How to Comprehend, Rationalize and Develop Version Control Systems? Case: Aalto University

ANTTI MIETTINEN AND JANI NOUSIAINEN
Aalto University
IT Services
Vaisalantie 8, FI-02130 Espoo
FINLAND
{antti.miettinen; jani.nousiainen} @aalto.fi

JYRI RAJAMÄKI
Laurea University of Applied Sciences
Research, Development and Innovations
Vanha maantie 9, FI-02650 Espoo
FINLAND
Jyri.Rajamaki@laurea.fi http://www.laurea.fi

Abstract: - The topic of the study is topical, as the various universities and non-governmental companies are looking for ways to improve their processes to gain budget savings. This study will debate on the development, comprehension and rationalization of version control system (VCS) usage at Finnish Aalto University. The applied research methodology is a multiple case study analysis. The study has been conducted during a VCS project, which aimed to clarify the usage of VCS at Aalto University by providing a centrally managed system instead of multiple systems which are currently used among the departments of Aalto University. Objective of the study is to examine the possibility to substitute current systems to a one, centrally managed system and clarify the steps important to a VCS acquisition project and suggest improvements to the process of the project. The project itself was conducted at the premises of Aalto University from 2015 to 2016. The most important result 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 on the development of systems instead of just focusing on administrating multiple systems.

Key-words: - Case study; Version control system; Acquisition; Rationalization

1 Introduction

First version control systems (VCS) were developed to help programmers in controlling changes in the source code currently in development [1]. When the VCS locked the file when a user was editing it, the newer systems allow multiple users to access and edit the same document at the same time [2]. The subject of this research is development, comprehension and rationalization of the usage of VCS in Aalto University. The objective of the research is to identify how the reduction of overlapping VCS will affect 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. Time consumed administrating multiple systems can be routed to the development of a 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 descend.

The aim 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 on the development of systems instead of just focusing on administrating multiple systems.

1.2 Operational Environment

Aalto University was established in 2010 when three 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 a network environment that covers the entire university, Aalto-network.

The University 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.

The 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. The IT Department installs 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 the 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 the 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 on what the staff has required from the local administrators. The current situation is that Aalto 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 cur-rent systems need to be replaced with a new and possibly 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 make 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.

2 Research Approach

The subject of the study is "How can the usage of version control systems be developed, comprehended and rationalized in a university's operational environment?" and the study takes place in Aalto University's Otaniemi campus between spring 2015 and fall 2016. The study focuses on the current VCS administrated at different schools of the University and the new VCS that the centralized IT department of the 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 methodological approach to this study is a multiple case study analysis [4], in which a series of case studies were conducted to obtain knowledge in order to answer the research question of this study.

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 three smaller questions asked during the data collection process: 1) How could the usage of version control systems be improved in Aalto University? 2) What are the preparations prior a version control system project? 3) 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?

3 Empirical Cases

This section briefly describes the four empirical studies that belong to this multiple case study analysis.

3.1 Study I: Customer Demand Survey

The first study aimed to examine how the usage of VCS could 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. The motive for the survey was to expand the existing knowledge about VCS in use and the usage customs of the users.

Study I 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 explain 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.

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 Helsinki Institute for Information Technology (HIIT). According to the received answers, these departments had more than one VCS in use. The interviewees were asked the following questions: 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 do 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 tips for what kind of questions should be asked during the following interviews.

3.1.1 The Findings of the Survey

The interviewees 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 better suited 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 one system to another.

The Department of Mathematics and Systems Analysis had somewhat fewer 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 asks for it. Git is more common and the administrators recommend 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 of the 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 number of suggestions of features that the new system should have. The first one was shared repositories, so that the lecturer could read and make comments to students' theses. The second one was the user control of the user's own repositories, so that access could be granted to people outside of the university. The 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 final 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 their 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, of 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 quest 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. Both of them function 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.

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. It aimed to study the preparation process and seek an answer to special question one, "what are the preparations prior to a version control system project". 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.

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, and 11) Assignments.

The arranger of the project was the project office of the IT department. 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. A member of HIIT also joined the group because HIIT was looking for a replacement version control system to their current system. The criteria for the project members were necessary knowledge of 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 others. One of them worked as a Linux specialist, two participants worked in 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 with version control systems with definitions. For this the group had to examine different version control systems in order 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 then discussed their findings and completed each other's knowledge about the VCS. This eliminated misconceptions among the members.

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 in order to find out what VSC's would be the most suitable for the university's IT environment. 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. The examined system features were related to authentication methods, workflow management systems, server management, the number of repositories and simultaneous users and the 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.

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. During the feasibility process, it became apparent that HIIT was also planning to change their VCS 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 the 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.

After the feasibility study the project manager made a suggestion of investment for acquiring GitLab's community edition. The suggestion had the following categories (in alphabetical order): 1) Benefits, 2) Compatibility with the IT architecture and information security, 3) Concept and functionality in the environment, 4) Cost plan, 5) Cost, 6) Evaluation of the final result, 7) Human resources, 8) Implementation options and suggestion of decision, 9) Properties of the information in the system, 10) Risks and dependences, 11) Workload.

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. They were also amenable and the actual project could begin.

Even though the project had been approved there were still a couple of things that had to be done. These were risk analysis of the project, communication plan for the project and work assignments.

Risk analysis analyzed all the possible threats 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.

The communication plan decided what kind of communication methods and tools the project group would use when communicating with each other. The selected communication tools were email, phone calls and IRC.

During the work assignments, each member of the project group was given at least one assignment. The project manager would 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 IT customer service staff when the final version of the system was released.

3.2.1 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 tasks can be done at the same time and some need to wait for an earlier task to be 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 this is why the project manager needs to create a preparation process timetable that has some extra time. There are also 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 to 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 1 that 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, the 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 of the 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.

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. An experienced project group could run the preparations with fewer tasks and a faster pace.

3.3 Study III: IT Security Features

The third study examined the IT security features of the project. The new VCS needed to fulfill the security requirements that Aalto IT has for all of its programs and services. The study took place during

Table 1 Compilation of tasks, results and benefits

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

the preparation phase and actual project. It aimed to answer the questions in the third 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?”

The motive for the study was to produce the documentation of the process so that the university could use the findings in future projects, just like in the preparation study.

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 University’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 this time. The requirements of IT security were: 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

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.

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 of 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. The possible vulnerability of the VCS’ web user interface was also recognized.

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. Another 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. In addition, the risk management plan and the organizational environment must be understood. They establish the environment for risk evaluation [5]. 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.

With the help of the risk assessment table 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.

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 in order to get the risk priority.

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 fixed 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 require action. Overall, all the IT security risks were fixed with a fast schedule. The filled risk assessment table used in the project is classified [3].

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 [6], Vahti's instruction 2/2010, 3/2010 and 3/2012 [7] and Government Decree 681/2010 [8].

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, 2) Staff security, 3) Physical safety and 4) Information security.

There are three different levels of demand in each sector: the minimum requirement, demand of the basic level and demand of the raised level. 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. 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 commentated. 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 if that requirement is relevant to the version control system and the project. GitLab passed all the necessary requirements.

3.3.4 Testing of IT Security

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.

If the system had failed one or more of the tests, the IT security would have reported it to the project group, which would have fixed 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 Findings of the Study III

The results of this study show that there are many different ways to examine IT security prior to and during a project. IT security must be taken into account already during the preparation phase of the project and it must be examined several times during the actual project. This has to be done so that the project can be actualized, the project doesn't fail due to a critical threat and that the actual system can be installed into the university's IT environment.

IT security analyzing methods for a new version of VCS are the IT security exploration of the chosen system or systems, risk analysis, information security checks with the IT security framework and testing of IT security. 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 Study IV: Testing of VCS

The aim of the 4th study is to provide important information about testing during a project to Aalto University's centrally managed IT and an effective way for them to conduct tests during different projects in the future.

This study is based on the testing conducted in Aalto University during a VCS project which was started in 2015. In addition, using the data compiled during the project, also literature and publications about software testing will be compared with the research results. The research questions in the study were following: *1) How should the testing be performed during a VCS project? 2) How can the test results be used to improve the user experience?*

By finding answers to the research questions, will improve the testing conducted by centralized IT and knowledge about how to exploit the test results to supply even better user experience for the end-users of the system.

3.4.1 Project and Testing Adapted to Waterfall Model

At the beginning of the project, a need for a centrally managed version control system was discovered based on the user interviews. The design for the future system was planned after a project group was established, project group consisted of people, who had prior knowledge of the future system and the way the system should work.

After the project group was established, certain areas of responsibilities were distributed among the group, the more technical people were responsible for building the environment and less experienced people focused on marketing plans and planning the training requirements. During the verification phase, a group of pre-planned tests were performed and the test results were carefully documented. The last step of waterfall model [9], maintenance step will encase monthly administrative tasks, such as version updates for the system and possible feature up-dates which has been received from the users of the system.

3.4.2 Testing

What is testing and why is it done? What can be achieved when an already released program is tested by the company that is integrating a new software into its repertory?

The art of software testing [10] defines testing as a way of finding errors in the program while it is used. The definition is quite accurate, but in Aalto's case, one could also describe testing as a method of enhancing the future user experience by implementing certain functions into the software which are not automatically included in it when the software is the first time installed. Such features were related to the login, and administrating the repositories.

3.4.3 Planning the tests

During the planning phase, it is a required custom to create a certain number of test cases which are documented during the planning. In the project concerned, the test plan had various test cases which spanned from testing security to maintenance and from usability to accessibility.

Most of the test cases were planned so that they could be carried on by a user who has never used the system which was to be tested. Before the planning of test cases started, certain required fields for the test had to be determined. The testing was done by following pluralistic methods introduced by Nielsen [11]. The fields in this project are presented in Table 2.

Table 2. Fields used in the testing document

Field	Content of the field
Test area	Which area of system were tested.
Features	Specific area or function which was bee tested.
Test case description	Short description of the test case, what was tested in the scenario.
Setup	What are the pre-requirements before the test could be conducted?
Execution	A more specific, step by step instructions how the test should be conducted.
Expected result	What was expected to happen during the test scenario, how the test was supposed to end.
Ending	Where does the specific test scenario end?
Date performed	When the test was performed, information is vital to determine the version of software which was in production during the test scenario.
Result	The person who conducted the test scenario will fill out observations done during the test scenario.
Comments	Observations about the scenario, possible development proposals to the technical team.
Name of the tester	The person who performs the test will mark his or her name to the field, the name will help the people who are responsible for the program development, to contact the tester if they require some additional information about the results.

3.4.4 Conducting the tests

The test cases were conducted by the project group as no one of the personnel who works for Aalto's central IT, has specialized in testing. The project group consisted of six people, three of them had previous knowledge of VCS systems and the way the system should work.

As mentioned before, each test scenario was carried out by each member of the project group, so the testing group consisted from only six people. The project manager of the VCS project was very confident that the results of the testing will be sufficient though the test group was quite small. At this point, it also wise to mention that in Aalto's VCS project, every other area than security, will be tested manually without any software that would automate the tests. The vulnerability assessments for the software will be scanned with an automated scanner provided by Tenable Network Security, Inc.

Nielsen [12] backs up the small size of the test group and he also states: "The best results come from testing no more than five users and running as many small tests as you can afford." He also explains why increasing the number of testers is only basically waste of resources and there is not that much to gain when you are using more than five testers. As Figure 1 shows, over 80% of the usability problems can be found with only five testers.

As the testers in Aalto University's VCS project were the same people who were responsible for setting up, updating and developing the system, the problems noticed during the tests were solved also during the tests. Nielsen also backs up this method in his article by “after the first study with five participants has found 85% of the usability problems, you will want to fix these problems in a redesign [12].”

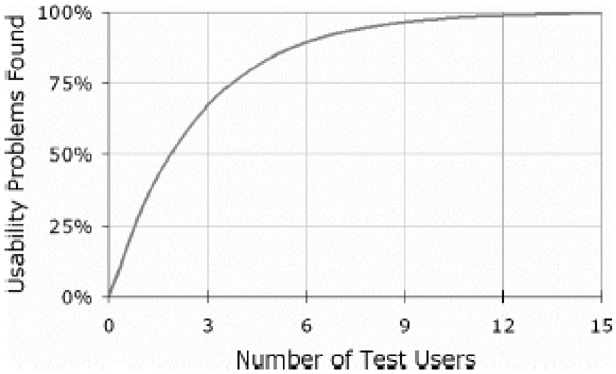

Figure 1. Problems found vs. the number of testers [12]

The testing phase in VCS project could be compared with the iterative waterfall software development model presented in [9]. The issues discovered during the tests could be restored back to the design – stage, and after that the feature could be patched and then implemented back to the system and tested again. In some test cases, the result of the test could even undergo the analysis again to determine, will the feature be implemented into the system or could it be replaced with another similar feature, such feature being one of the login features in the system.

3.4.5 Reporting the test results

”Always use a certain form to gather a certain kind of information [13].” Test reports can be considered as a vital information for the people who develop the systems. The more rigorously the test results are documented, the easier the results can be used to patch the issues found during the tests. The most practical way to report the test results is to store them to the same document where the test cases are created during the planning phase were documented.

In Aalto University’s VCS project, the test scenarios were documented in an Excel-sheet and after a scenario was completed, the tester would document the results as accurately as possible to the same Excel-sheet. After the tests were done, the sheets were gathered and results were analyzed. Based on the analyzation, required action was taken and possible issues were patched.

3.4.6 Answering the research question

As mentioned in this study, testing is an essential part of enhancing the user experience. Though every bug cannot be found while conducting the tests, most of them can be removed during and after the tests. When bugs are removed, the experience for the user is more fluent, a fluent work platform encourages users to use the new system and transfer their data from the old systems to the new, centrally managed VCS system.

When it comes to answering the research questions, one could say, that according to the literature and researches published about testing, there is no correct way to conduct the tests in a VCS project, rather the testing should be a hybrid of many different test methods. One could also mention, the test methods only provide a guideline how the tests could be conducted, but it is always up to the project manager to decide how the tests are conducted in different projects.

The secondary research question was already answered in this study earlier. The better the results are documented; easier they can be used to patch the issues found during the tests. The easier the issues can be introduced to the system, the better the user experience will be.

3.5 Cross-case Conclusions

The results of individual case studies I-IV are summarized in Table 3.

Table 3. Summary of Studies I-III

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
Testing the version control system	How should the testing be performed during a VCS project? How can the test results be used to improve the user experience?	Testing requires no more than 5 persons. Testing should be accurately documented Results can be used to technically enhance the system to meet the customer demands

The results give 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 a single department had multiple systems in use. In addition, 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 departments had differing opinions about changing their current systems to a new one. The 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 before. 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 tasks are done and why they are necessary. The preparations were a vital condition for the project and all of the 12 tasks that were accomplished were an 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 affected 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 of 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 unknown before. 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 of the analyses 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 had been delayed. The final tests conducted by the IT security group were obligatory. All of the new software is tested thoroughly when integrated into the IT environment. The reason for this is that a single faulty software could disturb the environment and even shut down systems. All of 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 usage of version control systems has 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. 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 public about the new VCS and welcomed everyone to try it out and with this information all of 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 just another version control system among 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.

5 Conclusions

The subject of the study is very topical as in future, numerous companies will find their selves in a similar situation where functions must be re-examined in order to find an edge in the field of fierce competition they are operating on. In this scenario, the little things can make the difference in between the survival of the company or ceasing to exist. The current state of economics forces companies to seek savings in their budgets, and in this scenario, overlapping systems are one of the most likely target of saving resources which lead the companies to save money when they do not have to administrate multiple systems at the same time.

By rationalizing the operational environment, both, universities and non-governmental companies enable their workers to focus more on their primary duties as the workers do not have to use several different systems to complete their daily tasks.

On the administrative side, the rationalization of operative environment means at least few different things. Instead of investing the limited work time into administrating multiple systems, work can be done more effectively and the time saved from focusing functions on fewer systems, can be focused on the development of current systems or searching for a substituent for systems that still need to be replaced in order to acquire cost savings.

Based on this study, and the observations made during the project, it is fair to say that if possible, once a project has been completed successfully, the same people would be used to another project. As the people gain experience on project working, they will work faster and more efficiently as they already know what is expected of them and how the goals of the project are achieved. The role of the project manager is vital in an acquisition project. Manager is the one person, who will determine the schedule, and it is up to administrators to meet the schedule the manager has determined. Though the project manager is the one who composes the schedule, it is important that he consults with the administrators before drawing up the schedule for a project, as the administrators may be involved in several projects at the same time, this will allow the manager to compose a realistic schedule and then the schedule will not have to be re-adjusted during the project itself.

Based on the study observations, it is fair to say that when you are planning a project that will affect directly the end-users, if possible, include the view of end-users in the project plan before starting the project itself. In this study, the user interviews compiled before the project started, offered valuable information about the usage of current systems and about the expectations for the future system. Without this information, a great system can be built, but it is a completely different thing if the end-users are willing to use the new system as it does not meet their expectations.

References:

[1] Rochkind, M. J. (1975). The source code control system. *IEEE Transactions on Software Engineer-ing*, (4), 364-370.

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

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

[4] Yin, R. K. (2009). *Case study research: Design and methods* (4th ed ed.). Thousand Oaks (Ca-lif.): Sage Publications.

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

[6] Ministry of Defence, Finland (2011). "KATAKRI, National Security Auditing Criteria. Version II," 2011.

[7] Ministry of Finance, Finland (online). The first version of the structural VAHTI-website. https://www.vahtiohje.fi/web/guest/home

[8] Ministry of Justice, Finland (2010). “Government Decree on information security in central government.”

[9] Trivedi, P., & Sharma, A. (2013). A comparative study between iterative waterfall and incremental software development life cycle model for optimizing the resources using computer simulation. *2nd International Conference on Information Management in the Knowledge Economy (IMKE)*, pp. 188-194.

[10] Myers, G. J., Sandler, C., & Badgett, T. (2011). *The art of software testing* John Wiley & Sons.

[11] Nielsen, J. (1994). Usability inspection methods. *Conference Companion on Human Factors in Computing Systems*, 413-414.

[12] Nielsen, J. (March, 2000). Why you only need to test with 5 users. Retrieved from https://www.nngroup.com/articles/why-you-only-need-to-test-with-5-users/

[13] Hutcheson, M. L. (2003). *Software testing fundamentals: Methods and metrics* John Wiley & Sons.