Project Plan

**Group 6:** Joakim Karlsson, Fredrik Olsson, Martin Stålberg and Mikaela Åhlén

State of the project:

We have finished the design of the robot. It consists of a standard vehicle platform with two driving wheels on the middle of the sides, one free-moving stabilization wheel at the back, a long light sensor array just above the ground in the front of the vehicle, covering the entire width of the vehicle and extending outwards to the sides a couple of centimeters, and a camera mounted on top of the vehicle on a horizontally rotating axle centered right on top of the center of the driving wheels axis. The camera can sweep horizontally 360° alternating between clockwise and counterclockwise rotations to prevent twisting up the cable connecting the camera and the brick.

The robot currently operates in a rectangular area enclosed with black walls rising above the horizontal plane in which the camera rotates (from here on referenced as the camera plane). In three of the corners of the area are landmarks, red paper pillars with triangular cross sections, stretching from the ground plane to about 10 centimeters above the camera plane. The camera is calibrated to detect red objects and returns the bearing to detected objects, i.e. the angle between a line from the robot to the landmark and a line starting in the landmark with the same direction as the robot.

An algorithm that performs a 360° scan with the camera and returns bearings to found landmarks is finished. Red objects larger than a minimum size, completely inside the field of view and with centers below the camera plane are detected. The camera performs one 360° sweep and sends an array of all bearing data to the computer via bluetooth. On the computer the received data is clustered and the bearings to the three landmarks are estimated as the mean values of the three largest clusters. Clusters are defined as groups of data points in which the distance to the closest neighbour is below a threshold value, currently 5°. Choosing only the largest clusters and having a low threshold distance for clustering serves to remove outlier data.

The coordinates of the landmarks are known in advance and the bearings are associated with the correct landmarks with a data association algorithm resembling a maximum likelihood algorithm that maximizes the probability of the data association given estimations of the bearings derived from the estimated position and heading provided by the robot’s odometry.

With correctly associated landmark bearings and landmark coordinates the position and heading of the robot is calculated via a triangulation algorithm. The triangulated pose (position and heading) and the pose provided by the odometry are fused by a particle filter to a final estimate of the robots pose. The fusion takes into account the different variances of the triangulated pose and the pose from the odometry.

Two different particle filters have been implemented, one for the position of the robot and one for the heading. It uses the odometry from the robot as the model and the pose given from the camera as observations. These two particle filters are then used to correct the position and heading of the robot, since odometry alone would cause the estimated position and heading of the robot to drift from the true values. The particle filters are done, but to fuse them together with everything else has yet to be done.
The system can currently send a waypoint from the computer to the robot, drive the robot to the waypoint and estimate the robot’s position with odometry and camera sweeps.

To do:
Write code for the light sensors and integrate it with the rest of the system. We have just barely started working with the light sensors.

The graphical interface displaying the map representation must also be further improved. As it is now we can not distinguish between discovered and undiscovered areas in the GUI.

The particle filter is currently being fused together with everything else, so we need to make sure everything works and start debugging.

The positioning system must be carefully tested so that we know how well our robot performs.

Time schedule (~deadlines):
Division of labour in parentheses from 13 dec and onwards.

6 dec
Merge camera code and code for particle filter.
Determine relation between bearing errors and position error.
Finish the pathfinding algorithm (map, pathfinding and path simplification and optimization).

13 dec
Test the positioning system. (Everyone)
Integrate the light sensors in the system. (Mikaela and Joakim)
Improve the GUI. (Martin and Fredrik) Done
Improve, clean up and comment the codebase. (Everyone)

20 dec
Finish and test the robot. (Everyone)

10 jan
Project report finished. (Everyone)

15 jan
Presentation finished. (Everyone)