

TEAM 2

Software Development Plan

Version History

Version	Date	Responsible	Description
1.0	2015-09-16	EA, LC	Baseline

Contents

1	References	1
2	Introduction	1
3	Project Goals	1
4	Development Model	1
4.1	Limitations	1
5	Development Organization	1
5.1	Section Manager	1
5.2	Reviewers	2
5.3	Experts	2
5.4	Project Organization	2
6	Customer	3
7	Time Plan	3
8	Follow Up and Quality Evaluation	4
9	Configuration Management	4
10	Risk Analysis	4
10.1	Pre-Developed Components	5
10.2	Loss of Personnel	5
10.3	New Team	7
10.4	Varying Knowledge Levels within Team	7
10.5	Varying Ambition Levels within Team	7
10.6	Unrealistic Time Schedule	7
10.7	Absence of Team Member(s)	8
10.8	Feedback Delay	8
10.9	Change of Demands on System During Development	8
10.10	Lost Work due to Computer Crash	8
10.11	Gold Plating	8
10.12	Error Missed in Test	9
10.13	Wrong User Interface	9

1 References

Ref1 ETSN05 Course Project: IoT and MVD

Ref2 Projekthandledning - Programvaruutveckling för stora system

Ref3 Veckoschema PUSS154251

Ref4 Gantt schema PUSS154252

2 Introduction

A team of nineteen people will during the timespan of almost two months develop an Android application for a customer. The application shall through a Minimum Viable Device, turn on and off a lamp, change its color and read data from a sensor device.

3 Project Goals

The main goal of this project is to develop a user-friendly app to control sensors through a Minimum Viable Device according to Ref1. The app should pass an acceptance test issued by the customer at the end of week 42.

4 Development Model

The project will follow the Waterfall Development Model, as described in Ref2. It will go through the following four phases:

1. Specification
2. High Level Design and Test Instructions
3. Low Level Design
4. Integration and System Testing

4.1 Limitations

The Waterfall Model has been chosen due to the limited time frame set for this project. A limitation in the Waterfall Model is the lack of communication with the customer during the development. The finished product will simply be delivered to the customer without much prior feedback on it and hopefully pass the acceptance test.

Another limitation to this project is its inflexibility, due to the fact that a large part of the system has already been developed and cannot be altered during the development of this project.

5 Development Organization

5.1 Section Manager

The section manager is the project group's top manager and is responsible for helping the group with non-technical problems that arises.

Henrik Cosmo is the section manager for this project.

5.2 Reviewers

The reviewers are there to do the formal and informal reviews for quality control and make sure the development model is being followed.

Alma Orucevic-Alagic is the reviewer on the formal reviews and team members will act as reviewers on informal reviews.

5.3 Experts

There are three experts who can be consulted in the development of the product:

- Technical Expert: Alma Orucevic-Alagic
- Design Expert: Anders Bruce
- Test Expert: Hussan Munir

5.4 Project Organization

The project group is divided into four sub-groups:

- Project Managers
- System Architects
- Developers
- Testers

5.4.1 Project Managers

There are two project managers in the group who have the overall responsibility for the project and the group, see Ref2 for a more detailed description of their main assignments.

Emma Albertz and Linnéa Claesson are project managers.

5.4.2 System Architects

The system architects group consists of three people, one of whom is the system manager. The system architects are responsible for the technical progress of the project and in addition to this, the system manager is also responsible for, together with the test manager, seeing to the consistency of the Software Requirements Specifications (SRS) and Software Verification and Validation Specifications (SVVS). For a more detailed description, see Ref2.

Jacob Mejbvik is system leader and Oscar Axelsson and Daniel Olsson are system architects.

5.4.3 Developers

The developers main assignment is to develop the functionality of the project. There is also a group leader among the developers, to help with communications between the different groups. See Ref2 for a more detailed description.

Developers for this project are:

Carl Rynegardh (group leader)	Marcus Hilliges
Daniel Dornlöv	Niklas Ovnell
Fredrik Månsson	David Cartbo
Filip Månsson	Madeleine Boström

5.4.4 Testers

The testers are responsible for testing of the developed system. One person in the group is test manager and is responsible for dividing the assignments among the group members, reporting to the project managers and also manage the consistency of the Software Requirements Specifications (SRS) and Software Verification and Validation Specifications (SVVS), together with the system manager. For more details, see Ref2.

Oskar Fällström is test leader.

Testers are:

Hanna Autio

Ulf Hörndahl

Moa Eklöf

Måns Andersson

Jonathan Lundholm

6 Customer

The customer has placed the order and is the end recipient of the finished product.

Henrik Cosmo is the customer and the finished product will be delivered to him.

7 Time Plan

See Ref3 for a detailed view of how time will be divided on different assignments. Ref4 displays an overview of when the phases will begin and end, along with the time frames for the corresponding reports of each phase.

The table below shows how many hours should be spent on the different activities of the project, both total hours for the entire project and how much time each person working on the particular activity should spend.

Activity	h total	h/person	Contributors
SDP	16	8	Project manager.
SRS	56	8/4	Primarily system architects/help from developers
SVVS	48	8	Testers
STLDD	72	8/6	Primarily system architects/help from developers
SVVI	60	10	Testers
SDDD	242	25/14	Primarily developers/with help from system architects
SVVR	108	18	Testers
PFR	114	6	Everyone
SSD	12	6	Project managers
IA & TR	-	14	Everyone
Administration ¹	60	30	Project managers
Meetings	21	21	Everyone
Reviews	42	30	Everyone
Unexpected issues	50	-	Everyone

To make time estimations, the effort-method was used together with group members' experiences from projects in the past. The previous experience of the group members varies greatly

¹Administration refers to tasks such as sending emails, signing time reports and prepare meetings.

when it comes to programs and devices used in this project. This makes time estimations difficult since learning time has to be included, but varies significantly from individual to individual.

8 Follow Up and Quality Evaluation

It is the project leaders' responsibility to see to that the time plan is being followed. They will check in with the system leader and architects to make sure that the developers are on schedule with everything and the test leader to make sure that the testers are on schedule.

There will be one big team meeting each week and one meeting with project managers, system leader and architects, test leader and group leader of the developers.

If one (or more) groups have difficulties meeting the deadlines, the project managers and group leaders will have a meeting to evaluate the situation e.g. if they need more help from the other groups, more work outside of scheduled hours or if the deadline has to be pushed forward. Changing the deadlines will be avoided as much as possible.

A formal review will be held at the end of phase 1, 2 and 4 for quality evaluation. Each formal review will be preceded by an internal informal review. Phase 3 will be followed by an informal review.

9 Configuration Management

Identification of configuration items and version naming will follow the standard in Ref2. With an addition of the Weekly Schedule, Gantt Schedule and Test Matrices, which will be given activity numbers 51, 52 and 53 respectively.

All reports and code relating to the project will be available to all team members on a repository on GitHub. A Drive folder will also be used for project meeting protocols, time reports and other information relevant to the team members, such as contact information.

Change management will follow the standard in Ref2.

Documents subject to configuration management are:

- SDP - Software Development Plan
- SRS - Software Requirements Specification
- SVVS - Software Verification and Validation Specification
- STLDD - Software Top Level Design Document
- SVVI - Software Verification and Validation Instructions
- SDDD - Software Detailed Design Document
- SVVR - Software Verification and Validation Report
- SSD - System Specification Document
- PFR - Project Final Report

10 Risk Analysis

Risk analysis is an important part of every new project. Below are some possible risks identified, analysed and actions to take to prevent and/or reduce damage of risk are presented.

10.1 Pre-Developed Components

There are a lot of already developed components that are to be used for this project, each of which can cause problems. There are no alternative components available and the existing ones cannot be altered.

- Risk: High
- Damage Effect: High
- Prevention: Regular meetings with technical expert, to exclude problems with pre-developed components.
- Damage Control: Consult with technical expert.

10.2 Loss of Personnel

There can be loss of personnel on several different levels within the team, with different probabilities and effects. The risk estimation is based on how many people are in the particular group and the level of commitment they have. A big group where each person does not have as much responsibility has been estimated to have a larger probability of losing personnel than a smaller group where each person has a lot of responsibilities towards the team and project. In the same way, the consequences of loss of personnel in a small group where each individual has a lot of responsibilities will be greater than for a large group with fewer responsibilities per individual.

To minimize the problems due to loss of personnel, knowledge and documents will be shared with the entire group. This to avoid loss of information in the event of personnel loss.

10.2.1 Project Leaders (2 persons)

- Risk: Low
- Damage Effect: High
- Prevention: -
- Damage Control: The other project leader will take on more responsibilities and also rely on and delegate more to the other group leaders.

10.2.2 System Leader (1 person)

- Risk: Low
- Damage Effect: High
- Prevention: -
- Damage Control: The system architects will divide the system leaders responsibilities among them.

10.2.3 System Architects (2 persons)

- Risk: Low
- Damage Effect: High
- Prevention: -
- Damage Control: The remaining system architect and system leader will divide the responsibilities among them, but the system architect will take on the larger part of them.

10.2.4 Group Leader, Developers (1 person)

- Risk: Low
- Damage Effect: Medium
- Prevention: -
- Damage Control: The developers will choose a new group leader among themselves.

10.2.5 Developers (7 persons)

- Risk: Medium
- Damage Effect: Low
- Prevention: Create a good working environment among the developers, make sure they have what they need and know what is expected of them.
- Damage Control: The responsibilities of the lost developer will be divided among the other developers by the system architects and group leader of the developers.

10.2.6 Test Leader (1 person)

- Risk: Low
- Damage Effect: High
- Prevention: -
- Damage Control: The testers will choose a new group leader among themselves.

10.2.7 Testers (5 persons)

- Risk: Medium
- Damage Effect: Medium
- Prevention: Create a good working environment among the testers, make sure they have what they need and know what is expected of them.
- Damage Control: The responsibilities of the lost tester will be divided among the other testers by the test leader.

10.3 New Team

The team has not previously worked together, which makes it difficult for project managers to estimate efficiency and skill sets of team members.

- Risk: High
- Damage Effect: Medium
- Prevention: Assigning tasks in collaboration with team mates, according to previous experiences. Continuous communication and guidance within team.
- Damage Control: As a last resort, changing of assignments within the group.

10.4 Varying Knowledge Levels within Team

- Risk: High
- Damage Effect: Medium
- Prevention: -
- Damage Control: The individuals with more knowledge are identified and are encouraged to share their knowledge and help where they can. Forums such as Piazza will also be used to enable the team members to ask each other questions.

10.5 Varying Ambition Levels within Team

- Risk: High
- Damage Effect: Medium
- Prevention: Create a positive working environment for the entire team, to keep them motivated to do a continuously good job. This will be done through team activities outside of the work. Continuous and clear information on what needs to be done and when is also important.
- Damage Control: If a person does not seem motivated, the group leader and/or project managers will sit down with this person and try to work it out.

10.6 Unrealistic Time Schedule

Project leaders have no previous experience with coordinating a project of this magnitude. In addition to this, the team has not worked together before and therefore the knowledge levels and efficiency of the team members are not known. This makes time estimations difficult.

- Risk: High
- Damage Effect: Medium
- Prevention: Good planning from the beginning and consultations with group leaders, clear instructions on what needs to be done and when. Continuous meetings every week throughout the project.
- Damage Control: Extra work on evenings and/or weekends.

10.7 Absence of Team Member(s)

- Risk: High
- Damage Effect: Low
- Prevention: -
- Damage Control: Have the absent team member do the task when he/she gets back or delegate to other present team member if the task is time critical.

10.8 Feedback Delay

- Risk: Medium
- Damage Effect: Medium
- Prevention: Continuous group meetings, both with the entire team and within groups. Efficient communication channels, team members are expected to respond to queries within a day.
- Damage Control: Evaluate what is lacking in communications and manage the situation.

10.9 Change of Demands on System During Development

- Risk: Low
- Damage Effect: Medium
- Prevention: Good planning and communications with customer early on and continuously throughout the project.
- Damage Control: Be clear with the customer with what can and cannot be done within the set budget and time frame of project.

10.10 Lost Work due to Computer Crash

- Risk: Low
- Damage Effect: High
- Prevention: Continuously uploading work done to GitHub server.
- Damage Control: Establish what work has been lost and how to get it back. Worst case scenario it has to be re-done.

10.11 Gold Plating

- Risk: Low
- Damage Effect: Low
- Prevention: Make sure the team understands the product description. If any uncertainties or new ideas occur, consult group leaders or technical expert.
- Damage Control: System architects will evaluate the situation to determine a solution.

10.12 Error Missed in Test

- Risk: Low
- Damage Effect: High
- Prevention: Test matrices mapping tests to requirements, to make sure all requirements are tested.
- Damage Control: Evaluate the error and make necessary adjustments.

10.13 Wrong User Interface

- Risk: Low
- Damage Effect: High
- Prevention: Make sure the team understands the project description. Continuous evaluations of the design and consultations with design expert if any uncertainties occur.
- Damage Control: Evaluate the situation and make necessary adjustments.