LAUREA
AMMATTIKORKEAKOULU
Yhdessä enemmän

Version Control Systems - development, comprehension and rationalization of usage: Case Aalto University

Miettinen, Antti

2016 Laurea

Laurea University of Applied Sciences

Version Control Systems - development, comprehension and rationalization of usage: Case Aalto University

Miettinen, Antti
Information Systems
Master's Thesis
December, 2016

Laurea University of Applied Sciences
Degree Programme in Information Systems
Master's Thesis

Abstract

Miettinen, Antti

Version Control Systems - development, comprehension and rationalization of usage: Case Aalto University

Year 2016 Pages 67

IT environments are a continuously changing entirety. Software and systems in these environments go through their lifecycles and then they are replaced with more advanced and economical solutions. This is a normal phenomenon in education and corporate life. Thesis focuses on Aalto University's version control system's acquisition project and debate on how the version control systems can be developed, and how their usage can be comprehended and rationalized in the university's environment. The research method for this thesis was case study which consisted of three smaller case studies. Case study was chosen for this study because it is used for describing and making observations. All the studies were conducted by following Yin's case study research process: plan, design, prepare, collect, analyze and share. The main research question for the thesis was "How can the usage of a version control system be developed, comprehended and rationalized?".

The thesis was composed during Aalto's VCS project. The project aimed to acquire and provide a centrally managed system for the whole university, which could replace most of the current systems used in the different departments of the university. The thesis examines if and how the different systems could be replaced with a single version control system and how to prepare and execute an acquisition project of a new IT system. The smaller case studies were conducted during the project and its preparations. These studies examined the usage of version control systems at the university before the project, preparations of the project and the IT security methods used during the whole process. This was done by taking part in the processes and by using literature and studies related to the different tasks of the process. The comprehension was also completed by taking a part to some other tasks of the project.

The most important findings of the study are that users are ready to change their VCS tools to new ones, if they get to manage their own repositories. The departments are ready to renounce their own systems if the new system has all the same functions as the old systems. The reduction of the overlapping systems will also save money and the resources of the departments. These findings and the results of the smaller studies can be used as a general guideline, in a similar type of projects, in the university or outside of it. This study could be extended with the results of those projects.

The topic of the thesis is topical as always. The reducing funding of the schools, the competition in the corporate life and digitalization drives both to look for new, cheaper solutions, with as much functions as possible. By reducing overlapping systems and by acquiring and cheaper solutions, both can reduce their expenses and the thesis tells them how it could be actualized.

Keywords: Version control system, Acquisition, Project, Rationalization

Laurea-ammattikorkeakoulu **Tiivistelmä**
Tietojärjestelmäosaamisen koulutusohjelma
Tradenomi (Ylempi AMK)

Miettinen, Antti

Versionhallintajärjestelmät - Kehittäminen, käytön ymmärtäminen ja järkeistäminen - Tapaus Aalto-yliopisto

Vuosi 2016 Sivumäärä 67

Tietotekniikkaympäristöt ovat jatkuvasti muuttuva kokonaisuus. Niiden ohjelmistot ja järjestelmät menevät elinkaariensa läpi ja sitten ne korvataan kehittyneemmillä ja taloudellisemmilla ratkaisuilla. Tämä on normaali ilmiö kouluissa ja yrityselämässä. Opinnäytetyö kohdistuu Aalto yliopiston versionhallintajärjestelmän hankintaprojektiin ja keskustelee siitä, miten versionhallintajärjestelmiä voidaan kehittää ja miten niiden käyttöä voidaan ymmärtää ja järkeistää yliopiston ympäristössä. Tutkimusmenetelmänä tätä tutkielmaa varten oli tapaustutkimus, joka muodostettiin kolmesta pienemmästä tapaustutkimuksesta. Tapaustutkimus valittiin sen vuoksi, että sitä käytetään kuvaamaan asioita ja tekemään havaintoja. Kaikki tehdyt tapaustutkimukset noudattivat Yinin tapaustutkimusprosessia: suunnittele, mallinna, valmistele, kerää, analysoi ja julkaise. Tämän opinnäytetyön päätutkimuskysymys oli ”kuinka versionhallintajärjestelmän käyttöä voidaan kehittää, ymmärtää ja järkeistää?”.

Opinnäytetyö tehtiin Aallon versionhallintaprojektin aikana. Projekti pyrki hankkimaan ja tarjoamaan keskitetysti hallitun järjestelmän koko yliopistoa varten, joka voisi korvata useimmat nykyiset järjestelmät, joita käytetään yliopiston eri kouluilla ja osastoilla. Tutkielma tutkii, jos ja miten eri järjestelmät voitaisiin korvata yksittäisellä versionhallintajärjestelmällä, ja kuinka uuden tietotekniikkajärjestelmän hankintaprojektin tulisi valmistautua ja kuinka se tulisi suorittaa. Jo aiemmin mainitut kolme pienempää tapaustutkimusta tehtiin projektin esivalmistelujen ja varsinaisen projektin aikana. Nämä tutkimukset tutkivat versionhallinnan käyttöä ennen projektia, projektin esivalmisteluja, sekä koko prosessin tietoturvatoimia. Aineistoa tutkimuksiin kerättiin osallistumalla prosesseihin, sekä kirjallisuuden ja tutkimusten avulla. Ymmärrystä täydennettiin myös osallistumalla muutamiin muihin projektin askareisiin.

Tutkimuksen tärkeimmät havainnot ovat, että käyttäjät ovat valmiita vaihtamaan tällä hetkellä käyttämänsä versionhallintajärjestelmät toisiin järjestelmiin, sillä ehdolla, että he saavat päättää omien tietovarastojensa pääsynhallinnasta. Koulut ja niiden osastot ovat valmiita luopumaan omista järjestelmistänsä, niin kauan kuin uudella järjestelmällä on kaikki samat toiminnot kuin vanhalla järjestelmällä. Päällekkäisten järjestelmien väheneminen myös säästää rahaa ja resursseja. Näitä pienempien tutkimusten havaintoja ja löytöjä voidaan käyttää yleisinä ohjeistuksina saman tyyppisissä projekteissa, yliopistolla tai sen ulkopuolella. Tätä tutkimusta voitaisiin laajentaa noiden projektien tuloksilla.

Opinnäytetyön aihe on aina ajankohtainen. Koulujen alati vähenevä rahoitus, kilpailu yrityselämässä ja digitalisaatio ajavat molempia yrittämään löytää uusia, halvempia ja monipuolisempia ratkaisuja. Vähentämällä päällekkäisiä järjestelmiä ja hankkimalla halvempia ratkaisuja molemmat voivat vähentää kulujansa. Tämä opinnäytetyö kertoo, kuinka se voitaisiin toteuttaa.

Asiasanat: Versionhallintajärjestelmä, Hankinta, Projekti, Järkeistäminen

Table of Contents

Abbreviations

CS = Case Study Research

CVCS = Centralized Version Control System

DVCS = Distributed Version Control

Git = Version Control System software

GitLab = Version Control System software

HAKA = The identity federation of the Finnish universities, polytechnics and research institutions

HIIT = Helsinki Institute for Information Technology

IRC = Internet Relay Chat

IT = Information Technology

LDAP = Lightweight Directory Access Protocol

SGPP = Steering Group of Projects and Services of Aalto IT

SVN = Apache Subversion, version controls system software

VCS = Version Control System

1 Introduction

The subject of this research is development, comprehension and rationalization of the usage of version control systems in Aalto University. The objective of the research is to identify, how the reduction of overlapping version control systems will affect on the usage, development and support of the remaining systems.

The departments at Aalto University currently host more than 10 different version control systems. Hosting and administrating these systems takes out time from development and user support. The aim of the project is to offer one centrally managed version control system for the staff, students and other personnel at Aalto University.

Time consumed administrating multiple systems can be routed to development of centrally managed system, and at the same time, costs for maintaining several systems will be reduced as the staff costs and expenditure on equipment will also descent. Since this study is closely related to information technology, it has plenty of information technology abbreviations. These abbreviations have been reviewed before the introduction section. The single most important result of the research is that the reduction of overlapping systems will significantly improve the usage of the system and the local support provided for the system. At the same time, reduction will also offer a possibility to focus the development of systems instead of just focusing on administrating multiple systems.

1.1 Operational environment and context of study

Aalto University was established in 2010 when 3 Finnish universities (the Helsinki School of Economics, Helsinki University of Technology and the University of Art and Design Helsinki) were merged together. Currently, the university consists of six schools and two other units. These schools each have a small number of IT personnel of their own. The schools have local IT systems, network environments and access to network environment that covers the entire university, Aalto-network.

The university also has a centralized IT department, which offers its services for the whole university. These services are such as network services, software and IT purchases to mention but a few. The IT department is responsible for the Aalto-network. The IT department consists roughly of 120 employees, who are divided into different units, that are all focused on different areas of information technology. These units' upkeep and maintain the current systems and develop and acquire new systems. IT department also provides IT support in the form of IT service desk. The service desk consists of 7 units, which are located on different campuses and buildings on campuses. The rest of the department works in one building.

The workstations supported by the IT department and that are connected to the Aalto-network have Linux, OSX or Windows operating systems and the department has acquired licenses for a large group of most commonly used programs. IT Department will also install self-acquired software case-by-case. Version control systems are used widely in the university and the main user group consists of students and researchers.

The main use for version control among the students is to keep track of the software code, which they develop for the courses held at university. The most used version control software among students is Git. The second largest user group at the university are researchers and research groups. Researchers either use a version control system provided by their department, or a web based service. Research groups use both. A third group actively using version control systems are the administrators at the university. Administrators store puppet repositories in the GitLab version control system and use it to share settings and programs to department's Linux environment.

Currently, all the version control environments are administrated by different departments. One department may host more than one version control environment depending what the staff has required from the local administrators. The usage of the different systems in the 2015 is shown in the Figure 1 (Lähteenmäki, 2015a).

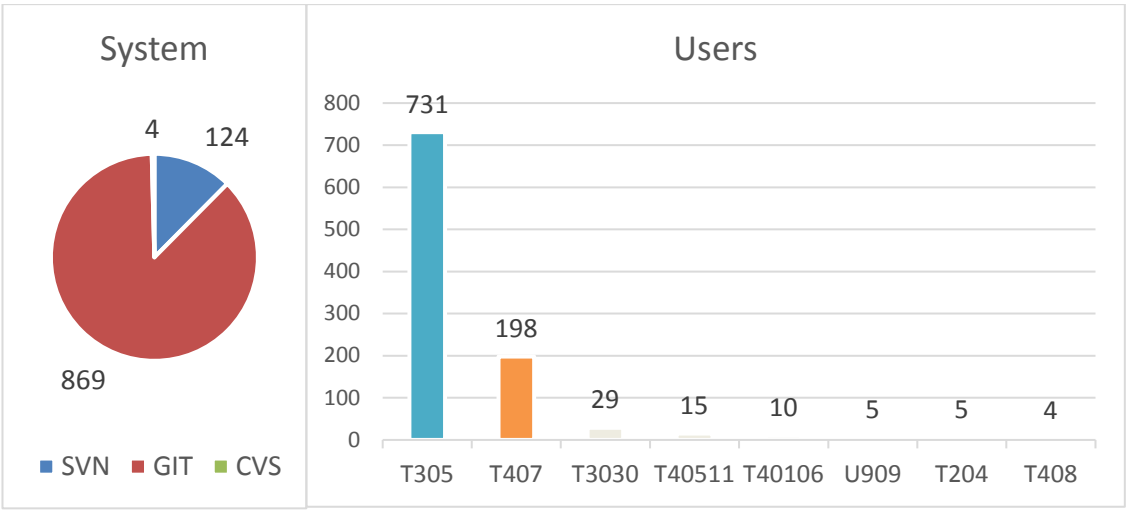

Figure 1 : Statistic of different system usage based on interviews and preliminary report

Figure 1 shows the usage of current systems in Aalto University. The figure clearly shows that version control systems based on GIT-technology are by far the most popular among the users of the university. Almost 87% of the users using VCS's are using GIT, 12% are using SVN and 1% are still using CVS which was developed in the mid 80's.

The current situation is that the university is trying to reduce the number of the overlapping systems. Sometimes this can be done simply by shutting down some of them and other times the current systems need to be replaced with a new and possible better system before the old one can be shut down.

Aalto University has very little or no written information about some of its processes. There are documentations and employees also have experience and information, but there are risks that this information could vanish. The university is constantly forced to reduce the number of staff due to the ever-decreasing funding and thus making work assignments and processes more efficient. This might lead in the future to a situation where some of the core employees have left the university or have forgotten the information. In this case, the study will become useful. It will provide written information, which can be used in later projects.

1.2 Theme and setting of the study

The subject of the study is "How can the usage of version control systems be developed, comprehended and rationalized in university's operational environment?" and the study takes place in Aalto University's Otaniemi campus between spring 2015 and fall 2016. Study focuses on the current version control systems administrated at different schools of the university and the new VCS that the centralized IT of university will purchase and administrate. The study aims to define the acquisition process of the new system, so that the results of this study can be used in future acquisition projects. The study explores the situation of the systems before the acquisition and various parts of the acquisition project. The theme of the study is to build a general view of the VCS acquisition process, so that findings can be used as a guideline in future projects.

Realization of this will be constructed with both theory and by participating in the acquisition process. The research methodologies, research questions, and analysis methods are described in detail later in this study. The methodological approach to this study is case study research (CS). "A qualitative case study examines a phenomenon within its real-life context. Data are collected on or about a single individual, group, or event." (Guest, Namey, & Mitchell, 2012, p. 14.) A series of case studies were conducted to obtain knowledge to answer the research question of this study. These case studies are described in Chapter 3 of the study.

1.3 Research environment and reasoning

The study was compiled prior and during an ongoing VCS project at Aalto University. The research was a qualitative study as the people and the system under study were studied in the context where they normally act (Kaplan & Maxwell, 2005, p. 30; Runeson & Höst, 2009, p.131).

As mentioned before, several different version control systems are maintained at the premises of Aalto University, and the reasoning for the VCS project is to reduce the amount of overlapping systems. By reducing the number of systems administrated, the contribution of local administrators can be focused on more relevant systems.

The primary goal for this project is to create a centralized version control system for the whole university. The secondary goal for this project, and the primary goal for this research is to provide knowledge how the overlapping systems can be reduced and what kind of steps can be taken prior and during the project to have a successful project. The aim of the research is to provide such information and methods, that they can be replicated in similar projects at the university.

1.4 Expected results & Utilization

The results of this study will include data about the tasks which should be done prior, during and after a version control acquisition project. Research will gather data about the composition of the project group and what kind of knowledge is required by the members of the project in different phases of the project to get the full gain of the people participating in the project. The results of this study are expected to give beneficial information about project management. How the project group can be formed successfully, what are the necessary and beneficial tasks that should be done before the actual project, what are the necessary tasks during the actual VCS acquisition project and how they should be done to complete the project successfully. What kind of unexpected changes can there be during a project and how the project should handle them.

The results of this research should be usable during similar projects within the premises of Aalto University. On the other hand, even though the study has been composed at the university environment, there is no reason, why the results could not be used outside the university. The same methods can be adapted to another universities or the corporate life.

1.5 Literature review

The study leaned heavily on literature and theories presented in it. Methodologies used in this study were based on the literature about each methodological procedure. Literature was used to gain knowledge about the processes and how to analyze the data received from the case studies. The two most used literature subjects during the studies were literature about the methodologies and literature about version control systems and their use.

The most used subject for literature was literature regarding different methodologies and use of them for research purpose. The whole study was a Case study methodology was adapted from Yin's book, Case study research: Design and methods (Yin, 2009). The book presents case study methodology and means for a successful study. Due to the qualitative nature of the study the sage encyclopedia of qualitative research methods (Given, 2008), was used as a guidebook. The book elaborates information, techniques, and examples about the qualitative study research. Since the study was about information systems (IS), Design Research in Information Systems by Hevner and Chatterjee. This book is about Design Science Research (DSR) and covers such topics as people and design, the past and present of software designs, evaluation methods and focus-group use (Hevner & Chatterjee, 2010). The research framework was also adapted from Hevner, from his A Three Cycle View of Design Science Research, which analyzes design research science with a help of three cycles of activities (Hevner, 2007). Also, other literature, such as books and articles were used to support the theory presented in these books.

The second most used subject for literature discussed about version control systems, and the use of them as a course platform. A study compiled by Lassi Haaranen et al. (Haaranen & Lehtinen, 2015) regarding the usage of GIT as a course platform lead the researchers of this thesis to familiarize themselves to other studies where the usage of version control system as study platform was studied (Biňas, 2013; Kelleher, 2014; Kertész, 2015). A fair amount of studies regarding VCS usage as study platform have been released in recent years. Using VCS as study platform could be considered as one of the development objects, therefore the same subject will be a part of this study too.

The third literature subject discoursed on process management. The second study addressed the problems with the project preparations and processes completed during the preparations. During the second study, the research focused on practices, how the processes could be optimized so that the costs during the project could be reduced. The success of a project should not be measured merely by financial facts (Davenport & Beers, 1995, p. 58), but rather with other factors, such as customer satisfaction. When the processes are optimized, the project proceeds fluently and the product, this case a new VCS can be made available to the customers without unnecessary delays.

1.6 Evolution of VCS

The trajectory of version control systems can roughly be divided into three different generations, 1st generation, 2nd generation and 3rd generation. First version control systems were developed in the early 1970's when Source code control system (SCSS) was released. It was developed to help programmers control changes in the source code currently in development (Rochkind, 1975, p.364). Since then there have been a great number of version control systems and the current systems in daily use represent the 3rd generation of the software.

The era of 1st generation systems spanned from early 1970's to mid-1980's. Typical for these systems were that the software stored data locally on computer and used locking method as a conflict resolution. The 2nd generation systems started to arise at the mid 1980's, replacing the old 1st generation software.

The era of the 2nd generation VCS continued till 1999. First feature that the 2nd generation systems introduced, were a centralized client-server data model which required an active network connection to function properly. The usage of the 1st generation software required that all developers of a project had to be on the same machine as the single central project repository, and the 2nd generation software changed that by allowing developers to access the single repository from another machine over the network. The second introduced feature was merge before commit as a conflict resolution. In the 1st generation software, the files were in read-only format and when someone wanted to edit a file the system would make the file writable and lock it, so that no one else could edit it at the same time. In the 2nd generation software, the system noticed when a file has been changed during the time a person has been editing it and requires that the conflict is resolved before the file can be saved. The reason for this is that someone else edited the file at the same time and saved the file (Raymond, 2007; Ruparelia, 2010, p. 5).

The 3rd and the current generation of VCS began in 2000, when Bitkeeper was released. Since then, a number of different version control systems have been released which all have some different functionalities and target audiences. The biggest overhaul when comparing 2nd and 3rd generation systems is the decentralized data model. In a decentralized repository model, both, the server and the client computer has a version of the repository. As the client computer, has also an offline version of the repository, files and documents stored on the local repository can be edited even when the computer does not have access to the network.

The changes can be committed to the repository on a server when the computer is connected to the network. The conflict resolution method as also improved between generations. When the 1st generation systems locked the file when a user was editing it, the newer systems allow multiple users to access and edit the same document the same time. In multiple edit – scenarios, when user tries to commit changes to the repository at the server, the system will inform user that the repository on the server has already been edited and the system will allow user to merge the changes he has done to the repository on the server (Sink, 2011, p. 1). A list of some of the most notable version control systems can be seen in table 1.

System name	Release year	Conflict resolution	Model
SCCS	1972	Locking	Local access
RCS	1982	Locking	Local access
CVS	1985	Merge-before-commit	Centralized client – server
Subversion	2000	Merge or locking	Client – server
Bitkeeper	2000	Commit-before-merge	Distributed
Mercurial	2005	Merge	Distributed
GIT	2005	Merge	Distributed

Table 1 Some of the most notable VCS releases (Chacon & Straub, 2014; Raymond, 2007; TutorialConflict, 2013)

Table 1 presents some of the most notable version control systems and the year were released. The table also shows the evolution of conflict resolution method, the early systems prevented conflicts by locking the edited file for one editor, current systems have built-in merge functions which can be used in a case of conflict resolution.

1.7 Research questions, unit of analysis and objectives of the study

A research question is a statement that announces the phenomenon that will be studied to understand it. This was the very first that was decided at the beginning of the study. "The first step in the survey process is to determine the research objectives. The researchers must agree on a well-defined set of research objectives. These are then translated into a set of key research questions. For each research question one or more survey questions are then formulated, depending on the goal of the study." (De Leeuw & Dillman, 2008, p. 4.)

In this study, the research question is "How can the usage of a version control system be developed, comprehended and rationalized?" the answer to this question was found with a group of smaller questions asked during the data collection process. "Research questions are usually distinguished from the questions researchers actually ask participants in interviews or in the course of field observations or ask of data in the course of analyzing them. Although the questions participants answer and the constant questioning process that defines qualitative data collection and analysis are in the service of answering research questions, they are not equivalent to them." (Given, 2008, p. 787.)

When analyzing the research data, the first step is choosing the unit of analysis. This will tell how the data should be analyzed. It is the phenomenon that is analyzed in the study. The unit of analysis was quite clear after the main research question was set. As Yin stated in his book "Selection of the appropriate unit of analysis will start to occur when you accurately specify your primary research questions." (Yin, 2009, p. 30.) The unit of analysis in the study is a

version control system. In a qualitative research process, all the smaller studies can have their own units of analysis, which will help to form the primary unit of analysis. "As a general guide, your tentative definition of the unit of analysis (which is the same as the definition of the "case") is related to the way you have defined your initial research questions." (Yin, 2009, p. 30.) In this case the unit of analysis is the same during the whole study and it was used in the analysis phase of the data collection processes of the smaller studies. The objectives of this research were to discover answers to the research questions represented in each of three special questions. The research questions are listed below, in table 2, organized by special questions. Answers to the questions 1 and 2 were examined by both researchers and both researchers had their own question 3, which they examined separately.

Special question 1	How could the usage of version control systems be improved in Aalto University?
Special question 2	What are the preparations prior a version control system project?
Special question 3	What are the IT-security analyzing methods for a new version control system? What is the analyzing process when comparing the security features of a version control system to the university's IT-security requirements?

Table 2 Special questions

2 Research methodology

In this chapter, the researchers will discuss the methodologies utilized during each step of the project. Methodologies utilized in each process, will also be assimilated to the research literature published about each methodology to justify the usage of chosen methods during the study in hand. The four most fundamental methodologies applied in the study will be presented later in Chapter 2.1.1.

2.1 Purpose of the study

The purpose of this study is to understand the phenomenon called a version control system (later abbreviated as VCS). The study aims to understand the VCS in theory and how could a new VCS be acquired and integrated into a current IT environment and what kind of procedures that inquires. Rationalization for the study was to determine the best practices for implementing a new system alongside existing systems.

The project consisted of several different tasks which all relied on different types of methodologies. In the following chapters, we will explain the tasks and methodologies used in different projects. In the future, the results of this study can be utilized in similar projects to avoid possible problems appointed in this study. With the help of the results of this study, project managers can get acquainted with the methods found useful in both data gathering and processing.

At the beginning of the study, the data collection was decided to implement with a qualitative research method. Since the focus of the data was in quality instead of quantity. John Creswell defines qualitative research in the following way: "Qualitative research is a means for exploring and understanding the meaning individuals or groups ascribe to a social or human problem. The process of research involves emerging questions and procedures, data typically collected in the participant's setting, data analysis inductively building from particulars to general themes, and the researcher making interpretations of the meaning of the data." (Creswell, 2013, p. 4.)

The study was composed by combining several smaller studies. During the study, the staff and administrators of Aalto University were interviewed and based on the interview results in a VCS project was engineered. The study will also debate on the processes adopted before and during the actual VCS project.

2.1.1 Research approach

The theme for this study is to comprehend, rationalize and develop the usage of version control systems at Aalto University. The methodology for the entire study is case study research. The case study consisted of several minor studies which all applied their own methodologies. The common nominator for all the separate studies is a qualitative approach to the study results. As the studies were conducted as a qualitative study, the research approach to the studies was a general inductive approach. Per David R. Thomas, general inductive approach can be seen as a simple, nontechnical method for analyzing qualitative data (Thomas, 2006, p. 245). The aim of using a general inductive approach in this study was to make broad generalizations based on the study results discovered during each individual study. Based on the generalizations, the

main theme of the study could be answered. The Unit of analysis (UoA) in this study is a version control system.

The entire study was a compilation of three minor studies. The results of the minor studies were analyzed to answer the research question of the main theme of the study. The most fundamental methodologies and use of purpose of using them, alongside the ones mentioned earlier, will be described in the following paragraphs.

The first study was conducted to understand the current situation of version control usage at the university. This was a focus group research, which was conducted by making semi-structured interviews. The results were used to supplement a preliminary report conducted before the project. Focus group research suited well for the purposes of the semi-structured interviews as focus group research encompass a wide range of discursive practices, spanning from structured interviews with delimited topics to open-ended conversations (Kamberelis & Dimitriadis, 2013, p. 4).

The second study focused on the preparations of the version control system acquisition project. The second study was a participatory research, where the data was gathered with literature and by taking part to the tasks that were done during the preparations. Participatory research emphasizes the participation, contribution empowerment and emancipation of all relevant parties in examining a common problem (Reilly, 2010, p. 558). The tasks that were examined during the preparation were market analysis and feasibility study where the possible future systems were inserted into a matrix and compared against the requirements which were based on user interviews and the preliminary report.

The third study was conducted to understand the IT Security aspects of the project. This study was also a participatory research and the data was gathered with literature and by taking part to the IT security related tasks that were done during the preparations and the actual project. These tasks were IT security exploration for the suggestion of investment, risk analysis and information security check with IT security framework. These tasks examined the IT security features of version control systems and analyzed risks and compared the security features of the new version control system with IT security framework. Risk analysis and framework comparison both used an iteration method, where the data were compared with the demands and discovered faults were corrected and the data was analyzed again until the results were evaluated admissible. Like Srivastava and Hopwood note in their article, reflexive iteration visits and revisits the data and connects them with emerging insights, which leads to better focus and understandings (Srivastava & Hopwood, 2009, p. 77).

2.1.2 Framework applied in the research

The framework applied to the study as an analytical tool was Hevner's Design Science Research Cycles (Hevner, 2007, p. 2), seen in the figure 2.

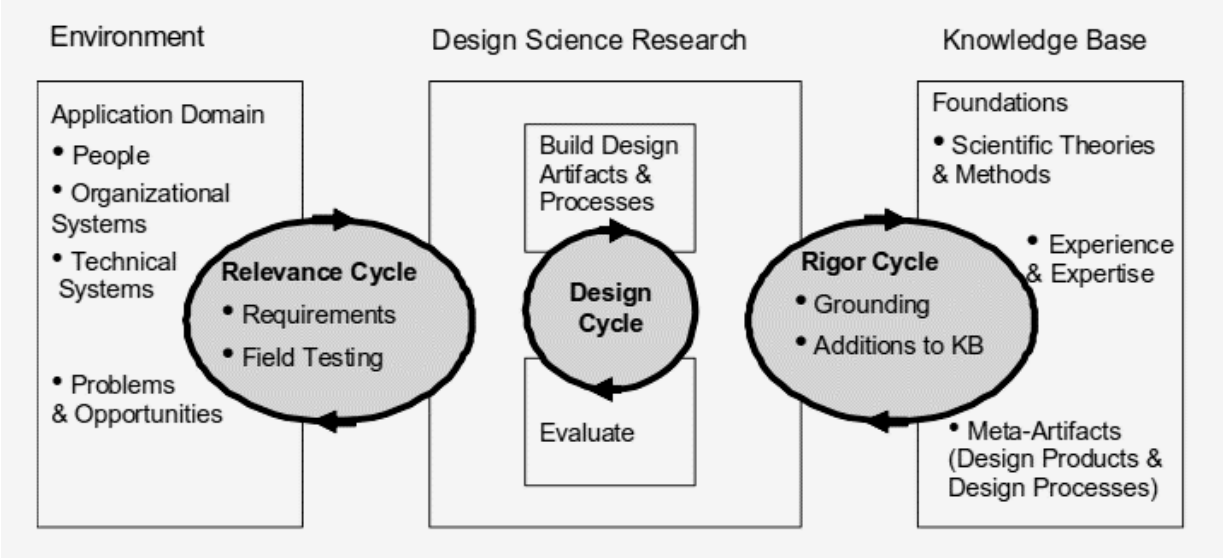

Figure 2: Design Science Research Cycles per Hevner (Hevner, 2007, p. 2)

The framework consists of three cycles, which are used throughout a design research project. The arrows in each cycle symbols an iteration process. Iteration is used until the outcome of each process fulfills the set demands.

The relevance cycle connects the specific environment with the research. It is used to examine the environment's requirements and for the field testing of the product, in this case the VCS. The design cycle iterates between product building and its evaluation and the rigor cycle connects the knowledge database of scientific theories, experience and expertise to the research and product building. The cycle model is used from the beginning of the study and it is used until the new product is ready to use.

2.1.3 Triangulation

Lisa M. Given states in the book she has edited the following "Triangulation in qualitative research has come to mean a multimethod approach to data collection and data analysis. The basic idea underpinning the concept of triangulation is that the phenomena under study can be understood best when approached with a variety or a combination of research methods." (Given, 2008, p. 892.) The types of triangulation that were used were multiple investigator triangulation, theory triangulation, the triangulation of methods and triangulation of data

sources (Given, 2008, p. 893). In multiple investigator triangulation, more than one investigator collected and analyzed the data.

1. In theory triangulation, the researchers examined data from multiple theory sources.
2. The triangulation of methods, the researchers used multiple research methods.
3. During the triangulation of data, the researchers collected data from several sources, such as written documents and interviews.

The theory triangulation examined the theory of version control systems with multiple literature resources, such as Pro GIT (Chacon & Straub, 2014) and Version Control by Example (Sink, 2011). With the help of the theory sources, the researchers acquired an understanding of the version control systems, how they function and how they were evolved during the years. Multiple sources were used to avoid a unilateral and concise view of the systems.

In triangulation of methods, multiple research methods were used to acquire a sufficient amount of explicit data. These methods were also used to avoid making incorrect assumptions and other errors. This way the research process proceeded logically and with right amount of pace. These methods included such research methods as case study, a market survey, semi-structured interviews, surveys, qualitative data analysis, theory analysis, feasibility study and participation in to the VCS project.

In the triangulation of data, Multiple source of data were used to provide sufficient amount of research data and extensive conception of the studied phenomenon. This provided more extensive and more exact results and this effected positively to the credibility of the results of the study. This is in line with Runeson & al. who state that different types of evidence, figures, statements, and documents are combined to support a strong and relevant conclusion (Runeson & Höst, 2009, p. 16).

2.1.4 The background and roles of researchers

The study is partly compiled by two students of Laurea University of Applied Sciences, who both also work at the Department of Information Technology at Aalto University. The study consists of case study research questions. Two out of three questions were made jointly and the third question was separately done by the writer of this thesis. While the thesis was written, both worked as a part of the version control system acquisition project which is also the subject in this thesis. The introduction, methodology and description of operational environment were compiled together, the rest of the thesis was done separately. As both were part of the VCS project, it was a natural decision of studying the project together and compile the thesis based on the observations made during the project. Since the project was made during

normal working hours, neither of them participated in all the processes of the project, because they also had their normal work assignments. The parts that they couldn't participate in are also mentioned in this study.

The researchers of this study participated in most of the steps of the project. The only step that they didn't participate was the actual VCS installation process. Researchers were also part of the preliminary project group which composed the preliminary data. Though both researchers had solid knowledge in the field of information technology, the project could be seen as their first substantial project they participated actively from the beginning to the end. Besides working for the project, the researchers had to familiarize themselves in the effective project working during the lifecycle of the VCS project. "To be reliable and fulfill project objectives, the investigator must be trained how to document his or her observations and must have a strong understanding of research goals." (O'Grady & O'Grady, 2009, p. 34.)

Anne Sofia Fink divides the research process into seven stages in her article The Role of the Researcher in the Qualitative Research Process. A Potential Barrier to Archiving Qualitative Data. According to her those stages are thematising, designing, interviewing, transcribing, analyzing, verifying and reporting (Fink, 2000, p. 198). The researchers had a role in each of these stages. The stages of the research process are described in Table 3.

Stage	Task for the stage
Thematising stage	Decision about what will be studied, why and how it will be done
Designing stage	The planning of methodologies that will be used in the study. How the interviews are going to be done, how will the data from the interviews be transcribed and analyzed, how the data will be verified and reported.
Interviewing stage	Research data collecting by interviewing the people who use VCS systems
Transcribing stage	Interviews will be transcribed
Analyzing stage	Data collected during the interviews will be analyzed
Verifying stage	The results will be verified against the field notes and interview recordings
Reporting stage	Results of the interviews will be reported

Table 3 Stages of the research process

2.2 Data collection

As Kumar Singh states, behavioral science data gathering is generally done in various ways. (Singh, 2006, p. 218). As this study is a result of several smaller studies, the means of gathering data varied quite a bit from study to study. Eisenhardt debates also on the data gathering during case studies, stating that collecting data during case studies often relies on interviews, questionnaires and observations (Eisenhardt, 1989, p. 534).

The preparations for the preliminary project started in March 2015 and the kick-off day for the actual project was in September 2015. The following studies were conducted during that time period.

1. A preliminary report was composed (n=1)
2. The preliminary report was complemented with the results received from the case study based on user interviews (n=1)
3. Based on the enhanced preliminary report a comprehensive market analysis was done by the administrators of Helsinki Institute for Information Technology (later in the text abbreviated as HIIT)
4. Study about preparations prior to the version control project was conducted and reported.

A test phase was also included in the project. During the test phase, the project group mainly focused on non-functional tests. A more descriptive description of the steps taken before the official project kick-off is presented in Chapter 3.2. All the three conducted studies required and produced vast amount of data. The data used and gathered are listed and described below:

Document	**Content**
The preliminary report (n=1)	The preliminary report was conducted by the manager of the project before the actual project started. The preliminary report included statistics regarding the current usage of version control systems in Aalto University. Report also included preliminary requirements for the future system, information about the service concept and resource requirement estimation. The preliminary report was later supplemented with the customer survey findings.

Meeting transcripts (n=15)	Project meetings took place weekly at the beginning of the project, after that, meetings were arranged when necessary. All the meetings were documented and the documentation was used in study 2 when the preparation process was evaluated. The evaluation focused on the meeting transcripts composed during the preparation stage.
Interviews of administrators and users (n=3)	Interviews were conducted at the beginning of the preparations of the project. Two researchers conducted the interviews. Each of these interviews produced a data document, which were then compiled into a single document.
Product comparison (n=2)	Product comparison consisted of two different documents, the first document was composed by the administrators of HIIT, the document described the current version control system usage at HIIT, requirements for the future systems and suggestions for the future system. The secondary document was composed by two researchers. The second document focused on the product comparison of selected VCS's and examined their features and functions and compiled a document based on the results.
Project plan (n=1)	Project plan was made by the project manager during the preparations of the project. It had a rudimentary plan for the tasks and progression of the project.
Suggestion of investment for the university (n=1)	Suggestion of investment was made by the project manager during the preparations of the project. It suggested that the university should invest in the acquisition of GitLab's community edition. The suggestion had 11 different categories, that explained, for example, the concept and functionality in the environment and benefits of investment.

The risk analysis of the project (n=1)	The risk analysis of the project was conducted by the project group during the preparations of the project. It produced a risk assessment table, which had the name and subscriptions of each possible risk.

Table 4 Documents gathered and used in the studies

2.2.1 Preliminary report

The data for the preliminary report were gathered with methods similar used in focus group research. Focus group research was a suitable method for the study "Focus groups are ideal for exploring people's experiences, opinions, wishes and concerns." (Kitzinger & Barbour, 1999, p. 5.) Data gathering methods resembled ways which data are gathered in focus group researches, but in Aalto's case, instead of having 3 to 5 bigger groups where the user would freely discuss the subject, we had 3 smaller groups which contained the administrators of current systems (n=9).

Administrators discussed freely about the current systems and possible future centrally managed system. A preliminary report was conducted according to these group meetings. The preliminary report also contained requirements received via email from the users of the current VCS's.

2.2.2 Case study

According to Benbasat & al. there are three reason why case study research is a viable information system research strategy. Firstly, the researchers can study information systems in their natural environment and create theories from practice. Secondly, the case study allows researchers to answer "why" and "how" questions and thirdly, a case study is an appropriate method of conduct to research in an area, which has been studied rarely in the past (Benbasat, Goldstein, & Mead, 1987, p. 370).

The case study process adapted Yin's case study research process (Yin, 2009, p. 1). Steps of the process Yin introduced in his study, can be seen in Figure 3. Yin's process starts from planning the study, and proceeding from there onwards until the study results are shared. Yin's process is somewhat agile; some phases of the process can be overlooked, but it is always possible to go back to the disregarded step to carry out that step with newly gained knowledge.

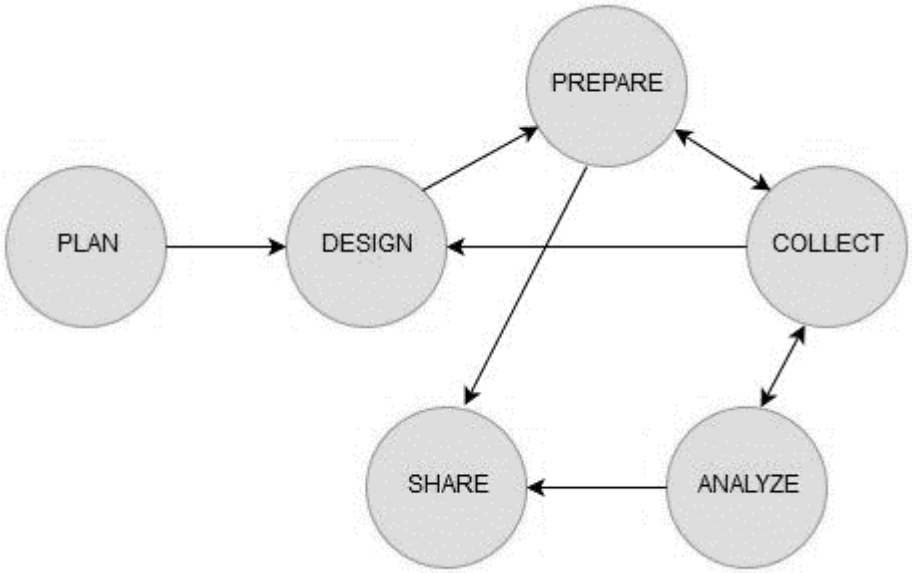

Figure 3: The case Study Research Process (Yin, 2009, p. 1)

Study was performed by interviewing the users and administrators of current systems (n=4). The data gathering in the case study was done in a very similar way described by Kaplan & Duchon in their own system acquire study (Kaplan & Duchon, 1988, p. 577). The results of the interviews were analyzed and then they were presented to the project manager from the university's centralized IT department's project office who oversaw the version control development program at the university.

The interviews were done as semi structured interviews. "Like the structured interview, the semi structured interview tells you how respondents answered the standard questions. In addition, the semi structured interview allows the investigator to ask additional questions to follow up on any interesting or unexpected answers to the standard questions."(Mitchell & Jolley, 2012, p. 277.)

The interview process was divided into four stages:

1. Requests for interviews were sent
2. The interview questions were send in advance to the interviewees
3. Interviews were held
4. Transcription of the interviews.

During the interviews, the interviewer wrote down the answers and recorded each conversation. The researchers transcribed the interviews afterwards. The recorded audio-files were used for this and were deleted after the process.

As a part of the case study, two researchers also familiarized themselves with the available documentation about version control systems. The unit of analysis in this case study was a version control system.

2.2.3 Market analysis

Market analysis was performed by one of the administrators of HIIT (n=1). As the department had a similar ongoing project, centralized IT was able to take advantage of the market analysis they had already compiled. The methodology used in market analysis was evaluative research. "Evaluation research is applied in that the aim is to produce knowledge that will contribute to greater understanding of the effect of a defined activity. This activity may be referred to in a number of ways such as intervention, initiative, and policy." (Given, 2008, p. 303.)

Certain requirement points were discovered before the analysis, and by comparing available products and the features provided by the products, HIIT administrators were able to determine the program which would be the most suitable for their use. The evaluation process was done in a very similar way as Ronald. R. Powell explains in his study which was composed for libraries (Powell, 2006, p. 115). As the project was very similar to the one which was already in progress by centralized IT, HIIT abandoned their own VCS project and started to collaborate with the centralized IT.

2.2.4 Project preparations study

The study about the preparations before the actual project was conducted by two project group members from the IT customer services of the university (n=2). The research question for this study was "what are the preparations prior to version control project".

The data for the study were gathered by using two different methods. The two members participated all the gatherings that the project group had before the project kickoff and took notes. They also carried out several preparation tasks, such as feasibility study, themselves and made documentations about them. During the feasibility study, several different version control systems were chosen for comparison. Comparison was done by placing the chosen systems in a matrix where the requirements for the new system were also placed. By comparing the systems against the requirements, project team was able to narrow down the options and decide the most suitable version control system for this project. This process can be seen in Figure 4.

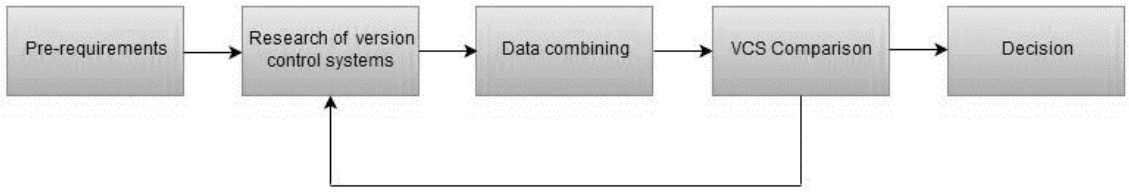

Figure 4 Comparison process

A framework for feasibility study was also used for this study. The framework has two different divisions, concept planning and feasibility analysis. The concept planning was dispensed and study begin from the beginning of feasibility analysis. In our research, product comparison was a part of the feasibility study, after comparison process, the product chosen underwent the evaluation steps similar to the feasibility framework shown in Figure 5.

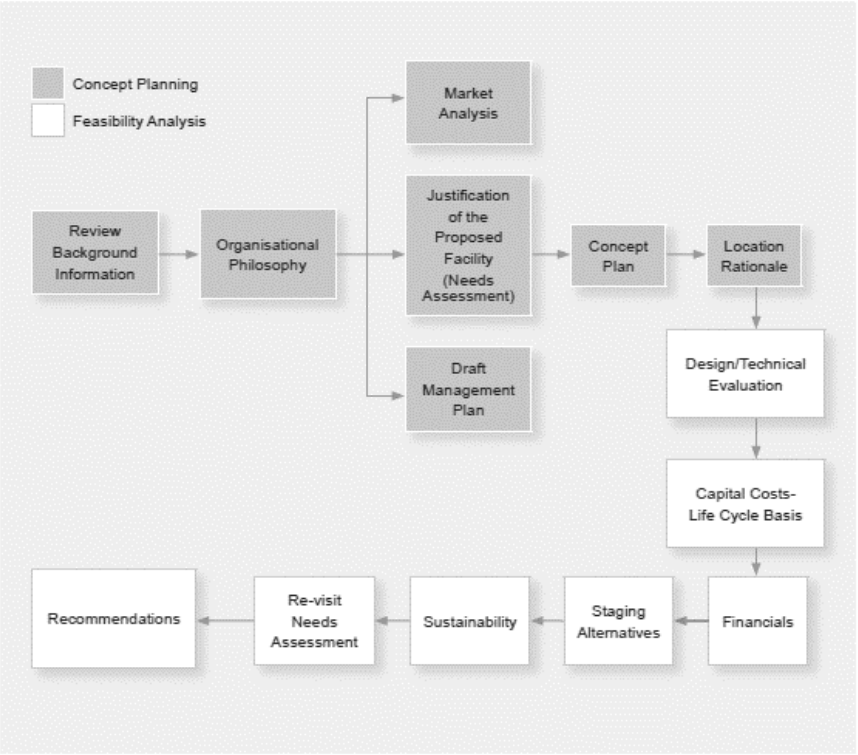

Figure 5: Framework for feasibility study (Didcoe R., July, 2007)

During the feasibility study, the project group received a design & technical evaluation from the information security group of Aalto IT. The project manager evaluated the financial and sustainability possibilities and after that he made an acquisition proposal for the system to Aalto University's SGPP. Project preparations study was made solely to find an answer to the main research question.

2.2.5 IT security requirements

The IT security requirements of the VCS project was examined 4 times during the project. The requirements need to be fulfilled, so that a new system can be integrated into Aalto's IT environment. The questions asked during the process were, what are the IT-security analyzing methods for a new version control system and what is the analyzing process when comparing the security features of a version control system with the university's IT-security requirements. These questions were asked to find an answer to the main question.

First the project group needed an IT security statement for the statement of suggestion. For this statement, the project group formed a small subgroup of 3 people to conduct a small study about GitLab's security features (n=3). Both GitLab Enterprise Edition and Community Edition. The security features were gathered from the GitLab Documentation found at GitLab's homepage. After the study, the results were presented to a person from IT security group (n=1), who then gave the statement.

Second step was the risk analysis of the project. This was put into practice soon after the project kickoff by the whole project group (n=5). It was done as a workshop, where the possible threads were identified, discussed, analyzed and the possible solution were figured out. The project group used a risk assessment table as an evaluation tool. The group was able to name the risks, write a detailed description of each of them, calculate the risk priority, write down recommended actions for each risk, name an owner to a particular risk, update the risk status and when the information about that risk was last updated.

As a third step, the subgroup analyzed the information security of GitLab (n=3). This was done by comparing the features of the system with the IT security framework of Aalto IT. The group used basic level demands of the framework, and it was edited so that all the unnecessary demands were deleted. After this GitLab's features were comparted to the framework's demands. GitLab passed all the necessary requirements. Finally, after the project group had done all its own security tests with the VCS in test environment, the IT security group conducted its own tests. They tested most of the functions and features of the forthcoming systems and tested the security features with a series of random testing. GitLab passed all the tests.

2.2.6 Test phase

Testing during the project was mainly focused on the nonfunctional requirements, but before nonfunctional requirements could be tested, a set of functional tests had to be run to determine if the software is working as desired. As the version control system, which was chosen during the feasibility research, was already well-established, the project group decided to focus the testing on nonfunctional functionalities, which according to Dean Leffingwell, can be divided

into qualities such as usability, reliability, performance, supportability and so on (Leffingwell, 2010, p. 75).

Usability and reliability were tested with a group of user stories developed for the testing purpose. Testing was completed in two different batches, the first batch was completed in the test environment, and after the issues found during the testing had been redeemed, the system was transferred to another server which will act as production environment and the second batch of testing was conducted.

The methodology used during the tests, and the whole of the project, could roughly be compared with the waterfall method of software development. Once a step in the model was completed, results were evaluated and if the evaluation was passed, the project was ready to move to the next step. If a certain step failed, project group started over the current step from the situation it had been developed into, and correct the malfunctions detected during the evaluation. When looking at the steps, the VCS project, and especially the testing phase was very much an iterative waterfall.

These tests were a crucial part of the installation process of the new system. Not only would they give valuable information about possible system flaws, but can also help to develop the system in the future. Jay Nunamaker agrees with this and brings this out in the study, while also adding that in addition to system development, testing and observing a new system can even lead to a discovery of a new theory of the system (Nunamaker Jr, Chen, & Purdin, 1990, p. 100). In Aalto University's VCS project, issues found by the testers were redeemed and in addition, based on the test results, few additional functionalities were added to the system.

2.2.7 Data analysis

As a result of the data collection studies there was a great amount of raw data. The raw data of each study were analyzed and categorized by using the chosen unit of analysis. The analysis had 4 phases.

1. The raw data were taken apart to smaller units of data
2. The smaller units of data were analyzed
3. The units with the same or similar data were categorized
4. The data were consolidated.

The analysis of the results was a necessary part of the study to understand and answer the research question. "The process of data analysis involves making sense out of text and image data. It involves preparing the data for analysis, moving deeper and deeper into understanding

the data, representing the data, and making an interpretation of the larger meaning of the data." (Creswell, 2013, p. 217.) Data analysis can be seen as process, where the data gathered are interpreted in the way they will make sense (Merriam & Tisdell, 2015, p. 175). The data acquired from the survey process had to be processed into a form, where it would be easy to interpret and at same time the integrity of data would not be compromised. To retain the integrity of data, each interview (n=3) conducted during the case study was transcripted from the recordings of interviews into a written document.

By transcripting the interviews, researchers were able to debate on each interview question one at the time, making it also possible to compare the answers of each question between separate interviews on a literal level. The importance of transcripting is pointed out by Sharon Ravitch as she describes transcripting by saying that transcripting is not only important for data collecting, but transcripts are also useful for valid and rigorous data analysis (Ravitch & Carl, 2015, p. 157).

After the interviews were transcribed, the data were compared with each other. The challenge of comparing data was to get all the data into the same form, in this study this challenge was conquered by conducting interviews in a semi-structured model (Folkestad, 2008, p. 10). The results of the common attributes identified by this method can be seen in Chapter 2.2.8.

The data that were analyzed after the second study consisted of meeting transcripts, field notes composed by two researchers, literature and studies published about each task conducted during the preparation period. Furthermore, the observations done during the preparation period were also included in the data. As the data were collected from multiple sources the researchers ended up evaluating the data with evaluative methods. The researchers analyzed the preparation tasks from the viewpoint of time-efficiency and necessity. By doing this, the researchers aimed to optimize the tasks which could be a part of similar projects in future.

After the preparation tasks, the field notes composed by the researchers were organized according to tasks. The field notes were supplemented with meeting transcripts. After combining the data, each researcher familiarized themselves with the literature and studies available from each subject. By comparing methods used in studies and subjective information regarding each task, researchers were able to complete their perspective for optimized tasks.

The data of the study number 3 consisted of the results of the IT security exploration for the suggestion of investment, risk analysis and information security check, with IT security framework. All the data from these studies were gathered and analyzed separately and then combined at the end of the study 3.

The data for the IT security exploration were collected as a part of the product comparison process. Security features, such as authentication methods, known vulnerabilities and the date of the last known system update were examined. The data were collected by using available information, such as official webpages and literature about each examined system. The data collection tools that were used were checklist and data compilation forms (Chaleunvong, 2013, p. 10). After collection, the data were organized and presented to the project manager, who then analyzed the results with me. The analysis was done by comparing the data to the common IT security requirements. After the analysis, he considered if the data were sufficient. In this case, they were and he wrote the IT security part of the suggestion of investment based on the data. This was an iterative process and in case that the data had been insufficient, the process would have been repeated from the beginning.

The risk analysis was conducted as a workshop and all the possible risks were discovered with brainstorming method. The brainstorming had four rules: generate as many ideas as possible, freewheeling is encouraged, criticism and evaluation are not allowed, and combine and improve ideas. These same rules were used in study by Rietzschel al. (Rietzschel, Nijstad, & Stroebe, 2006, p. 246). Every member of the project group participated in the workshop and every single risk that came to someone's mind was written down and after that the risks were compiled and evaluated and ranked. Most improbable and irrelevant risks were left out. Finally, the group evaluated every remaining risk and decided what to do about them. These decisions were written down and checked again during project meetings and updated if needed.

The information security check with IT security framework was also conducted as a workshop, where the project manager, a person from the IT security group and I, compared the features of GitLab with the framework. The used data for this comparison were the same that were collected earlier as a part of the product comparison. All the demands of the framework were checked and commented with the collected data, then the demands and comments were compared with each other and the results were written down and evaluated one by one. This was an iterative process, and in case that the results wouldn't have been sufficient, more data would have been collected and the security check would have been done again. Finally, the results were presented to the rest of the project group.

2.2.8 Attributes identified in study

Though the person, who were interviewed, had a different point of view about the usage of VCS, we were able to discover at least two common attributes that were mentioned by everyone during the interviews. All the phases from the interviews to the identified attributes are described in the table 5.

Phase	Process	Result
Phase 1	Reading of the interview notes, hearing of the interview recordings	Understanding of the interview answers
Phase 2	Fulfilling of the notes with the information from the recordings	Answers have more details
Phase 3	Dividing of the answers according to their themes	Several different themes, waiting for a deeper analysis
Phase 4	Combining and editing of each theme's answer material	Combined and edited answers of every theme
Phase 5	Analysis of the combined answers of each theme	Rationalization of the answers
Phase 6	Possible rollback to phase 1, if the analyzed answers are insufficient	Better understanding of the interview results
Phase 7	Composition of attributes	Common attributes that were discovered from the interviews

Table 5 Phases from the interviews to attributes

The identifying process started by the iteration of the interviews, filling the notes with details that might have been missed during the interview sessions. Then the answers were divided according to their themes and after that combined, edited and analyzed. In case that the analysis would have given insufficient data, the process would have rolled back to phase 1 and continued from there. Finally, the attributes were composed. The recognized attributes were:

Subject	Common attribute identified
Repository management	Normal user should be able to create repositories. User should also be able to delete repositories which are no longer used.
Accessibility	The version control system should be accessible outside university's network. System should be accessible with the most common operating systems & devices

User management	Owner of a repository should be able to manage user rights to the repository he owns. Owner should be able to set the repository as public or private, this should also be possible throughout the whole lifecycle of a repository.

Table 6 Common attributes identified in interviews

3 Stages of the study

In the following chapter, I will describe the data collection process for this study. Data for this study were collected with three smaller studies. Like Dubé & Paré have said, case study research has frequently been criticized on its dependence on a single case study renders and thus it cannot provide generalizable conclusions (Dubé & Paré, 2003, p. 609). Therefore, the strategy of three studies was chosen. This chapter describes what these studies were about and how they would help to answer the research question of this study.

3.1 Study I: Customer demand survey

The first study was conducted in the early stage of the preparations phase of the acquisition process of the version control system and it aimed to examine how could the usage of version control systems be improved in Aalto University. This study was performed by interviewing the administrators of the current version control systems which are hosted by the departments at the university and it was conducted by both researchers. The motive for the survey was to expanse the existing knowledge about the version control systems in use and the usage customs of the users. The results of this study were used to complement the preliminary report done prior to this survey.

3.1.1 The survey process

The study began by contacting the local administrators of the different departments at Otaniemi campus area and asking them, if their department used version control and if yes, could they informally tell what systems the departments use and for what purposes. The administrators were also asked to suggest some active VCS users, who we could interview regarding their usage habits of version control systems. “If one chooses the organizational impact aspect, then you would conduct studies using quantitative surveys or qualitative interviews.” (Hevner & Chatterjee, 2010, p. 119.)

About half of the local administrators answered the inquiry. They gave us very detailed information about the VCS environment they administrated. This included user information, system information and information about the system architecture in their department. Some of them also agreed to be interviewed about their own user habits of version control systems. The interviewed administrators were from the Department of Computer Science, Department of Mathematics and System Analysis and HIIT. According to the received answers, these departments had more than one VCS in use. The following questions were asked from interviewees:

1. Which version control software tools are you using?
2. Are you using more than one version control system?
3. How often do you use version control tools?
4. For what purpose, you use version control?
5. Which functions of the system are necessary for you and which are essential?
6. How could (or should) the system be improved?
7. Is the system you are using, easy to use? Have you encountered problems in system usage, if yes, what kind of problems?
8. Would you be willing to change the version control system you are using to another system?

Since the participants were academic staff from the university, they also analyzed the questions sent to them and the whole interview process and gave us tips for what kind of questions we should ask during the following interviews. One interviewee had even though some questions of his own and demanded that those questions were asked from him. This helped us to understand some aspects of the VCS that we might have otherwise missed.

3.1.2 The findings of the survey

The interviewed used different systems for different purposes. The most common uses were coding and the writing process of a research paper. The Department of Computer Science also used their VCS for their courses and kept all the learning material stored in the system. The learning material was distributed with VCS and students could also return their assignments via VCS. In these cases, the repositories were public. The administrators of the department had modified Git with their own code so that their VCS suites better for their department, students and staff. The department also has other systems that rely on Git: many actions between Git and the other systems are automated. In order to change the VCS, they would have had to make reconfigurations and possibly some hardware changes.

The Department of Computer Science was not going to relinquish their Git, but the administrator told their opinions about the possibility of a new system. They thought that the new system

should be easy to use for the basic users, but it should also have advanced functions for advanced users. They also talked about the systems user interface and told that the best option would be to have a system with a text-based user interface (TUI) and with a graphical user interface (GUI). Their last opinion was that the servers and repositories should be held at the university or at least geographically close for security reasons. Many of the files on the repositories may contain personal or otherwise valuable data. The last thing they suggested was that the university should acquire a repository moving tool that can transfer the data automatically from system to another.

Department of Mathematics and Systems Analysis had a somewhat less VCS users compared with the Computer science department. The local administrator told us that the department has locally administrated Git and SVN. The usage of the SVN is not common and new repositories are created infrequently. A new SVN repository is usually created when there is some international project and an attending professor ask for it. Git is more common and the administrator recommends it to the staff members who have a need for a VCS. The department didn't use version control in its courses. The VCS was used for coding and writing research papers.

When interviewing one of the department's lecturers, the answers were very similar to the answers of the administrators of the Computer science department. He used both systems and a web service called Bitbucket. It is code management service that has private repositories that can be shared with other people. It supports both Git and SVN. The lecturer used version control systems for coding and writing research papers. In his opinion, SVN was better when there is a group coding project. In these projects, the participants are usually from several different countries and a shared repository is necessary. The lecturer used both systems for writing research papers. Some of these papers have more than one author and are easier to share and edit in a shared repository.

When the lecturer was asked about the possibility of a new centralized system, he was ready to change VCS and had a group of suggestions of features that the new system should have. First one was shared repositories, so that the lecturer could read and make comments to students' thesis. Second one was the user control of the user's own repositories, so that the access could be granted to the people outside of the university. Third suggestion was a timeline feature for documents. This would help to see what was edited, when it was done and who did it. The fourth and the last suggestion was the possibility to split the code of a coding project into branches and then merge them back together. With split and merge the coders can use the same source code and code different things simultaneously. The new system should also be easy to use. During the past 30 years that he has used VCS the complexity hasn't been reduced.

HIIT had its' own VCS that they were going to replace soon. One option for the new system was the possible new system of the university. Their systems had own servers, own user accounts and authentication methods. HIIT used Git, SVN and Track, from which the SVN was the most popular. Track was used in project management. Version control was also used for coding and writing research papers. At that moment HIIT had about 200 repositories. One of the HIIT's VCS administrators was interviewed. He talked about the current situation and about the requirements towards their new revision control system. The administrator himself administrated the local servers and the HIIT's SVN and used version control daily. He himself used the systems for storing code, configuration settings and writing documents. The users of the HIIT's version control systems could be divided into two separate groups. Older staff members used SVN and younger staff didn't even know what it was and preferred to use Git. Git was mainly used for coding projects and HIIT's research groups preferred to use it. There was also usage of SVN, mainly for writing papers.

The requirements for the new system were very similar when compared with the other interviewees. The new system should be more manageable. The users should be able to create and manage their own repositories. HIIT has customers from corporate life and they need to share certain data with them and VCS would be the easiest solution for this. At the moment HIIT has guest accounts with very limited access to their services. When asked about the willingness to change the current VCS to another, the administrator was ready to do that. The only requirement was that the new system should have all the same functions as Git and SVN and many advanced features for advanced users.

All the interviewed administrators also agreed on the fact that a new centrally managed system could cut down their departments expenses and could reduce their workload of supporting their systems. This could give them more time for their other work tasks. Based on the results the two most popular systems among the departments and users were Git and SVN. Each of them functions better in different types of use. Git was more suitable for storing code and SVN was better for document writing processes. This is the main reason why most of the departments had parallel systems. Though the person who were interviewed had different points of view about the usage of VCS, we were able to discover at least two common attributes that were mentioned by everyone during the interviews. These attributes were the things that the new system should definitely have or be able to do. These attributes are listed in Chapter 2.2.8.

All the survey findings were reported to the project manager and most of them were added to the preliminary report. Only the most irrelevant data were left out. The results of the customer survey study also answered to special question 1. The discovered common attributes were the things that the new system should have and how the users should be able to use it. The name of the administrators who took part to this survey were promised anonymity and their names

will not be shown in this thesis. More detailed data collection process for this study is described in Chapter 2.2.2. and the results of the interviews are gathered to the table 7.

Interviewee	**Systems used at the department**	**System used for**	**Suggestions**	**Ready to change VCS**
The administrators of the Department of Computer Science	Git	Coding, writing research papers and storing learning material of different courses	GUI and TUI interfaces, servers and repositories at the university or at least geographically close	No
The administrator of The Department of Mathematics and Systems Analysis	Git and SVN	Coding, writing research papers	No	No strong opinion
The administrator of HIIT	Git, SVN and Track	Code storing, configuring settings of other systems, writing of documents	Users should be able to create and manage their own repositories, have same functions as Git and SVN, advanced features.	Yes
Lecturer of the Department of Mathematics and System Analysis	Uses Git, SVN and Bitbucket	Coding, writing research papers	Shared repositories, user control of own repositories, timeline for documents, split and merge of code	Yes

Table 7 The results of the interviews

3.2 Study II: Study about preparations prior the VCS project

The second study was conducted between the beginning of the project preparations and project kickoff, 17.3.2015-28.9.2015. During that time period the project group had 13 meetings. "During this phase, qualitative researchers plan all tasks, responsibilities, and time estimates associated with the project" (Given, 2008, p. 688).

The second study aimed to study the preparation process and seek the answer to special question one, "what are the preparations prior a version control system project". This study was conducted by both researchers. They took part in the preparation process and documented what they had learned during that period. The user interviews are described in Chapter 3.1 and thus not described again. The motive for the study was to produce the documentation of the process, so that the university could use the findings in future acquisition processes or other types of projects, that have the same or similar kinds of tasks.

3.2.1 The preparation tasks

The preparation period consisted of 11 different tasks. Some of these tasks were done one at the time and others at the same time, overlapping, when it was possible. This was done to save time and keep the preparation process as active as possible. The preparations done during the preparation period were (in chronological order):

1. Forming of the project group
2. Definitions
3. Timetables
4. User interviews
5. Product comparison
6. Feasibility study
7. Project Plan
8. Suggestion of investment for the university
9. Risk analysis of the project
10. Communication plan for the project
11. Assignments.

The arranger of the project was the project office of the IT department and the project manager was chosen from the office and he chose the other members for the project group. The rest of the group were chosen from the different groups of the university's centralized IT department. Also, a member of Helsinki Institute for Information Technology, HIIT joined the group, because HIIT was looking for a replacement version control system to their current system.

The criterions for the project members were necessary knowledge from project work, the IT architecture of the university, initialization of services, information security, version control systems and authentication methods. When the project group was formed, it included five people, the project manager and four other. One of them worked as a Linux specialist, two participants worked in the customer service and the last participant was an administrator for HIIT. The project also had visiting experts from the IT department when their expertise was needed.

As soon as the project group was formed, the group familiarized themselves to version control systems with definitions. For this the group had to examine different version control systems, to learn the right technical terms and how the systems work. There was some knowledge among the project group before the gathering of definitions, but this clarification chore ensured that everyone understood the technical details and each other. The group the discussed about their findings and completed each other's knowledge about the VCS. This eliminated the misconceptions among the members. "In working life, learning often takes place in groups and as part of a whole work community's changing capacities." (Pirinen, 2013, p. 14.) The timetables were also planned at the beginning of the preparations. The timetables were not permanent, but adaptable. The project group members chose the preparation tasks they were going to do and gave their estimation of the time that the process would take. The project manager set the timetable so that there was time for unexpected delays. He also had to take possible vacations and sick leaves into account.

The product comparison was made to find out what VSC's would be the most suitable for the university's IT environment. "Technologies are often developed in response to specific task requirements using practical reasoning and experiential knowledge." (Hevner & Chatterjee, 2010, p. 15.) The products that were compared had to fulfill some predefined requirements, such as the overall price (the cost of the product, support and update costs and system expansions costs), support for the system from the developer, the possibility to customize the system and specific system features. Full listing of the requirements can be found in the appendix 1 and the comparison results from Appendix 2.

The examined system features were related to authentication methods, workflow management systems, server management, the number of repositories and simultaneous users and possibility to integrate the product with other systems. The support for the system, including possible updates, was also examined and it was considered a necessary feature. The new system also needed to be customizable to suit the needs of the university. Finally, the systems that fulfilled these requirements were selected for feasibility study. More details about the comparison process can be found in Chapter 2.2.4.

After the product comparison, a feasibility study was conducted. The main goal for this was to find out if any of the selected products would suit the university's needs from the technical, economic and operational point of view. These three categories of the feasibility study by Rodney Overton, listed below, were adapted to the version control system acquisition project (Overton, 2007, p. 6).

1. Technical feasibility told if the suggested VCS could be supported with the existing technology of the IT department.
2. Economic feasibility told if the suggested VCS was cost effective.
3. Operational feasibility told if the VCS would work in the organization if implemented.

The feasibility study of the project group found several suitable VCS systems. In case that the study wouldn't have found one, the project would have had to rethink the requirements for the new VCS system. "A Feasibility Study can assist with making the crucial decision of whether to proceed or not in a wide range of business issues" (Overton, 2007, p. 7.)

During the feasibility process, it became apparent that HIIT was also planning to change their version control system and had already conducted similar studies, such as a market survey. HIIT also had similar requirements for the system and the project utilized their market survey. HIIT had come to a conclusion that GitLab would suit their needs. After a short analysis, the project group came to the same conclusion. GitLab fulfilled the requirements of the feasibility study. More details about the feasibility study, such as the used framework, can be found in Chapter 2.2.4.

After the feasibility study the project manager made a suggestion of investment for acquiring GitLab's community edition. The categories of the suggestion with a short description of each are listed in alphabetical order in the table 8 (Lähteenmäki, 2015c).

Category	**Description**
Benefits	How would the new system benefit the university
Budget of the project	Suggestion of the amount of money that the acquisition would need.
Compatibility with the IT-architecture and information security	How would the new system match with the current IT-architecture and information security
Concept and functionality in the environment	Description of the suggested system and how it could be used at the university

Evaluation of the end results	What kind of results each of the suggested implementation option would have, such as estimation of cost, strengths and weaknesses and estimation of project schedule
Human Resources	How many people would the project require and how many people would be needed to administrate it
Implementation options and suggestion of decision	Are there any other acquisition options (other systems or services) and reasoning why the suggested system should be acquired
Overall cost	How much would the acquisition and upkeep of the system cost
Properties of the information in the system	Form of the data, how can it be accessed and how it is protected
Risks and dependencies	Risks of the acquisition and the project, possible dependencies that could affect the acquisition or usage of the system
Workload	How many working hours would the acquisition project take

Table 8 The categories of the suggestion of investment

The suggestion of investment was first presented to the steering group of operative IT and they were amenable. Next it was presented to the steering group of projects and services (SGPP). They were also amenable and the actual project could begin.

Even if the project had been approved there were still a couple of things that had to be done. These things were risk analysis of the project, communication plan for the project and work assignments. Risk analysis analyzed all the possible threads that the project could encounter. The most threatening risk was that the members of the project group wouldn't have enough time for the project and the project would be delayed or cancelled. More detailed description of risk analysis is described in Chapter 2.2.5.

The communication plan decided what kind of communication methods and tools the project group would use when communicating between each other. The selected communication tools were email, phone calls and IRC. During the work assignments, every member of the project group was given at least one assignment. The project manager will manage the project, the Linux specialist and the administrator for HIIT were in charge of establishing the new system to the IT environment of the university, and the two customer service person were responsible

for the test planning, testing the new VCS system and making instructions for users and training of IT customer service staff when the final version of the system was released.

3.2.2 The findings of the study II

The findings of the preparation study show that there are many different tasks that need to be done before the actual project can begin. Some of the task can be done at the same time and some need to wait that the earlier task is finished. Sometimes the situation changes when some unexpected event happens, like in this case, the market survey from HIIT. These changes can be positive or negative and change the timetables of the project and therefore the project manager needs to create a preparation process timetable that has some extra time. There are also all other kinds of risks, that the preparations might encounter and these risks can be avoided by doing a comprehensive risk analysis.

The short answer to special question 2 "what are the preparations prior a version control system project?" is the list of the preparations mentioned earlier in this chapter, with the addition of the market survey conducted by HIIT, total of 12 different preparation tasks. The survey would have been conducted in any case. Finally, all the preparations and studies done during the preparation phase need to be in order, so that the project manager can present the suggestion of investment and the actual project can begin. The findings of the preparation study have been compiled into table 9.

Table 9 shows the multiple stages of the preparations phase with the results and benefits of each stage. They are listed in chronological order. The table presents the results and advantages of each stage either for the preparation stage, or the project itself, in simplified form. By interpreting the results and benefits of each stage, a conclusion was made, that few of the stages could have been avoided by combining some of the stages.

Market survey and product comparison could have been one entity, where comparison could have been done simultaneously, when the available programs were charted. As both stages addressed practically the same subject, it would be reasonable to complete them at the same time in order to avoid the unnecessary use of time by, for example, documenting both stages one by one when only one documentation would have been enough.

Preparation task	Results	Benefits for the project
Forming of the project group	Workers with time, knowledge and intensity will actualize the project	The project has workers who will actualize it
Definitions	Project workers will know and understand the basics of VCS	The project workers will understand the context of the project
Timetables	The project will have an assessment about how much time each project task will take	Possibility to estimate how long the whole project will take
Surveys / User interviews	Knowledge about the current situation	The project will have data about what the new system should have
Market Survey	List of VCS that fulfill most or all of the pre-requirements	Information about the most common VCS' on the market
Product comparison	Information about each examined VCS's features and functions	Knowledge about the possible "plan b", in case the original VCS isn't suitable after all
Feasibility study	Suitability Information of different examined systems from technical, economical and operational point of view	Most suitable system will be acquired and installed
Project Plan	Rudimentary plan for the tasks and progression of the project	The project will advance in organized manner
Suggestion of investment for the university	Green light for the actualization of the project	The project will be actualized, instead of shutdown
Risk analysis of the project	Risks that threaten the project are recognized	The project can be concluded without any disruptions
Communication plan for the project	Project will use agreed communication tools	The project won't have information disconnections
Assignments	Each member had several task assignments	All the project tasks will be accomplished

Table 9 Compilation of tasks, results and benefits

All of these tasks are not compulsory, but in this case, they were carried out. The essential tasks for similar projects should be chosen based on the experience and the knowledge of the project members. Experienced project group could run the preparations with lesser number of tasks and faster pace.

3.3 Study III: IT security features of the VCS project

The 3rd and last study examined the IT security features of the project. The new VCS needed to fulfill the security requirements, that Aalto IT has for all its programs and services. Also, the IT security of the acquisition process was part of this study. The study took place during the preparation phase and actual project. It aimed to answer the questions in the 3rd special question "What are the IT-security analyzing methods for a new version control system?" and "What is the analyzing process when comparing the security features of a version control system to the university's IT-security requirements?". This study was conducted solely by me. More information about how the security features were examined can be found in Chapter 2.2.5.

The motive for the study was also to produce the documentation of the process, so that the university could use the findings in future projects, just like in the preparation study. I also had a personal interest in information security check with IT security framework. I had studied about the frameworks earlier, but only in theory, and now I had a change to put the theory into practice.

3.3.1 IT security exploration for the suggestion of investment

First, an IT Security study was conducted for the suggestion of investment. Part of the suggestion was an IT security statement from Aalto's IT security group. For this, the security features of the chosen system, GitLab, were examined. Both GitLab Enterprise Edition and Community Edition were studied.

Aalto's IT security has a set of common requirements, that need to be fulfilled, so that a new system or software can be integrated into Aalto's IT environment. The IT security also has an IT security framework, which is used for more detailed IT security analysis, but it wasn't needed at this time. The requirements of IT security were (Salin, 2014a):

1. Aalto's system log rules must be fulfilled
2. The system needs to have an administrator from the university
3. Possibility to do research based on the systems user management (for example user rights and adding and removing credentials)
4. The system should have a test environment.

There are six reasons why the logging of a system is required. These rules are mentioned in Aalto University's document of log rule made by the IT security group (Salin, 2015):

1. Securing of the legality of the operation of Aalto University
2. Following of the load, quality assurance of services and developing of the system
3. Focusing and billing of costs
4. Information security differing observation, control and prevention
5. Clarifying of fault situations and mistakes
6. The legal protection of all parties.

In this case, the number three wasn't relevant since the costs aren't divided between different cost pools or departments at the university. Also, the number six is used only when a system is used by students, staff members or other employees and stores their data.

GitLab fulfilled all the requirements and the member of the IT security gave a statement to the suggestion of investment. The statement emphasized that during the project one had to make sure that the system carries out the acquisition demands of the information security of Aalto University, which are in accordance with data content and in accordance with use cases, and especially from the part of the logs and of the user control of the system (Lähteenmäki, 2015c).

3.3.2 Risk Analysis

The project group also conducted a risk analysis of the project. The analysis contained risks from all the different fields of the project, such as change in the priority of the project, the configuration of the system and lack of possible administrator the system. Some IT security risks were also found, such as Aalto's requirements for the system logs and if the new system would be able to produce them and information leaks from the system, because of human error or bugs in the system. Also, a possibility for VCS' web user interface vulnerability was recognized (Lähteenmäki, 2015b).

A Critical step in the risk management process is to find probable risks, instead of highly improbable scenarios. The risks cannot be assessed before realistic possibilities are identified and described in an understandable way. Other critical step is to describe the risks in detail, instead of just saying "cost" or "schedule", as they are not risks themselves. In order to conduct an effective risk identification, the basic project documentation must be in order. Also, the risk management plan and the organizational environment must be understood. They establish the environment for risk evaluation (Pritchard & PMP, 2014, p. 33-34). The risk analysis was put into practice soon after the project kickoff and it was done with a risk assessment table as an evaluation tool. Example of the table can be seen in Figure 6.

ID	Name	Description	Probability	Impact	Risk Priority	Recommended Actions	Owner	Status	Updated
1	Example	Risk	3	4	12	Fix		Recognised	16.6.2016
2									
3									
4									
5									
6									

Figure 6 Example of a risk assessment table

With the help of the risk assessment table the group could name the risks, write a detailed description of each of them, calculate the risk priority, write down recommended actions for each risk, name an owner to a particular risk, update the risk status and when the information about that risk was last updated. The probability is given a value between 1-3, where 1 is low and 3 is high. The impact is also given a value between 1-6, where 1 is minor and 6 Critical. After this the probability and impact are multiplied together to get the risk priority (Enterprise Risk Management Program, 2016, p. 10-14).

As a result, 23 individual risks were identified, where the highest calculated risk priority was 12 and the smallest was 1. The actions between different risks varied from a quick procedure to no action. All the recognized risks with high priority were fix with a fast schedule, some midlevel risks were recognized and were taken care of after fixing the high priority risks. Most of the low priority risks were small and easy and fast to fix and were fixed when there was spare time. Few risks were so insignificant that they weren't considered as threats and didn't need action. Overall, all the IT security risks were fixed with a fast schedule. The filled risk assessment table used in the project is internal report and classified (Lähteenmäki, 2015b).

3.3.3 Information security check with IT security framework

After the initialization of the project, the information security of GitLab was examined. This was done by comparing the features of the system with the IT Security Framework of Aalto IT. According to the project manager, the security framework*, requirements for security, data protection and IT security for procurements*, that IT uses in the VCS project is a combination of Katakri 2.0, Vahti's instruction 2/2010, 3/2010 and 3/2012 and rule 681/2010 of the council of state of Finland.

The demands in the original master version of the framework consist of the basic requirements and log demands and the demands which are in accordance with the required level and are set on the sectors of safety, privacy protection and information security. The sectors are:

1. Administrative safety (the part A)
2. Staff security (the part P)
3. Physical safety (the part F)
4. Information security (the part I).

There are three different levels of demand in each sector: the minimum requirement, demand of the basic level and demand of the raised level (Salin, 2014b). The frame is edited to every project so that a demand which is only in accordance with the level which is used in the project will be left in and others are removed. A sample of the master version of the used framework (in Finnish) can be seen in Figure 7 (Salin, 2014b). The different sectors are separated from each other with colors.

Osa-alue	Vaatimukset	Lähde/ lisätietoa
Perustietoturvavaatimukset		
Perus 1 Ympäristön suojaus	Toimittajan ympäristö on suojattu tietomurtoja ja verkkohyökkäyksiä vastaan. Toimittajan ympäristöt on suojattu injektiohyökkäyksiltä (SQL, LDAP jne.) sekä muilta tunnetuilta tietoturvauhilta.	ks. http://www.owasp.org/index.php/Top_10_2013-Top_10
Perus 2 Tietoturvaominaisuuksien testaus	Järjestelmän tietoturvaominaisuudet on testattu kattavasti. Toimittaja sitoutuu toimittamaan tietoturvatestauksen testiraportin pyydettäessä.	1) OWASP Top 10 https://www.owasp.org/index.php/Category:OWASP_Top_Ten_Project 2) SANS 20 Critical Security Controls http://www.sans.org/critical-security-controls 3) NIST Technical Guide to Information Security Testing and Assessment http://csrc.nist.gov/publications/nistpubs/800-115/SP800-115.pdf
Perus 3 Käyttäjätunnusten ja salasanojen hallinnointi	Toimittajan Palvelun tuottamiseen käyttämät tietojärjestelmät ja toimintamallit varmistavat paikallisten käyttäjätunnusten ja salasanojen turvallisen hallinnoinnin (so. käyttäjätunnusten ja salasanojen luonti, salasanan toimittaminen unohtuneen tilalle, jne.).	
Perus 4 Salasanavaatimukset	Salasanan on oltava vähintään 10 merkkiä pitkä. Salasanassa on oltava vähintään yksi iso kirjainmerkki, yksi pieni kirjainmerkki ja yksi numero tai erikoismerkki. Salasanassa ei saa olla enempää kuin kaksi samaa merkkiä peräkkäin.	
Perus 5 Salasanojen salaus	Salasanat talletetaan järjestelmään vahvasti salattuna (vähintään AES-256 tai vastaava).	
Perus 6 Salasanojen selvittäminen	Järjestelmä ei sisällä toiminnallisuuksia, joilla salasanat ovat selvitettävissä. Edes järjestelmän pääkäyttäjä ei pysty selvittämään salasanoja (lokaalit tunnukset, läpimenevä autentikaatioliikenne).	
Perus 7 Automatisoitujen sisäänkirjautumisyritysten estäminen	Toimittajan tietojärjestelmät estävät toistuvat automatisoidut sisäänkirjautumisyritykset.	
Perus 8 Kirjautumisen varmenteet	Järjestelmään ja sen kautta kirjautuminen on mahdollista ainoastaan yleisesti hyväksytyltä ja tunnustetulta varmentajalta (CA) hankitulla varmenteella tai muuten erikseen varmistetulla varmenteella ja käyttäen salattua	

Kansilehti | Aalto-Perus | Aalto-lokisaanto | A100 | A200 | A300 | A400 | A500 | A600 | A700 | A800 | A900 | P100 | P200

Figure 7: A sample from the master version of the used framework

The VCS project used the basic level demands. This was decided by analyzing the importance of the system, the importance of the software codes for research and teaching inside the system, and the licenses to use a system to the outsiders and the visibility to the whole Internet. After it was decided that the basic level demands were adequate, the framework was edited, leaving only basic level demands to each category. Finally, each part of the framework was reviewed and commented. If a requirement was fulfilled, it was commentated with a short

note, like “pass” and if a requirement was not fulfilled, it would be commentated with the reason why it didn’t pass and it that requirement is relevant to the version control system and the project. GitLab passed all the necessary requirements.

3.3.4 Testing of the IT security

Finally, after the test system had passed all the tests conducted by the project group, Aalto’s IT security group did its own tests for the environment. The security tested the system in general and conducted a series of random tests, where they randomly selected system features and tested the security features. The progress of the testing process can be seen in Figure 8.

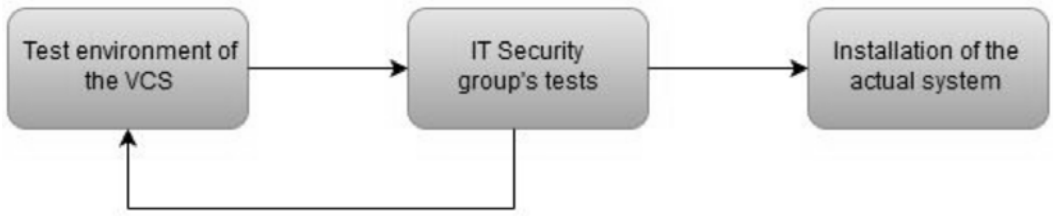

Figure 8: Progress of the IT security group’s testing process

If the system had failed one of more of the tests, the IT security would have reported it to the project group, which would have fix the failed features. After the project group, had done the necessary procedures, the IT security would have retested the system and so on. This would have continued until the system would have passed all the tests of the IT security. When the system passed the tests, the IT security gave the permission to continue onward, and the project group began the installation of the actual system.

3.3.5 The findings of the study III

The results of this study show that there are many kinds of ways to examine IT security prior and during a project. IT security must be taken in note already during the preparation phase of the project and it must be examined several times during the actual project. This must be done so that the project can be actualized, the project doesn’t fail due to a critical thread and that the actual system can be installed into university’s IT environment. The summary of the used IT security methods can be seen in table 10.

Short answers to the questions at the beginning of this chapter: the IT-security analyzing methods for a new version control system are the IT security exploration of the chosen system or systems, risk analysis, information security check with the IT security framework and testing of the IT security.

IT security method	Performed by	When
IT security exploration of the chosen VCS	The two researchers	Project preparations
Risk analysis of the project	Project group	Project preparations
Information security check	Project sub-group	Project
Testing of the IT Security	IT Security group	Project

Table 10 The summary of the different IT security methods

The analyzing process when comparing the security features of a version control system with the university's IT-security requirements were: Choosing a suitable IT security framework, in this case Aalto IT's framework, after that choosing a level of demand from the framework, and after that editing the framework to suit the needs and finally reviewing and commenting each part of the framework and if necessary, making necessary actions based on the results.

3.4 Brief summary of the process

The process of acquiring data for this study can be divided into five separate stages. These stages were the preliminary report, the survey, the project preparations, the acquisition project and the initialization of the new system. It began from the conduction of the preliminary report and ended with the initialization. This process is shown in Figure 9.

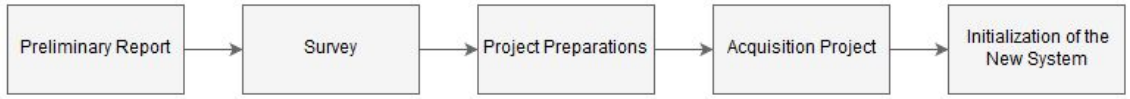

Figure 9: Acquisition of the new VCS

The data for the preliminary report were collected from the administrators of the different departments. The data were then expanded with surveys. Next the preparations for the project were made and after the project got green light from management, the acquisition project took place. The final of the process was the initialization of the system. This whole process took more than 1.5 years to accomplish.

4 Discussion

In this chapter, I will present the final results of the study and discuss them. To be more precise, I will summarize the results of the smaller studies presented in Chapter 3 and describe how those results gave an answer to the research question. I will also discuss the results.

4.1 Results of the study

The results for this study are combined from the data acquired from the smaller studies, that were conducted during the data collection phase. These studies, their questions and expected results are summarized to Table 11.

Study	Question of the study	Expected results
Customer demand Survey	how could the usage of version control systems be improved in Aalto university?	Suggestions of improvement from the interviewees
Preparation tasks	What are the preparations prior a version control system project?	Research data about preparation tasks
IT security features	What are the IT-security analyzing methods for a new version control system? What is the analyzing process when comparing the security features of a version control system to the university's IT-security requirements?	Descriptions and processes of the analyzing methods

Table 11 Summary of the smaller studies

The results of the preliminary report were that there were many version control systems at the different schools and departments at the university. Those systems had users among the staff and students.

Our survey expanded the preliminary report with interviews. The results of those interviews were that, The Department of Computer Science didn't want to change their current system, The Department of Mathematics and Systems Analysis didn't have a strong opinion about the possible change and HIIT was ready to change their VCS, if the new system had all the same features as the current one. The features that were the most wanted from the new system were users' possibility to create and delete repositories themselves, possibility to access the VCS from outside of the university, accessibility with most of the common operating systems and devices, the manageability of user rights of one's own repository and owner's possibility to set the repository as public or private. And according to the interviewed administrators, a new

centrally managed system could cut down departments expenses and would reduce workload of administrator.

We also examined the preparations that were done before the actual project. We participated in the process, studied the processes and wrote down the answers to our question. There were 12 distinct preparation tasks total. Those tasks were described earlier in this study, in Chapter 3.2.1.
Last, I examined the IT security Features of the acquisition process and how the IT security Framework was used in the project. I participated in three out of four of the tasks that were done to ensure the safety of the system and success of the project. These tasks were: IT security exploration for the suggestion of investment, risk analysis, Information security check with IT security Framework and testing of IT security. The usage of the framework was described earlier in this study, in Chapter 3.3.3.

4.2 Limitations and shortages

Even though the results of this study were in line with the hypothesis of the study, there were some things that might have extended them. The sample of the interviewees was small and they were all very experienced users of different kinds of version control systems. The interviews might have needed answers from the basic users with very little or no user experience. This would have brought a different kind of viewpoint to the study and the project. These users use the systems irregularly and because of that they might have difficulties with the basic functions of the system. These results might have helped us to identify more attributes and might have driven the project in a different direction.

The second thing that can be seen as a limitation was the inexperience of the two researchers. Either one of them had any previous experience from working in an acquisition project. They both participated in the preparations and the project and used a multiple resource of the knowledge, such as literature, studies and the help of the rest of the project group, but inexperience may cause errors and these errors might change the outcomes of certain tasks, the direction of the project and even the outcome of the project. As few minor limitations were discovered when reflecting the whole project, the results of the research can still be considered accurate.

4.3 Replication logic

This study can be replicated on the literal-level, however, as the environment, where the study was composed is a closed one, specific technical data about the environment and set up cannot be shared in this study. “Qualitative researchers, in contrast, typically use criterion sampling. This means that participants are selected based on a set of prescribed criteria established by

researchers. Participants possess the variables of interest for study and may or may not represent others in the population from which the sample was drawn. Due to this choice of sampling methodology, qualitative researchers likely will always have a weak case for external validity when focusing solely on one individual study." (Given, 2008, p. 754).

The fact that the interviewees were picked by the project group and local administrators may cause issues with the possible replication attempts as part of the study data solely rely on the opinions of the users of current systems. The replication of the study can be confrontational as it is highly unlikely that the observed configurations, people, social structures or programs are the same (Lee, 1989, p. 40).

4.4 Implications

The results give an important information about the VCS in the university's environment and how it could be comprehended at the beginning of the study and after the acquisition project. In the beginning, the studied version control systems were seen as an important tool, but the systems had their flaws, one system was better when handling code and another when writing research papers. In some cases, this had led to a situation, where single department had multiple systems in use. Also, these systems were inaccessible from outside of the departments' own network. This was the reason the users were open-mindedly ready to try new compensatory options, hoping that they would get easier access, more control to their own repositories and access management to their repositories. Since the new system offers these features, there is a good chance that people will begin to use it.

The department had a differing opinion about changing their current systems to a new one. Department of Computer Science was against it, because their current system was so integrated into the department's environment. If the department's staff begins to use the new system, for example, as a teaching tool, the department's own VCS will have a different role that it was earlier. It will be solely used by the administrator to control the local IT environment. The new system will also reduce the workload of the administrators, as its administrative responsibilities will be handled by the centralized IT unit. The Department of Mathematics and Systems Analysis didn't have anything against changing their VCS and HIIT was willing to change their system. In both of these cases, the usage will move to the new system and their own systems will be eventually shut down. This will reduce the work assignments of the administrators and save both departments' expenses.

The preparations of the project gave an interesting view about how the project is prepared and what kind of task are done and why they are necessary. The preparations were vital condition for the project and all the 12 task that were accomplished were important part of the process. For example, if the market analysis or feasibility study had been left out or done carelessly,

the results might have been different and that would have effected on the acquisition decision. Also, the market survey, that HIIT conducted shows, that unexpected things happen and those can affect the process positively or negatively, luckily in this case the outcome was positive. Some of these 12 tasks were done by the project manager and others by the rest of the project group, but overall, all the tasks can be adapted to a similar project.

The part of the project that focused on the IT security features of the project and the chosen system gave a viewpoint to a process that was earlier unknown. The four-staged process began already during the preparations and continued almost to the end of the project. IT security can be seen as a vital part of the process. Without the first IT security check, there wouldn't have been an IT security statement and the project couldn't have been initiated. Both, the risk analysis and the IT Security framework check were also essential parts of the process. Both analysis' informed the project about possible security risks concerning the project and IT problems with the new system. There is a possibility that without these tasks the project could have failed or at least be delayed. The final tests conducted by the IT security group were obligatory. All the new software is tested thoroughly when integrated into the IT environment. Reason for this is that a single faulty software could disturb the environment and even shutdown systems. These security check steps were necessary to the project and they could and should be used in a similar project.

The comprehension and rationalization of the version control have become clearer at the end of the project. Version control systems are an important tool to administrators, students and other staff members. VCS might become even more important and popular now that the new GitLab is accessible from all the departments and outside of the university. The easier access and the new features could allure the users to use the new system. The rationalization can be divided into two different parts. At the end of the project, the project group informed the publicly about the new VCS and welcomed everyone to try it out and with this information all the departments can rationalize their own systems. They have the option to shut down their own VCS environments and start to use the new GitLab, or they can continue to use their own systems. Either way, they will rationalize their usage of the systems. Giving up of their own systems would give the administrators more time to focus on their other tasks and it could reduce departments costs. Overall, the trend seems to be that the users want more control and more features and this trend will have an influence on the next acquired VCS system, in the distant future.

After the project, the new GitLab is only another version control system among the other systems. Only time will tell if it survives, gains popularity and becomes the dominant system, that will surpass the others. This could lead to VCS shutdown projects, but only time will tell.

4.5 Ethics and reliability of the data

Given describes reliability in her book as followed "Reliability, in the field of research, is broadly described as the dependability, consistency, and/or repeatability of a project's data collection, interpretation, and/or analysis." (Given, 2008, p. 753.) In this study, reliability was ensured with dependability and repeatability.

Dependability of qualitative research investigates if the researcher has made any mistakes during the data collection process, when interpreting the data or when reporting the data. These were avoided by using a dependability audit in where an outside person with knowledge about the subject of the study analyzes the actions of the researchers and gives feedback. In this study, there were two auditors. One was from the university and the other was from outside of the university.

Repeatability in qualitative research investigates what kind of results would the study give if it would be done again and how similar would the results be with the original results. The study can be partially repeated. As the data gathering relied partially on information acquired with semi-structured interviews, results can differ from results presented in this study.

In this study the willingness of the interviewees was guaranteed by asking their permission to record the interview sessions and by promising them that the audio files will be deleted after the transcription. The interviewees were also promised a total anonymity and their identities were hidden even from the management of the project. In qualitative research "Qualitative researchers focus their research on exploring, examining, and describing people and their natural environments. Embedded in qualitative research are the concepts of relationships and power between researchers and participants. The desire to participate in a research study depends upon a participant's willingness to share his or her experience" (Orb, Eisenhauer, & Wynaden, 2001, p. 93). The possibility of the anonymity couraged the interviewed to take part in interviews and to speak more openly about the possible faults of the current systems. For these two reasons, the interviewer should offer the possibility to give anonymous answers.

4.6 Thoughts about the study

There are multiple reasons why this study was important to the university and generally. Since the different schools and departments of the university have been more or less independent about their IT, there hasn't been any greater understanding how many similar version control systems there have been altogether. The results of this study shed light on this situation. Also, the interview answers of the survey process share important information about user demands.

These findings might not only affect VCS, but also other systems at the university. Other projects might also use them as a base for their preliminary inquiries. These answers can also be generalized to common assumptions and used in any project where old system is replaced with a new one.

The results of the preparation study also give valuable information about IT acquisition projects. These results can be used as a guideline in a similar project at the university, in centralized IT or at the schools. Another project manager can study the results and adopt all the useful tasks. This can also be applied to projects outside of the university. In both cases, the results might help project managers with a little experience from project management. From my own experience, the one preparation task that should be complied with high accuracy is the timetables. Small delays are acceptable for a good reason, but in this case, the project manager didn't manage to keep the project on schedule and the project was completed more than six months after the original deadline.

The results from the IT security study brought useful knowledge about the IT policies of the university. These results can be adapted to another project at the university and they tell how to measure the IT security and at what point of the lifecycle of the project each task should be carried out. The one who gains the greatest benefit from the results is a project manager with just a little knowledge about the IT Security aspect of an acquisition project. These results can also be generalized to common guidelines and can be used in any project where the IT security needs to be regarded.

Even though the studies conducted for this study were only from the preparations and IT security, there were other tasks I participated during the project, such as the marketing plan, the training plan and the composition of register description. I also participated in almost all meetings of the project, where the knowledge about the different tasks was shared and the progression of the project was reviewed. There were also project tasks that I didn't participate, such as the actual procurement event of the software from the manufacturer, setup the systems test environment and later the final version, HAKA module installation, marketing and teaching the new system to IT Service Desk. The reason for this is that I had to do my normal work assignments beside the project and fulltime participation to the project wasn't possible. However, the project group was very active and everyone shared all their procedures with high details to the other members and this helped me to gain the gain knowledge about the tasks that I didn't participate and the entirety of the project.

When thinking about the future, the natural extension to the VCS project could be the update of GitLab from the community edition to enterprise edition. If the new system becomes popular among the users, but it gets feedback for lack of functions or features, the enterprise edition

may be the resolution. The other option is to wait for the next generation version control systems and make it the new VCS for the whole university. There is no telling when the next generation's systems are coming out and if they even fit the IT environment, but it can be examined when the time is right. From these two options, the update is more realistic.

4.7 Suggestions for further studies

It is possible to carry out studies based on the research material and results of this study. We examined the situation before the acquisition of the new system, preparations of the project and procedures done during the project. The aim of the project was to require the new system and offer it as a service for the whole university and some of its associates. We didn't participate any technical procedures, such as the installation process or HAKA authentication implementation and this study could be expanded with the knowledge of those processes. This would require interviewing the project members who participated in these parts of the project. This way the university would have a record of these phases if the project members decide to leave the university.

The new system is the GitLab community edition. It has the basic features of GitLab. If it becomes popular around the university, it might need an update to the enterprise edition. The reason for this could be that the features of the community version would be insufficient for some of the users. The researcher could study the current situation a few years from now to see if there is a need for the update and what kind of features the users are missing. The researcher could also examine how much would the update cost and what kind of procedures would the update take and offer the results to the project office and maybe even participate in the new update project, if it is decided that there will be one.

GitLab could also be modified with the modifications of the IT department in the future. Reason for this could be that the current version of the VCS doesn't offer a certain feature that the university needs or that the system needs to be moved to a new IT environment and it is not possible without some system modification. A researcher could conduct a study about how the system can be improved and how it is done and hold a survey and ask the users and administrators what features the system is missing and report the finding to the project office. The researcher could also examine how the modifications are made and added to the system.

This study could also be expanded by using a different point of view to the whole acquisition process. One could examine the progression of the project from the viewpoint of the project manager and ask questions like how to design and manage an IT acquisition project. The results of this study could give even greater understanding to the comprehension of the usage of VCS and the results could be used when planning a new similar project.

The last suggestion is to take the methods and results of this study and apply them to a new project and examine if the same methods in a different environment give similar results. This project could be in a different environment, outside of the university. The researcher could use the same techniques during the data collection phase and examine if the new project requires all the same steps that the VCS project did. Finally, the researcher could compare the gained results with my results and make conclusions and even theory based on those conclusions.

4.8 Final Remarks

This study was conducted to comprehend, rationalize and to understand how to update version control systems in university's environment. University's centralized IT unit arranged a VCS acquisition project, which aimed to provide a new version control system for the whole university. The researcher participated in the project to gather research data about preparations of the project, the actual project and the IT security aspect of the process. Three smaller studies were conducted during the preparations and the actual project. Each of these studies had their own questions. These studies provided data about VCS situation at the university before the project, what kind of preparations an acquisition project needs and how the preparations are made, and how the IT security is taken into account.

The results of these studies tell that in the beginning departments had their own VCS and some of them were ready to change that system to the new coming system and some not, users were ready to change to the new system, if they get more control to their repositories. And the administrators agreed that a new centrally managed system could save their time and departments money. Preparations included twelve different tasks, which all were important to the project and the IT security was measured with four different methods. The results of these smaller studies helped to answer the question of this study. Version control systems are seen as an important tool among administrators, students and other staff members. When updating to a new system, the new systems should give users more control over their own repositories and wider access to the system. The project also helped the centralized IT and the departments to rationalize the usage of the version control systems.

The results of this study were used in a conference paper "Version Control Systems - Development, Comprehension and Rationalization of Usage: Case Aalto University". It will be presented at the 5th International Conference on Management, Marketing, Tourism, Retail, Finance and Computer Applications (MATREFC '17). The abstract of that paper can be found in Appendix 3.

References

Benbasat, I., Goldstein, D. K., & Mead, M. (1987). The case research strategy in studies of information systems. *MIS Quarterly,* , 369-386.

Biňas, M. (2013). Version control system in CS1 course: Practical experience. *IEEE International Conference on Emerging eLearning Technologies and Applications (ICETA), , 11* 23-28.

Chacon, S., & Straub, B. (2014). *Pro git* Apress.

Chaleunvong, K. (2013). Data collection techniques. *Training Course in Reproductive Health Research Vientine.Available on: Http://Www.Gfmer.Ch/Activites_internationales_Fr/Laos/Data_collection_tecniques_Chaleunvong_Laos _2009.Htm Access on October 20th,*

Creswell, J. W. (2013). *Research design: Qualitative, quantitative, and mixed methods approaches* Sage publications.

Davenport, T. H., & Beers, M. C. (1995). Managing information about processes. *Journal of Management Information Systems, 12*(1), 57-80.

De Leeuw, E. D., & Dillman, D. A. (2008). *International handbook of survey methodology* Taylor & Francis.

Didcoe R., S. B. (July, 2007). *Feasibility study guide.* ().Department of Sport and Recreation. Retrieved from http://www.dsr.wa.gov.au/support-and-advice/facility-management/developing-facilities/feasibility-study

Dubé, L., & Paré, G. (2003). Rigor in information systems positivist case research: Current practices, trends, and recommendations. *MIS Quarterly, 27*(4), 597-636.

Eisenhardt, K. M. (1989). Building theories from case study research. *Academy of Management Review, 14*(4), 532-550.

Enterprise Risk Management Program. (2016). The risk management process. Retrieved from www.uvm.edu/~erm/RiskAssessmentGuide.pdf

Fink, A. S. (2000). The role of the researcher in the qualitative research process. A potential barrier to archiving qualitative data. *Forum Qualitative Sozialforschung/Forum: Qualitative Social Research, , 1*(3)

Folkestad, B. (2008). *Analysing interview data possibilities and challenges* (no. 13). *Eurosphere Working Paper Series. Online Working Paper,*

Given, L. M. (2008). *The sage encyclopedia of qualitative research methods* Sage Publications.

Guest, G., Namey, E. E., & Mitchell, M. L. (2012). *Collecting qualitative data: A field manual for applied research* Sage.

Haaranen, L., & Lehtinen, T. (2015). Teaching git on the side: Version control system as a course platform. *Proceedings of the 2015 ACM Conference on Innovation and Technology in Computer Science Education,* 87-92.

Hevner, A. R. (2007). A three cycle view of design science research. *Scandinavian Journal of Information Systems, 19*(2), 87-92.

Hevner, A., & Chatterjee, S. (2010). *Design research in information systems: Theory and practice* Springer Science & Business Media.

Kaplan, B., & Duchon, D. (1988). Combining qualitative and quantitative methods in information systems research: A case study. *MIS Quarterly,* , 571-586.

Kaplan, B., & Maxwell, J. A. (2005). Qualitative research methods for evaluating computer information systems. *Evaluating the Organizational Impact of Healthcare Information Systems,* , 30-55.

Kelleher, J. (2014). Employing git in the classroom. *Computer Applications and Information Systems (WCCAIS), 2014 World Congress On,* 1-4.

Kertész, C. (2015). Using GitHub in the classroom-a collaborative learning experience. *Design and Technology in Electronic Packaging (SIITME), 2015 IEEE 21st International Symposium For,* 381-386.

Kitzinger, J., & Barbour, R. (1999). *Developing focus group research: Politics, theory and practice* Sage.

Lähteenmäki, I. (2015a). *Preliminary report on version control system.* (Internal report). Espoo: Aalto University.

Lähteenmäki, I. (2015b). *Risk assessment table for version control system project.* (Internal report). Espoo: Aalto University.

Lähteenmäki, I. (2015c). *Suggestion of investment for a version control system.* (Internal report). Espoo: Aalto University.

Lee, A. S. (1989). A scientific methodology for MIS case studies. *MIS Quarterly,* , 33-50.

Leffingwell, D. (2010). *Agile software requirements: Lean requirements practices for teams, programs, and the enterprise* Addison-Wesley Professional.

Merriam, S. B., & Tisdell, E. J. (2015). *Qualitative research: A guide to design and implementation* John Wiley & Sons.

Mitchell, M. L., & Jolley, J. M. (2012). *Research design explained* Cengage Learning.

Nunamaker Jr, J. F., Chen, M., & Purdin, T. D. (1990). Systems development in information systems research. *Journal of Management Information Systems, 7*(3), 89-106.

O'Grady, J. V., & O'Grady, K. (2009). *A designer's research manual: Succeed in design by knowing your clients and what they really need* Rockport Publishers.

Orb, A., Eisenhauer, L., & Wynaden, D. (2001). Ethics in qualitative research. *Journal of Nursing Scholarship, 33*(1), 93-96.

Overton, R. (2007). *Feasibility studies made simple* Martin Books.

Pirinen, R. (2013). Towards realization of research and development in a university of applied sciences. *Publications of the University of Eastern Finland, 108*

Powell, R. R. (2006). Evaluation research: An overview. *Library Trends, 55*(1), 102-120.

Pritchard, C. L., & PMP, P. (2014). *Risk management: Concepts and guidance* CRC Press.

Ravitch, S. M., & Carl, N. M. (2015). *Qualitative research: Bridging the conceptual, theoretical, and methodological* SAGE Publications.

Raymond, E. (2007). Understanding version-control systems (draft). Retrieved from http://www.catb.org/esr/writings/version-control/version-control.html

Rietzschel, E. F., Nijstad, B. A., & Stroebe, W. (2006). Productivity is not enough: A comparison of interactive and nominal brainstorming groups on idea generation and selection. *Journal of Experimental Social Psychology, 42*(2), 244-251.

Rochkind, M. J. (1975). The source code control system. *IEEE Transactions on Software Engineering,* (4), 364-370.

Runeson, P., & Höst, M. (2009). Guidelines for conducting and reporting case study research in software engineering. *Empirical Software Engineering, 14*(2), 131-164.

Ruparelia, N. B. (2010). The history of version control. *ACM SIGSOFT Software Engineering Notes, 35*(1), 5-9.

Salin, T. (2014a). *General IT security requirements for acquisition.* (Internal report). Espoo: Aalto University.

Salin, T. (2014b). *Requirements for security, data protection and IT security for procurements.* (Internal report). Espoo: Aalto University.

Salin, T. (2015). *Log rule of aalto university, version 1.2.* (Internal report). Espoo: Aalto University.

Singh, Y. K. (2006). *Fundamental of research methodology and statistics* New Age International.

Sink, E. (2011). *Version control by example* Pyrenean Gold Press.

Srivastava, P., & Hopwood, N. (2009). A practical iterative framework for qualitative data analysis. *International Journal of Qualitative Methods, 8*(1), 76-84.

TutorialConflict. (2013). Tutorial - merging conflicting changes. Retrieved from https://www.mercurial-scm.org/wiki/TutorialConflict

Yin, R. K. (2009). *Case study research : Design and methods* (4th ed ed.). Thousand Oaks (Calif.): Sage Publications.

List of Figures

List of Tables

List of appendixes

Appendix 1: List of requirements

Functional requirements

- Control of user rights for a single repository, which the end-user can manage
- Private repositories and a possibility to use shared repositories
- Git push notifications
- Export / download the repository as a zip-file
- Easy to maintenance and manage
- Controlled lifespan of repositories and credentials (for administration)
- Easy conversion / migration to the new repository keeping the history (a tool for transferring old Git/SVN repositories to the new system)
- Hooks for Push/Pull functions
- No limitations for the number of active projects
- Possibility to change the ownership of projects
- Responsive support to increase project limit

Use and usability

- User friendly
- Support for all device platforms
- Support for most common development environments
- Possibility to use Aalto's credentials, SSO, username/password and SSH keys
- HAKA authentication
- Possibility to collaborate with people outside of the university (username and password possibility for outsiders)
- The students should be able to do group assignments with the new system and single student must be able to define the members of the group
- Access from the Internet
- Possibility to SSH & HTTPS Push / Pull from all the networks
- Product branding. The new VCS must be recognized as university's service

Form of data

- Text based data, mostly programming code, also configuration files, latex files and so on
- Binary files (pictures, PDF, docx) only infrequently

Information security

- The system needs only basic level information security (ST IV, internal). It can therefore be applied to a basic-level's list of security requirements
- See the separate excel file about information security requirements

Backups

- Backups are needed. The VCS system will be backed up in a similar way that file systems or virtual machines are backed up
- The size of the code is gigabytes (possible binary files are big). The size of all code (all the versions of the code) of Aalto should fit into 500 gigabytes

Data protection and privacy

- Only the relevant information for usage is stored (for example username, email address, organization)
- Possibility to separate Aalto's users from other users
- "Hidden mode" – the repository doesn't show in any listings, access only if the person knows direct address to the repository, the repository won't be indexed
- Possibility to change privacy settings of a single repository at any time during its' lifespan
- Rules of HAKA (the requirements that all the services attached to HAKA must fulfill)

Data Content and confidentiality

- In most cases the information isn't public, but not entirely confidential or delicate. In which case basic level information security (ST IV, internal) fulfills the requirements
- In some projects enterprise and business secrets require case-by-case negotiations, if Aalto's usual solution isn't suitable, it might be best to use a separate environment (or environment offered by the company). These have been estimated at a minimum
- Configurations are internal and not necessary to include to the VCS system. Configurations can be kept in a different service, which can only be accessed from the intranet

Availability

- 99% availability is not required, the system isn't critical and can be offline for few moments
- Short outage doesn't cause damage for most of the user base (deadlines vary). GIT users have possibility to use local copy during an outage
- For most people of the user base a short outage doesn't cause
- Monitoring? Possibility to inform the users about outage at webpage
- Users can be informed of a break eg. Web page (cf. Wwwdown.aalto.fi)? At least the normal practices of error information

Continuance

- Continuance will be ensured with maintenance documentation and by ensuring maintenance knowledge with deputies
- Reliability and integrity are important; information must remain unchanged in a safe place.

Appendix 2: Product comparison table

Software	**Repository model**	**LDAP Support**	**SVN Support**	**Supported platforms**	**Cost**
GIT	Distributed	X		Unix, Windows, OSX	Free to use, Prices of host services vary
GitHub Enterprise	Client-Server	X	X	Unix, Windows, OSX	$2500 for team size of 10, $23,750.00 for team of 100
GitLab CE	Client-Server	X	X	Unix, Windows, OSX	Basic: $39/year, Standard: $49/year, Plus: $149/year. Prices are for a single-user licenses
Subversion	Client-Server		X	Unix, Windows, OSX	Free. Prices of host services vary
Mercurial SCM	Distributed	X	X	Unix, Windows, OSX	free software licensed under the terms of the GNU General Public License Version 2 or any later version.

Appendix 3: Abstract and keywords for conference paper “Version Control Systems - development, comprehension and rationalization of usage: Case Aalto University”

Abstract: - IT environments are a continuously changing entirety. Software and systems in these environments go through their lifecycles and then they are replaced with more advanced and economical solutions. This is a normal phenomenon in education and corporate life. The study examines if and how the different systems could be replaced with a single version control system (VCS) and how to prepare and execute an acquisition project of a new IT system. The research method is a multiple case study composed of three case studies. The findings show that users are ready to change their VCS tools to new ones, as long as they get to manage their own repositories. The departments are ready to renounce their own systems as long as the new system has all the same functions as the old systems. The reduction of the overlapping systems will also save money and the resources of the departments.

Keywords: Case study; Version control system; Acquisition; Rationalization