Volvo CoPilot Safe Assist
Group 3
Project Plan

November 24, 2017

Contents

1 Introduction **3**

2 Project Organization **3**
2.1 Project Group 3
2.2 Steering Group 3
2.3 Client 3
2.4 Project Group Name 3
2.5 Roles and responsibilities 4
2.6 Organization and communication 4
2.6.1 Group Organization 4
2.6.2 Communication 4
2.7 Worked hours 4
2.8 Activity Plan 5
2.9 Quality assurance 5

3 Description of the system to be developed **6**
3.1 High level description of the domain and the problem 6
3.2 Description of existing systems 6
3.3 Project Description 6
3.4 Requirements 6
3.4.1 Suggestions 7
3.5 High-level description of the desired functionality 8
3.5.1 Actors 9
3.5.2 Use case description 9
3.6 Functional or non-functional requirements 13

4 Initial project backlog **13**

1 Introduction

This project is started as an assignment from our client, Björn Brattberg, in the Software Engineering 2 course at Mälardalens Högskola. The goal of this project is to develop a Safe Assist system, which will help to prevent accidents at construction sites.

Our client acts as a representative of Volvo Construction Equipment. Volvo Construction Equipment is a company that develops, manufactures and markets equipments for construction and related areas of work. It is based in Eskilstuna and its products range from articulated haulers and wheel loaders to automatic gearboxes and road machines. Volvo has recently introduced and included in its vehicles CoPilot, a display which provides real-time intelligence to the operator. It is an Android-based tablet with applications that have the ability to receive information from the actual machine (speed, direction, load). New future software updates are easily implemented in it and the system is 3G-compatible when equipped with a SIM Card.

The Safe Assist system will make use of the location information generated by both operators' hand-held devices and the CoPilot device in the construction machines. When an operator, who is not the driver of a certain vehicle, approaches that vehicle, the system will issue alarms. The operator will be notified of this alarm in his hand-held device and at the same time an alarm will be issued by the CoPilot of the machine.

Adnan Causevic acts as our university contact. His involvement in this project will be technical assistance while running our system in the simulating lab at Mälardalens.

We will follow the agile model of software development. Underlying design documents will be generated.

2 Project Organization

2.1 Project Group

1. Álvaro Sánchez de Lucas
   aas17008@student.mdh.se
2. Dara Ahmadi
   dai15002@student.mdh.se
3. Fernando González Casillas
   fgs17002@student.mdh.se
4. Jonas Ahlin
   jan15021@student.mdh.se
5. Pablo de la Fuente Alonso
   pdo17001@student.mdh.se
6. Rickard Lundberg
   rlg15001@student.mdh.se
7. Sidorela Suli
   ssi17005@student.mdh.se
8. Vasja Çollaku
   vcu17001@student.mdh.se

2.2 Steering Group

Jan Carlson

Juraj Feljan

2.3 Client

Björn Brattberg, Volvo Construction Equipment

Adnan Causevic, Mälardalens Högskola

2.4 Project Group Name

By consensus of all members, the group name was set as "Optimus Octavian".

2.5 Roles and responsibilities

Roles and responsibilities of each member of the project depend on the weekly set tasks. However, general tasks to be accomplished during the overall duration of the project have been assigned to the members, as following:

- Álvaro Sánchez de Lucas - Notetaker
- Dara Ahmadi - Presentation Overseer
- Fernando González Casillas - Configuration Manager (Android Studio)
- Jonas Ahlin - Presentation Overseer
- Pablo de la Fuente Alonso - Configuration Manager (Trello)
- Rickard Lundberg - Configuration Manager (GIT)
- Sidorela Suli - Client Contact, Documentation
- Vasja Çollaku - Group Manager

Availability of the members will be 50% and the estimated period is from 06-11-2017 until 10-01-2018.

2.6 Organization and communication

2.6.1 Group Organization

The project group will meet a minimum of once a week. Additional meetings might be planned according to deadlines and other tasks. Meetings will normally happen at Mälardalens Högskola, unless otherwise specified. Visits to the simulating lab at MDH might be performed as well. Every Monday of each week of the course duration, the group will conduct meeting with the Steering group, in order to report for the tasks accomplished the previous week, their ongoing tasks and the planned activities for the week (according to the Trello board which will be updated weekly). Working hours per member during the previous week will be reported as well.

2.6.2 Communication

Communication in this group shall be performed in English language. By consensus, the main communication platform will be Facebook, where a Facebook group has been created for this project. Whatsapp will serve as an emergency communication portal, for urgent notifications to the members. Trello will be used as a tool to keep an organized display of the working plan. Google Drive shall serve as a repository for every document related to this project (shared between members). A GitHub will serve as a repository for final documents, presentations, and deliverables. The steering group has been given access to the Trello Board and GitHub.

2.7 Worked hours

Worked hours shall be reported through a shared Excel document, available in Google Drive, where each project member shall report worked hours daily.

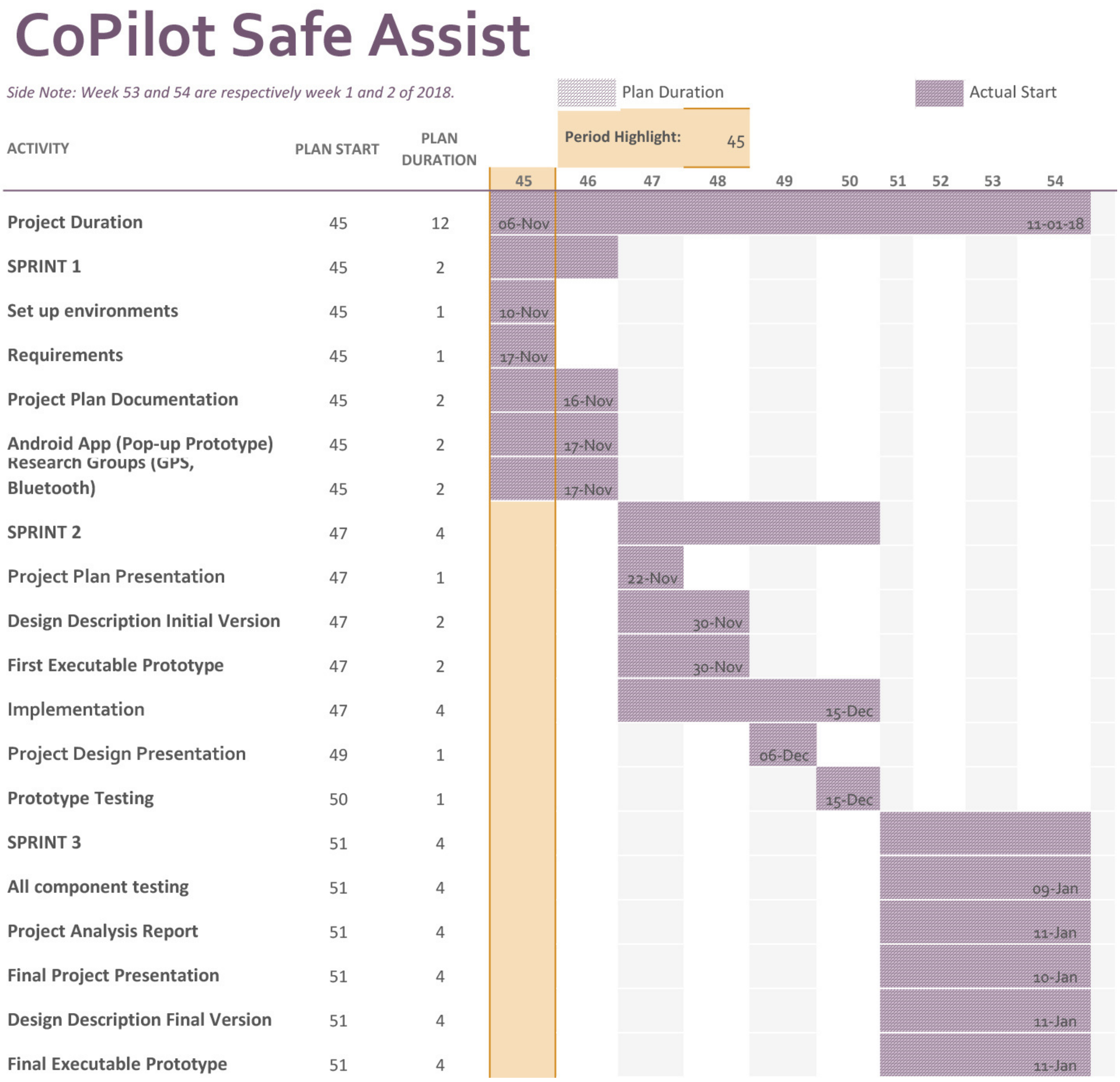
The allocated hours per person, taking holidays etc. into account, are:

Member	W45	W46	W47	W48	W49	W50	W51	W52	W1*	W2*	Total
Álvaro Sánchez de Lucas	12	18	22	23	23	23	23	12	12	12	180
Dara Ahmadi	12	18	22	23	23	23	23	12	12	12	180
Fernando González Casillas	12	18	22	23	23	23	23	12	12	12	180
Jonas Ahlin	12	18	22	23	23	23	23	12	12	12	180
Pablo de la Fuente Alonso	12	18	22	23	23	23	23	12	12	12	180
Rickard Lundberg	12	18	22	23	23	23	23	12	12	12	180
Sidorela Suli	12	18	22	23	23	23	23	12	12	12	180
Vasja Çollaku	12	18	22	23	23	23	23	12	12	12	180

*Note - Week 1 and Week 2 are in year 2018.

2.8 Activity Plan

The Activity plan and time schedule of the project are as following:



2.9 Quality assurance

In order to ensure work progress, quality assurance and to prevent internal and/or external threats, the team members have agreed to conduct the work and their behavior according to the following agreement.

- All required documentation and deliverable will be kept in English and clear, easy to read and interpret.
- Documents written and/or delivered will be reviewed by (at least two) other team members.
- The practice of code reviewing will be undertaken only when a difference in efficiency and complexity is evident and necessary.
- All members will follow the same coding standard in order to avoid differences and unstructured coding.
- Good communication will be kept between the customers, the project manager and the rest of the team members.

- Milestones will be followed up and kept track of.
- The design and the implementations will be inspected to ensure that they can be traced back to a requirement.
- All members will have tasks to conduct individually or in teams during each week. If a member, for different reasons, cannot fulfill his tasks on time, he must notify other members and ask for help before the deadline.
- Members will be flexible to switch tasks and help other members, if the situation requires it.

3 Description of the system to be developed

3.1 High level description of the domain and the problem

Safety Assist is the catchphrase for the system. An application that assists in regards to safety, specifically on work sites and around heavy machinery. Safety has been and still is a major concern at work sites that have heavy machinery, for example construction sites and mining facilities. To a various degree in severity, there's a certain percentage of workers that end up involved in accidents that in one way or another leads to workers being taken off duty. As an example, in 2014 there were a total of 3180103 reported accidents within EU, of which 3739 were fatal [1]. The indirect threat is to the work they were doing, but the personal loss for these workers are of course serious.

3.2 Description of existing systems

Our application will run on hand-held Android devices as well as the Volvo CoPilot system. The latter is a product developed by Volvo which is a tablet computer based on Android. Volvo CoPilot is a real-time system used in Volvo Construction Equipment (Volvo CE) to assist the operator with utilities and essential information. Currently, the CoPilot offers applications like Dig Assist, Load Assist and Pave Assist, to name some. Different applications have different uses with the common goal of assisting the operator in a more efficient and economical as well as safe use of the vehicle.

Volvo CoPilot is used in Volvo CE which span over a broad spectrum of usage. The range of Volvo CE covers forestry, asphalt and excavation work as well as loaders and articulated haulers. As a safety option in Volvo's construction vehicles, they can be equipped with a rear view camera or the Volvo Smart View system. Volvo Smart View provides a real-time, top-down view of the vehicles surrounding, displayed on a screen inside the vehicle.

3.3 Project Description

As a way to try and counter accidents that could've been prevented, Volvo has asked us to develop an application that can work on hand-held devices such as a smart-phone, and also on their own platform, CoPilot. The core functionalities of this application are to give a proximity alert whenever a worker gets within a machine-specific radius of the active machine, which will also be set in tiers depending on just how close to the machine the worker gets. More importantly however, it will give the operator of the said machine a warning that there are unprotected persons close by. There will also be a warning issued to the workers that enter a working site.

3.4 Requirements

ID-1: The Safe Assist should work on hand-held Android mobile devices and the Volvo CoPilot System.

ID-2: The Safe Assist should not issue an alert when the driver approaches his/her own machine.

ID-3: The Safe Assist should issue a proximity alert for both the machine operator and workers on site.

ID-4 The Safe Assist should issue an alert to the worker on entering a work site.

ID-5: Working areas should be defined in the Safe Assist.

ID-6: Increasingly degree of warning as a worker gets closer to a machine.

ID-7: The application should have a lower power consumption when a device is outside a working site than otherwise.

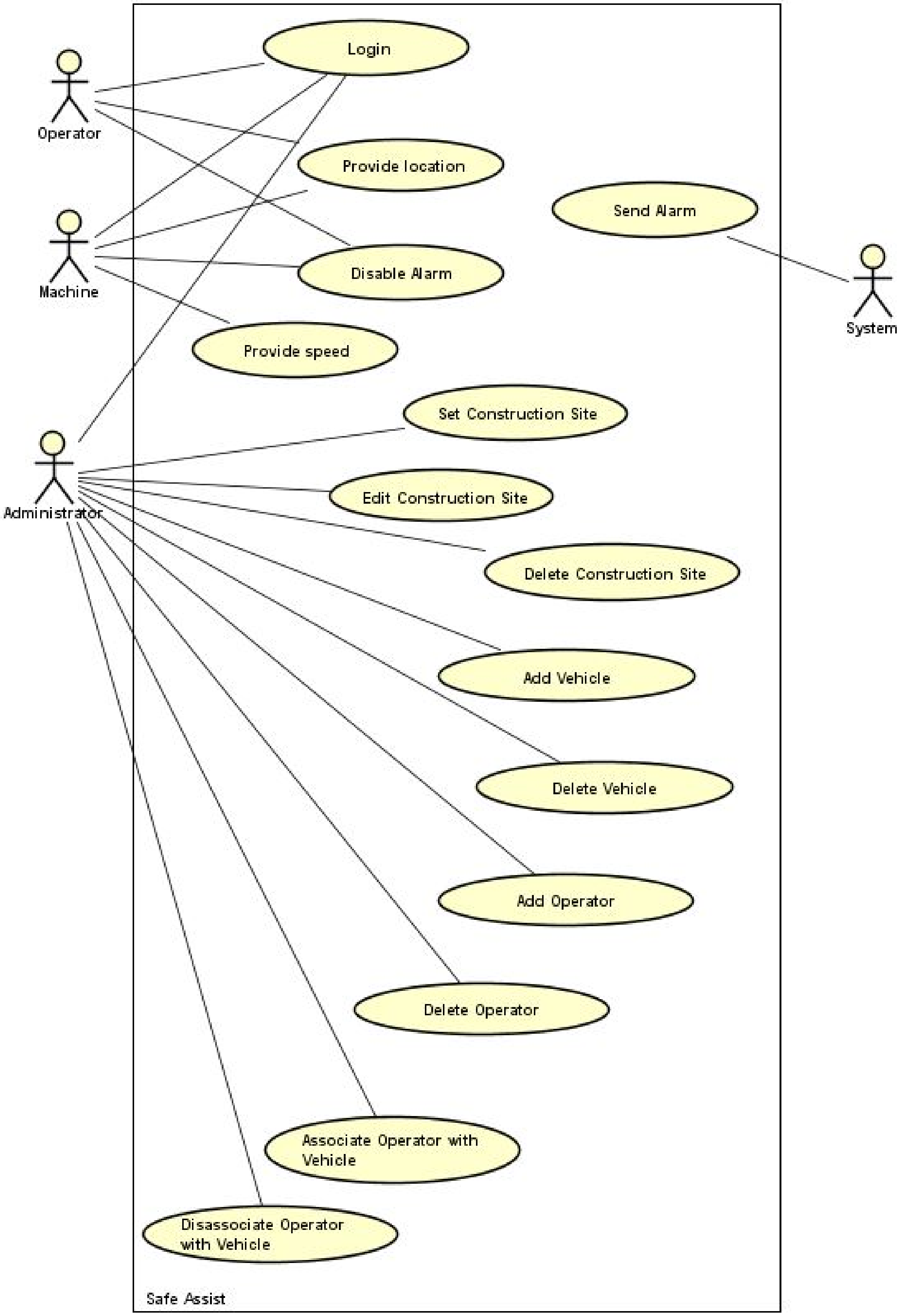
3.4.1 Suggestions

ID-S1: A user should be able to add/modify working areas via the application.

ID-S2: All devices should have unique identifiers in the system.

ID-S3: For the project, the system should work on devices with at least Android version 15.

3.5 High-level description of the desired functionality



3.5.1 Actors

Operator: Worker on the construction site. Can login into the Safe Assist System, provide location to the system and disable ongoing alarm.

Machine: Represented by the CoPilot. Can login, provide the location, disable ongoing alarm and also provide the speed of the construction vehicle.

Administrator: Supervisor of the Safe Assist in the construction site. Sets, edits and deletes the construction site area as well as adding and deleting operator. This actor also both associates and disassociate worker with vehicle.

System: Is the database system. It sends alarm to the Machine which is represented by the CoPilot as well as to the Operator.

3.5.2 Use case description

- Use Case 1
  - **Name:** Login
  - **Initiator:** Operator, Machine, Administrator
  - **Goal:** Enter into the system to use the services offered by the application
  - **Main Scenario:**
    1. User enters the info that is requested by the system.
    2. The system checks if the given info is valid.
    3. System allows the user to get into the system.
  - **Extensions:**
    1. Given info not valid
    a) System throws an ERROR message
    b) Resume at 1
- Use Case 2
  - **Name:** Provide location
  - **Initiator:** Operator and Machine
  - **Goal:** For the CoPilot and workers hand-held devices to provide their location using the GPS sensor.
  - **Main Scenario:**
    1. The worker logs in from the hand-held device or inside the machine with the CoPilot.
    2. The logged in application automatically requests for the location.
    3. The device provides the its GPS-location upon the request.
  - **Extensions:**
    3. The device isn't able to provide the location at a moment.
    a) Wait for the next location update
    b) Resume at 2
- Use Case 3
  - **Name:** Disable Alarm
  - **Initiator:** Operator and Machine
  - **Goal:** For the CoPilot and workers hand-held devices to disable an ongoing alarm on the device.

- **Main Scenario:**
  1. The worker logs in from the hand-held device or inside the machine with the CoPiot.
  2. An alarm request is sent from the server to the device.
  3. An alarm is received by the device and shown to the user.
- **Extensions:**
  3. The alarm failed to receive.
  a) The server sends the alarm request again.
  b) Resume at 3

• Use Case 4

  - **Name:** Provide Speed
  - **Initiator:** Machine
  - **Goal:** For the CoPilot device to receive the speed of the vehicle.
  - **Main Scenario:**
    1. The CoPilot device has logged in.
    2. The device requests the speed of the vehicle.
    3. The speed is received and processed in the application.
  - **Extensions:**
    3. The machine failed to send the current speed.
    a) The device makes a new request
    b) Resume at 3

• Use Case 5

  - **Name:** Send Alarm
  - **Initiator:** System
  - **Goal:** The system detects that two devices are close to each other or not
  - **Main Scenario:**
    1. The system has information about one or two connected devices.
    2. The system checks the for any collisions between the existing or connected devices.
    3. The system sends an alarm to the devices as soon as the check for collisions between the devices are true.
  - **Extensions:**
    1. There are no provided information about the connected devices.
    a) The system waits for devices to connect
    b) Resume at 1

• Use Case 6

  - **Name:** Set Construction Site
  - **Initiator:** Administrator
  - **Goal:** For the administrator to set a new construction site
  - **Main Scenario:**
    1. The administrator has logged in.
    2. The administrator chooses to set a new construction site.
    3. Location and the scale of the site is inserted and set by the admin to the system.
  - **Extensions:**
    3. The information about the site is invalid.
    a) System throws an ERROR message
    b) Resume at 2

• Use Case 7

- **Name:** Edit Construction Site
- **Initiator:** Administrator
- **Goal:** For the administrator to edit an existing construction sites
- **Main Scenario:**
  1. The administrator has logged in.
  2. The administrator chooses to edit an existing construction site.
  3. The new values of the site are inserted by the admin.
- **Extensions:**
  3. The information about the site are invalid.
  a) System throws an ERROR message
  b) Resume at 2

- Use Case 8
  - **Name:** Delete Construction Site
  - **Initiator:** Administrator
  - **Goal:** For the administrator to delete an existing construction sites
  - **Main Scenario:**
    1. The administrator has logged in.
    2. The administrator chooses to delete an existing construction site.
    3. The administrator chooses the site and deletes it.
  - **Extensions:**
    3. The existing site could not be deleted.
    a) System throws an ERROR message
    b) Resume at 2

- Use Case 9
  - **Name:** Add Vehicle
  - **Initiator:** Administrator
  - **Goal:** For the administrator to add a new vehicle
  - **Main Scenario:**
    1. The administrator has logged in.
    2. The administrator chooses to add a new vehicle.
    3. Information about the new vehicle is inserted and set by the admin to the system.
  - **Extensions:**
    3. The information about the vehicle is invalid.
    a) System throws an ERROR message
    b) Resume at 2

- Use Case 10
  - **Name:** Delete vehicle
  - **Initiator:** Administrator
  - **Goal:** For the administrator to delete an existing vehicle
  - **Main Scenario:**
    1. The administrator has logged in.
    2. The administrator chooses to delete an existing vehicle.
    3. The administrator chooses the vehicle and deletes it.
  - **Extensions:**
    3. The existing vehicle could not be deleted.
    a) System throws an ERROR message
    b) Resume at 2

- Use Case 11
  - **Name:** Add Operator
  - **Initiator:** Administrator
  - **Goal:** For the administrator to add a new operator into the system
  - **Main Scenario:**
    1. The administrator has logged in.
    2. The administrator chooses to add a new operator.
    3. Information about the new operator is inserted and set by the admin to the system.
  - **Extensions:**
    3. The information about the operator is invalid.
    a) System throws an ERROR message
    b) Resume at 2
- Use Case 12
  - **Name:** Delete operator
  - **Initiator:** Administrator
  - **Goal:** For the administrator to delete an existing operator
  - **Main Scenario:**
    1. The administrator has logged in.
    2. The administrator chooses to delete an existing operator.
    3. The administrator chooses the operator and deletes it.
  - **Extensions:**
    3. The existing operator could not be deleted.
    a) System throws an ERROR message
    b) Resume at 2
- Use Case 13
  - **Name:** Associate Operator with Vehicle
  - **Initiator:** Administrator
  - **Goal:** For the administrator to associate an operator with a vehicle
  - **Main Scenario:**
    1. The administrator has logged in.
    2. The administrator chooses to associate operator with vehicle.
    3. The administrator chooses existing operator and vehicle and associates them.
  - **Extensions:**
    3. The association between the operator and vehicle failed.
    a) System throws an ERROR message
    b) Resume at 2
- Use Case 14
  - **Name:** Disassociate Operator with Vehicle
  - **Initiator:** Administrator
  - **Goal:** For the administrator to disassociate an operator with a vehicle
  - **Main Scenario:**
    1. The administrator has logged in.
    2. The administrator chooses to disassociate operator with vehicle.
    3. The administrator chooses existing associated operator and vehicle and disassociate them.
  - **Extensions:**
    3. The disassociation between the operator and vehicle failed.
    a) System throws an ERROR message
    b) Resume at 2

3.6 Functional or non-functional requirements

For this project we have made the decision to limit ourselves to Android version 15. This is mainly due to time constraints but also because Volvos' CoPilot runs on Android. The client also expressed a wish for the application to be restrictive with energy consumption and we will of course work with this in mind but at this time we can not predict in what manner. An additional functionality of issuing a warning when entering a working site has been suggested to us and efforts will be put to implement it. On the other hand, with the above description, we have already established that we will need a way to make unique identifiers so that an operator does not get redundant warnings when walking to, from or around his own machine. This, in turn, led us to the fact that an operator should be able to register his hand-held device to the CoPilot so that the operator does not get continuous warnings on the hand-held device while working the machine or whenever someone gets too close, those warnings should be on the CoPilot alone. Another proposed idea is that the radius of a machine needs to be set depending on the type of machine and in which speed it is moving.

4 Initial project backlog

To do this project we were given a few requirements and a specific goal. Due to this, we are required to research about the possible implementations and find the most efficient one. Therefore, we are working as an agile development team, which limits our knowledge of the implementable user features, only knowing the short-term functionalities.

This is why, currently, we are only able to define the effort and required time of the immediate deliverables, which are quite general development tasks, rather than the previously said specific user features.

Necessary effort will be defined in a range of 0-10, being 0 very little effort and 10 very hard effort. However, importance will be defined in a range of 0-100, being 0 not important and 100 very important.

The initial list of system features to be implemented, the effort and the importance for each feature are described in the following table.

		Effort	Importance
Product plan	Requirements	8	90
	Project plan documentation	7	70
	Set up environment	4	75
	Research groups	4	90
Design description	Design description draft	4	60
	Final design description	7	90
Product	Android pop-up test app	6	65
	First executable prototype	7	70
	Prototype testing	8	75
	Final executable prototype	10	90
	Final testing	10	95
Project report	Project analysis report	8	70
	Final project presentation	10	85

References

[1] Eurostat. (2014) Number of non-fatal and fatal accidents at work, 2014 (persons) kernel description. [Online]. Available: http://ec.europa.eu/eurostat/statistics-explained/index.php/File:Number_of_non-fatal_and_fatal_accidents_at_work,_2014_(persons)_YB16.png