



Bachelor's Thesis

Analys av användbarhet och arbetsflöde vid modernisering av applikation

Av

Richard Nilsson & Pontus Rohdén

Department of Electrical and Information Technology
Faculty of Engineering, LTH, Lund University
SE-221 00 Lund, Sweden

Sammanfattning

Detta examensarbete har utförts i samarbete med Knowit Core i Malmö för att utveckla nästa generation av ett system för markupplåtelser, som används på Malmös fastighets- och gatukontor. Målet med arbetet har varit att undersöka hur ny teknologi och kunskap kan användas för att förbättra användbarhet och arbetsflöde i ett förlegat system.

För att bygga en kravspecifikation över det nya systemet har intervjuer och fokusgrupper hållits med användare av det gamla systemet i Malmö och Lund. Intervjuer har även hållits med fastighetskontoret i Helsingborg för att få en bild av hur kommuner utan markupplåtelsemjukvara arbetar. Kravspecifikationen som skapades efter denna eliciteringsprocess användes som ett levande dokument under hela utvecklingsperioden.

Mjukvara för prototypframtagning undersöktes därefter för att se om det enkelt gick att ta fram en visuell presentation av ett nytt system. För att kunna testa de förbättringsalternativ som togs fram beslutades det att en MVP (Minimum Viable Product) skulle utvecklas. För att kunna skapa ett system med hög modularitet valdes React som programmeringsspråk. React är ett komponentfokuserat javascriptbibliotek utvecklat av Facebook. En backend-server för systemet byggdes i Node-baserade Express Js då det är ett minimalistiskt ramverk med stort stöd för plugins.

Examensarbetet resulterade i en MVP som demonstrerar funktionalitet som förbättrar arbetsflödet från det gamla systemet och en kravspecifikation som innehåller krav för implementerad funktionalitet och krav för funktionalitet för ett framtida system. Det har också sammanställts en lista över funktionalitet som intressenterna önskar finnas i ett framtida system, om ett sådant skulle komma att utvecklas.

Nyckelord

Användbarhet, Mjukvara, Elicitering, Markupplåtelser, MVP

Abstract

This thesis work has been done in collaboration with Knowit Core in Malmö to develop the next generation of a system that handles leases, used by the housing and street office of Malmö city. The purpose of this thesis has been to research how new technologies can be used to enhance the useability and workflow in an outdated system.

Interviews and focus groups have been held with users from Malmö and Lund to develop a requirements specification for the new system. To further gain an understanding of cities without a dedicated software for handling leases an interview was also held with the office in Helsingborg. The requirements specification that was developed after this elicitation process was used as a live document to be edited throughout the whole project.

Software was researched to find out if there was a simple way to produce a visual prototype of the new system. To test the improvements, it was decided to develop an MVP (Minimum Viable Product). A programming language that supported a high level of modularity was researched and the decision landed on React. React is a component based javascript library developed by Facebook. A backend server was developed with Node based Express js which is a minimalistic framework that supports a large library of plugins.

This thesis work has resulted in an MVP that demonstrates functionality that enhances the workflow in the outdated system as well as a requirements specification with requirements for the functionality that has been implemented, as well as requirements for future development. A list of functionalities which the stakeholders are asking for from a future system has also been compiled, in the case that such a system was to be developed.

Keywords

Useability, Software, Elicitation, Leases, MVP

Förord

Vi vill tack Magnus Pettersson för att han trott på oss och gav oss chansen att utföra detta arbete, och Johan Matsson för all hans hjälp genom hela projektet som handledare. Vi vill också tacka alla de handläggare i de olika kommunerna som har ställt upp på intervjuer.

Även ett stort tack till vår handledare på LTH, Christin Lindholm, och vår examinator Christian Nyberg.

Richard Nilsson & Pontus Rohdén

Innehåll

Sammanfattning.....	2
Nyckelord	2
Abstract	3
Keywords.....	3
Förord.....	5
1. Inledning.....	11
1.1. Bakgrund	11
1.2. Syfte.....	12
1.3. Målformulering	13
1.4. Problemformulering.....	13
1.5. Motivering av examensarbetet	13
1.6. Avgränsningar.....	14
2. Teknisk bakgrund	15
2.1. React.....	15
2.1.1. Material-UI	15
2.1.2. Google-Maps-API.....	15
2.1.3. React-Pdf-js.....	16
2.1.4. ExpressJs.....	16
2.1.5. Invision	16
2.1.6. Github.....	16
2.1.7. Lequinox	17
2.1.8. Rest-server.....	17
3. Metod/analys	19
3.1. Inläsning	20
3.2. Elicitering.....	20

3.2.1.	Intervjuer	21
3.2.2.	Prototypdemo/Fokusgrupp	24
3.3.	Intervjuer i andra kommuner	26
3.3.1.	Malmö	27
3.3.2.	Lund	27
3.3.3.	Helsingborg	28
3.3.4.	Sammanfattning	28
3.4.	Kravspecifikation	29
3.5.	Design	30
3.6.	Implementering	33
3.7.	Testning och Utvärdering	34
3.8.	Kommunikation	35
3.9.	Anpassningar under arbetets gång	35
3.10.	Källkritik	36
4.	Resultat	41
4.1.	Inläsning	41
4.2.	Elicitering	41
4.2.1.	Intervjuer	41
4.2.2.	Fokusgrupp	41
4.3.	Intervjuer i andra kommuner	43
4.4.	Kravspecifikation	43
4.5.	Design	44
4.6.	Implementering	45
4.7.	Testning och Utvärdering	46
5.	Slutsats	47
5.1.	Vad är kärnuppgifterna hos handläggarna?	47

5.2.	Hur ser arbetsflödet ut vid handläggning i det nuvarande systemet?	47
5.3.	Hur kan ett förbättrat arbetsflöde se ut?	48
5.4.	Hur skapas en användbar prototyp?	48
5.5.	Vilka teknologier ska användas vid utveckling av MVP:n?	49
5.6.	Hur ska känslig information hanteras och lagras?	49
5.7.	Hur ska MVP:n utvärderas och testas?	50
5.8.	Reflektion över etiska aspekter	51
5.9.	Framtida utvecklingsmöjligheter	52
5.9.1.	Upplåtelsehantering	52
5.9.2.	Bärbar version av systemet	52
5.9.3.	Kartfunktioner	53
5.9.4.	Fakturahantering	54
5.9.5.	Mailfunktioner	54
5.9.6.	Pdf-scanner	54
6.	Terminologi	57
	Appendix	61
	Intervjufrågor	61
	Flödesdiagram Markus	65
	Arbetsflöde vid handläggning	66
	Kravspecifikation - Markupplåtningssystem	69
	Terminologi	69
	Funktionella krav	70
	Uppgifter	70
	User-stories	71
	Övriga funktionella krav	73

Datakrav	73
User-stories	73
Övriga Datakrav	74

1. Inledning

Vid modernisering av gammal programvara genom nyutveckling är det viktigt att förstå vilka problem den gamla programvaran försökt att lösa för användarna, och vilka problem som fortfarande finns kvar eller har skapats.

1.1. Bakgrund

Knowit är ett konsultbolag som erbjuder digitala lösningar från tre affärsområden, Experience med inriktning mot frontend och användbarhet, Insight för managementkonsulter och Solutions för backend och systemutveckling. Knowit etablerades 1990 och har idag cirka 2400 medarbetare i Sverige, Norge, Danmark Finland och Tyskland. Knowit har på uppdrag av Malmö stad utvecklat ett system för fastighets- och gatukontoret som hanterar stadens markupplåtelser. Vid markupplåtelse ansöker till exempel en restaurang om att få bygga en uteservering, eller en förening om att spärra av delar av staden för att arrangera ett maraton. Denna upplåtelse hanteras av handläggare vid Malmö stad med hjälp av Knowits system, Markus, som i dagsläget är över tio år gammalt och börjar bli svårt att underhålla. Många av de buggar som finns går inte att spåra i koden utan de löses genom att starta om systemet (*Figur 1*), vilket tar mycket onödig tid som kunde lagts på utveckling av systemet. Hela det nuvarande systemet behöver förnyas med avseende på utseende, funktioner, underhållbarhet och användbarhet.

Logg		Antal: 18 st
Händelse	Datum	
Upplåtelse ändrad	2019-03-20	
Beslut ändrat	2019-03-20	▲
Upplåtelse ändrad	2019-03-20	
Notering skapad	2019-03-20	
Ärendet ändrat	2019-03-20	
Text till journal.category.leaseupdated saknas!	2019-03-20	
Text till journal.category.registration saknas!	2019-03-20	▼

Figur 1. Buggexempel, inkorrekt händelsetext visas

För att lösa ovan nämnda problem skall examensarbetet resultera i en delvis implementerad applikation, även kallad Minimum viable product (MVP) med ny design, en kravspecifikation som beskriver denna MVP, samt en vidareutvecklingsplan som beskriver hur utvecklingen skall fortsätta efter examensarbetet. Applikationen ska underlätta arbetet för handläggarna genom förbättrad användbarhet, vilket är det största behovet hos dem. Den ska även minska behovet av support från Knowit genom förbättrad underhållbarhet samt underlätta vidareutveckling genom modulär design. Designen baseras på intervjuer med handläggarna vid Malmö stad om hur deras nuvarande arbetsflöde ser ut, vad deras kärnuppgifter är samt om det finns någon funktionalitet som de saknar i det nuvarande systemet. Med kärnuppgifter menas vad handläggarna behöver få gjort i samband med markupplåtelse till exempel generera fakturor som lagras i en viss databas, osv. Arbetsflödet beskriver hur handläggarnas arbete utförs i dagsläget med hjälp av det nuvarande systemet.

1.2. Syfte

Syftet med examensarbetet är att undersöka om arbetsflödet vid markupplåtelse för Malmö stads handläggare kan effektiviseras och om användarupplevelsen kan förbättras jämfört med det nuvarande systemet.

1.3. Målformulering

Examensarbetet ska undersöka hur en applikation med bättre användbarhet kan designas och implementeras. Handläggarnas arbetsflöde ska undersökas för att se om det finns möjlighet för förbättringar. En MVP ska utvecklas som ska vara en grund för vidareutveckling av en ny applikation. MVP:n ska designas på så sätt att underhåll och vidareutveckling underlättas.

1.4. Problemformulering

1. Vad är kärnuppgifterna hos handläggarna?
2. Hur ser arbetsflödet ut vid handläggning i det nuvarande systemet?
3. Hur kan ett förbättrat arbetsflöde se ut?
4. Hur skapas en användbar prototyp?
5. Vilka teknologier ska användas vid utveckling av MVP:n?
6. Hur ska känslig information hanteras och lagras?
7. Hur ska MVP:n utvärderas och testas?

1.5. Motivering av examensarbetet

En stor anledning till att vi valde att söka detta examensarbete var att det gav oss en möjlighet att utveckla en större lösning på ett problem och att det ger oss en chans att testa vårt framtida arbetsliv. Vi var inte lika intresserade av att göra en enbart undersökande studie som vi var på att utveckla en användbar produkt/prototyp. Både Knowit och Malmö stad känns intressanta att arbeta med och vi tror att detta kommer vara ett givande examensarbete. För Malmö stad är Examensarbetet intressant då resultatet förhoppningsvis kommer underlätta det dagliga arbetet. Knowit kommer att gynnas då den nya applikationen kommer vara mer modulär vilket kommer underlätta vidareutveckling samt att mer moderna teknologier kommer att användas, vilket kommer underlätta support.

1.6. Avgränsningar

Arbetet kommer inte resultera i en fullständig produkt, utan kommer inriktas på att ta fram en MVP för handläggning av uteserveringar som demonstrerar funktionalitet och kan användas för att ta fram en slutprodukt. MVP:n kommer fokusera på uppvisbar funktionalitet och mindre fokus kommer läggas på uppbyggnad av bakomliggande arkitektur. MVP:n kommer inte att integreras mot ekonomisystem eller andra tredje parter.

2. Teknisk bakgrund

Detta avsnitt beskriver kort de tekniker och mjukvaror som använts under examensarbetet.

2.1. React

React är ett Javascriptbibliotek utvecklat av Facebook och privata individer. React används främst för att utveckla enkelsidiga hemsidor eller mobila applikationer. Dess huvudsakliga uppgift är att rendera data till sin virtuella DOM (Document Object Model) (Reactjs 2020).

Användandet av en virtuell DOM gör det möjligt för utvecklaren att vid ändringar endast ladda om de delar som ändrats, och därmed spara mycket tid. Reactutvecklare använder sig av något som kallas komponenter för att bygga delfunktioner till sina applikationer. Dessa komponenter kan enkelt länkas samman och utbyta information i form av properties, eller “props” (ReactJs 2020).

Det finns många andra bibliotek som hade kunnat användas så som Angular.Js, Vue.js eller något som inte är Js-baserat, men React valdes för att det är modernt, smidigt och har mycket material som gör det enkelt att komma in i det.

2.1.1. Material-UI

Ett komponentbibliotek till React som används av bland annat Netflix, Amazon och Betheseda som underlättar utvecklingen av webbsidor och appar med färdiga komponenter (Material-ui 2020).

2.1.2. Google-Maps-API

Ett Javascript-API som låter utvecklaren skraddarsy kartan i sin applikation eller på sin webbsida (Google-Maps-API 2020). API:et har använts för att bygga den karta som handläggarna frågat efter, som nu bland annat kan rita ut områden och visa information om området när det klickas.

2.1.3. React-Pdf-js

För att generera utlåtanden i PDF-form i systemet används React-PDF-js. En React komponent som hittades på Github (Cousins 2020). PDF:en genereras från en JSX-fil som skrivits av examensarbetarna, innehållandes View- och Text-taggar som delar upp innehållet i PDF:en för att efterlikna den utlåtandemallen som erhållits från handläggarna i Malmö.

2.1.4. ExpressJs

Ett minimalistiskt Backend-ramverk för Node.js med öppen källkod. Framtaget för att utveckla webbapplikationer och API:er (ExpressJs 2020). Node.js är även känt för att enkelt kunna skalas upp för att hantera större belastning, vilket kan vara intressant om Knowit vill kunna använda det nya systemet för fler/större städer än Malmö.

2.1.5. Invision

För att ta fram prototyper som handläggarna kan validera användes Invision, som är en applikation som effektivt kan bygga interaktiva prototyper (Invision 2020). Skärmdumpar av den framtagna prototypen laddas upp på hemsidan, sedan kan knappar enkelt markeras och länkas till de uppladdade skärmdumparna. Detta ger känslan av en fungerande navigation. I efterhand så blev aldrig denna interaktiva prototyp klar.

Det finns många teknologier som liknar Invision och då flera testades så valdes Invision främst för att den inte låste så många funktioner i gratisversionen, och för att den var enkel att förstå. Det var inte mycket funktionalitet som krävdes, och det var inte en så stor del av projektet så det lades inte så mycket tid på undersökning av alternativ.

2.1.6. Github

Github är ett Webbhotell som används för versionshantering via Git (Github 2020). Github har använts av examensarbetarna för att enkelt kunna dela nyutvecklad kod med varandra. Detta verktyg har underlättat mycket då mycket av utvecklingen har skett på distans.

Github valdes till versionshantering då det var verktyget som användes av Knowit.

2.1.7. Lequinox

Lequinox är en decentraliserad plattform för hantering av digitala signaturer, kontrakt och kontroll av processer med juridiskt bindande steg utvecklad av Lequa AB. Den har även stöd för rollbaserad access och signering. All information om lequinox kommer från två möten med VDn för Lequa AB. Lequinox plattform hade varit intressant i ett större projekt, men sattes utanför examensarbetets ram.

2.1.8. Rest-server

REST står för: Representational State Transfer. En REST-server är tillståndsfri och fränkopplad från klienten vilket innebär att klient och server kan uppdateras separat så länge de är överens om formatet på meddelanden mellan varandra. Kommunikationen mellan klient och server sker med “request” från klient och “respons” från servern.

3. Metod/analys

Examensarbetet handlade om att skapa en design och MVP för att förbättra handläggarnas arbetsprocess som är ganska omfattande och innehåller många moment som för examensarbetarna var okända. Det fanns inte från början några begränsningar från Knowit på vilka delar av processen som skulle vara en del av ramen för examensarbetet.

Viktiga steg i metoden var att bygga upp examensarbetarnas förståelse för handläggarnas arbete och processens begränsningar, planera en ny design med kravspecifikation som ska underlätta arbetet, och sedan utveckla en begränsad del av designen till en MVP. Då vattenfallsmodellen är en förenklad bild av verkligheten vid kravframtagning (Lauesen 2002, s.3) valdes en mer agil metod, där dokumentationen hölls levande genom hela utvecklingsperioden. Designen och kravspecifikationen kom då att utvecklas samtidigt som MVP:n implementerades.

Arbetet planerades upp i följande faser:

1. Inläsning
2. Elicitering
3. Intervjuer i andra kommuner (tillkom senare)
4. Kravspec
5. Design för MVP
6. Implementering av MVP
7. Testning och utvärdering

När eliciteringen med Malmö stad var genomförd fick examensarbetarna kontakt med fastighetskontoren i Lund och Helsingborg och valde därför att även hålla intervjuer med dessa kommuner. Detta för att få en insikt i hur arbetet med markupplåtelse fungerar i andra kommuner.

3.1. Inläsning

Första fasen var inläsning som började med undersökning av eliciteringstekniker och projektmodeller. När det var bestämt att det skulle hållas intervjuer så studerades dessutom ett par källor om intervjuteknik (Trost 2010; Kvale, Brinkmann & Torhell 1997). Parallellt hölls en teknisk förstudie där utvecklingsteknologier och prototyping-teknologier undersöktes.

Senare, under designfasen kom det att åter bli en del inläsning för att förstå hela processen från att en kund gör en ansökan hos polisen, tills det att kunden faktureras av kommunen. Informationen var väldigt utspridd då ärendet passerar genom två olika myndigheter med var sina hemsidor. För att få en fullständig bild studerades även Markus.

3.2. Elicitering

Den planerade kravprocessen gick ut på att följa fyra steg med ett femte tillagt ganska tidigt i planeringsprocessen. Stegen var:

1. Intervjuer med handläggare
2. Fokusgrupp med handläggare
3. Prototyp-demonstration för handläggare
4. Utvärdering av resultat med slutdesign framtagen
5. Kontinuerlig återkoppling och demonstration för handläggarna under implementering.

Enligt Cao och Ramesh (2008) uppmuntrar det agila arbetssättet konstant respons från intressenterna, därför togs beslutet att hålla ett levande kravdokument under hela processen. Detta underlättar de ändringar som kan uppstå från intressenternas respons.

3.2.1. Intervjuer

Målet med intervjuerna var att få förståelse för hur handläggning fungerar och hur Markus används till detta i nuläget, samt att diskutera vilka problem och önskemål som handläggarna har med systemet. För att ta reda på hur systemet används, vilka kärnuppgifterna är och vilka framtida utvecklingsmöjligheter som finns användes en semistrukturerad intervjuemetod.

Det fanns enligt en föreläsning i kravhantering inom utbildningen av Christin Lindholm tre möjliga alternativ att strukturera intervjuerna på. Dessa var: strukturerade, ostrukturerade och semistrukturerade. En strukturerad intervju har förbestämda, stängda frågor, ofta med förbestämda svarsalternativ. Fördelen med detta är att det är systematiskt och effektivt, vilket underlättar insamling av kvantitativa data och sparar tid om man ska ha många intervjuer. Nackdelar är att intervjuaren måste vara väldigt insatt i området och att man kan missa områden som den intervjuade hade kunnat lyfta ifall denne fått möjlighet. Ostrukturerade går ut på att man ställer väldigt öppna frågor, ofta oförberedda och låter den intervjuade svara utförligt. Fördelen är att det krävs väldigt lite förberedelse inför intervjuerna och man får mer utförliga svar. Nackdelar är att det är stor risk för att mycket av intervjutiden läggs på oväsentligheter samt att intervjuaren behöver vara duktig på att styra intervjun och komma på frågor i stunden. Semistrukturerade intervjuer är ett mellanting där man förbereder en del innan, men har gärna en del öppna frågor och är inte rädd för att prata mer om något som inte var planerat. Nackdelarna är dels att man måste ta reda på fakta om intressenterna i förväg dels att intervjuaren måste kunna styra intervjun. Fördelen är att man kan anpassa delar av intervjun till de olika modellerna. Man kan till exempel undvika ledande frågor genom att göra dem lite mer öppna.

Trost (2010, s39-42) beskriver standardisering som “graden till vilken frågorna är desamma och situationen är densamma för alla intervjuade” och strukturering som i vilken nivå svarens form är densamma mellan intervjuade.

Den semistrukturerade metoden valdes på grund av den begränsade kunskapen som exjobbarna hade om det ursprungliga systemet. Denna metod gav möjligheten att förbereda ett dokument med frågor i förväg, men lämnade det öppet att under intervjun ställa öppna följdfrågor beroende på vad den intervjuade gav för svar. Att hålla intervjun mer som en konversation ger även den intervjuade större möjlighet att ge mer utvecklade svar och att ta upp saker som inte har frågor formulerade kring sig. Enligt Trost's definitioner (2010, s39-42) så skulle intervjuerna kunna sägas ha medelgrad av standardisering och låg grad av strukturering. Medelgrad av standardisering då vissa av frågorna ändrades mellan intervjuer och att tiden för intervjuerna varierade. Låg grad av strukturering då många av frågorna var öppna eller subjektiva och inte hade förbestämda svarsalternativ.

Intervjuerna strukturerades enligt information från huvudsakligen två källor. Dessa var: Lauesen (2002) och Kvale, Brinkman (1997).

“Initially you should ask broad questions about day-to-day work, day-to-day problems, and other items on the list of things to elicit. Make sure to ask about *critical tasks*. When does the user work under stress? When is it highly important that things are 100% correct?” (Lauesen 2002, s340)

“Later, when you have identified critical issues, you may ask more detailed questions, e.g about data volumes, task times, detailed work procedures.” (Lauesen 2002, s340)

Dessa citat bidrog till struktureringen av intervjuerna som började med breda frågor för att förstå de intervjuades arbetsuppgifter och vana av Markus, för att sen gå vidare mot deras kärnuppgifter, och sist mot hur de arbetar med handläggning, hur de jobbar i Markus och hur de utför specifika uppgifter i Markus.

De första två intervjuerna hölls med handläggare vid Malmö stads stadskontor. När intervjuerna planerades uppskattades de ta cirka 40

minuter. Detta var en felberäkning då båda intervjuerna tog 1,5 timme vardera. Det som drog ut på tiden var huvudsakligen uppgifterna som handläggarna skulle genomföra för att ge en klarare bild av arbetsflödet i ett ärende, men också att det är svårt att tidsuppskatta öppna frågor. Den första intervjun hölls med en handläggare som jobbat i systemet i tre år och var, som han själv kallade sig, relativt oteknisk, den andra intervjun hölls med en handläggare som varit med och tagit fram systemet för tolv år sedan och för närvarande är administratör i Markus. Denna skillnad mellan de intervjuade gav en bra bild av de kontraster mellan handläggarna som finns, vilket också gav underlag för design där användarnas kunskap kan skiljas åt drastiskt. Intervjuerna hjälpte också mycket med att förstå handläggarnas arbetsuppgifter och gav en grundläggande förståelse för de moment som är kritiska och måste finnas med i handlägningsprocessen.

Samtliga intervjuer avslutades med att ställa frågan “Har du något att tillägga till detta?”, samt att ta upp några av de viktigaste punkterna. Vilket gav de intervjuade möjligheten att lyfta något som inte fanns i det ursprungliga dokumentet, eller att utveckla sitt svar kring något som diskuterats under intervjun. Detta var något som togs upp som viktigt i Kvale & Brinkmann (2014, s.171).

Intervjuerna dokumenterades genom ljudinspelningar och anteckningar. Ljudinspelningens huvudsyfte var att förtydliga de anteckningar som blev otydliga. Vid intervjuerna ställde en av exjobbarna frågor medan den andre antecknade i ett dokument. Svaren sammanställdes sedan i ett dokument och problem som tagits upp sparades för att kunna ta upp i fokusgruppen.

Även efter intervjuerna så var bilden av handlägningsprocessen i Markus inte tillräckligt tydlig. Detta ledde till att det var svårt att börja skriva krav eller göra listor på problem. Det bestämdes därför att skapa flödesdiagram för att kartlägga arbetsflödet i det nuvarande systemet. Detta blev ett verktyg för att öka kunskapen om dagens arbetsflöde och en hjälp vid identifierandet av problem i Markus. Handledare och Projektansvarig på Knowit rekommenderade att inte ta fram detta

flödesdiagram och ville minimera inblicken i Markus, för att inte den gamla designen skulle influera den nya för mycket. Examensarbetarna bestämde för att ändå ta fram flödesdiagrammet (appendix, Flödesdiagram Markus) som ett verktyg för att öka kunskapen om Markus och minska risken för förvirring och missförstånd. Detta beslut togs baserat på att Lauesen (2002, s.336) rekommenderar kartläggning av nuvarande arbete och problem som första stegen i eliciteringsprocessen enligt citatet:

“It is not possible to elicit requirements right from the beginning. The requirements are the end of the elicitation process, and usually many intermediate work products are needed. Here is a list of important work products:

A description of the present work in the domain.

A list of the present problems in the domain.

A list of goals and critical issues (preliminary requirements).

Ideas for the large-scale structure of the future system.

.....”

(Lausen 2002, s.336)

Tack vare flödesdiagrammet över systemet blev examensarbetarnas kunskap om problemen i Markus mycket djupare och insikten i det framtida systemets funktionalitet förbättrades. Det skapades också flödesdiagram för handläggningen på en mer abstrakt nivå (appendix, Arbetsflöde vid handläggning). Detta för att få en tydlig bild av vilka moment processen handläggning består av, snarare än hur Markus används. De första kraven till ett nytt system skapades, och listor på problem i Markus gjordes för att få mer användbar information till Fokusgruppen.

3.2.2. Prototypdemo/Fokusgrupp

Vid första mötena visade handläggarna ganska litet intresse för designrelaterade frågor, så det bestämdes att designförslag/prototyper skulle tas fram för att utvärderas med handläggarna på ett möte. En

fullständigare prototyp skulle sedan tas fram som grund till applikationen.

En prototyp är användbar som ett verktyg att kommunicera med kunden. Den gör det tydligt hur utvecklarna har tolkat kundens krav samtidigt som den ger en referenspunkt för kunden att kommentera om önskade förändringar och därmed identifiera nya krav. Prototypen hjälper således till med att både identifiera och kvalitetssäkra krav. (Lauesen, 2002, s338-348).

För att slutföra kravbilden över systemet planerades en fokusgrupp med handläggarna. En fokusgrupp underlättar arbetet med kravspecifikationen då det uppmuntrar intressenterna att komma på problem i det nuvarande systemet och sedan lösningar på dessa problem (Lauesen 2002).

Trost (2010, s44-45) menar att fokusgruppsintervjuer innebär att den som intervjuar försöker få deltagarna att själva hålla en konversation som sedan observeras och styrs. Förhoppningen är då att interaktionen mellan deltagarna ska öka engagemang och kreativitet.

Fokusgruppen strukturerades enligt Lauesens lista (2002, s352-354).

1. Invite participants

Samtliga handläggare bjöds in till att delta. Lauesen rekommenderar åtta till sexton deltagare. Under tillfället kunde fem handläggare delta, men då de är åtta totalt ansågs det tillräckligt för att få en god representation av samtliga handläggningsområden.

2. Open the meeting

Målet med fokusgruppen och examensarbetet presenterades. Handläggarna informerades om att det var en MVP som skulle utvecklas och vad detta betyder.

3. Bad experiences

Här fick handläggarna möjlighet att lyfta alla de problem som finns i Markus. Vid några tillfällen fick de påminnas om begränsningarna som kom av att utveckla en MVP samt att de buggar som finns i Markus, såsom att text inte sparas, inte kommer att föras över till det nya systemet. Samtliga problem antecknades på en whiteboardtavla i mötesrummet.

4. Imagine the future

I detta steg fick handläggarna framföra önskemål om vilken funktionalitet ett nytt system skulle kunna innehålla. Samtliga önskningar antecknades på whiteboardtavlan.

5. List the issues

Whiteboardtavlan innehåll nu en stor mängd problem och önskningar på funktionalitet. Det som antecknats strukturerades och problem och önskningar som hörde samman grupperades. Detta för att hjälpa både handläggarna och examensarbetarna att få en bättre överblick av det som tagits fram.

6. Prioritize the issues

När materialet på Whiteboardtavlan var strukturerad kunde handläggarna tillsammans prioritera vilken funktionalitet och vilka problem som var viktigast i det nya systemet.

7. Review the list

Avslutningsvis gick examensarbetarna genom listan med handläggarna och såg till så att alla var nöjda med resultatet av prioriteringen.

Med hjälp av prioritetslistan (4.1) samt önskemål från Lunds och Helsingborgs kommun kunde en design och en kravspecifikation av det nya systemet börja tas fram. En design med fokus på att lösa de effektivitetsproblem som handläggarna upplevde i Markus.

3.3. Intervjuer i andra kommuner

Efter intervjuerna med handläggarna i Malmö så hade bilden av handlägningsprocessen blivit tydligare. Då många av problemen i

processen verkade ganska generella så bestämdes att undersöka hur några andra kommuner arbetar för att få mer kunskap om vad som önskas av ett handläggningssystem. Detta gjordes genom intervjuer i Lund och Helsingborg. Dessa intervjuer utgick från samma frågor som de i Malmö gjorde men antalet frågor relaterade till arbete i Markus förminskades för Lund och togs sedan bort helt Vid intervjun i Helsingborg. Dessutom lades det till frågor som rörde skillnader i arbetssätt och frågor för att bekräfta ifall de höll med handläggarna i Malmö om vilka problem som var viktiga.

3.3.1. Malmö

Malmö har ca 10 handläggare som arbetar med markupplåtelse, och är den största kommunen som undersöks i arbetet. De jobbar med många olika typer av ärenden och kan behöva tydligare struktur i ett system än de andra. Malmö arbetar med att sammanställa en karta med olika slags information om staden där de bland annat vill kunna ha med alla aktiva upplåtelser i staden.

Malmö har använt Markus de senaste 10 åren och är originalkunden för systemet. Det var Knowit som utvecklade systemet åt Malmö stad för drygt 10 år sedan. Eftersom det är Malmö stad som äger Markus så ligger beslut om resurser för vidareutveckling av existerande system på dem. Om Malmö stad vill ha en helt ny mjukvara att ersätta Markus med behöver denna läggas ut som en offentlig upphandling.

3.3.2. Lund

Ansvarig för markupplåtelser inom Lunds kommun kontaktades för att ge insikt i vad en kommun som använder Markus men inte har varit med och tagit fram systemet tycker om systemet. Lund hade från början inget system för markupplåtelse utan fick med Malmös godkännande kopiera Markus och har, som han kallade det, "Lundifiera" det. Med detta menade han att de tog bort funktioner som Lunds kommun inte behövde, till exempel stöd för markupplåtelse av hamnplatser. Lund är missnöjda med Markus och berättade att de kollade på alternativ för att ersätta systemet.

Till skillnad från Malmö, som är en mycket större stad med cirka sex gånger så många upplåtelser om året, har Lund endast en handläggare som har hand om samtliga upplåtelsetyper. Eftersom arbetsbelastningen är så mycket lägre har Lund inte samma behov av ett automatiserat system. Ännu en anledning till detta kan vara att Lunds kommun är mycket hårdare med att ansökan skall vara korrekt ifylld, de kontaktar inte kunden om det är något som saknas utan nekar då ansökan och låter kunden skicka en ny. Vad Lund var ute efter var ett system som var mycket mer pålitligt än vad Markus är. Ett system där handläggaren inte behöver gå tillbaka i systemet och sedan klicka sig in i ärendet igen för att se om informationen sparats korrekt. Ett annat stort krav Lund hade på ett nytt system var att det skulle innehålla kartfunktioner för att förbättra överblicken.

3.3.3. Helsingborg

Helsingborg använder i nuläget en stor mängd exceldokument för att hantera alla markupplåtelser i kommunen. De har under en tid letat efter ett system som de kan använda sig av för att bättre hantera alla upplåtelser, men har inte hittat ett med de funktioner de önskar.

Helsingborg hade stora synpunkter på hur ansökningsprocessen borde gå till, och var kritiska mot hur den var i dagsläget. De önskade att ansökningar skulle kunna komma från deras "system" till polisen för att kunna sälla bort ansökningar innan de kommer till polisen ifall exempelvis platsen för ett event redan är uppbokad en viss dag samt för att göra det tydligt vilka specialregler osv som gäller vid tillstånd i just deras stad.

3.3.4. Sammanfattning

Samtliga kommuner letar efter en effektivare och mer pålitlig mjukvara med modernare funktioner. Den efterfrågade mjukvaran skall underlätta det manuella arbetet och automatisera så mycket som möjligt för att på så sätt spara tid för handläggarna.

Vad som skiljer kommunerna åt är bland annat deras tillvägagångssätt när det kommer till kontakt med kunderna. Om en ansökan som inte är

korrekt ifylld kommer till Helsingborg eller Malmö försöker handläggarna kontakta kunden för att komplettera informationen eller få ett förtydligande. I Lund får ansökan direkt avslag och kunden för göra en ny ansökan med rätt information hos polisen. Något som också skiljer sig är hur tillgängliga kommunerna vill vara för allmänheten. Malmö vill att systemet skall vara så stängt från allmänheten som möjligt så att ingen känslig information läcker. Helsingborg däremot vill vara så tillgängliga som möjligt, gärna med ett system där kunderna har egna användarkonton som kan användas för att logga in, se alla stadens upplåtelse och göra en ansökan om markupplåtelse.

Något som samtliga kommuner var överens om var att polisens arbetssätt försvårar processen. I nuläget skickas ansökan till polisen som tar en avgift, därefter mejlar polisen till kommunen som handlägger. Detta är enligt de som intervjuats onödigt extraarbete eftersom informationen måste fyllas i två gånger. Kontakt med polisen om deras arbetssätt är utanför examensarbetets ram och kommer inte tas upp. Istället skall systemet byggas så att det automatiserar det extraarbete som blir av polisens arbetssätt.

3.4. Kravspecifikation

Med det insamlade materialet från fokusgruppen kunde även arbetet med det nya systemets kravspecifikation börja. Enligt resultat från studien av Cao och Ramesh (2008) dyker krav upp under utvecklingsprocessen vilket betyder att de inte är fördefinierade. För att hålla en agil kravspecifikation skall arbetet med kravspecifikationen fortsätta i början av varje utvecklingscykel eller sprint (ibid). I detta arbete kan utvecklingen som är gjord av examensarbetarna ses som en utvecklingscykel.

Krav för systemets grundfunktionalitet skrevs tidigt ner, till exempel vilken information handläggaren ska kunna skriva in i ärendet och vilka roller som skall finnas. Krav som dessa gav en bild av vad ett väldigt grundläggande system skulle kunna göra, men detta räckte inte för att få en klar kravbild för MVP:n.

Ranglistan (kapitel 4.1) som togs fram under fokusgruppen användes som ett verktyg för att skriva krav för funktionalitet som var högt prioriterad av handläggarna. Dessa krav kom att stödja funktioner som handläggarna saknar i Markus.

En majoritet av kraven är skrivna som user stories, detta för att skapa krav som är lätta att förstå och som motiverar sig själva.

Efter fokusgruppen fick examensarbetarna även tillgång till ett dokument, Processbeskrivning-Malmö (appendix), som är en PDF gjord av Malmö stad som beskriver hela handläggningen, från det att polisen skickar remissen till Malmö tills det att ärendet har gått ut och skall arkiveras. Detta dokument innehöll mycket användbar information som användes för att bland annat skriva krav för vilka textfält som skulle finnas vid varje steg i processen.

Kravspecifikationen har under implementeringen använts tillsammans med prioritetslistan för att planera vilka funktioner som skall implementeras. Efter att ny funktionalitet blivit implementerad har examensarbetarna gått tillbaka till kravspecifikationen för att se vad som är nästa logiska steg mot en fungerande MVP.

Även krav som inte var realistiska att uppfylla inom ramen för detta examensarbete skrevs ner. Detta för att kunna lämna över en väl dokumenterad kravbild till Knowit om de skulle anse att detta arbete är något de vill gå vidare med i framtiden. Kravspecen blir en viktig del av resultatet för att möjliggöra MVP:ns vidareutveckling efter examensarbetet.

3.5. Design

Designen på det nya systemet har fokus på effektivitet. Handläggarna som ska använda systemet arbetar med det varje dag. Detta leder till att om ett designbeslut ligger mellan att göra något intuitivt men mindre effektivt och att göra det mindre intuitivt men effektivare, så kommer

beslutet bli att göra det effektivare. Norman (2013) definierar Human-centered design som:

“The process that ensures that the designs match the needs and capabilities of the people for whom they are intended”

Så eftersom handläggarna ska jobba väldigt mycket i systemet så är det viktigt att ingen information som kan vara användbar tas bort av designskäl.

Efter undersökning av Markus och intervjuer med handläggarna så blev det tydligt att handläggningen av ett ärende har ett ganska naturligt flöde. Steg i flödet är saker som: registrera information från ansökan, kontrollera platsen för önskad upplåtelse och skapa fakturaunderlag. Det är viktigt att detta flöda reflekteras i det nya systemet för att göra handlägningsprocessen så naturlig som möjligt och för att bryta ner en stor, överväldigande uppgift i flera små. Detta ska göras genom att följa designprinciper och undersöka andra system som har tydliga flöden. Exempel på sådana är bokningstjänster och e-handelstjänster, där det är viktigt att hemsidan är lätt att förstå sig på då försäljaren inte vill förlora kunder.

Vid undersökning av bokningssystem och e-handelstjänster, bland annat Junkyard.se och supersavertravel.se så märktes att flera av dessa hade gemensamt att de använde sig av något som kallas för “stepper” (*figur 2*). En stepper kan ge en tydlig överblick när man har mycket information att fylla i och ger små, tydliga grupperingar för deluppgifterna enligt regel 4 i appendix (Schneidermann’s eight golden rules). En stepper är ett verktyg för att uppnå logisk gruppering och närhet i en applikation, vars vikt förklaras i Norman (2013)

“Groupings and proximity are important principles from Gestalt psychology that can be used to map controls to function: related

controls should be grouped together. Controls should be close to the item being controlled”. (Norman. 2013. s22).

Ett annat tillfälle på principen om gruppering enligt citatet ovan används är vid filtrering och sökning av ärenden i listan på startsidan. Dessa kontroller ligger intill listan för att göra det tydligt att de är kopplade samt att alla alternativ för filtrering är grupperade efter vad de påverkar.



Figur 2. Exempel på en steppar till systemet

Schneidermann’s “Eight Golden Rules of Interface Design” (appendix) har använts under designprocessen för att försäkra att designen av samtliga komponenter i systemet uppfyller den användbarhet som eftersträvades i projektet.

Ett annat exempel är designen av knappar och upplistade ärenden som har tagits fram med regel nummer tre i åtanke. När muspekaren hovrar över knappen byter den färg vilket förtydligar för användaren att “detta är något du kan trycka på. I kartan byter muspekaren ikon till en hand om det är ett objekt som går att interagera med.

Vid design av inloggning och ifyllande av formulär användes principerna i regel fem (appendix, Schneidermann’s eight golden rules). Om användaren skriver in fel uppgifter skickas felmeddelande beroende på vad som gick fel. Det finns även stöd för de textfält som används att ge errormeddelanden om de inte fyllts i korrekt. Om till exempel företagets organisationsnummer inte skrivs in korrekt kan fältet rödmarkeras och en errortext som varnar om felet visas. Detta gör det inte bara tydligt att något blev fel utan också vad för åtgärd som behövs.

Det bestämdes tidigt att designen av det nya systemet skulle centreras kring att arbeta i en karta vilket önskats av samtliga handläggare. Kartan är starkt kopplad till ärenden och kan filtreras tillsammans med de upplistade objekten. I kartan kan både punkter och ytor ritas ut. Ytor kan användas för att underlätta prisuträkningar som beror på area.

Designen utvecklades kontinuerligt under implementeringen med input från handledare och projektledare varje vecka. I slutet av projektet fick designen en översyn från en UX-specialist på Knowit. Denna översyn resulterade i förslag på ett uppdaterat grafiskt utseende för login- och startsidan.

3.6. Implementering

Implementeringen av det nya systemet var den största delen av projektet och varade i drygt två månader. Då det blev tydligt ganska tidigt att ett fullständigt system där samtliga varianter av upplåtelser kan handläggas inte skulle vara möjligt så lades fokus på att utveckla en applikation som kunde demonstrera önskad funktionalitet så bra som möjligt. Mindre fokus lades på att optimera de implementerade funktionerna.

Då React var nytt för examensarbetarna tillbringades en del av tiden i början med att lära sig verktyget. Detta ledde till att det blev svårt att uppskatta vilken tid det skulle ta att implementera funktionalitet i systemet till en början.

I början av varje vecka har examensarbetarna satt sig och, med hjälp av kravspecifikationen, planerat vad som var rimligt att hinna med under den veckan. Programmeringen delades sedan upp och arbetades på parallellt. På fredagar visades funktionalitet upp för handledare och projektansvarig i videomöten.

För att sköta versionshanteringen av koden har Github använts. Detta har underlättat då mycket av utvecklingen har skett på olika datorer och

på distans. Det gjorde det även enkelt då kodgranskning gjordes av en konsult från Knowit. Kodgranskningen ledde till bättre strukturering av kod och komponenter.

Kodgranskaren gav även rådet att använda ett komponentbibliotek som heter Materials-ui, för att få ett mer enhetligt utseende på exempelvis de knappar och textfält som används i systemet. Detta ledde också till att mycket tid sparas då komponenterna i Materials-ui inte behövde lika mycket CSS.

En stor del av utvecklingen blev att få de kartfunktionerna som önskats att fungera, att koppla alla komponenter till rätt data samt att hantera denna data i en REST-server.

Varje vecka hölls ett statusmöte där veckans resultat presenterades. Detta var användbart för att se till att projektet rörde sig i rätt riktning.

3.7. Testning och Utvärdering

För att validera kravspecifikationerna och för att låta handläggarna utvärdera systemet bokades ett möte in via videolänk in med samtliga handläggare på Malmökontoret där de kunde få se en demo av systemet. Demon gick till så att examensarbetarna förklarade hur utvecklingsprocessen gått till och hur de arbetar med prioriteringslistan (4.2). Systemet gick sedan igenom på ett sätt som efterliknar det arbetsflöde (appendix, Arbetsflöde vid handläggning) som jobbat fram. Handläggarna uppmuntrades till öppen dialog under hela demon för att alla tankar och funderingar skulle kunna antecknas.

Demo via videolänk var inte optimalt då handläggarna inte själva fick testa systemet. Det blev också tydligt att folk inte hade lika lätt att prata öppet och diskutera via videolänk som de hade i fokusgruppen, något som ledde till att responsen inte blev så omfattande som hade önskats.

3.8. Kommunikation

Kommunikationen under projektet har behövt anpassas under arbetets gång på grund av Covid-19. I början av arbetet tillbringades nästan all tid på Knowits kontor i Malmö. Detta gjorde det enkelt att hålla god kontakt med handledare och projektledare för att få feedback på arbetet. Minst en gång i veckan hölls ett avstämningsmöte där nya funktioner i systemet visades upp och examensarbetarna kunde be om hjälp med diverse saker som dykt upp under veckan. Efter pandemins utbrott så behölls statusmöten som ett zoommöte i veckan, men den naturliga kontakten på kontoret försvann. Detta påverkade kommunikationen och produktiviteten negativt eftersom det inte var möjligt att dagligen diskutera examensarbetet och få hjälp med småsaker. Även kommunikationen mellan examensarbetarna blev lidande då de gick från att sitta bredvid varandra på kontoret till att kommunicera mer via meddelanden och röstsamtal.

Kommunikationen med handläggare skedde via mejl med kontaktpersoner samt genom inbokade möten i verkligheten och Teams. Kommunikation med polisen inleddes via mejl för att få en inblick i deras arbetssätt, men då de tog över två veckor på sig att svara ganska kortfattat samt att projektet behövde avgränsas ännu mer så släpptes den tråden. Kommunikation med Lequinox skedde via mejl samt genom två möten på deras kontor.

3.9. Anpassningar under arbetets gång

Det var i början av processen tänkt att steget efter intervjuerna skulle vara prototypdemonstration. Tidsbrist ledde till att inga fullt utarbetade visuella designförslag hann göras till andra mötet, men idéer som inte var interaktiva diskuterades med handläggarna i en fokusgrupp enligt Lausen (2002, s.343) där det togs hjälp av handläggarna att ta fram en lista på problem i det befintliga systemet och rangordna dessa från viktigast till minst viktiga. Ändringen skedde till stor del på grund av att möten med handläggarna kunde ta veckor att få inbokade

Tidigt i implementeringsfasen fanns också en plan för att bygga grunden för en databas. Idén var att kunna ha databasen och servern i molnet. För att göra detta skulle företaget hjälpa till med kunskap inom detta område, men då folk försvann från kontoret försvann också denna del av projektet och examensarbetarna valde att fortsätta att använda sig av en lokal server på sina datorer.

I slutet av projektet var en demo för handläggarna planerad. Här skulle de få testa funktionaliteten genom att lägga till en ny kund i systemet och sedan lägga till ett ärende för denna kund. Därefter skulle de få skapa en upplåtelse i detta ärende där datum, upplåtelseid och beskrivning kan läggas till. Här skulle de även få testa de ritfunktioner som lagts till. Förhoppningen med denna demo var att ge examensarbetarna värdefull feedback och kritik på det som utvecklats samt ge underlag för en vidareutvecklingsplan. På grund av Covid-19 blev denna demo istället en videodemo där systemet visades via skärmdelning, examensarbetarna genomförde uppgifterna och handläggarna kommenterade under tiden.

MVP:n blev således inte så mätbart verifierad som hade önskats, men handläggarna verkade ändå uppskatta de intrycken de fick av systemet samt hoppades på vidareutveckling. MVP:n tillsammans med den slutgiltiga kravspecen bör vara en bra grund för att se till att ett nytt system realiserar.

3.10. Källkritik

I detta delavsnitt motiveras pålitligheten hos de källor som använts i arbetet.

Software Requirements - Styles and Techniques

Skreven av Sören Lauesen, för närvarande Professor emeritus vid Köpenhamn IT-universitet.

Denna bok användes i kursen Kravhantering, ETSF30.

The Design of Everyday Things

Don Norman är chef för designlabbet vid University of California och hedersprofessor vid Tongji Universitys College of Design and Innovation.

Boken används som kurslitteratur i fem kurser på LTH och har enligt Google Scholar citerats över 116 000 gånger.

Den kvalitativa forskningsintervjun

Steinar Kvale var professor i pedagogisk psykologi vid Psykologiska Institutet på Aarhus Universitet.

Handledaren på Knowit tipsade om denna bok då han använt den själv i sitt eget examensarbete på LTH.

Designing the User Interface

Ben Shneiderman är professor och forskar inom interaktion mellan datorer och människor.

Rekommenderad av handledare på Knowit

Reactjs.org

Ansvarig utgivare för hemsidan är Facebook Open Source. Syftet med hemsidan är att göra det enkelt för nya utvecklare att börja utveckla ett projekt i Reactjs.

Agile Requirements Engineering Practices: An Empirical Study

Cao är lektor inom informationsteknik och beslutsvetenskap vid Old Dominion University och Ramesh är professor inom informationssystem vid Georgia State University. Texten har citerats i över 150 olika artiklar och publikationer.

Kvalitativa intervjuer

Jan Trost är professor vid Uppsala universitet och forskar inom sociologi och socialpsykologi

Dataskyddsförordningen

Dataskyddsförordningen har skrivits av Datainspektionen.

Då Datainspektionen är en statlig myndighet under justitie departementet som har i uppgift att skydda individers uppgifter på internet anser vi att de är pålitliga.

Material-ui

Eftersom komponenterna i biblioteket har gått att implementera i det nya systemet anser examensarbetarna att materialet är tillräckligt pålitligt att använda.

Google-Maps-api

Ett API utvecklat av google. Materialet är anpassat för att uppmuntra utvecklare att använda det, vilket leder till intäkter för google genom användning av API-nyckeln.

React-pdf-js

En komponent utvecklad under MIT-licens, vilket betyder att koden kan återanvändas av andra privatpersoner. Då komponenten fungerar väl i systemet har examensarbetarna valt att använda just denna.

Github

Versionshantering på Knowit sköts via github, där det skapades ett repository som examensarbetarna kunde använda för att lagra sin kod.

Invision

Informationen är skriven som reklam, men då programmet testades och fungerade som önskat så behövdes inte större tillit till källan.

ExpressJs

Dokumentation för API som använts som referens under projektet. källan försöker vara objektiv och har endast som syfte att vara en tydlig beskrivning av vad som är möjligt.

REST-server

Codeacademy är ett utbildningsföretag. Då deras affärsmodell baseras på att erbjuda en bra utbildning anser vi att de är pålitliga med information om programmeringsteknologier.

4. Resultat

Eliciteringsdelen av projektet resulterade i att examensarbetarna fick god insikt i de problem som fanns att lösa och fick mycket material för att kunna bygga en MVP.

4.1. Inläsning

Inläsningen resulterade i en förståelse för hur intervjuerna bör struktureras för att vara så användbara som möjligt. Resultatet var att det valdes en semistrukturerad intervjumetod samt att en fokusgrupp hölls för att komplettera och följa upp.

4.2. Elicitering

Eliciteringen resulterade i väl besvarade intervjuer, en rangordnad problemlista och två flödesdiagram över: arbetsflödet i markus och handläggningens deluppgifter.

4.2.1. Intervjuer

Svaren från intervjuerna sammanställdes till ett dokument för att kunna vara underlag vid vidareutveckling av MVP:n. Resultatet från intervjuerna användes också för att förbereda fokusgrupp, krav och design.

4.2.2. Fokusgrupp

Resultatet från fokusgruppen var en kombination av punkt 2 & 3 i listan på Work products i Lausen (2002, s.336). Den rangordnade listan med diskuterade lösningar ser ut som följande:

1. Jobbigt med signatur på papper
 - Digital signatur
2. Dålig Överblick (på flera sätt. bl.a.datum registrerad istället för startdatum visas)
 - Vy över ärenden som tagit slut

- Schema för evenemang (kalender per handläggare)
 - Färgkodning av ärenden
 - Kartvy med ytor och info (punktmarkering till en början)
 - Förfrågningar (preliminära bokningar utan polistillstånd)
3. Mejl syns inte i ärenden
 - Koppla mejl till ärende (olika kategorier av mailkonversationer)
 - Ansökning till systemet (slippa ladda ner bifogade filer)
 4. Jobbigt att ladda upp flera saker samtidigt
 - Drag and drop
 5. Jobbigt att registrera ärenden manuellt
 - Skrapa basinfo från handläggarkladd
 6. Systemet är inte portabelt
 - Bärbar version
 - Ta bilder till ett ärende
 - Kolla tillstånd på en plats
 - Enkelt skicka en notis till kunden att kontroll är avklarad
 7. Inget svar från ekonomisystemet (får nu kund + summa (ingen unik id))
 - Försök att få svar som innehåller unik identifierare till upplåtelse
 8. Kan ej ändra underlag under fakturering
 - Tillåt uppdatering av fakturaunderlag under fakturering
 9. Påminnelser till kunder saknas.
 - Möjlighet att skicka ut automatiska påminnelser till kunden att tillståndet är på väg att utgå.

4.3. Intervjuer i andra kommuner

Resultatet från intervjuerna i de andra kommunerna blev ökad förståelse för vad handläggningsprocessen kan innebära och dess problem, samt fler förslag på önskad funktionalitet till ett handläggningssystem. Förståelsen för vilka behov som var gemensamma för alla kommunerna var viktig vid framtagandet av designen.

Viktigaste saker för Lund:

- Kartfunktioner
- Digitala signaturer
- Pålitlighet

Viktigaste saker för Helsingborg:

- Modulärt,
- Tillgängligt för medborgare i staden
- Möjlighet att hantera ansökningar innan de kommer till polisen

Svaren från intervjuerna sammanställdes tillsammans med svaren från de första intervjuerna i Malmö.

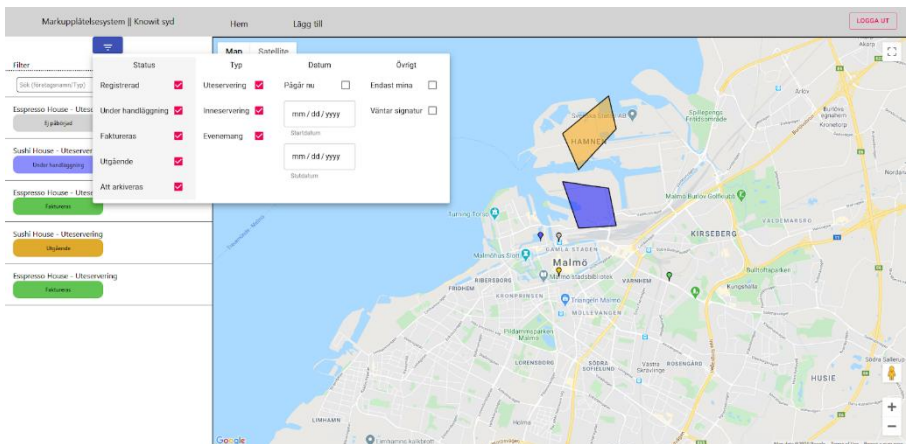
4.4. Kravspecifikation

Kravspecifikationen (appendix) har varit under utveckling under hela projektets gång, och innehåller både saker som är implementerade i MVP:n och saker som är planerade för framtida utveckling.

Kravspecifikationen bör tillsammans med sammanställningen av intervjuerna vara en grund som möjliggör vidareutveckling med MVP:n som grund.

4.5. Design

Designarbetet har resulterat i en MVP som ger bättre överblick över handläggarens ärenden. I vänsterspalten av figur 3 visas handläggarens ärenden. Det färgade fältet som finns i varje ärende visar, tillsammans med texten i fältet, vilken status ärendet har. Färgkodningen stämmer även överens med de färger som visas i kartan. Ärendelistan har även en filtreringsfunktion där handläggaren kan välja att filtrera listan för att till exempel endast visa ärenden med statusen “Under handläggning”.

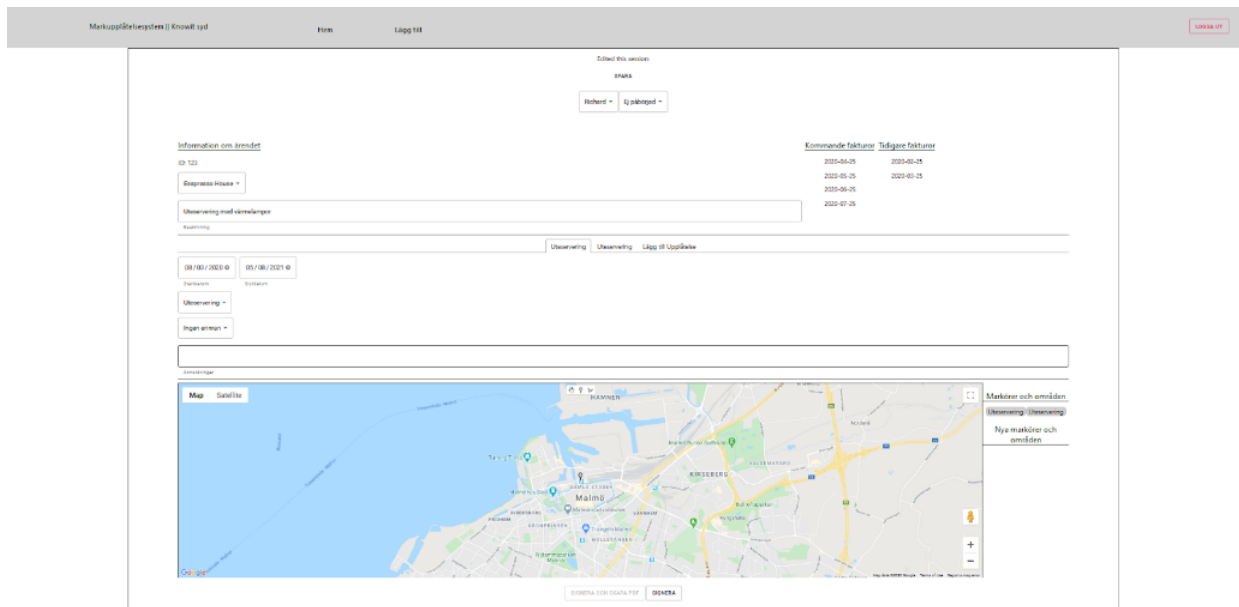


Figur 3. Hem-sidan i systemet med filtreringsmenyn öppen

För att underlätta arbetet för handläggarna valde examensarbetarna att samla de informationsfält som fylls i under handläggningen på samma sida (figur 4). Den övre delen visar information om ärendet och kunden. Nedre hälften visar information om en upplåtelse som hör till ärendet.

Med hjälp av flikar kan handläggaren enkelt byta till en annan upplåtelse i samma ärende, utan att behöva navigera till en annan sida. I upplåtelsen ligger även kartan där handläggaren kan placera en markör eller rita ut ett område där kunden har angivit att uteserveringen skall vara. Denna karta är programmerad att visa endast de markörer som hör till upplåtelsen.

Översynen av designen från UX-specialisten gav förslag på ett grafiskt mer tilltalande utseende på login- och hem-sidan.



Figur 4. Ärendesida, utzoomad för att få plats med allt i en bild

4.6. Implementering

De kartfunktioner som implementerats fungerar väl och fyller syftet att visa grundtanken för den funktionalitet som är tänkt i systemet.

Överst i ärendet (figur 4), under texten "Edited this session" finns en lista som ännu inte är färdigimplementerad, men som användes för att ge handläggarna stöd med ändringshistorik i ärendet. När information på sidan ändras, läggs till eller när något ritas i kartan loggas detta i listan. När man sedan klickar på "Spara" skickas dessa ändringar till servern där de sparas i ärendet.

Längst ner på sidan finns två knappar, en för att skapa en PDF som innehåller den information från ärendet och upplåtelsen som behövs vid ett yttrande, och en som signerar ärendet genom att skicka en uppdatering till servern. För att visa funktionalitet för handläggarna har knappen för att skapa pdf och signera implementerats så att den är avstängd om användaren inte är angiven som “Signing-officer”.

4.7. Testning och Utvärdering

Testningen och utvärderingen kunde på grund av covid-19 inte ske som planerat med interaktiva tester där handläggare skulle utföra olika uppgifter. Istället hölls demonstration av MVP:n i ett videomöte där handläggare var inbjudna. Resultatet blev då främst små kommentarer och frågor. Det kom främst positiva kommentarer, men det märktes att fränkopplingen som videomötet ledde till gjorde det svårare att få användbar kritik.

5. Slutsats

Detta kapitel svarar på frågeställningarna i början av rapporten. Här presenteras även framtida utvecklingsmöjligheter för systemet.

5.1. Vad är kärnuppgifterna hos handläggarna?

En handläggares huvuduppgifter består av att undersöka om de ansökningar om markupplåtelse som kommer in till polisen går att genomföra från den information som kunden har skrivit i sin ansökan. I det fall att upplåtelsen gäller en uteservering innefattar detta ofta att ta sig ut till platsen för att mäta upp området och kontrollera att det inte kommer för nära en bilväg eller trottoar. Handläggaren skall också se till att uteserveringen följer de regler som finns satta av Malmö stad. När all denna information är kontrollerad och stämmer skall ett beslut skapas och skrivas under av ansvarig på avdelningen som skickar beslutet vidare till polisen. När polisen i sin tur mottagit beslutet skickar de tillbaka en bekräftelse på detta och handläggaren kan skicka fakturaunderlaget som skapats från den information som samlats under handläggningsprocessen. Detta fakturaunderlag skickas till sist till ett ekonomisystem som hanterar skapandet av fakturor till kunden. När tillståndet tar slut ska ärendet arkiveras och fakturering ska avslutas.

5.2. Hur ser arbetsflödet ut vid handläggning i det nuvarande systemet?

I nuläget finns ett system som uppfyller de grundläggande krav för ett system som ska klara av att handlägga upplåtelse och kan kommunicera med ekonomisystemet. Vad detta nuvarande system saknar är funktioner som underlättar handläggarnas arbete, så som en integrerad kartfunktion och ett tydligt flöde genom hela upplåtelsen. För att all information ska kunna läggas in behöver man klicka sig mellan många olika sidor, vilket lätt gör att man tappar bort sig. Handläggningen består också av många manuella, repetitiva steg så som att manuellt skriva över information till separat kartsystem eller manuellt kopiera alla delar av ett ärende till ett nytt när ett ärende ska kopieras.

5.3. Hur kan ett förbättrat arbetsflöde se ut?

För att förbättra arbetsflödet är det inte endast handläggarnas arbetssätt som behöver ändras. En sak som behöver ändras är ärendets flöde i systemet. I det nuvarande systemet behöver handläggare klicka runt till olika sidor för varje funktion i ärendet vilket lätt kan göra användaren förvirrad. Istället bör en sida för handläggning designas på så sätt att all den väsentliga informationen kan fyllas i och ses på ett och samma ställe. Detta kommer ge en bättre översikt över hela ärendet och leda till mindre förvirring.

Vid polistillstånd äger polisen processen och kommunen agerar endast som remissinstans, vilket betyder att ett nytt arbetsflöde måste anpassas efter deras arbetssätt. Detta innebär att ansökningar måste komma in till polisen för att sedan skickas vidare till kommunen, även om det är handläggarna som gör hela arbetet med upplåtelsen. Många förbättringar hade kunnat göras i ansökningsprocessen om polisen inte hade ägt ärendet, vilket i sin tur hade underlättat arbetet för handläggarna. Till exempel vill Helsingborgs fastighetskontor att kunderna ska kunna göra sina ansökningar direkt på deras hemsida för att ha ansökningar som rör staden samlade på ett ställe och för att kunna se till så att de som ansöker fyller i information som staden behöver men som kanske inte polisen behöver.

5.4. Hur skapas en användbar prototyp?

För att kunna ta fram en väl utvecklad prototyp som kan användas för validering av kravspecifikationen behöver en väl strukturerad eliciteringsprocess genomföras. Användarna är en av de viktigaste intressenterna och den grupp som vet bäst vad som behöver finnas i systemet som ska utvecklas. Deras kunskap behöver därför efterforskas grundligt och gärna med olika metoder. Till exempel fungerar intervjuer bra för att få en överblick av problemet som systemet skall lösa, medan en fokusgrupp fungerar bättre för att få fram önskemål och krav på mer detaljerade funktioner som skall ingå i systemet. Prototypen får gärna vara "klickbar" eller interaktiv för att ge en mer arbetsliknande känsla. Prototypens största styrka är att den gör det

enklare för användarna att ge önskemål då de har en utgångspunkt att gå från.

På grund av tidsbrist kunde ingen prototyp visas vid fokusgruppen, något som var planerat från början för att validera den kravspecifikation som tagits fram. Beslutet att inte genomföra prototypdemon kan i efterhand ses som rätt val då handläggarna inte intresserade sig speciellt mycket för det grafiska vid den slutliga demon.

5.5. Vilka teknologier ska användas vid utveckling av MVP:n?

Ett tidigt beslut i implementeringsprocessen var valet av teknologier. Beslutet för front-end-bibliotek landade i React som valdes för möjligheten att utveckla modulära applikationer med dess komponentbaserade filosofi. Andra viktiga faktorer var stort stöd för olika moduler som vi visste kunde bli användbara under processen. Några sådana som användes var: Material-UI, Google-Maps-API och React-Pdf-js.

Express.js användes för att skapa en enkel REST-server som hanterar datan. Ingen komplett databas skapades då MVP:n som skapades saknar stora delar av bakgrundsarkitekturen som krävs i ett komplett system.

5.6. Hur ska känslig information hanteras och lagras?

Ett markupplåtelsesystem behandlar personuppgifter och det blir därför nödvändigt att tänka på hur dessa ska lagras. Det kan också finnas tillfällen då ett ärende behöver få ett yttrande från en tredje part som man inte vill ska ha tillgång till att se eller ändra någon mer data än den nödvändigaste. Ett sätt man kan hantera detta på är rollbaserad access.

Lequinox är en plattform utvecklad av Lequa AB och är till för att hantera digitala processer som har juridiskt bindande steg i sig. Kontakt med Lequa AB togs då handläggare klagat på att yttranden i dagsläget signeras för hand och för att projektledaren på Knowit hade sedan tidigare haft kontakt med företaget. Plattformen erbjuder säkra digitala signaturer och möjligheter till rollbaserad access. De föreslog en decentraliserad plattform som sökande, kommun och polis alla kommer åt för att göra sina delar av processen. Detta är för sökande och kommunen en bättre lösning, men kräver en ändring i polisens arbetssätt.

I nuläget är systemets säkerhets baserad på att samtliga sidor, utom login-sidan, kräver en JSON Web Token för åtkomst. Denna token genereras i servern när användaren loggar in och sparas i webbläsarens session-storage. Token skickas med vid signering för att kontrollera att användaren har denna behörighet. Denna kan också användas till att begränsa data som visas för visa användare eller roller.

5.7. Hur ska MVP:n utvärderas och testas?

Under projektets gång hölls kontinuerliga möten med handledare och projektansvarig. Vid dessa möten gavs generell feedback på de saker som utvecklats under veckan. Något som planerats från början var att vid slutet av projektet hålla ett prototyp-test för handläggarna där de själva fick testa systemet för att kunna validera kraven. I testet skulle handläggaren få en känsla för hur det nya flödet i systemet fungerade. Resultatet från detta test skulle också ge underlag för att kunna färdigställa vidareutvecklingsplanen.

På grund av covid-19 var ett sådant här test inte genomförbart. Då det inte gick att träffa handläggarna för att hålla ett test var en demo via videolänk det alternativet som fanns. Demovisningen strukturerades så att examensarbetarna visade upp systemet och berättade om vad de utvecklat och vilka designprinciper som låg bakom det nya systemet. Handläggarna uppmanades att komma med kommentarer under uppvisningen, men det lämnade mycket att önska efteråt. Efter att

handläggarna fått se hela systemet var de positiva till de förändringar som gjorts och var hoppfulla över hur det nya systemet kan komma att se ut. Något som saknades var konstruktiv kritik om de ändringar som gjorts i arbetsflödet.

Något som kunde gjorts för att undvika detta problem är att ha bjudit in en person från Malmös fastighetskontor till de avstämningsmöten som hölls, för att hålla de mer uppdaterade om vad som hände. Detta hade också gjort att det inte var lika mycket information att ta in vid ett tillfälle. Att hålla kunden mer kontinuerligt uppdaterad hade också kunnat underlätta utvecklingsprocessen då de kunnat ge mer detaljerade synpunkter på det som utvecklats under veckan.

5.8. Reflektion över etiska aspekter

Eftersom de ansökningar som skickas in kan innehålla personuppgifter till privatpersoner behöver systemet följa de riktlinjer som satts av dataskyddsförordningen GDPR (Dataskyddsförordningen 2018). Det skall till exempel vara möjligt att kontakta fastighetskontoret och be om att få utdrag av den information de har sparat om dig som privatperson, denna information ska även kunna raderas helt ur systemet om kunden så önskar.

För att få tillåtelse att spara personinformation behövs saklig grund för detta. Vilket fastighetskontoret kan bestyrka då personuppgifterna är kritiska för att kunna fakturera kunden.

Vidare så räknas upplåtelser som offentliga handlingar och kan därför inte döljas helt. Det finns intressenter som begär ut listor på pågående upplåtelser i staden. Dessa listor kanske kan ses över så att de inte innehåller någon information som inte är nödvändig.

5.9. Framtida utvecklingsmöjligheter

Här presenteras funktionalitet som är väsentlig för att få ett fullständigt system för markupplåtelser av uteserveringar samt funktionalitet som är högt prioriterad av handläggarna för att underlätta deras arbete.

5.9.1. Upplåtelsehantering

Eftersom det utvecklade systemet är en MVP för handläggning av uteserveringar finns stora möjligheter till vidareutveckling. Eftersom den information som behövs för att kunna bestyrka en upplåtelse för torgförsäljning skiljer sig från vad som behövs för att bestyrka en uteservering behöver systemet anpassas för samtliga upplåtelse typer. Detta hade kunnat lösas genom att ärendesidan genererar fält/komponenter baserat på upplåtelse typen. En lösning av denna typ förstärker också den modularitet som eftersträvats i systemets design då inköparen av systemet kan välja de upplåtelser som är relevanta för deras arbete.

5.9.2. Bärbar version av systemet

Enligt ranglistan som togs fram under fokusgruppen var en bärbar version av systemet mycket efterfrågad då handläggarna ofta är ute på plats för att kontrollera att upplåtelsen är genomförbar, att kunden följer de riktlinjer som satts av handläggaren eller att platsen är i godkänt skick efter att upplåtelsen har utgått. I nuläget skriver handläggaren ut ärendet i pappersform och tar med till platsen för att kunna verifiera att allt står rätt till, men en bärbar version av systemet hade kunnat förenkla denna kontroll på många sätt. En bärbar version kan stödja sökning i systemets databas och på så sätt hade handläggaren kunnat se ansökan och verifiera att det som kunden ansökt om är genomförbart. Även en funktion för att föra anteckningar hade underlättat då dessa hade kunnat sparas till ärendet direkt i systemet och vara tillgängliga för handläggaren när hen är tillbaka på kontoret. Möjligheten att med den bärbara version, vare sig den är på en mobiltelefon eller läsplatta, ta foton som sparas i ärendet hade sparat mycket tid för handläggaren då denna inte behöver flytta foton till ärendet manuellt när hen är tillbaka på kontoret.

5.9.3. Kartfunktioner

De funktioner som finns för systemets karta ger i nuläget möjlighet att rita ut områden tillhörande en upplåtelse, visa information om upplåtelsen i ett informationfönster när man klickar på markören eller området samt grundläggande filtrering för att kunna visa de markörer som är aktuella för handläggaren.

För att öka användbarheten i kartan kan “marker clustering” implementeras enligt Figur 5. Detta hade gjort så att om kartan är såpass utzoomad att markörer och områden överlappar varandra, något som gör det svårt att få en klar bild av vad som finns på kartan, kommer dessa att samlas till en markör med en siffra inuti som står för hur många markörer den representerar.



Figur 5. Exempel på markörkluster

En funktion för att beräkna areor för utritade områden behöver implementeras då priset för upplåtelsen baseras på områdets storlek. Denna funktion behöver också kunna visa områdets sidlängder för att underlätta ritandet för handläggaren. I ansökan kommer det ofta med en ritning som visar hur uteserveringen planeras att byggas, att kunna se längden på de sidor som ritas i kartan gör både arbetet lättare och mer precist.

Det fanns en önskan att få chansen att implementera mer avancerade funktioner till exempel sidlängder för polygoner, automatisk beräkning av area och adressökning direkt i kartan. På grund av tidsbrist gick inte detta att förverkliga, men har antecknats i kravspecifikationen.

5.9.4. Fakturahantering

En central del i vidareutvecklingen av systemet är att det ska kunna kommunicera med fakturasystem. Tidigt togs beslutet att det var utanför ramen för arbetet att implementera fakturahanteringen och därför ingen större undersökning i vad det skulle krävas för att implementera sådana funktioner.

5.9.5. Mailfunktioner

För att öka användbarheten och även underlätta spårbarheten i systemet behöver diverse mailfunktioner läggas till. Till exempel vill handläggarna att mejl som hör till ett specifikt ärende skall vara tillgängligt på ärendesidan och även att det enkelt skall gå att skicka mejl från sidan. Om sådan funktionalitet implementeras kommer all kommunikation tillhörandes ett ärende vara enkelt tillgänglig.

Vidare kom det fram önskemål under fokusgruppen att kunna skicka ut påminnelser till kunder vars upplåtelse är på väg att gå ut, med en integrerad mailfunktion skulle handläggaren redan under handläggningsperioden kunna förbereda ett mailutskick som till exempel två veckor innan upplåtelsen löper ut skickas till kunden för att påminna denne.

5.9.6. Pdf-scanner

För att underlätta arbetet med ärenderegistrering önskar handläggarna att denna process automatiseras. Eftersom formatet på remissdokumentet är standardiserat kan en scanner implementeras till att skrapa data i dokumentet och automatiskt fylla i den i textfälten steppern där ärenden läggs till. När data har lagts in kan registratören kontrollera att allt stämmer innan denne godkänner ärendet. Ansökan

ska alltid bifogas i ärendet, så om någonting skulle gå fel kan detta åtgärdas även efter att ärendet skapats.

6. Terminologi

- Markus - Det ursprungliga systemet som skall nyutvecklas
- Offentlig upphandling - En process för inköp av varor och tjänster som görs av den offentliga sektorn och som regleras av speciella lagar och regler. Syftet med reglerna är att säkerställa att myndigheter behandlar alla som vill få affärskontrakt med offentlig sektor lika.
- Remiss - en hänvisning av ett ärende till myndigheter eller organisationer, så kallade remissinstanser, för yttrande innan beslut fattas.
- Yttrande - Beslut fattat om ärendet är genomförbart.
- Ärende - Ett ärende skapas från den information som kommer från polisens remiss. Ärenden har unika id i systemet.
- Upplåtelse - En upplåtelse är en del av ett ärende och avser en viss typ (ex. uteservering) under en viss tid.
- Handläggare - Handläggare på kommun som handlägger ansökningar om markupplåtelse
- Signing-officer - Handläggare som har rättighet att signera yttranden
- Hem-sidan - Startsidan av systemet där handläggaren enkelt får tillgång till sina ärenden och kan se en karta som visar alla upplåtelser i systemet.
- Ärendesidan - En sida i systemet där handläggaren fyller i information om ärendet och upplåtelsen för att kunna använda som fakturaunderlag.

Källförteckning

Lauesen, S. (2002). Software Requirements, Styles and Techniques

Norman, D. (2013). The design of everyday things

Kvale, S, Brinkmann, S & Torhell, S. (1997). Den kvalitativa
Forskningsintervjun

Schneiderman, B. (2014). Designing the user interface: strategies for
effective human-computer interaction.

Reactjs.org (2020). Tillgänglig: <https://reactjs.org/> [20-02-10]

Cao, L och Ramesh, B. (2008). "Agile Requirements Engineering Practices:
An Empirical Study," i IEEE Software, vol. 25, no. 1. s60-67.

Trost, J. (2010). Kvalitativa intervjuer

Datainspektionen (2018). Dataskyddsförordningen.
[https://www.datainspektionen.se/lagar--
regler/dataskyddsförordningen/](https://www.datainspektionen.se/lagar--regler/dataskyddsförordningen/) [20-06-06]

Material-ui (2020). <https://material-ui.com/> [20-04-29]

Google-Maps-API
[https://developers.google.com/maps/documentation/javascript/t
utorial](https://developers.google.com/maps/documentation/javascript/tutorial) [20-03-15]

Cousins, M. (2020). <https://github.com/mikecousins/react-pdf-js> [20-
05-12]

Github (2020). <https://github.com/about> [20-03-11]

Invision (2020). <https://www.invisionapp.com/cloud/prototype> [2020-06-06]

ExpressJs (2020) <https://expressjs.com/en/4x/api.html> [2020-06-06]

Rest-server (2020) <https://www.codecademy.com/articles/what-is-rest> [2020-06-16]

Appendix

Intervjufrågor

Öppnings-/intressentfrågor

- Hur länge har du jobbat som handläggare (även på andra ställen)?
- Hur länge har du jobbat med Markus?
- Har du tidigare erfarenhet av liknande System?
- Ungefär hur många ärenden handlägger du under en vanlig dag?
- Hur tycker du Markus fungerar överlag?
- Finns det några situationer då arbetet brukar bli stressigt?

Kärnuppgifter

- Vilka är de vanligaste ärendena du får?
- Hur ser ett exempel på flöde ut från ansökan till avslutat ärende?
- Hur vet du att en markupplåtelse är genomförlig?
- Hur ser din interaktion med polisen ut?
- Hur ser din interaktion med kunderna ut?
- Vilka saker har du som handläggare inte behörighet att godkänna?
- Vad händer vid återkallelse av polistillstånd, och hur märker du det?
- Kontrolleras tillstånd/avtal för specifika verksamheter?
- Hur räknas pris för markupplåtelse ut?
- Hur fungerar syning av plats i processen?
- När är det viktigt att saker går 100% rätt? Vilka steg är kritiska?

Faktafrågor (Ställs troligtvis bara till en person)

- Vilka dokument skickas mellan Malmö stad och polisen (remiss, yttrande, beslut)?
- Vilka aktörer är med i hela processen?

- Interagerar kunderna något med malmö stad direkt, eller går allt via polisen?
- Finns det ett specifikt format ansökningar kommer på (PDF, förbestämd stil, osv..)?
- Vilka delar av processen kräver signaturer, och av vem?
- Hur fungerar syning med avseende på systemet?
- Hur förlängs ett ärende?
- Hur fungerar kraven på spårbarhet och vilka saker är det som behöver loggas?
- Vad är signing officer, vid skapande av ny användare?
- Vilka olika användare finns det och vad innebär de? (admin, handling officer, super admin, top admin, user)
- Vad är Raindance(fakturering)? Är det ett system malmö använder till allt?
- Vad händer när man arkiverar ett ärende?
- Vad används kpi till?
- Var kommer externt id från?

Frågor om Markus

- Finns det något med Markus som gör ditt arbete omständigare än det behöver vara?
- Är det något som fungerat tidigare som blivit sämre med tiden?
- Finns det funktionalitet i Markus som du sällan eller inte alls använder?
- Vilka är de bästa funktionerna i Markus?
- Om något går fel, vad brukar det vara?
- Finns det någon roll/användare som du känner saknas?
- Kan man ha ett ärende utan en registrerad kund? Om inte, brukar ni behöva radera många kunder?

Arbetsflöde

Öppna frågor:

- Vad är det första som händer när du får ett ärende?
 - *Fortsätt med liknande frågor för att se hur handläggaren jobbar genom hela ärendet*

- Hur ofta brukar man behöva uppdatera ett ärende?
- Hur gör du för att kontrollera att polistillstånd är godkänt/finns?
- Är det något i processen som behöver upprepas många gånger?
- Vilka moment tar mest tid?

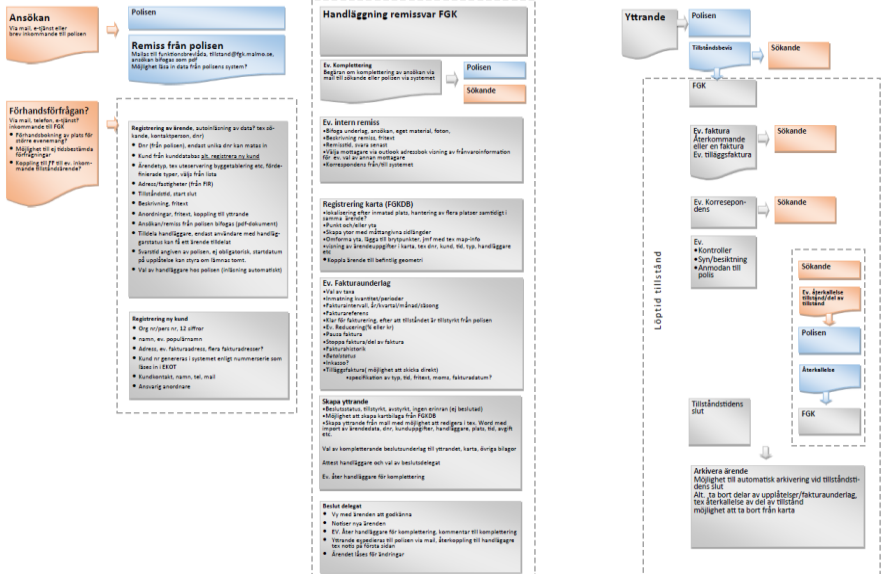
Uppgifter (stegvis beskrivningar):

- Skapa ett ärende för en ny kund
- Räkna ut pris för ett registrerat ärende
- Generera faktura för registrerat ärende
- Uppdatera informationen hos en kund
- Aktivera periodiska fakturor för ett ärende
- Uppdatera information i registrerat ärende
- Kontrollera och avsluta ärende
- Ta fram och granska ett registrerat ärende
- Kontrollera ifall syning är utförd i pågående ärende
- Kontrollera polistillstånd i pågående ärende

Avslutande

- Finns det något övrigt du vill berätta för oss om systemet?

Processbeskrivning Malmö



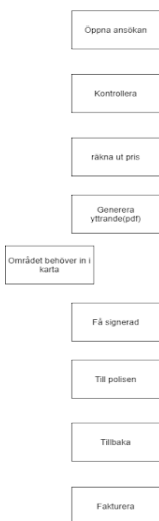
- Administration**
- Lägg upp användare med olika behörighet och andra behörigheter, ta bort användare
 - Läsa rättigheter
 - Ändra tillägare
 - Handläggare med begränsade rättigheter. Tex för att lägga in syn-/beskrivningsprotokoll
 - Kopiera fakturor till EKOT
 - registrator
 - Admin/superadmin
- Byta handläggare på flera ärenden samtidigt
- Uppdatera kunduppgifter
 - Ändra status på ärenden tex arkiverat till under handläggning
 - Lägg upp och uppdatera yttrandemallar
 - Lägg upp och uppdatera kategorier för bilagor och tjänsteanteckningar
 - Lägg upp och uppdatera konteringsinformation
 - Lägg upp och uppdatera taxa
 - Lägg upp och uppdatera upplåtelsestyper
 - Ta bort gamla kunder/inaktiva
 - Arkivering/galtning/statistisk sammanställning
 - Ta bort gamla kunder, GDPR?
- Rapportfunktioner**
- Delegationslista
 - Arrangemanglista
 - underlag påminnelser, mail/adresslöser
 - efter typ, inläsnings, ärendes status,
 - Ekonomiuppföljning?
- Spara ner tex tillsämbetsbevis eller andra dokument utförn sökning? Vid begäran av offentlig handling etc.
- Ärendesammanställning för "GDPR-utdrag"
- Sökfunktion**
- plats, namn, org nr/pers nr, handläggare, tillsämbetsperiod, anteckningar, typ, dir, ärendes-status
 - Avancerad sökning/"googlesökning"
 - Sortera på olika kategorier, datum, typ, plats, ärendesstatus etc
 - Skapa rapport till excel från sökresultat
- Övrigt**
- Viss om kund har obetalda fakturor, tex flagga när kund knyts till ett ärende
 - Ärenden uppgifter i ärendet efter att handläggningen är klar, tid, typ, plats, faktura, andra handläggare
 - Göra tjänsteanteckningar
 - Bifoga dokument, "drag and drop", bifoga flera bilder som ett samlat dokument
 - Arkivera flera ärenden samtidigt
 - Byta handläggare på enskilda ärenden
 - Skapa återkommande upplåtelsestyper, tex varje lördag maj-oktober
 - Vy beslutsdelegat
 - Ärenden ett godkända Åter handläggare för komplettering, ev kommentar
 - Skapa underlag för kreditfaktura från systemet, digital attestering? Möjlighet till val av bifogade handlingar från systemet
 - Tydliga lög som visar ändringar i ärendet
 - Visning av alla ärenden på kund, gamla och pågående?
 - Skapa krediteringsunderlag, mallas till redovisningsenheten,

- Mobilhet**
- Sä ärendesinformation i mobil eller läsplatta
 - söka upp ärenden
 - bifoga foto
 - egna tjänsteanteckning
 - Vissa ärenden i karta utförn GPS-position?
- Ärendesstatus**
- registrerat
 - under handläggning
 - intern remiss
 - Väntar komplettering
 - Väntar delegationsbeslut
 - Yttrande skickat/väntar beslut från polisen
 - under fakturering/klar, arkiverad
- Notiser**
- Nytt ärende, registrator/handläggare
 - Remissid gått ut
 - Ända remissvar inkomna
 - Egna påminnelser
 - Svarstid för polisremiss går snart ut?
 - Komplettering inkommet
 - Delegationsbeslut/yttrande skickat till polisen
 - Tillsämbetsbevis från polisen
 - Ärendeslutas inkommet
 - Övrigt? (annan korrespondens eller inkomna handlingar)

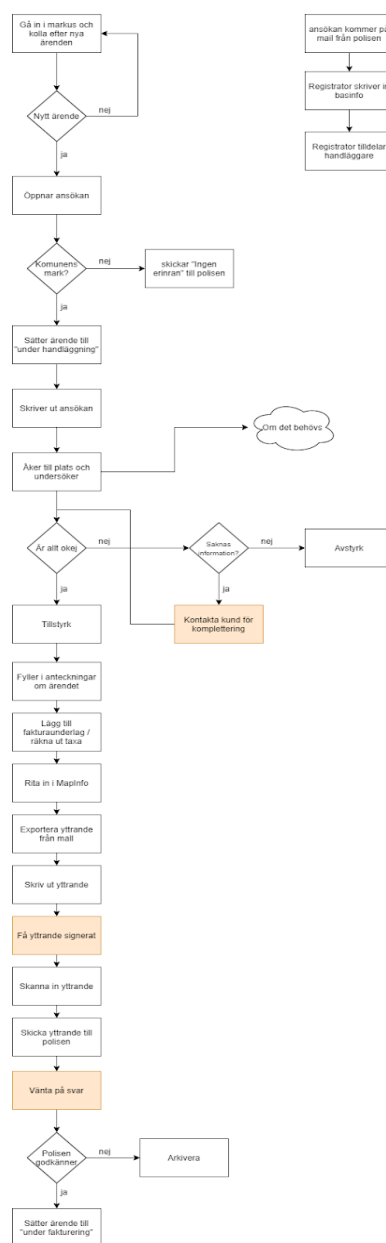
- Yttrande**
- Yttrande skickat/väntar beslut från polisen
 - under fakturering/klar, arkiverad
- Startida**
- Visning av ärenden på första sidan, "nya händelser", som standard egna ärenden med möjlighet att välja alla handlingar
 - registrator
 - under handläggning
 - intern remiss
 - Väntar komplettering
 - Väntar delegationsbeslut
 - Yttrande skickat/väntar beslut från polisen
 - Sökfunktion
 - Registrera ärende
- Åtkäva ärende**
- Möjlighet till automatisk arkivering vid tillsämbetsbevis slut
 - Ät: ta bort delar av upplåtelse/fakturaunderlag, tex återbetala av del av tillsämbetsmöjlighet att ta bort från karta
- Yttrande**
- Yttrande skickat/väntar beslut från polisen
 - under fakturering/klar, arkiverad
- Åtkäva ärende**
- Möjlighet till automatisk arkivering vid tillsämbetsbevis slut
 - Ät: ta bort delar av upplåtelse/fakturaunderlag, tex återbetala av del av tillsämbetsmöjlighet att ta bort från karta

Arbetsflöde vid handläggning

Handläggning i Markus avskalad



Handläggning i Markus i malmö



Registrering till Markus



Schneidermans “Eight Golden Rules of Interface Design”

1. Strive for consistency.

Consistent sequences of actions should be required in similar situations; identical terminology should be used in prompts, menus, and help screens; and consistent commands should be employed throughout.

2. Enable frequent users to use shortcuts.

As the frequency of use increases, so do the user's desires to reduce the number of interactions and to increase the pace of interaction. Abbreviations, function keys, hidden commands, and macro facilities are very helpful to an expert user.

3. Offer informative feedback.

For every operator action, there should be some system feedback. For frequent and minor actions, the response can be modest, while for infrequent and major actions, the response should be more substantial.

4. Design dialog to yield closure.

Sequences of actions should be organized into groups with a beginning, middle, and end. The informative feedback at the completion of a group of actions gives the operators the satisfaction of accomplishment, a sense of relief, the signal to drop contingency plans and options from their minds, and an indication that the way is clear to prepare for the next group of actions.

5. Offer simple error handling.

As much as possible, design the system so the user cannot make a serious error. If an error is made, the system should be able to detect the error and offer simple, comprehensible mechanisms for handling the error.

6. Permit easy reversal of actions.

This feature relieves anxiety, since the user knows that errors can be undone; it thus encourages exploration of unfamiliar options. The units of reversibility may be a single action, a data entry, or a complete group of actions.

7. Support internal locus of control.

Experienced operators strongly desire the sense that they are in charge of the system and that the system responds to their actions. Design the system to make users the initiators of actions rather than the responders.

8. Reduce short-term memory load.

The limitation of human information processing in short-term memory requires that displays be kept simple, multiple page displays be consolidated, window-motion frequency be reduced, and sufficient training time be allotted for codes, mnemonics, and sequences of actions.

Kravspecifikation - Markupplåtningssystem

Terminologi

- Ärende - En remiss från polisen innehållande ansökan från kund, identifieras med ett externt Id som också kommer från polisen. Ett ärende kan innehålla flera upplåtelser. Ett ärende hanteras av en handläggare.
- Upplåtelse - En upplåtelse har en adress, en typ och en period. Den har också faktureringsunderlag kopplad till sig.
- Faktureringsunderlag - information för att kunna generera fakturor för en upplåtelse. Inkluderar val av taxa beroende på typ av upplåtelse samt eventuell period och kostnad.
- Kund - En kund är en person eller organisation som kan ha ett eller flera ärenden. En kund har ett organisationsnummer, kontaktpersoner och adresser för fakturering.
- Tillstånd - typ av ärende som är kopplad till polistillstånd
- Handläggare - Person på Fastighet- och gatukontoret som hanterar ärenden.
- Registrator - Handläggare vars huvudsakliga uppgift är att registrera ärenden från polisen i markus och tilldela dem en handläggare.
- Yttrande - Kommunens yttrande om ett ärende, som remissinstans för polisen
- Användare - Samtliga roller i Markus (user, Handling officer, admin).
- Signing officer - En användare i systemet med rättighet att signera ärenden.
- Drag & drop - Kopiera en fil till systemet genom att dra en fil från utforskaren till ett fält i systemet.
- Yttrandemall - Förprogrammerad pdf som automatiskt fylls i med information från ärendet.

1. Funktionella krav

1.1 Uppgifter

1.1.1 Registrering av ärende

Deluppgifter

- Ange externt id (från polisen)
- Ange kund
- Ange beskrivning
- Bifoga ansökan från polisen
- Tilldela handläggare
- Lägg till upplåtelser
 - Ange upplåtelseyp
 - Ange tidsintervall
 - Rita in punkter och ytor

1.1.2 Registrering av kund

Deluppgifter

- Ange organisations/personnummer, 12 siffror
- Ange namn
- Ange address
- Ange kundens namn
- Ange kundens telefonnummer
- Ange kundens e-mailadress
- Lägg till kontaktpersoner

1.1.3 Komplettering av ärende

Deluppgifter

- Ange kund
- Ange ärendetyp
- Ange adress
- Ange tillståndstid
- Ange beskrivning
- Bifoga ansökan till ärende
- Tilldela handläggare
- Bifoga bilder
- Bifoga filer
- Välja taxa

- Ange beskrivande text
- Ange anordningar i ett textfält

1.2 User-stories

Registrator

- 1.2.1 Som Registrator vill jag kunna utföra uppgift 1.1.1 för att kunna registrera ärenden åt handläggare.
- 1.2.2 Som Registrator vill jag kunna utföra uppgift 1.1.2 för att kunna lägga till nya kunder i systemet.
- 1.2.3 Som Registrator vill jag att informationen från polisens remisser skall läsas och automatiskt flyttas till ett nytt ärende för att spara tid.

Handläggare

- 1.2.4 Som handläggare vill jag kunna logga in i systemet för att kunna behandla mina tilldelade ärenden.
- 1.2.5 Som handläggare vill jag kunna se en vy av ärenden tilldelade till mig för att enkelt få en överblick av mitt arbete.
- 1.2.6 Som handläggare vill jag kunna utföra uppgift 1.1.3 för att kunna komplettera ett ärende med ofullständig information.
- 1.2.7 Som handläggare vill jag kunna bifoga flera filer samtidigt till ett ärende för att det tar lång tid att bifoga en fil åt gången.
- 1.2.8 Som handläggare vill jag kunna bifoga filer till ett ärende med drag & drop för att underlätta bifogandet.
- 1.2.9 Som handläggare vill jag kunna gå till en kartvy av en upplåtelse från ärendets sida så att jag kan få en bra överblick över upplåtelsen.
- 1.2.10 Som handläggare vill jag att det på startsidan skall synas när en upplåtelse snart går ut för att enkelt ha tillgång till dessa.
- 1.2.11 Som handläggare vill jag kunna se ärendets kommande fakturor för att kunna kontrollera att fakturaunderlaget är korrekt.
- 1.2.12 Som handläggare vill jag på startsidan kunna söka efter ett ärende med ärendenummer för att kunna hitta det.

- 1.2.13 Som handläggare vill jag på startsidan kunna söka på företagsnamn för att få fram de ärenden som hör till det företaget.
- 1.2.14 Som handläggare vill jag på startsidan kunna söka på ett organisationsnummer för att få fram de ärenden som hör till organisationsnummret.
- 1.2.15 Som handläggare vill jag på startsidan kunna söka på en adress för att få fram ärenden som ligger i närheten av den platsen.
- 1.2.16 Som handläggare vill jag på startsidan kunna söka efter en handläggare för att få fram de ärenden som hör till handläggaren.
- 1.2.17 Som handläggare vill jag på startsidan kunna söka på en tillståndsperiod för att kunna se ärenden som är aktuella under den perioden.
- 1.2.18 Som handläggare vill jag på startsidan kunna söka på upplåtelseyp för att kunna se ärenden av den typen.
- 1.2.19 Som handläggare vill jag på startsidan kunna se en karta som visar de upplåtelse som finns i systemet för att enkelt kunna se var de finns.
- 1.2.20 Som handläggare vill jag ha tillgång till systemet från en läsplatta för att ha tillgång till ärendet när jag är ute hos en kund.
- 1.2.21 Som en handläggare vill jag att alternativen vid handläggning ska anpassas efter ärendets status så att det blir tydligt vad som behöver göras för att gå vidare.
- 1.2.22 Som en handläggare vill jag kunna se sidlängd på den area jag ritat på kartan för att kunna följa de dimensioner som kunden har ritat i sin ansökan.
- 1.2.23 Som handläggare vill jag att prissättning av upplåtelsen ska beräknas efter upplåtelsens area för att underlätta prissättningen.
- 1.2.24 Som en handläggare vill jag kunna välja att en påminnelse ska skickas till kunden via e-mail när ett tillstånd är på väg att gå ut.

Signing officer

- 1.2.25 Som Signing Officer vill jag se en PDF-sammanställning av ärendet med dess fakturaunderlag innan jag signerar för att kontrollera att vad som skrivits är korrekt.

- 1.2.26 Som Signing Officer vill jag kunna signera ärenden i systemet så jag inte behöver skriva ut ärenden.
- 1.2.27 Som Signing Officer vill jag att ändringar i ärenden loggas så att jag kan se vem som gjort ändringar och när innan jag signerar.

Admin

- 1.2.28 Som admin vill jag kunna exportera en lista, månadsvis, med alla upplåtelser som har yttranden för att kunna skicka den till (de som skall ha den).
- 1.2.29 Som admin vill jag kunna lägga till nya handläggare så att alla som arbetar i systemet kan ha en egen inloggning.
- 1.2.30 Som admin vill jag kunna lägga till användare med rättighet att endast bifoga filer till ärendet så att externa entreprenörer kan bifoga besiktningsprotokoll.
- 1.2.31 Som Admin vill jag kunna byta handläggare på flera ärenden samtidigt för att spara tid.
- 1.2.32 Som Admin vill kunna lägga till nya yttrandemallar för att kunna anpassa vid specialfall.
- 1.2.33 Som Admin vill jag kunna uppdatera befintliga yttrandemallar för att kravet på pdf:en kan ändras.

1.3 Övriga funktionella krav

- 1.3.1 På samtliga sidor skall det finnas en knapp som tar användaren tillbaka till startsidan
- 1.3.2 Vid ifyllnad av fält ska önskat format vara tydligt.
- 1.3.3 Vid ifyllnad av fält ska inkorrekt ifyllda fält rödmarkeras

2 Datakrav

2.1 User-stories

- 2.1.1 Som Signing officer vill jag kunna sammanställa ett ärende till en PDF för att kunna skicka som yttrande till polisen.

2.2 Övriga Datakrav

- 2.2.1 Ärenden skall sparas i databasen i tio (10) år och därefter raderas
- 2.2.2 Som handläggare vill jag kunna exportera koordinaterna för de upplåtelser som finns i systemet till andra kartsystem.
- 2.2.3 Som handläggare vill jag få en bekräftelse när faktureringsprogrammet har mottagit mitt ärende för att veta att allt gått rätt.