Projektjournal (PA1414)

**onsdag 12 sep**Idag träffade vi våra kunder för första gången. Första mötet var ganska konstigt eftersom kunderna planerat vårat projekt som ett grupprojekt istället för ett individuellt projekt. Jag påpekade detta flera gånger men kunderna stod på sig och menade att ”det är så här ni kommer jobba i arbetslivet senare”.

Uppdraget kändes överväldigande. Kunderna har en plats i skolan där de har robotar med olika kameror, och de vill att vi ska utveckla en del av ett system som kan användas för att exvis ansluta kamerorna till en app och känna av rörelser etc. Detta ska utvecklas med *microservice architecture*, vilket innebär att systemet består av ett antal fristående delar, och tanken är att var och en av oss i gruppen ska utveckla en del var, som sedan i slutänden kan sättas ihop till en fungerande produkt. Vi bestämde ytterligare ett möte med kunden **samma tid nästa vecka** och vi ska internt i gruppen mötas **fre 14 kl 10.00** för att diskutera hur vi kan dela upp uppdraget.

**fredag 14 sep**

Jag ska göra en iOS app. Filer skickas i json-format.

* Kolla upp delevoper account
* Läs på om swift
* Kolla upp format som kan streamas till iOS

**Frågor till nästa tillfälle**:

* **Vad får man för data från kamerorna?**  - endast en videofeed
* **Vad ska appen göra mer exakt?** – ta emot live feed
* **Navigering, sensorer på robottar?** – endast kamera
* **Behövs en server?** – videofeeden ska troligvis streamas från kameran till en server, som sedan kan användas av alla olika appar.

Vi kom fram till att

* Vidar gör webbapp och android app
* Ludvig gör motion detection
* Jag gör iOS app
* Simon tar live feeden
* Hampus tar styrning

**måndag 17 sep**

Kollade på en tutorial <https://www.youtube.com/watch?v=5b91dFhZz0g> om hur man börjar utveckla en iOS-app. Följde med steg för steg i videon, men den simpla appen som skulle byggas fungerade ändå inte. Det har antaligen någonting med att jag har en annan swift/xCode-version än den som används i videon. Men jag började i alla fall förstå lite hur språket fungerar.

Skaffade ett developer account, verkar vara gratis så länge man inte vill publicera appen på App store.

**onsdag 19 sep**

Pratade lite mer om vad var och en ska göra under projektet, men det blev inte mycket tydligare. Men jag vet i alla fall att jag ska börja lära mig swift (visste redan innan), och att jag ska fixa en app som på något vis tar in en dummy-videofeed, som senare ersätts av en riktig feed från radion. I appen ska man även kunna skriva in olika värden, exempelvis justera hastigheten på roboten.

Vi fick förtydligat att kameran (till en början) kommer att sitta ovanför en rektangulär plan där robotarna kommer att röra sig och hela planen kommer att vara synlig hela tiden.

Till nästa möte ska vi skriva user stories och se till att vi har en bra backlog.

See display with object detection. Kunden skall till nästa gång fixa ett gemensamt repository på github där vi kan ha alla saker gemensamt, men i separata mappar.

**Fredag 21 sep**

Kollade och hängde med i en swift-tutorial där man bygger en kalkylator-app. Lärt mig hur man kan koppla knappar till funktioner, läsa av värden från knappar etc. Börjar skriva projektplan.

**söndag 23 sep**

Fortsatt lite med swift tutorials. Har problem med att kompilera/köra appen trots att jag skrivit exakt likadant som personen i guiden. Testade även att kopiera koden och får även då samma problem. Ska testa någon annan tutorial, antaligen någonting som ändrats med syntaxen i den nyare swift-versionen.

Hittade en annan bra tutorial i serien ”Developing iOS11 apps with swift”, som finns på iTunes. Följde med i den hela 1 timmes-långa guiden och fick allting att fungera i slutänden. Lite knixit, men börjar förstå en aning hur swift fungerar. Nästa gång ska jag lära mig om MVC-arkitekturen, vilken man alltid använder när man skapar en iOS-app.

**söndag 30 sep**

AVPlayer kan användas för att strömma video in i en layer. AVKit och AVFoundation

Kollade en tutorial hur man använder AVKit för att spela upp en lokal video i en iOS-app, men får det inte att fungera. Tutorial som jag kollade på var från 2016, och syntaxen verkar ha förändrats en hel del sedan dess. Nästan allting som jag skrev var fel, och xcode föreslog ändringar. Dock vet man inte om det som xcode föreslår har samma funkionalitet som den kod som används i videon. Oavsett så fungerade det inte.

<https://www.youtube.com/watch?v=fhD7hXrpExE>

**guard** – används som en sorts if /else sats. Skriver före variabel, om denna variabel inte får ett värde körs else-satsen som kommer efter variabeldeklarationen. Används exvis för att kolla om man lyckats läsa in data från en text-fil. Om det inte lyckats, meddela detta.

Varför används frågetecken/utropstecken i Swift?

**?:**

**optionals:** Ett optional value innehåller antingen ett värde eller nil. ”?” används efter en typ för att visa att värdet är **optional**

**måndag 1 okt**

Läst om ett framework i swift som heter Codable, detta kan användas för att parsa json-objekt. <https://medium.com/xcblog/painless-json-parsing-with-swift-codable-2c0beaeb21c1>

**onsdag 3 okt**

Lyckades läsa in en video från http länk i Swift mha AVkit och AVFoundation. Nästa steg blir att läsa in en live ström.

HLS format stöds av apple och android.

Hittat en bra video-serie om hur man tar fram iOS-apps.

**fredag 5 okt**

Börjar med wireframes till appen. Kollar lite hur man ansluter till en UDP-server, vilket vi kommer använda till live streamen. UDP är ett anslutningslöst protokol, så man ansluer inte direkt till servern. Däremot kan UDP-paket tas igenom genom att man gör dessa steg:

* Skapa en UDP-socket
* Binda socket till port
* Joina multicast group
* Appen kommer börja ta emot UDP-paket

<https://stackoverflow.com/questions/50603869/java-send-json-over-udp>

Update: verkar som att UDP inte fungerar för att ta emot live stream. Behöver kolla upp HLS istället, som streamar över http.