

SDP - Software Development Plan

September 28, 2018

Contents

1	Project Organization	3
1.1	Project Leaders (PG)	3
1.2	System Architects (SG)	4
1.3	Test Group (TG)	4
1.4	Developers (UG)	5
1.5	Change Control Management(CCM)	5
1.6	External Members	5
1.7	Contact	6
2	Time Plan	6
2.1	Effort and time estimations	7
2.1.1	Phase 1 - Specification	7
2.1.2	Phase 2 - High Level Design and Test Instructions	8
2.1.3	Phase 3 - Low Level Design	8
2.1.4	Phase 4 - Integration and System Testing	8
2.2	Estimation of documents	8
2.3	Time estimation of other activities	8
2.4	Time estimation: Groups	9
2.5	Estimation method and uncertainties	10
2.6	Calendar	10
3	Project Library and Configuration Management	10
3.1	Document Version Number	11
4	Tools and Methods	11
5	Follow-up and Quality Evaluation	11
5.1	Follow-up	11
5.2	Quality Evaluation	12
6	Risk Analysis	12
A	Version Control Management	13

Document history

Version	Date	Resp	Description
0.x	2018-09-14	PG	First draft
0.2	2018-09-24	PG	Second draft
1.0	2018-09-28	PG	Baseline

1 Project Organization

The project requires a total of 16 people. The current grouping is

- PG(2)
 - Assar Pettersson
 - Erik Gralén
- SG(3+1)
 - Jascha Thiel(Leader)
 - Simon Hyttfors
 - Viktor Claesson
 - Daniel Karlsson
- TG (3+1)
 - Jesper Berg(Leader)
 - Jesper Grahm
 - Axel Peterson
 - Tuan Nam Vuong
- UG(3*2)
 - Front-end
 - * Jan Zubac
 - * Filip Karabeleski
 - Back-end
 - * Alexander Pålsson
 - * Sani Mesic
 - Algorithms
 - * Isabella Gagner
 - * Felicia Carlsson

1.1 Project Leaders (PG)

The project leaders are responsible for bringing the desired product to the client. They need to ensure that the other members of the group are doing the re-

quired work by distributing the technical tasks and giving each group member a role(SG, TG or UG). The PG is responsible for creating a proper time schedule in consultation with the other groups and ensure that this schedule is followed throughout the project.

The project leaders are responsible for identifying and creating a list of the necessary configuration units needed for the project, as well as writing/contributing to some of these such as the SDP, SSD and PR. The configuration units need to be accessible at a (online) document library that the PG also is responsible for managing. They are required to have good contact with the client and the other team members by attending and setting up meetings at least once a week.

The PG has to make sure that each document has an informal and formal examination and distribute the examination work among the group members. They are also active in the Change Control Management(CCM).

They have to time report everything they do. The PG is also responsible for completing the final report with metrics and comments from the whole project group.

1.2 System Architects (SG)

The system architect group has the responsibility for the technical aspect in the project. They have a leader who is responsible for reporting in to the PG, and for distributing the work among the members of the SG group. The group leader also has to make sure that the Software Requirement Specification(SRS) is consistent with the Software Verification and Validation Specification(SVVS).

The SG is required to communicate with all the other groups, to ensure that the developer group(UG) and testing group(TG) are working on the same page as well as delegating work tasks to the UG. They are also responsible for configuration management of the work library. The SG are responsible for the compilation of the SRS, STLDD and the SDDD. They are also part of the CCM.

The SG has to time report everything they do,individually. They are required to participate at formal and informal reviews. During a review they may participate as author of the document or as a reviewer of the document. They may only be a reviewer during informal reviews. They also have to write their conclusions and comments to the Project Final Report(PFR) with their individual experiences.

1.3 Test Group (TG)

The test group responsibility is to ensure that everything in the development is properly tested. They have a group leader whose responsibility is to ensure good division of labor, maintain contact with SG and PG, and report all discovered errors to SG.

The group is responsible for creating the SVVS,SVVI and SVVR, as well as to ensure that the SVVS and SRS are consistent. They are also required to participate and help when the UG has to produce automatic unit tests. The TG

has the responsibility of the system construction and to ensure that the system is regression tested.

The TG has to time report everything they do, individually. Participate as author and/or examiner in document examinations(both informal and formal). They also has to compile their conclusions and comments to the end report coherent with what they have done.

1.4 Developers (UG)

The developers are responsible for the development of the functionality to the project. They are divided into three sub-groups: Back-end, front-end and algorithms. Each sub-group is responsible for the functionality coherent to that sub-group. The functionality is required to follow and be consistent to requirements given from the SRS. They are required to produce sub-chapters to the SRS,STLDD and SDDD.

The UG have to time report everything they do,individually. They are required to participate at formal and informal reviews. During a review they may participate as author of the document or as a reviewer of the document. They may only be a reviewer during informal reviews. They also have to write their conclusions and comments to the Project Final Report(PFR) with their individual experiences.

1.5 Change Control Management(CCM)

The Change Control Management consists of the PG and SG. The purpose of this group is configuration management of all the produced documents. The SG has main responsibility for this and the PG participates with making decisions of change operations that requires resource and time-planning.

1.6 External Members

The organization has several external members. The external members are the section manager, the client, a formal examiner and some technical experts. The section manager, responsible to help the project groups if they are in dire need with anything non-technical. The client is the mission-giver to the group and also the recipient of the final product. The formal reviewer is conducting the formal reviews and controls that the development plan is followed.

There are three technical experts that the group can consult if their respective expertise is needed. A requirements expert, a test expert and a design expert.

The external members are:

- Section manager - Martin Höst - Martin.Host@cs.lth.se
- Requirements expert - Johan Linåker - Johan.Linaker@cs.lth.se
- Design expert - Sergio Rico - sergio.rico@cs.lth.se

- Test expert - Martin Höst - Martin.Host@cs.lth.se
- Formal reviewer - Anders Bruce - Anders.Bruce@cs.lth.se

1.7 Contact

Below is a contact list of all group members

- Assar Pettersson - mat13ape@student.lu.se
- Erik Gralén - dat15egr@student.lu.se
- Jascha Thiel - dat14jth@student.lu.se
- Simon Hyttfors - simon.hyttfors.214@student.lu.se
- Viktor Claesson - viktor.claesson.923@student.lu.se
- Daniel Karlsson - daniel.karlsson.536@student.lu.se
- Jesper Berg - jesper.berg.850@student.lu.se
- Jesper Grahm - jesper.grahm.770@student.lu.se
- Axel Peterson - axel.peterson.282@student.lu.se
- Tuan Nam Vuong - tuan.n.vuong@gmail.com
- Jan Zubac - jan.zubac.424@student.lu.se
- Filip Karabeleski - filip.karabeleski.643@student.lu.se
- Alexander Pålsson - alexander.palsson.143@student.lu.se
- Sani Mesic - sani.mesic.348@student.lu.se
- Isabella Gagner - tfy15iga@student.lu.se
- Felicia Carlsson - felicia.carlsson.347@student.lu.se

2 Time Plan

The project is divided into four phases. Each phase has a set of deliverables that has to be sent in within the given deadline. Before the deadline, the deliverables are to be reviewed internally within the group. After the internal review the deliverables shall undergo a formal review by a reviewer chosen by the section. A phase is finished when the reviewer approves the deliverables.

The phases are:

1. *Specification* - During this phase three documents are produced. The first document is the Software Development Plan(SDP) which contains detailed information about the project and a time plan. Also, during this phase the requirements of the product are specified into a document called the Software Requirements Specification(SRS), this is the second document. The final document is the Software Verification and Validation Specification(SVVS), this document contains details about how the testing of the development will be conducted, quality assurance and a validation specification. The deliverables of this phase is the SDP, SRS and SVVS.
2. *High Level Design and Test Instructions* - In this phase the high level architecture of the system will be decided along with test instructions based on the cases from the SVVS. The high level design about the different components and how they communicate will be documented to the Software Top Level Design Document(STLDD) and the test instructions will be documented to the Software Verification and Validation Instruction(SVVI). These two documents are the deliverables of this phase.
3. *Low Level Design* - This phase focuses on the implementation of the high level design. During this phase a document called the Software Detailed Design Document(SDDD) is produced that contains specifications of all system components along with the implemented code. The SDDD is the single deliverable of this phase.
4. *Integration and system testing* - In this phase the system components are integrated to a single system. The system also has to be verified by the function and system tests. All verification and validations during the project are to be documented to the Software Verification and Validation Report (SVVR). If the system quality is acceptable it is to be delivered to the client and the delivery is documented to a System Specification Document(SSD). The project along with the group members individual experiences is documented to the Project Final Report(PFR).

2.1 Effort and time estimations

2.1.1 Phase 1 - Specification

Phase 1 begins at 06/09, and has a deadline at 16/09. This gives 10 days of work spent on this phase. During this phase the SDP, SRS and SVVS are to be produced. These documents are produced by PG, SG and TG respectively.

The most important document of this phase is the SRS and for that reason SG has the most amount of work this phase. The UG group also has to contribute to the SRS but not as much as the SG group.

Since SVVS has to be coherent with SRS, the SRS should be done before some parts of the SVVS is written. The production of the SDP and SRS should start at the same date as the phase begins. The deadline of all these documents is the same as the phase deadline.

2.1.2 Phase 2 - High Level Design and Test Instructions

Phase 2 begins at 17/09 and has its deadline at 27/09. As with phase 1 this gives the group 10 days to complete the STLDD and SVVI.

During this period the PG and UG do not have much required work. For SG the workload is quite heavy during this period, the STLDD is a crucial document for the next phase and contains much information about the high level design. During this phase the TG are responsible for creating the SVVI.

The production of STLDD should start at the same date as the phase begins, as well should the SVVI for good measurements. The deadline of these documents are a couple days before the phase deadline so there is time for an informal review as well as any corrections from this review.

2.1.3 Phase 3 - Low Level Design

This phase begins at 01/10 and has a preliminary deadline at 14/10. During this phase most of the work is done by the UG since they are responsible for the code implementation. SG is responsible for creating the SDDD during this phase.

The production of the SDDD starts when the phase begins. As with phase 2 the document should be done a few days before the deadline to make sure the group has time to do an informal review and any corrections from this review to the SDDD.

2.1.4 Phase 4 - Integration and System Testing

The phase begins at 15/10 and has a deadline at 24/10. It possible that this phase may begin earlier and run parallel with the end of phase 3 but it depends on how well phase 3 is going. In this phase PG is responsible for producing the SSD, as TG is for the SVVR. All groups must also contribute to the PFR.

The production of SVVR may start when the phase begins, as with the SSD. When these documents are done all effort should go to the production of the PFR. The deadline for all documents is the same as the phase deadline.

2.2 Estimation of documents

Table 1 contains a time estimation of each document. The time is in days.

2.3 Time estimation of other activities

During the project there are other activities such as meetings, reviews and configure management. Each week there will be at least one project meeting for the whole group. This meeting will last approximately 30 minutes.

After each phase, a formal review meeting will be held. There are at least four formal meetings and each meeting will last approximately 1 hour. If a baseline is unapproved, the reviewer may require an additional formal meeting. Before each formal review an informal review must be held. The group has

Activity	Time in days
SDP	10
SRS	8
SVVS	10
STLDD	8
SVVI	10
SDDD	14
SVVR	9
SSD	9
PFR	9

Table 1: Time estimation of documents

chosen to do the informal reviews digitally. An informal review may take up to 1 hour or more depending on size and amount of errors on the document being reviewed.

Time will also be spent on configuration management. Configuration management is the process of a change being treated to a document. This is necessary because it enables parallel work between the phases and documents. Configuration management is only done by PG and SG and the amount of time spent on this depends on how well and error-less the development is.

Activity	Time per week in hours
Informal Review	1-2
Formal Review	1 + additional preparations
Configuration Management	1-2
Project Meeting	0.5

Table 2: Time estimation table of other activities

2.4 Time estimation: Groups

In figure 3 is a table of estimated work hours per each group, for each week.

	Week 1	Week 2	Week 3	Week 4	Week5	Week 6	Week 7	Week 8	Total
PG	10	10	7	7	5	5	10	10	64
SG	10	10	10	10	8	8	10	10	76
TG	10	10	10	10	5	8	10	10	73
UG	5	5	5	5	15	15	10	10	70

Table 3: Time estimation for each group.

2.5 Estimation method and uncertainties

As a method of estimating the work time, the amount of credits the course was worth were used in the calculations as well as some amount of guessing.

One of the uncertainties with the time estimations is if a baseline is not approved after the formal review. If the baseline is unapproved, work has to be done to fix this, and the amount of time needed to be spent on this is hard to predict. If much work is needed some deadlines may have to be pushed.

The amount of hours each group has to work each week is also an uncertainty. The numbers may vary depending on how well or bad the development is going.

2.6 Calendar

In table 4 is a preliminary calendar overview of the project.

Phase	Activity	Week 1	Week 2	Week 3	Week 4	Week 5	Week 6	Week 7	Week 8
Phase 1	SDP	PG							
	SRS	SG							
	SVVS	TG							
Phase 2	STLDD			SG					
	SVVI			TG					
Phase 3	SDDD					SG, UG			
	Code					UG			
Phase 4	SVVR							TG	
	SSD							PG	
	PFR							PG,TG & UG	

Table 4: Calendar overview of what each group will work on during each week

In table 5 is a schedule of planned meetings. There may occur more normal meetings if the group desires it and there may occur more formal meetings if the formal reviewer decides it.

	Week 1	Week 2	Week 3	Week 4	Week5	Week 6	Week 7	Week 8	Total
Meetings	1	1	1	1	1	1	1	1	8
Formel Meetings			1		1		1		3

Table 5: Number of planned meetings

3 Project Library and Configuration Management

All documents and code which are version controlled will be stored on the team's Git server. The master branch will only contain documents and code that have been approved by CCM. The project library has its own repository on Github

called documentLibrary. The repository consists of a folder with latex files of the configuration units, a folder for readable pdfs for each baseline and a folder with other documents such as compilations of meetings.

3.1 Document Version Number

All of the documents' baseline will have version number 1.0. After this, all minor changes will increase the version number by 0.1. Minor changes include changes such as corrections and clarifications. All major changes will round the version number up to the nearest whole number. Major changes include changes that will impact the way the group works or the product.

4 Tools and Methods

The following tools and software will be used by team members when working on the project:

- Slack for all communication within the group.
- Overleaf and LaTeX to write documents.
- Github and Git to handle version control of code and documents.
- E-puss to handle time reporting, error reporting and status reporting.

The following tools will be used to develop the product:

- Java, Jetty, and Jersey in back-end implementation.
- SQL and H2 to implement database.
- HTML, CSS, Javascript and Bootstrap in front-end web client implementation.
- Jenkins to automate unit-tests.

5 Follow-up and Quality Evaluation

5.1 Follow-up

As the project progresses, the group may start to deviate from the original plan. This can be either voluntary or involuntary changes. It is important to take note of these changes so that they can be handled and evaluated accordingly. As such, all deviations will be discussed in our weekly meetings and/or reported through other means. While everyone in the group must communicate if they notice or predict deviations, the main responsibility of handling and evaluating them lies with the PG.

Since changes are likely to occur, it is important that the group has a clear guideline to handle them.

1. *Identify* - If any group member notices or predicts problems or changes, this person must inform the rest of the group. This can be done during one of our meetings or through Slack.
2. *Analyze* - PG and key team members analyze potential effects. Additionally, the PG may perform a root cause analysis if it seems appropriate.
3. *Response* - PG and key team members decides the appropriate response and informs the rest of the group.
4. *Evaluate* - PG and key team members evaluate if the response was effective, if this deviation was predictable, and what measures could have been taken to avoid it.

All of these steps should be recorded and discussed internally.

5.2 Quality Evaluation

During the project, quality evaluations will be done through formal reviews, informal reviews, and testing. Formal reviews will be held during each phase with appropriate team members and external members. Informal reviews will be held before each formal review and when otherwise deemed appropriate by PG. PG will either select a few members to review or select the whole group. This group must then individually read and review the documents and send feedback via Slack. Testing will be in the form of both unit tests (UG) and black-box testing (TG).

6 Risk Analysis

Disengagement, *low risk, medium impact* - A member becomes disengaged with the project. This can mean not doing the required work, not showing up to meetings or not communicating well. This can be solved by trying to reengage the member, changing roles or redistributing work.

Internal conflicts, *high risk, low impact* - All conflicts between members will hopefully be handled professionally. If the members can not solve the conflict themselves, other team members can help. If major conflict occur, members should discuss and decided whether to try to solve it internally or externally.

Poor communication, *medium risk, medium impact* - This may be between groups, in the whole group, internally within a group, or with external members. If this problem arises, more meetings can be arranged to hold discussions. During each meeting, the team should discuss the level of communication.

Poor planning, *medium risk, medium impact* - Due to inexperience with the project process, there is a risk of certain aspects being overlooked or estimated poorly. Provided resources should therefore be utilized effectively. PG and others should continuously evaluate the progress and plan.

Lack of technical competence, *low risk, high impact* - Although there will be many new concepts to learn, our team will be able to support each other and take time to learn. In meetings, the team should discuss if there are some things that members should learn beforehand. If required, PG can reassign roles, provide support from other groups, or redesign the plan to be more realistic. The development phase is also expected to include a learning curve.

Poor design, *low risk, high impact* - The high-level design might need to be redesigned and large amounts of work needs to be redone. Since our project is rather small and built upon a preexisting system, the risk is rather small. To further minimize the risk, all designs should be properly evaluated.

Improper reporting, *medium risk, medium impact* - Members might fail to properly report, for example, changes and time. This might lead to miscommunication or inaccuracies. All members should utilize the course material and PG communicate the importance of it.

Unexpected amount of paperwork, *high risk, medium impact* - Members might have an unrealistic expectation regarding the amount of paperwork. This can result in slower progress or not following the project process correctly. The group should discuss this early and prepare.

A Version Control Management

Version control of the documents will be done on github. Github has his a history feature that can be used to look up and restore to older versions of the document.

B Configuration Management List

This is a list of all documents that will be used during the project and with their respective document number.

- Software Development Plan(SDP) - PUSS18001
- Software Requirement Specification(SRS) - PUSS18002
- Software Verification and Validation Specification(SVVS) - PUSS18003
- Software Verification and Validation Instructions(SVVI) - PUSS18004
- Software Top Level Design Document(STLDD) - PUSS18005
- Software Detailed Design Document(SDDD) - PUSS18006
- Software Verification and Validation Report(SVVR) - PUSS18007
- System Specification Document(SSD) - PUSS18008
- Project Final Report(PFR) - PUSS18009