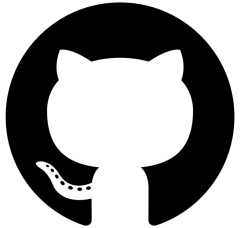
Project Plan Presentation

Tommy Ernsund
Viking Forsman
Joaquín García Benítez
Iván Muñiz
Mathias Svensson Karlsson
Clara Torre García-Barredo

Client and problem description

The team and how we work

Tommy Ernsund - Android app developer.
Viking Forsman - Supporting.
Joaquín García Benítez - Project manager.
Iván Muñiz - Android app developer.
Mathias Svensson Karlsson - Android app developer.
Clara Torre García-Barredo - Client contact.



Roles and responsibilities

- Developer: Everyone
  - Write documentation and code
- Project manager:
  - Present deliverables and go through achieved work during steering meeting.
- Communication manager / Client contact:
  - Communicate with client via email when necessary.
- Android expert:
  - Advice on android app development autonomous when necessary.
- Support:
  - Take notes during steering meetings, allocate to task that lacks behind if necessary.

Interactions with client

Initial meeting and guided tour at Volvo Construction Equipments facility in Eskilstuna.

Further communication with the client handled via email by Clara Torre, our client contact.

Weekly meetings with the client where project implements are presented.

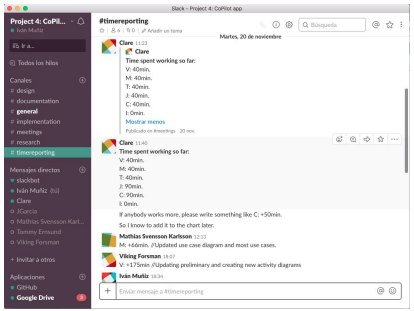
Skype for Business

The team's schedule and meetings

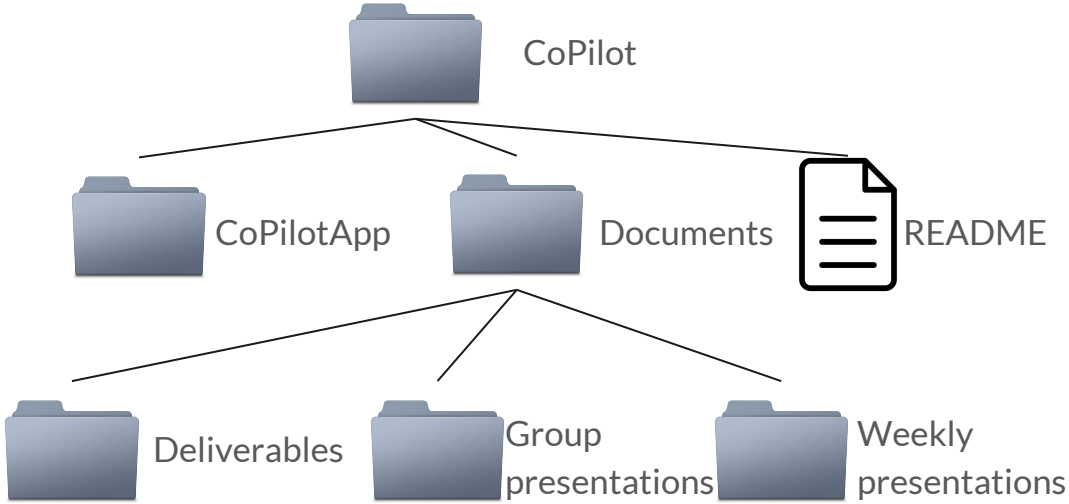
Monday	Tuesday	Wednesday	Thursday
13:30h-15h. Team meeting. 15h-15:25h. Meeting with the steering group.	8:15h-12h. Team meeting.		13:30h-14:30h. Team meeting. 14:30h-15:30h. Client meeting.

Time reporting routines

- Measure the time we spend working.
- Share that daily in SLACK with a brief summary of the work session.
- Add that up to the table in the weekly presentation.
- Add that up to the total amount of time spent.



Git folder structure



Quality ensurement routines

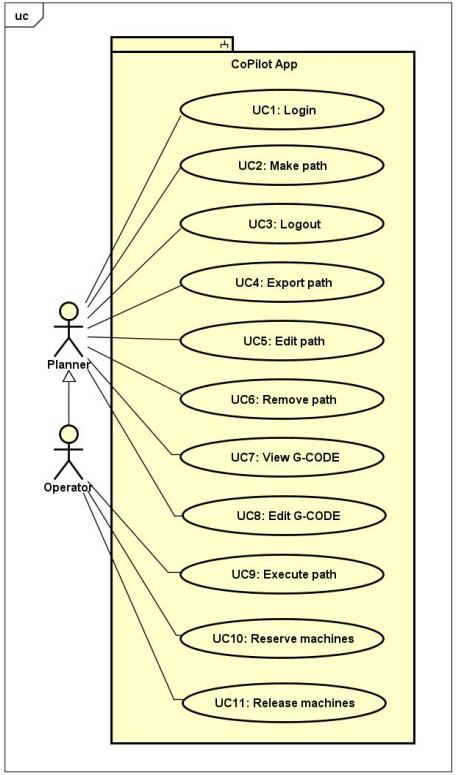
- Peer review.

- Continuous contact with the client.

- Clear documentation.
- Functionality testing.

- Acceptance testing.

Functional requirements - Use case diagram



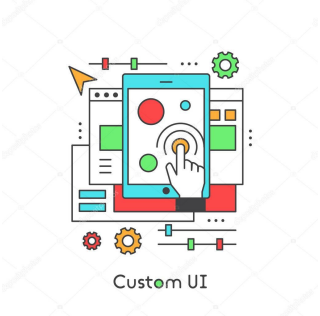
Example of use case explanation

ID:	UC10
Title:	Reserve machines
Priority:	Medium
Description:	Let an operator reserve a number of autonomous vehicles
Primary actor:	Operator
Pre-conditions:	1. User is logged in 2. Existing machines are available 3. Machines are released
Post-conditions:	1. Machines are reserved
Main success scenario:	1. User clicks reserve machines 2. User selects machines from list 3. User clicks reserve 4. User clicks the yes button on the confirmation to reserve 5. System reserves the machines
Alternative flow:	1. User clicks the cancel button 2. User clicks the cancel button 4. User clicks the no button 5. System error - cannot reserve machines
Frequency of use:	Every time the autonomous vehicles are changed.
Created by:	Mathias Svensson Karlsson
Date created:	2018/11/20
Last updated by:	Mathias Svensson Karlsson
Last updated:	2018/11/20

Important non-functional requirements

G-CODE

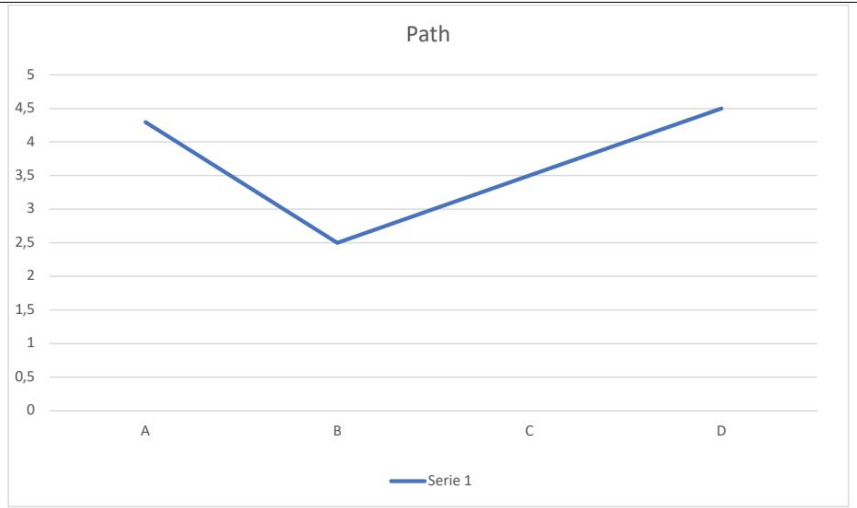
- Ease of use.



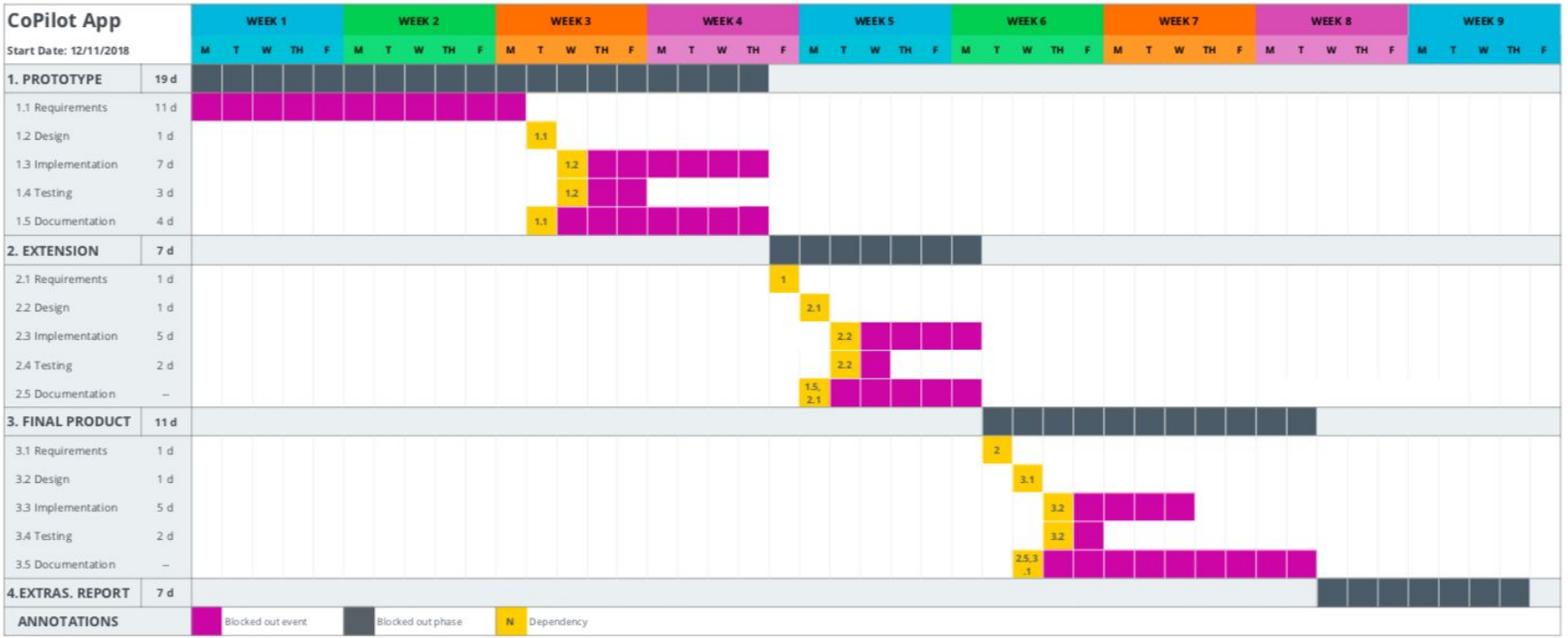
- Understandability of the code.

- Performance.

G-code table example

Company										Date	
Work site										Form filled in by	
Total excaveted volume, bm^3 or BCY										Body volume m^3 or CY	
Material										Body volume m^3 or BCY	
Excavation class										Body volume m^3 or LCY	
Density kg/bm^3 or lb/BCY										Load mass tonnes or tons	
Swell factor										Working shifts per year	
Loading equipment										Operating hours per working shift	
Bucket volume lm^3 or LCY										Productive time min/h	
Cycle time of loading equipment											
Road stretch	Length m(yd)	Gradient %	Rolling Resistance	Coefficient of traction	Curve radius m(yd)	Ground structure class	Note	Travelling time minutes		Loading	
								laden	unladen	Travelling laden	
A-B										Manouverung for tipping	
B-C										Planned activities	

Organisation of the workload



The initial backlog

- Create a "Wizard of Oz" representation of the app.
- Create use-case diagram for the system requirements.
- Create activity diagrams for each use case.
- Install Android Studio.
- Set up a GitHub repository for Android Studio app.
- Set up a kanban board on GitHub (for synchronisation).
- Set up a wiki page for the project on GitHub.
- Set up a SLACK group for communication within the team.
- Write project plan in accordance to the course description.
- Write the first draft of the Detailed Design Description.
- Estimate effort and create Gantt chart.
- Implement the GUI design for the app (colors, font, logo, etc).

The initial backlog (cont.)

- Implement a prototype version of the app (only basic functionality).
  - Add all activities
  - Add navigation between activities
  - Add login functionality
  - Add logout functionality
  - Add functionality to display created paths
  - Add functionality to create paths
- Implement extended functionality prototype
  - Add functionality to edit paths
  - Add functionality to remove paths
  - Add functionality of path validation
  - Add functionality to toggle view mode (visual representation or G-CODE)
  - Add functionality to export path to other users

The initial backlog (cont.)

- Implement the final product
  - Add G-CODE instruction for loading and unloading.
  - Perform quality assurance methods and remove bugs.
  - Perform acceptance testing.
- Write the project report
  - Write a brief introduction about the client and the project.
  - Write section about the planning phase.
  - Write section about the design phase.
  - Write section about the implementation phase.
  - Write section about project result and conclusion.
  - Write about the result of acceptance testing and how it was implemented.

Thank you for your attention! Any questions?