Expertise
and insight
for the future

Sofia Biström

Developing the Roles and Responsibilities for Continuous Software Service Provision

Metropolia University of Applied Sciences

Bachelor of Engineering

Industrial Management

Bachelor's Thesis

28 January 2019

Author Title Number of Pages Date	Sofia Biström Developing the Roles and Responsibilities of Continuous Software Service Provision 57 pages + 13 appendices 28 January 2019
Degree	Bachelor of Engineering
Degree Programme	Industrial management
Professional Major	International ICT-business
Instructors	Customer Support Manager of Case Company Nina Hellman, Head of Industrial Management, Senior Lecturer Sonja Holappa, Senior Lecturer

The objective of this thesis is to propose clear roles and responsibilities related to the operations of providing a continuous software service. The case company is going to go through an audit to update its current quality certificate to the ISO 9001 and the ISO 14001 certificate, due to which the company is seeking to develop selected process descriptions.

The study included seven different stages. Based on the findings of the current state analysis, selected areas were defined for development. These development areas to be solved in the proposal were 1) lack of structure of the operations, 2) uneven distribution of tasks, and 3) lack of transparency regarding change management. The case company needs and requirements identified both by data collection and by working in the company were carefully included and assessed in the process of building the proposal.

The proposal created as a result of this thesis included an organizational aspect in the form of creating a new team called Operations. The roles and responsibilities of continuous software service provision were defined based on centralizing the main responsibility of continuous service provision to a new Operations team. The tasks and their responsibilities were visualized in a RACI table. Documenting procedures were defined for both event management operations and for the SQL data content changes.

This proposal includes a structured and transparent way of conducting the continuous service provision operations. It includes both increased consistency in operations and a manageable way for performing the different tasks.

Keywords	Continuous service, RACI, Documentation, Responsibility

Tekijä Otsikko Sivumäärä Aika	Sofia Biström Jatkuvan palveluntuottamisen vastuuttaminen 57 sivua + 13 liitettä 28.1.2019
Tutkinto	Insinööri (AMK)
Tutkinto-ohjelma	Tuotantotalous
Ammatillinen pääaine	Kansainvälinen ICT-liiketoiminta
Ohjaajat	Yrityksen Asiakaspalvelupäällikkö Nina Hellman, Tuotantotalouden tutkintovastaava, Lehtori Sonja Holappa, Lehtori

Tämän opinnäytetyön tavoitteena oli esittää ehdotus jatkuvan palveluntuottamisen vastuuttamisesta. Kohdeyrityksen suunnitelmana on päivittää nykyinen laatusertifikaatti ISO 9001- sekä ISO 14001- laatusertifikaattiin. Tämän johdosta yrityksen tavoitteena on kehittää valikoitujen prosessien kuvauksia.

Tutkimus on suoritettu seitsemässä eri osassa. Kehityskohdat, jotka identifioitiin nykytilaanalyysista, olivat seuraavat: 1) toiminnan struktuurin puutteellisuus, 2) epätasaisesti jakautuneet tehtävät sekä 3) läpinäkyvyyden puutteellisuus muutosten hallintaan liittyen.

Kohdeyrityksen tarpeet ja vaatimukset identifioitiin sekä tiedonkeruun että omien kohdeyrityksessä työskentelyn perusteella hankittujen kokemusten perusteella. Identifioidut tarpeet ja vaatimukset otettiin tarkoin huomioon lopputuotosta luodessa.

Insinöörityön lopputuotos sisälsi organisatorisen aspektin uuden Operations-tiimin pystyttämisenä. Jatkuvan palveluntuottamisen vastuiden määrittely luotiin keskittämällä päävastuu uudelle Operations-tiimille. Jatkuvan palveluntuottamisen tehtävät ja niiden vastuut kuvattiin RACI-taulukossa. Dokumentointikäytännöt määritettiin niin herätteiden hallintaa, sekä SQL-tietosisältömuutoksia varten.

Lopputuotos sisältää kuvauksen suunnitellusta sekä läpinäkyvästä tavasta suorittaa jatkuvaa palveluntuottamista. Se mahdollistaa myös systemaattisen sekä hallitun tavan suorittaa jatkuvaan palveluntuottamiseen liittyviä tehtäviä.

Avainsanat	Jatkuva palveluntuottaminen, RACI, Dokumentointi, Vastuuttaminen

Contents

Appendices

List of Abbreviations

SaaS	Software as a Service. Software that is provided as a service in a cloud.
ITIL	Information Technology Infrastructure Library. Best practice framework on Information technology service management
CTO	Chief Technology officer
RACI	Responsibility assignment matrix. Describes the division of Responsible, Accountable, Consulted and Informed roles.
CSA	Current state analysis
SLA	Service level agreement

1 Introduction

For an IT service to be delivered fulfilling quality expectations, the operations must be defined and well structured. This may be more challenging when there are sub-contractors involved. When the solution provided is in the form of Software as a Service, it is important to ensure the appropriate operability of the software. This is preferably done by preventing harmful incidents from occurring. One of the important things is to have responsibilities clearly defined and communicated to all parties involved. This is a critical requirement for the different tasks to be performed as expected.

1.1 Business Context

The case company of this study is a Finnish group of companies that provides a wide range of different services, of which one is software. Founded in the 20th century, the company has many years of expertise in its field and is today operating globally. The company employs over 900 professionals and had a yearly revenue of around 70 million euros in 2017.

This study will be concentrating on the software department and the operations related to providing their Software as a Service solution. The department itself consists of multiple different groups including key account managers, both domestic and international sales, software development, product management and an innovation team.

1.2 Business Challenge, Objective and Outcome

The company seeks to be able to continue competing with the rapidly growing quality expectations, due to which a decision has been made to update the current quality certificate to the ISO 9001 certificate and at the same time certify for the ISO 14001 environmental certificate. At the time of writing this thesis (Autumn 2018), this project is already ongoing. Due to this all of the processes will go through an audit to determine their compliance.

The software department is seeking to update selected process descriptions and related documentation before the audit period. One of these descriptions is the one describing the roles and responsibilities of the operations enabling the continuous software service provision. These operations include actions that are required to continuously provide the service with the appropriate operability, as stated in the service level agreement. In the case company the definition of these continuous service operations can be divided into maintenance and monitoring tasks. The maintenance tasks include the incident and change management process and the software updates. The monitoring tasks include the software monitoring undertaken to detect functional incidents.

In the process of reviewing current documentation for the audit, it was observed that the current division of roles and responsibilities related to the continuous service provision are not defined clearly enough. The department therefor seeks to add increased structure to the roles and responsibilities.

The business challenge of this thesis is therefore focused on the operations that are conducted to continuously enable the appropriate operability of the software. **The objective of the study is to propose clear roles and responsibilities related to the operations of providing a Continuous Software Service.**

The outcome of this study is to present a proposal on how the roles and responsibilities of the continuous software service operations could be defined to add clarity to the division of tasks. The result of this thesis is vital from the business context point of view, since it will be included in the audit that the company is going to go through during spring 2019. Due to this, it is also important for this project to be completed in accordance to the schedule.

1.3 Thesis Outline

This thesis is restricted to developing the roles and responsibilities related to the operations of providing a continuous software service. The study will examine the current operating way and propose a renewed structure of the responsibilities. The study is also limited to developing the responsibilities of continuous service provision of one specific software, later referred to as Software X, provided by the case company.

This thesis was conducted with methods including researching the current process in place and analysis of it, of interviews with employees of the case company including product owner, software development and additional key individuals regarding the provision of the software service. Including the company professionals by interviewing and taking their opinions into account, the study ensures that the proposal will be suitable for the company, and serve its needs well.

The thesis contains 7 sections. The first section consists of an introduction of the thesis and the company which it was carried out for. The second section presents the methods used to conduct the study. The thesis then continues with section three, presenting a detailed analysis of the case company's current division of roles and responsibilities. The fourth section introduces you to available literature and the conceptual framework of the study. The outcome of the study as well as a description of how this proposal was built is being presented in section five, followed by section six that includes the final validation of the outcome. Section seven, which is the final section, consists of a summary and an overall evaluation of the project.

1.4 Key Concepts

The following terms are key concepts in this study:

Continuous service	A service that is continuously provided in accordance to the design standards and the service promise
Service provider	A company delivering a service
Sub-provider	Provider delivering a service for another company that is delivering a service
Application	Computer program/software that has a purpose
User interface	The view that is visible for the users of a software
Showstopper	A critical bug in a software that has an immediate effect on the availability of the product and customer satisfaction

2 Method and Material

This section introduces the methods and materials used to conduct the study. These are presented in three parts including research design, project plan, the collection of data used for the study and the analysis of it. It will first present the research design giving an overview of how the study was carried out. The detailed planning and scheduling of the tasks are presented in the project plan in the second part. The final and third part of this section presents the methods of data collection and analysis that was used to carry out this study.

It is worth mentioning, that as an employee of the company I as the author, have used the knowledge and insight gained from working in the company to conduct this study. These insights have also been taken into account in the proposal building stage.

2.1 Research Design

This study is conducted in 5 different stages, as shown in the diagram below. The diagram visualizes which data sources were used for the different stages. Also it visualizes which outcomes were reached from each stage.

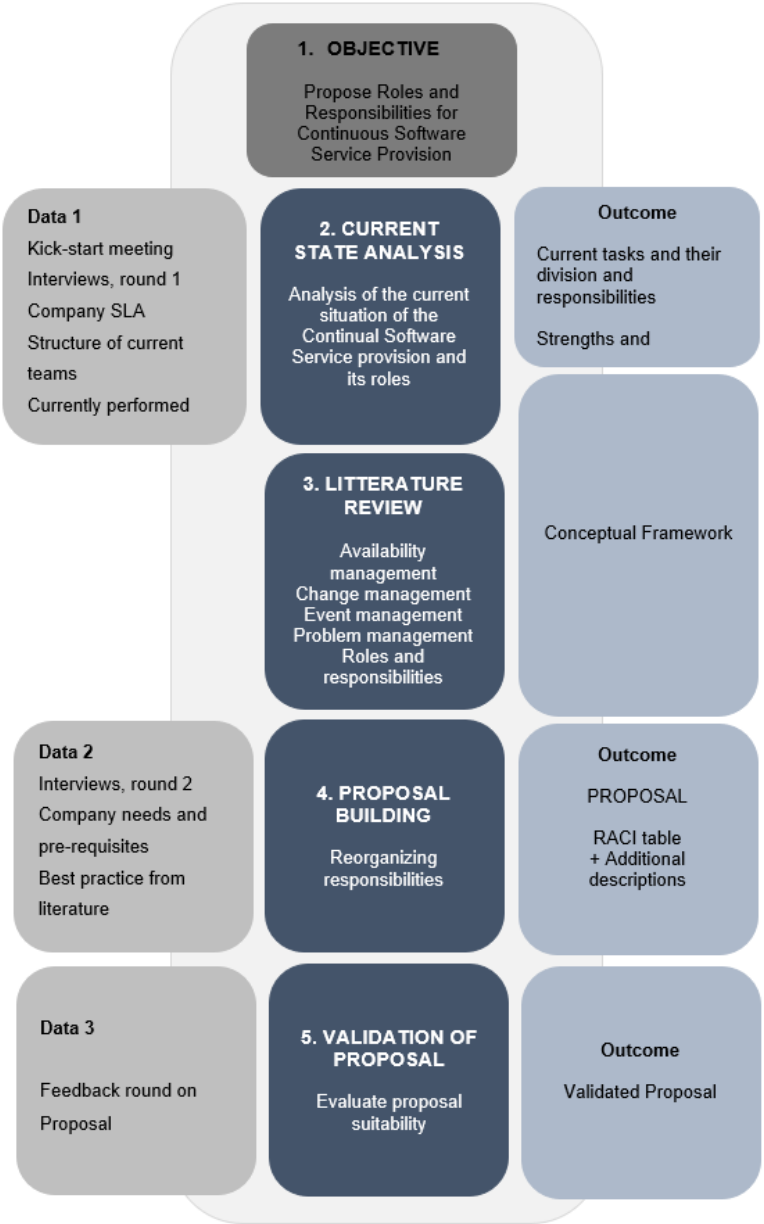


Figure 1. Research design

As shown in Figure 1 above, the study starts by defining the objective of the study, meaning the challenge that is supposed to be solved. As stated, the objective of this study is to *"Propose roles and responsibilities for Continuous Software Service Provision*". In the next stage, after defining the objective and the outcomes of the business challenge, the study focuses on analysis of the current state of the operating way. Data 1 is used for this stage and the stage results in two outcomes. One of the outcomes is to map the currently performed tasks and their responsibilities. The other outcome is to identify the main strengths and weaknesses of the current way of operating.

After evaluating the current state the study continues with researching available literature. Available literature used for this study includes different sections in ITIL, Information technology infrastructure library, for instance availability, change, event and problem management. Also available knowledge regarding organizing of roles and responsibilities will be looked into. The outcome of this stage is the conceptual framework of this study.

The fourth stage of the study is the proposal building stage. This stage draws information from Data 2, and the outcome of this study is the proposal in the form of a RACI table. After this the proposal is validated according to feedback received from Data 3. Finally, the proposal is modified if required.

2.2 Project Plan and Schedule

This project is carried out by a student graduating from the study programme of Industrial engineering at Helsinki Metropolia University of Applied Sciences. This project forms the Bachelor's Thesis of this student. For the final outcome to be reached by the deadline opted, a detailed project plan was created. The project was carried out from the end of August 2018 until the end of December 2018.

The project plan consisted of bigger sections each including multiple different actions. Figure 2, presenting the full schedule of the project, visualizes the duration of each section, and that the completion of each of these formed an individual outcome, a sort of milestone.

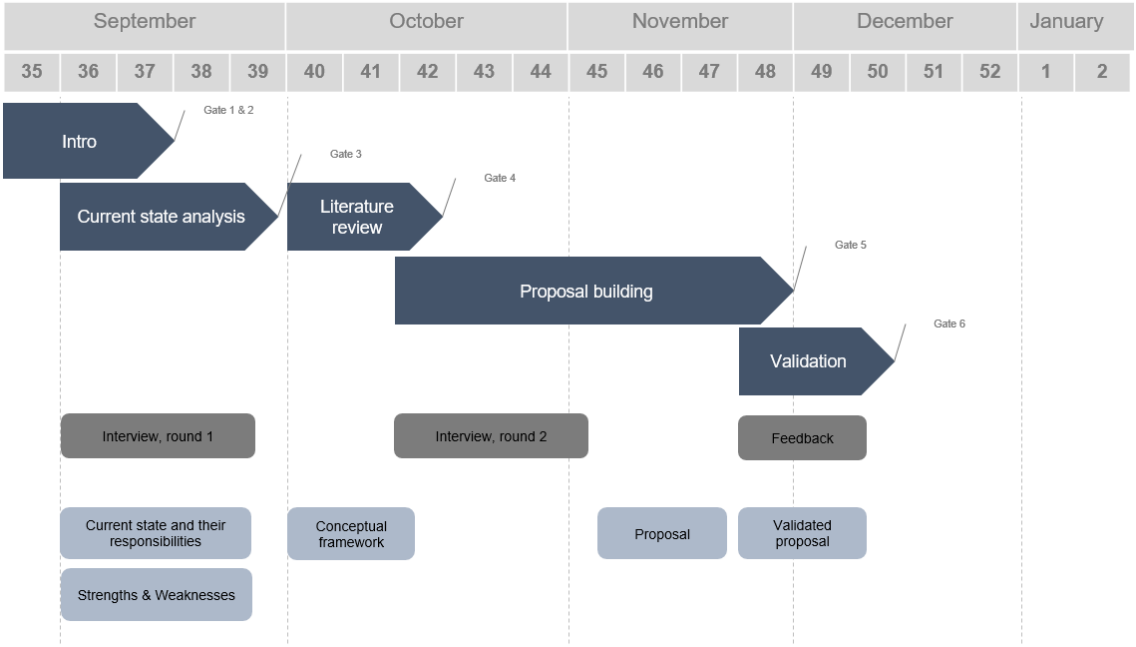


Figure 2. Project schedule

The full project plan is available in Appendix 1. The different data sources used along the study are presented in detail below.

2.3 Data Collection and Analysis

The data used for this study was collected from multiple different types of data sources to ensure the reliability and trustworthiness of this study. The data rounds that were carried out are presented in Table 1 below.

Table 1. Details of interviews, workshops and discussions, in Data 1-3. *(based on: Aittola 2015)*

	Participants / role	Data type	Topic, description	Date, length	Documented as
	Data 1, for the Current state analysis (Section 3)				
1	Group interview: Product manager, Customer support manager, Quality director	Face to face workshop	Workshop on current maintenance operations. Go-through of current roles and responsibilities regarding Service continuity operations.	5 Sep, 2 hours	Field notes
2	Interview 2: Product development director and CTO	Face to face Interview	Service continuity actions performed in the software development.	7 Sep, 1 hour	Field notes
3	Interview 3: Expert	Face to face Interview	How errors are escalated to incidents in the software department	11 Sep, 1 hour	Field notes
4	Interview 4: Experts (2)	Face to face Interview	Databases in the service, Service continuity actions regarding databases	11 Sep, 1 hour	Field notes
5	Interview 5: Head of IT	Face to Face Interview	Sub-contractor contract	12 Sep, 1 hour	Field notes
	Data 2, for Proposal building (Section 5)				
6	Head of IT & CTO	Face to Face meeting	Discussion about server provider and what the responsibilities are regarding the servers	19 Oct, 1 hour	Minutes of meeting
7	Head of Operations team	Face to Face meeting	Organizing of the Operations team. Documentation procedures	21 Nov, 1hour	Field notes
	Data 3, from Validation (Section 6)				
8	Feedback discussion	Face to Face meeting	Validating the operability and suitability of the proposal.	27 Nov, 1,5 hour	Field notes

As seen in the table presented above, the data for this study was collected in three rounds. The first round, Data 1, was collected for the current state analysis. This was done to get as good of a view as possible of the starting state. It also enabled the collection of the opinions of the employees, to then be able to approach the project from the right angle.

The second round of data collection, Data 2, was collected to gather further information about the departmental needs related to specific areas and also current prerequisites for developing the operations.

The data from round one was used to get a clear view of the starting stage of the project. With the help of Data 1, also the unclear parts regarding the division of responsibilities in between the parties were identified. Data 2 was used to map what sort of prerequisites the department currently had, and to identify specific parts especially critical to the success of the implementation of the proposed solution. This was done, so that the proposal presented as an end result would be as suitable as possible.

In this study the interviews formed the primary method of data collection. This is due to the fact that it was very important to create a process as suitable as possible for the department and based on its employees capabilities. The first and second round of interviews were conducted as face-to-face interviews with predefined questions to discuss. The interviews and meetings were recorded as field notes (see Appendices 2-8). The third round was conducted as a group meeting, facilitating a possibility for the stakeholders to provide feedback on the proposed solution. Notes are available in Appendix 9.

The method used for developing the proposal was a collaborative approach in the form of workshops. Workshops and additional meetings were orchestrated in which the proposal was formed and defined in close collaboration with the employees of the company.

The internal documents that were used for the current state analysis are presented in Table 2.

Table 2. Internal documents used in the current state analysis, Data 1 (Lapin 2014).

	Name of document	Number of pages	Description
A	Case company's standard SLA	3 pages	Service Level agreement
B	Case company's General Terms and Conditions	3 pages	General terms and conditions of the software service
C	Sub-contractor RACI model	1 table	Responsibilities based on the contract

As seen in Table 2 above, the study also included internal case company documents that were reviewed. The documents were mainly used for the current state analysis, since they included relevant information to get a deep understanding of the overall picture. Thematic content analysis was used to analyze the data.

Most of the data analysis was used for the current state analysis. The findings of the current state analysis are discussed in Section 3 below.

3 Current State Analysis

This section presents the results from the current state analysis of the operations enabling the continuous provision of the software solution, and their related roles and responsibilities. The current state analysis includes 6 different headings. First there is an overview of the current state analysis explaining how the analysis was carried out. This is followed by a description of the structure of different actors related to the operations of the continuous provision of the solution. After this the documentation process related to this area is introduced. The current state analysis then continues with presenting the current operations of continuous service provision. This is followed by a presentation of the findings from the analysis. The chapter will then ends with summarizing the key findings gained from the current state analysis.

3.1 Overview of CSA Stage

The current state analysis was conducted in three steps. The first step was to gather information about how the different parties were organized regarding the continuous service provision. It was relevant to understand the company's current operating model including the division of tasks between the different parties. A workshop concerning the topic was organized, to view current documentation and to draw a process chart describing the current way of operating.

The second stage included interviews with key individuals related to the continuous service provision. These interviews consisted of pre-defined questions (Appendices 2-6) created based on the knowledge gained in the first part of the current state analysis. The focus of these interviews was on gathering information about the actions that were currently conducted and also giving the people involved the opportunity to share their views and opinions of the current way of operating.

In the last stage of the current state analysis, all of the information gathered was summed up to identify the development needs regarding the organization of the continuous software service provision. In the analysis stage also relevant company documents were viewed to deepen the understanding.

The information gathered in the current state analysis stage is discussed in the following chapters 3.2 – 3.4.

3.2 Structure of the Current Organization

Performance of the operations of the continuous service provision are currently being divided between the product management and the product development team. To fully understand the difficulties of the current operating model, it is mandatory to describe the structure of these teams on a high level. Since the case company also has sub-contractors, these are also mentioned, and the roles of these sub-contractors are explained.

The organization involved in the continuous service provision consists of around 40 people, excluding the service providers. The customer support center is included in these 40 people.

How the actions of these actors link together is explained together with describing the incident and change management process in chapter 3.4.1.

3.2.1 Product Management

The product management team includes sub-teams that are responsible for different tasks. The head of product management is the product manager. The customer support center is a division of the product management team. The task description of the customer support center includes handling of customer support in the form of calls, emails and chat. The customer support center is also responsible for informing customers about incidents if these occur.

According to the product manager, another division of the product management team is the section specializing in data management in the SQL database. These tasks in the SQL database are limited to data content, excluding tasks related to the structure of the SQL database. Currently the task description of this section consists of a somewhat wide range of different tasks. According to the specialists interviewed from this division, these tasks include change management related to the data content, different sorts of project work, big volume data input, creation of integrations and monitoring of logs.

3.2.2 Product Development

According to the product development director and the chief technology officer, later referred to as CTO, the product development team consist of a range of developers and some more specific task description. These roles include release manager, who is handling the new software version releases, the CTO and the database manager. The chief technology officer retains the full understanding of the technology behind the software, including integrations. The tasks that the database manager is responsibility for include ensuring the quality of the changes to be made in the database and the creation of the scripts for updating the new software version release including these changes. Monitoring the capabilities of the databases and servers is conducted by the release manager, the CTO and the database manager.

According to the CTO, most of the currently performed maintenance and monitoring tasks, for ensuring the continuous operability of the system, are currently being handled by individuals from the product development team.

The company has also contracted sub-contractors to provide server and domain hosting. Information regarding this can be found in Appendix 7.

3.3 Documentation in the Software Department

Two different ticketing systems are currently in use in the software department. These are both necessary and are being used by different functions. According to information gathered in the first workshop regarding the current maintenance operations, these two systems are the ticketing system, which is used by the customer support center, and the change management software, that is mainly used by the software development team.

The ticketing system being used by the customer support center is used to log contact from customers. This excludes contacts by a phone call during which the reason for the call was resolved. The contacts are logged as tickets, and each action related to a specific contact is recorded until the ticket is closed.

The change management tool used by the product development team serves the need of tracking change needs. It also enables following up on which change needs are currently in progress, and which ones have already been implemented.

3.4 Continuous Service Provision Operations

This section introduces the different operations that are currently being conducted to enable the continuous service provision. These operations consist of different tasks performed to ensure that the software is performing accordingly. These operations can be divided into maintenance and monitoring tasks. The maintenance tasks include incident and change management operations and the systematic software updates. The monitoring tasks consist of operations conducted to detect and prevent errors that may cause functional disruptions.

3.4.1 Incident and Change Management

According to the product manager, incident and change management are at the case company considered a parts of the operations enabling the continuous service provision. Due to this, it is relevant to describe these operations on a high level. Explaining this process also describes the linkage between the product management and development team as well as the sub-contractors.

Process description

There is an incident management process in place in the case company. A process chart to demonstrate the current state on a high level was created as a part of this thesis. The process flow was visualized during the workshop regarding current maintenance operations. The process chart visualizes the division of tasks between the product management and software development teams. As previously mentioned, the customer support center is a division of the product management team. Due to the division of tasks, the customer support center is presented separately. The process description is based on the information attained from the product manager and the customer support manager.

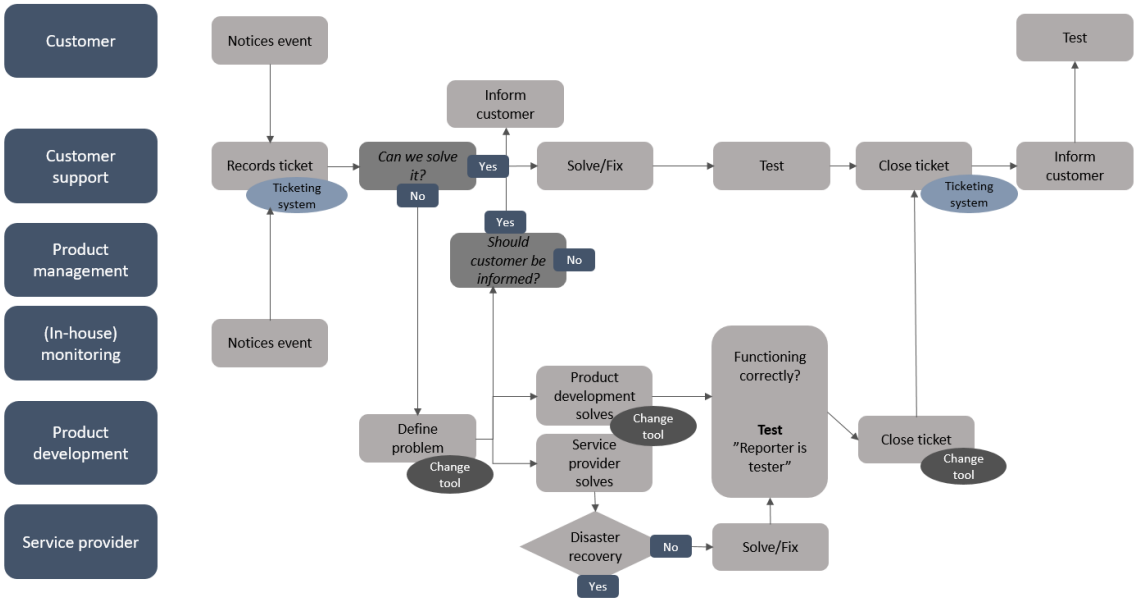


Figure 3. Incident management process roles

As seen in Figure 3, the incident management process is triggered by an incident being detected either in-house or by a customer. Once the customer support service is contacted, a ticket is logged in the ticketing system. The customer support center then determines whether they are able to solve the incident or if it needs to be recorded as a change need for the product development team to handle. In case customer support is able to solve the incident, then they provide the resolution, test the resolution, and close the ticket. After this they inform the customer about the resolution.

The other possibility is that the customer support center identifies that knowledge of the product development team is required for solving the incident. In this case the customer support center records it as a bug in the change management system. The ticket in the ticketing system remains in progress and is attached with a note including the ticket number of the change management ticket.

The bug is then managed based on its priority. Based on root cause analysis conducted by the product development team, the resolution is either provided by the product development itself or by the sub-contractor. After the root cause analysis, the product management team is consulted whether the severity of the incident requires informing end users.

The change management ticket status is updated throughout the process, and the customer support is notified. After the change has been tested the ticket in the change management system is closed. After this the customer support center is able to close the ticket in the ticketing system and contact the reporter about the solution.

Exceptions from the Standard Operating Model

According to the customer support manager this previously described standard incident model is not applied in all cases. These cases occur when someone from the department notices an error in the system. If this person identifies that the error can quickly be solved through the SQL database, then instead of following standard incident reporting process, the SQL team is contacted. The error is then fixed by the SQL team, but the actual root cause remains. According to customer support manager none of the actions done in the SQL database are recorded, also including these sorts of error related changes.

The major incidents also count as incidents that are not handled in accordance with the standard incident model. These are handled based on the disaster recovery plan, which the company already has in place.

3.4.2 Software Updates

According to the product development director, software updates are conducted on a monthly schedule, 11 times a year. The definition of the software update procedures are defined in the area of product development, and are not described here in greater detail.

The release manager, appointed for this task, has the responsibility of the monthly updates. As previously mentioned the release manager is a part of the software development team. This results in also the update task being a part of the software development team's task list. This task includes both the version update and the provision of information including new features and fixes that are communicated towards the stakeholders in touch with a new release.

3.4.3 Service Monitoring

The service monitoring that is currently being done include both automatic and manual actions. According to the chief technology officer, these actions are mostly conducted by the product development team. At the time being, performance monitoring of the system is not conducted systematically.

Service Availability

Automatic monitoring of the service availability is being conducted. The software used for monitoring is a robot, which is ensuring that the availability based on the percentage stated in the service level agreement is reached each month. The monthly service availability promised is 97%.

In case the robot detects that the connection to the domain is down, meaning that the service is unavailable, it will send out an automatic alert. These alerts are sent to both the collaborative chat-tool used by the product development and to pre-defined email addresses. These email addresses belong to defined individuals from both the software development and product management team. After receiving the alert someone of these individuals carries out an analysis of what activated the alert. Currently there is no specified person responsible for taking action in case this alert is received. After the analysis has been done, the corrective measures are taken based on the root cause. In case the root cause id defined to be in-house, measures are taken based on processes defined in the area of product development. In case the root cause is defined to be the responsibility of either one of the sub-contractors, the handling of the problem is directed to them.

Due to the lack of a responsible person for conducting the analysis, the responsibility of communicating towards the sub-contractors in a case like this is also missing.

Error Logs

According to the chief technology officer, monitoring of error logs is conducted on a weekly basis. This is done to detect error peaks. Error peaks would, in case they are detected, indicate the existence of an underlying fault in either the software application or in some of its back end systems.

Operating System on Server Level

The function of the operating system on server level is currently monitored by the subcontractor. Automatic alerts are being sent based on changes detected in the user interface of the operative system. Each of these alerts are being analyzed by specialists from the case company product development team. The analysis is mandatory to figure out the relevance of the messages. There is currently no individual who has been given the responsibility of analyzing these alerts when they occur. An individual in the product development team is performing this task each week.

Integrations

According to the CTO, there are some customer cases where integrations are made between the software provided by the case company and additional services being used by the customer. Integrations are not included in the standard software provided, but the function of these is crucial for the solution they are attached to. The performance of these integrations is currently monitored on a weekly basis. Automatic error alerts regarding the integrations are being received and analyzed manually. What makes the monitoring somewhat difficult is the fact that the automatic messages are not received from all integrations made. Since there is currently not a standard way of creating integrations, the alerts are received based on different criteria, and are also received from successful ones. The analysis is done by the software development team, but currently no person is defined responsible. One individual is performing the analysis each week.

3.5 Findings from the Current State of Continuous Service Provision Operations

This section presents the findings from the current state analysis. Figure 4 presents the logic of the current state analysis. The data that was collected in the current state analysis was concluded and presented in headings 3.2 - 3.4. The findings extracted from this data are presented in this chapter, with the key findings forming the conclusion of this chapter. These key findings then determine the key elements of the proposal to be created as an outcome of this study.

Figure 4. Logic of Current State Analysis

The findings of the currents state analyses are visualized in Figure 5. It shows how the responsibilities related to the different tasks of continuous service operations are currently divided. The tasks have been organized based on different areas in ITIL. The tasks are presented on individual specialist level to add clarity at this point. The important aspects to pay attention to are marked with yellow in Figure 5. Some tasks have been identified to be lacking a responsible person, as also visible in Figure 5. The sections to pay attention to have been colored in orange. These are related to the monitoring of the application availability and to the communication towards the subcontractors.

What also is visible in Figure 5, is that many of the monitoring task are currently conducted by the CTO. Comparing to the rest of the individuals related to the continuous service provision, this person is performing most of all the tasks related to monitoring and event management. Taking into account the fact that the CTO also has other tasks on his table, some of these tasks could be delegated to some other individual. Also all monitoring tasks are currently conducted by the product development team, and none are currently performed by the product management team.

	Product management	Product manager	Customer support center	Expert 1	Expert 2	Expert 3	Product development	Release manager	Database manager	CTO	Developers
Release management											
Monthly updates								x			
Creation/release of update scripts									x		
Change management											
Bug fixes											x
Quality confirmation of the database changes									x		
SQL data content, changes				x	x						
Disaster recovery plan - initiation											
Start of plan		x									
Event management											
Server and database capabilities monitoring								x	x	x	
Integration monitoring (weekly)										x	
Integration error, analysis					x					x	
Error log analysis (weekly)										x	
Energy integration, errors				x							
Availability management											
Monthly availability score, check						x					
Application availability error alert, analysis											
Vendor management											
Communication (vendor contact person(s))											
Documenting changes											
Change management tool											x

= Observe

Figure 5. Summary of findings from the current state analysis

The following sub-headings discuss the strengths, weaknesses and challenges identified from the current state analysis and related to these identified findings.

3.5.1 Strengths

The operations required to enable the continuous provision of the service were identified. These operations were being conducted at the time of doing this study. Operations referred to include release management, configuration management the incident and change management process, tasks related to delivery of the service and monitoring of the system.

There is very deep in-house knowledge of different tasks. The individuals retaining this deep knowledge are currently performing tasks requiring specifically this knowledge. It is safe to state that the resources are currently used accordingly.

3.5.2 Weaknesses

The current division of responsibilities is incomplete. Even though most of the required tasks get done, there are some identified tasks without defined responsibilities. Due to missing responsibilities, there is no clear view on who should be performing these. This is causing the operations not running as smoothly as they could. The incomplete responsibilities include both in-house and sub-contractor responsibilities. Even though the current responsibilities for most of the tasks have been identified, these have not been communicated clearly enough in-between the different actors.

As stated in the identified strengths, there is deep in-house knowledge retained. The most competent individuals are currently performing tasks requiring specifically that knowledge. The difficulty is, however, that these persons cannot be replaced by any other employee in the company. This is causing a risk in the form of overtime and potential knowledge loss.

The manual monitoring tasks currently conducted are not distributed evenly. Most of the manual monitoring tasks are currently performed by the CTO. This is causing an extra burden, taking into account the other tasks for which this individual is responsible. There is no defined schedule for performing these monitoring operations. Also, there is no structured way of conducting or recording this monitoring.

There is also a lack of transparency regarding the changes made in the data content of the SQL database. As already stated earlier in chapter 3.3 Incident and change management process, these tasks count as change needs that should be recorded in the change management system based on the standard operating model. The reason why this is currently not done is due to the product development team having restricted resources, and not able to provide a resolution as fast as required. By operating in this way, the root cause of the incident still remains. The main problem with this way of operating is that when an error caused by a data change occurs, it is very difficult to track the changes that have been made in the database.

3.5.3 Challenges

There are some challenges that can be identified from the current state. As already stated in chapter 3.5.2 Weaknesses, the current roles are strongly individualized. This will most probably cause some difficulties when organizing the responsibilities based on roles. Currently only a couple of identifiable roles exist, which are presented in the table below.

Table 3. Identified roles

	Roles	Identified person
1.	Release manager	Yes
2.	Database manager	Yes
3.	Chief technology officer	Yes
4.	Product manager	Yes

As listed in Table 3, there are four identifiable roles that are related to the continuous provision of the software service. These are release manager, database manager, chief technology officer and the product manager. These four aforementioned are the only roles that are identified.

3.6 Summary of Key Findings from the Current State Analysis

From the findings presented in this chapter, three key findings have been identified. These are the overall lack in structure of the continuous service provision operations, the uneven distribution of the tasks in between relevant actors and the gap in transparency regarding the changes made in the SQL database.

Overall, there is a lack in structure of the continuous service provision operations. Most of the required tasks related to the operations are currently conducted, but not in a clearly defined way. The monitoring of the availability of the software can be mentioned as an example. Many different types of alerts are getting analyzed, but the structured way for further use of this information is lacking. The roles and responsibilities related to all of these tasks that are not communicated clearly enough has led to a situation where nobody is certain about which individual should handle what. The outcome of this is that the tasks are currently not conducted systematically, and the resolution of possibly occurring problem states has a longer duration than required. As a result of the current state analysis, capable individuals that are currently performing the continuous service operation tasks have been identified for most tasks. Still, there are some tasks left without a specific person identified.

The uneven distribution of tasks was identified from Figure 5, which presents the tasks and the currently responsible person identified from the current state analysis. Most of the maintenance and monitoring operations are currently conducted by the product development team. This is resulting in excess workloads compared to the work capacity of individuals and the whole team. The main focus of the product development team should be in handling change needs.

The third one of the key findings is the lack of transparency regarding the changes being made regarding the data content in the SQL database. This is due to the lack of a defined documenting procedure for this type of changes. As already mentioned, this way of operating is causing problems in case of an error occurring as a result of the data content. Without proper documentation of changes it is difficult, if not impossible, to track exactly what has been done.

The study focuses therefore on building a proposal based on these three aforementioned key findings that were retained from the current state analysis. The goal is to define responsibilities for the operations related to continuous software service provision. This is to be done based on a suitable structure developed in the process of conducting this study. In combination to this, the proposal will seek to add transparency by defining a suitable documentation procedure for the SQL database changes.

Figure 6 visualizes how this study is set to move forward to developing the proposal. It presents the findings that were chosen to be addressed, the challenges related to these findings and the theory to be used for managing these challenges in the process of creating the proposal.

FINDINGS	CHALLENGES	RELATED THEORY
1. Lack in structure of the operations	- Overall division of responsibilities is unclear - Many of the required event-, and availability management measures are being conducted, but not systematically	- ITIL: Event management - ITIL: Availability management - Organizing responsibilities
2. Uneven distribution of tasks	- Tasks related to event management are currently strongly individualized based on knowledge	- ITIL: Event management - Organizing of responsibilities
3. Gap in transparency in change management	- How to include the SQL changes in the process maintaining the fast workaround?	- ITIL: Change management - ITIL: Problem management

Developing the Roles and Responsibilities for Continuous Software Service Provision

Figure 6. Findings and related theory

In the following chapter this study focuses on discussing best practice and available knowledge that serves as a basis when developing the current way of operating and creating the proposal.

4 Available Knowledge and Best Practice of Continuous Service Provision Operations

This section discusses relevant knowledge and best practices in the field of providing a continuous software service. The aim of this chapter is to provide relevant knowledge about different areas related to solving the two key findings identified in the current state analysis. The areas discussed in this section are event management, problem management, change management, availability management and the organizing of tasks and responsibilities. This literature study draws knowledge from multiple different sources.

4.1 Event Management According to ITIL

Based on ITIL an event is any sort of detectable occurrence that in any way has an effect on the management or the delivery of an IT service. This means event is any sort of change in the system, but does not necessarily mean a fault. Due to this, it is important to determine what types of events are significant, and need to be detected. It is also important to differentiate event management and monitoring. Events are usually detected through some sort of notifications or alerts. Monitoring on the other hand is a wider term referring to increasing control of a software or a service. (Cannon & Wheeldon 2007: 83) The monitoring that results in an alert can be conducted either with active or passive monitoring tools. The event management enables comparison of design standards of the service and the actual performance of the service. In ITIL it is stated that “event management provides mechanisms for early detection of incidents”. (Cannon & Wheeldon 2007: 35-37)

Figure 7. Events link to problems

Event management is related to multiple other processes of service management. It can be applied to any part requiring control. These include incident-, problem- and change management, capacity- and availability management, configuration management and asset management, since changes occur in all of these areas. (ITIL- A guide to event management, UCISA) Event management includes for example tracking of normal activity for instance usage, the performance of configuration items and changes related to these, and security monitoring. (Cannon & Wheeldon 2007: 36)

Having event management in place enables actions to be taken to manage incidents before they cause disruptions in service. The linkage between event management and problems is visualized in Figure 5. (Cannon & Wheeldon 2007: 35-37). Usually there is no specific individual appointed to the role of an Event Manager. This is due to the amount of events and the variation of causes. It is thought important that the procedures are directed to ensure that they are being conducted in an effective way. (UCISA, ITIL - A guide to event management, n.d.)

4.1.1 The Process of Event Management

Based on ITIL an event creates a notification. Events can be detected by determining specific criteria for configuration items, which when criteria is met, will generate a notification. Configuration items do not send any notifications in case not specifically defined to do so. Different sort of “agent” software can be used to detect notifications. The events should then be filtered, since as already stated, not all events require actions. (Cannon & Wheeldon 2007: 39)

When filtering, the significance of the event should be determined based on different categories. Informal meaning event not requiring action, warning meaning notification for required measures to be taken to prevent exceptions from happening and exceptions meaning that the software provided is not functioning accordingly and that business is being affected. (Cannon & Wheeldon 2007: 39-40).

Informal events are usually logged in an event log. Events that are defined as exceptions should be directed to the Incident, Problem, and Change management processes. The events that are warnings are not yet visible for the end user. As stated, warnings indicate that preventive measures might be required. Based on different triggers, the events are then correlated and required measures are determined. These different measures might be that if the cause is as a preventive measure directed to Incident, Problem, and Change management processes where they will be recorded as tickets. An alert is generated indicating that human response is required, or then the event might not be defined significant, due to which it is not requiring actions and is then logged without measures. Despite what actions have been taken, their effectiveness should then be reviewed after which the event should be closed. (Cannon & Wheeldon 2007: 38)

4.1.2 Event information management

A big amount of different type of important information is retained from event management. These include messages that inform about the technical function of components, configuration of system parameters that determine the normal function of the system, to which the occurrences are compared to when events are generated by for instance agent software that monitors the functions. Regarding the recording of events, there is no standard way to create event records. Events do though include a lot of important information that can be used for the continual service development, which is why they should be recorded and analyzed. Metrics that can be used for the analysis are for instance amount of events of different categories and the significances of these, the percentage of events that in the end resulted in incidents and the availability issues caused by events. (Cannon & Wheeldon 2007: 44)

4.2 Problem management

The definition of a problem is, based on ITIL, the root cause to one or multiple incidents that occur. The term problem management therefore refers to the management of these problems with the aim of preventing these. This will then lead to prevention of the incidents that these problems might have caused. The scope of problem management also includes coming up with, and implementing, the resolutions to these identified problems. (Cannon & Wheeldon 2007: 58-59)

Based on ITIL, problem management also includes the maintenance of information regarding known workarounds for identified problems. These workarounds then enable fast resolution in case a similar problem reoccurs. Even though incident and problem management are two different processes based on ITIL, they are strictly combined. Often same procedures will be used for implementing the resolutions. (Cannon & Wheeldon 2007: 58-59) When a workaround is used to resolve an incident caused by a problem, the problem ticket recorded should remain open to enable resolution of the actual root cause. (Cannon & Wheeldon 2007: 64)

Increased service availability, due to less downtime causing disruptions to the service, is a result of problem management. Overtime permanent solutions to recurring problems will be identified, decreasing the amount of problems. This leads to sustaining the quality of the service. (Cannon & Wheeldon 2007: 58-59)

ITIL identifies different paths that problems can be detected through in the reactive problem management. These are for instance through detection of incident, through technical support group analysis or automated monitoring of faults in either the infrastructure or the application. Based on ITIL the problem identified should be recorded and all relevant information, also including the resolution, should be attached. By categorizing the problems, the information can later be extracted for management purposes. Prioritization of problems should be conducted as for incidents. Based on ITIL, the same prioritization system can be used for both, but regarding problems the severity should specifically be taken into accountancy. (Cannon & Wheeldon 2007: 61)

4.3 Change Management

ITIL defines the change management process to be a way to minimize risks when making changes in an IT service. Based on the framework it is a part of the service transition section. The standard process based on ITIL includes evaluation, prioritization, quality assurance through testing, and the recording of changes made. The main aim is for the changes to be implemented quickly and efficiently but with minimal risk of disruption in the service. To make things run smoothly, both change models and standards changes can be used. The aim of change models is to have standardized implementation procedures of specific types of changes. The standard changes count as one of these change models but which are pre-approved. Standard changes have minimal risks related. Based on ITIL there should be a change advisory board with a change manager that approve all changes made in the system. (IT Change management, BMC)

The process of change management includes based on ITIL 8 different steps. These steps include the creation of the change request, reviewing, planning, testing, and creation of a change proposal, implementing change, reviewing performance of the change and closing of process. In the first reviewing phase the information attached to the change request will be evaluated. Both the priority and the practicality of the change will be evaluated and determined. In case the change request is not relevant, it will not be approved. By not being relevant, the change request may for example be related to a problem that has already been addressed. In case of a change being approved, it will be directed for fulfillment and addressed with a responsible person. (IT Change management, BMS)

The change needs to be planned, including execution, required resources and timeline. The execution will need to be tested before the change is implemented to identify problems. This should be done related to any changes regarding functions of the software and/or, cases that are related to debugging. After doing this, a change proposal will be made to state the importance and all other information related to the change and to determining its priority. (IT Change management, BMS)

Implementation of the change is followed by testing its successfulness and by performing required measures in case of faults. At the end a final review will be made, to determine the effectiveness of the change procedure. The review should include all areas starting from reviewing. After the evaluation has been made, the process will be closed by ensuring all relevant documentation exists. (IT Change management, BMS)

4.4 Availability Management Based on ITIL

Availability management refers to management of availability targets to be achieved. This is done by measurement of service performance. The process of availability management should be applied to all technology, especially services that are provided based on a Service Level Agreement. The availability management incudes identification of occurring incidents that are effecting the availability of the system. The availability is ensured by continuous monitoring of the system. Based on ITIL, there needs to be an understanding about different areas related to the business perspective of the service. This is to ensure that the availability management is being conducted in a correct and suitable way. These business perspectives include the business processes and their requirements, the targets of delivery of the service, the IT attained in the delivery of the service and the business impact of the service. Relevant for this study is to understand the measurement and management point of view of the availability management. In addition to these, the ITIL definition does include other areas which have not been addressed in this study. (Lloyd & Rudd 2007: 97-98)

4.4.1 Monitoring and Measurement

Monitoring and measuring, including analysis and reporting, are counted as reactive activities of availability management. Availability management should be done in all different areas of delivering a service, which is also why it is closely linked with for example incident and event management. Monitoring enables following up on the targets stated in the service level agreements being met. The monitoring should take into account both the user and the business perspective since their impact differs. Disruptions in the availability affects the user regarding how frequently downtime occurs, how long it lasts and how large the impact of the downtime is. Based on ITIL, what is not measured cannot be managed and neither can it be improved. (Lloyd & Rudd 2007: 99 -104)

4.4.2 Reporting and Analysis

When monitoring of availability is conducted, this information should be reported and analyzed. Compiling the information for the reports does require manual work, which is why as much as possible should be automated. The reports should be conducted systematically. Availability reports should be created from the business and user perspective. The reports will give important insights that can be used for the continual development of the service, giving the ability to identify patterns and critical points. (Lloyd & Rudd 2007: 104-105)

4.4.3 Deployment of Availability Management

The implementation of availability management is a maturing process. When starting the deployment, there should be a definition of what tools will be used for the monitoring of the availability and what the aim of this monitoring is. Available pre-requisites should be discovered, since some of these can be exploited in the availability monitoring. (Lloyd & Puff 2007: 124-125)

4.5 Organizing Tasks and the Responsibilities

The targets of business operations are reached through clearly organized operating ways. Managing teams require authority, responsibility and delegating, for tasks to be performed systematically as planned. As stated by Serban (2017), authority that exists without responsibility results in unaccountability. In other words, authorities need to be clearly defined and delegated for getting the assurance. Defining authorities mean that there are objectives that need to be met. The person having the authority is then responsible of conducting the measures for these objectives to be met. Responsibility in itself cannot be delegated. Delegation of authorities should be done to decrease overloading. It should though be done accordingly, based on the skills. (Serban 2017)

4.6 Conceptual framework

Section 3. Current state analysis of Continuous Software Service provision process and section 4. Available knowledge and best practice on Continuous Software Service provision form the outcome of section 4, which is the conceptual framework of this study. The conceptual framework is presented in Figure 8.

FINDINGS	CHALLENGES	RELATED THEORY
1. Lack in structure of the operations	- Overall division of responsibilities is unclear - Many of the required measures are being conducted but not systematically	*4.1 Event management according to ITIL* *4.4 Availability management* *4.5 Organizing tasks and responsibilities*
2. Uneven distribution of tasks	- Tasks are strongly individualized based on knowledge	*4.1.1 The process of event management* - *"How event management link to incident-, problem- and change management"*
3. Gap in transparency in change management	- How to include the SQL changes in the process maintaining the fast workaround?	*4.2 Problem management* *4.3 Change management*

Developing the Roles and Responsibilities for Continuous Software Service Provision

Figure 8. Conceptual framework

After presenting the relevant literature for this study, the following chapter introduces how the proposal of this study was built.

5 Building Proposal for the Roles and Responsibilities of Continuous Service Provision Operations

This section describes the proposal building of developing the roles and responsibilities for the operations enabling the continuous software service provision. The proposal is built upon the knowledge gained from the data collected, the available literature discussed in this study and insight and experiences of working in the case company. This chapter includes an overview of the proposal building stage, referring to the key findings from the different data sources. The chapter ends with a summary of the proposal that was created.

5.1 Overview of Proposal Building Stage

The objective of this study was to develop the roles and responsibilities for the continuous software service provision. During the process of building the proposal, some sections included in, or strongly related to, the continuous software service provision required assessment in greater detail. This was inevitable for enabling the defined outcome to be reached.

The development of the roles and responsibilities of the continuous software service provision was reasoned by the knowledge gained during the study. The study started off with investigating the current state of the continuous service provision of Software X, and the related responsibilities. This analysis enabled the identification of the relevant areas in literature. After studying literature, some additional interviews were conducted to gain deeper knowledge related to the departmental objectives and needs. The proposal was built based on the applicable elements from the aforementioned and also taking into account the author's own experiences and insights related to the department and its pre-requisites.

The proposal created as the outcome of this study, was developed in close collaboration with the employees of the company. This was done by organizing workshops and additional meetings in which the proposal was formed and defined. The proposal strives to combine all relevant aspects into an easily understandable solution. The logic of the proposal building is visualized in Figure 9.

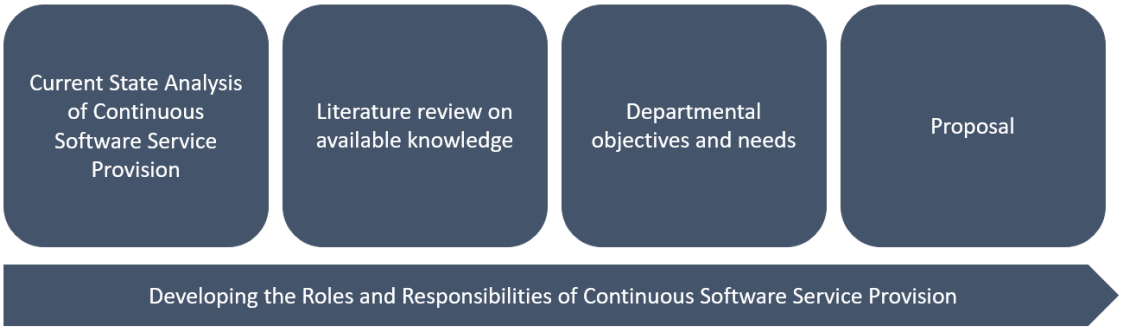


Figure 9. Proposal building logic

The following section introduces the proposal building based on key finding from current state analysis.

5.2 Proposal Building

The different elements that have influenced the proposal building are presented in the following section. These key findings were formed from elements from the observations made during the analysis of the current state, combined with reasoning based on the insights collected from the available literature. The proposal building was strongly affected by both the stakeholder's and author's own experiences and understanding of the department and its abilities. This understanding had been gained from working in the company. The key findings, describing all aspects influencing the proposal building, are presented in the following sections. The different parts of the proposal are presented together with the aforementioned.

5.3 Proposition to Solve the Lack in Structure of Operations

One of the key findings from section *3. The Current State Analysis of Continuous Software Service Provision* was the lack of structure in the operations currently conducted. This included the lack of clearly defined roles and responsibilities of the tasks. The structure was also lacking in the way the actions were being conducted, resulting in the tasks not being systematically performed. Neither was there any defined way of recording the problems identified through different monitoring solutions. The records were mostly maintained in e-mails.

Best Practice from Literature

From literature, more precisely from section *4.5 Organizing tasks and responsibilities*, it was stated that for tasks to be performed systematically, responsibilities need to be clearly delegated.

In section *4.4.2 Reporting and analysis*, it was stated that information gathered from monitoring should be reported in a systematic way and analyzed. The reports should be compiled systematically for enabling information to be drawn for management and further development.

Findings from Current State

As stated in section *3. Current state analysis of Continuous Software Service Provision*, the lack of responsibilities resulted in an unstructured way of working, which was causing confusion and in the end unnecessary work. As an example could be mentioned the management of the service providers. Due to the lack of a specified contact person from the case company software department, there were some unclear parts regarding the operating ways of the service providers. This was due to nobody having taken the ownership of the task. Also the alert analysis being performed required a more structured operating procedure to be defined.

A preference from the department was that the continuous service provision tasks would somehow be centralized. From working in the company, suitable individuals were identified for the forming of a team specialized in continuous service provision. The team will from now on be referred to as Operations team.

5.3.1 Operations Team

The reasoning behind the creation of the Operations team is that it enables centralizing the continuous service provision tasks to a specific team, which results in an increase in clarity in the department. The scope of the team was defined in cooperation with the product manager. The team was formed of a combination of the individuals from the previous SQL-team and an individual from the product development team to function as the team leader.

The operations team will have the focus and main responsibility of the continuous service provision. Finalizing the responsibilities of the team was done in a workshop with the product manager and the customer support manager. The tasks and the concrete performing of these was discussed and defined in a meeting with the team leader of the Operations team. They will also manage the task of being the contact person towards the service provider used in the provision of the software. This means that the team will manage the contact towards the service provider in all occasions required. Also, this includes managing the relationship and ensuring that the collaboration is functioning appropriately. Operations team will also be creating a monthly report of the service availability to follow up on the fulfillment of the SLA during the previous month. This report is introduced in greater detail in section *5.2.1.3 Software availability report.*

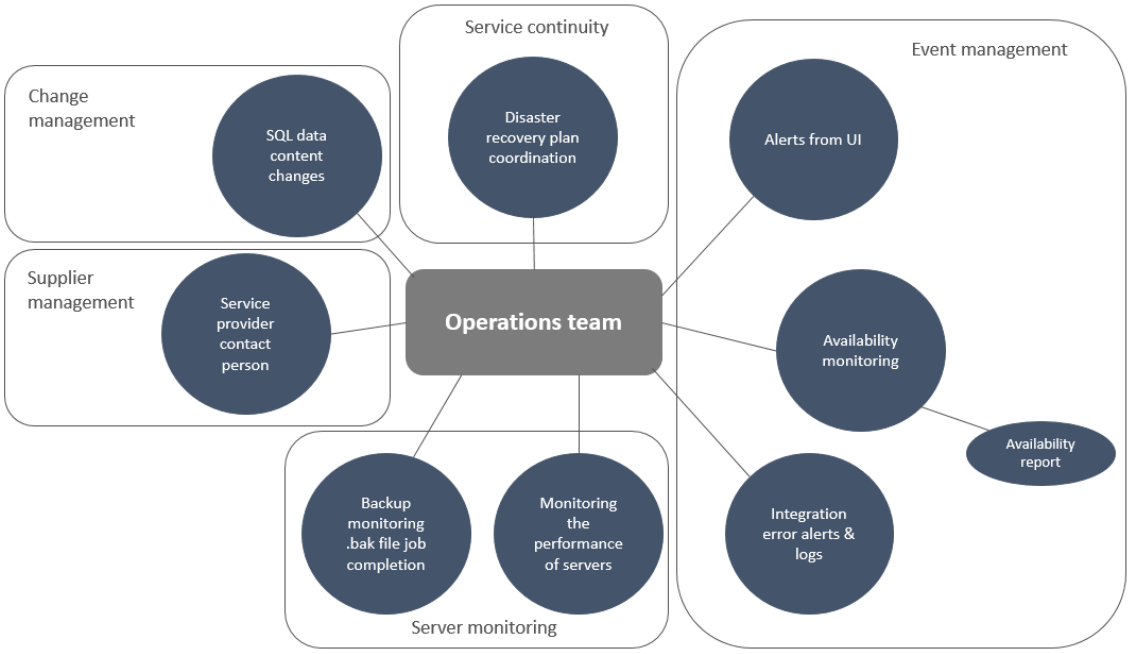


Figure 10. Operations team, tasks

Figure 10 visualizes the tasks that the new Operations team will manage. The objective is, that also the monthly version updates will be transferred to the Operations team sometime in the future.

An overview on all actions included in the continuous software service provision will be presented in section *5.2.2.2 Roles and responsibilities for continuous software service provision tasks,* where the RACI table created as the main result of this thesis will be presented.

5.3.2 Processing and Documenting of Alerts, Event management

The possibilities of systematically recording the alert analysis were evaluated in cooperation with the person assigned to be the leader of the Operations team, and with also the Quality manager present. The outcome of this meeting was the documenting procedure, enabling a structured way of conducting event management. The documenting procedure is presented below in Figure 11. The recording will take place in the change management tool already in place. The recording will be conducted in a similar way as in the standard incident management procedure.

The analysis of the different alerts will from now on be recorded in case of the alert indicating an error in Software X. The recording will take place in the change management tool based on defined criteria in the tool. Exceptions are though the actions that count as normal functions, but are performed based on alerts received. An example of this sort of action is for example disc space. Automatic alerts are received after which space will be added based on different criteria defined in collaboration with the product management and the Operations team. This action will not be recorded in the tool.

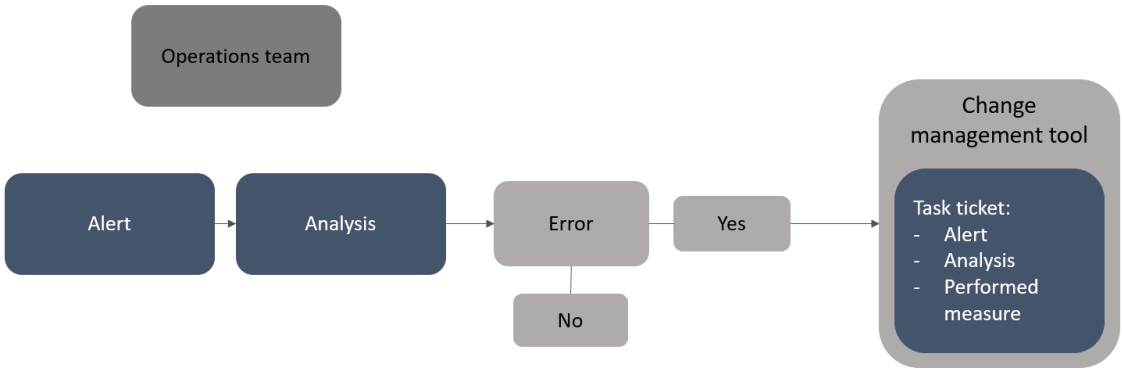


Figure 11. Alert recording

Having a defined operating way for processing of the alerts enables systematic event management to be performed.

In the process of defining the documenting procedures for the event management actions, the documenting requirements regarding the quality audit were taken into account.

5.3.3 Software Availability Report

In the process of developing the proposal, also a report template for the monthly availability report was created. The template is available in Appendix 11. The report template was developed to ensure that the documenting requirements regarding the upcoming quality audit were being fulfilled. By creating this monthly report, follow up on availability based on the service level agreement is monitored. The report is created taking into account both the business and customer point of view. By also analyzing the impact of showstoppers and additional disruptions, the actual availability from the user perspective is taken into account.

The examination period is the last 30 days. The report includes the availability score, extracted from the automatic monitoring for all production servers. Based on occurrence, the report includes a listing of showstoppers and additional disruptions in the service during the past month. In the occurrence of the aforementioned, the report also includes an analysis on how these have affected the service availability during the month, which production server/servers these had an effect on, and which customers it especially concerned. In the end the estimated availability score based on the decrease caused by either showstoppers or other disruptions will be stated.

5.4 Proposition to Solve the Uneven Distribution of Tasks

The uneven distribution of the responsibilities of the different tasks regarding the continuous service provision was pointed out as a key finding from section *3. Current State Analysis of Continuous Software Service Provision.* This is strongly related to the previously presented lack in structure of the tasks performed, and the fact that the overall picture hadn't previously been reviewed. As previously mentioned, many of these tasks included analysis of different sorts of automatic alerts.

Best Practice from Literature

Analysis of different alerts related to event management, knowledge was drawn from section *4.1 Event management.* Based on the theory, event management plays a significant role in preventing incidents. When organizing the operations of event management it should also be taken into account that events are related to multiple different areas. In literature it was also stated, that event management is strictly linked to the processes of incident, problem, and change management, which was also taken into account when organizing the responsibilities in the process of building the proposal.

Findings from Current State

As also stated in the current state analysis, the tasks were strongly individualized. In some cases it had to do with different levels of knowledge, but there were though occurrences where multiple people had the required knowledge to conduct some of the tasks. In most cases, the knowledge for at least conducting an initial analysis existed after which other people could be consulted. The use of this knowledge was to be included in the proposal created, enabling all available resources to be used in the most appropriate way. Based on this reasoning, the situating of the Operations team managing the event management was done so that it enabled the possibility of redirecting tasks, where additional consultation was required. This is visualized in the workflow presented in Figure 12 found in the following section.

5.4.1 Situating of Operations Team

The pre-requisites of the department, in addition to event management process having a strong linkage between incident, problem and change management, were taken into account when creating the proposal and situating the Operations team. A workshop with the product manager, the Quality manager and the customer support manager present, was organized to determine where to situate the team. The team forms an additional level between the customer support and the product development team, as visualized in Figure 12 below. The team itself counts as a part of the Product development team.

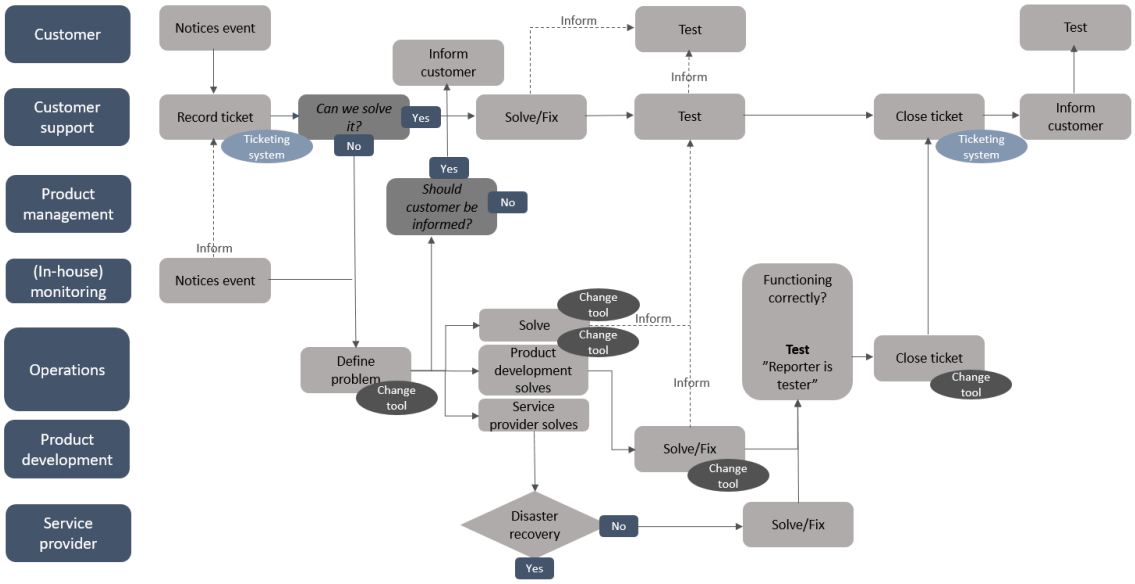


Figure 12. Incident management process including Operations team

As visible from the process chart presented in Figure 12, the Operations team managing the event management, will take initial actions regarding corrective measures required. In the identification of additional support required, Operations team is able to redirect or/and consult, based on root cause, either the product development team, or the service provider. Operations team defined as the contact person towards the service providers is also justified by making this process as clear as possible.

5.4.2 Roles and Responsibilities for Continuous Software Service Provision Tasks and Related Documentation

For describing the responsibilities of all tasks related to the continuous software service provision, a RACI table was created and visualized in Figure 13. This table visualizes the division of tasks including all actions conducted for enabling the continuous software service provision. The table also includes a short description of the tasks and the interval when these will be performed. The table is created based on the information gathered during this study. The relevance of the table was then evaluated, and required changes conducted, in a workshop with the product manager and the customer support manager.

The responsibilities were defined by teams and sub-teams based on the fact that there was a wish to decrease the current individualization of tasks. Also this was considered a detailed enough level keeping in mind the upcoming quality audit. The tasks were organized based on different processes in ITIL. This was done with the aim to ease the understanding of the different sections included in the continuous service provision.

The division is concentrated on centralizing all the continuous service provision tasks for the new Operations team, as already mentioned earlier. Based on this proposal, the Operations team will take main responsibility of the software continuously functioning appropriately. Many of the tasks of the Operations team are tasks that were previously managed by the CTO.

	Product management	Customer support center	Product development	Operations	Service provider	Department manager	Description	Interval
Release management								
Development of version			R . A				Includes also creation and publishing of update scripts to T-disc (T-levy)	Monthly
Version update	A	I	R				Updating the newest version to production	Monthly
Change- & Incident management								
Bug fixes	C	I	R . A					When required
Quality confirmation of database changes			R . A					On a continuous basis
Changes related to the SQL data content			C . I	R . A			Tasks performed based on request. Performs required measures, documentation in Jira.	When relevant
Disaster recovery plan								
- Start of Disaster recovery plan	R . A	I	C . I	C	C . I	I	Desicion to start plan	When relevant
- Deployment of Disaster recovery plan actions	I	I	R . C	A	R . C	I	Coordinate. Ensure that the tasks are being conducted	When relevant
Event management								
Error backlog			R . A	C			(Raygun ja Live log), Monitoring and reaction	On a continuous basis
Automatic alerts from user interface	I	I		R . A	C		Analysis	When occurs
- Disc space fill up	C			R . A	I		Analysis & Reaction	When occurs
Server performance monitoring	C		C	R . A	C		Cygate reports, following trends + analysis + adding capacity when required	Monthly/When required
Availability monitoring								
- Uptime robot automatic alert	I	I	C	R . A	C		Analysis	When occurs
- Availability report	I	I		R . A			Creation of report	Monthly, previous month
Integrations								
- Integration error alerts	I	I	C	R . A			Analysis (successful or error) + corrective measures	When occurs
- Integration log monitoring	I	I	C	R . A			Successful (yes/no)	When relevant
- Energy integration, error	I	I	C	R . A			In the occurance of detecting data not being transmitted. Analysis & corrective measures	When noticed
Monitoring								
SQL backups								
- .bak file job completion			C	R . A	I		Monitoring, has the job been succesfully completed	Daily
- Server backups				I	R . A		Final backup taken from the .bak file job.	Daily
Supplier management								
Contact person	C		C	R . A			Managing contact to server provider in all occasions.	When relevant

Figure 13. RACI table on Continuous Service Provision of Software X

Figure 13 visualizes the full RACI table created for the continuous software service provision.

Additional Tasks Included in the RACI table

Each of the operational tasks included in the RACI table were assessed separately in greater detail. This was done to ensure that a defined and clear way to perform each task was available. This action included the responsibilities of the disaster recovery plan being reviewed, and the unclear parts regarding the cooperation with the server provider being defined as a result of which actions were being taken. In the process of assessing each task separately, some additional tasks that had not been identified in the current state analysis were discovered. These were identified to be individual parts to be included in the continuous service provision. Figure 14 includes a more detailed caption on especially these additional tasks.

Automatic alerts from user interface	I	I		R , A	C		Analysis	When occurs
- Disc space fill up	C			R , A	I		Analysis & Reaction	When occurs
SQL backups								
- .bak file job completion			C	R , A	I		Monitoring, has the job been succesfully completed	Daily
- Server backups				I	R , A		Final backup taken from the .bak file job.	Daily

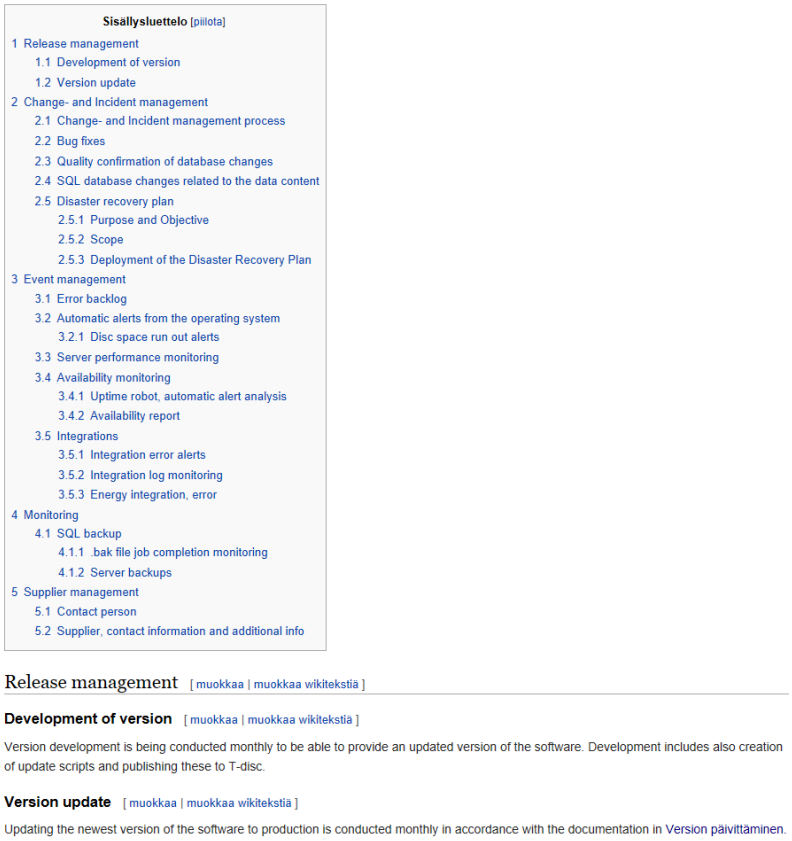
Figure 14. Additional tasks in the RACI table

Automatic alerts from the user interface was added in the RACI table. These alerts include different sorts of categories of alerts, that are all generated by events in the user interface. These are provided by the server provider. Currently the only clearly identified category is disc space. The responsibility of defining the additional categories has been delegated to the Operations team.

Regarding backups that are taken from the system on a continuous basis, there was identification of the need to monitor the success of an automated task related to this. The SQL backup is performed as a sequential job, which required monitoring of the successful completion of the first phase to be added as a task in the RACI table. Related to this, the server backup made in the second phase was also added for consistency.

5.4.3 Continuous Service Provision Documentation

As a part of this thesis, required documentation regarding continuous service provision operations was created to fulfill the requirements of the upcoming quality audit. Figure 15 is a caption taken from the user interface of the in-house tool used for documenting of instructions.



Sisällysluettelo [piilota]

Release management [muokkaa | muokkaa wikitekstiä]

Development of version [muokkaa | muokkaa wikitekstiä]

Version development is being conducted monthly to be able to provide an updated version of the software. Development includes also creation of update scripts and publishing these to T-disc.

Version update [muokkaa | muokkaa wikitekstiä]

Updating the newest version of the software to production is conducted monthly in accordance with the documentation in Version päivittäminen.

Figure 15. Continuous service provision documentation

This documentation functions as supporting material for the RACI table. As mentioned, each of the tasks were assessed separately for their concrete content to be identified. This information gathered was applied when forming the descriptions included in this documentation. The descriptions were recorded in a centralized place. The full documentation created is available in Appendix 12.

5.5 Proposition to Solve the Gap in Transparency in Change Management

In section *3. Current State Analysis of Continuous Software Service Provision*, it was identified that when making changes to the data content in the SQL database, there was a gap in transparency. This raised the question of how the SQL team could most easily be included in the change management process that was currently in place.

Best Practice from Literature

The importance of recording changes, problems and identified root causes were pointed out in both chapter *4.2 Problem management* and chapter *4.3 Change management.* Also, these records should be maintained centralized. Based on literature, the requests could then be analyzed and reports could be drawn in case further analysis was required.

Findings from Current State

As stated in chapter *3. Current state analysis of Continuous Software Service Provision,* there is a recording tool being used for recording the incidents and the change needs. The only changes that are not currently being recorded on a continuous basis are the SQL changes and the underlying root causes for these. The most optimal way for adding transparency in this area, still keeping the process as simple as possible was identified in cooperation with the team leader of the new Operations team, in which the individuals of the SQL team will be included.

5.5.1 Documenting Procedure of Data Content Changes in the SQL Database

A chart visualizing the documenting procedure can be seen in Figure 14. A suitable documenting procedure was discussed and defined in a workshop with the team leader of Operations team. The outcome was a defined documenting procedure for adding transparency to the actions.

The changes made related to the data content of the SQL database will be recorded in the change management tool as tasks. By recording these changes, transparency will be added in the work amount, but also regarding the tasks to be conducted. This will enable software development to solve the root cause when required to. The documenting procedure also enables the tracking of these changes when required.

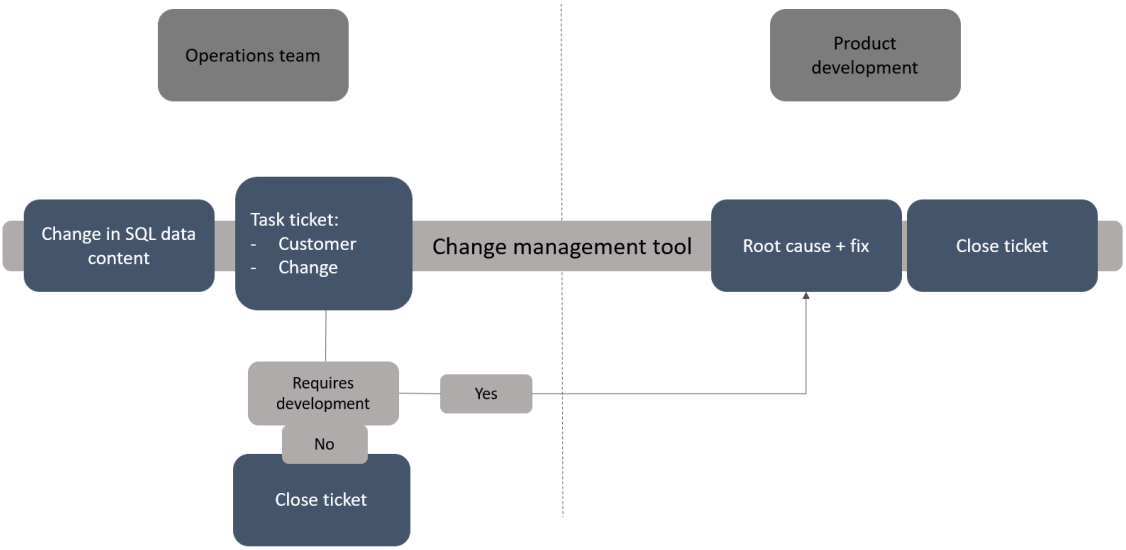


Figure 16. Documenting process of SQL changes

As visualized in Figure 16, the changes made will be recorded in the change and incident management tool, including all required information. In case further development is required, then the information about the task will be forwarded to the product development team. Since the required change may be an indication of an underlying fault, the product development is able to address the root cause based on the ticket created in the tool.

In the process of defining the documenting procedures for the changes performed in the data content of the SQL database, the documenting requirements regarding the quality audit were taken into account.

5.6 Summary of Proposal

This section presents a summary of the proposal created regarding the roles and responsibilities of continuous software service provision. The different parts included in the proposal were reasoned and presented in detail in the previous chapter. The proposal is visualized in Figure 17. below.

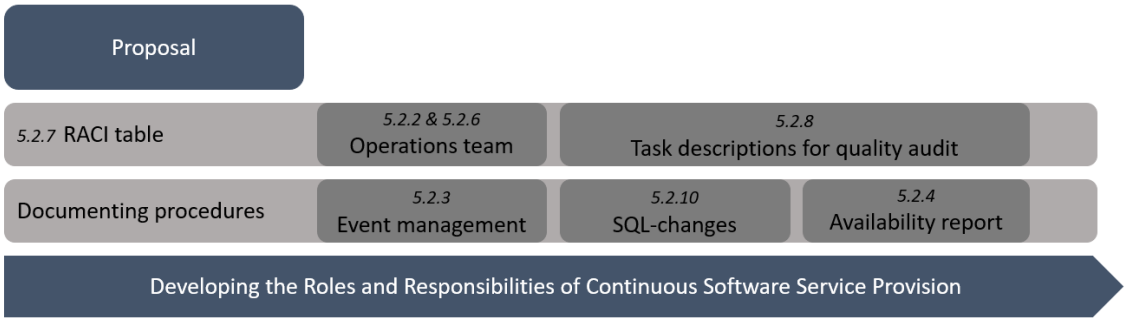


Figure 17. Proposal summary

The proposal for developing the roles and responsibilities of the continuous software service provision includes the RACI table and the documenting procedures regarding the tasks. The RACI table, which is presented in detail in section 5.2.7, is the main output of this thesis. It gives the full view of all actions to be conducted regarding the continuous service provision as well as their defined responsibilities. The proposal includes an organizational aspect in form of organizing the Operations team, entailing the main responsibility of continuous service provision, and the re-organizing of the tasks based on this new team. The Operations team is introduced in detail in section 5.2.2 and the situating of the team is described in section 5.2.6. For fulfilling the requirements of the upcoming quality audit, supporting documentation in the form of task descriptions was created on the basis of the RACI table. This documentation is presented in section 5.2.8.

Documenting procedures for continuous service provision operations were defined. This enables a structured way of conducting event management and an increase in transparency regarding the SQL data content changes to be made. The recording procedures were defined in alignment with the requirements of the quality audit. The documenting procedures of event management are introduced in section 5.2.3, procedures for the SQL-changes in section 5.2.10, and the availability report template developed to fulfill the documenting requirements of the quality audit, is presented in section 5.2.4.

6 Validation of the Proposal

This section describes the validation stage of the Proposal developed in Section 5. The chapter starts by introducing the logic in the validation stage. Then, it describes the feedback, implementation and evaluation of the Proposal.

6.1 Overview of Validation Stage

A validation of the proposal developed was conducted to ensure that the solution would with most certainty fulfill the needs of the case company. The objective of the validation was to verify the suitability of the proposal for the software department and to implement it. Figure 18 visualizes the validation stage.

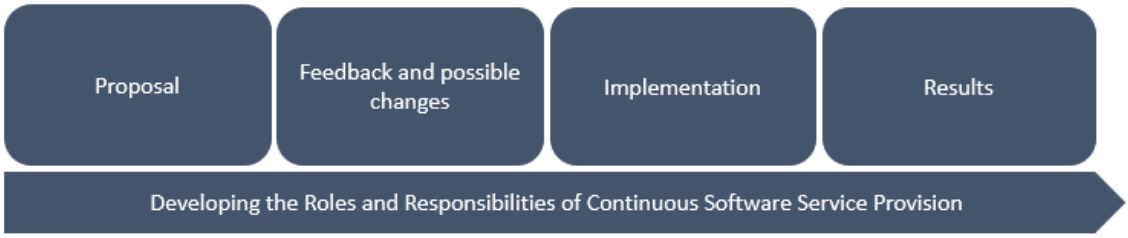


Figure 18. Validation stage overview

Feedback gathered from a validation meeting was incorporated in the proposal. This feedback consisted of minor changes. The proposal was then implemented.

6.2 Further Developments to RACI table

Some minor changes related to the different sections of the RACI table were identified based on the feedback. Many of these had to do purely with adding clarity and changing specific words to more descriptive for avoiding misperceptions. The RACI table was modified accordingly.

6.3 Implementation of the Operations Team

The Operation team was implemented during the process of conducting this thesis. It started its functioning as a virtual team. The responsibilities were communicated, and authorities for conducting the different tasks were delegated to the Operations team. The aim is, that after the turn of the year the organizational change regarding the Operations team will be concrete.

Due to the upcoming quality audit, the operating way with the newly defined responsibilities needs to be completely implemented in January and complied with from then forward.

6.4 Summary of Results Achieved

The objective of this thesis was to develop the roles and responsibilities related to the operations of providing a Continuous Software Service. The proposal created as a result of this study fulfills the expected outcome defining the roles and responsibilities of the operations related to continuous software service. A summary of the proposal and the results achieved is presented in Figure 19 below.

FINDINGS	PROPOSAL	EXPECTED BENEFIT
1. Lack in structure of the operations	Operations team RACI table	Centralizing the responsibility of continuous service provision to specific sub-group will increase clarity of operations.
2. Uneven distribution of tasks	Re-organizing tasks based on Operations team RACI table	Division of tasks done appropriately. Added productivity and clarity in operations.
3. Gap in transparency in change management	Documenting procedure	Added transparency and traceability of changes made.

Developing the Roles and Responsibilities for Continuous Software Service Provision

Figure 19. Summary of results

The RACI table increases the clarity regarding actions being performed. As it also indicates the interval for the different tasks, it enables the continuous software service provision operations to be conducted systematically. By centralizing the main responsibility of the continuous software service provision to the Operations team, the team is now expected to adopt the ownership of that function.

By the tasks being re-organized on the basis of centralizing most tasks to the Operations team, implies the tasks to be divided more evenly in the software department. The burden of the CTO will decrease due to Operations team taking the ownership of continuous service provision operations. Operations team documenting the tasks they perform, will most certainly bring transparency to their overall work amount. This might require some assessment in the future, in case it is understood that the team performing SQL tasks not being their responsibility.

Defining a documenting procedure for the SQL data content changes increases transparency of operations. It also increases the traceability of changes being made. Defining the documenting procedures regarding all the operational tasks related to the continuous software service provision, also results in the requirements of the quality audit to be fulfilled.

The new responsibilities and operating procedures were implemented in the company. The Operations team implemented the documenting procedures and took the change management tool in use to document both SQL-changes and event management alerts when required. During the first month 166 SQL task were recorded in the tool.

The following section includes the summary and conclusion of this thesis.

7 Summary and Conclusion

This section presents the summary and the conclusion of the thesis. This section consists of four different sections. These sections include an executive summary, next steps and tips for implementation of the proposal and an evaluation of the results. This chapter ends with some final words.

7.1 Executive Summary

The objective of this thesis was to propose roles and responsibilities related to the operations of providing a continuous software service. The case company is going to go through an audit to update its current quality certificate to the ISO 9001 certificate, due to which the company is seeking to develop selected process descriptions. Due to this process being included in the audit, it is especially important for the proposal to be suitable for its purpose as well as implemented by the time of the audit. The different sections included in the proposal building are visualized in Figure 20.

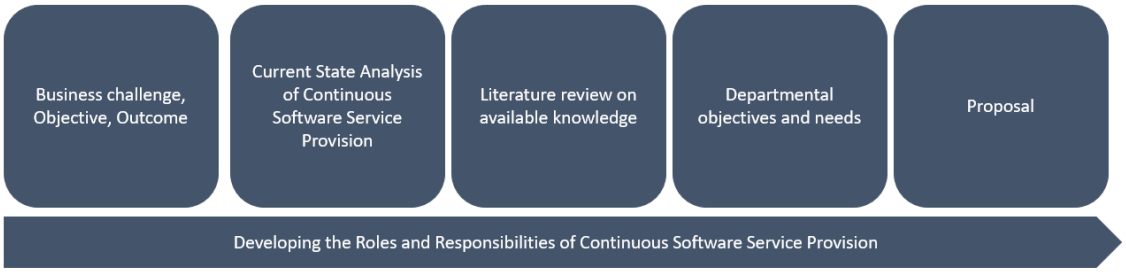


Figure 20. Building proposal

The study was carried out by first investigating the current state of the operations and their related responsibilities. Based on this, selected areas were defined for development. The key findings of the current state, which made the main development areas to be solved in the proposal were 1) lack of structure of the operations, 2) uneven distribution of tasks, and 3) lack of transparency regarding change management. Information was collected both through interviews with stakeholders and by participating in additional meetings. The current state analysis was presented in detail in Section 3.

The study included five different stages. After investigating the current state and identifying the main development areas, the study then drew knowledge from relevant literature to produ proposal. The insights gathered from stakeholders were carefully taken into account. The case company needs and requirements identified both by data collection and by working in the company were carefully included and assessed in the process of building the proposal. The research design is presented below in Figure 21.

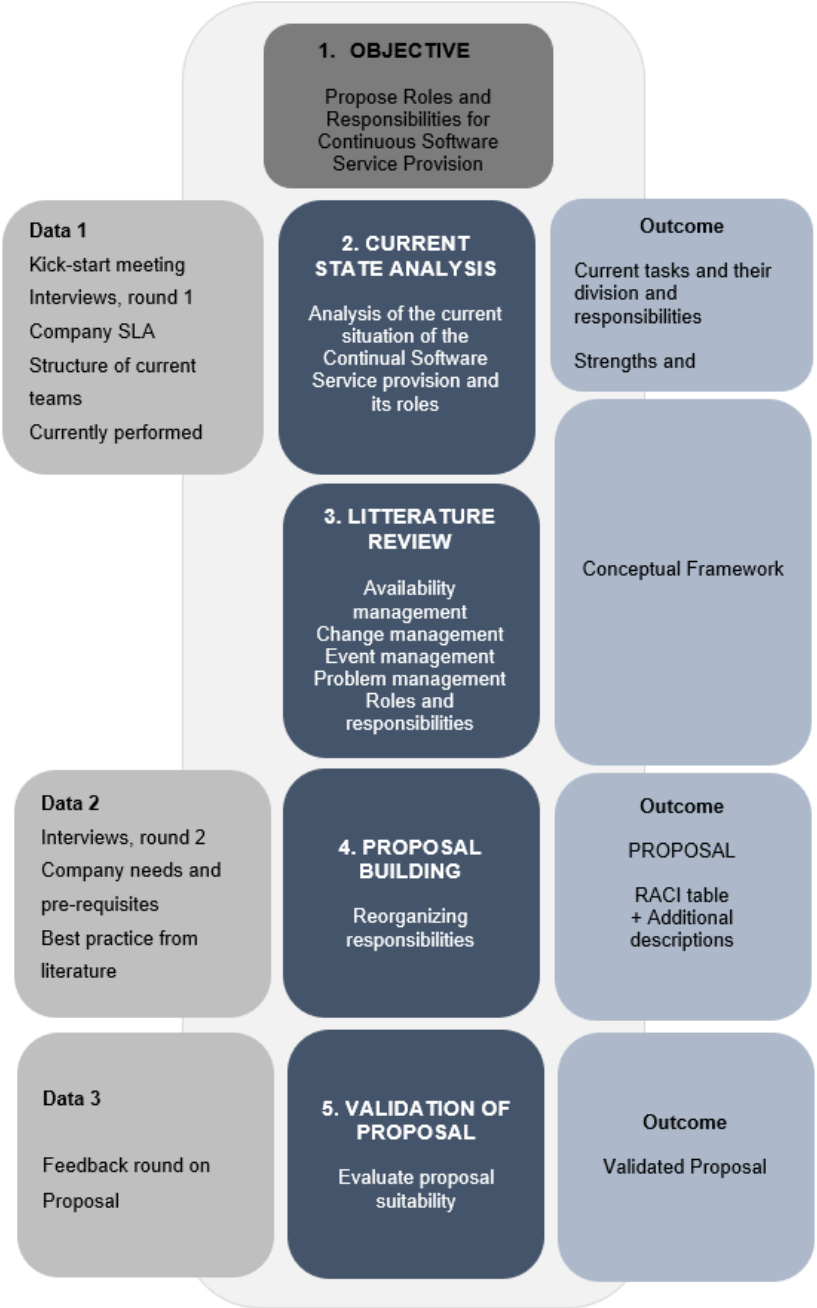


Figure 21. Research design

From the current state analysis the three key findings were identified. Most of the tasks were currently being conducted, but a structured way to perform these was missing. Analysis of many different types of alerts were being conducted, but the defined processing way, enabling further use of this information, was lacking. The division of responsibilities was also incomplete. From the current state analysis, tasks and their performers were identified. The responsibilities were, however, incomplete, due to which the performing actor was not always clear. The tasks were not distributed evenly, with most of them being performed by the CTO. There is very deep in-house knowledge attained in the department, but many of the tasks are very individualized. The SQL data content changes not being recorded in any specified way was an identified weakness due to the lack of transparency of what was done in the system. This was specifically a challenge in the case where problems occurred and there was a need to track the actions performed in the system. Figure 22 summarizes how the proposal was developed based on the three different findings.

FINDINGS	CHALLENGES	RELATED THEORY	PROPOSAL
1. Lack in structure of the operations	- Overall division of responsibilities is unclear - Many of the required measures are being conducted but not systematically	*4.1 Event management according to ITIL* *4.4 Availability management* *4.5 Organizing tasks and responsibilities*	Operations team RACI table
2. Uneven distribution of tasks	- Tasks are strongly individualized based on knowledge	*4.1.1 The process of event management* - *"How event management link to incident-, problem- and change management"*	Re-organizing tasks based on Operations team RACI table
3. Gap in transparency in change management	- How to include the SQL changes in the process maintaining the fast workaround?	*4.2 Problem management* *4.3 Change management*	Documenting procedure

Developing the Roles and Responsibilities for Continuous Software Service Provision

Figure 22. Key findings leading to proposal

The proposal created as a result of this study is visualized in Figure 23. The numbers in the figure indicate corresponding chapters. The proposal included an organizational aspect in the form of forming a new team called Operations. This team was formed out of individuals entailing knowledge about the SQL database and an individual from the product development team to function as the leader of the team. The roles and responsibilities of continuous software service provision were defined based on centralizing the main responsibility of continuous service provision to the new Operations team.

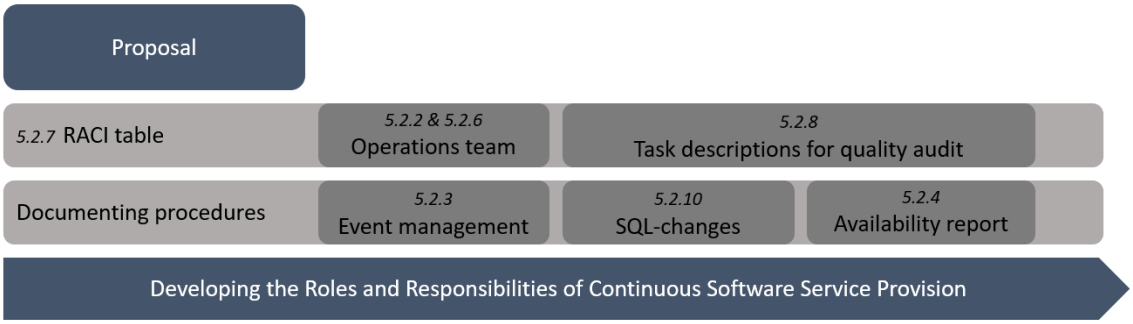


Figure 23. Summary of proposal

Each task was assessed in detail to ensure a defined and clear operating way was in place. During this process, some additional tasks to be listed individually were identified. In addition, the cooperation with the server provider was assessed in detail, and any unclear areas were resolved. The tasks and their responsibilities were visualized in a RACI table. The table also includes a short description and the interval to be conducted. In addition to the RACI table, supporting documentation in the form of task descriptions were created to fulfill the requirements of the quality audit. Documenting procedures were defined for both event management operations and for the SQL data content changes.

The Operations team was implemented and started its functioning as a virtual team during the process of creating the proposal. The ownership of the continuous software service provision tasks was transferred to the new team and the implementation will be fully completed in January in time for the auditing period. The expected benefits gained from the proposal are summarized in Figure 23 below.

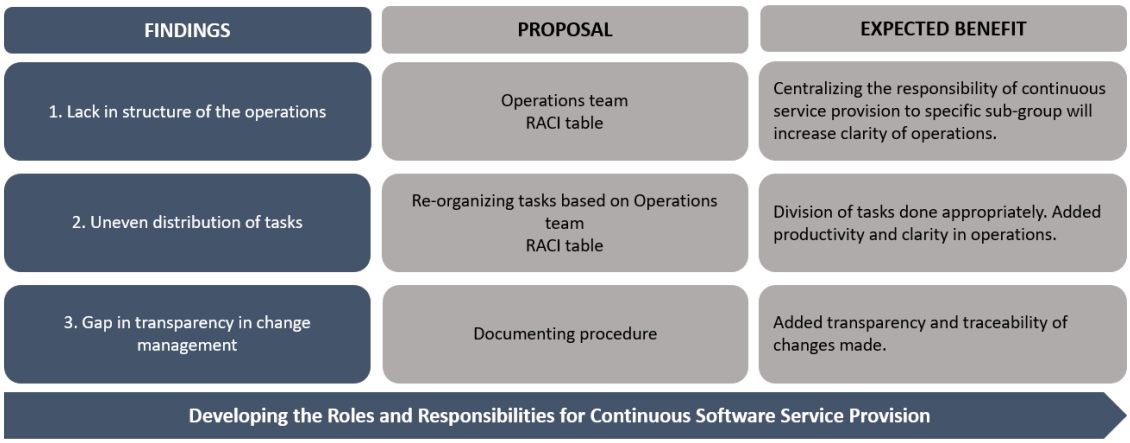


Figure 24. Expected benefits from the proposal

This proposal includes a structured and transparent way of conducting the continuous service provision operations. It includes both increased consistency in operations and a manageable way for performing the different tasks. With these clearly defined responsibilities, ownership for different tasks now exists. The proposal, including the related documentation, also fulfills the requirements of the quality audit

7.2 Next Steps and Tips for Implementation of the Proposal

As earlier mentioned, the proposed responsibilities were implemented during the validation of this thesis. The main responsibility of the continuous software service provision was transferred to the new Operations team. The next steps related to the implementation of the proposal are presented in Table 4.

Table 4. Next steps for Implementation

	Step	Content
1	In-house audit	Ensuring that the operations are conducted as they are defined prior to the actual quality audit.
2	Evaluation of the work amount of Operations team	Is the work amount optimal/too large? Are all the tasks being performed by the team their responsibility, or should some tasks be transferred elsewhere.
3	Development of the responsibilities of the Operations team	Transfer of the version update task. All continuous software service provision tasks centralized to the same team.

The first of the proposed next steps to be taken regarding the implementation, is to conduct an in-house audit. The organizing of this is already in progress. The benefit of this is that the operations are being conducted in the appropriate way in alignment with the defined process.

The second step to be proposed is an evaluation of the work amount of the Operations team. This would be good to conduct to ensure, that the team is not overloaded with work. This evaluation also might give some insight regarding the nature of work that they conduct, and possible identification of task types that should be transferred elsewhere.

A third step that should be taken is the transfer of the version update task to the Operation team. This should not be performed until the Operations team has the ability to acquire this task.

7.3 Thesis Evaluation: Objective vs. Results

The aim of this thesis was to develop the roles and responsibilities of continuous software service provision. Comparing the objective to the proposal created as the result of this thesis, it can be stated that the expectations have been met. The proposal clearly states the division of responsibilities regarding the operations related to the continuous software service provision.

An area that could have been done differently was the data collection. Data that could have been discovered at an earlier state, during the first data gathering round, was not discovered until the proposal building stage of the thesis. This would have been possible if the interview questions had been asked from a different, and deeper, angle. The process of conducting this study was a learning process. The required information was identified and sought after the understanding increased.

The individual assessment of each of the operational tasks in the process of developing the proposal was an area where great success was achieved. This assessment lead to additional, but very relevant, clarifications to be made. These clarifications were made regarding many smaller sections, which are also strongly related to the overall success of the continuous service provision.

This proposal was created specifically to suit the needs of the case company. The guidelines of the solution are nevertheless also applicable to other cases. The different areas included in the continuous service provision apply to all software provided as SaaS. This is due to the main focus being the maintenance of the appropriate functionality of the software. The guidelines of the solution are to some extent also applicable for other sort of software.

The research design created in the beginning of the study worked as a guideline to follow throughout the study. The outcome of the study was corresponding to the objective defined in the beginning. A broad range of individuals were interviewed for the data collection conducted at specified points during the research. The literature from which information was drawn from was relevant and gave excellent insight to be used for building the proposal. The reliability of the proposal can thus, on the whole, be stated as good.

7.4 Final Words

This thesis has been a valuable learning experience for me. I feel my understanding and professional thinking have developed and deepened significantly during the writing of this thesis.

This thesis has also given me an excellent opportunity to apply the theoretical knowledge I possess in a real life company case. In addition, I have had the opportunity to develop a section that brings value to the company and is an important part regarding the quality audit for the ISO 9001 and the ISO 14001 certificates.

References

BMC. (2016) *ITIL Change management.*
Available: http://www.bmc.com/guides/itil-change-management.html (Accessed 2 October, 2018)

Comella-Dorda, Santiago & Logiya, Swati & Speksnijder, Gerard. (2016).
An operating model for company-wide agile development.
Available:https://www.mckinsey.com/business-functions/digital-mckinsey/our-insights/an-operating-model-for-company-wide-agile-development (Accessed 9 October, 2018)

Gupta, Ashim. (2016). *An Effective Model of Cross-Functional Team*.
Available:http://www.practical-management.com/Organization-Development/Effective-Model-of-Cross-Functional-Team.html.
(Accessed 9 October, 2018)

Harrin, Elisabeth. (2018). *Pros and cons of project organizational structures.*
Available: https://www.thebalancecareers.com/pros-and-cons-of-project-organizational-structures-4105214
(Accessed 3 October, 2018)

Haverila, Matti j. & Uusi-Rauva, Erkki & Kouri, Ilkka & Miettinen, Asko. (2009).
Teollisuustalous. 6th edition. Tampere: Hämeen kirjapaino Oy.
ISBN 9789519676562

Cannon, David & Wheeldon, David. (2007). ITIL Service Operation. 2nd edition. United Kingdom: The Stationary Office.

Lloyd, Vernon & Rudd, Colin. (2007). ITIL Service Design. 2nd edition. United Kingdom: the Stationary Office.

Serban, Remus. (2017). *Getting Organized Through Authority, Responsibility, and*

Delegation. Available: https://www.hubgets.com/blog/getting-organized-authority-delegation/ (Accessed 2 October, 2018)

Software Project Monitoring and Control. (n.d.) *Organization and Team Structures.* Available: https://nptel.ac.in/courses/Webcourse-contents/IIT%20Kharagpur/Soft%20Engg/pdf/m12L30.pdf (Accessed 3 October, 2018)

UCISA. (n.d). ITIL- A guide to event management. UCISA

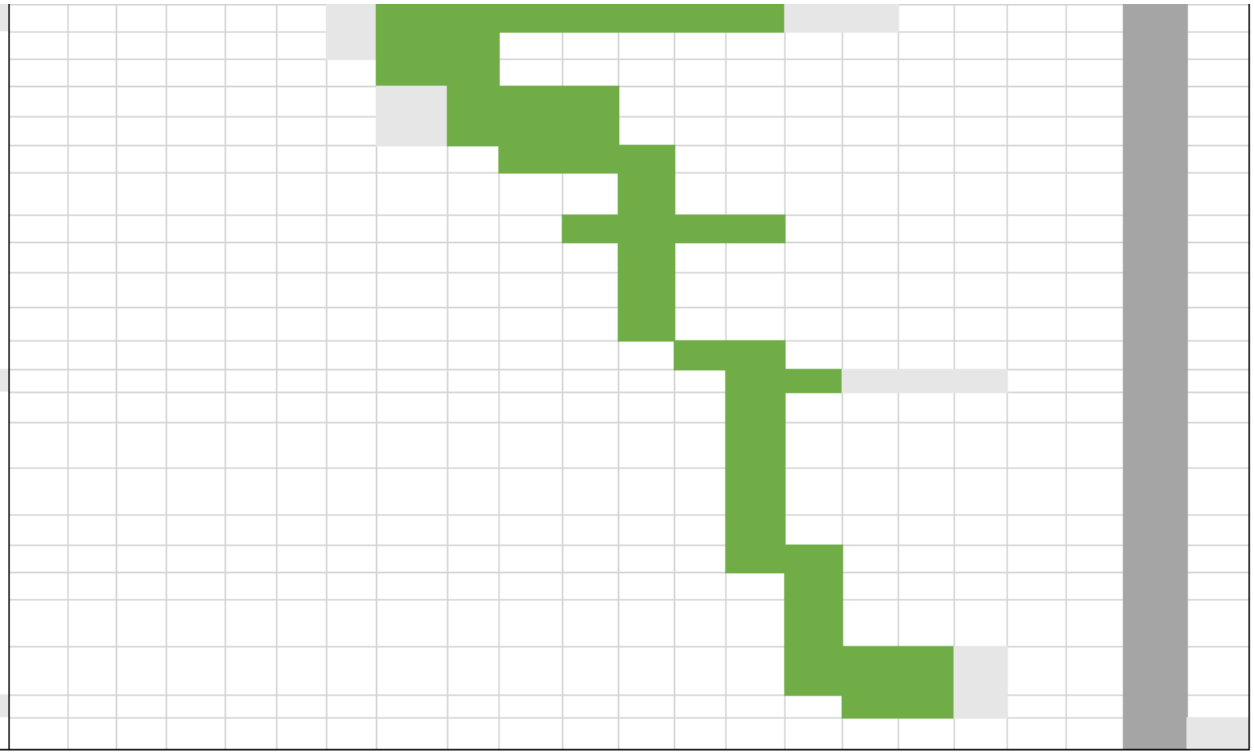
Appendix 1: Project plan, full version

Thesis, Project plan

Executed | Scheduled | Absent

	Month	August	September				October					November				December				January			
	Week	**35**	**36**	**37**	**38**	**39**	**40**	**41**	**42**	**43**	**44**	**45**	**46**	**47**	**48**	**49**	**50**	**51**	**52**	**1**	**2**	**3**	**4**
	Dates	27.8 - 2.9	3 - 9.9	10 - 16.9	17 - 23.9	24 - 30.9	1 - 7.10	8 - 14.10	15 - 21.10	22 - 28.10	29.10 - 4.11	5 - 11.11	12 - 18.11	19 - 25.11	26.11 - 2.12	3.12- 9.12	10.12 - 16.18	17.12 - 23.12	24.12- 30.12	31.12 - 6.1	7.1 - 13.1	14 - 20.1	21 - 27.1
	Task	3	3	3	2	3	3	3	5	3	3	2	3	3	4	4	5	5	2	3	5	0	5
1	Project guidelines																						
2	Project kick-off meeting																						
3	**Gate 1 Introduction**																						
4	**Gate 2 Method and Material**																						
5	**Gate 3: Current state analysis**																						
6	- Relevant company documents																						
7	- Case company SLA																						
8	- Workshop: Current maintenance operations, 5.9																						
9	- Steering meeting: 5.9																						
10	- Send interview invites																						
11	- Interview questions, Data 1																						
12	- Interview 1: CTO and Product development director, 7.9																						
13	- Interview 2: Expert, 11.9																						
14	- Interview 3: SQL Experts (2) 11.9																						
15	- Interview 4: Case company IT manager, 12.9																						
16	- Findings from current state																						
17	- Key findings identification																						
18	- Process chart																						
19	- Steering meeting: go-through of current state, 19.9																						
20	- Preliminary agenda																						
21	- Visual presentation of findings from current state																						
22	**Gate 4: Litterature review**																						
23	- Finalizing project plan for the rest of the project																						

24	**Gate 5: Proposal building**
25	- Proposal creation
26	- Server provider, define unclear areas
27	- Team Operations
28	- RACI versio 1
29	- Domain provider, define unclear areas
30	- Meeting: required documentation for quality audit, 13.11
31	- Documenting procedure for team Operations
32	- Steering meeting: go-through of Proposal, 14.11
33	- RACI changes
34	- Quality audit documentation of RACI
35	- Documenting procedure of error alerts
36	**Gate 6: Validation of proposal**
37	- Validation meeting of Proposal, 27.11
38	- Meeting: Team Operations documenting procedure with the team, 27.11
39	- Ensuring error alerts are recieved by all relevant individuals
40	- Meeting: Server provider
41	- Availability report, content
42	- Availability report, Final version + location
43	- Finalize all areas of the text reagarding the proposal
44	- Modify&Finalize thesis document based on feedback
45	**Gate 7: Summary of thesis**
46	Plan & Present Final proposal for department



Appendix 2: Field notes for CSA – Product manager, Customer support manager, Quality director

Interview Field Notes
Participant: Product manager, Customer support manager, Quality director
Date and time: 5 Sep, 2018 at 12am – 2 pm

Current maintenance operations

- Due to information not being shared and actions not being recorded unnecessary work is being conducted.
- Communication between the customer support centre and the product development isn't good enough
- Risk scenarios should be listed and required measures should be defined for each
- Currently the case company head of IT is included in but this should not be the case
- Areas included in the software maintenance process
    - Service monitoring
        - Availability of application
        - Integrations
    - Maintenance tasks (version updates)
    - Customer support (own process)
    - Incident management (own process)
    - Disaster recovery plan (own process)
- Increased structure in operations is required

Appendix 3: Interview field notes for CSA – Product Development Director and CTO

Interview Field Notes
Participant: Product Development Director, CTO
Date and time: 7 Sep, 2018 at 9 -10am

Question	Answer
1. What sort of rotating roles exist in the product development team? What does these different tasks include?	-Testing, user interface design, database. -Database manager and Release manager and CTO are the only identifiable roles currently. These roles are very individualized
2. What do you think about the idea of forming a sort of ”Back office”- team to manage all tasks related to continuous service provision?	- Might not be possible to transfer all tasks to a new team during a time span of half a year. - Monthly updates can’t be transferred in half a year. - The current focus should be on planning how this theoretical team would function. - The teammates should be included in planning the function.
3. Who are currently involved in the maintenance of the software? Which other should be involved and why?	- What services could the service provider regarding the servers provide us with what comes to maintenance. - Also it is a bit unclear what maintenance services are already included in the current contract. - Currently error alerts are received by multiple people in the software department, after which someone does an analysis on them. - Monitoring is currently not conducted systematically.
4. How are we currently monitoring that Software X is functioning accordingly?	- Software development conducts testing on the software but this actually doesn’t count as maintenance. - There is no systematic way to conduct monitoring, but monitoring is usually conducted on a weekly basis regarding the integration, error peaks and overall checking that the logs of the software look ok. In case the logs wouldn’t be alright, then it would indicate there being a problem in the backend system. - The possibility would be to install some tracking system. - There has been some planning on getting a system that would enable better logging. - Currently automatic spam alerts are received from the

	user interface, each of which need to be analysed. The person conducting this analysis is the CTO but it could be transferred to this so called Back office team. After initial analysis it could be then sent forward for further actions.
4.1 How do we monitor the availability in percentage? (97% based on SLA) What measures are taken in case the SLA isn't fulfilled?	- Automatic monitoring with a robot that sends an alert in case some website is down. It also counts the availability score for the last 30 days.
5. Is the current RACI-table with the responsibilities in between the case company and the server provider descriptive enough? Are there some scenarios that are unclear?	- That RACI table includes only the contractual responsibilities. - Not fully satisfied with the problem identification conducted by server provider. - No clear understanding on what the server provider actually monitors, since often we identify problems that are in their end. - There are multiple details that require additional clarification.
5.1 What instructions could we provide the server provider to enable them to function in the way we expect them to?	- ftp server booting - The informing responsibilities of the server provider should be defined. What are the things they should monitor and inform us about. - They should monitor the user interface and its functionality. - Very often we identify a problem that actually would be theirs to detect.
6. Are the responsibilities regarding the different servers defined and documented?	- In the production we only have one server provider. - In development we have another but it is unclear whether this cooperation will continue. - Responsibilities are defined but not clearly enough. - The hardware and the user interface are the responsibility of the server provider. - The application and everything on top of that is our responsibility. - There are some unclear parts, for example whether ftp and IIS servers are included in the user interface or not.
7. Do we have the ability to notice if an integration is down?	- Currently automatic alerts are received from some integrations, based on different criteria. - Automatic alerts are though not sent from all integrations

	that exist. - The message sent has to do with how the integration has been created, since there are multiple different ways. - It is not possible to create all integrations in the same way, but for example 3 different ways would be okay compared to our 50 current ones. - The automatic messages might also be received about successful integrations. Also the fault the alert informs about might not be business relevant. - Currently the integration alerts are sent to the CTO, customer support centre and one SQL expert. During the summer this SQL expert handled all of these messages and their analysis.
7.1 What measures are taken after an integration error alert is received?	- Analysis on what it informs about. - Integration monitoring should be a part of the so called Back office team.
8. Are there different people responsible for different servers in the case company?	- Product development is active regarding the servers that are relevant for our software department. - Local installations have responsible person's identifier in case specifically required. Product development does not have any part in these servers.
8.1 Should each server be defined with a responsible person? **8.2** Should there be a definition, that in case a new server is required, what all thing should be defined?	8.1 - Product development doesn't wish for any new servers. It might though be required for example regarding the international business. 8.2 - Installation instructions exist. A description on the hardware requirements also exists. - Expert X is able to install software on a new server.
9. Has a risk analysis been conducted?	No
10. Strengths of the current state?	- Best and most competent employees are conducting the tasks.
11. Weaknesses and biggest challenges of the current state?	- Currently the task are strongly individualized. - Some tasks can't be performed by anyone else.
12. Risks of the current state?	-

13. What measures are taken when doing a monthly update? What is done before and after a production server is updated?	- All of this is documented

Appendix 4: Field notes for CSA – Expert

Interview Field Notes
Participant: Expert
Date and time: 11 Sep, 2018 at 10 -11am

Current incident management process

- Incident management process currently consists of software development managing incidents in the change tool.
- Would require some actual prioritization to fulfill the idea of incident management.
- Current prioritization is based on customer bugs, either ”high showstopper” and ”non-urgent”
- At some point some criteria for different incident classes and defined reaction times should be defined.

Functioning of department

- Currently operations haven’t been organized well enough.
- “Someone will handle”- mentality
- Processes haven’t been described well enough and tasks are too personated.

Terms used in the case company

- ISO standard includes terminology that should be used.
- Usage of terms does though differ based on who you ask.
- From the software development point of view this can be described as following:
    - Error= error in code, either semantic or syntax. Missing definition. Fault in the code.
    - In case the faulty code isn’t run, the incident will never occur.
    - Incident is the only one that can be reported.
    - A person not being able to perform a specific action is experiencing a bug.

Appendix 5: Interview field notes for CSA – SQL Experts

Interview Field Notes
Participant: SQL Experts
Date and time: 11 Sep, 2018 at 10 – 11am

Question	Answer
1. Do you use the change management tool? Do you ever record the changes you have made in the change management tool?	-Not using the change management tool. If changes need to be made, then the customer support centre logs it in the tool. -Previously have created new change tickets in the tool but not anymore.
2. What sort of tasks do you handle? Where do you receive these from? Do you record changes made in the SQL database?	- One is an expert regarding energy integrations. - One has very deep SQL knowledge regarding the actual database. - Both do projects and maintenance of data content. - The changes made in the SQL data content do not affect the actual structure of the SQL database.
3. What sort of maintenance tasks do you do regarding the database? Are these tasks systematically performed?	-Maintenance of databases is strictly dependant on contact with customer. -Sometimes perform tasks for solving why some user isn't able to log in. - Each night all new customer databases that have been created during the last 24h are extracted with automation and sent by automatic email to customer support centre. - No systematic maintenance performed.
4. Can you think of any actions that could be performed regarding the SQL database, for us to ensure Software X is functioning properly?	-Automatic messages that are currently received from some, but not all, integrations. - To inspect logs for ensuring the proper functions of the software would require some automation. - Automatic alerts regarding the energy integrations is difficult to create due to the fact that there are so many different systems.

5. What actions are required regarding the China server?	-One of the expert has handled all tasks regarding China server. -Updates are conducted when asked for. -China server is a bit difficult since it functions differently than the ones in Finland. It is slower to manage. - Like any other local installation, the tasks performed regarding these differ based on agreement. China is though managed as a SaaS, since it includes multiple different customers.
6. Does the aforementioned also concern the other 4 SaaS servers?	-Yes - All of these servers are provided by the same server provider.
7. The case company head of IT is also currently involved in the SaaS documents. How is the responsibilities divided between him and the software department?	-In case we would be able to get a connection to the databases or servers then we would contact the head of IT. - This also applies in case VPN would be working. - Also in case we would run out of IP addresses.
4. Strengths of the current situation?	- Wide range of knowledge.
5. Weaknesses and challenges of the current situation?	-There is no clear view on who is responsible of what regarding the different servers. - Contact and task division between server provider and software department could - There should also be a clearer division of responsibilities regarding the software department.

Appendix 6: Interview field notes for CSA – Head of IT

Interview Field Notes
Participant: Head of IT
Date and time: 12 Sep, 2018 at 11-12am

Question	Answer
1. Does the server provider deliver servers as a service for other departments of the case company and not exclusively for the software department? 1.1 Are there any underpinning contracts? Have there been any definition of stricter contractual terms regarding the software department providing a software as a service for its customers?	- Yes - Everything is on the same contract. - There haven't been any definition or stricter contractual terms regarding SaaS, but probably should be. - Currently no MOG (Management operation guide) in place, but this is in progress.
2. How do the responsibilities differ between the server provider and the software department?	- Server provider is responsible regarding the user - interface and everything below that. Meaning hardware. - Software department is responsible regarding everything on top of the user interface, meaning SQL server and so on.
3. Have there been discussion about unclear areas regarding the service being bought?	- There was one incident that resulted in some discussion due to us expecting server provider to monitor and manage a specific server, but due o lack in instructions they weren't. - Product management and Product development should create these instructions since experts in the field.
4. How does the monitoring of the different production servers differ? Are all production servers being provided by the same server provider?	- Monitoring can be defined to include all of these servers.
5**.** How can we add clarity to the division of responsibilities in between server provider and software department?	- If the user interface doesn't start, then the server provider knows what measures to take. - But in case our software doesn't start, then we need to provide instructions for this scenario.
6.	- Systematically organized quality meetings are being

How can we get the unclear areas solved?	conducted monthly. - Software department have been asked to join. - In these meetings all unclear areas and development need can be discussed with the key account manager from the server provider. - Currently there has been the expectation that I (Head of IT) would manage also the software related things regarding the server, but this is responsibility of software department.
7. Is there the possibility to create a listing of everything the server provider monitors and what actions they take. This would enable all parts to be on the same page.	- Yes, this is what will be included in the MOG, management operation guide.
8. How should we proceed in solving the obscurities in the cooperation?	- In- house definition of the unclear parts - Quality meeting with the key account manager to solve these.
9. What sort of monitoring is being conducted by the server provider? What additional services could they offer us?	- They monitor remotely that the application is running. - Reports page including information about the servers. (CPU usage, processor capacity, alerts) - Disc space (SQL is added automatically since it is so crucial) - We are able to order additional services from them in case required.

Appendix 7: CSA – Service providers

Server provider

The case company has contracted a sub-contractor for delivering servers as a service. According to the case company IT manager, the server space is also provided for other business areas and not particularly for the software department. The contractual division of responsibilities, which is based on the service contract signed by the case company and the server provider, have been defined in a RACI-table.

The contact person from the case company towards the server provider is the case company IT-manager. This IT-manger is though not included in the software business area, and is due to this not obligated to communicate and manage the needs of the software department towards the sub-contractor. According to the IT manager, there are regular meetings being conducted with the server provider to monitor the quality and needs. Individuals from the software department have been invited to join these discussions.

From the Software X point of view, the server provider is responsible of the hardware and everything related to that. The provider is also responsible for the operating system. The responsibility of the case company software department includes the SQL database and everything situated on top of that.

The responsibilities currently haven't been defined on a level detailed enough, which is causing confusion in the software department. According to the IT manager, no MOC (management operation guide) has been created in between the parties. This means that actions the server provider is expected to conduct related to the Service X needs haven't been documented and communicated.

Domain provider

There is a sub-contractor for the company website, where the login screen to the SaaS solution is situated. This sub-contractor is providing the domain as a service on the behalf of the communications department having the responsibility of the company

website. In case of a restriction visiting the company website, the login to the software is possible from the actual login site that has a different web address.

Appendix 8: Field notes for Proposal building – Head of IT, CTO

Interview Field Notes
Participant: Head of IT, CTO
Date and time: 19 Oct, 2018 at 1.30 – 2.30 pm

Server backups

- In case maintenance has been ordered for the server, then backups of the server are being made.
- AD
  - There are two AD servers that are backups of each other.
  - Server provider makes backups of the AD on item level. Individual items can be restored when required. There is a scenario, where a Windows-update would break the AD. This would mean that the virtual environment would break down.
  - AD-server breaking would have catastrophic consequences.
  - Existing AD accounts have been identified.
- IIS
  - When user interface backups are being created, also backups of the IIS server/servers are being done.
- Application server backup
  - Application server backup means backup of the whole C-disc.
  - In case hard drive breaks down then situation is critical.
  - Backups of the whole system are currently being done in a secondary server room.
- SQL
  - SQL server is very critical. Backup of SQL server is being made as a sequential job:
    Case company .bak file job.
    Responsibility of case company to ensure that the .bak file job has been completed.
    Server provider backup from .bak file job.

    <u>Find out how the sequential jobs are currently scheduled?</u>
- SFTP and FTP
  - Case company responsibility.

- Servers without maintenance
    - Test servers

Critical notices regarding disaster recovery plan

- AD server hard disk should definitely not fill up.
- In case of disastrous situation case company is very dependent on the actions of the server provider.

Reports

- Access to reporting site provided by server provider. Enables monitoring of servers.

Next steps

- Meeting with all relevant individuals to discuss areas that require further definition.
- Quality meeting in a month. Concentrate on the departmental needs of the software department.
- Someone from the software department is responsible of participating in the monthly quality meetings with the server provider.

Appendix 9: Field notes for Proposal building – Head of Operations team

Interview Field Notes
Participant: Head of Operations team
Date and time: 21 Nov, 2018 at 12 – 1 pm

Documentation in the change management tool

- Go-through of possibilities
- All tasks should be received by specific email boxes
- Solution: log all tasks that should be taken care of in the change management tool (Operation team area)
    - Adds transparency in working
    - Enables follow up
- Kanban board solution
    - 3 types of tasks identified currently
    - Mother issues that the child issues (the created tasks) will be linked to
    - Flagging option to mark that task is waiting or actions from someone outside the Operations team

Appendix 10: Field notes for Validation of RACI-table

Interview Field Notes
Participant: all process relevant individuals
Date and time: 27 Nov, 2018 at 10.30 – 12am

Introduction

- How the RACI was created
    - Interviews and data gathering
    - Includes all actions conducted to ensure appropriate delivery of SaaS
    - Actors (product management, product development, customer support center, operations team, service provider and departmental manager)
    - Task, short description and interval
    - R, A, C, I

Walk-through of RACI + comments

- Release management
    - CTO: in the future the version updates will be moved to the Operations team to perform. Currently it will stay in the responsibilities of product development.
    - To create and publish scripts is two different things. Lines 4 and 5 (Version update and update-script creation) should change order to add logic.
    - Creation of scripts could be its own line in case we want to
    - Change instead to development of version and version update
- Change- & Incident management
    - Change interval of bug fixes from on a continuous basis to when required
    - CTO: Should the scope of the task changes in the data content in the SQL database be specified in greater details. Yes, add comment that based on ordered work.
- Disaster recovery plan
    - OK

- Event management
    - Error backlogs are regarding software bugs. Due to this the responsibilities should be changed to product development and operations could be consulted.
    - Automatic alerts from the user interface. There are also other types than regarding disc space, but the definition process regarding these is ongoing. When defined, these can be added as additional task rows below the disc space.

    - Energy integrations is a task that will be excluded at some point. Regarding the integrations we should at some point look at the possibility of adding automation to the monitoring of their functionality.
    - The monitoring of integration logs is conducted since there might be some issues that do not generate an automatic alert to be sent. Then the logs are the only place to find out about these.
- Monitoring
    - Server and database capabilities monitoring should be renamed to apply to only servers.
        - Very strongly related to the automatic messages sent from the user interface.
        - Server performance monitoring can be conducted either by viewing the server provider reports page or by following live situation from the logs of the server.
        - Some automation should be considered at some point.
        - Change interval to monthly/when required
    - SQL backup requires another row called Server backups
        - Regarding the .bak file job completion, some automation to simplify our work should be considered. That has already been looked into.
        - The responsibility of the server backup should be at the service provider
- Supplier management
    - The admin provider company does not need to be mentioned in this table, since it is no relevant in the service provision chain.

Appendix 11: Literature review – Organizational structures

4.5.1 Organizational structures

Teams can be organized in different ways. This chapter will discuss the project- and the functional organization structure as well as the cross-functional structure. The project approach is specifically suitable for project delivery. Each project team then attains all required knowledge for the delivery of a project. The functional format on the other hand means that teams are organized based on tasks, and each team has their specific responsibilities. (Organization and Team Structures, n.d.)

Functional structure

The functional organizational structure is considered the traditional operating model. Its main advantage is that resources and available knowledge are used in a coordinated way. A functional structure aligns the process, and the performance of the structure is usually high. (Haverila & Uusi-Rauva & Kouri, Miettinen 2009: 104) Regarding software production, the approach is especially suitable for small teams, since resources are easily managed. Collaboration and consulting of others is easy since each individual in the team is specialized on the same sorts of tasks. The overall management of the team in addition to reaching the objectives set for the team is easy. There are though also disadvantages which result in this approach not being suitable for all sorts of functions. Since tasks are divided in between teams, this means that these functions are conducted in different silos. Each team needs to have a team manager, and in case the managers of different teams do not work together, then this might cause some difficulties in reaching the overall strategic goals. (Harrin 2018)

Product-oriented organizational structure

In a product-oriented structure all resources are organized around the product, forming a structure that supports the deliverable from end to end. This sort of structure eliminates operating in the form of projects and instead supports the agile way of managing a product and its requirements. To support the product, an agile team should be formed. This team would traditionally include both product owners, developers and additional, and manage the needs regarding development and maintenance of product.

This team would then retain the main responsibility of maintaining the software, but would have the possibility to consult experts when required to. Role- or topic based teams are critical regarding agile software maintenance, since coordination and knowledge sharing in between the teams will be more effective and supporting the agile philosophy. (Comella-Dorda & Lohiya & Speksnijder 2016)

Continuously improving the performance is important, and strongly supported by the agile way of operating. It is important to unite individuals that understand the business context point of view with individuals attaining technical knowledge in the process of discussing improvements and changes regarding the software. Instead of one product owner managing the division of all tasks as in a waterfall model, each agile team performs self-organizing of the tasks assigned for them. Transparency in agile is a critical factor. There should be clear guidelines defined for each team regarding the decision making and expected outputs. (Comella-Dorda & Lohiya & Speksnijder 2016)

Cross-functional team

A cross-functional team consists of individuals that are reporting to different superiors, but who's knowledge is required in the team. Cross-functional teams can exist in an environment that is otherwise structured as a functional organization or a product organization that is less hierarchical. The need to form a cross-functional team is usually raised by the complexity of a task. The need usually is short-term, but it does occur that the need is permanent. (Gupta 2016)

There should be a cross-functional leader defined to maintain the focus of the team, also being responsible for taking the decisions. Regarding decision making there should be a clear division on to which extent the leader makes the decisions and when to let the team decide. Transparency is a key factor and communication is primarily conducted through regular meetings. (Gupta 2016)

The study will continue with presenting the expert insight that was gathered to be able to start the proposal building.

Appendix 12: Availability report template
(Only for Case Company use)

Appendix 13: Internal documentation, descriptions on continuous software service provision tasks
(Only for Case Company use)