UNIVERSITY OF TARTU
Institute of Computer Science
Conversion Master in IT

**Mari Vaiksaar**

# Requirements Management in Off-The-Shelf Software Implementation Projects: A Case Study in Playtech

**Master's Thesis (15 ECTS)**

Supervisor: Fredrik Payman Milani

Tartu 2019

## Requirements Management in Off-The-Shelf Software Implementation Projects: A Case Study in Playtech

**Abstract:**

Requirements management is considered a core competency for delivering quality software solutions. It is also counted among the main causes for project failure. This is true in the context of greenfield development as well as off-the-shelf (OTS) based software solutions. The challenge in OTS software implementation projects today from the software provider's perspective is ensuring successful completion of the solution setup that meets customer needs and satisfies requirements without compromising on delivery time and cost. Today, there is no requirements management approach that would consider the specifics of OTS based software implementation projects from the supplier's perspective and the particularities of the online gambling industry. This thesis addresses the lack of systematic approach to requirements management in OTS based online gambling software context in case company Playtech and attempts to answer the research question of how requirements can be managed when implementing OTS based online gambling solutions. Based on analysis and best practices from background research, a process is suggested for efficient requirements management in OTS gambling software implementation projects. The proposed process incorporates activities that are not present or are present only partially in the current practices, such as needs assessment, requirements management planning, requirements monitoring and controlling, reporting lessons learned and support transition. Alignment between all stakeholders as well as management is required to enable successful establishment of the requirements management process.

**Keywords:**

Requirements management, off-the-shelf software, online gambling, Playtech

**CERCS:** P170 Computer science, numerical analysis, systems, control

## Nõuete haldus valmistarkvara rakendusprojektides: Playtechi juhtum

**Lühikokkuvõte:**

Nõuete haldust peetakse kvaliteetsete tarkvaralahenduste pakkumisel üheks põhipädevuseks. Samas on see ka üks peamisi tarkvaraprojektide ebaõnnestumise põhjuseid. Mõlemad väited kehtivad nii rätsepalahenduste kui valmistarkvara-põhiste lahenduste korral. Tarkvara pakkujate jaoks on väljakutse tagada edukas valmistarkvara rakendusprojektide elluviimine nii, et kõik kliendi vajadused ja nõuded saaksid rahuldatud ja et aja- ja ressursikulu oleks sealjuures võimalikult väike. Tänapäeval ei leidu nõuete haldusele ühtset lähenemist, mis oleks kohandatud valmistarkvara-põhistele rakendusprojektidele tarkvara pakkuja vaatest ning ühtlasi arvestaks interneti hasartmängutööstuse eripäradega. Magistritöö eesmärgiks on täita see tühimik, tuginedes juhtumiuuringule internetipõhist hasartmängutarkvara tootvas ettevõttes Playtech, ning leida vastus küsimusele, kuidas saaks interneti hasartmänguvaldkonnas valmistarkvara-põhistes rakendusprojektides nõudeid hallata. Tuginedes analüüsile ja teaduspõhistele parimatele praktikatele nõuete halduse valdkonnas, pakutakse magistritöös välja protsess nõuete efektiivseks haldamiseks hasartmängu-valmistarkvara rakendusprojektides. Nimetatud protsess hõlmab tegevusi, mis praeguses praktikas puudu või olemas vaid osaliselt, nagu näiteks vajaduste hindamine, nõuete halduse planeerimine, nõuete seire ja kontrollimine, süsteemne vigadest õppimine ja projekti tugivastutuse üleandmine.

Nõuete halduse protsessi eduka juurutamise eelduseks on joondumine kõikide sidusgruppide ning juhtkonnaga.

**Võtmesõnad:**

Nõuete haldus, valmistarkvara, interneti hasartmängud, Playtech

**CERCS:** P170 Arvutiteadus, arvanalüüs, süsteemid, juhtimine (automaatjuhtimisteooria)

# Table of Contents


1 Introduction ..... 5
1.1 Research question ..... 6
1.2 Research design ..... 6
2 Background ..... 8
2.1 Core concepts of requirements management ..... 8
2.2 Issues, best practices and research directions in requirements management ..... 11
2.3 OTS-based solutions ..... 14
2.4 Gap identification ..... 16
2.5 Gambling industry, software implementation projects and Playtech ..... 16
3 Case study ..... 19
3.1 Setting ..... 19
3.2 Design ..... 20
3.3 Results ..... 21
3.3.1 Requirements life-cycle ..... 22
3.3.2 Requirements documentation and management ..... 24
3.3.3 Assessment of current requirements management practices ..... 26
4 Discussion ..... 29
4.1 Lack of systematic approach ..... 29
4.2 Documentation ..... 30
4.3 Information management ..... 30
4.4 Proposed requirements management process ..... 30
4.5 Threats to validity ..... 35
5 Conclusion ..... 37
6 References ..... 39
Appendices ..... 41
I Interview questions ..... 41
II Licence ..... 42

# 1 Introduction

Fine requirements are the foundation of quality software solutions. Software-based innovations, increasing complexity, pressure to reduce costs, increasing time pressure and growing quality demands – these aspects describe the challenges of software development today [1]. In this context, the quality of requirements and the efficiency in their management lays the foundation for delivering value to the customer through IT solutions and thereby satisfying the customer needs. Practicing high quality requirements management however tends to be complicated and difficult to achieve. Already back in 1994, The Standish Group revealed in their CHAOS Report that the top reason for project failure was related to incomplete requirements, accounting for 13.1 percent of all causes for project failure mentioned, while causes relating to incomplete or changing requirements accounted for roughly a quarter of reasons for challenges experienced in projects [2]. Two decades later, while the complexity of technological solutions has indisputably increased, the situation has not improved in terms of statistics, even the opposite: in 2014, 47 percent of the reasons for missing original project goals and business objectives were related to poor requirements management [3].

Requirements management (RM) is not only relevant in the context of greenfield development but has also been recognized as a critical part of off-the-shelf (OTS) based software solutions [4]. OTS applications and components are becoming increasingly widespread as parts of larger systems. OTS component is essentially a standard component that is integrated into a broader system and functions as part of it. In the context of this thesis, it is important to differentiate between the commonly used term *commercial off-the-shelf (COTS)* and the less usual *off-the-shelf*. These terms are the opposite sides of the same coin. While COTS is generally viewed from customer perspective and considered a software product supplied by a vendor [5], in this paper the viewpoint of the software provider is taken and the standard components used by the vendor to deliver solutions are being referred to as OTS. The usage of OTS components reduces the development effort in solution implementations from vendor perspective and significantly speeds up the software order-to-delivery process.

Considering the vast amount of different types of information technology related solutions and projects, in the context of this thesis, and for lack of a better term, the focus will be on OTS software implementation projects. While *OTS software implementation project* does not exist as an established term in the available literature, in this paper it is defined as a project, run by the vendor company, that consists of implementing standard components that require configuration and customization for a certain customer, as well as development of specific functionalities and adaptations to meet customer and regulatory requirements.

One of the keys to success in the OTS software implementation projects is the effective management of requirements. There is ample literature available on requirements management approaches that are applicable to all sorts of software solutions. Being widely applicable however is due to these approaches being very generic. This, in turn, makes them less pertinent for specific industries and context. There have been initiatives to accommodate in the RM process the particularities of certain domains, such as automotive development [6], software product line engineering [7], and COTS systems like ERP-solutions [8] and space technology development [9]. All these domains have certain characteristics that need to be considered in the RM process. Nothing of this kind can be found in the available literature that would relate to the vendor point of view of OTS implementation or the online gambling industry specifically.

The gambling industry is subject to strict rules relating to legislative, regulatory, anti-money laundering, risk and societal aspects. It experiences changes at high pace due to rapidly

developing industry. As the end-customer actions in gambling essentially include monetary transactions, the domain could relatively easily be compared to the finance industry and therefore the difficulties in delivering online gambling solutions can also be similarly appreciated. Online gambling software providers generally have a refined, off-the-shelf portfolio of products and services that has been developed in line with the strict rules of the industry. Delivering components from the existing portfolio to business-to-business customers through complex software implementation projects however requires configuration changes, customizations, integration and oftentimes also custom development due to customer-specific needs and regulatory requirements.

The challenge in the OTS software implementation projects today is ensuring successful completion of the solution setups that meets the customer's needs and satisfies the requirements without compromising on delivery time and cost. Establishing systematic RM practices within these projects is indisputably one of the most solid ways to address that challenge. It is evident from the available literature that as of today, there is no RM approach that would consider the specifics of the online gambling industry and this type of software implementation projects. Seemingly, no research has been conducted on the implementation of online gambling solutions, however this might be due to competition and confidentiality restrictions in the industry that prohibit publishing such information. This paper attempts to fill that void.

## 1.1 Research question

This thesis intends to address the lack of systematic approach to RM in the online gambling software context. It seeks to answer the following question: **how can requirements be managed when implementing off-the-shelf based online gambling solutions?** It aims to suggest a process for efficient RM in software implementation projects while taking into consideration the specificities of the gambling industry. Since RM in software implementation projects is always a cross-team initiative, it is important to ensure that the suggested process is role-independent to increase its resistance to potential structural changes in the company as well as expand the variety of stakeholders who could benefit from and contribute to the process.

The main contribution of this thesis is a process according to which requirements in OTS-based online gambling solutions should be managed. This process should allow employees involved in the implementation projects approach the RM in a systematic and clear way, enable them to have a better overview of the status of requirements at any point in time, enhance organizational learning and cooperation, and ensure the continuous success in delivering complex online gambling software solutions.

## 1.2 Research design

In order to answer the proposed research question and suggest a systematic approach to RM in OTS-based online gambling solution implementation projects, a case study is carried out in Playtech, a world-renowned online gambling software company. The status quo of RM practices in OTS software implementation projects in the organization is examined. Current practices in RM are mapped, problematic areas and gaps are identified. By deducing from existing practices what works well, considering the particularities of the online gambling industry and addressing the shortcomings with RM best practices, a process for efficient RM is determined that is applicable for OTS implementation projects in Playtech. This process could be of significance and reference for other organizations in the industry that offer structurally similar solutions.

The thesis is divided into five core sections. In accordance with the numbering, the first section is the introduction. The second section provides an overview of the core concepts and processes in RM domain and OTS-based solutions. It gives an understanding of the particularities of the gambling domain and Playtech in the context of OTS software implementation projects. The third section consists of the case study description, execution and results. The fourth section discusses the findings and suggests a RM process for OTS-based solution implementation projects. This section is concluded with threats to validity. The fifth section summarizes and concludes the thesis.

# 2 Background

This section introduces the key concepts and processes of requirements management (RM) within software development. It gives an overview of the current issues and best practices in the RM field. Additionally, RM practices and research directions in OTS-based solutions are discussed. A gap in currently available literature is identified. The section is concluded with an elaboration on the particularities of the gambling industry, the nature of OTS software implementation projects in that context and a brief overview of Playtech.

## 2.1 Core concepts of requirements management

Requirements management is an essential process in software development that determines and manages the criteria which the software solution is based on. The definition of RM in this thesis follows that of PMI's *Requirements Management: A Practice Guide* [10]: it consists of requirements development and requirements management. Requirements development refers to the elicitation of requirements, their analysis, planning, documentation and validation. Requirements management comprises the tasks of determining the requirements baseline, ensuring traceability and managing change.

PMI distinguishes seven standardized activities for requirements process [10], brought out here along with a brief explanation:

1. Needs assessment – the output of this initial requirements analysis activity is a high-level needs definition that is used to decide on eventual solution options keeping in mind a business problem or an organizational need. This activity starts before the initiation of the project.
2. Requirements management planning – this activity determines the plan for covering the life-cycle of requirements engineering and outlines how elicitation, analysis, monitoring and controlling, and solution evaluation will be managed.
3. Requirements elicitation – this activity consists of discovering information from various sources and stakeholders that allow developing and implementing a solution that meets stakeholders' needs. This information-gathering is iterative, the results are documented and communicated.
4. Requirements analysis – requirements documentation is followed by analysis that is also an iterative process and allows for more in-depth examination and prioritization of the requirements to better understand the features and solution to be delivered.
5. Requirements monitoring and controlling – this activity refers to the management of changes to existing requirements and includes monitoring the baseline of requirements to avoid scope creep due to requirements instability.
6. Solution evaluation – throughout the evaluation activity, the delivered solution is reviewed and estimation is given to what extent the business needs have been met. This analysis can result in the need to refine the solution.
7. Project or phase closure – this step refers to the finalization of the solution, it may include support transition and reporting lessons learned, and officially completes the project.

The requirements process is fundamentally iterative, allowing for continuous elaboration and fine-tuning. This sequence of activities presented does not have to be strictly followed, instead these activities can occur in a different order, iteratively, and a certain activity could also be omitted if not relevant for a certain project. The RM process as described by PMI is shown in Figure 1.

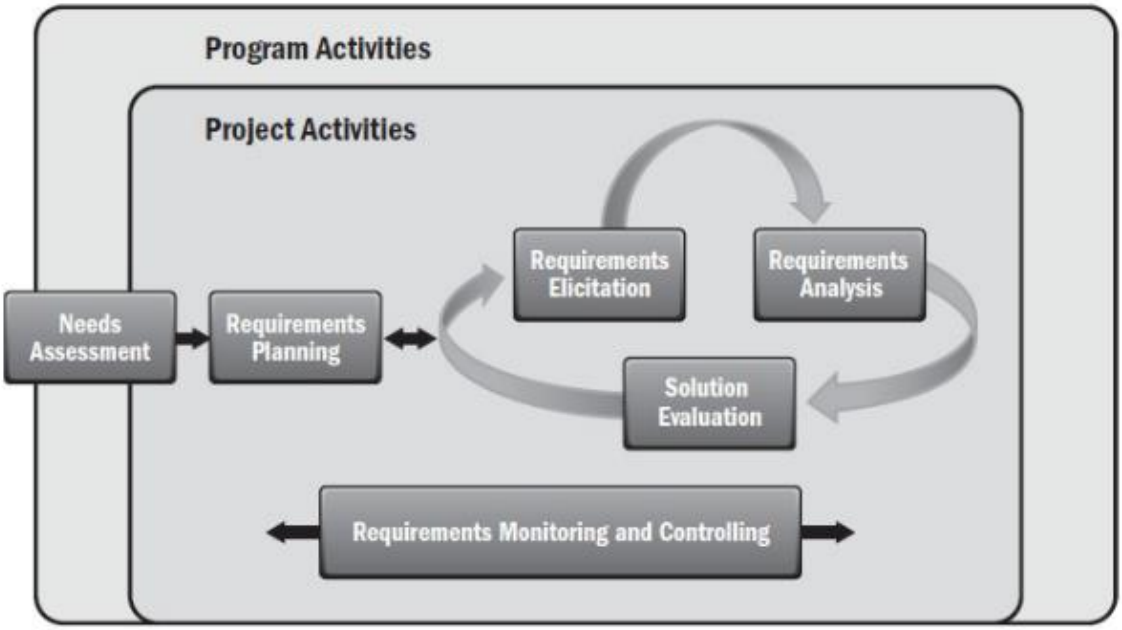


Figure 1. *Requirements process diagram* [10].

Microsoft Press's publication on software requirements by Karl Wiegers and Joy Beatty [11] introduce a somewhat different, but still largely overlapping set of RM activities. Similarly to the PMI's point of view, they differentiate between requirements development and requirements management, however they have preference for referring to the entire domain as *requirements engineering*. Wiegers and Beatty bring out four subdisciplines or sub-activities of requirements development (illustrated within Figure 2):

1. Elicitation – activities relating to requirements discovery and gathering, including stakeholder and user classes identification.
2. Analysis – in-depth review of requirements for a more thorough understanding, decomposition of requirements, their allocation to software components, identifying gaps.
3. Specification – documenting the requirements in a format that can be shared with relevant audience for review and reference.
4. Validation – review of requirements to ensure any problems are eliminated before the start of further project work, determining acceptance criteria and tests for eventual solution validation.

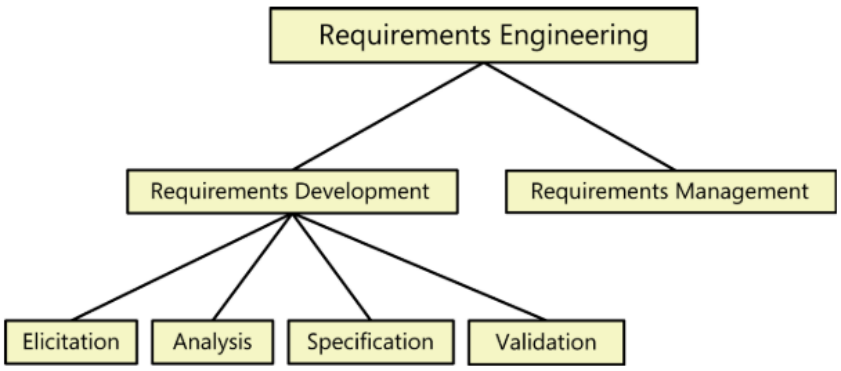


Figure 2. *Subdisciplines of software requirements engineering* [11].

Requirements management activities involve determining the baseline for requirements, managing proposed changes to requirements, aligning project plans with requirements, agreeing on new commitments resulting from changes, defining dependencies, tracing the requirements to their designs and tests, and tracking their status during the life-cycle of the project. Wiegers's and Beatty's proposed breakdown of RM activities is shown in Figure 3.

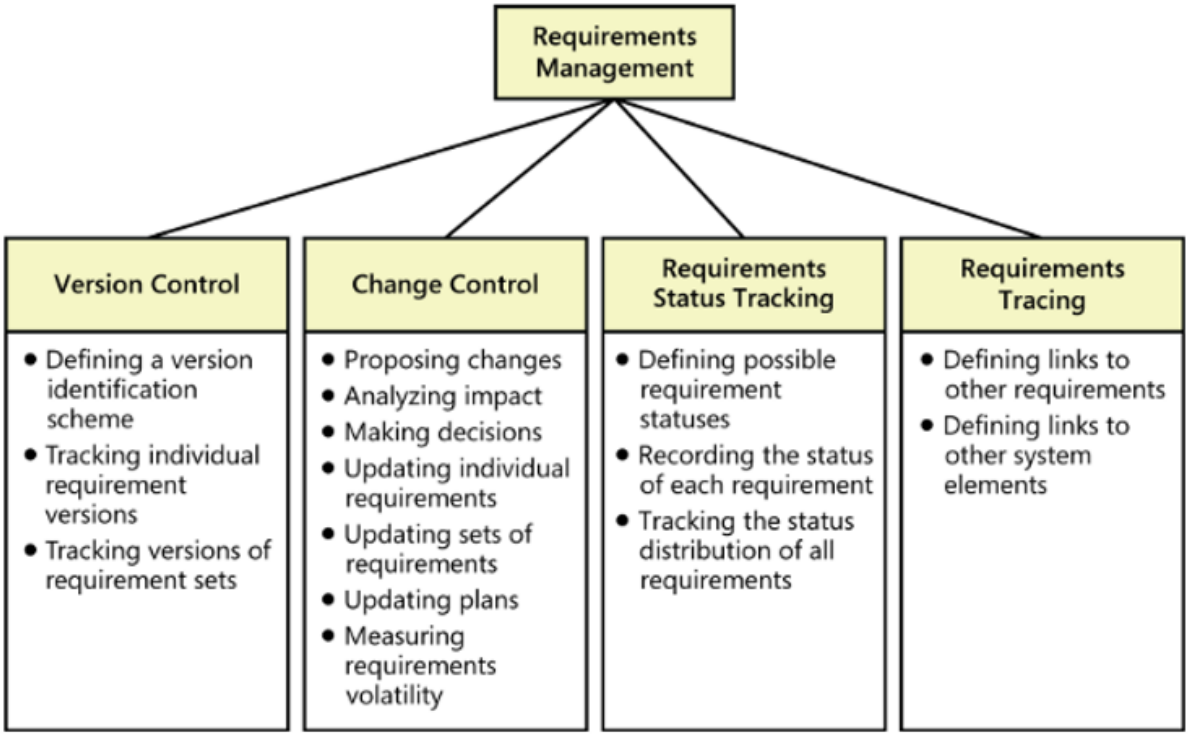


Figure 3. *Major requirements management activities* [11].

The two introduced approaches do not contradict each other, however PMI's list of activities does cover a somewhat larger area, adding the "before" and "after" activities in addition to the core steps. PMI also explicitly stresses the importance of planning for the requirements life-cycle management and highlights this as a separate activity. Wiegers and Beatty do also assign importance to the planning activities but this is generally part of the subdisciplines they outline. In contrast, the latter approach provides a more granular description of the RM activities. Regardless of the slight differences, both approaches provide a solid overview and understanding of the key processes and activities commonly considered as part of RM practices [12], [13].

A large-scale study was conducted by PMI in 2014 [3] that stressed the importance of RM in project success and failure, referring to it as a core competency within projects. The research claims there exists a lack of meaningful studies on how RM is being regarded and approached by organizations. Organizations are too often not capable of developing their people in the RM domain and management dismiss the importance of high-quality RM practices. The findings claim that in order to overcome these obstacles, the focus needs to be directed to people, processes and culture. In terms of processes, companies are suggested to formalize these to allow for consistent high-quality RM across all their projects. The study concludes also that RM practice is role-independent, meaning that the role who performs RM bears no importance in terms of the outcome of a project.

In terms of research directions in the RM domain, many of the questions raised by researchers ten years ago still appear relevant today. A set of new challenges has been identified resulting from the increased complexity of the software systems today, such as how should requirements be captured to ensure their availability for the lifetime of the system, or what is the most efficient approach for developing a number of requirements involving multiple stakeholders on a continuous basis [14].

The constantly changing environment and inability to set in stone the requirements at an early stage of a project have paved the way for agile practices. The times of pure waterfall and pure agile approaches may have passed. There is more and more discussion around the need to combine traditional and agile project management approaches, Mario Špundak's research is one such example [15]. Špundak's focus remains on project management methodologies, however the same logic applies to RM since it is frequently considered as a component of project management practices. It is concluded that benefits and downsides can be found in both traditional and agile approaches, but it is always the project and its needs that must remain the central point so that methodology is adapted to it, not the other way around [15]. Agile approach generally prioritizes time-to-market over extensive requirements documentation and tracking [16]. In case of an initiative to improve RM processes in a context where agile principles are being followed, this aspect must be considered as it may complicate establishing proper RM practices.

In addition to considering the particularities of agile projects, there are also other types of software projects that to a large extent follow the main principles of RM but require some adaptations due to their nature. Wiegers and Beatty distinguish seven types of project classes to which they suggest RM approach modifications: these are the already mentioned agile projects, but also enhancement and replacement projects, packaged solution projects, outsourced projects, business process automation projects, business analytics projects, and embedded and other real-time systems projects [11]. Based on the explanations provided by Wiegers and Beatty, the project type of interest out of the ones mentioned above and in relation to OTS-based solutions, is packaged solution projects. OTS-based solutions will be discussed in the next subsection.

## 2.2 Issues, best practices and research directions in requirements management

Requirements management field is challenged by a number of problems. Firesmith [17] lists twelve common requirements problems based on his own expertise in the domain, the negative impact of these problems and the industry best practices for remedy: poor requirements quality, over emphasis on simplistic use case modelling, inappropriate constraints, requirements not traced, missing requirements, excessive requirements volatility including unmanaged scope creep, inadequate verification of requirements quality, inadequate requirements validation, inadequate requirements management, inadequate requirements process, inadequate tool support, and last but not least, unprepared requirements engineers [17]. While the negative consequences can be easily deduced, all suggested solutions to these problems touch upon establishing a complete RM method, maintaining a central repository for all requirements and suggesting a modern, iterative approach to RM having in mind their changing nature.

The iterative nature of RM process and the need for tools that would correspond to this have been discussed widely. Pandey et al. [13] propose a requirements engineering process model that is iterative and should be fit for producing quality requirements. The model is comprised of four phases: requirement elicitation and development, documentation of requirements, validation and verification of requirements, requirements management and planning. Each

phase is in turn broken down to sub-phases as shown in Figure 4. This model should be best applicable to larger software development process that entails frequent requirement changes.

The main value of this process model is claimed to lie in the fact that first, the process is iterative in nature, and secondly, the requirements engineering activities are closely connected to the software development phases, allowing for easy recovery and change in the requirements when necessary [13]. This model is aligned with the principles and RE activities that were discussed in subsection 2.1. It remains unclear, however, if this specific model has been validated and whether any considerations should be made when applying it to OTS software implementation projects in specific industries where standard OTS components constitute more than half of the solution.

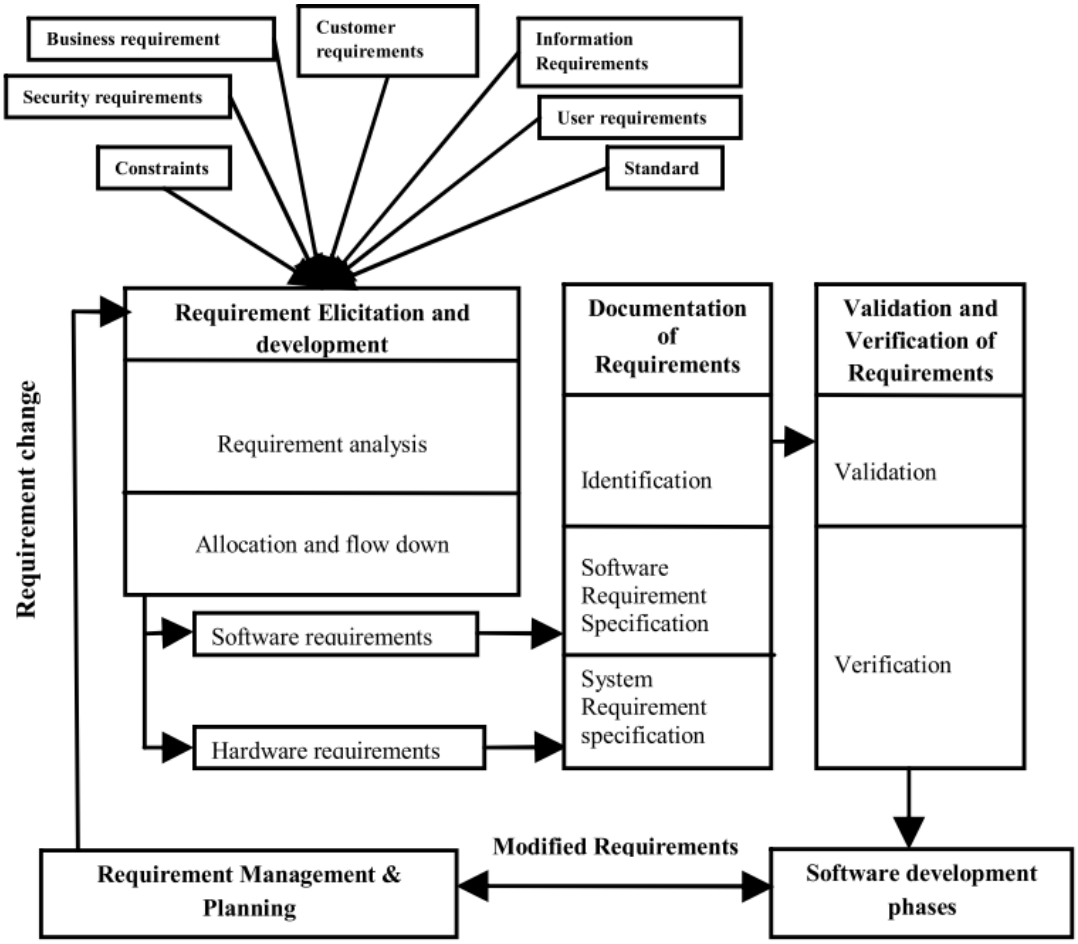


Figure 4. *Requirement engineering process model* [13].

The problem of changing requirements can be viewed from different angles. Ahmad et al. propose a requirements classification model that aims to minimize the impact of requirements change [18]. The core idea of the model is to divide the requirements into three categories based on the past experiences of the RM specialists: fixed, less likely to change, more likely to change. The authors suggest defining software modules based on these categories for minimizing the effect of change on certain solution component. It allows requirement engineers to prevent premature requirements going into development which potentially result in rework further down the road. The proposed development structure is shown in Figure 5.

The applicability of this model to all types of software projects is questionable. Software components might not always be separable in a way that would allow to select only the requirements in the fixed or less likely to change category. Requirements that are more likely to change could be inherent to a certain component to be developed and therefore could not be omitted from the plan.

The advantage of this model over several other existing systems is claimed to be the requirements categorization and expected change repository. When a requirements change request comes in, it is first checked against the change repository that includes the expected changes and therefore facilitates the application of the change [18]. This model is still to be validated by the authors.

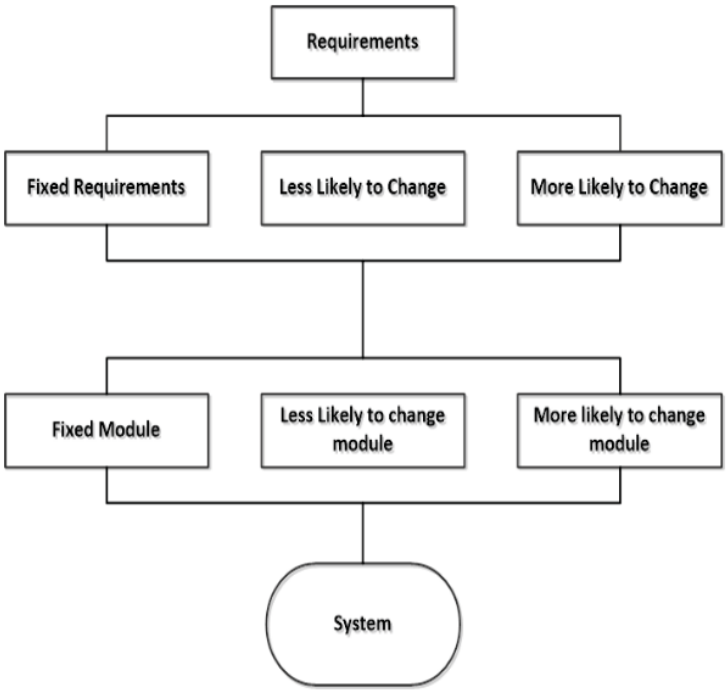


Figure 5. *Proposed development structure* [18].

Maxwell et al. go as far as proposing an adaptability framework for understanding how proposed compliance rules evolve into final rules [19]. This framework aims to help engineers predict which parts of a rule are more likely to change, so that work can start on the steadier sections of the law. As result, costs of adapting legacy systems to such changes would be reduced and the organization has higher chance of being first to market with compliant software. The framework splits each regulatory change into three components: rationale, to describe why a regulation changes; adaptability taxonomy, to describe how regulations change; and adaptability heuristics that is used for predicting whether or not a regulation will change. The framework is visualized in Figure 6.

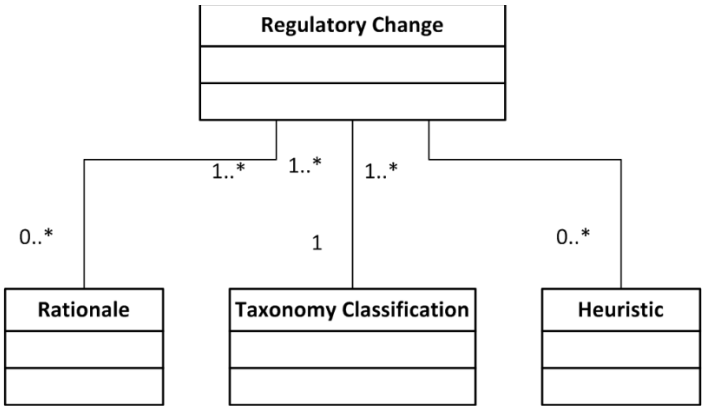


Figure 6. *Adaptability framework* [19].

This approach is especially interesting in a domain where strict legislative and regulatory rules apply, such as healthcare, finance or gambling industry. The effort required to perform such evaluation might although be disproportionate to the gain achieved from it, in case the regulatory requirements do not constitute a significant part of all solution requirements.

Today, there are various sources of acknowledged best practices to align an organization's RM processes to, such as the already referenced PMI's *Requirements Management: A Practice Guide* [10] and Microsoft's *Software Requirements (3rd Edition)* [11] to mention a few. Even though such sources often remain generic and leave significant room for interpretation, they have proven to be a solid starting point for several companies on the road to a systematic RM process establishment.

## 2.3 OTS-based solutions

Over the last couple of decades, off-the-shelf based solutions have increased in popularity. OTS component usage as a building block for larger complex systems has become widespread as it allows to reduce the effort of having to develop all software from scratch [4]. An OTS-based system is a combination of standard software components that have a certain purpose. In the context of this thesis, OTS and COTS are considered as two opposite viewpoints of the same phenomenon. When the term *COTS* is used, the perspective of the customer is followed. In case of the term *OTS*, the perspective of the software provider is taken.

In the available literature, the standard component based solutions are usually defined from the customer or buyer perspective [5], [20]. Yacoub et al. [21] touch upon the provider viewpoint and define OTS as component that is developed in-house and is reusable in similar product line applications. They regard COTS as executable software product that is licensed or sold to the customers. While this definition does not allow to clearly conclude that that OTS and COTS refer to the same phenomenon simply seen from different perspectives, they will be considered as such in the context of this thesis. OTS components combined and adapted by the software provider constitute a COTS solution from the perspective of the customer.

Due to the lack of available literature on OTS software solutions, i.e. as seen from the software provider point of view, in this subsection an overview of COTS-based systems and RM practices within this domain is given with the aim to gain valuable insight into what implications could be drawn that would apply to these solution implementations from supplier's viewpoint.

The COTS-based, packaged solutions can vary in terms of the requirements and development work needed. Based on this effort, COTS-based solutions can be split into out-of-the-box, configured, integrated and extended solutions [11]. In case of online gambling software and as seen from the supplier's viewpoint, the solutions can never be fully out-of-the-box because certain adaptation or configuration is always needed, and therefore it would fall into the categories of configured, integrated or extended solutions. Morisio et al. [5] define COTS as a vendor-supplied software product having determined functionality that acts as part of a larger system as a result of being integrated into it. In their study, they disregard the vendor perspective in the COTS package adaptation, configuration and integration work, which is specifically of interest in case of OTS solution implementation.

The initial steps in the requirements analysis phase of COTS-based development are similar to the traditional RM approach. In case of COTS solution and in contrast to traditional software development projects, some requirements may be satisfied by the COTS either partially or fully. There might also be requirements that are incompatible with the packaged component. From the customer perspective, COTS-based solutions demand more effort in terms of requirements, test and integration, while there is less work to be done in terms of design and code [5]. The same can be concluded for the OTS software implementation as it expresses similarly low demand for the provider in terms of design and code since the standard components already exist. Considerably more effort is needed when it comes to client or regulation specific customizations and integration of the whole system.

The proposed process for COTS projects by Morisio et al. [5] divides the project flow into four main categories: requirements, design, coding and integration (see Figure 7). In regard to requirements, they suggest the requirements analysis and COTS selection to be performed at the same time. The key decision is whether or not to use a COTS component and more responsibility for this decision should be assigned to the technical staff of the customer. Once requirements are described on a high level, a COTS component can be identified and selected as needed.

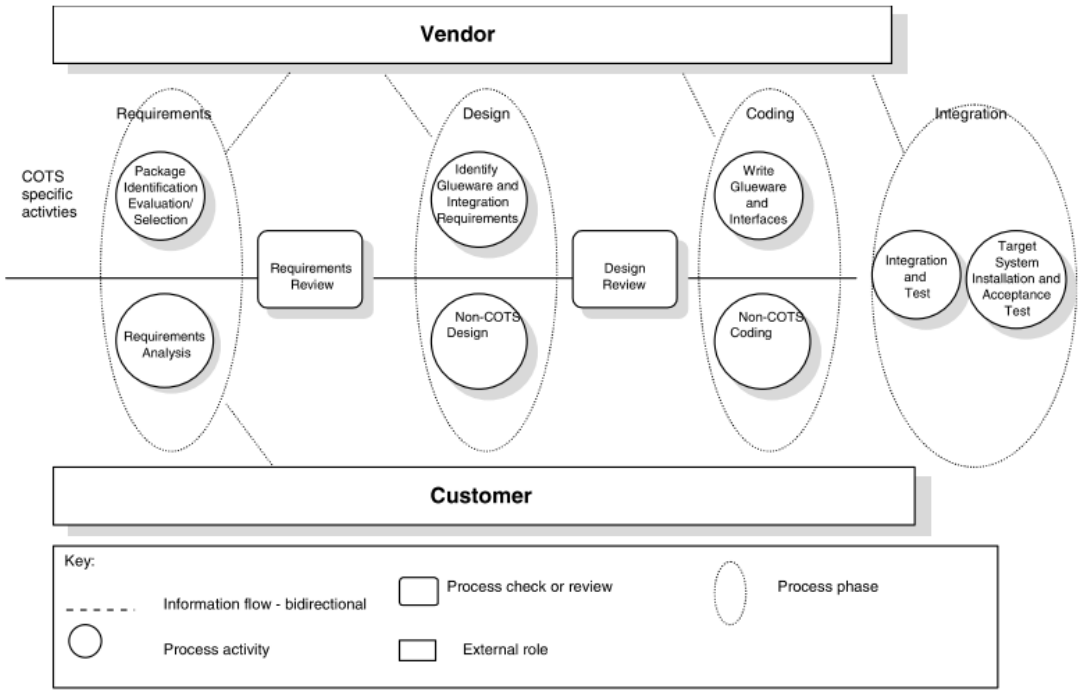


Figure 7. *Proposed COTS process* [5].

In case of OTS projects, while it is similarly important to make sure an OTS component covers certain requirements, it becomes not a question of selecting another OTS but rather enhancing the existing OTS so that the requirements would be met. The logic behind the process for the vendor compared to the buyer is therefore essentially different.

Perrone [20] differentiates between COTS product and COTS-based system. The main characteristics for COTS product include being offered as-is and as identical copies, and it can be sold, leased or licensed. For defining COTS-based system, Perrone refers to Carney [22] who distinguishes three types of COTS-based systems: turnkey such as Microsoft Office suite; mid-spectrum that relies on a commercial product together with customized elements, e.g. Oracle; and at the other end of the spectrum complex combinations of commercial and non-commercial elements. Perrone covers only turnkey COTS systems and disregards the other end of the spectrum that would be specifically of interest for the purposes of this thesis due to its likely proximity to the OTS gambling software implementations.

OTS software solutions bear a certain resemblance to the COTS-solutions described by Perrone in terms of the core value offered being essentially the same in case of each customer. The core of the OTS standard component remains the same, even though extensive configuration and adaptation might be required. As the main business need of a customer can usually be covered with the OTS products and services, a large part of the requirements process consists of a gap analysis to determine the need for additional customization and development [4].

## 2.4 Gap identification

From the background research and above discussion a gap can be identified. A lot of research focuses on RM practices either from traditional or, increasingly, agile point of view. It proposes generic best practices or models that would be at least to some extent applicable for all types of software projects. In terms of accommodating generic RM practices to different types of projects, packaged solutions aka COTS-based systems have been covered rather extensively and that essentially from the customer perspective. Research appears to be scarce or inexistent however on how such COTS solution delivery is seen from the vendor's perspective, i.e. how are requirements managed in OTS-based solution implementations from software provider's viewpoint. Additionally, it remains unclear from the literature how should RM be approached in case the standard components are used solely by the software provider as a building block for a larger system, not provided to the customer as pure COTS and left for them to adapt and enhance.

This thesis aims to address this gap by studying RM in the context of OTS software implementation projects where standard components are used as building blocks of the eventual solution, address this from the software provider's perspective, and suggest a systematic approach to RM that would be applicable for an online gambling software provider such as Playtech.

## 2.5 Gambling industry, software implementation projects and Playtech

Gambling is generally understood as betting money on an uncertain outcome. On a high level, gambling sector can be split into two main categories: land-based and online. While offline, i.e. land-based gambling has its roots in distant history, the first online casino was launched in 1994 [23]. The main products available in online gambling are sports betting, poker, casino (based on random number generator (RNG) as well as live dealers), bingo and lottery. If a software provider's product portfolio includes any of these gambling verticals, this means the products

exist as in-house building blocks ready to be implemented. Today, the industry standard is such that gambling products are expected to be available anytime, anywhere and on any device. This expectation adds to the struggles to deliver high quality technical solutions and ensure highest availability in accompanying services.

Gambling industry is becoming increasingly regulated. While some years ago an overseas online gambling license would have sufficed to launch an online casino in several countries, more and more governments are working on developing their own criteria for gambling regulation, often fully blocking operators who do not possess the relevant licence. The increasing complexity by the introduction of new regulations demands more effort from both the operators and software providers to meet the regulatory requirements in technical solutions.

Solutions that online gambling software companies provide to their operators (i.e. customers) are commonly based on a revenue share model agreement. Therefore, continuous cooperation and business relationship between the software provider and operator is inherent to these solutions. This guarantees the software supplier's interest in helping the operator maximize its business to, in turn, increase their own revenue. Consequently, the software solutions that are delivered through the implementation projects are not simply setup and sold. Instead, the software provider must be capable of maintaining the quality of the solution and accompanying services throughout the duration of the business relationship. It becomes more about designing and delivering a service instead of a static solution. Therefore, agreeing on the commercial, service level, hosting and other terms is characteristic of such business-to-business relationship.

From a technical perspective, an online gaming solution requires hardware and software. Some operators might already have their own hardware or decide to procure it. Many customers might need to use that of the vendor. This is especially true for standalone solutions that fully rely on the vendor's hardware and software components, needing no additional integration. An established gambling software supplier usually has extensive hardware solutions in place and therefore can provide hosting and access to applications centrally. In case of solutions where customer already has an external primary system, the supplier is still being relied on for certain services and integration aspect becomes of importance.

As for software, the main gambling products, e.g. casino, poker and the central platform, are to a very large extent already existing standard packages of software, i.e. OTS components, that need to be adapted for each customer. Choosing the right components, applying the right configuration, performing a certain amount of custom development, applying branding and putting all the systems together by means of elegant integration constitutes the main part on OTS-based online gambling solution implementation project work.

Playtech is one of the largest online gaming software suppliers in the world offering top quality gaming solutions to gambling businesses across the globe. The company serves approximately 140 global licensees (i.e. operators) and operates in 20 regulated jurisdictions [24]. In an increasingly regulated world of gaming, Playtech finds its strength in delivering cutting edge technical solutions through projects that range from almost fully off-the-shelf to complex off-the-shelf based customized solutions.

These projects require the involvement and cooperation of specialists from various business units in the company, especially in a situation where merges and acquisitions of other service providers in the gambling industry are increasingly common. As the company grows through merge and acquisition, new kinds of approaches and processes are being brought on board that require alignment with the rest of the company. In such context, one of the prerequisites for

solution implementation success is the efficient collaboration between these parties. The other aspect of success is the methodical management of requirements. The lack of a systematic RM approach may lead to inefficient use of financial as well as human resource to deliver a solution, resulting in higher project cost and delivery time than estimated. This is the lack that is being addressed in this paper.

# 3 Case study

The method chosen to address the research question of this thesis is case study. Case study is an empirical method that investigates a phenomenon in its context, it is most useful when the context of the phenomenon cannot be easily detached from the phenomenon due to blurring lines between these two, and it helps to bring about growth in our knowledge of the studied field [25]. RM practices in a company depend on the characteristics of that company and the domain it is operating in. Case study allows us to investigate the RM field while taking into consideration the specificities of the surrounding organizational structure and conditions.

Various types of case studies exist. Robson differentiates between four of them: exploratory, descriptive, explanatory and improving case studies [26]. The current case study is mainly of improving nature, although it cannot take place without a certain amount of exploration. Improving case study aims to improve some aspect of the phenomenon whereas exploratory case study attempts to gain an understanding of an existing phenomenon and obtain new insights [27]. To be able to answer the research question of how requirements can be managed when implementing OTS based online gambling solutions, it needs to be first explored how this is being done today, what works well and what not, and only then improvement possibilities can be determined considering existing best practices in the domain. The relation between case studies and software process improvement is understandable due to the focus of case studies on specific phenomena in their specific context [28].

In this thesis, an improving case study is conducted in Playtech with the objective to examine the status quo of RM practices, understand what works and what are the problematic areas, to be able to suggest a process for efficient RM in OTS software implementation projects and thereby answer the main research question of this thesis. The following subsections provide a more detailed overview of the case study setting and design. Description of the execution of the case study is given and the results are reported.

## 3.1 Setting

The case study is conducted in an online gambling software providing company Playtech. Playtech is among the largest online gambling software suppliers in the world and it offers complex gambling software solutions to approximately 140 customers in various countries and regulations across the globe. At the core of the software solutions provided are the OTS products such as casino, poker, bingo or sports betting, that require configuration, adaptation and sometimes also custom development depending on a variety of customer demands and specific regulatory conditions.

The core weight of responsibility for RM activities in OTS software implementation projects lies on the shoulders of solution architects and technical project managers, belonging to the Solutions Architecture and Integration business unit. Therefore the focus of this study will be on this department. In Playtech context, a solution architect is responsible for defining and maintaining the structure of the software solution, i.e. ensuring that the implementation and integration of chosen OTS products is technically feasible, effective and in line with the company's strategy as well as technical standards. Solution architect is also accountable for making sure that the solution meets requirements demanded by the customer or the target market legislation and regulation. The solution architect is generally involved in the earlier stage of the project lifecycle, while the technical project manager becomes included at a later stage when the actual solution implementation work is started. A technical project manager serves as a single point of contact at the solution implementation stage for all technical queries

within a project coming from either internal or external stakeholders, and coordinates the work between technical teams. At any point in time, each solution architect and technical project manager is involved in more than one project, the exact number depending on the size and complexity of the projects.

Requirements management is inherent to each OTS software implementation project. The extent of the RM activities although depends on the scope of a specific solution to be delivered. The OTS products provided are essentially standard components developed and maintained by relevant product departments in the company. The role of solution architects and technical projects managers lies mostly in determining and implementing the full software solution and coordinating the configuration implementation and adaptations, depending on which OTS products have been selected by the customer as well as on the limitations prescribed by regulations.

Additional stakeholders exist in this process that the main carriers of the responsibility heavily interact with. First, the commercial and sales units who agree on business deals and sign agreements ideally considering the input from key people from other business units including service operations; then the representatives of compliance and legal departments; the business project management unit that acts as the key driver in the solution delivery; the teams involved in the hardware and infrastructure setup; the central platform provider that often needs to accommodate developments to comply with new directives; and those from each gaming product that will be part of the solution. Therefore, the RM activities demand synchronization between various teams.

## 3.2 Design

The data sources that this qualitative study relies on are interviews, organizational documentation and guidelines on RM practices, and documentation of a set of completed projects.

First, the interviews are conducted with eleven employees of two different roles in the company who belong to the same business unit and are among the key stakeholders in software implementation projects. These roles are solution architects and technical project managers. The study involves all solution architects in the business unit and one fourth of all the technical project managers. The reason for such subgroup of technical projects managers is that the selected employees have been more extensively involved in RM activities in projects compared to other team members. All interviewees are closely involved in RM activities, although some more in the requirements development phase and some in requirements management phase. The interviews are semi-structured, enabling an open discussion. Interview questions can be found in Appendix I. All interviews are recorded for the purposes of analysis. The recordings are only being used for the author's reference and will not be shared. In case sharing is required, relevant interviewee will first be consulted for consent. A confidentiality agreement with each interviewee must be reached accordingly.

Secondly, a review of documentation and guidelines relating to RM practices is conducted. This consists of determining existing sources and policies that describe the RM process. The managers of relevant teams are consulted for obtaining this documentation. Additionally, the organization's internal wikis are reviewed to ensure no relevant documentation is missed.

Thirdly, a set of completed implementation projects is selected with the help of relevant business unit managers. Considering that at any point in time, approximately hundred solution

implementation projects are being run, a set of ten projects of different scope and solution are considered a representative set by the business unit managers. The RM aspect in each project is studied and analysed based on project documentation. Similarities and differences in approaches are determined and problem areas are identified.

## 3.3 Results

The interviews were conducted during week 48. Eleven employees were separately interviewed: eight solution architects and three technical project managers. Nine employees were interviewed face to face, two employees were consulted over a call. Each interview was recorded as an audio file. The duration of the interviews varied from 25 minutes to 1 hour, the average interview length being 40 minutes. Interviewees were informed of the background and purpose of the interview within the meeting invitation. At the beginning of the interview, the interviewee was informed once again of the topic of the research, the purpose of the interview, and of the fact that the interview is being recorded for data gathering purposes. It was also agreed with the interviewee that confidentiality will be ensured: the recording will not be shared without previous consultation and agreement with the interviewee and the data gathered will not be traceable to a specific person.

The interviews were conducted in a semi-structured manner, relying on a list of questions (see Appendix I) that guided the discussion. While progressing through the interviews, it became apparent that not all questions were equally relevant and hence the focus was directed to those that allowed to unfold the topic more effectively. All interviewees were cooperative and glad to be providing their input to the research.

All interviewees have been working in the company for more than three years, six of them for nine years and more. Each of the participants has years of experience being involved in RM activities in one way or another. The professional background of the interviewees varies. Several of them have previous experience is software development as programmers, project managers or technical account managers. Six of the participants are regularly working together in one room, the other employees are scattered either in the same office building or other company offices abroad. The profile of interviewees can be found in Table 1.

Table 1. *Profile of the interviewees*.

| Employee | Role | Employee location | Interview method | Years in company |
|---|---|---|---|---|
| Employee 1 | Solution architect | Domestic | Face to face | 11 |
| Employee 2 | Solution architect | Domestic | Face to face | 11 |
| Employee 3 | Solution architect | Domestic | Face to face | 9 |
| Employee 4 | Solution architect | Domestic | Face to face | 11 |
| Employee 5 | Solution architect | Domestic | Face to face | 4 |
| Employee 6 | Solution architect | Domestic | Face to face | 7 |
| Employee 7 | Solution architect | Foreign | Remotely (call) | 5 |
| Employee 8 | Solution architect | Foreign | Remotely (call) | 4 |
| Employee 9 | Technical project manager | Domestic | Face to face | 11 |
| Employee 10 | Technical project manager | Domestic | Face to face | 9 |
| Employee 11 | Technical project manager | Domestic | Face to face | 5 |

The documentation review included the analysis of existing process guidelines. The documentation consisted of the business unit's internal wiki pages that were studied to get an understanding of the level and quality of information available. Due to the rapidly changing

nature of all the information in the organizational context, it is generally preferred to store product as well as processes related information in a collaborative working space such as Confluence[1], instead of static documentation in the form of Word or PDF files. Confluence also includes version history functionality which is beneficial for tracking changes.

The wiki pages included brief descriptions (i.e. one or two paragraphs) of high-level responsibilities and roles. From this documentation review it can be concluded that technical project managers are said to serve as a single point of contact for all technical needs in relation to projects and act as a coordinator between other technical teams. Solution architect's role is described as the person responsible for defining and maintaining the structure of the solution with the aim to ensure that the solution will meet the business requirements. The core responsibilities include understanding of the requirements, formulating the design, communicating the architecture and verifying the implementation. It is not evident from either definition however what the cooperation of solution architects and technical projects managers should entail, what principles their collaboration with other departments should follow and how these should be established, especially having the management of solution-related requirements in mind. No guidelines or process description for RM practices can be found in the wiki space.

The analysis of completed implementation projects involved the review of available documentation specific to ten selected projects. This set of projects was selected with the help of relevant business unit managers who considered it a representative selection of concurrently running implementation projects at any point in time. Such solution related information is generally stored in Confluence and in two separate sections – one created by solution architect and the other by technical project manager. The main output of the solution architect work is the solution document that is essentially a description of the technical solution to be implemented. The solution is intended to meet the needs and business requirements of the licensee, however it does not include a clear list of requirements. The technical solution description is often developed in Confluence so that it could be exported and shared externally as technical solution overview. Sometimes the solution document is simply uploaded to the wiki space as a Word document and requires download to be viewed. In some cases, additional project-related information is stored in the same wiki page but there appears to be no defined list of topics that is covered, rather the decision to document information seems to depend on the employee and specific needs of the project. The wiki page created by technical project manager tends to include more detailed information about the actual solution implementation, such as site and gaming client details. The format of the pages again varies per person and same applies to the amount of the information documented.

### *3.3.1 Requirements life-cycle*

Based on the data gathered from the interviews, the high-level as-is requirements process as seen by solution architects and technical project managers was mapped (see Figure 8). The initiative starts with a commercial deal being made. In most cases, the solution architects who later become responsible for eliciting, gathering and documenting the requirements and technical solution, are not involved in the initial stage. The scope of the project is being agreed on among the commercial stakeholders on a higher level, determining the gaming products to be offered, the regulation that the licensee is aiming to operate in and various service level

[1] Confluence is an online collaboration workspace developed by Atlassian.

criteria. All activities outlined in Figure 8 overlap to some degree, the extent to which this happens depends on the specific project and people involved.

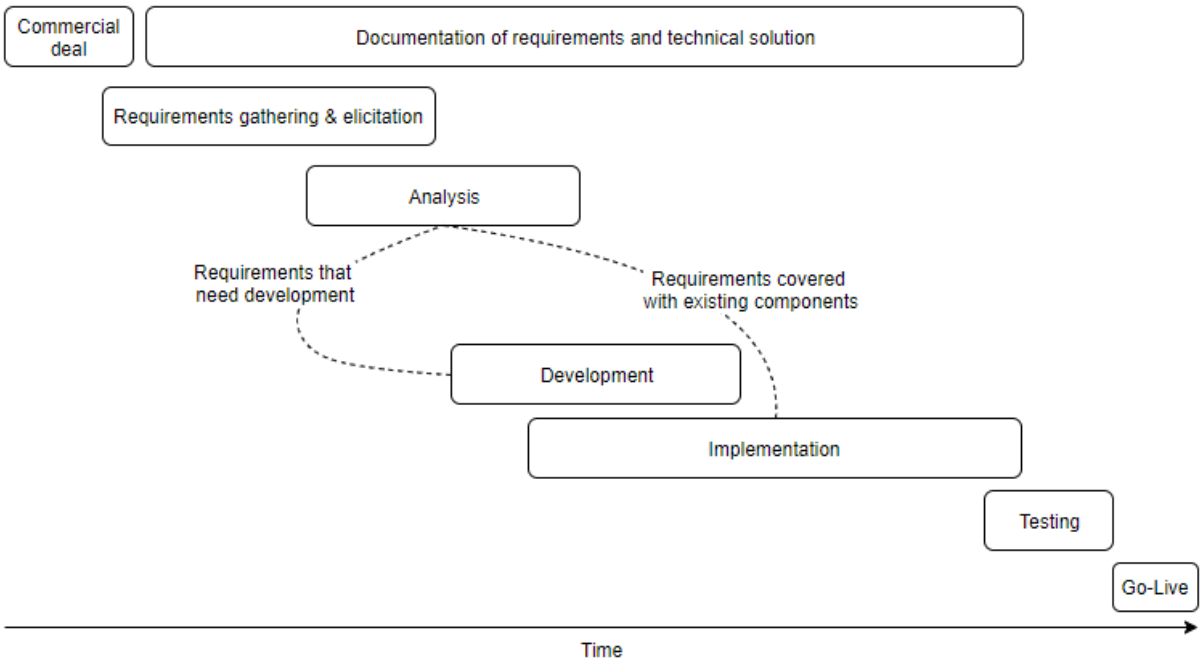


Figure 8. *High-level as-is OTS implementation project life-cycle from RM perspective*.

Once the solution architect has been involved, the requirements gathering and elicitation starts and in parallel, the gathered information is being documented on a continuous basis. While the requirements gathering lasts – and to a lesser extent it could last until the end of the implementation – the analysis starts. The first step in the analysis phase is to determine which requirements are supported by existing functionality and which require development. Those that are covered with existing standard components, can move forward to the implementation phase and this is where the technical project manager usually becomes involved. Those for which development is needed, will move forward gradually as soon as they are fit for implementation. At the end of the implementation phase, both internal and external testing is performed to ensure compliance with agreed requirements, followed by the go-live of the solution.

There are various stakeholders in the requirements engineering process that solution architects and technical projects managers either work closely with or consult upon need (see Figure 9). One of the main sources for input on requirements is the licensee who needs to put forward their business requirements. Commercial and Sales teams provide input on what terms have been agreed on with the licensee on the commercial level. Compliance and Legal teams are closely involved in clarifying the requirements when new regulations come into play. Business Project Management unit is responsible for the overall management of the project, including streamlining information flow and management of project and development timelines. The Integrated Management Solution (IMS) as the platform and operations management system department needs to be able to accommodate developments required to meet the needs of the project. Product owners or product analysts are consulted in case the requirements pertain to a certain product and might result in a development by that product. Infrastructure unit helps to make sure the physical sites are fit for the suggested solution in terms of capacity and other related aspects, and also provide support with performing the technical setup of the application or product and preparing the environment for further implementation. Integration managers

become involved in case integration with external licensee systems is needed. Solution architects and technical project managers are meant to oversee the requirements process and ensure all gaps are filled.

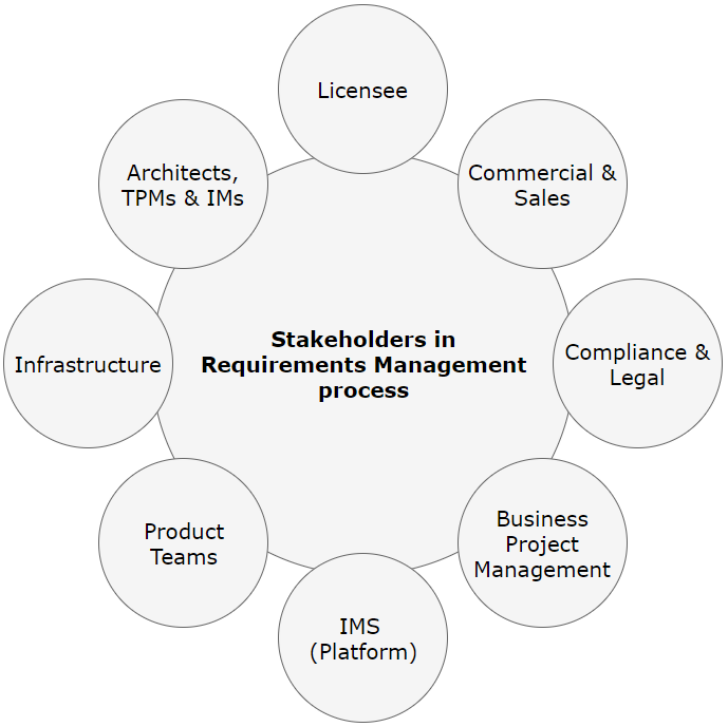


Figure 9. *Key stakeholders in requirements management process*.

Today there appears to be no standard way for gathering the requirements. In some cases, face to face workshops are being held with the licensee to elicit and clarify the requirements at an early stage of the solution discussions, however in many cases the requirements that have already been gathered in some way are sent by various people in different formats per email to the solution architect. Although it is not common, occasionally licensees have their own templates for providing requirements in a certain format which facilitates the elicitation process for the implementation project team. Regulatory requirements are usually provided by Compliance team or, in some cases, by the licensee.

One of the issues that emerged from the interviews was the fact that the scope that is commercially agreed on can be vague in terms of content, timelines and effort, and therefore it is unclear for the solution architect what exactly needs to be delivered. Also, not all projects involve face to face workshops at the start of the project that would allow for a closer familiarization with the licensee's business and their needs especially in case of a completely new customer. In case of an existing customer when both parties are already familiar with each other's setup and particularities, a face to face workshop is considered to add less value.

### *3.3.2 Requirements documentation and management*

At the point where at least a certain portion of the requirements have been provided, the analysis and documentation start. As the first step, the requirements are reviewed and marked as supported by a standard component or filtered for further discussion with relevant product teams for potential development. The way the requirements are being documented today varies: they can be stored in a Confluence wiki space, in a personal Excel sheet or an online shared

spreadsheet, or in JIRA[2]. Sometimes a 3rd party collaboration tool is chosen by the licensee with the suggestion to manage all requirements there. One approach that all the interviewees agree on is that IMS development requests are recorded and managed in JIRA. There is no unanimity however in terms of who is responsible for managing and tracking those development requests. There has been an initiative to start logging all technical solution related requirements in central JIRA, not only the ones that need development, but this is currently performed only as a proof of concept. Other tools mentioned that are used in the analysis and documentation phase are Visio and Enterprise Architect for modelling diagrams and process flows.

In case Confluence is used as the central requirement repository per project, it is possible to link JIRA tickets with that Confluence page thanks to the integration between the two systems. The wiki page also allows to store relevant sources of requirements in the form of documents or other types of files, increasing the level of traceability of the requirements. From the documentation review however, it can be stated that this is not a standard practice. In general, traceability is not something that is considered by the interviewees to be maintained in written form, rather the trace exists as long as the person working on the project remembers these links.

It is unclear to what extent the technical solution description in the solution document should be updated. In practice, there are cases where the solution document includes several *to be determined* elements and has remained a draft. In some cases, efforts are made to keep the document as up to date as possible. It was pointed out by the technical project managers that ideally the solution document would cover the actual solution that was implemented, not the one that was planned and hence the changes that occurred during the project should be reflected in the document accordingly.

As mentioned earlier, the solution document does not include a clear list of requirements. Even if clear lists of requirements exist in some personal Excel files, they are generally not stored in the central wiki repository. Additionally, it has been pointed out that even if new requirements are explicitly defined in written form at some location, those requirements that are covered with standard components and functionality, are mostly marked with *existing functionality is sufficient* or similar comment. It is not always guaranteed that the details and description of this existing functionality are available somewhere in a written and holistic format. This vagueness complicates the tracking of the fulfilment of the supposedly well-understood requirement as well as eventual testing because there is no clear record of it.

There are various requirements grouping approaches used by the solution architects. The most common one is to group requirements and technical solution description according to the products and services they relate to. Several interviewees have their own list of items that they group the needs and requirements to and which they rely on when compiling the technical solution description. This list is compiled based on each individual's own experience in the field. The idea of having a pre-agreed and shared approach to such grouping to facilitate the requirements mapping process was expressed a few times throughout the interviews.

There are certain requirements relating to the technical solution that only come up at a later stage in the project. These requirements are usually handled by the technical project manager or the business project manager and since the level of complexity of those requirements is

[2] JIRA is an issue and project tracking software developed by Atlassian.

generally lower (e.g. configuration related, capacity estimates), might be implemented or forwarded to relevant teams for actioning straight away without being clearly noted down.

The practice of explicitly baselining the requirements is not common. An agreement with the customer on the technical solution is more commonly reached on the basis of the solution description, not specific requirements. A part of the requirements is oftentimes subject to change and due to the nature of the business relationship with the licensees, the focus is on achieving the goal and meeting the customer's business needs, even if they change, unless this has negative impact on any other aspect of the operations. Requirements change management does not follow any certain official policy. If a request for change comes in, it is being discussed and evaluated the same way as the initial requirements. There appears to be no issue with that approach, until the aspect of tracking the changes is considered. Depending on the requirements management tool used, it might have a version tracking functionality, such as Confluence and JIRA. In case of a static or online spreadsheet, the change tracking becomes impossible and is a potential source of confusion.

It is difficult to determine what is the exact extent of requirements in an average project that are covered by standard components. Across approximately hundred licensee projects that are running at any point in time, the rough estimate is that more than half of the requirements are met with existing functionality, i.e. OTS components. Few difficulties are usually experienced with determining the items that are supported with off-the-shelf components. Most of the focus in the RM practice is directed to the not-yet-supported requirements, their analysis and guiding the internal discussion with various product representatives to find the best solutions through adaptations and development.

### *3.3.3 Assessment of current requirements management practices*

It was evident from the interviews as well as from the documentation review that there is no official policy or guidelines for developing and managing requirements. Also, there is little understanding of how other people in the same business unit perform their work. While no form of structured knowledge sharing or lessons learned sharing is being practiced, this is being done unofficially especially among the employees based in the same office and room. New employees joining the unit are expected to learn by doing, have a can-do attitude and are encouraged to ask questions. This lack of organized, systematic approach was considered by some employees as inevitable and even a positive phenomenon claiming that it allows people to make the best use of their personal toolbox without overly restricting them.

Except for information sharing during small breaks from work, knowledge sharing was claimed not to be a common practice today, however the interviewees would consider this valuable and enriching. Over the last couple of years, there have been initiatives to organize such sessions but they have mostly occurred on a per project basis, not focusing on aligning employees in regard to more general practices or processes that are relevant for their work. Additionally, the difficulty of aligning all employees in different locations and finding suitable time in everyone's busy schedules was brought out as one of the reasons for the lack of more formal knowledge sharing. From documentation review it can be concluded that gathering practical information and how-to's in a Confluence knowledgebase is being occasionally done but it does not seem to be actively managed. Information is not always easy to find as it is not well structured. At this point in time, there does not seem to be a clear approach to information management within the business unit.

The interviewees were asked to express their opinions on the current RM practice and point out which parts of it are running smoothly as well as highlight those that are most problematic. The reality seems to be that all projects are eventually launched, so in one way or another the business goal is achieved. It was pointed out during the interviews that the key stakeholders in the RM activities are capable of and used to quickly react to and accommodate changes. Communication is considered as key enabler and once most of the requirements have been obtained, the internal alignment with product stakeholders is considered to run rather smoothly. Few issues occur when all requirements are clear or covered with existing off-the-shelf components and functionality.

Problems occur when the input during requirements gathering is vague, regulatory requirements need to be translated from another language and hence the likelihood of misinterpretation rises, there is a need for a significant cross product development synchronization or when not all project stakeholders are involved in the conversations where they should be. Problems might arise also when there is no clear plan introduced at the beginning of the project as to how requirements will be gathered and managed, including agreeing on which tools to use and who takes ownership of what actions in the process. The main problematic areas outlined by technical project managers were related to the lack of documentation of requirements and solution by solution architects, in cases where a solution might have been implemented in haste without leaving a considerable trace in the form of documented solution. It becomes very difficult to retrospectively understand what was implemented and the reasons behind.

The idea of lessons learned sharing is regarded as controversial by some interviewees. Not all stakeholders tend to perceive it in the same way. There have been occasions where lessons learned have been documented by technical project managers with the purpose of not making the same mistakes again next time. Such initiative was perceived as blaming other departments, not taken as constructive criticism.

In regard to improvement possibilities of the current requirements practices, it emerged from many interviews that one central location for gathering requirements would facilitate the process and enhance visibility for several stakeholders. All information relating to the environments to be used, configuration aspects such as languages and currencies, time constraints, regulatory and other specific requirements – having this is one location would be considered a significant value add. The problem with using the existing systems such as Confluence and JIRA however appears to be the fact that not all internal stakeholders currently have access to them. It was also pointed out that ideally external parties should have access to the same requirements repository as well. Granting access to such internal systems to external parties has not been possible so far due to company policies. This became evident during an initiative by some employees in the business unit a couple of years ago to establish a collaborative requirements management system with the licensee in JIRA. The initiative was put on hold.

Even if such central system is being used, the question remains who is taking the lead in managing the requirements. Currently these lines of responsibility are rather vague between the solution architects, technical project managers, business project managers and other stakeholders in the project. It remains similarly unclear who should take ownership of following through on the development requests once these have been raised.

This central requirements management system is expected to enable grouping of requirements, marking them as covered with existing components or requiring development, linking them to development tasks, assigning them to relevant teams and specialists for actioning, linking

sources, building hierarchies, marking statuses and percentage of completion. As a result, it should be possible to have an overview of all requirements and hence get an understanding of the actual progress of their implementation with the possibility to export the status in suitable format to be shared with external parties.

Another aspect that could be improved and which is closely related to the previous idea, is the agreement on approach, framework, plan and tools to be used in the requirements management process which should also be aligned with the licensee at the start of any project to reach an agreement on how the work will be performed. Most of the interviewees mentioned that it would be beneficial to have at least some kind of high-level guidelines for the requirements process that are followed by all but that are sufficiently flexible at the same time. It was especially highlighted by the technical project managers that if all architects had similar style and approach to requirements management, it would simplify their work in terms of being able to grasp faster the scope and content of the solution.

The idea of assigning solution architects and technical project managers not per project but per licensee was reflected by a few interviewees. Currently most of them are shuffled between projects. Being dedicated to certain customers is expected to allow for a more in depth understanding of the licensee's business model, resulting needs and requirements. Additionally, it is believed to enhance cooperation and understanding between the solution architect and technical project manager who would work as a team on projects of certain customers.

Due to the lack of formal guidelines and knowledge sharing, it was stated during the interviews that there is a lot of reinventing-the-wheel taking place. If someone in the business unit has come up with a proven successful approach or a well-structured template document approved by many, this could be shared and made a standard to contribute to an established process. In terms of template documentation, an idea of having an agreed form for licensee requirements gathering was raised since many licensees do not have any standard way of documenting their needs and requirements. This could increase the quality of the input received.

# 4 Discussion

This section discusses and analyses the findings from the case study in the context of the background information provided in section 2, with the purpose of proposing a process for managing requirements when implementing OTS based online gambling solutions. Additionally, threats to the validity of this process are considered.

The literature on requirements processes in the scope of in-house OTS based solutions is scarce and especially in the online gambling software context it is literally inexistent. Therefore the reference points used in this thesis have been found in more generic RM best practices which, combined with the results of the case study and author's knowledge of the online gambling software industry as well as the case company, allow to work out a RM process fit for the case company.

From the case study results it appears that the way requirements development and management work is being performed today in Playtech enables to achieve the business goals and deliver online gambling solutions to licensees. The efficiency of the delivery process is however questionable, including but not limited to the RM practices involved. The eventual outcome is likely to owe its success to the resourcefulness and goodwill of several employees regardless of inefficient practices in place. The same and even better outcome should be achievable with considerably less overhead. The case study findings resonate with the common requirements problems mentioned by Firesmith [17], especially with the problems of inadequate requirements management, process and tool support.

## 4.1 Lack of systematic approach

The lack of systematic and agreed approach to RM brings us back to the three critical areas pointed out by PMI: people, processes and culture [3]. While the importance of knowledgeable specialists and the development of their skills can be appreciated, as well as the relevance of standardising and formalising processes for achieving consistency, it cannot be successfully achieved without the culture of the whole organization adapting to this mindset starting from the top management. The management commitment and support are the enablers of proper alignment on lower levels of the organization, resulting in streamlined processes, less waste and more success.

Cultural differences and dissimilar view of work ethic act as an impediment to the establishment of a consolidated approach. Not all cultures adapt to new processes and policies in the same way. Some cultures are more at ease with adapting to and following agreed processes than others who might be more oriented on the outcome with less respect to systematic approach. Although the cultural particularities resulting in different attitude towards work arrangement might complicate establishing a methodical approach to requirements practices, it has become evident from the interviews and documentation analysis that value is being seen by the key stakeholders in establishing a process for efficient requirements management.

Process establishment requires thorough alignment with all stakeholders in the RM process. As appears from the previous section, the current lines of responsibility are unclear and inconsistent from project to project, raising the question of ownership in various project aspects including RM. The importance of ensuring that RM process flows in accordance with the management of the rest of the project and its timelines cannot be overlooked.

Due to no process nor agreement between business units being in place, RM activities in a project are currently not being planned. PMI brings out requirements management planning as one of the eight steps in the requirements process and considers it an essential aspect of the overall project planning activities [10]. As per PMI, such plan helps to ensure that the RM approach in a project is optimal and clear for all parties. It also demands the commitment of stakeholders involved in the process as well as effective integration to the project management plan overall.

## 4.2 Documentation

The requirements documentation practices vary greatly within the technical solution business unit. While one of the outputs of requirement analysis is the solution document, there is no clear agreement on what should be documented, where it should be documented and how. The fact that various systems and tools are being used depending on the employee's preference makes it difficult for other stakeholders in the process to optimally contribute because each new project might require getting accustomed to a new approach. Additionally, in case insufficient trace is left by the employees documenting the solution and requirements, it complicates the support processes post-launch since at a later stage it becomes difficult to determine what the solution was supposed to entail. While it could be considered unreasonable and time consuming to document every single step, it is important to find balance between writing up too little and too much.

## 4.3 Information management

To transfer knowledge to other projects or programs in the company, it is critical to collect, document and apply lessons learned for continuous improvement [10]. This is no less important for passing on information to other employees, either existing specialists or newcomers. There is currently no established approach to knowledge management or knowledge sharing, including lessons learned sharing, within the business unit. It could partially be due to the fact that employees are scattered across different offices but having today's communication tools at hand, this should not constitute an impassable obstacle. In a domain where technological solutions advance at high speed and new information becomes constantly available, it is inevitable that the knowledgebase cannot always be kept up to date. This however is not a valid excuse for not documenting knowledge, instead this might not even present a significant problem if content curation rules have been agreed upon and employees maintain a critical perspective to the available information.

## 4.4 Proposed requirements management process

Based on the knowledge extracted from the RM background and best practices research and findings concluded from the interviews and documentation review, a process proposal for managing requirements of OTS based online gambling solutions is made.

It is suggested that in order to increase the effectiveness of requirements management capabilities, one needs to address the three critical areas of people, processes and culture [3]. For being able to establish an effective process, the surrounding culture must support the initiative. There is a set of accompanying and enabling activities that help to create a culture that supports process establishment. As follows, a list of four enabling activities is brought out, derived from the gaps determined throughout the analysis.

First, to allow for a systematic approach to RM, alignment with management and all other stakeholders in the RM process is needed. Recognition for RM as a core competency starts at the top [3]. Any process improvement initiative requires a process owner who would be accountable for defining and maintaining a process [29]. Alignment would also include agreeing on responsibilities between all parties in the requirements process. This could be done in the format of a RACI[3] matrix.

Secondly, it is essential to bridge the gap in information management principles. Therefore, these should be agreed on within the Solutions Architecture and Integration business unit. The principles could cover aspects such as where all project related information should be stored, what kind of information should be included, which templates it should follow and who is responsible for keeping that information up to date. Not less important would be agreeing on principles for lessons learned sharing to enable continuous improvement of the quality of the work.

Thirdly, the establishment of a RM process should entail explicit description of the process. This could be a written and centrally available list of steps in the RM process. Additionally, the templates to be used in this process would need to be clearly described and maintained.

Lastly, following the process description, it should be made clear what are the tools and systems to be used in the RM process for storing information, recording and managing requirements and creating diagrams or process flows. As could be concluded from the interviews and documentation review, currently there are different tools being used for all these actions, making it difficult to streamline the process. This can be avoided by reaching an agreement with all key stakeholders. The suggested enabling activities along with sub-activities are summarized in Table 2.

Table 2. *Suggested enabling activities with sub-activities.*

| # | Enabling activity | Sub-activities |
|---|---|---|
| 1 | Alignment with management and all stakeholders involved in RM process | 1. all stakeholders to be consulted to agree on the process as well as agree on task ownership, especially in case any action or cooperation is expected from them;<br>2. process owner to be appointed;<br>3. alignment required between solution architects, technical project managers and business project managers;<br>4. responsibilities to be agreed on and recorded centrally in Confluence, using a RACI matrix or similar tool;<br>5. request continuous support from management and project sponsors to enforce the process. |
| 2 | Agree on information management principles within Solutions Architecture and Integration business unit | 1. one dedicated page per project in Confluence, consolidating solution architect and technical project manager pages into one;<br>2. each project page follows the same template structure to cover all relevant areas;<br>3. technical solution always documented in Confluence to be centrally accessible;<br>4. agree on ownership for keeping information up to date;<br>5. establish guidelines for joint lessons learned sharing to allow for continuous improvement among all process stakeholders;<br>6. establish principles for training and new employee on-boarding. |

[3] RACI is a responsibility assignment matrix that stands for Responsible, Accountable, Consulted and Informed.

| | | |
|---|---|---|
| 3 | Describe the RM process | 1. record a high-level description of steps in RM process and relevant tools to be used centrally in Confluence visible to all stakeholders;<br>2. describe the templates to be followed for documenting the solution and the requirements. |
| 4 | Agree on tools and systems used in the RM process | 1. Confluence to be used for gathering all solution related information including storing relevant requirements related and other documentation;<br>2. JIRA to be used for logging all solution requirements at an agreed level of detail and linking requirements to actual development requests when relevant to enhance visibility for all stakeholders in terms of the scope of the solution;<br>3. agreeing on tools for diagram creation;<br>4. making sure all stakeholders have access to these systems. |

The above discussed enabling activities help to set the stage for the establishment of a new process. As indicated above, current RM practices have proven to be sufficient for avoiding failure in launching projects and achieving business goals. There are evidently some parts of the current approach that work well and serve a purpose. Considering the as-is OTS implementation life-cycle (see Figure 8 in subsection 3.3), deducing from interviews and documentation review the main gap areas and factoring in the best practices from the background research, a systematic approach to RM is proposed. Figure 10 illustrates the high-level to-be OTS implementation project life-cycle from RM perspective.

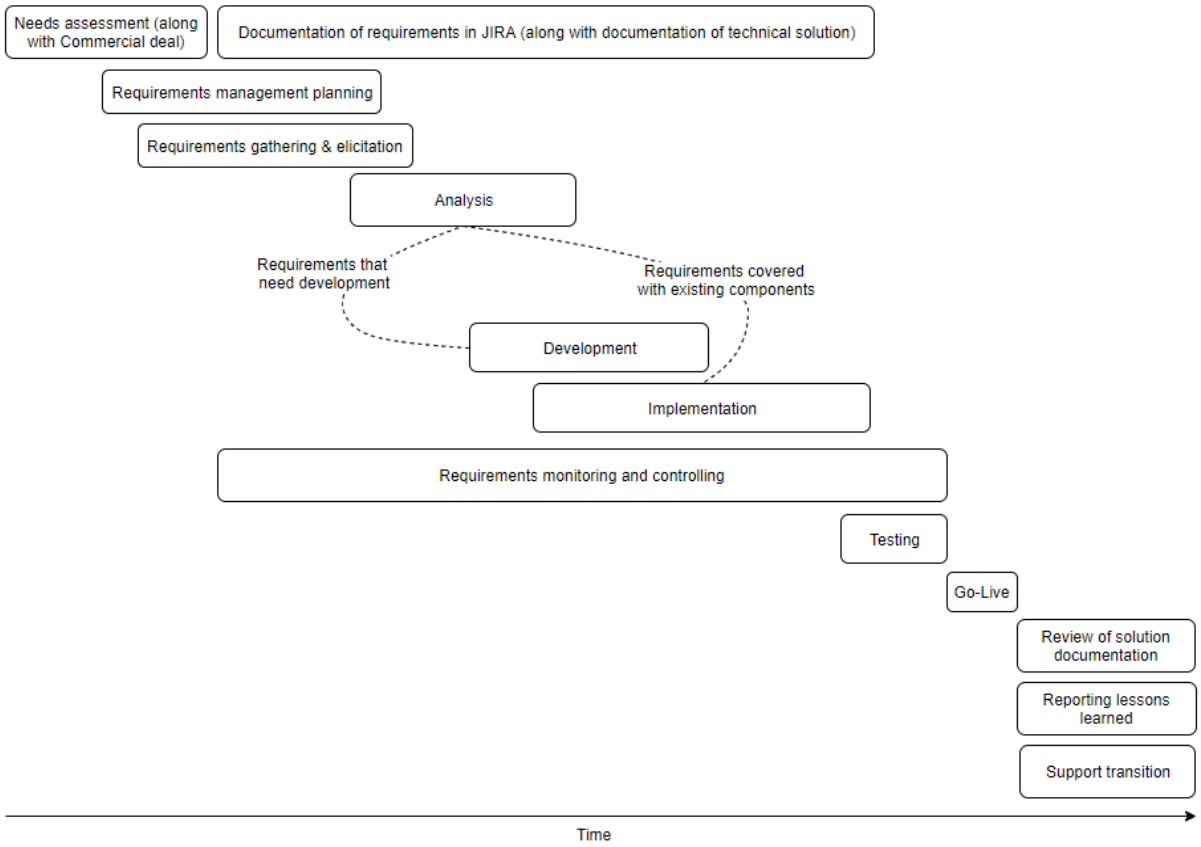


Figure 10. *High-level to-be OTS implementation project life-cycle from RM perspective*.

While most of the core steps of the process are the same as in the as-is model, the to-be model introduces needs assessment, requirements management planning, documentation of

requirements in JIRA, requirements monitoring and controlling, review of solution documentation post launch, reporting lessons learned and support transition. By incorporating the mentioned steps into the RM process, it allows to address the lack of systematic approach, poor documentation and inadequate information management – all which were brought out in the interview and documentation analysis. Establishment of activities such as needs assessment, requirements management planning, requirements monitoring and controlling, reporting lessons learned and support transition is also supported by the best practices in RM domain [10]. Even though there is no clear conflict with the COTS process proposed by Morisio et al. [5] (see Figure 7 in subsection 2.3), the as-is as well as to-be process models position themselves further away from the COTS proposal due to including significantly more company specific activities.

The requirements process starts with needs assessment. The initial scope is being determined in cooperation with commercial and management stakeholders. Specialists to be working with requirements post commercial agreement should be involved as early as possible in this step so that important technical aspect could not be overlooked or overpromised.

Around the same time once the scope of the solution becomes clearer, the requirements management plan is to be created. Key stakeholders per project should be mapped in Confluence and roles and responsibilities determined. Estimations could be given for required meetings and workshops and these estimations should be documented at the same location. Links to other systems and tools should be created, e.g. link to the project space in JIRA. Project scope description should be provided for it to be centrally accessible to all stakeholders in the company who might not have all the context that the key project team does.

Requirements gathering and elicitation is initiated almost in parallel to the needs assessment. Requirements or references to requirements should be gathered in Confluence. The stakeholders in the elicitation process should be outlined in the planning phase. This step involves also holding meetings, calls or workshops with internal stakeholders or customer for gathering requirements.

While the requirements gathering and elicitation is ongoing, the documentation of the gathered information should start. The solution requirements are to be logged in JIRA following a template list of product areas to cover. Each requirement logged in JIRA should include a field for marking it either supported by existing component or requiring development. Each JIRA task should be linkable to product development requests where relevant and have a status field. The list of available statuses is to be established. The JIRA task should also include a field for percentage of completion and a field to mark the owner, i.e. person responsible for the next action (e.g. analysis or implementation).

The analysis of requirements starts with initial gap analysis with the purpose to determine which requirements can be met with OTS components and which require development. Unclear requirements need to be discussed and further analysed with respective product teams and eventual outcome should be logged in JIRA. If a product team files a development task, it should be linked to the solution requirement task in JIRA. Requirements that need development will be passed on for further action to the relevant product teams. The implementation of these requirements will move forward gradually as soon as they are fit for implementation.

In agreement with the project team and customer, the implementation of the solution starts. This can happen in parallel to the requirements analysis and development work. Since the moment the requirements are recorded in JIRA, they should be continuously monitored and controlled.

The logged requirements in JIRA should be reviewed regularly to ensure the details are up to date and that a requirement is still in project scope. In case a requirement has fallen out of scope, it should be marked accordingly under the JIRA record so that this information could reach also product teams who might already be undertaking development to satisfy the requirement.

At the point where the solution implementation is in its final stages or fully completed, testing by Playtech as well as by the customer starts. Among other mutually agreed flows, testing should verify the requirements that are in scope according to JIRA. Testing is followed by the launch of the solution, provided that all relevant approvals from legal, commercial as well as support perspective for go-live are received.

Post-launch or in parallel to it, the review of project and solution documentation should be performed to ensure that the documentation corresponds to the actual solution launched. This is essential from the perspective of the support teams who will need to be able to resolve any issues that might appear in relation to the solution throughout the lifetime of the solution. Along with documentation review, relevant lessons learned would also need to be recorded in Confluence knowledgebase and discussed with relevant stakeholders for continuous improvement. In case changes to any of the used templates or the requirements process itself are needed, this should be highlighted to the process owner or the person responsible for content curation for further clarification with other relevant stakeholders, resulting in eventual adjustment of the process or templates.

As the last step, the ownership for support of the solution is handed over for long-term support. All open issues should be gathered from the stakeholders and documented, including a list of requirements or developments that could not be included in the initial scope for launch. An educated decision must be taken on the missing functionality along with the sign-off from the customer and management to handle those items post-launch either as a second phase of the project or as business-as-usual. The list of activities in the suggested requirements management process is summarized in Table 3.

Table 3. *Steps in the suggested requirements management process*.

| # | Step |
|---|---|
| 1 | Needs assessment |
| 2 | Requirements management planning |
| 3 | Requirements gathering and elicitation |
| 4 | Documentation of requirements in JIRA |
| 5 | Analysis of requirements |
| 6 | Development |
| 7 | Implementation |
| 8 | Requirements monitoring and controlling |
| 9 | Testing |
| 10 | Go-live |
| 11 | Review of project and solution documentation |
| 12 | Reporting lessons learned |
| 13 | Support transition |

The above suggested process requires further cooperation and alignment with the key stakeholders from Solutions Architecture and Integration business unit in order to create the

templates to be used in Confluence and in JIRA as well as determine other details such as relevant JIRA task statuses. The establishment of a new process should be gradual so that progress could be made incrementally and in alignment with other stakeholders in the company. Similarly to the RM process itself, the process establishment should include iterative feedback loops to enable continuous improvement and change of direction when needed.

In terms of task ownership and responsibilities in this process, and referring back to the enabling activity 1 (see Table 2), a responsibility assignment matrix applicable to the proposed process is suggested (see Table 4). This matrix is subject to further discussion and revision among all mentioned stakeholders.

Table 4. *RACI matrix for suggested requirements management process*.

| # | Activity | Licensee | Commercial & Sales | Compliance & Legal | Business Project Management | IMS (Platform) | Product Teams | Infrastructure | Solution Architect | Technical Project Manager | Integration Manager |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | Needs assessment (along with Commercial Deal) | R | A R | C | I | C | C | C | R | | |
| 2 | Requirements management planning | | | | A R | | | | R | I | |
| 3 | Requirements gathering and elicitation | C | C | C | A | C | C | C | R | I | |
| 4 | Documentation of requirements in JIRA (along with documentation of technical solution) | | | | A | | | | R | I R | |
| 5 | Analysis of requirements | C | I | C | A | C | C | C | R | I R | |
| 6 | Development | | | | A | R | R | | C | A | |
| 7 | Solution implementation | | I | | A R | C | C | C | C | A R | C |
| 8 | Requirements monitoring and controlling | | | | A | | | | R | I R | |
| 9 | Testing | | | | A | | | | | R C | C |
| 10 | Go-Live | C | I | I | A R | I | I | I | I | R | I |
| 11 | Review of project and solution documentation | | | | A R | | | | R | R | |
| 12 | Reporting lessons learned | | | | | | | | R | A R | |
| 13 | Support transition | C | C C | C | A R | C | C | C | R | R | C |

It is suggested that similarly to the current practices, the business project manager would be accountable for the progression of the project, including the requirements aspect of it. In addition to the business project manager, the core project team includes also the solution architect and technical project manager who should share most of the responsibility in the requirements life-cycle.

## 4.5 Threats to validity

It is important to consider the threats to validity of the suggested requirements management process to assess the trustworthiness of the results. The aspects discussed in this subsection relate to external validity and reliability of the results. Runeson et al. [28] consider external validity as the level of applicability and generalizability of the findings outside the studied case. They define reliability as the level of dependence of research results on specific researchers.

In terms of external validity, this case study focused on one single business unit that solution architects and technical project managers belong to. As mentioned earlier, these roles are among the key stakeholders in the RM process. It can be argued that to fully map, understand and cover all aspects of the requirements lifecycle, other stakeholders in the process should also be consulted and not omitted. Involving additional stakeholders in the research are likely to uncover additional aspects of the RM practices that are essential to establishing a systematic process. Since this thesis focuses specifically on the part of RM process that this technical solution business unit is involved in, the other departments have not been consulted as part of the case study. Therefore, the identified problem areas, suggested process and responsibilities assignment might not be in line with other departments' understanding and practices. Additional input should be requested from other business units and the process would need to be aligned with relevant stakeholders for further adjustment prior to being put into practice.

Additionally, the fact that the study was conducted only in one company does not allow to generalize the results outside of the case company. The suggested process could not be considered applicable to any other domain or company without actual further research. Other online gambling software providing companies could potentially rely on the presented findings for process establishment, however the organization structure might differ from the current case and therefore include different constraints.

One of the ways to improve the validity of results is to use triangulation which means that the studied phenomenon is being looked at from different angles to allow for grasping the larger picture, and its need is especially evident when the study is based on qualitative data [27]. Even though the principle of triangulation was followed in terms of data sources, the depth of project documentation analysis could have been deeper to potentially provide additional findings. The analysis was largely relying on the interview data but supported with findings from documentation. The problem with documentation was also the fact that it was only possible to analyse documentation that had been made available in the shared Confluence wiki space. There is a chance that relevant documentation had not been shared and in such case, should have been requested directly from the project stakeholders.

In terms of reliability, the way the findings have been presented might be unintentionally biased by author's personal point of view as being part of the same business unit, involved in the work with requirements and also having seen the same process from other business unit's perspective in the past. Efforts have been made to retain objectivity. Nevertheless, it cannot be considered impossible that another researcher would arrive at slightly different results when conducting the same investigation, that especially when involving additional stakeholders and business units in the case study.

# 5 Conclusion

This thesis set out to suggest a requirements management process for off-the-shelf based online gambling solution implementations, applicable in the case company Playtech. The investigation was carried out using case study as this empirical method allows to explore a phenomenon in its context. Requirements management process is highly dependent on the organizational and industry context, therefore an improving case study appeared to be the best approach for addressing the research question. Playtech was chosen as the case company because it offers complex off-the-shelf based online gambling software solutions which often require customizations, configuration changes and development when implemented.

To put forward a proposal for the requirements management process, first a background research was conducted which revealed that public research on software processes in online gambling software companies is inexistent. The literature proved also to be scarce on OTS-based software solutions from the software provider's perspective. Therefore, relevant background information was extracted from the literature relating to requirements management core concepts, issues and best practices in requirements management domain, off-the-shelf based software solutions, and the particularities of Playtech and the gambling industry. Subsequently, current requirements management practices were mapped based on interviews with eleven employees at Playtech who are among the key stakeholders in the requirements managements process in OTS implementation projects. Additionally, documentation review was conducted with the focus on existing processes and responsibilities in the requirements domain as well as on the documentation of ten completed past projects. The results were analysed in the context set by the background information.

The analysis revealed that there is a clear lack of systematic approach to requirements management in OTS implementation projects. Even though current requirements management practices have proven to be sufficient for avoiding failure in launching projects and achieving business goals, there is evidently room for streamlining the process to optimise effort spent and to facilitate work for various stakeholders by introducing and agreeing on an efficient process. Additionally, it became clear that as part of the lack of systematic approach, the documentation on delivered solutions during and post project can be poor. Last but not least, there are no clear information management principles in place to cater for knowledge and lessons learned sharing and proper support transition practices involving the handling of requirements.

The proposed process for requirements management for OTS based online gambling solution implementations addresses the gaps identified during the analysis, relies on best practices from the requirements management domain and considers the particularities of the gambling domain as well as the case company. The new process explicitly incorporates activities such as needs assessment, requirements management planning, requirements monitoring and controlling, reporting lessons learned and support transition – all of which was close to being inexistent in the current practices. Furthermore, a set of enabling activities was determined to prepare ground for effective process establishment, these activities being alignment with management and all stakeholders involved in requirements management process, agreeing on information management principles within Solutions Architecture and Integration business unit, recording the requirements management process as a policy, and agreeing on tools and systems to be used in the requirements management process.

The main value of this thesis lies in the process for requirements management in OTS online gambling solution implementation projects tailored to the conditions and needs of Solutions Architecture and Integration business unit in an online gambling software providing company

Playtech. This process should allow employees involved in the implementation projects approach the requirements management in a systematic way and ensure the continuous success in delivering complex online gambling software solutions. The contribution of this research can also be appreciated when considering the fact that the literature on requirements management in such OTS software implementation solutions from the supplier's perspective is extremely scarce or inexistent, and that especially in the gambling industry. This research could be of significance and reference for other organizations in the industry that offer structurally similar solutions.

As the next step, this research should be shared with the management of the case company to agree on further actions with the aim to start putting the suggested process gradually into practice. Further research could be conducted in the same case company by involving additional stakeholders and thereby bringing on board more perspectives that are likely to provide valuable findings that have remained unreported in this paper. Studying project management process as a whole, which requirements management is considered a part of, would very likely unfold additional gap areas in the process to be addressed. Further research is required to render the results generalisable to other software providers in the gambling industry or to companies in other industries providing structurally similar software solutions.

# 6 References


[1] K. Pohl, *Requirements Engineering: Fundamentals, Principles, and Techniques*. Berlin Heidelberg: Springer-Verlag Berlin Heidelberg, 2010.

[2] C. M. Lewandowski, "The CHAOS Report," 1994.

[3] PMI, "Requirements Management: A Core Competency for Project and Program Success," *PMI's Pulse Prof.*, pp. 1–20, 2014.

[4] S. Hansen, N. Berente, and K. Lyytinen, "Requirements in the 21st century: Current practice and emerging trends," in *Lyytinen K., Loucopoulos P., Mylopoulos J., Robinson B. (eds) Design Requirements Engineering: A Ten-Year Perspective. Lecture Notes in Business Information Processing*, 2009, vol. 14 LNBIP, pp. 44–87.

[5] M. Morisio, C. B. Seaman, V. R. Basili, A. T. Parra, S. E. Kraft, and S. E. Condon, "COTS-based software development: Processes and open issues," in *Journal of Systems and Software*, 2002.

[6] M. Weber and J. Weisbrod, "Requirements engineering in automotive development-experiences and challenges," *Proc. IEEE Int. Conf. Requir. Eng.*, 2002.

[7] M. Arias, A. Buccella, and A. Cechich, "A Framework for Managing Requirements of Software Product Lines," *Electron. Notes Theor. Comput. Sci.*, vol. 339, pp. 5–20, 2018.

[8] M. Daneva, "Lessons learnt from five years of experience in ERP requirements engineering," in *Proceedings of the IEEE International Conference on Requirements Engineering*, 2003.

[9] A. Parra, V. Basili, and S. Burke, "The Package-Based Development Process in the Flight Dynamics Division," pp. 1–24, 1998.

[10] Project Management Institute, *Requirements Management: A Practice Guide*. Pennsylvania: Project Management Institute, 2016.

[11] K. Wiegers and J. Beatty, *Software Requirements (3rd Edition) (Developer Best Practices)*. 2013.

[12] B. H. C. Cheng and J. M. Atlee, "Current and future research directions in requirements engineering," in *Lecture Notes in Business Information Processing. Design Requirements Engineering: A Ten-Year Perspective.*, 2009, vol. 14 LNBIP, pp. 11–43.

[13] D. Pandey, U. Suman, and A. K. Ramani, "An Effective Requirement Engineering Process Model for Software Development and Requirements Management," *2010 Int. Conf. Adv. Recent Technol. Commun. Comput.*, pp. 287–291, 2010.

[14] B. R. Eds, "Design Requirements Engineering: A Ten-Year Perspective," 2009, vol. 14, no. June.

[15] M. Špundak, "Mixed Agile/Traditional Project Management Methodology – Reality or Illusion?," *Procedia - Soc. Behav. Sci.*, vol. 119, no. March 2014, pp. 939–948, 2014.

[16] B. Ramesh, L. Cao, and R. Baskerville, “Agile requirements engineering practices and challenges: an empirical study,” *Inf. Syst. J.*, vol. 20, no. 5, pp. 449–480, 2010.

[17] D. Firesmith, “Common requirements problems, their negative consequences, and the industry best practices to help solve them,” *J. Object Technol.*, vol. 6, no. 1, pp. 17–33, 2007.

[18] Z. Ahmad, M. Hussain, A. Rehman, U. Qamar, and M. Afzal, “Impact minimization of requirements change in software project through requirements classification,” *Proc. 9th Int. Conf. Ubiquitous Inf. Manag. Commun. - IMCOM ’15*, pp. 1–5, 2015.

[19] J. C. Maxwell, A. I. Anton, and P. Swire, “Managing changing compliance requirements by predicting regulatory evolution,” in *2012 20th IEEE International Requirements Engineering Conference, RE 2012 - Proceedings*, 2012, pp. 101–110.

[20] V. Perrone, “A wish list for requirements engineering for COTS-based information systems,” in *COTS-Based Software Systems*, 2004, pp. 146–158.

[21] S. Yacoub, A. Mili, C. Kaveri, and M. Dehlin, “A Hierarchy of COTS Certification Criteria,” in *Software Product Lines: Experience and Research Directions*, 2000, pp. 397–412.

[22] D. Carney, “Assembling Large Systems from COTS Components : Opportunities , Cautions , and Complexities,” *SEI Monogr. Use Commer. Softw. Gov. Syst.*, no. June, 1997.

[23] Gamblingsites.org, “History of Gambling Sites - When and Who Started Online Gambling.” [Online]. Available: https://www.gamblingsites.org/history/. [Accessed: 23-Oct-2018].

[24] Playtech, “Company Overview Playtech.” [Online]. Available: https://www.playtech.com/about-us. [Accessed: 23-Oct-2018].

[25] R. K. Yin, *Case Study Research : Design and Methods (3rd edition)*. SAGE Publications, 2003.

[26] C. Robson, *Real world research, 2nd edition*. Oxford: Blackwell Publishing, 2002.

[27] P. Runeson and M. Höst, “Guidelines for conducting and reporting case study research in software engineering,” *Empir. Softw. Eng.*, vol. 14, no. 2, pp. 131–164, 2009.

[28] P. Runeson, M. Höst, A. Rainer, and B. Regnell, *Case Study Research in Software Engineering: Guidelines and Examples*. 2012.

[29] M. Niazi, D. Wilson, and D. Zowghi, “Critical success factors for software process improvement implementation: An empirical study,” *Softw. Process Improv. Pract.*, vol. 11, no. 2, pp. 193–211, 2006.

# Appendices

## I Interview questions

1. How many years of experience do you have in your current role/working with requirements?
2. How would you define *requirements process* in the context of solution implementation projects in Playtech? What does this process consist of?
    a. Please describe the process for requirements management (i.e. requirements development and requirements management) within an average project you have worked on.
    b. What are the main steps in the requirements management (RM) process?
    c. What are the main sources of requirements related information?
    d. How are the requirements documented?
    e. Are the requirements baselined?
    f. What kind of analysis is conducted in the process? What is the output of the analysis?
    g. How are requirements being communicated?
    h. Is there any specific grouping of requirements that you use? (e.g. per product areas, any other logical classification across projects, regulatory, hardware, instance setup related (CCP, languages, currencies, etc.)) How are requirements linked to the relevant components?
    i. How much reuse of past requirements is there in the RM process?
3. To what extent is the validation of the requirements and getting commitment from the relevant stakeholders part of the process?
4. How are changes in requirements managed?
5. How is (bidirectional) traceability of requirements ensured?
6. How are requirements prioritized?
7. Who are the main stakeholders you interact and consult with in the requirements process (both internal and external)? Which units and roles are involved in the process?
8. How is consistency between requirements and project plans ensured?
9. Are there any other factors that have an effect on the RM process that have not been mentioned so far?
10. Which tools do you use in the RM process and for which purposes? (e.g. Confluence, JIRA, email, etc.)
11. What steps in the RM process are generally going well? Where do you experience problems most often?
12. Is there a certain policy/plan to be followed when it comes to RM? Is there a defined process in place? How is knowledge being transferred to a starting employee?
13. How is the process monitored and controlled?
14. What kind of training and/or support is provided to people who deal with requirements?
15. How do you feel about knowledge sharing/lessons learned sharing in the RM domain in the company? Is it important? Is it being done?
16. What could be improved in the requirements process?

## II Licence

**Non-exclusive licence to reproduce thesis**

I, **Mari Vaiksaar**,

1. herewith grant the University of Tartu a free permit (non-exclusive licence) to

reproduce, for the purpose of preservation, including for the purpose of preservation in the DSpace digital archives until expiry of the term of validity of the copyright

**Requirements Management in Off-The-Shelf Software Implementation Projects: A Case Study in Playtech**,

supervised by Fredrik Payman Milani, PhD.

2. Making the thesis available to the public is not allowed.

3. I am aware of the fact that the author retains the right referred to in point 1.

4. This is to certify that granting the non-exclusive licence does not infringe the intellectual property rights or rights arising from the Personal Data Protection Act.

Tartu, **9.01.2019**