

Team reflektion

Kundvärde och fokusområde

Gruppen har fått en ökad kommunikation vilket har blivit en framgångsfaktor för gruppen i termer av mål och kommunikation. Genom en ökad kommunikation mellan samtliga medlemmar har gruppen även fått ett utökad lärande om flutter, firebase samt ett förbättrad teamwork. Detta samarbete och gruppinsats har även gett möjligheter för en god och förutsägbar enligt scrum som dessutom gett förbättrade möjligheter för vidare arbete i nästa sprint. Prioritet från förra veckan var att se till att alla har en fungerande utvecklingsmiljö och fatta beslut kring vilken API man ska använda sig av. Gruppen har avklarat punkten av att välja en en utvecklingsmiljö (Android studio eller VSCode) samt tagit information från en filmdatabas. Detta innebär att gruppen fått en relativ god grund i att komma igång med att utveckla en nativ mobilapplikation.

User Stories

Vi har fått en bättre struktur i vår backlog under veckan efter att produktägaren har jobbat mycket med att fixa med acceptanskriterier så att det blir tydligare vad som krävs för att färdigställa user stories. Framöver kommer vi jobba för att göra dem ännu tydligare för oss som arbetar med dem genom att göra acceptanskriterierna mer tekniska. Kunden kommer ju inte ta del av dessa och därför behöver de inte skrivas på ett sätt som är förståeligt för gemene man. Tidigare har vi i princip baserat våra "effort estimations" på medel och stor med siffrorna 5 & 10. Planen är nu att utveckla detta med ett bredare spektra av siffror, t.ex Fibonacci och därigenom kunna skilja bättre på hur lång tid en user story beräknas ta.

Designval

Vi har skapat en gemensam fil där vi tydligt förklarar dem flesta designmönster som vi använda oss i projektet. Vi använder api:et från The Movie DataBase (TMDB), vilket resulterar till en avgiftsfritt utnyttjande av deras database så länge källan anges. Detta möjliggör att kunden kan förbrukar appen utan att oroa sig av olika kommersiella rättigheter.

Vi har valt att utveckla appen enligt Model- View-Controller design pattern (MVC).

MVC kommer hjälpa att minska på programmets komplexitet så att vi lättare kan utöka appens funktionaliteter vid kundens behov. Slutligen valde vi att integrera Firebase i appen, vilket gör det möjligt för kunden att i realtid spara, hämta och interagera med information från andra användare utan en direkt uppkoppling. Detta möjliggör funktioner som gemensamma listor och cloud storage.

Agila tillämpningar

Daily Scrum

Vi har en kanal i Discord som används för vår Daily Scrum. I kanalen så meddelar vi hur långt vi har kommit i den nuvarande Sprint:en, om man eventuellt stött på ett problem i utvecklingen av produkten och om man behöver anpassa Backlog:en efter rådande omständigheter. Användningen av denna kanal har gått bra till och varje medlem i gruppen har kunnat få en inblick på hur väl andras user stories har gått för de.

Det som kan förbättras är frekvensen som man använder denna kanal (början/slutet av dagen), men man kan även jobba på att klargöra vissa detaljer (hur man har gått tillväga för att göra sin del av Sprint:en, vilka problem man har stött på och löst, m.m.).

Sprint Review

Förra veckan hade vi sprint review/retrospective kombinerat och lite ostrukturerat där vi allmänt gick igenom vad alla i laget upplevt och fått gjort under veckan. Det kändes ostrukturerat och inte jättegivande. Dock med tanke på att mycket av projektet inte hade kommit igång var det inte hela världen.

Denna veckan hade vi de mötena separat och använde oss av definitionerna på [scrum.org](https://www.scrum.org) för inspiration till struktur. Resultatet var främst att vi fick mer ut av retrospective och fick tydligare åtgärder för nästa sprint. Med separat sprint review var största nyttan att vi fick insyn i vad alla haft för problem så vi hade informationen till vad som behöver förändras. Vi har dock ännu inte tillräckligt högt tempo med våra stories för att få full nytta av en komplett sprint review.