording
- Referral / declined

To what extend do stakeholders participate during the requirements engineering process?

Not at all				A great deal
1	2	3	**4**	5

Actions that the requirements engineer took to solve conflicting demands:
Wrote it down for further investigation, for example the PI coverage if it should be included or not. The RE will get back on this.

Actions that the requirements engineer took to gather the requirements:
Ask experts and developer about what need to be in the system

Actions that the requirements engineer took to prioritization and validate the requirements:
Validated with other experts to understand and see if it is actually necessary to build into the system. The developer gave his opinions if it will take extra time.

Actions of the stakeholders during the negotiation process of the requirements:
Stakeholders gave their opinions about the decisions of the RE. They said if it should be included or not.

Other observations:
During the meeting several actions were discussed, these are;

- Question set: Q5 in the BS document should be included or not. National Product Manager will try to find it out
- Wording: Original wording should be used. This need to be put into the document. There is also a template for the exclusions. This need to be in the BS document as well.
- Appendix B in the BS document is an old version. This should be replaced with the most recent one. National Product Manager will provide the Business Analyst the most recent version of the template.

Appendix V: Scope of iClose project

MFT Forefront Pilot Scope

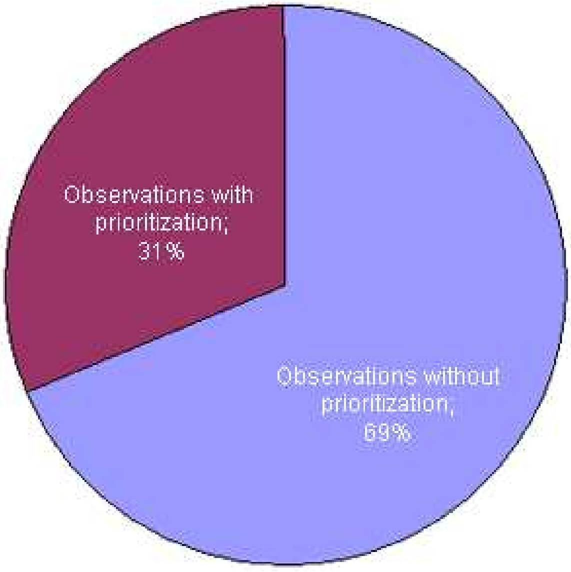
In scope	Outside scope
New business process build in iClose (Rate/Quote/Bind/Issue) -New Business process should cater for renewals -Need to provide policy number on issue through iClose	Mid term adjustments in iClose
Manual referral process	Referral functionality in iClose
Forefront	All other products
Analyse effort to implement xml loader	Implement xml loader
Update NUC processes to enter business into Prism	
Standard iClose MIS reporting	Changes to Chubb MIS system
Marketing collateral for Forefront	
Compliance review of policy wording	
Development of training, support materials for implementation	

Appendix W: Prioritization during meetings

Prioritization during meetings

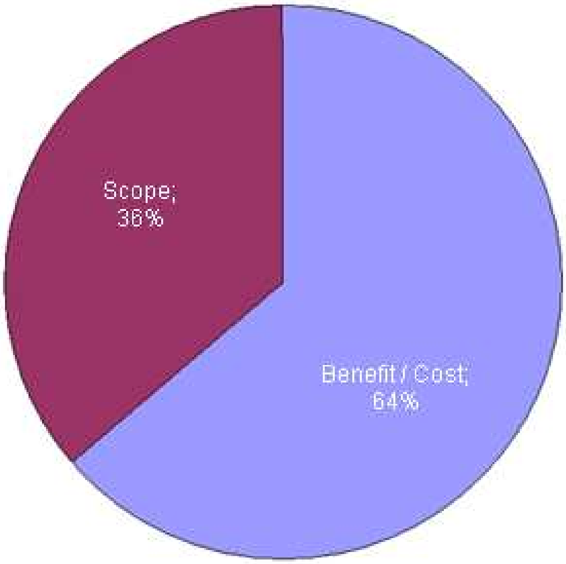
Meeting No	Prioritization
2	Benefit/Cost
5	Benefit/Cost
11	Benefit/Cost
14	Benefit/Cost
15	Benefit/Cost
18	Benefit/Cost
19	Benefit/Cost
22	Scope
7	Scope
12	Scope
24	Scope

Amount of observations	Amount of prioritization
24	11



Main prioritization mechanisms

Benefit/Cost	Totaal
Benefit/Cost Scope	7 4
Eindtotaal	11


Scope; 36%
Benefit / Cost; 64%