== Week 2 - 1 May ==
"Notes taken by Mike."

Every following meeting requires concrete goals in order for the process to make sense. An agenda is welcome, though it does not need to be as strict as the ones used in DBL projects. The main goal of this meeting is to get to know the expectations of the design document that needs to be handed in next Monday, and which should be presented next Wednesday. These and other milestones, as well as intermediate goals, are to be described in a week-based planning in this Wiki.

=== Design document ===
The focus of this document lies on the process of making the robot software succeed the escape room competition and the final competition. It requires a functional decomposition of both competitions. The design document should be written out in both the Wiki and a PDF document that is to be handed in on Monday the 6th of May. This document is a mere snapshot of the actual design document, which grows and improves over time. That's what this Wiki is for. The rest of this section contains brainstormed ideas for each section of the design document.

Requirements:
* The entire software runs on one executable on the robot;
* The robot is to autonomously drive itself out of the escape room;
* The robot may not 'bump' into walls, where 'bumping' is judged by the tutors during the competition;
* The robot has five minutes to get out of the escape room;
* The robot may not stand still for more than 30 seconds.

Functions:
* Detecting walls;
* Moving;
* Processing the odometry data;
* Following walls;
* Detecting doorways (holes in the wall).

Components:
* The drivetrain;
* The laser rangefinder.

Specifications:
* Dimensions of the footprint of the robot, which is the widest part of the robot;
* Maximum speed: 0.5 m/s translation and 1.2 rad/s rotation.

Interfaces:
* Gitlab connection for pulling the latest software build;
* Ethernet connection to hook the robot up to a notebook to perform the above.

=== Measurement plan ===
The first two time slots next Tuesday have been reserved for us in order to get to know the robot. Everyone who is able to attend is expected to attend. In order for the time to be used efficiently, code is to be written to perform tests that follows from a measurement plan. This plan involves testing the limits of the laser rangefinder, such as the maximum distance that the laser can detect, as well as the field of view and the noise level of the data.

=== Software design ===

The overall thinking process of the robot software needs to be determined in a software design. This involves a state chart diagram that depicts the global functioning of the robot during the escape room competition. This can be tested with code using the simulator of the robot with a map that resembles the escape room layout.

=== Tasks ===
Collin and Mike: write the design document and make it available to the group members by Saturday.

Kevin and Job: write a test plan with test code for the experiment session next tuesday.

Yves: draft an initial global software design and make a test map of the escape room for the simulation software.

== Week 3 - 8 May ==
"Notes taken by Collin."

These are the notes from the group meeting on 8th of May.

=== Strategy ===
A change was made to the strategy of the Escape Room Challenge. The new strategy is in two parts, a plan A and a plan B. First the robot will perform plan A. If this strategy fails, plan B will be performed. Plan A is to make the robot perform a laser scan, than rotate the robot 180 degrees and perform another scan. This gives the robot information about the entire room. From this information the software will be able to locate doorway and escape the room. This plan may not work, since the doorway may be too far away from the laser to detect it, or the software may not be able to detect the doorway. Therefore, plan B exists. This strategy is to drive the robot to the a wall of the room. Then the wall will be followed right hand side, until the robot crosses the finish.

=== Presentation ===
A Powerpoint presentation was prepared by Kevin for the lecture that afternoon. A few remarks on the presentation were:
* Add the 'Concept system architecture', modified to have a larger font.
* Add 'Communicating the state of the software' as a function
* Keep the assignment explanation and explanation of the robot hardware short

=== Concept system architecture ===
The concept system architecture was made by Yves. The diagram should be checked on its English, since some sentences are unclear. A few changes were made to the spelling. The content of the architecture remained mostly the same.

=== Measurement results ===
The first test with the robot did not go smoothly. Connecting with the robot showed more difficult than expected. When the test program was run, it was discovered that the Laser Sensor contained a lot of noise. A test situation, like the escape room, was made and all the data from the robot was recorded and saved. From this data, a algorithm can be designed to condition the sensor data. The data can also be used for the Spatial Feature Recognition.

=== Tasks ===
The task to be finished for next meeting:
* Spatial Feature Recognition and Monitoring: Mike, Yves
* Laser Range Finder data conditioning: Collin

* Control: Job
* Detailed software design for Escape Room Challenge: Kevin (Deadline: 9/5/2019)

The next robot reservations are:
* Tuesday 14/5/2019, from 10:45
* Thursday 16/5/2019, from 14:30

Next meeting: Wednesday 15/5/2019, 13:30 in Atlas 5.213

== Week 4 - 15 May ==
"Notes taken by Collin."

These are the notes from the group meeting on 15th of May.

=== Escape Room Challenge ===
The test of the software for the Escape Room Challenge was successful. Small changes have been made to the code regarding the currents state of the software being shown on the terminal. Also, the distance between the robot and the wall has been increased and the travel velocity of the robot has been decreased.
A state machine has been made and put on the Wiki which describes the software.

=== Wall detection ===
A Split and Merge algorithm has been developed in Matlab. It can detect walls and corners. The algorithm needs to be further tested and developed. Furthermore, a algorithm needs to be developed to use the information from the split and merge to find the position of the robot on the map. The current plan is to use a Kalman filter. This needs to be further developed.

=== Drive Control ===
The function to smoothly accelerate and decelerate the robot is not yet finished. Once the function has been shown to work in the simulation, it can be tested on the robot. This will be either Thursday 16th of May or the Tuesday after.

In order to succeed in the final challenge, better agreements and stricter deadlines need to been made and followed by the group.

=== Tasks ===
*Yves: Filter double points from 'Merge and split' algorithm.
*Mike: Develop the architecture for the C++ project.
*Job: Code a function for the S-curve acceleration for x ,y direction and z rotation.
*Kevin: Develop Kalman filter to compare the data from 'Merge and split' with a map.
*Collin: Develop a finite state machine for the final challenge

The next robot reservations are:
* Thursday 16/5/2019, from 14:30

Next meeting: Wednesday 22/5/2019, 13:30 in Atlas 5.213

== Week 5 - 22 May ==
"Notes taken by Kevin."

These are the notes from the group meeting on 22th of May.

=== Finite State Machine and Path planning ===
Collin created a Finite state machine of the hospital challenge. The FSM is a pretty complete picture of the hospital challenge, but a different ‘main’ FSM needs to be made in which the actions of the robot itself are shown in a clear manner. Collin also came up with a path planning method. In this method important point are selected on the given map, which will be connected with each other where this is possible. The robot will then be able to drive from point to point in a straight line. If some time is left, we could eventually improve to robot by letting it drive between point in a more smooth manner.

=== Wall detection ===
Yves has continued working on the split and merge algorithm. He has tried to implement his matlab implementation in C++, but this has proved to be more difficult than anticipated. He will continue working on this.

=== Drive Control ===
Job has continued working on the drive control, which is now almost finished. Some tests needs to be done on the real robot to see if it is functioning properly in real life as well. Furthermore, the velocity in the drive control needs to be limited, as it is still unbounded at this time.

=== Architecture ===
Mike has worked on creating the overall architecture of the robot. All the other contributions can then be placed in the correct position in this architecture.

=== Spatial Awareness ===
Kevin has worked on the Kalman filter and spatial recognition. His idea is to first combine the state prediction and odometry information within a Kalman filter to give a estimated position and orientation. This estimation can then be combined with laser range data to correct for any remaining mistakes in the estimation.

=== Last Robot reservation ===
During this reservation we were finally able to quickly set up the laptop and the robot for the first time without any issues. During this test we collected a lot of data by putting the robot in different positions in a constructed room, and saving all the laser range data in a rosbag file. Most of the data is static, with the robot standing still, but we also got some data in which the robot drives forward and backwards a little bit.

=== Next robot reservations ===
The next reservation is Thursday may 23. During this reservation we will have two hours to test the drive control made by Job. Particular attention will be given to static friction and the maximum possible acceleration of the robot. Furthermore, since we want to implement multiple threads in our program, we would like to know how much the robot can handle in real life. As such, a stress test will be made to see how much the robot can handle. The reservation for next week will be made on Wednesday may 29, on the 3th and 4th hour.

=== Tasks ===
*Job: finish drive control and integrate it in the architecture. Also create a main FSM with Collin.
*Kevin: Work on a implementation of the Kalman filter and spatial recognition software.
*Collin: Continue working on path planning implementation. Also create a main FSM with Job.
*Yves: Continue working on the C++ implementation of split and merge. Also look into speak functions of robot.
*Mike: Work on collision detection and working on creating multiple threads.

*Everyone: Read old wiki's of other groups to get some inspiration.

Next meeting: Wednesday 29/5/2019, 13:30 in Atlas 5.213

== Week 6 - 29 May ==
"Notes taken by Job"

These are the notes from the group meeting of the 29th of May.

=== Progress ===
There has been little integration of functions and everyone has kept working on their separate tasks. It is vital to write the state machine in code so the different functions can be implemented and tasks that still need to be completed can be found.

Mike has worked on the potential field implementation and has achieved a working state for this function. This function needs to be expanded with a correction for the orientation of the robot.

Yves has worked on the spatial recognition integration of the Ransac function. This needs to be finished so it can be used for the Kalmann filter Kevin has worked on.

Kevin needs the work from Yves to finish the Kalmann filter and needs to add a rotation correction.

Collin has worked on the shortest path algorithm which is ready to be used.

Job has improved the Drivecontrol functions after last weeks test session and discussed the integration with the potential fields with Mike.

=== Planning ===
Since time is running short, hard deadlines have been set for the different tasks:

*State machine (+ speech function integration) - 02-06-2019, 22.00 - Collin + Job (+ Mike)
*Kalmann filter - 04-06-2019, 22.00 - Kevin + Yves
*Presentation - 04-06-2019, 22.00 - Kevin
*Driving - 05-06-2019, 22.00 - Mike + Job
*Cabinet procedure - 02-06-2019, 22.00 - Collin + Job
*Map + Nav-points - 05-06-2019, 22.00 - Yves
*Visualisation OpenCV - Extra task, TBD

=== Test on Wednesday 14.30 - 15.25 ===
*Test spatial recognition

=== Test on Thursday 13.30 - 15.25 ===
*Driving + Map
*Cabinet procedure
*Total sequence

== Week 7 - 6 June ==
"Notes taken by Mike"
=== Progress ===
Kevin has been working on the presentation and the perception functions that fit the map on the detected features. Simple tests suggest that it works by manually feeding the functions with made

up realistic points, as well as random points that need to be ignored by the function. For some reason the code does not execute repeatedly though. Either way, the code requires some fine-tuning. This function takes the estimated robot position (odometry) and the LRF data as inputs and has the corrected position as an output. It needs to be extended to take the previous navpoint as the origin of movement. Kevin expects this to be ready for testing by tomorrow's session.

The presentation is almost done. The architecture slide needs to be simplified to prevent an overwhelming amount of information being visible on screen. The same goes for the state machine.

Collin has integrated the state machine as much as possible. More public functions need to be made in the WorldModel object that allows the state machine to check whether the program can progress to any following state or not.

Job and Mike were working on the DriveControl object. The current challenge is driving from point to point. This involves correcting the angle when it deviates from its straight trajectory as a result of the potential vector pushing it away. This is going to be tested in the testing session after this meeting. It also requires implementing the relative position of the end point of the current line trajectory, from Kevin's position estimation function.

Yves (absent) has been working on implementing the published world map and supplying it with navpoints.

=== Planning ===
Thursday 6-6: appointment to work together in Gemini South OGO 0 from 9:00 to 10:45, then in Gemini South 4.23 until 12:30. The testing session is from 13:30 until 15:25. The plan is to attempt to integrate "everything" before this session to simply test as much as possible.

Tuesday 11-6: appointment to work together from 8:45 until the testing session from 9:45 until 10:40. The entire code "should" be done by now. After the testing session, everything should be fine-tuned in the simulation environment.