***Note***: Ingick i embedded delen. Blev mycket ”vi” då vi arbetade tillsammans på plats mestadels av tiden.

**Vecka 1**

* Veckan började vi med att ha ett möte där vi diskuterade vidare kring uppgiften.
* Först skapade vi GitHub och Trello, sedan gick vi gemensamt igenom vad vi skulle dela upp våra High level requirements till för tasks för dagen.
* Det slutade med att min del (embedded/robot delen) skulle göra följande:
* Bestämma vilken modell av robot att bygga
* Bygga modellen
* Dessa två gjorde vi sedan tillsammans. Utöver detta så tittade jag på programmet som används för att programmera roboten: mBlock. Jag tittade lite på dokumentation och en video för att se hur, när vi drar i gång, vi ska göra för att få roboten att röra på sig, vilket blir vår Hello World.

Senare i veckan fokuserade vi mer på kodning av roboten. Först fixade vi en arduino fil, driver och så vidare så att vi kan flasha roboten från arduino IDE. Sedan fixade vi så att roboten håller sig inom en area och kollision detektion. Jag kommenterade koden. Vi började också titta på hur raspberry pi:n kan kopplas till roboten och prata med den.

**Vecka 2**

* Veckan började vi med sprintplanning, sedan gick vi in i våra ”avdelningar” och estimatade och prioriterade lite tasks.
* Sedan arbetade vi i våra avdelningar:
  + För mig innebar detta arbete på kommunikationen mellan Raspberry pi och arduino. Tillsammans med Viktor skötte vi Raspberry pi delen och kunde läsa meddelande och skicka (seriellt). Sedan tittade vi lite på nästkommande tasks.
* Senare i veckan fortsatte vi med att få igång raspberry pi kameran. Vi fick den att ta bild och började sedan att arbeta med att få arduino att skicka seriellt vid en collision till raspberry pi:en.
* Vi skrev först lite kod och laddade upp på github, som vi sedan ladda ner på raspberry pi:en. Provade sen på pi:en kopplad till arduinon. Efter några ändringar i koden hur det seriella kommandot togs in så funkade bildtagning (bilden sparas på PI). Nästa arbete var att få bilden att efter blivit tagen, uppskickad till backend. Detta gjorde vi då med en POST request.
* Nästa dag forstsatte vi med backend och såg till att bilden faktiskt kom upp på backend, vilket funkade efter en liten ändring av Adina (ändra resolution). Sedan började vi arbeta på att få robotens sväng-logik. Nu svänger den i stället 5 ggr till höger sedan 5 ggr till vänster. Sedan skrev vi Low level requirements i Trello för varje High level requirements.

**Vecka 3**

* Började veckan med sprint planning och genomgång av Trello med hela gruppen, sedan hoppade vi in i våra separata grupper och gick in mer i detalj tasks för veckan.
* Sedan arbete i skolan med att få igång Bluetooth modulen på roboten.
* Vi fick igång bluetoothen med appen som MBot ger ut.
* Vi skrev sedan lite custom kod att styra med en egen eller serial terminal bluetooth app.
* Vi upptäckte sedan att seriella kommunikation och bluetooth modul va på samma TX0 RX0 port.
* Vi bestämde då för att köra bluetooth på raspberry pi:en istället, som sedan pratar med arduinon.

**Vecka 4**

* Veckan fortsatte med arbete att få igång Bluetooth på Raspberry PI. Funderade på om vi skulle köra sockets, serial osv. Lutar mot sockets.
* Fick några problem med hur vi skulle överföra filer och så vidare så vi tittade sedan lite på headless SSH connection från datorn till PI. Det fick vi att funka så vi kan arbeta snabbare i framtiden.
* Nu fortsätter vi med att prova våra sockets filer (client och server, server till PI). Jag gick också och pratade lite med backend om strukturen av kommunikationen mellan robot och backend, gav lite feedback osv.
* Vår Bluetooth Service kunde inte starta så de fick vi också fixa med en del felsökande.
* Började titta på hur Gyrot fungerar vilka funktioner och så vidare. Pratade med Mobile-team om att de kan börja implementera Bluetooth, vår Bluetooth server på raspberrypi verkar funka och redo att testas.

**Vecka 5**

* Vi började med sprint planning och att kika lite mer på Bluetooth server. Efter hittat lite andra tutorials så bytte vi lite av vår kod, t.ex. tog bort mac adress och ändra lite ports osv.
* Sedan hade Adina jobbat lite på positionering men upptäckt lite bugs. Så vi arbetade sedan med att försöka räkna ut timing för hur långt roboten åkt framåt, svängt och så vidare så vi kan använda det i våra kalkylationer senare.
* Nästa dag provade vi Bluetooth servern på RPI med android appens bluetooth lösning. Det gick då att connecta och skicka meddelande. Nästa steg blev att ta emot dessa meddelande, skicka de seriellt till arduino (mBOT) och sedan gör det som sägs. Denna kod hade redan påbörjas så några fixes las till. Detta ska vi prova nästkommande dag. Jag och Viktor kikade lite på bootscript för att få igång Bluetooth servern vid boot av RPI och Adina tittade mer på arduino koden.