

Reduktion av trafikskador med hjälp av Geofencing



Besnik Redzepi
Yacine Bessadi

Division of Industrial Electrical Engineering and Automation
Faculty of Engineering, Lund University

Reduktion av trafikskador med hjälp av Geofencing

Av:

Besnik Redzepi och Yacine Bessadi

EIEL05

2020/09/25

Sammanfattning

“Reduktion av trafikskador med hjälp av geofence” är ett projekt som syftar till att reducera antal skador som kan uppkomma inom busstrafiken och med detta minska kostnader. För att lösa uppgiften utvecklades en applikation, som skulle varna chaufförerna när de närmar sig ett riskabelt område såsom vid en skarp kurva eller en brant lutning. Tanken är att ett varningsmeddelande ska skärpa chaufförernas uppmärksamhet och därmed minska risken för att en eventuell skada uppstår.

Geofence har en avgörande roll i vår implementation. Den utformas som ett staket runt kring en specifik plats på kartan och varnar systemet när en auktoriserad enhet rammar staketet.

Chaufförerna, som representerar vår målgrupp, har en betydelsefull roll i vårt projekt och därför valde vi att inkludera deras åsikter och perspektiv i form av enkäter. Nobinas Skadekuben var också till stor hjälp i vår elicitering och tillsammans med enkäter, hade vi förutsättningar för att påbörja utvecklingen.

Arbetet med pappersprototypen fokuserades mest på att designa själva plattformen och integrera Geofence. Nästa steg var att skapa en datorprototyp utifrån pappersmodellen, och optimera den genom att lägga till ytterligare funktioner. Sista steget resulterade i en datorsimulering i Android Studio.

Slutresultatet ser lovande ut och utvecklingen har resulterat sig i en användbar teknisk lösning utformad som en applikation. Vi har lyckats att infria våra förväntningar genom att integrera geofence på bästa sätt efter företagets behov. Dock kunde vi inte värdera vårt projekt enbart utifrån datorsimulering. Denna begränsning har inneburit för oss att en eventuell estimering av fallet i antal skador och därmed av den ekonomiska vinsten inte är möjlig.

Nyckelord:

OBIS: är en Android platta som finns tillgänglig i förarplatserna. Skärmen är begränsad till vissa system som chauffören loggar in sig på innan avgång. Därefter fås tillgång till andra funktionaliteter till exempel översikt på körrutans hållplatser, se sin miljökörning samt få meddelanden av Trafikledningen.

OMS: står för Operations Management System och är Nobinas system för planering av trafik och resurssättning.

Skadekuben: En av företagets databaser med all information kring skador från alla Nobinas depåer.

Emulator: En inbyggd virtuell enhet i Android Studio som används för applikation simulering.

APIs: Står för “application programming interfaces”, utvecklad av Google med syfte på att möjliggöra tillgång till Googles erbjudna tjänster och kommunikation med Googles integrerade tjänster.

Abstract

“Reduction of traffic injuries by means of Geofence” is a project, which aims to reduce the number of injuries on Nobinas vehicles and thereby reduce costs. To solve the task, an application was developed with the purpose of warning the drivers on approaching a risky area such as a sharp turn or a steep slope. The idea is that warning notifications will sharpen the attention of the drivers and thereby reduce the risk of a possible injury. Geofence plays a central role in our implementation where a virtual fence is created in the shape of a circle around a specific spot on the map, and the system is warned when an authorized unit reaches the fence.

The drivers, who represent our target group, play an important role in our project and therefore we chose to include their points of view and perspectives in form of surveys. Nobinas Skadekuben (a database infrastructure) was very helpful in our elicitation and including the surveys, we were off to a good start of our development.

The work on the paper prototype focused mainly on the design of the platform and on integrating Geofence. Next step was to create a data prototype based on our paper model, and optimize it by adding further functionalities. The last step resulted in a data simulation in Android Studio.

The final result looks promising and we have created a useful, technical solution - our application. We have succeeded in meeting our expectations and integrated Geofence in the best possible way according to the needs of the company. However we were not able to validate our project solely on the base of the data simulation. This limitation has prevented us from doing an estimation of the reduction of injuries, and in turn the cost reduction.

Keywords:

OBIS: is an Android tablet which is available from the driver's seat. The screen is limited to certain systems in which the driver logs in before departure. Afterwards the driver has access to other functionalities such as an overview of stops on the route, monitoring of ECO driving and notifications from traffic management (Trafikledning)

OMS: Stands for Operations Management System and is Nobina's system for traffic planning and resource allocation.

Skadekuben: One of the company's databases containing all information regarding damages from all Nobina's depots.

Emulator: a built-in virtual unit of Android Studio used for application simulation.

APIs: Stands for "application programming interfaces", developed by Google to give access to Google's services and allow communication with other integrated Goggle services.

Innehållsförteckning

1. Inledning	8
1.1 Bakgrund.....	8
1.2 Syfte.....	9
1.3 Mål.....	9
1.4 Problemformulering.....	9
1.5 Motivering av examensarbetet.....	10
1.6 Metod.....	10
1.7 Avgränsningar.....	10
1.8 Resurser.....	11
1.9 Tidsplan.....	11
2. Teknisk bakgrund	12
2.1 Geofencing.....	12
2.2 Skadekuben.....	13
2.3 Azure.....	14
2.4 Proto.io.....	15
2.5 Android Studio.....	15
3. Metod	17
3.1 Projektmodell.....	17
3.2 Kommunikation.....	18
3.3 Målgrupp.....	18

3.3 Enkäter och intervjuer.....	18
3.4 Källkritik.....	19
4. Analys.....	21
4.1 Elicitering.....	21
4.2 Pappersprototyp.....	24
4.3 Datorprototyp.....	27
4.4 Utveckling.....	30
5. Resultat.....	36
6. Slutsats.....	39
6.1 Reflektion över etiska aspekter.....	42
6.2 Framtida utvecklingsmöjligheter.....	43
7. Terminologi.....	45
8. Källförteckning.....	46
9. Appendix.....	49

Förord

Detta examensarbete är det sista steg innan mållinjen på vår treåriga högskoleingenjörsutbildning i datateknik. Under våra tre år har vi uppnått en kunskap inom IT och datateknik som vi tidigare aldrig trodde fanns. Därför vill vi tacka Lunds Tekniska Högskola och Campus Helsingborg för dem tre fantastiska åren vi fått tillsammans.

Vi vill även tacka både vår handledare Christian Nyberg och vår examinator Mats Lilja, för den tiden de ägnat på att ge oss akademiska stöd och vägledning under arbetets gång. Ett stort tack till Nobina Lund, Nobina Sverige och alla chaufförer på Nobina Lund för att vi fick möjligheten att ta del av deras verksamhet och utföra detta projektet under denna svåra pandemi tid.

Tack till Maria Wahledow, Tobias Gabrielsson, Niklas Thorslund och Markus Stymne för att vi fick möjligheten till att vara en del av Nobinas projekt "En skadefri vardag". Tack för all den tid ni investerat på oss för att ge feedback, att stödja med kunskap och information samt alla möten över Microsoft Teams under projektets gång.

Vi önskar er en fortsatt framgångsrik digitaliseringen av Nobinas verksamhet.

Besnik Redzepi & Yacine Bessadi

1. Inledning

Detta kapitel av examensarbetet beskriver Nobina som företag och vilka problem de har samt ger en översikt av vårt examensarbete.

1.1 Bakgrund

Nobina grundades 1911 och kallades då för SJ buss. SJ buss slogs ihop med GDG (Trafikförvaltningen Göteborg-Dalarna-Gävle) år 1989-1990 till Swebus. Företaget är idag ett av nordens största operatörer för bussar i kollektivtrafiken. Nobina har i dag bland annat kontor i Malmö, Lund och huvudkontor i Solna, Stockholm, men finns även i Danmark, Norge och Finland. Företaget arbetar idag med en hel del tekniska system som används för att underlätta chaufförernas körupplevelse.

Ett av dessa är till exempel Azure (Microsofts molnplattform) där IoT gör beräkningar på bussens position och hur den förhåller sig till geografiska punkter såsom grindar och busshållplatser som Nobina har definierat med JSON. Systemet använder DataWarehouse för att hämta in data via Blobstorage och utföra alla de nödvändiga beräkningarna såsom bussens hastighet. Företagets admin får tillgång till datan via en tabulärmodell som kan administreras i Visual Studio. Koden samlas sedan på GIT. I PowerBI, som är kopplade mot tabulärmodeller, visas rapporter till företagets slutanvändare. Datawarehouse kör en utvecklingsmiljö, en testmiljö och slutligen en produktionsmiljö och använder sig av T-SQL samt Python och C#.

Förutom Azure använder företaget ett system som kallas OBIS för att varna chaufförerna för trafikomläggningar i form av varningsmeddelande. Chauffören har två skärmar lokaliserade vid förarplatsen. Den ena skärmen används för att logga in och för att sedan se vilka hållplatser som chauffören ska köra till. Den andra skärmen visar hur miljövänligt chauffören kör men även varningsmeddelanden som trafikledningen kan skicka ut till chaufförerna. Ett annat existerande system som kallas OMS används av företaget för att registrera och dokumentera bland annat skador och felrapporter.

Företaget har upplevt stora skador på sina bussar vilket har medfört omfattande reparationskostnader. Olika åtgärder har varit i bruk för att hantera problemet, eller åtminstone

minska det, men problemet kvarstår. Företaget har påbörjat ett nytt projekt som kallas “En skadefri vardag” för att lösa detta problem. Vårt examensarbete är en del av detta projekt där vi ska utveckla en teknisk lösning i form av en prototyp som ska ligga till grund för företagets framtida hantering av problemet.

Vår prototyp ska användas som ett informationssystem. Bussens GPS-position och geofencing ska användas för att kunna varna chaufförerna i förväg med hjälp av olika varningsmeddelanden. Beroende på situationen, triggas dessa varningsmeddelande när bussen närmar sig ett potentiellt område där det finns risk för skador.

1.2 Syfte

Syftet med examensarbetet är att utveckla en prototyp, som baseras på det nuvarande informationssystem för att minska såväl kostnader som risken för skador.

1.3 Mål

Nobinas nuvarande kommunikationssystem skickar ut varningsmeddelanden till chaufförerna på deras skärmar (OBIS), som finns lokaliserade vid förarplatsen.

Målet med vårt examensarbete är att minska skador, kostnader samt identifiera riskområden. Vi vill även försöka skapa en tryggare miljö för resenärerna genom att ändra chaufförernas beteende och körsätt.

Den tilltänkta prototypen ska simuleras genom att testa de mest väsentliga delarna av användargränssnittet.

1.4 Problemformulering

I detta examensarbete kommer följande frågor att besvaras utifrån vårt projekt och arbete.

1. Hur kan man använda geofencing för att åstadkomma vår/den önskade prototyp?
2. Kan vårt system integreras med Nobinas grundsystem?
3. Hur kan vår lösning ge den bästa effekten utan att distrahera chaufförernas körning?
4. Hur kan man på bästa möjliga sätt få en chaufför att undvika en skada?

5. Hur kan man identifiera vanligt förekommande skaderiskområden?

1.5 Motivering av examensarbetet

Med detta examensarbete vill vi skapa en teknisk lösning till ett vanligt förekommande problem för kollektivtrafiken och de kollektiva operatörerna och försöka skapa en tryggare och säkrare kollektivtrafik, fri av skador och onödiga kostnader. Vi vill även åstadkomma en djupare förståelse för geofencing.

1.6 Metod

Till en början kommer vi att utföra intervjuer och ge ut enkäter till chaufförerna för att ta reda på deras åsikter kring problemet samt samla in information. Den insamlade informationen analyseras för att få en överblick och sedan jämförs med företagets statistik på deras skador från Skadekuben.

Efter granskning och analys av den insamlade informationen skapas en pappersprototyp, där vi skissar hur systemet är tänkt att fungera och se ut.

Utifrån pappersprototypen kommer en datorprototyp att framställas och den ska avspegla hur grundsystemet är tänkt att fungera.

Validering av systemet kommer ske genom att testa prototypen i några av företagets bussar som körs dagligen.

1.7 Avgränsningar

Under examensarbetet kommer det inte att ingå någon implementation till företagets grundsystem, då företaget sa att det beräknas ta mer tid att utföra än vad examensarbetet tillåter.

Vårt examensarbete kommer inte kunna samla in en helhetsbild av alla bussar som finns i Lund utan kommer testas på några få bussar. Detta innebär att informationen vi får in baseras på de bussar vi testat vår prototyp på.

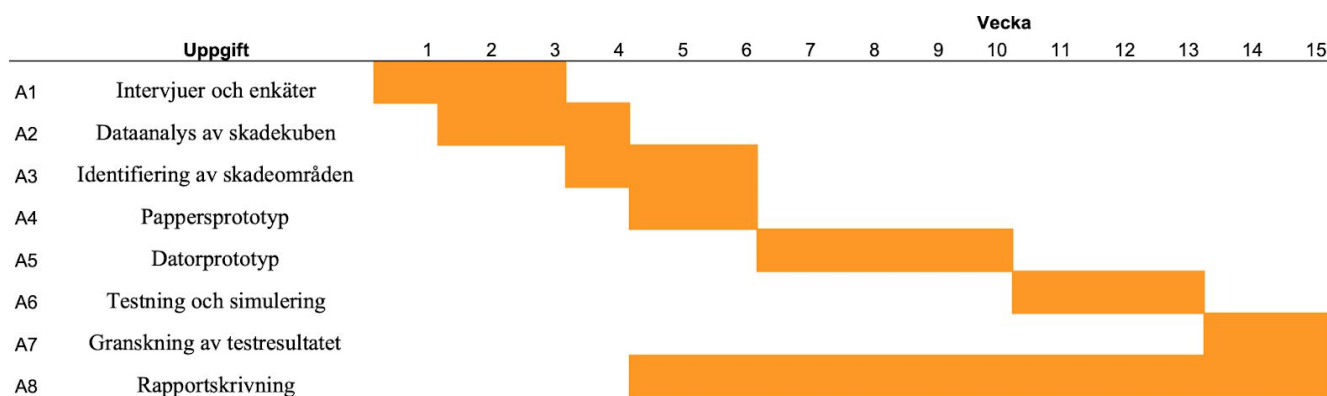
1.8 Resurser

För att kunna utföra examensarbetet behöver vi tillgång till en del resurser såsom företagets citrix receiver client för att kunna komma åt olika delar av deras system. Vi behöver dessutom inloggningar till citrixklienten, där olika behörigheter tilldelas, för att få tillgång till känslig information såsom statistik på skador.

Tillgång till bussarna samt en lokal arbetsplats är också nödvändigt för examensarbetet. Förutom detta behöver vi möjlighet att kunna intervjua anställda såsom chaufförer, driftpersonal samt andra ledande anställda på företaget till vår förundersökning.

Slutligen krävs även en välfungerande kommunikation mellan oss, vår handledare, och examinator på LTH är en avgörande faktor för att vi ska kunna lyckas med projektet.

1.9 Tidsplan



Figur 1. Tidsplan.

- **Intervjuer och enkäter:** Skapa och utföra enkäter, intervjua chaufförer/personal.
- **Dataanalys av Skadekuben:** Analysera företagets skadestatistik med hjälp av deras Skadekub där massa information om skador finns.
- **Identifiering av skadeområden:** Utifrån företagets statistik, chaufförernas åsikter samt fältarbete kan skadeområden identifieras.
- **Pappersprototyp:** Skapar en pappersprototyp utifrån insamlad data.
- **Datorprototyp:** Skapar en datorprototyp som är baserad på pappersprototypen.
- **Testning och simulering:** Simulera och testa datorprototypen.
- **Granskning av testresultaten:** Resultatet från testerna ska granskas och jämföras med tidigare siffror.
- **Rapportskrivning:** Dokumentering av arbetet.

2. Teknisk bakgrund

I detta kapitel behandlar vi alla tekniska system såsom hårdvara och mjukvara som använts under examensarbetets gång.

2.1 Geofence

Geofence är utvecklad av Google och kräver Google services API. Tjänsten är kostnadsfri och kan användas på både datorer, mobiler och surfplattor. En geofence kan definieras som ett virtuell staket, där specifika villkor gäller inom ett begränsat område. Området representeras med en cirkel, som dras kring en definierad plats på kartan (Figur 2). Cirkeln definieras av egenskaperna latitud, longitud, radie, period och transition types.

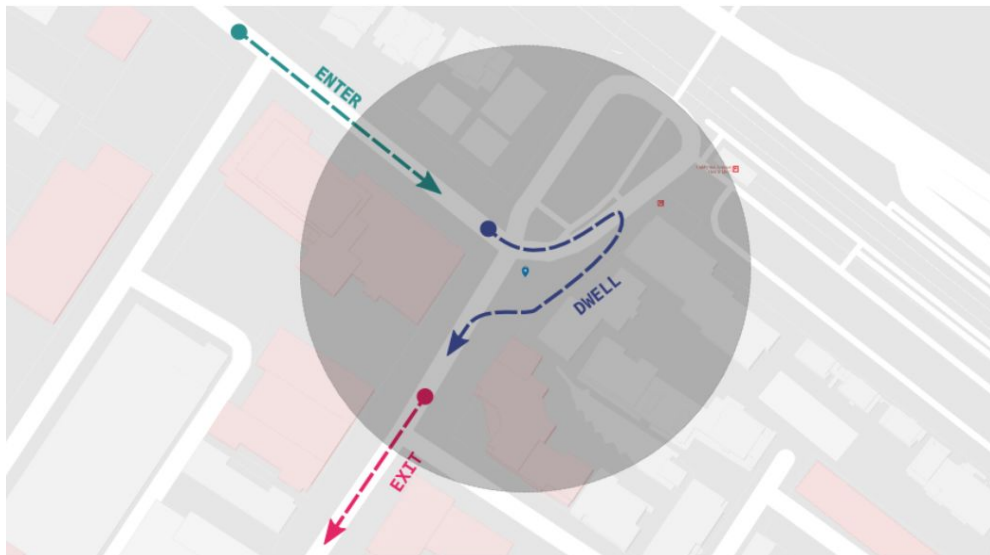
Det är administratören eller utvecklaren som står för konfigurering av geofence och ger auktorisering till en enhet. Det finns tre transition types; Enter, Dwell och Exit.

Beroende på hur geofencing är konfigurerad, kommer det virtuella staketet att utlösa ett svar i form av notifikationer, varningar eller textmeddelanden, när en auktoriserad enhet går in i området (Enter), närmar sig det kritiska punkten (Dwell) eller går ut ur området (Exit).

Tekniken kan användas i olika sammanhang och man kan ha flera aktiva Geofences samtidigt, dock med en gräns på 100 geofences per app, per enhet. För varje skapad geofence kan man be Location Service om att få meddelande på "Enter"- och "Exit"-händelser (Figur 2). Dessutom kan man begränsa tidsfördröjningen på geofence eller lägga till en delay innan dessa händelser triggas. Detta innebär att meddelandet triggas istället när den satta delay-tiden går ut istället för att triggas direkt när GPS-positionen går in i ett "Enter" eller "Exit" tillstånd.

Geofencing används dessutom idag som ett sätt att marknadsföra sin verksamhet. Företag och butiker placerar ut olika geofences där ens butik befinner sig på en karta eller i konkurrenters butiker för att exempelvis ta fram erbjudanden eller nyheter men även för att konkurrera mot andra verksamheter. Man kan dessutom få upp meddelanden på telefonen om man besökt en butik där man får möjligheten till att bedöma butiken genom att skriva en recension samt berömma med 1-5 stjärnor. Denna recension sparas sedan i Googles databaser. När man sedan söker på denna butiken får man upp tidigare bedömningar på butiken samt det kunderna skrivit om butiken eller verksamheten [13]. Ett exempel på geofence marketing är hur Pizza Hut

använde geofencing för att effektivisera sin verksamhet i USA. Restaurangen placerade ut geofence staket på 340 Pizza Hut butiker runt om i USA. Alla geofence staket sträckte sig en halv mil ifrån restaurangen. När en kund körde in i ett markerat geofence-område en halv mil bort fick denne ett sms skickat till sin telefon med olika typer av erbjudanden. Detta effektiviserade restaurangens verksamhet med 2.6 gånger mer än vanliga online ads [14].



Figur 2. Geofence med tre olika tillstånden en enhet kan befinna sig i.

2.2 Skadekuben

Skadekuben är en av flera SQL-databaser som finns tillgängliga i företagets grundsystem.

All information som finns i Skadekuben kommer från ett system som kallas OMS.

OMS används bland annat för att mata in data angående skador och felrapporter. Denna typ av information sparas då i en av flera databaser som till exempel Skadekuben.

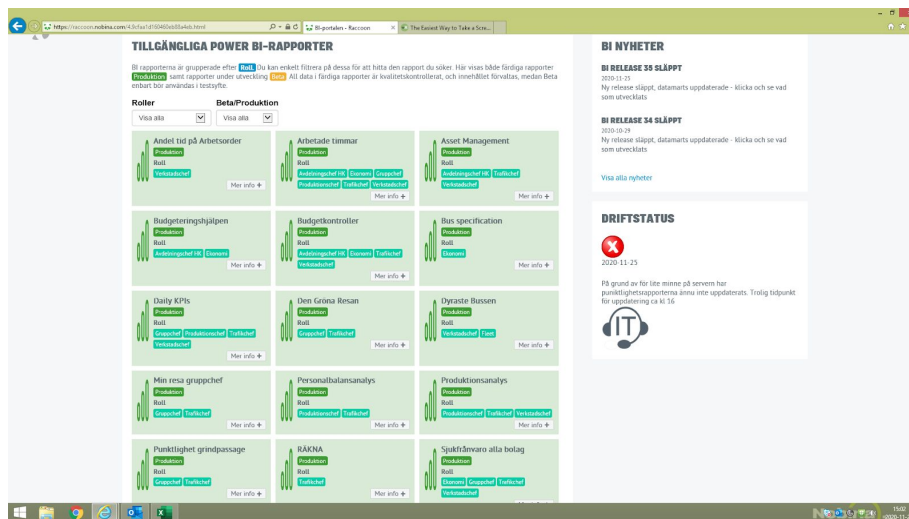
Skadekuben innehåller all data som finns angående skador på alla Nobina företag i Sverige och denna data kan enkelt tas fram i tabeller på Excel. Genom att ansluta sig till företagets SQL-server och SQL-databas kan man då komma åt olika databaser som till exempel Skadekuben. I Skadekuben kan relevant information tas fram kring skador och man kan filtrera detta på olika sätt som till exempel bussnummer, busslinje, vilken typ av buss det är, när och var

det inträffade, kostnader och så vidare. På detta sätt har vi sökt och tagit fram antal skador från år 2015 till 2020 för trafikområdet Lund.

2.3 Azure

Azure är en av Microsofts molnplattformar som erbjuder olika typer av tjänster till ens webbapplikation. Azure erbjuder tre typer av tjänster, Windows Azure, SQL Azure och Azure AppFabric. Windows Azure är ett molnbaserat operativsystem där man låter Microsofts datacenter hosta och köra ens applikationer. Operativsystemet fungerar bland annat som ett körsystem till ens applikation och erbjuder en mängd olika tjänster såsom möjlighet till att utveckla i vilket språk man vill, olika typer av lagring såsom blob storage, file storage och disk storage. Operativsystemet erbjuder dessutom styrning och hostning av sitt program på distans samt mycket mer [4]. SQL Azure fungerar som en molnbaserad version av Microsoft SQL Server och hanterar databaser fast i molnet. SQL Azure tillsammans med Power BI kan användas för att ta fram och förvandla bearbetningar till analyser, rapporter och diagram i realtid [5]. Azure AppFabric är en tjänst som i sin tur erbjuder tjänster som stödjer ens applikation både i och utanför molnet.

Azure används i vårt examensarbete för att ta fram statistik utifrån data som lagrats i Nobinas SQL-Azure databaser. Denna information tas utifrån Power BI som är kopplad till olika SQL databaser såsom Skadekuben vilket genererar diagram med information om till exempel skadekostnader och skador se figur 3.



Figur 3. Nobinas Power Bi portal med olika val av rapporter och statistik.

2.4 Proto.io

Proto.io är en plattform för att utveckla prototyper av applikationer där man enkelt kan bygga upp prototyper av mobilappar utan att behöva koda en enda rad. Systemet är skapat av företaget Proto.io Inc och används idag av företag som bland annat Amazon och Paypal. Proto.io är enkel att använda och programmet erbjuder en ”drag and drop” användarinterface. Man kan enkelt lägga till olika typer av knappar till färdiga templates. Proto.io erbjuder även färdiga interaktioner som enkelt låter dig lägga funktionalitet till knappar och sidor [7].

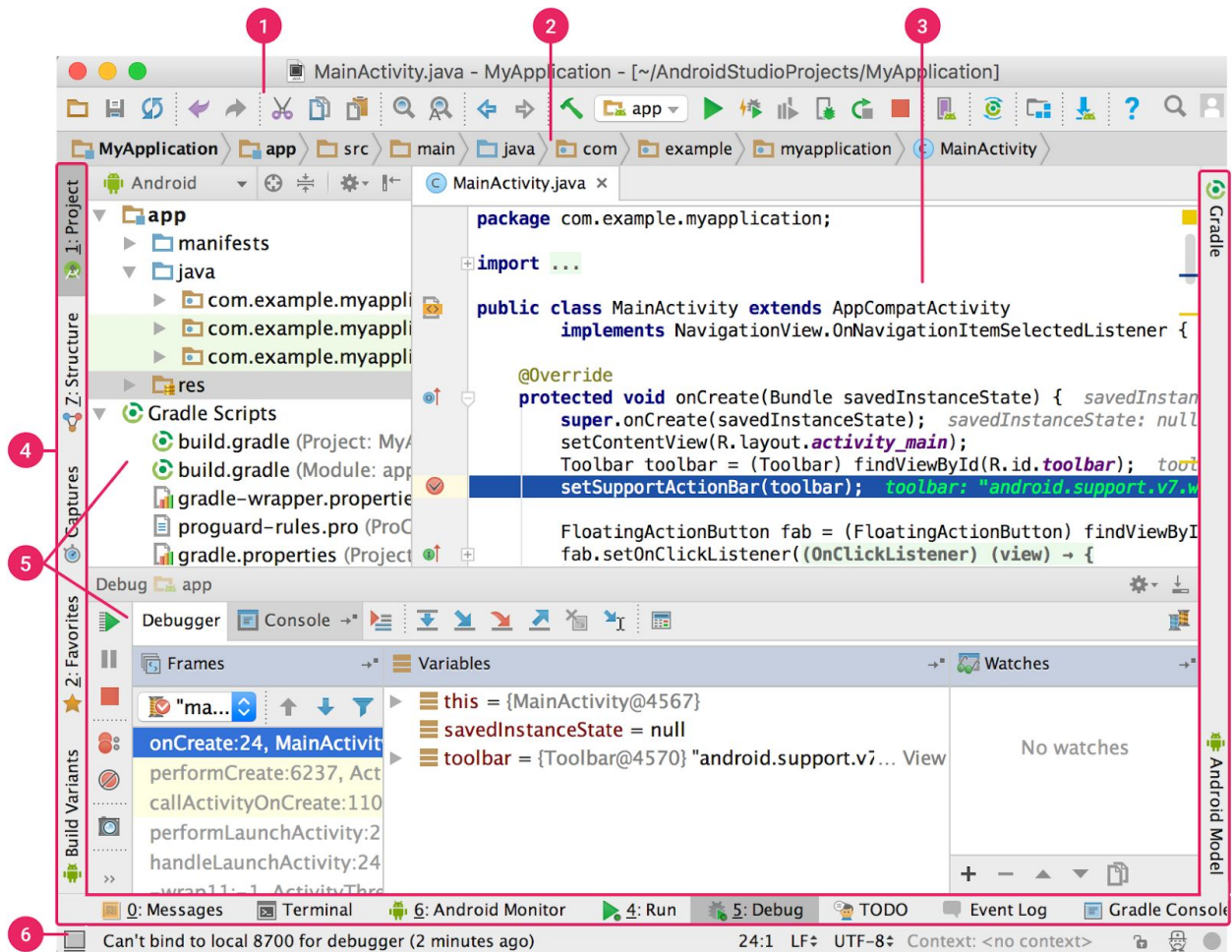
I vårt examensarbete har vi använt proto.io för att skapa vår datorprototyp utifrån vår pappersprototyp. Vi har med hjälp av proto.io:s rika funktionalitet kunnat ta fram en datorprototyp som hade en professionell design men även funktionell med alla typer av funktioner som det tilltänkta systemet är tänkt att ha.

2.5 Android Studio

Android Studio är ett av många system som Google utvecklat under åren. Android Studio är en integrerad utvecklingsmiljö med syfte på att utveckla applikationer till Android som operativsystem. Systemet finns tillgänglig för Windows, Mac OSX och Linux och de rekommenderade programspråken är Java och Kotlin men det finns även stöd för utveckling i C++. Android Studio erbjuder en mängd olika tjänster och funktioner för att underlätta apputveckling. En av funktionerna är Androids emulator som gör det lätt att bland annat simulera och testa sin applikation [6].

Utvecklingsmiljön erbjuder dessutom en användarvänlig vy där man enkelt kan få åtkomst till bland annat debugger, kör funktion i olika Android enheter samt diverse funktioner och extensions se figur 4.

Android Studio används i vårt examensarbete för att implementera det tilltänkta geofence-systemet i form av en applikation som i sin tur är tänkt att kunna simuleras med Androids emulator. Emulatorn ska kunna simulera en GPS-position som rör sig på google maps.



Figur 4. Utvecklingsmiljö vy med olika delar markerade. (1. Verktögsfält, 2. Navigationsfält, 3. Redigeringsfönster, 4. Sidoverktygsfält, 5. Debugger och source filer, 6. Statusfält).

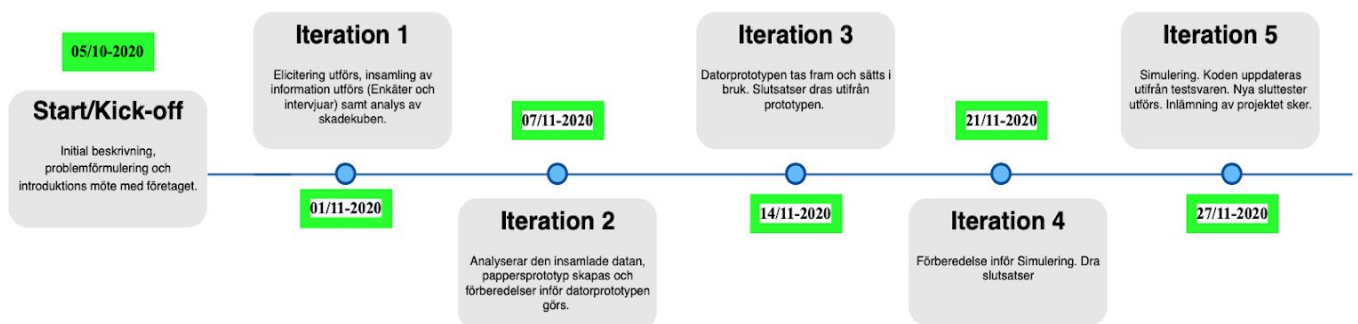
3. Metod

I detta kapitel berättar vi om vilka metoder som använts under examensarbetets gång samt vilka faser projektet genomgått under de fem iterationerna.

3.1 Projektmodell

I och med att vi saknade kännedom och kunskap om chaufförernas beteenden och åsikter, valde vi att följa en iterativ process. Den iterativa processen med prototyping består av följande steg: Design, build, test och learn.

Projektprocessen delades in i fem iterationer. I första iterationen skrev vi en initial beskrivning och hade ett introduktionsmöte med företaget. I den andra iterationen fick vi tillgång till nödvändig information genom enkäter, intervjuer samt fick tillgång till företagets Skadekub. Den tredje iterationen resulterade i en pappersprototyp, och den fjärde- och femte iterationen resulterade respektive i en datorprototyp och simulering (se figur 5). Datorprototypen utformades med hjälp av datorverktyget *Proto.io* och simuleringen med hjälp av *Android Studio*. Efter varje iteration sammanfattar vi resultaten i learn-steget där lärdomarna tages med till nästa iteration. Fördelen med att använda en iterativ process är att det finns utrymme för kontinuerlig förändring. En översikt över våra iterationer visas i figur 5.



Figur 5. Tidslinje över iterationerna i projektet.

3.2 Kommunikation

Vårt samarbete föregick mestadels online eftersom vi vistas på var sin sida av Öresund, och världen har stängts ner på grund av Covid-19 pandemin. Vi har alltså kommunicerat, både skriftligt och muntligt, med varandra via Messenger där vi också använde skärmdelning.

Under projektets gång var vi i kontakt med Nobina och utväxlade ideer och tankegångar. Vi hade samma vision redan från början, vilket underlättade vårt arbete. Besnik har jobbat på Nobina vilket innebar att de hade förtroende för honom och därmed kunde han få inloggning till Nobinas databas, alltså Skadekuben. Besnik befann sig redan på arbetsplatsen, därför hade han den personliga kontakten till de anställda på Nobina.

Introduktionsmötet med företagets ledare föregick online via Teams plattformen- oss och vår handledare i Lund och ledarna i Stockholm. Resten av kommunikationen med Nobinas ledare försiggick också online. Detsamma gäller vårt sätt att kommunicera med vår handledare på LTH via Zoom plattformen.

3.3 Målgrupp

Vår målgrupp är företagets chaufförer och detta var redan bestämt från examensarbetets början. Brist på tid gör att vi inte kan inkludera resenären. Anledningen till att chaufförerna är vår målgrupp är för att det är de som kör bussarna och är inblandade i bussarnas skador och företagets höga skadekostnader. Tanken är att använda enkäter och intervjuer som en metod för att samla in så mycket information som möjligt utifrån chaufförerna. Vi vill dessutom lyssna på chaufförernas åsikter för att se problemet från chaufförernas perspektiv och om det är möjligt, inkludera detta i vår tekniska lösning.

3.3 Enkäter och intervjuer

Vi har utfört en enkätundersökning med 50 enkäter, varav 44 var kompletta. Vi kunde inte göra mer än 1 intervju därför att chaufförerna inte var särskilt intresserade av att spendera deras rast på oss. Därför beslutade vi att bortse från intervjuerna och istället fokusera på enkäterna.

Syftet med enkäterna och intervjuerna är bland annat att fråga chaufförerna, om det nuvarande systemet är hjälpsamt eller om det är distraherande. Tanken är att om det nuvarande systemet är

användbart och fungerar som det ska, då kan vi vidareutveckla det.

3.4 Källkritik

Källkritik och källors trovärdighet är en viktig del av ett projekt. Denna del av projektet tar upp dem källorna som använts under examensarbetet samt deras trovärdighet utifrån olika aspekter.

Källorna som tagits fram och informationen som har använts i vårt projekt är förstahandskällor från företags hemsidor såsom Google, Android och Proto.io. Våra källor är dessutom framtagna utifrån trovärdiga sökmotorer som google scholar samt lubsearch som är Lunds Universitets egna sökmotor för akademiska artiklar och skrifter.

Det finns några källor som avviker från detta och dessa presenteras nedan.

[13] Brousell, L. (2013, August 28). *Five things you need to know about geofencing*, CIO Magazine [2020-12-01], Google Scholar

Denna källa är tagen från CIO Magazine och hittades när vi sökte i Google Scholar på geofencing i syfte att utforska temat. Källan går att lita på då skribenten är en Senior inom IT samt finns med i akademiska artiklar på Google Scholar.

[14] Boitnott, J. (2015, September 15). *Key takeaways from 5 of the most successful mobile marketing integrations*. Inc. Magazine. [2020-12-01] Google Scholar

Denna källa är tagen från Inc. Magazine och är en artikel som tar upp olika mobila sätt att marknadsföra. I artikeln är geofencing en av de fem alternativen till dagens mobila marknadsföring. Källan går att lita på då artikeln är framtagen ur Google Scholars akademiska artiklar och texter.

[15] *Personuppgiftslag (1998:204)* [2020-12-02]

Källan som handlar om personuppgiftslagen är en trovärdig källa då informationen är direkt hämtad från riksdagens hemsida angående Sveriges lagar.

[16] *Offentlighets- och sekretesslag (2009:400)* [2020-12-02]

Källan som handlar om Offentlighets- och sekretesslagen är en trovärdig källa då informationen är direkt hämtad från riksdagens hemsida angående Sveriges lagar.

4. Analys

I detta kapitel tar vi upp de olika typerna av analyser som vi utfört under examensarbetet samt presenterar vår enkätundersökning. Analyskapitlet tar dessutom upp vår pappersprototyp, datorprototyp samt implementeringen av dessa i Android Studio.

4.1 Elicitering

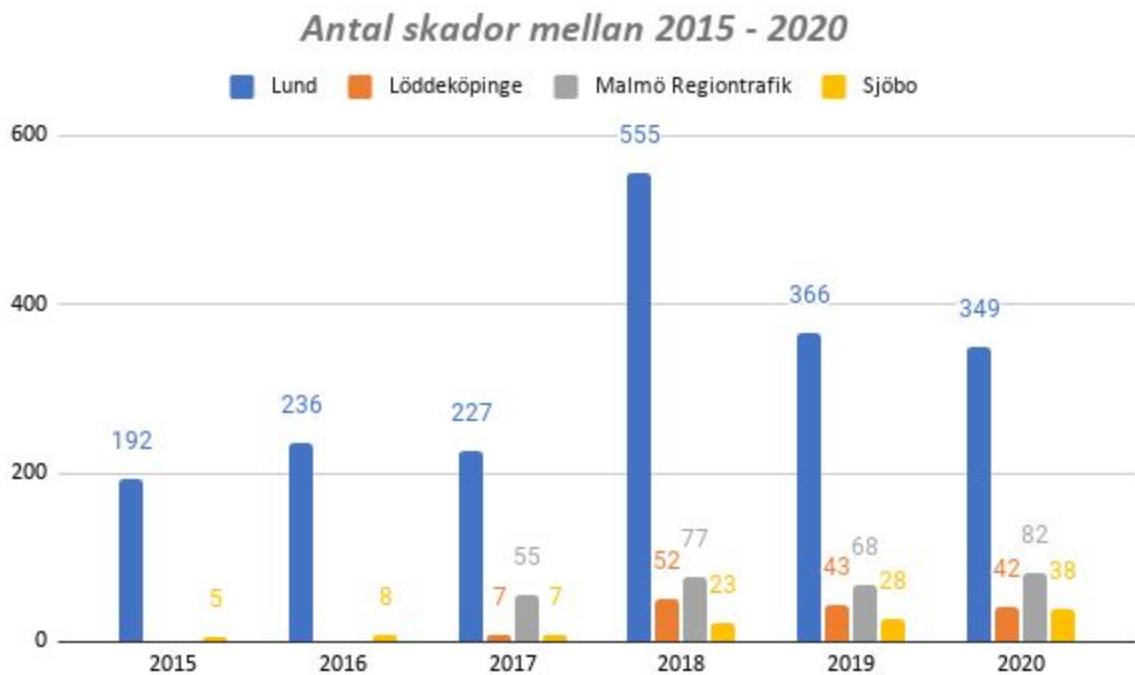
Nobina har upplevt många skador och därmed för höga skadekostnader. Enligt lagrad information från Nobinas Skadekub ökade skadekostnader från 1524521 kr i 2017 till hela 3219776 kr i 2018 (Figur 6). Det är mer än en fördubbling i skadekostnader, vilket är oerhört omfattande. Även om skadekostnader minskade gradvist under 2019 och 2020 med respektive 2668701 kr och 1985518 kr (Figur 6), kvarstår dessa tal som en stor belastning på Nobinas räkenskap.



Figur 6. Skadekostnader mellan 2017-2020.

Under perioden 2015-2020 har det skett en tydlig ökning i antal skador i trafikområdet Lund (Figur 7), särskilt på busslinjerna inom Lund. De näst mest drabbade busslinjerna är Malmö Regiontrafiken.

I vårt projekt har vi valt att fokusera på Skåne närmare bestämt på Nobinas linjer inom Lund, Malmö, Löddeköpinge och Sjöbo. Nobina Lunds sammanlagda antal skador under perioden 2015-2020 är 2460 skador (figur 7).



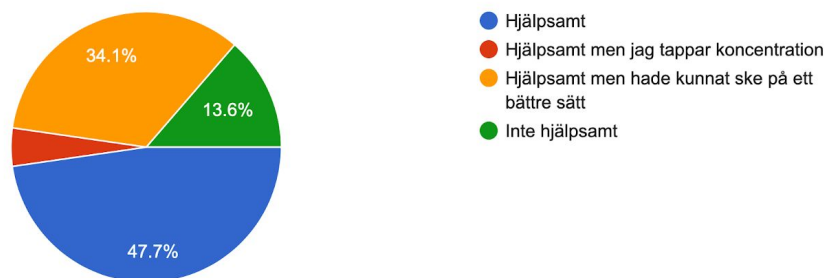
Figur 7. Antal skador i trafikområdet Lund mellan 2015-2020.

Resultatet av enkäterna har lett till att 47,7% av chaufförerna tycker att det är till hjälp när de från OBS:en får meddelanden på att en trafikomläggning är skedd (figur 8), medan hela 83,7%, ser det som en fördel att kunna få varningsmeddelande, när de närmar sig ett skaderiskområde (figur 9). Det är faktiskt en stor skillnad, men för att få en ide om varför skadorna uppstår har vi beslutat att utföra enkäter och intervjuer och på så sätt samla in information om chaufförernas åsikter och synpunkter. Dessa upplysningar är värdefulla och fundamentala i vårt projekt för att kunna utreda problemet, och skapa vår tekniska lösning.

En förklaring kan vara att vissa chaufförer tycker att meddelanden oftast kommer försent.

Hur upplever du trafikledningens meddelanden till OBIS:en när en trafikomläggning sker?

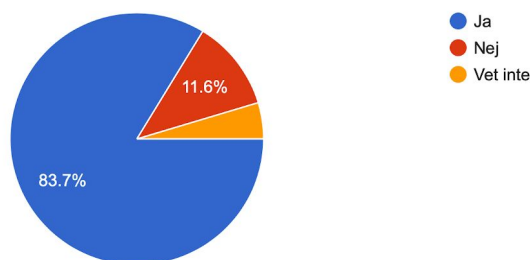
44 responses



Figur 8. Diagrammet i figuren visar chaufförernas synpunkt om OBIS:ens meddelanden.

Hade det hjälpt dig som chaufför om du fick upp ett meddelande som varnade dig när du närmar dig ett område där skador har skett tidigare?

43 responses



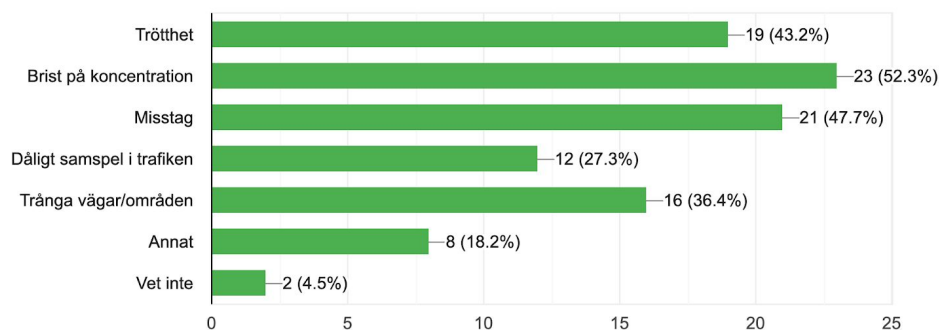
Figur 9. Diagrammet i figuren visar chaufförernas åsikt om varningsmeddelanden.

Den vanligaste orsaken till skador anges av chaufförerna att vara brist på koncentration. OBIS meddelande kan vara en distraktion och vara med till att minska chaufförernas koncentration, men 77,3% av chaufförerna tycker att de inte blir påverkade när de får dessa meddelanden (figur 11). Om chaufförerna faktiskt blir distraherade eller inte kan vi inte säga med säkerhet, men det är i alla fall inte deras uppfattning.

På andra och tredje plats kommer misstag respektive trötthet. Det är faktorer vi tyvärr inte kan påverka. På en fjärdeplats tillkommer trånga vägar/områden och här kan vi underlätta med hjälp av varningsmeddelanden (figur 10).

Varför tror du att en skada uppstår?

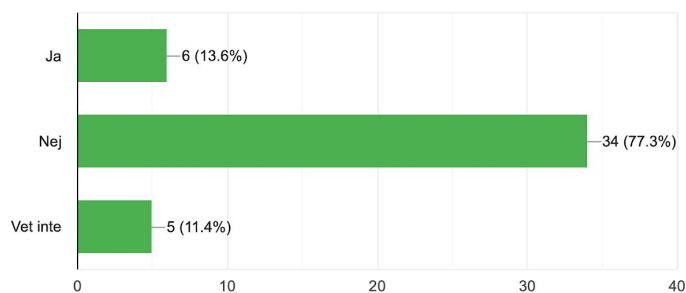
44 responses



Figur 10. Fördelning av de tillfrågade chaufförernas skadeorsak.

Blir du distraherad när du får ett meddelanden från trafikledningen på OBIS:en?

44 responses



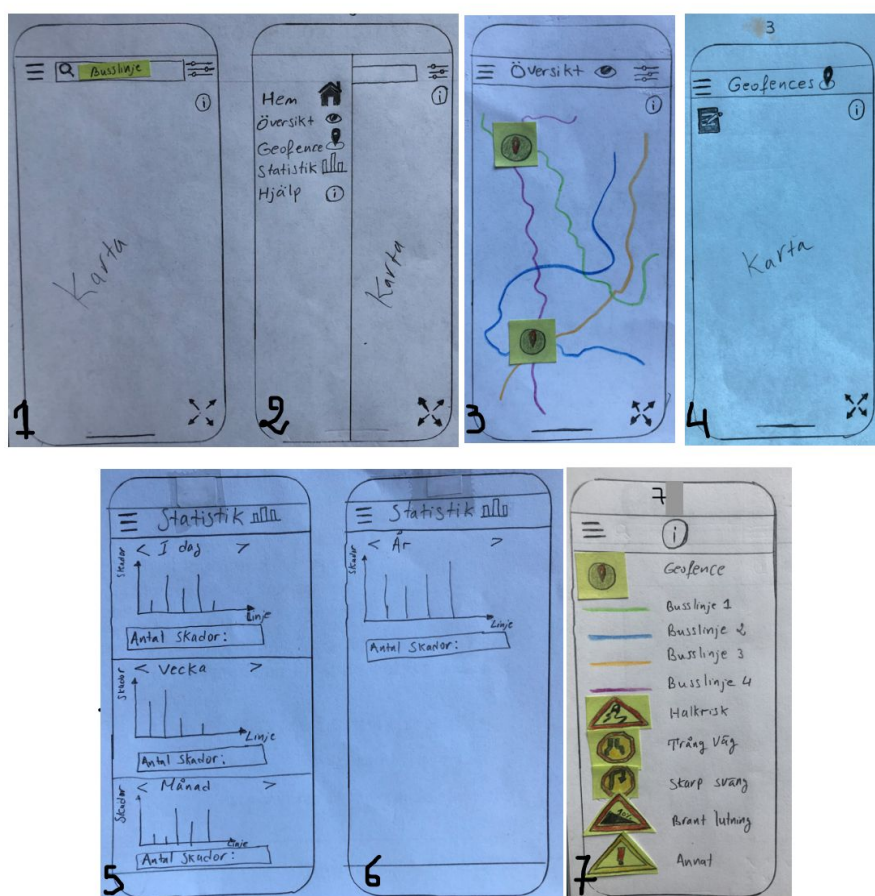
Figur 11. Fördelning av de tillfrågade chaufförernas åsikter om OBIS:en.

4.2 Pappersprototyp

Den tilltänkta prototypen skapades i den andra iterationen och består primärt av två olika prototyper, en prototyp riktad mot admin (figur 12 och figur 13) och en till OBIS:en (figur 14). Prototypen utgörs av flera pappersskärmar där knappar och användargränssnitt är ritade med blyerts. Post-It lappar används t.ex. till pop_ups och menyer, för att kunna klistras på och av

skärmarna.

Admins prototyp utformas som en app där hemskärmen består av en menu, filtrering, en ikon för information samt sökfiter, där man kan mata in numret på en busslinje (Figur 12. 1). Meny består i sin tur av översikt, geofences, statistik och hjälp (Figur 12. 2). Om man trycker på “Översikt” kan man se alla geofences och busslinjer. “Statistik” ger en översikt över skador per busslinje och via en rullmeny kan man växla mellan att få visat antal skador per dag, vecka, månad eller år (Figur. 12. 5 och 6). Ikonen (i) ger information om olika tecken och figurer som används i prototypen (Figur. 12. 7).

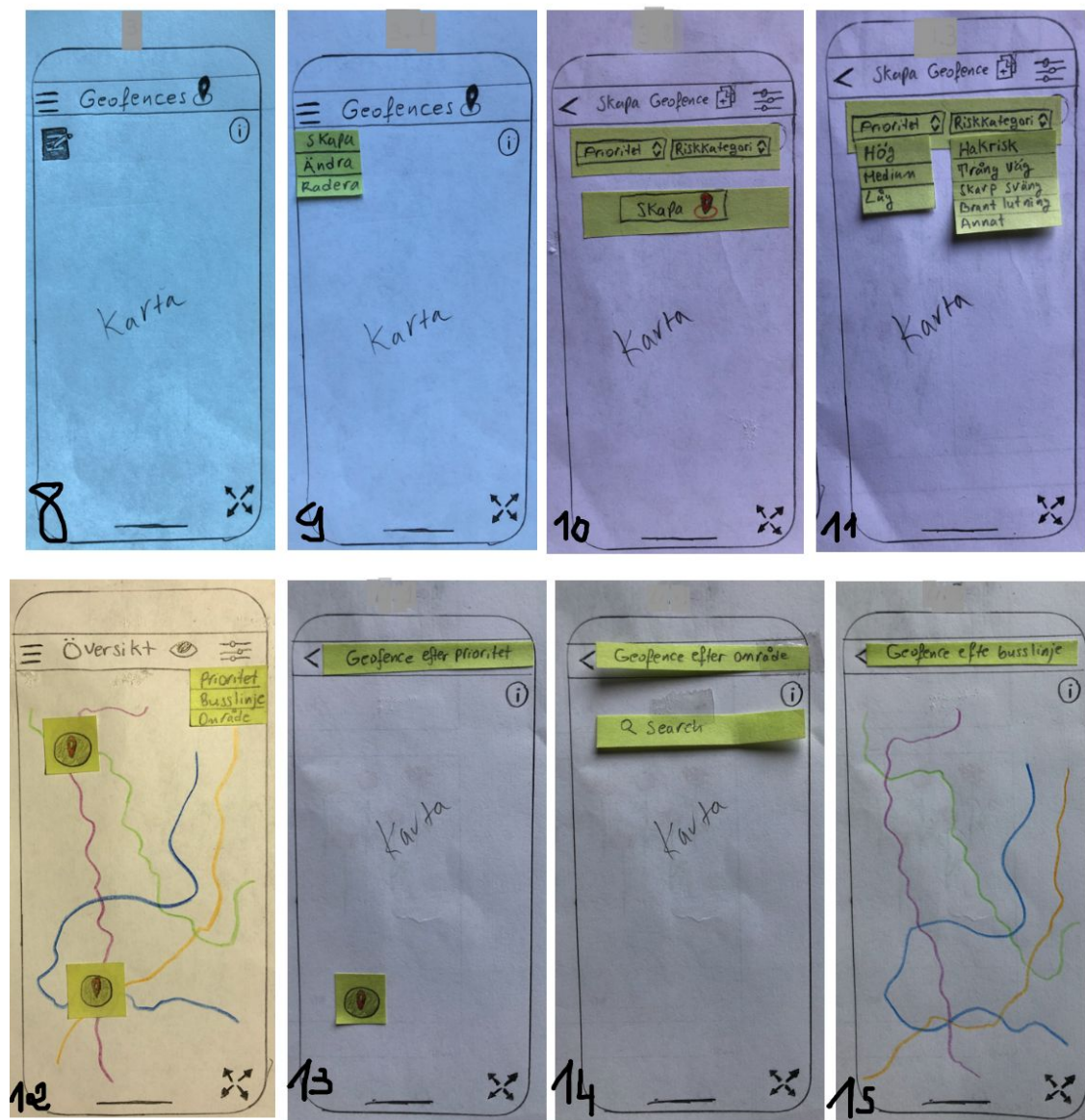


Figur 12. Admins pappersprototyp

Tryckning på “Geofences” under menyen leder till en ny sida där det finns en meny för att skapa, ändra eller radera en geofence (Figur. 13. 8, 9). “Skapa” leder till en ny sida (Figur. 13. 10) och där finns möjlighet för att specificera prioritet och riskkategori (Figur. 13. 11).

I sidan “Översikt” (Figur. 13. 12) har vi lagt in funktionalitet för att kunna filtrera sökningen

efter prioritet, område eller busslinje (Figur. 13. 13-15).



Figur 13. Pappersprototyp med undermenyer.

OBIS:ens prototyp, alltså pappersprototypen för hur meddelanden ser ut på förarskärmen i bussen, är baserad på nuvarande system och utgörs av 5 olika varningsskärmar (Figur. 14). Dessa varningsskärmar består av ett meddelande och en varningstriangel, som syftar till att varna chaufförerna när de närmar sig ett farligt riskområde, alltså en geofence. Med de 5 olika varningstriangelarna specificeras vilken sorts risk det finns inom aktuell geofence. Det är viktigt att det finns så kort text och så få knapptryckningar som möjligt, för att undvika att distrahera

chaufförens körning så att inte koncentrationen försämras. Därför är varningstriangelarna tagna från redan kända trafikskyltar som är snabbt avläsbara. Varningsmeddelandet kvarstår tills chauffören, genom att trycka på “kvittera”, har bekräftat att meddelandet har blivit läst (Figur 14. A-E).

När chauffören har passerat det farliga området, informeras denne om att faran är över, och vi berömmar chaufförens uppmärksamhet. Berömmet kan synas vara en obetydlig funktion, men kan ha stora effekter på längre sikt (Figur 14. F).

Dessutom ska OBIS:en utrustas med en funktionsbelysning, som med hjälp av inbyggda sensorer dämpar skärmkontrasten och ljusstyrkan, när det blir mörkt.



Figur 14. OBIS:ens pappersprototyp.

4.3 Datorprototyp

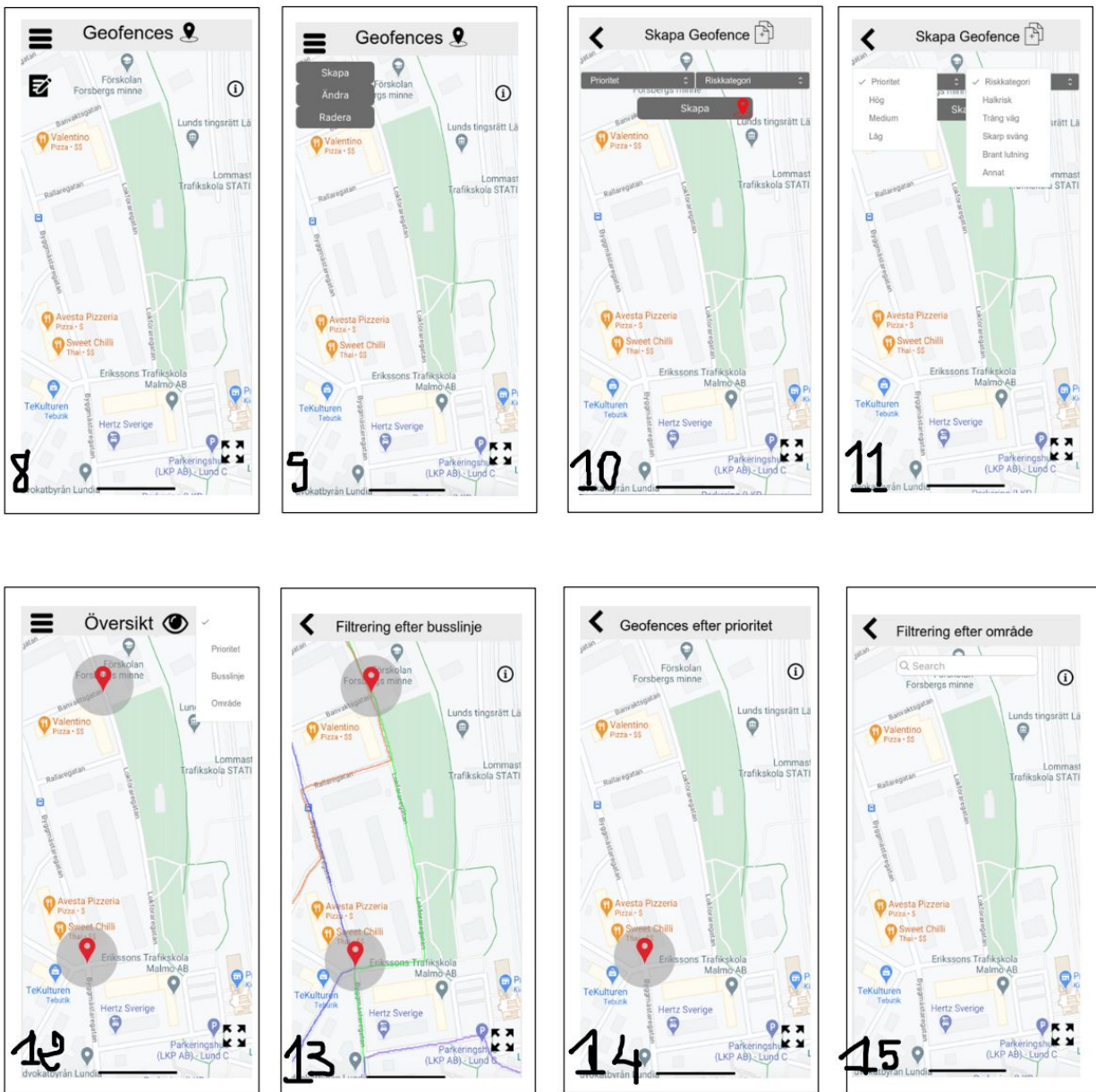
En datorprototyp utvecklas utifrån pappersprototyp under den tredje iterationen med hjälp av dataprogrammet Proto.io. Arbetsprocessen med prototypen gick efter principen dubbel-T metoden, där endast väsentliga funktioner har tagits med, medan resten av funktionaliteten ej gick att interagera med.

Datorprototypen har gett oss möjlighet att testa användbarheten av vår prototyp, genom att köra samtliga sidor och därmed visualisera funktionaliteten. Datorprototypen har dessutom fått en mer

användarvänlig design med olika ikoner som gör att man lätt kan navigera mellan sidorna och därmed optimera användbarheten (se Figur 15, 16 och 17). Detta sker genom att lägga till olika typer av interaktion vid till exempel knapptryck som då tar en från ena sidan till en annan beroende på vart man väljer att knapptrycket ska ta en.



Figur 15. Admins datorprototyp



Figur 16. Datorprototyp med undermenyer



Figur 17. OBIS:ens datorprototyp

Prototypen vi utvecklade i Proto.io har legat som utgångspunkt för hur vi tar oss vidare för att implementera geofence-systemet och hur vi därefter ska kunna simulera programmet.

4.4 Utveckling

Utvecklingsarbetet delades upp i två delar, prototyp- och simuleringsdelen. Tanken med simuleringen är att visa hur vår tekniska lösning är tänkt att fungera i verkligheten.

Handledarna på företaget berättade från början att en eventuell implementering inte kan vara realistisk på grund av tidsbrist och på grund av att behörigheter saknades, vilket har medverkat till att vi inte kunde implementera vår lösning på det sättet som vi hade tänkt oss. Därför var vi tvungna till att byta strategi och använda en datorsimulering istället.

För det första fick vi inte tillgång till företagets databaser och systemets interface med OBIS:en, i och med att detta krävde särskild behörighet. Detta medförde att vi inte visste hur front-end och back-end såg ut eller hur de kommunicerade med varandra.

För det andra hade vi inte tillgång till bussens position, och denna del är ett väsentligt verktyg i vår implementering.

Android Studios plattformen underlättar att simulera program i flera typer av virtuella enheter

som till exempel en vanlig androidtelefon till androidplattor. På detta sätt kan man enkelt se till, att programmet man utvecklat och simulerar, fungerar på olika typer av enheter som har olika storlekar. Emulatorn har dessutom ett flertal olika funktioner. Man kan till exempel rotera skärmen 90 grader till höger eller vänster. Man kan också höja och sänka ljudet, zooma in och ut i skärmen samt t.o.m. ta skärmbilder. Detta underlättar utvecklarens arbete och ger simuleringen en känsla att man utför det på en riktig enhet.

GPS

Datorsimuleringen [10] [11] utvecklades i Android studio med hjälp av Androids emulator [3]. Emulatorn tillåter användaren att skapa en GPS-position på en specifik plats med hjälp av de angivna koordinaterna (longitud och latitud). Här har vi valt att hårdkoda vår position till Malmö C (se programkod 1) där bland annat busslinje 133 och 172 kör.

Androids Studios emulator erbjuder att man kan se sin egen position när man trycker på “my position” i google maps. Funktionen använder sig av GoogleMaps platsdata och kräver tillåtelsen från användaren enligt programkod 1. Första gången man startar applikationen, dyker en notifikation upp där man ombeds ge tillåtelse. Framtida användning av applikationen kräver inte tillåtelsen från samma användare.

När man startar applikation efter att ha gett tillåtelse sätter den igång ett `ActivateLocation` och ställer in den på att hämta enhetens position enligt programkod 1. Då flyttas kameran till enhetens nuvarande position och syns i figur 18:s högra hörn på simuleringsenheten.

Därefter kan man få denna position att röra sig från en förutbestämd plats till en annan. Detta ska då representera vår buss position och kommer sedan närma sig ett område markerat med en eller flera geofences.

Geofence

Beroende på situationen och ändamålet, kan man antingen skapa ett enda objekt av en geofence eller en lista av geofences. Ett objekt av en geofence består av en unik id, cirkelns radie, transitions types samt platsens koordinater i form av latitud och longitud. Uppsättningen kräver tillgång till APIs location genom att skapa en instans av “Geofencing client” (se programkod 1). Dessutom krävs det också att applikationen får tillgång till nödvändiga behörigheter; *ACCESS_FINE_LOCATION* och *ACCESS_BACKGROUND_LOCATION* (se programkod 2).

För att påbörja övervakning av geofence/geofences, skickas en begäran till “Geofencing client” tillsammans med PendingIntent som ger instruktioner till API:et om hur händelser på geofence ska levereras till ens applikation (se programkod 1). Mer om avsnittet Intent och PendingIntent finns det under “Notifikationer”

```
private void handleMapLongClick(LatLng latLng){
    mMap.clear();
    BuildGeofence(latLng, GEOFENCE_RADIUS);
    addGeofence(latLng, GEOFENCE_RADIUS);
}

private void addGeofence(LatLng latLng, float radius) {
    Geofence geofence = geofenceAssistant.getGeofence(GEOFENCE_ID, latLng, radius, Geofence.GEOFENCE_TRANSITION_ENTER
|Geofence.GEOFENCE_TRANSITION_EXIT); //om transition type = 1 kommer man in i geofencen, om 4 lämnar man geofencen
    GeofencingRequest geofencingRequest = geofenceAssistant.getGeofencingRequest(geofence);
    PendingIntent pendingIntent = geofenceAssistant.getPendingIntent();

    if (ActivityCompat.checkSelfPermission( context: this, Manifest.permission.ACCESS_FINE_LOCATION) != PackageManager.PERMISSION_GRANTED) {
        return;
    }
    geofencingClient.addGeofences(geofencingRequest, pendingIntent)
        .addOnSuccessListener(new OnSuccessListener<Void>() {
            @Override
            public void onSuccess(Void aVoid) {
                Log.d(TAG, msg: "onSuccess: Geofence added...");
            }
        })
        .addOnFailureListener(new OnFailureListener() {
            @Override
            public void onFailure(@NonNull Exception e) {
                String errorMessage = geofenceAssistant.getErrorString(e);
                Log.d(TAG, msg: "onFailure: " + errorMessage);
            }
        });
}

private void BuildGeofence(LatLng latLng, float radius){
    MarkerOptions markerOptions = new MarkerOptions().position(latLng);
    mMap.addMarker(markerOptions);
    CircleOptions circleOptions = new CircleOptions();
    circleOptions.center(latLng);
    circleOptions.radius(radius);
    circleOptions.strokeColor(Color.rgb( alpha: 255, red: 255, green: 0, blue: 0));
    circleOptions.fillColor(Color.rgb( alpha: 64, red: 255, green: 0, blue: 0));
    circleOptions.strokeWidth(4);
    mMap.addCircle(circleOptions);
}
}
```

Programkod 1. Skapa och lägg till en geofence.

```

@Override
public void onMapReady(GoogleMap googleMap) {
    mMap = googleMap;
    // Lägg till en markör i Malmö C och flytta kamera ditt
    LatLng MalmöC = new LatLng( v: 55.6050, v1: 13.0038);
    mMap.moveCamera(CameraUpdateFactory.newLatLngZoom(MalmöC, v: 16));
    ActivateLocation();
    mMap.setOnMapLongClickListener(this);
}

private void ActivateLocation() {
    if (ContextCompat.checkSelfPermission( context: this, Manifest.permission.ACCESS_FINE_LOCATION) ==
        PackageManager.PERMISSION_GRANTED) {...} else {
        //Ombeds ge tillåtelse
        if (ActivityCompat.shouldShowRequestPermissionRationale( activity: this, Manifest.permission
            .ACCESS_FINE_LOCATION)) {
            //Förklara varför tillåtelse krävs, och sedan fråga efter den
            ActivityCompat.requestPermissions( activity: this, new String[]{Manifest.permission
                .ACCESS_FINE_LOCATION}, FINE_LOCATION_ACCESS_REQUEST_CODE);
        } else {
            ActivityCompat.requestPermissions( activity: this, new String[]{Manifest.permission
                .ACCESS_FINE_LOCATION}, FINE_LOCATION_ACCESS_REQUEST_CODE);
        }
    }
}

@Override
public void onRequestPermissionsResult(int requestCode, @NonNull String[] permissions, @NonNull int[] grantResults) {
    if (requestCode == FINE_LOCATION_ACCESS_REQUEST_CODE) {
        if (grantResults.length > 0 && grantResults[0] == PackageManager.PERMISSION_GRANTED) {
            //Vi har tillåtelse
            if (ActivityCompat.checkSelfPermission( context: this, Manifest.permission.ACCESS_FINE_LOCATION) !=
                PackageManager.PERMISSION_GRANTED && ActivityCompat.checkSelfPermission( context: this,
                    Manifest.permission.ACCESS_COARSE_LOCATION) != PackageManager.PERMISSION_GRANTED) {
                return;
            }
            mMap.setMyLocationEnabled(true);
        } else {
            //tillåtelse saknas
        }
    }
    if (requestCode == BACKGROUND_LOCATION_ACCESS_REQUEST_CODE) {

```

Programkod 2. GPS; visa användarens platsdata på begäran.

Utöver dessa händelser, tar applikationen emot en lista över aktiverade geofences och där kan man specificera vilka åtgärder som ska vidtas. Vår simulering består av ett enda objekt av geofence, och i uppsättningen av vår geofence har vi valt att endast trigga den vid in- och utkörning (se programkod 3).

```

List<Geofence> geofenceList= geofencingEvent.getTriggeringGeofences();

for(Geofence geofence: geofenceList){
    Log.d(TAG, msg: "onReceive: " + geofence.getRequestId());
}
// Location location = geofencingEvent.getTriggeringLocation();
int transitionType = geofencingEvent.getGeofenceTransition();

if(transitionType == Geofence.GEOFENCE_TRANSITION_ENTER){
    Toast.makeText(context, text: "Du närmar ett farligt riskområde!", Toast.LENGTH_SHORT).show();
    notificationAssistant.sendHighPriorityNotification("Du närmar ett farligt riskområde!", "", MainActivity.class);
}
else if(transitionType == Geofence.GEOFENCE_TRANSITION_EXIT){
    Toast.makeText(context, text: "Det farliga området är passerat!", Toast.LENGTH_SHORT).show();
    notificationAssistant.sendHighPriorityNotification("Det farliga riskområdet är passerat!", "", MainActivity.class);
}
}

```

Programkod 3. Skapa och trigga geofences notifikationer.

Ett geofence skapas vid ett knapptryck, med en varighet av minst två sekunder, på en specifik plats på kartan. För att åstadkomma denna funktionalitet krävs det att MainActivity förses med en lyssnare genom att implementerar OnMapLongClickListener och överrida metoden onMapLongClick.

OnMapLongClick måste vara anropade från main tråden och anropar handleMapLongClick, som i sin tur består av två privata metoder; buildGeofence och addGeofence, för att applicera geofencet på kartan se programkod 4.

```

@Override
public void onMapLongClick(LatLng latLng) {
    if(Build.VERSION.SDK_INT >=29){
        //WE NEED BACKGROUND PERMISSION
        if(ContextCompat.checkSelfPermission( context: this, Manifest.permission.ACCESS_BACKGROUND_LOCATION) ==
            PackageManager.PERMISSION_GRANTED){
            handleMapLongClick(latLng);
        }else{
            if(ActivityCompat.shouldShowRequestPermissionRationale( activity: this, Manifest.permission
                .ACCESS_BACKGROUND_LOCATION)){
                ActivityCompat.requestPermissions( activity: this, new String[] {Manifest.permission
                    .ACCESS_BACKGROUND_LOCATION}, BACKGROUND_LOCATION_ACCESS_REQUEST_CODE);
            } else{
                ActivityCompat.requestPermissions( activity: this, new String[] {Manifest.permission
                    .ACCESS_BACKGROUND_LOCATION}, BACKGROUND_LOCATION_ACCESS_REQUEST_CODE);
            }
        }
    }
    else{
        handleMapLongClick(latLng);
    }
}

private void handleMapLongClick(LatLng latLng){
    mMap.clear();
    BuildGeofence(latLng, GEOFENCE_RADIUS);
    addGeofence(latLng, GEOFENCE_RADIUS);
}

```

Programkod 4. Visa geofence vid tryckning på kartan.

Notifikationer

Notifikationer har med tiden blivit flitigt använt på nästan alla plattformar och applikationer. Det ger en överblick och håller användaren uppdaterad med relevant information utifrån olika informationskällor. I vårt system har vi använt oss av notifikationer för att varna chaufförerna när de närmar sig ett farligt riskområde (figur 20 och 22).

För enkelhets skull har vi i vår simulering valt att bortse från att trigga vår geofence, och därmed skicka in/få en notifikation, när vår spårbara enhet kommer in i den kritiska punkten; alltså “Dewell” (se programkod 3). Mer om detta finns i avsnittet Geofence under “Teknisk Bakgrund”.

Här kommer en snabb översikt över de viktigaste metoderna och funktionerna:

Notification-object: notifikation assistent representerar själva notifikationen, och bestämmer hur hur notifikationen ska bete sig.

Notifikation builder: är ett enkelt och praktiskt sätt för att konfigurera en notifikation. Builder pattern gör objektskapande enklare och förbättrar läsbarheten, genom att lätt kunna specificera och inkludera objektets egenskaper t.ex. fält, ordning osv.

Intent och PendingIntent: Intent är ett meddelande-objekt och används för att begära en åtgärd från en annan komponent. PendingIntent i sin tur är ett omslag-objekt kring Intent och syftar till att ge en främmande applikation tillåtelse till att använda sig av Intents innehåll även om processerna är olika.

Så PendingIntent fungerar som en framtida Intent, där användaren skickar till den aktivitet, vars notifikationer har blivit aktiverade.

5. Resultat

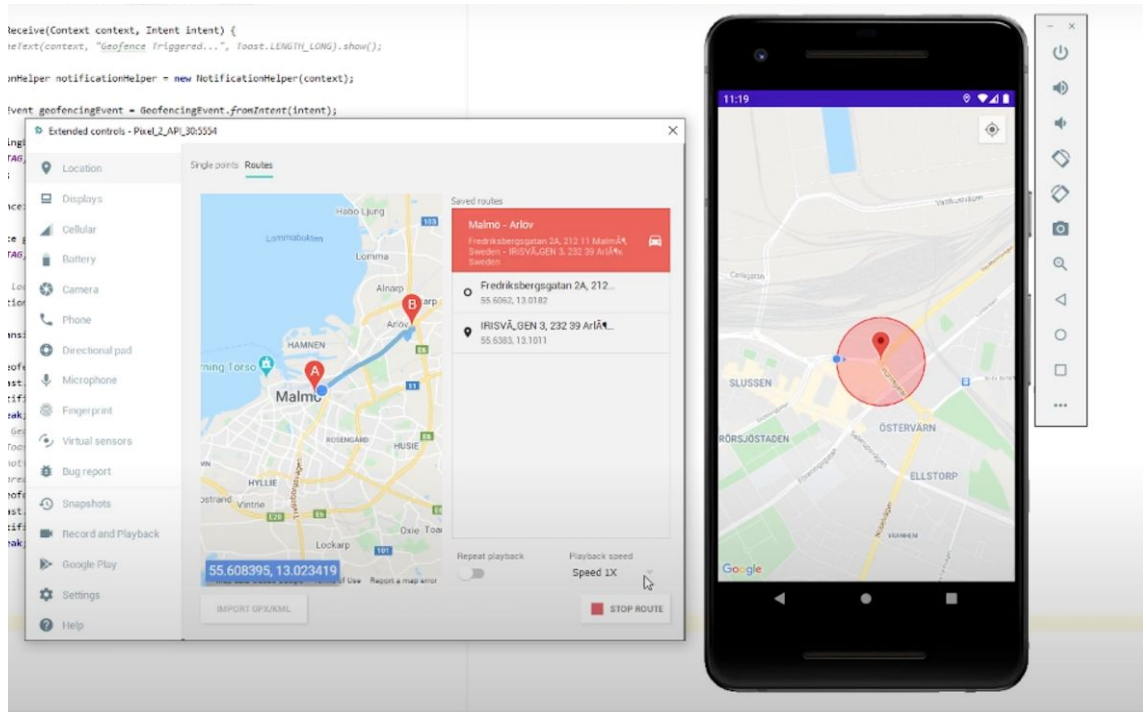
I detta kapitel presenterar vi resultatet som vår implementering av geofence-systemet har gett oss samt hur vi fått fram vårt resultat.

Implementationen av geofence-systemet i Android Studio gav oss ett tydligt resultat efter att ha programmet simulerats i Android Studios emulator vilket illustreras i figur 18.

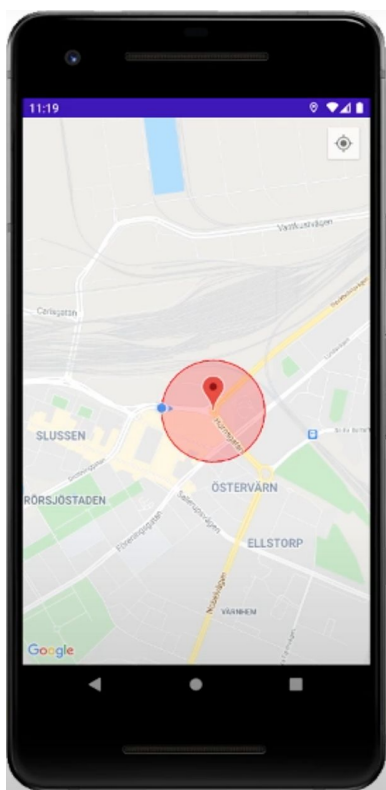
För att kunna simulera programmet öppnar man upp emulatorns extended controls se figur 18. Där kan man sedan välja routes som ger en möjlighet att välja en startpunkt och en slutpunkt för GPS-positionen.

I figuren har vi använt oss av Malmö central som startpunkt och Arlöv som slutpunkt då några av Nobinas busslinjer som vi tidigare nämnt som till exempel linje 172 kör just här. Vi väljer att placera ut ett geofence-staket vid en korsning på kartan där GPS-positionen kommer att passera [2]. När GPS-positionen närmar sig geofence-staketet (se Figur. 19), triggas det första varningsmeddelandet som varnar chaufförerna för ett farligt område där tidigare skador har skett (figur. 20). När GPS-positionen lämnar det markerade geofence staketet (Figur. 21), triggas ett ytterligare meddelande som bekräftar att det farliga området är passerat se figur 22. [3].

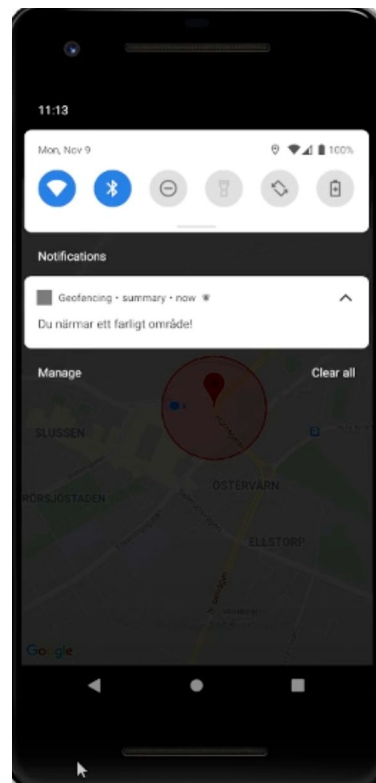
Varningsmeddelanden som triggas när bussens position närmar sig eller lämnar ett geofence kan man själv redigera till vilket meddelande som helst. Detta meddelande definieras när man skapar ett geofence.



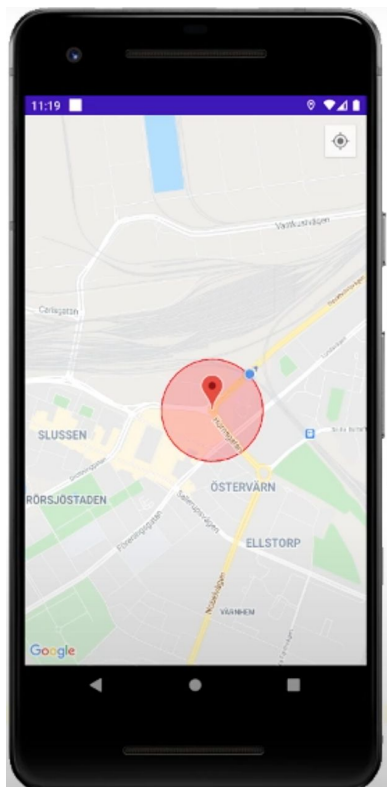
Figur 18. Emulatorns extended controls där simuleringen sker.



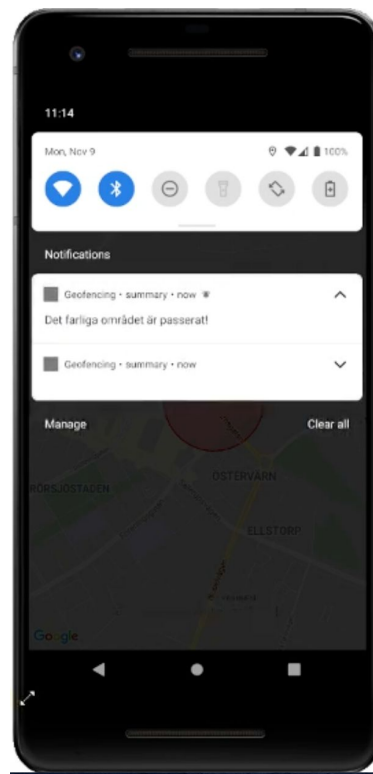
Figur 19. Enheten närmar sig geofence (Admins perspektiv).



Figur 20. Meddelande vid inkörning i riskområde (OBIS).



Figur 21. Enheten lämnar geofencet (Admins perspektiv).



Figur 22. Meddelande vid passering av riskområdet (OBIS).

6. Slutsats

Detta kapitel sammanfattar det viktigaste vi kommit fram till i resultatet av examensarbetet samt tar upp viktiga aspekter kring etik och framtidslösningar.

Sammanfattningsvis har projektet gett ett tydligt resultat. Geofencing är ett lovande system och en framtidslösning till företagets strävan att minska skador och skadekostnader. Vårt system hanterar de viktigaste delarna av geofencing tillsammans med en fungerande triggerlogik som skickar ut varningsmeddelanden. Systemet i sig är inte färdigutvecklat vilket beror på saknad åtkomst och behörighet till företagets databaser samt riktiga GPS-positioner. Detta har i sin tur hindrat oss från att skapa en komplett implementering. Den biten vårt projekt saknade var testning av vår implementering i en riktig testmiljö där bussarna får upp ett riktigt meddelande på skärmarna (OBIS). Utifrån detta hade man kunnat estimerat programmets riktiga potential samt antalet minskade skador per dag, vecka eller månad. Detta i sin tur hade gjort det möjligt att uppskatta företagets vinst på programmet beroende på hur bra systemet fungerade under testning.

Den lösning vi tagit fram är att försöka skapa en uppskattning av vad företagets vinster hade blivit om geofence systemet hade fungerat som tänkt. Det vi ser i figur 23 är en uppskattad vinst beroende på antal minskade skador i trafiken. Första stapeln visar hur mycket företaget hade vunnit på om vårt system hade minskat skadorna med en skada i månaden. Detta i sin tur ger 12 färre skador per år.

Från år 2015-2020 ligger företagets skademedelvärde på 403.8 skador per år och en medelkostnad på 1879703 kronor per år. Genom att beräkna minskningen av antalet skador per år dividerat med medelvärdet av antalet skador per år får vi fram hur många procent vi reducerar antalet skador med. Detta kan sedan användas för att ta fram hur mycket företaget vinner på vårt system om vi minskar skadorna med x antal skador. I figur 23 ser vi sex staplar som börjar på 12 skador (1 skada mindre per månad) hela vägen fram till 120 färre skador (10 skador mindre per månad). På detta sättet får vi en uppskattning på hur mycket företaget kan tänka sig minska sina stora utgifter med beroende på hur många skador vi kan minska i trafiken.



Figur 23. Uppskattad vinst utifrån minskade skador per år.

Det omfattande projektet vi utfört har dessutom gett oss svar på de frågeställningar vi ställt i kapitel 1. Våra slutsatser baserar sig på det arbete vi hunnit med under 15 veckor samt de möjligheter och behörigheter vi fått. I början av projektet formulerades det ett par frågeställningar som vi ville kunna svara på i slutet av projektet se Inledning 1.4.

Hur kan man använda geofencing för att åstadkomma den önskade prototypen?

För det första har vi kunnat ta fram en implementering av geofence-systemet som fungerar som tänkt fast utan företagets databas eller företagets busspositioner. Detta skapade begränsningar i hur långt vi under vårt examensarbete kunde ta projektet. Det vi åstadkommit har i alla fall lagt grunden för företagets framtida utveckling av geofencing och hantering av skador.

Kan vårt system integreras med Nobinas grundsystem?

Enligt överenskommelser har detta inte varit möjligt under en 15 veckors period utan det hade i så fall skett 1 år efter att det färdiga systemet tagits fram. Det vill säga att möjligheten finns där men i så fall längre fram i framtiden.

Hur kan vår lösning ge den bästa effekten utan att distrahera chaufförernas körning?

Att systemet inte får vara distraherande har varit ett viktigt krav under hela examensarbetets gång. Det har lyfts fram av chaufförer, ledningen samt av oss som en viktig fråga och aspekt. Vi har berört denna fråga i många delar av projektet och det finns en tydlig slutsats till detta. Chaufförer vill inte ha en ljus bakgrund på OBIS:en som kan blända eller få chauffören att tappa koncentrationen. Man vill inte heller göra många knapptryck på OBIS-skärmen utan systemet ska vara enkelt men effektivt för chaufförerna. Många framtidslösningar kring denna fråga finns att läsa om i 6.2 Framtida utvecklingsmöjligheter.

Hur kan man på bästa möjliga sätt få en chaufför att undvika en skada?

Utifrån vårt examensarbete och vår enkätundersökning handlar denna fråga om tre mänskliga faktorer: Trötthet, misstag och brist på koncentration. Trötthet kan endast undvikas av chaufförerna själva, och misstag är svårt att hantera. Däremot kan brist på koncentration påverkas genom ett system som geofencing. Genom att skicka ut varningsmeddelanden kan man påverka chaufförens beteende samt uppmärksamhet i trafiken. Detta stärks även av vår enkätundersökning då 83.7% av chaufförerna var positiva till ett system som geofencing. Genom att varna för olika skaderiskområden är det tänkt att man ska kunna reducera antalet skador genom att sträcka en hjälpsam hand till chaufförerna ute i trafiken.

Hur kan man identifiera vanligt förekommande skaderiskområden?

Identifiering av skaderiskområden är tänkt att ske med hjälp av geofence-systemets statistik, där man tydligt kan se hur många inrapporterade skador de finns för olika typer av områden och platser under en dag, vecka, månad eller år. På detta sätt kan man samla in information och använda denna information för att identifiera de vanligaste skaderiskplatserna. Detta kan även ske genom att ledningen tar ställning till och analyserar den information de besitter eller de nå ut till chaufförerna för att skapa en förståelse av var skaderisk platserna finns.

Sammanfattningsvis ligger vårt examensarbete till grund för en teknisk och digitaliserad utveckling av Nobinas dilemma med skador och höga skadekostnader. Vi har dessutom skapat goda förutsättningar till början på företagets stora projekt "Reduktion av trafikskador med hjälp av geofence". Det finns en hel del utvecklingsmöjligheter och tankar på hur man kan förbättra

eller utveckla vidare geofence-systemet. Men med tanke på den tidsram vi besitter har dessa istället blivit exempel på framtida utvecklingsmöjligheter.

6.1 Reflektion över etiska aspekter

När man talar om geofencing och hur programmet är uppbyggt talar man om tillgång till användarens position och information. Denna position kan vara kopplad till en personlig telefon eller arbetstelefon, olika typer av enheter eller objekt men dessutom fordon såsom bussar eller bilar. Man använder en användares information för att få programmet att fungera och vara användbart. Denna information är något som berör användaren och inte får läcka ut och med detta följer en hel del etiska aspekter.

För det första när man utvecklar ett sådant system som vi gjort under vårt examensarbete använder vi en GPS-position. Denna i sig är inte kopplad till någon individ utan används med hjälp av vår simulering. Men skulle denna position vara kopplad till ett flertal bussar eller ett flertal personers telefoner hanterar vi sekretessbelagd data som tillhör företaget samt personuppgifter om systemet är kopplade till en telefon med en fysisk användare. När vi talar om personuppgifter finns det en tydlig lag i Sverige som kallas Personuppgiftslagen som enligt lag skyddar människor mot att deras personliga uppgifter läcks eller missbrukas.

Personuppgiftslagen säger att “Syftet med denna lag är att skydda människor mot att deras personliga integritet kränks genom behandling av personuppgifter” [15]. GDPR kompletterar personuppgiftslagen med fler kriterier som till exempel att företag måste berätta och visa vad dem ska använda personuppgifterna till. Detta är viktigt då både vårt system men även nästan alla system idag använder människors personuppgifter på något sätt. Dessa måste skyddas och får inte användas i exempelvis i brottsligt syfte. Detta har vi till exempel gjort en lösning på genom att man först och främst frågar efter tillåtelse att använda en användares position vilket användaren då måste tillåta.

För det andra tangerar denna fråga även sekretess och konfidentiell information. Om till exempel företaget inte tillåter att deras information ska läcka ut, är det krav samt lag på tystnadsplikt. Detta innebär att om man tagit del av företagets sekretessbelagda information har man inte rätt till att dela med sig av denna till någon annan under några omständigheter. Sekretesslagen säger

att det är "Ett förbud att röja en uppgift, vare sig det sker muntligen, genom utlämnande av en allmän handling eller på något annat sätt" [16]. När man pratar om sekretess handlar detta just om konfidentiell information. Detta är något som man bör prata med företaget om för att veta vad som räknas som konfidentiell information där man måste bevara sekretess. Detta är viktigt då exempelvis konkurrenter kan ta del av denna information och använda det i konkurrerande syfte mot ett annat företag.

Alla dessa tre etiska aspekter faller under en och samma lag som många företag samt privatpersoner kommer i kontakt med. Denna lag kallas GDPR och står för The General Data Protection Regulation. GDPR är en lag som gäller i hela EU och efterliknar Sveriges Personuppgiftslag.

6.2 Framtida utvecklingsmöjligheter

Under examensarbetets gång och efter implementeringen av geofence-systemet har det kommit fram en hel del tankar kring vad man kan göra bättre eller lägga till om man hade haft tid till det. För det första kan man ändra i implementeringen så att storleken på geofence området går att variera. I dagsläget är radien på geofence cirkeln hårdkodad till 200 meter men detta ska kunna variera när man placerar ut geofencet genom att få upp en ruta där man väljer storleken på radien samt i samma ruta kunna definiera vad varningsmeddelandet ska vara.

För det andra hade man kunnat utnyttja geofence-systemet samt dess varningsmeddelanden på flera olika sätt. Exempelvis har det uppkommit en hel del skador på bussarna när chaufförerna backat. Det man hade kunnat göra är att varje gång bussen backar skicka ett varningsmeddelande till OBIS och varna för olika typer av hinder eller objekt som kan leda till en skada men även personskador. Detta har uppfattats som viktigt av både ledningen, chaufförerna samt en specifik chaufför som arbetat på Nobina i över 20 år. Han nämnde att detta skulle vara en bra lösning då bussarna 2552-2597 saknar backkameror. En annan viktig ide som chauffören tog upp var att varna chaufförer med ett varningsmeddelande när temperaturen understiger 0 grader celsius. Detta är ett väldigt viktigt varningsmeddelande då halka är en stor skaderisk för både bussen samt resenärerna och tas oftast som en självklarhet vilket kan skapa brist på koncentration och uppmärksamhet för halkrisken. Detta meddelande ska endast visas en gång och triggas när

chauffören startar bussen och temperaturen understiger 0 grader för att inte upprepa meddelandet allt för många gånger.

För det tredje hade man kunnat underlätta chaufförernas körupplevelse samt minska distraktionen genom att chauffören enkelt ska kunna trycka bort meddelanden som kommer till OBIS:en genom att trycka var som helst på skärmen istället för att specifikt trycka på en kvitteringsknapp. Skånetrafiken använder sig av olika typer av förprogrammerade röstinspelningar eller voice pattern som de också kallas, för att förmedla olika typer av budskap. Detta hade man också kunnat inkludera i vår implementering så att man minskar den distraherande faktorn genom att chauffören inte behöver lägga fokus på att läsa eller trycka på skärmen.

En fjärde punkt som också är essentiell i framtida lösningar är att man inte ska använda sig av röd färg när man skickar ut varningsmeddelanden för detta kan uppfattas som ”stop” eller rött stoppljus av chaufförerna. Man ska istället använda sig av gul färg för att presentera en varning och grön färg för att presentera att det är fritt fram eller att en fara är över.

Sist men inte minst finns det även möjlighet att utnyttja geofence-systemet för enkel marknadsföring för både Nobina och Skånetrafiken. Denna marknadsföring kan exempelvis vara olika typer av erbjudanden från Skånetrafiken till olika meddelanden där man till exempel vill tack resenärerna för att de väljer att resa med Nobina i olika områden som exempelvis Lund Central där många resenärer åker både tåg och buss.

7. Terminologi

I detta kapitel av examensarbetet förklarar vi vissa utvalda ord som vi tycker kräver en förklaring.

Trafikområde Lund: Omfattar fyra depåer som finns i Lund, Malmö, Sjöbo och Löddeköpinge depå.

Voice pattern: Är ett röstmeddelande som spelas upp som förmedlar ett budskap som till exempel när en röst i bussarna talar om att nästa hållplats är Lund Central.

Extended controls: Emulatorn i Android Studios olika funktioner och knappar finns under extended controls.

Power BI: Power BI är en del av Microsoft Azure och är en tjänst som används för visualisera data på webben eller i en app som sedan verksamheten kan ta del av.

8. Källförteckning

I detta kapitel finns alla källor som använts under examensarbete rangordnade med en referens siffra i börja så man enkelt kan hitta och söka på källan.

[1] *Provide contextual experiences when users enter or leave an area of interest* [2020-11-02]

<https://developers.google.com/location-context/geofencing>

[2] *Create and monitor geofences* [2020-11-03]

<https://developer.android.com/training/location/geofencing>

[3] *Run apps on the Android Emulator* [2020-11-24]

<https://developer.android.com/studio/run/emulator>

[4] *Upptäck fler alternativ med öppen källkod på Azure* [2020-11-25]

<https://azure.microsoft.com/sv-se/overview/choose-azure-opensource/>

[5] *Azure och Power BI* [2020-11-25]

<https://docs.microsoft.com/sv-se/power-bi/connect-data/service-azure-and-power-bi>

[6] *Meet Android Studio* [2020-11-25]

<https://developer.android.com/studio/intro>

[7] *Every feature you could ever need to build the ultimate prototype* [2020-11-26]

<https://proto.io/en/features/>

[8] *Dokument med Enkätfrågor:*

https://docs.google.com/forms/d/e/1FAIpQLSdU3Vv9gFo2eae3ujXeNsSHXw-8--8hIC0ezPIiDcSaoPq5dQ/viewform?usp=pp_url

[9] *Dokument med resultat från enkätfrågorna:*

https://drive.google.com/file/d/1vrdQnSGa3EQpZzF-PxFjB2ILHD_1AsYD/view?usp=sharing

[10] *Video på datorprototyp - Admin:*

https://www.youtube.com/watch?v=IPArQqJ_jng

[11] *Video på datorprototyp - OBISen:*

<https://www.youtube.com/watch?v=acvGFsoAP1M>

[12] *Video på simulering av geofence med Android Studio:*

<https://youtu.be/XcHdZm-STrQ>

[13] Brousell, L. (2013, August 28). *Five things you need to know about geofencing*, CIO Magazine [2020-12-01] Google Scholar

<http://www.cio.com/article/2383123/mobile/5-things-you-need-to-know-about-geofencing.html>

[14] Boitnott, J. (2015, September 15). *Key takeaways from 5 of the most successful mobile marketing integrations*. Inc. Magazine. [2020-12-01] Google Scholar

<http://www.inc.com/john-boitnott/key-takeaways-from-5-of-the-most-successful-mobile-marketing-integrations.html>

[15] *Personuppgiftslag (1998:204)* [2020-12-02]

https://www.riksdagen.se/sv/dokument-lagar/dokument/svensk-forfattningssamling/personuppgiftslag-1998204_sfs-1998-204

[16] *Offentlighets- och sekretesslag (2009:400)* [2020-12-02]

https://www.riksdagen.se/sv/dokument-lagar/dokument/svensk-forfattningssamling/offentlighets--och-sekretesslag-2009400_sfs-2009-400

9. Appendix

Dokument med Enkätfrågor:

https://docs.google.com/forms/d/e/1FAIpQLSdU3Vv9gFo2eae3ujXeNsSHXw-8--8hIC0ezPIiDcSaoPq5dQ/viewform?usp=pp_url

Dokument med resultat från enkätfrågorna:

https://drive.google.com/file/d/1vrdQnSGa3EQpZzF-PxFjB2ILHD_1AsYD/view?usp=sharing