



# GÖTEBORGS UNIVERSITET

*Institutionen för tillämpad IT*

*Systemvetenskap: IT, människa och organisation*

TIG167

HT-18

## **GÖSTA-appen**

*Projektrapport TIG167*

2019-01-19

Annie Petersson  
Johannes Carli

[guspetancd@student.gu.se](mailto:guspetancd@student.gu.se)  
[guscarlijo@student.gu.se](mailto:guscarlijo@student.gu.se)

## Contribution

Under hela projektet har grundidén varit att båda deltagarna ska vara delaktiga i varje del av utvecklingen av applikationen. Dels för att det gör våra kunskaper mer kompletta men också för att båda ska kunna arbeta på olika delar utan att vara beroende av den andra. Därför ansåg vi att det var viktigt att båda hade lika mycket kunskap om applikationens olika delar. Detta har också speglat projektet då det i stort sett inte finns någon klass där båda inte varit delaktiga. Till sist anser båda att vi haft ett otroligt bra samarbete och lyckats hitta en balans mellan att programmera ihop, eller ”parprogrammera”, och dela upp applikationen i mindre delar och delmål och arbetat med dessa på egen hand för att sedan mötas upp och sammanfoga våra delar.

När projektet väl delades upp fokuserade Annie främst på att ta fram nya funktioner, så som QR-läsare och skakningsfunktion för att slumpa fram företag medan Johannes fokuserade på användarvänligheten och GUI:t.

## Tools

Under projektets gång användes olika verktyg och program; Atom för att skriva den servlet-relaterade koden, emacs för att skriva skriptet som kompilerar servletkoden och startar servern, Android Studio för att skriva den applikationsrelaterade koden samt Git för att samarbeta. Ingen av oss har arbetat i Android Studio tidigare men tyckte att det kändes bra då det finns mycket dokumentation till hjälp. Det svåraste var att hantera att de olika androidversionerna ej stödjer vissa saker vilket gjorde att vi fick tänka om en del.

Emulatorn som tillhandahölls av Android Studio upplevde vi som mycket behjälplig, och var enkel att arbeta med för att testa och leta efter buggar i applikationen. Däremot tyckte vi att testa med en fysisk telefon var mycket svårare. Problemen med en fysisk telefon beskrivs mer ingående längre ned. Detta hade dock säkerligen kunnat bli smidigare om vi lagt ned mer tid på att skriva bra testkod och till exempel använda oss av en stub. Vi ansåg dock att det skulle ta för lång tid att skriva bra testkod själva som vi skulle kunna lita på, så att vi egentligen inte skulle vinna någon tid.

Det vi upplevde mest problem med var Git, det var väldigt svårt att förstå hur man skulle hantera mergekonflikter när bash ”öppnade upp konflikten” i någon form av program som inte verkade acceptera några kommandon. Men efter hjälp från Rikard kunde vi ställa in att Atom skulle vara standardprogram för att hantera sådant, det underlättade mycket. Vi fick lägga mycket tid på att försöka lösa sammanfogningskonflikterna och det var ofta svårt att förstå hur man skulle gå till väga. Denna tid hade istället kunnat läggas på utveckling. Projektet hade kunnat gå smidigare om båda hade lärt sig att använda Git korrekt.

Vi hade även lite problem med att testa på en fysisk telefon i början eftersom att den telefon som användes hade Android version 9 vilket visade sig inte stödja IP-adresser med tecken andra än siffror. Det tog ett tag innan vi listade ut detta och hann testa både det ena och det andra utan att lösa problemet. T.ex. testade vi att lägga till inställningar för nätverkssäkerhet men då detta inte hjälpte så testade vi med en äldre Android version och då fungerade det. Vi kom slutligen på vad det hela berodde på genom att läsa dokumentationen för förändringarna i och med uppdateringen till Android 9. Då vi hade begränsat med tid valde vi att fortsätta testa med en äldre telefon som alltså hade en tidigare Android version. Även där uppstod en del problem då kopplingen till servern var labil, det fungerade fint att nå servern via telefonens webbläsare men för applikationen fungerade bara serverkopplingen då och då. Eftersom att vi stötte på så mycket problem bestämde vi oss för att fortsätta använda emulatorn istället.

Slutligen tycker vi att användandet av bash/terminalen har varit väldigt hjälpsam i projektet när vi väl blev lite vana med att använda den. Det gjorde att vi lyckades skriva ett skript som startar servern istället för gör att göra det själv, men det har också underlättat att kopiera och flytta runt filer i projektmapparna samt använda git, och vi känner att vi är mycket bekvämare med terminalen nu.

### **API:s**

Inledningsvis var planen att implementera Googles API för att kunna använda mässkartan som en riktig inomhuskarta via Google Maps och därmed kunna se sin position i realtid. Detta gjorde att vi spenderade en stor del av tiden med att förstå deras API för att kunna implementera det i vårt projekt. Efter att ha förstått det någorlunda insåg vi dock att det dels var ett mycket komplext API samtidigt som det inte helt tillfredsställer våra behov av en klickbar bild i den bemärkelsen vi avsett. Därför valde vi att byta till ett API som bättre mötte våra behov, vilket blev ClickableAreasImage av Lukas Lechner. Detta API var mycket mer lättförståeligt och gick relativt smärtfritt att implementera. Det gjorde också mer eller mindre exakt det vi behövde för mässkartan.

För att bygga QR-läsaren använde vi ZXing ("Zebra crossing"), ett open source bildprocesseringsbibliotek för streckkoder. ZXing är framtaget och utvecklat av över 100 olika programmerare. Google använde det i sin streckkodsläsare vilket bidrar till att det finns mycket dokumentation och exempel på hur det kan användas. Utöver detta har vi även implementerat BottomNavigationView – som går att hitta i Android Developers egna dokumentation.

### **Developer documentation**

I vårt projekt har vi främst använt Android Developers egna utvecklingsdokumentation på [developer.android.com](https://developer.android.com). Det innehöll mycket information om det mesta vi behövde, men kunde stundtals sakna förklaringar eller exempel på implementation vilket hade underlättat när man skulle förstå vad vissa metoder gjorde. Detta gör till exempel Oracle väldigt bra i sin dokumentation. För informationen om ClickableAreasImage som vi implementerade, fanns dokumentationen på projektets GitHub-sida, <https://github.com/Lukle/ClickableAreasImages>, men var möjligtvis något fåordig men i övrigt bra. BottomNavigationView som också implementerade i vårt projekt fanns mycket dokumenterat på Material Design, <https://material.io/develop/android/components/bottom-navigation-view/>, och var till stort hjälp för att förstå hur den fungerade. Även för ZXing som användes för QR-läsaren fanns dokumentationen på projektets GitHub-sida, <https://github.com/zxing/zxing>. Dokumentationen var välutvecklad och till stor hjälp för utvecklingen av vår QR-läsare.

När vi stötte på problem som var mer unika för just vårt projekt så använde vi oss av olika forum samt vid ett tillfälle dokumentation från google maps platform. Den sistnämnda använde vi när vi försökte lista ut hur vi skulle lösa vår interaktiva karta över mässlokalen, vi tyckte dock att det var otydligt när och hur de olika "teknikerna" skulle användas. Även om det var google maps platform så framstod det som att dessa tillvägagångssätt skulle fungera även som mappning på en statisk bild men vi fick inte det att fungera så vi fick tänka om. Svaret hittade vi senare i en tråd på Stack Overflow, det var även där vi hittade många svar på de mer unika problemen vi stötte på.

### **Android platform**

Android som plattform tyckte vi generellt var intuitiv och enkel att förstå som helhet. Det fanns dock en hel ny begrepp, som till exempel hur Android-version korrelerar med vilka funktioner som finns inbyggda och att det då är viktigt att bygga en applikation med viss

bakåtkompabilitet. I övrigt tyckte vi om att arbeta i Android då det kändes smidigt med bland annat aktiviteter och hur uppdelningen av XML-filer och Java-klasser fungerade. Det gjorde att det blev en naturlig uppdelning av klasser, paket och XML-filer.

## **Json**

Json var något Johannes stött på under kortare perioder på sin arbetsplats och något han därför inte var helt ovan med. Vi upplevde att arbeta med Json som enkelt, intuitivt och mycket användbart. Vi gillade speciellt att det var så generellt, eftersom det går att läsa in och tolka Json-filer oavsett vilket språk som används. Vi hade inte heller några större problem med att hämta data från en databas och formatera om det till Json. Koden som krävdes för att producera en sida med Json var relativt enkel och enligt oss relativt lik koden för att hämta data från en databas vi lärde oss i TIG058. Denna del av projektet gick väldigt bra.

## **Java servlet**

Java servlet var ett helt nytt kapitel för oss och något vi hade stora svårigheter att förstå. Dels hade vi svårt att förstå själva konceptet server, även om vi förstår vad en server är på en generell, grundläggande nivå. Eftersom vi inte hade något att relatera till var det därför svårt att förstå helheten i hur en Java servlet fungerade och hur detta sedan skulle resultera i en applikation på telefonen med informationen. Detta var därför något vi lade relativt stor vikt vid i projektets inledning. Vi skapade en mindre databas med en enkel servlet endast för att få en grundläggande förståelse för hur det faktiskt fungerade. Detta var en mycket stor lärdom för båda och något vi tror vi kommer ha stor nytta av framöver. Men det var också något som var en stor del av projektet, eftersom vi ägnade mycket tid åt detta.