

## **SEMESTER 1:**

Shu - Hardware Design + Obstacle Avoidance

Ashley - Hardware Design + Color sensor orientation

Eric - Color Sensor orientation + Obstacle Avoidance

Andrei - PID line following + code integration

## **SEMESTER 1 GOALS:**

1. Robot design and assembly - Robot should be sufficiently compact so as to navigate smoothly around the track. Should be designed for optimized mobility, and design should pose no limiting factor to general movability.
  2. Research and field testing the color sensor - Get familiar with EV3 color sensor. Look at all sensor modes and methods in API. Perform some preliminary tests to determine which mode will be the most suitable. Then establish how to use that sensor mode for the PID system.
  3. Making the PID system - Create a PID system that will allow the robot to follow the line smoothly, with no jerking side to side. Achieving this requires PID constants to be optimized later through trial and error.
  4. Obstacle avoidance controller - Create a PD controller to allow the robot to navigate round the obstacles. Test the obstacle circumnavigation at various different distances from the obstacle. Identify edge cases and patch them over with contingency code.
  5. Tune both controllers optimally - Use rigorous process of trial and error to find optimal configuration of controller values for both systems. Start by setting all constants to 0, then increase  $K_p$  as much as possible until the motion of the robot becomes too erratic, then increase the other 2 one at a time. Do this until the system is smooth and stable and can work at a reasonably high speed
- Enable the colorsensor to be able to read a red line to stop its motion - Using the color sensors red mode (which we have decided to use for the line following PID controller) figure out a way to concretely identify the color red and code a sequence of actions that will allow it to pass over the line, and then resume the line following code as normal.
- End result: Robot should be able to successfully complete 4 laps in 5 minutes

## **TIMELINE**

WEEK 1 : Group formation and Hardware Design. Since we didn't have a clear idea of the robot worked and which sensors we needed this took some time

WEEK 2 : Hardware Design

WEEK 3 : Color Sensor testing and Orientation

WEEK 4 : Some Hardware re-design to accomodate for the color sensor. Pseudocode for color sensor written

WEEK 5 : PID Line following Pseudocode written. Hardware design reworked to include an ultrasonic sensor for PD obstacle avoidance

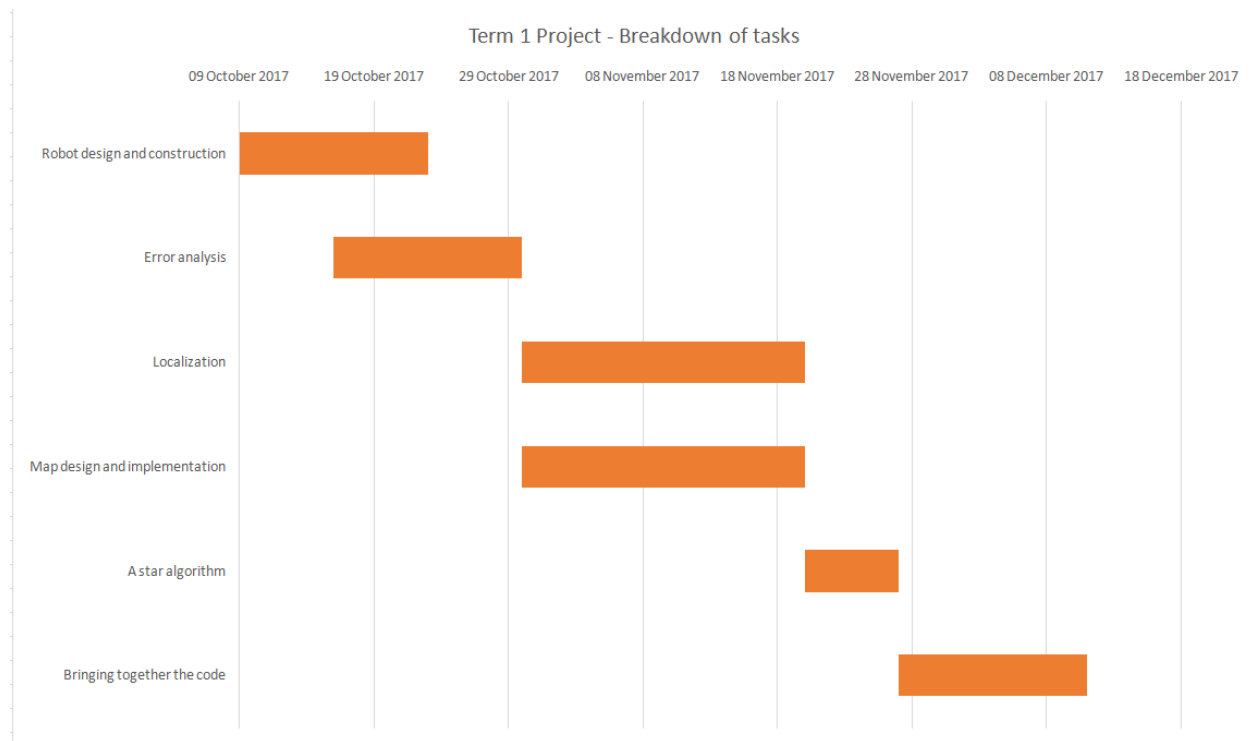
WEEK 6 : PID Line following code imlemented with color sensor code. PD Obstacle Avoidance Pseudocode written

WEEK 7 : PID Line following tuning. PD obstacle avoidance code written

WEEK 8 : PID Line following tuning

WEEK 9 : Code Integration and tuning

WEEK 10 : Final tuning and Grading



**Final Outcome:** Robot ran three laps but was unable to read a starting line.

## SEMESTER 2:

Shu - Hardware Design + Path Planning

Ashley - Hardware Design + Path Planning

Eric - Localization + Differential Drive

Andrei - Map + Differential Drive

### SEMESTER 2 GOALS:

1. Design a robot. The robot should be compact enough to navigate tight spaces yet mobile enough to be able to make tight turns without falling over.
2. Experiment with the gyrosensor (The gyrosensor was not used in the first task). Discover how the gyrosensor works, how consistent it is, and how to best work around its lack of accuracy.
3. Create a map of the arena. The map should be in java code and should correctly represent the arena and its obstacles to scale without being too overwhelming.
4. Translate the map into a readable format for the robot. The robot should be able to read the map and figure out a way to navigate the obstacles.
5. Create an A\* algorithm the robot can use to plan a path in the arena.
6. Localize the robot so that it is aware of its current location in the arena.

7. Translate the A\* algorithm and Localization into a Differential Drive so that the robot can control its speed and direction given instructions.

End Result: Robot should be able to successfully navigate the arena from the starting point to the goal and back twice in 5 minutes.

## TIMELINE

WEEK 1 : Hardware Design and Task Allocation

WEEK 2 : Hardware Design. Initial Gyrosensor related tasks Pseudocode written

WEEK 3 : Gyrosensor alignment tasks done and recorded

WEEK 4 : Board dimensions measured and Map dimensions planned

WEEK 5 : Map Code finished

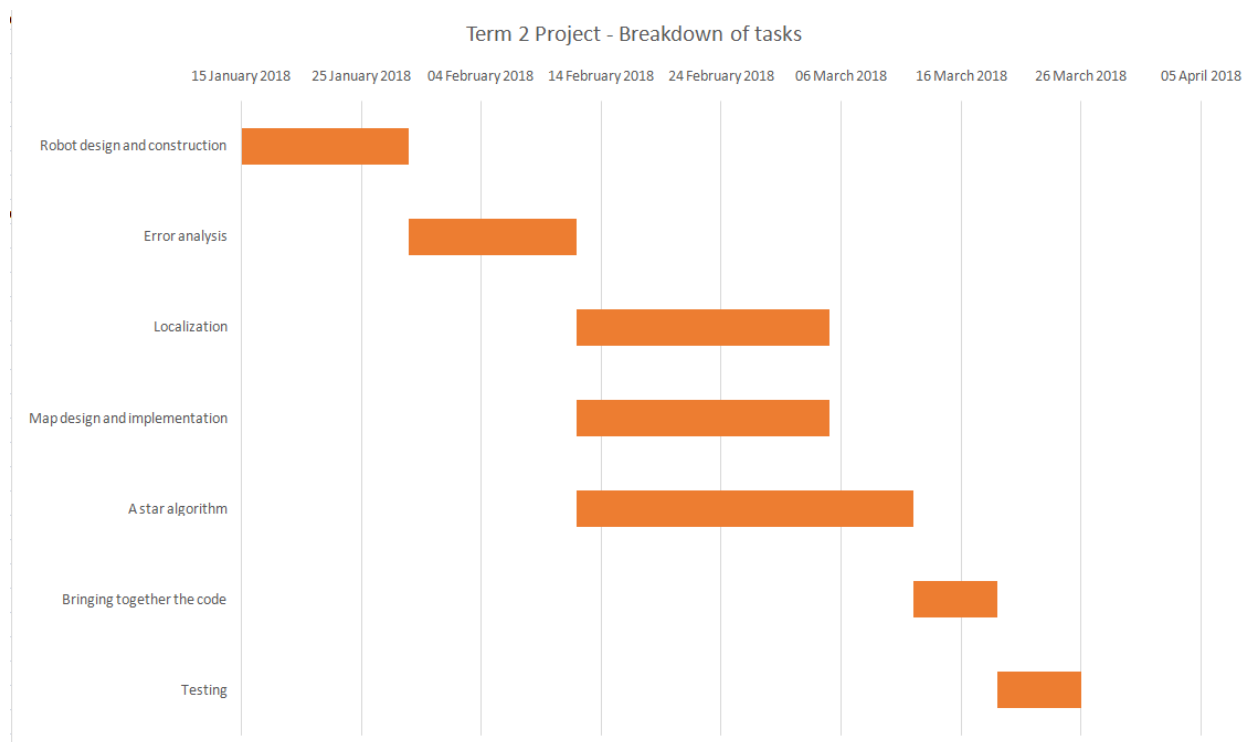
WEEK 6 : Localization and Path Planning pseudocode written

WEEK 7 : Pseudocode for Path Planning rewritten. Some Back end code for Path planning written

WEEK 8 : Path Planning finished

WEEK 9 : Localization + Differential Drive Code written

WEEK 10 : Grading



**Final Outcome:** Robot was unable to localize correctly during grading.

**Video Reshoot:** Robot was able to navigate to the goal and back to its original starting point.