HHSK_ny

Erik Knutsson

Tillämpa prototypverktyget Ozlab för olika mobiltelefoner i fält

Applying prototyping tool Ozlab for various mobile phones in the field

Informatik

C-uppsats

Termin: VT-14

Handledare: Lennart Molin

Examinator: John Sören Pettersson

# Abstract

Studier visar på att den absolut vanligaste IKT-enheten i utvecklingsländer är mobiltelefoner. För att kunna utveckla användarvänliga system som befolkningen i dessa länder kan använda sig av krävs någon form av testning. Många designers menar att användartester ska påbörjas långt innan färdiga system finns. Prototyping borde alltså vara en viktig del av utveckling av mobilbaserade informationssystem i utvecklingsländer.

Denna uppsats behandlar prototyping-verktyget Ozlab som tillämpar Wizard of Oz metodologin vid användartester. Framför allt diskuteras hur man kan använda sig av Ozlab som ett portabelt prototyping-verktyg för tester mot mobiltelefoner. Testsessioner om hur väl den portabla miljön fungerar och vad testledare anser om att köra Ozlab mot mobiltelefoner tas även upp. Vidare förklaras även vad som är viktigt att ha i åtanke vad gäller testning mot mobiltelefoner med Wizard of Oz-metodologin.

Resultaten visar att det är möjligt att skapa en portabel miljö av Ozlab-systemet som sedan kan användas i uppkopplingslösa miljöer trots att systemet numera är webbaserat. Däremot kan enbart relativt nyare mobiltelefoner ansluta till Ozlab och tester mot en faktiskt "feature phone" går inte att utföra vilket gör att man får överväga att använda olika typer av skärmbaserade mockuper. Tack vare testning med Wizard of Oz-tekniken kan man snabbt utesluta vissa lösningar då man får snabb feedback från slutanvändaren.

# Förord

Jag vill först tacka min sambo Linda Nilsson som har hjälpt mig oerhört mycket vid skrivandet av denna uppsats.

Vill även tack min handledare Lennart Molin för den vägledning som han givit mig. Framför allt vill jag tacka John Sören Pettersson som har gick goda råd och tagit sig tid att ge synpunkter på det jag skrivit.

Sist vill jag tacka alla testdeltagare men ett extra stort tack till testledarna som gjorde det möjligt att utföra denna uppsats.

- Erik Knutsson

Innehållsförteckning

[Abstract 1](#_Toc395815380)

[Förord 2](#_Toc395815381)

[1.0 Inledning 1](#_Toc395815382)

[1.1 Problemområde 1](#_Toc395815383)

[1.2 Syfte & Undersökningsfrågor 3](#_Toc395815384)

[1.3 Målgrupp 3](#_Toc395815385)

[1.5 Metod 3](#_Toc395815386)

[1.5.1 Kunskap 3](#_Toc395815387)

[1.5.2 Metoder för besvarande av undersökningsfrågorna 4](#_Toc395815388)

[1.5.3 Etiska överväganden 5](#_Toc395815389)

[2.0 Teoretiska utgångspunkter och perspektiv 7](#_Toc395815390)

[2.1 Prototyping 7](#_Toc395815391)

[2.2 Wizard of Oz (WOz) 9](#_Toc395815392)

[2.3 Ozlab systembeskrivning 11](#_Toc395815393)

[2.3.1 Första versionen av Ozlab 11](#_Toc395815394)

[2.3.2 Nuvarande version av Ozlab 12](#_Toc395815395)

[2.3.3 Sencha Ext JS 12](#_Toc395815396)

[2.3.4 Beteenden i nuvarande Ozlab 12](#_Toc395815397)

[2.3.5 Funktioner enbart för TL i nuvarande Ozlab 13](#_Toc395815398)

[2.4 Prototyping med Wizard of Oz 13](#_Toc395815399)

[2.5 Mobiltjänster i utvecklingsländer 14](#_Toc395815400)

[2.5.1 Mobiltjänst för bankärenden 15](#_Toc395815401)

[2.5.2 Läs- och skrivsvårigheter 15](#_Toc395815402)

[2.5.3 Human–computer interaction (HCI) for development 17](#_Toc395815403)

[2.5.4 Vilka typer av mobiler dominerar i u-länder? 18](#_Toc395815404)

[2.6 Varför Wizard of Oz? 19](#_Toc395815405)

[2.7 Implikationer för min studie 20](#_Toc395815406)

[3.0 Genomförande och Resultat 21](#_Toc395815407)

[3.1 Utredning av mobil kompabilitet i Ozlab 21](#_Toc395815408)

[3.1.1 Krav för godkänd enhet 22](#_Toc395815409)

[3.1.2 Testgenomförande 22](#_Toc395815410)

[3.1.3 Testresultat 22](#_Toc395815411)

[3.1.4 Sencha EXT JS - webbläsarstöd 23](#_Toc395815412)

[3.1.5 Websockets 23](#_Toc395815413)

[3.2 Installation av portabel ozlabmiljö 25](#_Toc395815414)

[3.2.1 Idén med den portabla ozlabmiljön 25](#_Toc395815415)

[3.2.2 Installation av virtuellmiljö och implementering av Ozlab-systemet 26](#_Toc395815416)

[3.2.3 Testning av den portabla miljön 26](#_Toc395815417)

[3.2.4 Nödlösning p.g.a. WiFi-komplikationen 26](#_Toc395815418)

[3.2.5 Testning av den routerbaserade lösningen 27](#_Toc395815419)

[3.3 Uppsättning av litet fälttest i labbmiljö 28](#_Toc395815420)

[3.3.1 Design av prototyp 28](#_Toc395815421)

[3.3.2 Pilottest – val av layout 30](#_Toc395815422)

[3.3.3 Pilottest åtgärder 30](#_Toc395815423)

[3.3.4 testplan 31](#_Toc395815424)

[3.3.5 Testmiljö 32](#_Toc395815425)

[3.4 Testresultat 34](#_Toc395815426)

[3.4.1 Sammanställning av intervjufrågor för TL 34](#_Toc395815427)

[3.4.2 observation av TL 34](#_Toc395815428)

[3.4.3 Diskussion med TL 34](#_Toc395815429)

[3.4.4 Sammanställning av intervjufrågor för TP 36](#_Toc395815430)

[3.4.5 Observation av TP 37](#_Toc395815431)

[4.0 Analys & Slutresultat 38](#_Toc395815432)

[4.1 U1: Vad är viktigt att tänka på när ”Wizard of Oz”-metoden används i utveckling för tjänster till mobiltelefoner i utvecklingsländer? 39](#_Toc395815433)

[4.2 U2: Ozlab, som nu är utformad som en webbtjänst, hur kan den köras ”lokalt”, speciellt mot mobiltelefoner? (Vad är systemkraven för att köra en demo eller ett test med Ozlab i fält där Internetuppkoppling inte finns?) 40](#_Toc395815434)

[4.3 U3: Vad är för- och nackdelar med den webbaserade Ozlab-tekniken mot mobiltelefoner? 42](#_Toc395815435)

[4.4 Förslag på fortsatt utvecklingsarbete 44](#_Toc395815436)

[Källförteckning 45](#_Toc395815437)

[Bilagor 48](#_Toc395815438)

[Bilaga 1 Medgivande för testperson 48](#_Toc395815439)

[Bilaga 2 Testsessions prototyp och dess navigering 49](#_Toc395815441)

# 

# 1.0 Inledning

I inledningen beskrivs vad prototyping är generellt samtidigt som en metod vid prototyping presenteras, Wizard of Oz. Här kommer även verktyget Ozlab att introduceras som är ett prototyping-verktyg som stöder Wizard of Oz-metodologin. Problemområdet för denna uppsats tar upp varför prototyping är ett bra verktyg vid systemutveckling.

## 1.1 Problemområde

Prototyping används inom systemutveckling för att i förväg illustrera ett system eller en del av ett system. Det blir lättare att få ett helhetsperspektiv på systemet och det går i tidigt skede att testa produkten. Vilket i sin tur ger utvecklarna möjlighet att justera systemet innan det ska sättas i produktion – utan prototyping kan tvingas till signifikanta ändringar och stora kostnader efter implementation (se till exempel kapitlet om prototyping i *Handbook of Human-Computer Interaction*; Houde & Hill, 1997).

Ett nyutvecklat system som inte lever upp till kraven kan i slutändan bli en kostsam process. Även om det tar tid att genomföra prototyping så kan det i slutändan ge en mer effektiv slutprodukt. Vid prototyping testar man hur väl en användare förstår hur ett system ska användas. Oavsett vilket system det gäller så är interaktionen mellan användaren och systemet ytterst viktig. Om användaren inte förstår vad som händer vid interaktion med ett system, hur vet de då att de har gjort rätt?

Under mina studier på Karlstads universitet har jag vid flera tillfällen kommit i kontakt med ett prototypingverktyg, Ozlab, som anammar prototypingtekniken Wizard of Oz. Bland annat har dessa tillfällen gett mig möjligheten att bidra till utvecklingen av Ozlab och framtagning av nya funktioner. Men framförallt har det handlat om att använda Ozlab i prototypingsyfte. Ozlab är utvecklat vid Karlstads universitet och är sedan 2013 en webbtjänst.

Vid diskussion med användarlabbet vid Karlstads universitet, framförallt professor John Sören Pettersson och labbingenjör Malin Wik, har vi tillsammans kommit fram till en intressant frågeställning som uppsatsen kommer att utreda: anpassning av Ozlab till användning ute i fält. Ozlab är en mjukvara som stödjer ”Wizard of Oz”-tekniken vid användartester och demonstrationer. Denna teknik låter användare tro att hen interagerar med en programvara, men i själva verket är det en annan person som agerar som programvaran.

Mjukvaran Ozlab är uppbyggt som en webbtjänst, där all kommunikation med mjukvaran sker via Karlstad universitets servrar och all interaktion med mjukvaran sker via webbläsare, t.ex. via en mobiltelefon. För att kunna använda sig av tjänsten behöver man antingen ha tillgång till internet eller använda dig av Karlstad universitets nätverk. Detta kan innebära vissa begränsningar ifall man vill utföra fälttester:

* Är det möjligt att komma åt systemet om man inte sitter innanför Karlstads Universitets brandväggar?
* Om systemet ligger lokalt på en dator hur kommer uppkoppling mellan mobiltelefon och dator se ut?
* Kommer äldre mobiltelefoner kunna hantera systemet – kan de ”rita” upp gränssnittet?

Ozlab-systemet kommer att ha stor betydelse vid utförandet av denna uppsats. Därmed kommer också prototyping och metodologin Wizard of Oz ha stor inverkan på uppsatsen.

Systemet har i denna C-uppstatsstudie använts vid testning av olika prototyper mot telefoner. Valet av Ozlab har gjorts på grund av att jag och en annan student, Jonatan Jansson, har varit med och vidareutvecklat det webbaserade Ozlab i en tidigare kurs vi läst under vårterminen 2014 och även för att det är tillgängligt för oss studenter på Karlstads universitet.

Uppsatsen kommer framför allt koncenteras på hur väl Ozlab kan användas med (äldre) mobiltelefoner som interaktionsverktyg för användaren. Detta för att det finns enligt Pettersson ett behov av interaktiv rapid prototyping i u-länder där varken elnät eller datorer finns. Däremot finns viss mobiltäckning och folk har ofta mobiler; se till exempel konferensvolymerna från HumanITs konferensserie *Mobile for Development* (M4D); Pettersson (red., 2008), Svensson och Wicander (red., 2010), Kumar och Svensson (red., 2012), samt Niang et al. (red., 2014; [www.m4d2014.net](http://www.m4d2014.net)). Problemet är att mobil­telefonerna som används i u-länder ofta saknar stöd för ny teknik. Eftersom den nya versionen av Ozlab är webbaserad och är utvecklad enligt nya webbstandarderna kommer det uppstå problem vid kommunikationen mellan de olika enheterna, då äldre mobiltelefoners webbläsare inte stödjer denna teknik. Hur man kan kringgå detta problem tas upp i denna uppsats.

Wizard of Oz-tekniken låter användaren tro att hen interagerar med en programvara men i själva verket interagerar hen med en annan person, den s.k. wizarden. Personen ”bakom skynket” simulerar interaktion åt användaren och ger indata när det behövs. Wizard of Oz gör det möjligt att skapa en illustration av mjukvara som man inte har tillgång till (Nilsson and Siponen 2006, s.6). Det är således möjligt att skapa interaktion mellan användare och mjukvara utan att behöva behandla de olika begränsningar som kan finnas vid ej färdigställda programvaror.

Den kände interaktionsdesignern Bill Buxton förklarar att ”The object is not to make the actual system, but to mock up something that users can actually experience, thereby enabling us to explore design concepts in action and as experienced far earlier in the process than would otherwise be possible” (Buxton 2007, s.240). Han nämner även att när man tar fram en sådan prototyp så ska den vara billig, snabb att förverkliga, och bara vara så trovärdig att den fungerar för ett visst ändamål under designprocessen (Buxton 2007, s.240).

Ozlab-systemet tillämpar ”Wizard of Oz”-tekniken genom interaktion med grafiska användargränssnitt. I Ozlab kallar man de gränssnitten för ”skal”, där wizard kan byta mellan de olika scenerna för att illustrerar navigation. Skalen byggs upp med bilder och/eller fördefinierade objekt som finns i Ozlab. På alla bilder och objekt kan wizarden sätta olika typer av funktionalitet; det krävs ingen programmering för att införa de olika funktionaliteterna.

## 1.2 Syfte & Undersökningsfrågor

Syftet med denna C-uppsats är att undersöka och ta reda på om det är möjligt att använda sig av det webbteknikbaserade Ozlab-systemet i fält med mobiltelefoner som interaktionenheter för användaren.

I denna uppsats kommer följande frågor besvaras:

1. Vad är viktigt att tänka på när ”Wizard of Oz”-metoden används i utveckling för tjänster till mobiltelefoner i **utvecklingsländer**?
2. Ozlab, som nu är utformad som en webbtjänst, hur kan den köras ”lokalt”, speciellt mot mobiltelefoner? (Vad är systemkraven för att köra en demo eller ett test med Ozlab i fält där Internetuppkoppling inte finns?)
3. Vad är för- och nackdelar med den webbaserade Ozlab-tekniken mot mobiltelefoner?

## 1.3 Målgrupp

Karlstads universitets användarlabb – det kan ge forskarna och lärarna nya möjligheter att använda Ozlab i miljöer utan internetuppkoppling.

Utvecklare och kravställare som vill kunna demonstrera och användartesta idéer utan att behöva programmera en färdig produkt.

## 1.5 Metod

Goldkuhl ger i kompendiet i ”Kunskapande” en beskrivning av ”olika typer av kunskaper som man kan eftersträva” vid kunskapsutveckling (2011, s.11). Inledningsvis kommer jag skriva kort om de olika ”kunskapsformer” som kommer att appliceras under denna studie. Sedan ges förklaring till hur undersökningsfrågorna kommer att besvaras. Detta sker stegvis och ger en utförlig beskrivning på hur data kommer att samlas in.

## 1.5.1 Kunskap

I detta underavsnitt följer tre av Goldkuhls sätt att beskriva kunskap som är relevanta som slutprodukter av mitt arbete (som svar på U1-U3). I nästa underavsnitt beskrivs hur sådan kunskap kan uppnås.

**Vägledande kunskap**

Goldkuhl ger en kort beskrivning om kunskapsformen:

”Vägledande kunskap innebär att man talar om hur man bör gå tillväga i olika situationer. Det kan vara regler, riktlinjer, råd, föreskrifter för handlande i olika situationer. Vägledningar innebär metodkunskap. Ibland kallas denna kunskapsform för normativ kunskap. Detta är dock en olyckligt vald term som snarare bör reserveras för värdekunskap (se nedan). Vägledande kunskap kan kallas preskriptiv kunskap.” (Goldkuhl 2011, s. 14)

Denna form av kunskapande beskriver alltså hur man bör gå tillväga i olika situationer för att nå fram till sitt resultat. I mitt fall skulle det alltså vara mina läsare som förstår hur de ska applicera Ozlab för att nå fram till resultat som är relevanta för deras frågor. Jag vill dock påpeka att ordet ”preskriptiv” inte känns träffande eftersom alla prototypingmetoder alltid måste anpassas både efter resultat man vill uppnå och den budget och de resurser man råkar ha.

**Karaktäriserande kunskap**

Kunskapsformen innebär beskriva egenskaper om det man studerar. Vilket i mitt fall blir Ozlab-systemet. Goldkuhl skriver att man ska ”tolka och klargöra vilka *egenskapstyper* som en viss klass av företeelse har” och ”beskriver de *egenskapsvärden* och den *variation* som en viss egenskapstyp kan anta” (Goldkuhl 2011, s.13). Han nämner även att egenskapstyperna kan beskrivas med antingen kvalitativ eller kvantitativ form. Egenskaper hos Ozlab och Wizard of Oz-metodologin är självklart av den kvalitativa formen men vissa egenskaperna hos Ozlab och mobiltelefoner blir av kvantitativ form, till exempel minivärden för hastigheter.

**Förklarande kunskap**

Under förklarande kunskap ska man beskriva varför något är som det är. Goldkuhl säger att ”Man anger orsaker, grunder, skäl eller förutsättningar” (Goldkuhl 2011, s. 13) till varförnågot är som det är. Jag beskriver varför vissa delar av Ozlab-systemet och dess kommunikation måste finnas och därför även varför det är som det är.

Nedan beskrivs hur detta används vid besvarande av undersökningsfrågorna.

## 1.5.2 Metoder för besvarande av undersökningsfrågorna

**U1. Vad är viktigt att tänka på när ”Wizard of Oz”-metoden används i utveckling för tjänster till mobiltelefoner i utvecklingsländer?**

Här används dels litteraturstudier och diskussioner med personer som har använt sig av en tidigare version av Ozlab, dels studier där Wizard of Oz används i fält. Speciella användargrupper för mobiltelefoner tas också upp, eftersom detta har visat sig vara avgörande för varför prototyping är viktigt; i många u-länder är telefonen den enda IT-prylen och detta innebär också att teknikvanan är låg.

Besvarandet av undersökningsfrågan sker i kapitel 2 som innehåller en blandning av alla kunskapsformerna nämnda ovan.

**U2. Ozlab, som nu är utformad som en webbtjänst, hur kan den köras ”lokalt”, speciellt mot mobiltelefoner? (Vad är systemkraven för att köra en demo eller ett test med Ozlab i fält där Internetuppkoppling inte finns?)**

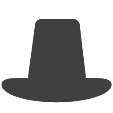
Här finns det två grenar att undersöka: hur de olika webbläsarna fungerar på mobiltelefonerna och vilka olika typer av anslutningsmöjligheter till en Ozlab-server de har.

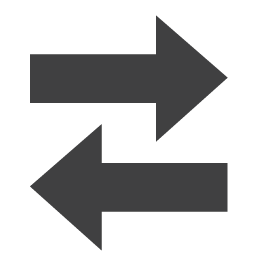
För första grenen utförs enkla funktionstester för att ta reda på möjligheten att sätt upp en lokal testmiljö. Vidare, eftersom jag vill besvara hur det går att köra Ozlab-systemet mot mobiltelefoner är det ytterst viktigt att ta reda på vilka mobiltelefoner som klarar av att köra systemet. Utan den kunskapen kan inte denna undersökningsfråga besvaras. För att ta reda på minimikraven för en mobiltelefon har analys och tester utförts med äldre och nyare versioner av mobiltelefoner mot Ozlab-systemet.

**U3. Vad är för- och nackdelar med den webbaserade Ozlab-tekniken mot mobiltelefoner?**

Förutom vad som kommer bli uppenbart från ovanstående undersökning så kommer det här arbetet innehålla test under fältliknande förhållanden – det innebär att jag anpassar Ozlab-systemet som därmed blir en prototyp för hur Ozlab ska vara utformad i framtiden (för att inte förvilla kommer jag inte att kalla prototypingverktyget Ozlab för prototyp). Här behöver man observera mina testdeltagare som i sin tur själva behöver testdeltagare. Jag kommer behöva undersöka både hur wizarden och hur de som wizarden interagerar med upplever testsituationen; se figur 1. Figur 1 visar hur observatören (till vänster), observerar wizarden (mitten) som i sin tur observerar testpersonen. Hur tillförlitliga sådana här resultat kan bli genom ett enkelt användartest kan förstås diskuteras (Rubin & Chisnell 2008; se diskussion om validitet och reliabilitet i Patel & Davidsson 2011).

Vidare kommer undersökningsfråga 2 gällande kraven för mobiltelefoner vara en viktig faktor vid besvarandet av denna fråga.

****



Figur 1, Observation av Wizard (i mitten) och testperson (till höger).

## 1.5.3 Etiska överväganden

Vetenskapsrådets tillhandahåller fyra forskningsetiska principer som skall följas när forsknings utförs: informationskravet, samtyckeskravet, konfidentialitetskravet och nyttjandekravet (Vetenskapsrådet 2012). Samtliga dessa krav uppfylldes vid testsessionerna.

För att uppfylla informationskravet informerades samtliga testdeltagare, innan testet påbörjades, att deras deltagande var helt frivilligt och att de kunde avbryta testet när de ville. Innan testsessionen startades talades även om i vilket syfte testet utfördes.

För att uppnå samtyckeskravet frågades samtliga testdeltagare om lov att använda deras svar på intervjufrågorna i studien. Se Bilaga 1 för utformningen av samtyckesblanketten.

Konfidentialitetskravet uppfylldes genom att testet var helt anonymt och inga personuppgifter kommer att lagras någon annanstans än på det papper som testdeltagarna signerat.

Nyttjandekravet uppfylldes genom att inga personuppgifter används vid analys och rapportering i denna uppsats. Dessutom används ingen insamlingsdata utanför denna uppsats.

# 2.0 Teoretiska utgångspunkter och perspektiv

För att ge en så bred bakgrund som möjligt till Wizard of Oz och Ozlab inledes detta avsnitt med en förklaring till prototyping och olika typer av prototyper. Därefter förklaras vad Wizard of Oz är och en systembeskrivning till Ozlab ges. Efter detta går avsnittet igenom olika fältstudier där mobiltelefonen ligger i fokus. För att avsluta med varför man ska använda sig av Wizard of Oz och även de vilka komplikationer som kan kommas att uppstå med fältstudier i utvecklingsländer.

## 2.1 Prototyping

En central del vid prototyping är att det är generiskt. Det innebär vid utveckling av prototyper så är det möjligt att ändra och justera prototypen ifall nya idéer dyker upp. Vid prototyping ger det en mer överskådlig bild av det man vill utveckla – det underlättar att få något att ”ta” på. Utveckling av en programvara kan vara väldigt komplex, det underlättar därför att ”skissa” på hur systemet kommer att bete sig vid interaktion av användare. Det blir lättare att identifiera olika typer av problem som kan uppstå under utveckling av ett system eller programvara (Zaki Warfel 2007, s.4ff). I *Handbook of Human-Computer Interaction*, 2:a upplagan, kan man läsa följande:

”Prototypes are widely recognized to be a core means of exploring and expressing designs for interactive computer artifacts. It is common practice to build prototypes in order to represent different states of an evolving design and to explore options. However, since interactive systems are complex, it may be difficult or impossible to create prototypes of a whole design in the formative stages of a project. Choosing the right kind of more focused prototype to build is an art in itself, and communicating its limited purposes to its various audiences is a critical aspect of its use.” (Houde & Hill, 1997 s.367).

För att prototyping ska vara användbart och generera idéer och krav behöver man alltså kommunicera med olika ”audiences”. En målgrupp som särskilt framhålls av vissa författare är de kommande användarna av programvaran som utvecklas:

”Stakeholder participation, eg. by end-users, has been considered critical for the successful development and implementation of information systems by a number of different researchers ([...]) and has also been considered good and a natural maturity of the ISD process.” (Larsson 2006, s.143).

Enligt kapitlet om ”Prototyping Tools and Techniques” i *The Human-Computer Interaction Handbook* så kan man karaktärisera prototypingtekniker utefter fyra olika mått: Representation, Precision, Interactivity och Evolution (Beaudouin-Lafon & Mackay 2003, s.1007). Nedan följer en förklaring till de fyra begreppen och i avsnitt 2.4 ges en förklaring till varför vissa av begreppen är applicerbara på WOz-metodologin och därmed också på Ozlab.

**Representation (Offline & online)**

Beroende på vilket syfte prototypen har, anser Beaudouin-Lafon och Mackay att det finns två olika former av Representation: offline- och onlineprototyp.

*Offlineprototypen*, som hörs på namnet, kräver ingen dator för att utveckla prototypen utan prototypen skapas med penna och papper. Det som utmärker en offlineprototyp är att den är billig och snabb att tillverka, utförs oftast i ett tidigt skede i utvecklingen och när den har använts så kasseras den (Beaudouin-Lafon & Mackay 2003,s.1007).

*Onlineprototypen* behöver nödvändigtvis inte vara tillgänglig via webben. Men den skapas med hjälp av dator för att ge egenskaper som interaktion, animation och gränssnitt. Det är fullt möjligt att även i offlineprototyper skapa interaktion. Det tar längre tid att utveckla en onlineprototyp eftersom det oftast krävs programmering för att skapa de olika egenskaperna till prototypen, vilket även leder till en högre kostnad. Beaudouin-Lafon och Mackay anser att det är effektivast att använda sig av dessa prototyper i ett senare skede i utvecklingen. De nämner att det är möjligt att få fram (skapa) olika strategier för hur interaktionen ska vara i offlineprototyper (ibid., s.1008f).

Däremot skriver Nilsson och Siponen att det går att skapa interaktion med pappersbaserade prototyper, när användaren simulerar en interaktion så skiftar man papper (Nilsson & Siponen 2006, s.2). Buxton skriver också om interaktiva pappers pototyper (Buxton 2007, s.371f).

Beaudouin-Lafon och Mackay nämner att programmerare föredrar programmerade prototyper eftersom de inte vill slösa tid på att skapa något som inte genererar någon kod. Men de konstaterar att de ännu inte hittat något som kan bevisa att det är en nackdel att börja använda sig av offlineprototyper tidigt i utvecklingsfasen, då de är billiga och snabba att utveckla.

**Precision** refererar till detaljrikedomen i prototypen. ”Prototypes force designers to *show* the interaction” (Beaudouin-Lafon & Mackay 2003, s.1008). Det kanske viktigaste är hur användaren navigerar runt för att hitta det hen söker. Men hur navigationsträden och hjälptexter ser ut kan vara mer eller mindre viktigt. Beroende på hur detaljrik och hur nära verkligheten prototypen är så skapar det en mer eller mindre trovärdighet på hur den faktiska slutprodukten kommer att se ut (ibid., s.1008).

Siponen och Nilsson beskriver att det finns en indelning av dessa prototyper, ”High- and Low-fidelity”. Man kan även dra en direkt parallell till representations prototyp, där Siponen och Nilsson talar om ”Implemeted automaticity”: High-fidelity hör till vad som Beaudouin-lafon och Mackay beskriver som onlineprototyp och Low-fidelity som offlineprototyp (Siponen & Nilsson 2006, s.10). Beaudouin-Lafon och Mackay föredrar att enbart använda termen precision för att de tycker att den refererar till prototypen själv och inte till dess relation till ett ännu inte färdig definierat system:

”The terms low-fidelity and high-fidelity prototypes are often used in the literature. We prefer the term precision because it refers to the content of the prototype itself, not its relationship to the final, as-yet-undefined system.” (s.1008).

**Interactivity**

”A critical role for an interactive system prototype is to illustrate how the user will interact with the system” (s.1009). Begreppet ”interactivity” beskriver till vilken grad användaren kan interagerar med prototypen. Som jag nämner ovan finns det inga hinder för offlineprototyper att skapa interaktivitet till användaren. Beaudouin-Lafon och Mackayanvänder sig av tre olika grader av interaktivitet; ”fixed prototypes”, ”fixed-path prototypes” och ”open prototypes”.

Till **fixed prototypes** tillhör prototyper där användaren inte alls kan interagera med, till exempel videoklipp. **Fixed-path prototypes** stödjer viss interaktion med användaren, det kan till exempel vara pappersprototyper. **Open prototypes** skall stödja hög grad av interaktion, så pass bra att det ska likna det tänkta systemet men samtidigt kommer det att vara vissa begränsningar (Beaudouin-Lafon & Mackay 2003, s.1008f).  
Det kan tilläggas att Siponen och Nilsson inför begreppet ”Perceived automaticity” är varken videoklipp eller pappersprototyper möjliggör till interaktion.

**Evolution**

”Prototypes have a different life spans” (Beaudouin-Lafon & Mackay 2003, s.1009). Vad de menar med fullfjädrade evolutionprototyper är att de är en del av systemet och utvecklas iterativt. Prototypen är det faktiska systemet under utvecklingsfasen men under tiden ändras då systemet ändras. Som Beaudouin-lafon och Mackayskriver: ”the prototypes are both representations of the final system and the final system itself” (Beaudouin-Lafon & Mackay 2003,s.1009).

I mitt arbete är prototyping dessutom metoden som används, eftersom det system som uppsatsen handlar om, se kap 1, är att betrakta som ett system under utveckling – alltså en prototyp! Närmast handlar det då om evolutionär prototyping, medan Wizard of Oz-tester använder sig av prototyper som inte vidareutvecklas i sig själva.

## 2.2 Wizard of Oz (WOz)

Som nämns tidigare i avsnitt 1.1 är WOz en teknik där testpersonen tror att hen interagrerar med ett system men egentligen sker interaktionen med en testledare kallad "wizard".

Bill Buxton berättar om flera exempel där WOz-tekniken används i praktiken: Redan 1971 beprövades WOz-tekniken genom att använda det på en flygplats till en självbetjänings flygbiljettsautomat. Systemet för att kunna genomföra ett köp för passageraren var inte färdigställt. För att kunna betjäna passageraren med en biljett mot betalning fick de låtas utåt sett att systemet var färdigställt. Det som faktiskt hände var att wizarden, som satt dold, matade in passagerarens uppgifter i terminalen manuellt för att sedan kunna ge biljetten till passageraren (Buxton 2007, s.241).

Ytterligare kanske ett ännu mer välkänt exempel på när WOz-tekniken användes är: Datorn som klarar av instruktioner från användaren. I början av 80-talet provades detta genom att använda sig av en mikrofon där användaren gav instruktioner till datorn. Användaren skulle sedan tala om för datorn vad som skulle göras och sedan satt wizarden och lyssnade av vad användaren sa och följde de instruktioner som gavs (Buxton 2007, s.243).

Dessa fall är typexempel där WOz-tekniken passar väl in. Båda exemplen var före sin tid och istället för att utveckla systemen utan vetskapen om att det blir användbart kunde de samla in användbarhets-data och testa teknik som inte var utvecklad.

Det var emellertid inte förrän 1983 som tekniken fick sitt namn, då kallat "OZ paradigm" av John Kelley. Tekniken användes under utvecklingen av en kalenderapplikation som skulle tillåta vardagligt språk vid inmatning av möten (1984, s.27f). Kelley använde sig av WOz tekniken på två olika tillvägagångssätt. Det första sättet simulerades systemet i sin helhet och i det andra används ett iterativ tillvägagångssätt efter varje testdeltagare. Efter varje testdeltagare i den iterativa metoden uppdaterades systemet utifrån deltagarens inmatningsstruktur. Det kunde på så vis vidareutveckla och förbättra systemet med verklig användbarhetsdata.

Vad som började som en experimentuppställning på 70-talet har nu blivit ett verktyg som gör att man kan utföra upprepade experiment. Nedan följer beskrivning av två olika verktyg som använder WOz-tekniken för att kunna utföra tester mot mobiltelefoner, framförallt mobiltelefoner för Android.

**A Wizard of Oz Tool for Android**

Ett verktyg anpassat för mobiltelefoner som kör operativsystemet Android har utvecklats vid Carnegie Mellon University Silicon. Verktyget ger möjligheten att skapa digitala prototyper eller använda sig av inskannade pappers-prototyper. Prototyperna skapas i en extern mjukvara som finns tillgänglig på dator. I mjukvaran är det möjligt att skapa prototyper med flera tillstånd i. Den tillhandahåller även vissa interaktiva objekt vad de kallar för "widgets": textrutor, radioknappar, checkboxar och knappar. När prototypen är klar och redo för testning, genererar mjukvaran en mapp med allt innehåll till prototypen. Den i sin tur måste manuellt överföras till mobiltelefonen. Under ett test körs prototypen på både mobiltelefon och dator. Wizarden använder sig av datorn och ser det aktuella tillståndet på mobiltelefonen och kan när som helst tvinga fram byte av scen (Linnell et al., 2012, s.67f).

**WozARd**

WozARd är ett verktyg för WOz-tester på Android-enheter: mobiltelefoner, surfplattor och glasögon. Utvecklarna koncentrerar sig på att använda verktyget för prototyping med Augmented Reality (AR) i användargränssnittet. En stor olikhet jämfört med det tidigare nämnda verktyget är att allt sker via två android-enheter som trådlöst, WiFi eller Bluetooth, kommunicerar med varandra (Alce et. al 2013, s.601).   
“It aims at offering a set of tools that help the test leader control the visual, tactile and auditive output that is presented to the test participant. Additionally, it is suitable for using in an augmented reality environment where images are overlaid on the phone’s camera view or on glasses” (Alce et. al 2013, Abstract).

Nästa avsnitt handlar om ett generiskt WOz-system som har utvecklats på Karlstad universitet.

## 2.3 Ozlab systembeskrivning

Ozlab är en mjukvara för skapa prototyper och genomföra interaktionstester med användare. Mjukvaran är utvecklad vid Karlstads universitet. Ozlab finns i två olika versioner som förklaras nedan; den första versionen används inte längre. Idén med Ozlab är att man simulera all interaktion från användaren och behöver därför inte programmera något vid skapandet av en prototyp. Ozlab anammar metodologin WOz där "wizarden" kallas för testledare (TL) och användaren som interagerar med prototypen för testperson (TP). Testpersonen tror sig interagera med ett system, men egentligen är det testledaren som utför all interaktion och navigering.

I Ozlab finns det ett verktyg, Skalbyggaren, där det är möjligt att bygga prototyper, i Ozlab kallar man dessa prototyper för skal. I skalet kan flera olika delar av ett system illustreras. Varje del i skalet kallas för scen. Det är i scenen man illustrerar systemets gränssnitt och lägger till objekt som kan vara i form av bilder, textrutor, knappar, listor, länkar m.m. För varje objekt är det fullt möjligt att lägga till flera olika beteenden som ett objekt ska ha. I den senaste versionen finns det totalt nio beteenden som ett objekt kan tilldelas, dessa beteenden förklaras nedan i kapitel 2.3.4.

När skalet är färdigställt och redo för en testsession ansluter TP mot skalet. Vid anslutningen använder TL och TP varsin dator som ansluter mot varandra. Det är vid testsessionen som det är möjligt för TL att följa, på sin skärm, TP-interaktioner mot skalet och därefter initiera en händelse efter det att TP interagerar med skalet. I den senaste versionen av Ozlab är det också möjligt för TP att använda sig av en mobiltelefon eller surfplatta för att ansluta sig mot testsessionen; även detta förklaras mer nedan.

### 2.3.1 Första versionen av Ozlab

Första versionen av Ozlab kom 2001 det är en mjukvara som måste vara installerad på båda datorerna som TL och TP ska använda. För att skapa skal och för att upprätta kommunikation mellan de båda datorerna krävs extern mjukvara för att kunna genomför det (Pettersson 2003, s.169).

Eftersom mjukvaran var tvungen att vara installerad på datorn gav detta begränsningar till att enbart utföra tester mot datorer. Vid detta tillfälle fanns det heller ingen möjlighet att utföra tester på mobiltelefoner.

Denna version av Ozlab används inte längre idag utan det är enbart det senaste webbaserade Ozlab som används.

### 2.3.2 Nuvarande version av Ozlab

Den nuvarande versionen av Ozlab kom under hösten 2013 och är en webbaserad tjänst. Allt som har med dagens Ozlab att göra utförs via en webbläsare. TL och TP ansluter till systemet via en webbläsare. Detta ger nya möjligheter att utföra tester mot till exempel mobiltelefoner och surfplattor, då de kan ansluta till systemet med en webbläsare. Däremot krävs det att enheterna som ska använda sig av systemet har tillgång till internet eller sitter inom samma nätverk som systemet.

### 2.3.3 Sencha Ext JS

Ozlab-systemet använder JavaScript-ramverket Ext JS som gör det möjligt att bygga interaktiva webbapplikationer. Ramverket är utvecklat av Sencha. Ozlab använder sig för närvarande av version 4.2. Ytterligare versioner har släppts av Ext JS där Sencha påpekar att i den senaste beta-versionen 5.0 ska ramverket vara mer anpassningsbart för mobiltelefoner då det ska klara av att hantera touch-gester (Sencha 2014).

### 2.3.4 Beteenden i nuvarande Ozlab

Som nämns ovan finns det totalt nio beteenden i den senaste versionen av Ozlab. Varav två av beteenden är av samma slag med enda skillnaden att det antingen är möjligt för TL att utföra beteendet eller att det är möjligt för TP att utföra det. Nedan följer beskrivning till varje beteende:

**GotoScen:** Används för att byta scen vid interaktion på objektet från TP. Detta underlättar även för TL då denne slipper att klicka fram nästa scen.

**MakeObjectSnap:** Objekt med detta beteende kommer att, om TP eller TL byter position på ett objekt, justera sin position till en den punkt som placeras på scenen.

**ObjectMovableForTL:** Gör objekten flyttbara enbart för TL. Objektet kommer att vara synligt för både TL och TP när det flyttas.

**ObjectMovableForTP:** Likadant som ovanstående med enda skillnaden att det endast är möjligt för TP.

**ObjectInvisibleMoveForTL:** Snarlikt de båda ovanstående med skillnaden från att objektet inte är synligt när det flyttas. Detta beteende gör det enbart möjligt för TL att flytta objektet. När objektet flyttas är det först synbart för TP när TL placerar objektet.

**ObjectInvisibleMoveForTP:** Skillnaderna från ovanstående är att det enbart är möjligt för TP att flytta objektet och objektet är först synligt för TL när TP placerar objektet.

**OpenLink:** Öppnar en länk i ett nytt fönster.

**SendAudio:** Spelar upp ett ljudklipp för TP eller TL vid interaktion mot objektet (ljudet aktiveras oavsett om det är TP eller TL som interagerar med objektet). Interagerar TP med objektet spelas ljudet upp för TL och vice versa. Beteendet måste även tilldelas en ljudfil.

**Vibrate:** Beteende för mobiltelefoner, vid interaktion från TL/TP mot objektet kommer mobiltelefonen att vibrera.

Det är möjligt att kombinera beteendena ObjectMovableForTL/TP och ObjectInvisibleMoveForTL/TP. De enda kombinationerna som inte är möjliga är ObjectMovableForTL med ObjectInvisibleMoveForTL och ObjectMovableForTP med ObjectInvisibleMoveForTP.

### 2.3.5 Funktioner enbart för TL i nuvarande Ozlab

Nedan förklaras de funktioner som TL kan använda sig av under en testsession. Det är dessa funktioner som ger TL kontrollen över systemet om några problem under sessionen skulle uppkomma. Samtliga funktioner har en aktivering av funktionen och en avaktivering.

**Start/Stop Session:** Hörs på namnet, startar eller stoppar en testsession. Är sessionen inte startad visas en laddningsskärm för TP.

**Show/Hide wait screen:** Visar och döljer laddningskärm för TP.

**Block/Unblock Input:** Med Block input är det inte möjligt för TP att interagera med objekt, däremot är det fullt möjligt för TP att klicka på prototypen och TL ser var TP klickade. Unblock input gör det möjligt för TP att interagera med objekt.

**Freeze/Unfreeze:** Låser TP skärm och ingen interaktion är möjlig för TP. Unfreeze tillgängliggör interaktion återigen för TP.

**Lock/Unlock movable objects:** Objekt som tilldelats ett beteende med flytt egenskaper för objekt för TP blockeras och kan inte förflyttas av TP under tiden Lock movable objects är aktiverat. Unlock movable objects gör att TP återigen kan flytta objektet.

## 2.4 Prototyping med Wizard of Oz

Detta avsnitt behandlar de fyra olika måtten som tas upp i avsnitt 2.1 och hur de är relaterade till WOz.

Representationsvärdet för WOz är uppenbarligen ”onlineprototyp”, i synnerhet gäller det Ozlabprototyper, men Ozlabprototyperna delar en del drag med det som utmärker offlineprotoyper, alltså att de är snabba och relativt billiga att skapa.

Precisionsvärdet är upptill varje skalbyggare att bestämma. Det är fullt möjligt att göra enklare skisser för layouten i Ozlab (t.o.m. att använda inscannade handritade GUI-fönster och -element) men det är även möjligt att skapa mer avancerade skal, vilket naturligtvis tar mer tid att utföra.

Interactivitygraden för WOz-prototyper ligger långt ifrån fixed-prototypes. Ozlabskal ligger betydligt närmare fixed-path och open prototypes med betoning på open prototypes, eftersom användaren i hög grad kan interagera med skalen (tack vare TLs agerande ”bakom scenen”). Det är med hjälp av användaren det är möjligt att samla in data på hur de använder ett system.

Evolutionsvärdet är lika med noll. En Ozlab prototyp blir aldrig en färdig produkt utan prototypen byggs i Ozlab-systemet och kan inte återanvändas för att skapa den faktiska programvaran som prototypen är designad för. Däremot går det att använda sig av de grafiska elementen som skapas under processens gång.

Om man jämför med Siponen och Nilsson, som gjorde sin analys i relation till det gamla Ozlab-systemet som Siponen programmerade, så är det endast tre dimensioner man ska mäta prototyperna längs. I basen finns det två axlar ”implementerad automatik” respektive ”upplevd automatik”. WOz har generellt sett inte mycket automatik, det är testledaren som ska sköta det. Medan det i Ozlab finns viss implementerad automatik som att gå till en specifik scen så finns däremot ingen naturligt språkigenkänning som nämns i avsnitt 2.2.

”It is possible to categories an Ozlab prototype as being of low- or high-fidelity depending on if it is the user's or the developer's view of fidelity that is used. Interaction shells also share many of the advantages and disadvantages ascribed to both low- and high-fidelity prototypes.” (Siponen & Nilsson 2006, s.11).

I deras tredje dimension ”precision" gäller samma som i avsnitt 2.1 (de refererar till Beaudouin-Lafon & Mackay för denna dimension).

Dessa båda sätt att karaktärisera prototyper belyser alltså det speciella i Wizard of Oz-prototyper och därmed också de speciella förutsättningarna för ett WOz-verktyg.

## 2.5 Mobiltjänster i utvecklingsländer

I en studie som gjordes under våren 2010 i Sydafrika av Gitau, Marsden och Donner kom man fram till att många av de tjänster som vi (som inte lever i utvecklingsländer) idag använder skapar problem för dem som inte har en så kallad smartphone. I deras studie använde de sig av åtta testpersoner som tidigare inte har använt sig av internet (några visste vad internet är), de hade heller aldrig tidigare använt sig av en dator. Alla testpersoner ägde varsin mobiltelefon som klarade av GPRS och hade WAP 2.0 som webbstandard.   
Testpersonerna kunde få åtkomst till internet med sina telefoner, men det krävdes en hel del av användaren att ställa in rätt inställningar för att få åtkomst till internet, då de var tvungna att ha kännedom om telefonens specifikationer vid inmatning av inställningar hos operatören. En del av studien var att testpersonerna skulle skaffa en emailadress. De motiverar det med: ”It is the cornerstone of an online presence, and without it, other services, such as Twitter, can be nearly impossible to access”. Trots att de använde sig av Opera Mini, som webbläsare då många av de hemsidor de besökte inte stödjer WML, hade de problem med att skaffa en emailadress hos Google. När de först försökte skaffa ett Gmail-konto använde de sig av mobiltelefonernas standard webbläsare fick de felmeddelandet ”Want a Gmail account? Go to www.gmail.com on your computer”. De bad då testpersonerna att använda sig av Opera Mini för att kringgå problemet. För en av testpersonerna fungerade det fortfarande inte då denne fick problem med att fylla i autentiseringskoden för att gå vidare och fick då felmeddelandet:

”If your mobile does not correctly display the image below please login successfully on the desktop to enable your mobile login again. Enter the correct password above and then type the characters you see in the picture below” (Gitau et. al 2010, 2603ff).

Även om de flesta av testpersonerna klarade av att registrera sig för en gmail så var de tvungna att använda sig av en extern webbläsare (Opera Mini). Hade de inte fått information av testledarna att använda sig av Opera Mini så hade de antagligen aldrig kunnat skaffa sig ett gmail-konton eftersom de inte hade tillgång till en egen dator.

Gitau, Marsden och Donner avslutar sin studie med att tala om att allt fler kommer att använda sig av internet med enbart mobiltelefon. Och att det är därför viktigt att när man utvecklar en webbtjänst bör det vara möjligt att ha åtkomst till tjänsten med enbart mobiltelefon annars finns det risk för att många kommer förbli uteslutna från internet och dess möjligheter.

### 2.5.1 Mobiltjänst för bankärenden

I Kenya finns en mobiltjänst, M-Pesa, som ger möjlighet för användarna att överföra pengar sinsemellan med hjälp av textmeddelanden (SMS). M-Pesa lanserades 2007 och sedan dess har det tillkommit flera nya funktioner där det är möjligt för användarna att utföra bankärenden. Användarna kan betala sina räkningar och företag kan betala ut löner via M-Pesa (Gebregziabher & Krauss 2014, s.282f).

I en artikel i Time berättar grundaren till Safaricom, som har lanserat mobiltjänsten, varför M-Pesa är framgångsrikt:

”Some businessmen want to deposit too little for them to stand for a long time in a queue. Others are shabbily dressed or illiterate or they forget their signatures. It's just easier to operate with M-Pesa.” (Time 2011-01-31, s.2).

### 2.5.2 Läs- och skrivsvårigheter

Två studier som presenterades under Mobile For Development (M4D) konferensen tidigare i år talar om vikten av läs- och skrivförståelse vid användning av mobiltjänster i utvecklingsländer, framförallt i Afrika då båda studierna är utförda där. Leah Farmer och Mark Boots (2014 s.208f) rapporterar att analfabetism når upp till 38% i subsahariska Afrikanska länder medan den andra rapporten av Mame Awa Ndiaye och Moustafa Zouinar (2014 s.272f) fokuserar enbart mot Senegal där analfabetism är 41% bland vuxna.

Analfabetism skapar naturligtvis problem vid användning av mobiltelefoner och deras tjänster. Många viktiga funktioner, som till exempel spara kontakter eller skicka/läsa sms, kan inte utnyttjas till fullo av dem som är analfabeter.

Ndiaye’s och Zouinar’s studie undersöker hur analfabeter använder sig av mobiltelefoner i Afrikanska länder där analfabetismen är hög, de koncentrerar sig på Senegal. Det var totalt 20 deltagare i studien där författarna har delat upp dem i tre olika grader (kategorier) av analfabetism: analfabeter (A, 7st), semi-analfabeter (B, 10 st) och avancerade semi-analfabeter (C, 3 st). Kategori A har inte gått i skola och kan inte läsa eller skriva. Den andra kategorin kan läsa och skriva enkla fraser i ett eller två språk. Den sista kategorin kan läsa och skriva mindre avancerade fraser upp till fyra språk (Ndiaye & Zouinar 2014, s.274f).

De olika kategorierna använder mobiltelefonerna på olika vis. Till exempel använder kategori A mobiltelefonen till mest för att ringa, de använder inte SMS och kan inte lägga till kontakter utan hjälp. Kategori B kan läsa enklare SMS men behöver hjälp med andra funktioner i telefonen. Sista kategorin kan läsa och skriva SMS men tar hjälp med rättstavning vid vissa fall, författarna skriver även att denna kategori använder mobiltelefonerna som en icke analfabet (Ndiaye & Zouinar 2014, s.275).

I Farmer och Boots studie har de undersökt fyra stora barriärer som man behöver kringgå för att kunna utföra ett framgångsrikt M4D projekt i utvecklingsländer:

"(1) literacy, accessibility and technology limitations of end-users, (2) challenges with Mobile Network Operators (MNO), (3) unreliable delivery and connectivity, and (4) lack of human design knowledge and best practices." (Farmer & Boots 2014, p. 208)

Den första barriären att komma runt är att förstå att alla inte är läs- eller skrivkunniga. I studien talar de om att under de senaste fem åren har 87% av de M4D-projektet som körts fokuserat på textbaserade projekt med tjänster som SMS, USSD och webb. De berättar om projekt där fokus låg på att använda sig av SMS-tjänster var svarsresponsen betydligt lägre vid användning av SMS-tjänst kontra taltjänst. I ett av fallen behövde 80% av testdeltagarna hjälp med att läsa och förstå vad som stod i meddelandet.

De berättar även i sin inledning:

”Mobile For Development (M4D) is an emerging field with the potential to address social and public-sector challenges in health, education, agriculture, project monitoring, and governance, by offering direct two-way communication with constituents at scale through pervasive mobile phone access.” (Farmer & Boots 2014, s.208).

### 2.5.3 Human–computer interaction (HCI) for development

I en artikel från 2013 talar den kände HCI- och ICT4D-experten Kentaro Toyama om vikten att vara på plats och veta om förutsättningarna där utvecklingen eller användartester ska utföras.

"First, anyone who lacks direct experience with a particular environment must be careful that any advising they do is limited to what they truly know. There is nothing wrong with making methodological suggestions, contributing to brainstorming, or acting as a sounding board, but final decisions about design or the interpretation of data should align with those who have deep experience with target users." (2013, s.66).

Toyama berättar om en kollega som bad flera erfarna HCI-utövare tillsammans med Toyama om att få synpunkter på ett HCI-projekt som en av dennes studenter utförde i Afrika. Projektets syfte var att ta fram en bra lösning för att hålla koll på försäljning och lagerhantering till lokala butiksägare i ett Afrikanskt land. Det fanns redan ett planerat tillvägagångssätt för projektet: butiksägaren skulle använda sig av SMS för att rapportera sin försäljning och lagerstatus. Det kollegan ville ha synpunkter på var hur projektet kunde öka användbarheten och köra effektivare tester. Toyama och hans kollegor ställde frågor om butiksägarna och om mobilanvändning i landet. De pekar på studier som visar att insamling av data via SMS är mindre precis än via samtal. Som även nämns i avsnittet 2.5.2 behöver en stor del av användarna hjälp med att besvara ett SMS på grund av läs- och skrivsvårigheter. Ett förslag var att använda sig av en telefonist som samlar in data istället för att butiksägaren ska behöva använda SMS tjänsten.

"As we spoke, Norm’s shoulders started to slouch and a worried look appeared on his face. He acknowledged that it might make sense to have human operators field calls, but there was a catch: Voice calls require little design. That solution would eliminate the need for HCI. What, then, would become of Norm’s project?" (2013, s.67).

Det ska tilläggas att kollegan som bad om att få synpunkter hade varit på besök i landet. Men antagligen har denne inte varit på plats tillräckligt länge för att förstå problemen som SMS-tjänsten bidrog till. Eller som Toyama skriver i artikeln:

"If we’re really concerned about development, the proof is in whether we’re willing at times to sacrifice HCI objectives for the larger cause." (2013, s.66).

### 2.5.4 Vilka typer av mobiler dominerar i u-länder?

I en artikel från 2011 konstateras det att den mest vanliga mobiltelefon i Afrika är en feature phone från Nokia, Nokia 1100, med 50 miljoner sålda enheter (Economist 2011). Feature phone beskrivs av Wikipedia:

”A feature phone is a class of [mobile phone](http://en.wikipedia.org/wiki/Mobile_phone" \o "Mobile phone); the term is typically used as a [retronym](http://en.wikipedia.org/wiki/Retronym" \o "Retronym) to describe low-end mobile phones which are limited in capabilities in contrast to a modern [smartphone](http://en.wikipedia.org/wiki/Smartphone" \o "Smartphone). Feature phones typically provide voice calling and text messaging functionality, in addition to basic [multimedia](http://en.wikipedia.org/wiki/Multimedia" \o "Multimedia) and [internet](http://en.wikipedia.org/wiki/Internet" \o "Internet) capabilities, and other services offered by the user's wireless service provider.” (Wikipedia 2014).

Även om det är över tre år sedan artikeln skrevs finns det fortfarande starka indikationer på att dessa typer av mobiltelefoner fortfarande dominerar, eftersom ungefär 1.2 miljarder människor saknar tillgång till elektricitet världen över och därav 550 miljoner i Afrika (The World Bank 2013). Farmer och Boots skriver att utan tillgång till ström används mobiltelefonerna enbart vid de tillfällena användaren behöver använda den (2014, s.211).

I en artikel skriven av Jon Russel 2011 talar han om en undersökning gjord av Vision Mobile som tar upp försäljning av smartphones kontra feature phones. Enligt undersökningen utgjorde smartphones 27% av totalförsäljningen världen över. I Afrika och mellanöstern såldes totalt 209 miljoner enheter varav 82% var feature phones (The Next Web 2011).

GSMA Mobile for Development (2013) har utfört sammanställningar av nätoperatörers nättäckning och den teknologi (2G, 3G och 4G) operatörerna erbjuder till användarna i utvecklingsländer. De länder som jag har tittat närmare på med hjälp av GSMA’s sammanställning är Ghana, Gambia, Kenya, Nigera, Uganda och Sydafrika. Tyvärr fanns ingen data för Senegal; i avsnitt 2.5.2 beskrivs ett exempel från Senegal.

I samtliga sex länder erbjuds stor täckningsgrad rörande 2G. Däremot saknar flera av dessa länder operatörer som erbjuder 3G eller 4G. I de länder som erbjuder 3G, Ghana, Nigeria, Uganda och Sydafrika, är täckningsgraden låg hos de två förstnämnda länderna, enbart få platser har tillgång till 3G. Uganda har flera punkter som har täckning för 3G, däremot är punkterna väldigt lokala och inte alls i den utsträckning som 2G. I Sydafrika har en operatör även 4G, men å andra sidan är täckningsgraden väldigt begränsad och inskränker sig till få områden, framför allt storstäder.

För att kunna utföra användartester med Ozlab med så kallade feature phones behövs först en utvärdering av Ozlabs systemkrav som krävs av enheten för att kunna utföra ett test. Utifrån de resultat som ges kan en prototyp utformas och utföra testsessioner mot prototypen. Testsessionen kommer att fokusera på att samla in data angående testledarens reflektioner kring testsessionen. Detta eftersom det är testledaren som kommer att samla in användbarhetsdata i vidare studier.

## 2.6 Varför Wizard of Oz?

Pettersson och Nilsson utförde en studie som la fokus på att vid ett så tidigt skede använda sig av användartester innan programmeringen påbörjas. De inleder sin rapportering av studien med:

"Frequent testing of developing software can certainly increase the usability in the program. However, as we found in a case study, the method seems to continuously introduce changed or new requirements which in turn results in more complex code and thereby more errors." (2009, s.1).

De fokuserar alltså kodkvalitet, inte användbarhet, men trots det pekar deras resultat på att användartester bör göras tidigt i utformningen av ett system, De påpekar även vikten av att involvera innehållsexperten som testperson, inte bara slutanvändare:

"In the HCI expert's (i.e. Wizard´s) opinion, most of the new requirements would have been possible to spot if the content expert had been included in the pre-testing, which could have been done without the wizard setting up special test scenarios for content experts." (2009, s.2)

En annan studie utförd för att anpassa WOz-tekniken mot mobiltelefon talar även den om att det är viktigt att veta kraven innan utvecklingen av systemet påbörjas. De skriver: "The Wizard of Oz (WOz) technique […] is very useful in situations where the development of a system is expensive and it is hard to know beforehand how users will behave." (Ardito et. al 2009, s.344).

De nämner även att WOz-tekniken är speciellt anpassbar vid testning av nya interaktionstekniker för att förbigå de interaktionsbegränsningar en enhet kan ha, i detta fall för mobiltelefoner (ibid.).

Ytterligare en studie som fokuserar på att ta fram en applikation för WOz-prototyping till android-telefoner nämner att WOz är användbart vid användartester i ett tidigt skede i utvecklingen. (Alce 2013, s.600, om arbete vid Sony Mobile i Lund.) Applikationen beskrivs mer utförligt i avsnitt 2.2.

Pettersson och Nilsson skriver att när nya funktioner tillkommer under utvecklingsfasen blir det svårare för programmeraren att hålla struktur i koden (2009 s.2). Detta problem skulle kunna undvikas (eller i alla fall minimeras) om alla systemkrav är fördefinierade och inte tillkommer under utvecklingsfasen.

Studier visar på att WOz är en bra teknik för att samla in data för teorier inom HCI (se till exempel Dahlbäck et. al 1993, s.4f). Även som nämns i avsnitt 2.3 kan WOz-tekniken anpassas till system som inte finns tillgängliga idag, det gäller ”bara” att komma på ett sätta att kunna demonstrera det man vill visa.

I fallen med SMS-exemplen i avsnitt 2.5.2 skulle det vara möjligt att i ett tidigare skede visa att utfallen skulle bli högre om taltjänster skulle ha använts istället för SMS-tjänster. Som Toyama påpekar är det viktigt att känna till förutsättningarna på den plats system ska utföras, se avsnitt 2.5.3. Men gör man inte det är det viktigt att testa tidigt och med ”lätta” metoder så att man kan pröva sig fram för att se hur väl olika typer av lösningar står sig.

Med Ozlab-systemet är det lätt att göra ändringar, det krävs ingen programmering som nämnts tidigare. Fördelen är att man enkelt kan byta ut olika komponenter och prova sig fram utan att behöva skriva om funktioner, man kan istället enkelt illustrera den nya funktionens syfte.

## 2.7 Implikationer för min studie

Detta kapitel har visat att WOz-tekniken är användbar i tidig utveckling av system då det kan ge värdefull data på hur användaren använder systemet utan att behöva börja med programmeringen till systemet. Som framgår från Toyamas artikel är det värdefullt att spendera tid på den plats där användartester och senare användingen ska ske. Eftersom det har framgått att det råder brist på uppkoppling till internet i utvecklingsländer kommer nästa kapitel handla om hur det går att anpassa Ozlab till en internetlös miljö. Även tester med Ozlab mot mobiltelefoner kommer att utföras då mobiltelefoner är den absolut vanligaste enhet som används i utvecklingsländer.

# 3.0 Genomförande och Resultat

Detta avsnitt utreder vilka typer av mobiltelefoner som klarar av att använda sig av Ozlab. Det tar även upp hur man kan sätta upp en portabel miljö av Ozlab-systemet.

## 3.1 Utredning av mobil kompabilitet i Ozlab

För att svara på undersökningsfråga 3 behövs tester av olika typer av mobiltelefoner mot Ozlab-systemet att utföras. Detta för att ta reda på om det är möjligt att utföra testsessioner med äldre mobiltelefoner mot Ozlab-systemet. Under utredningen användes flera olika typer av mobiltelefoner både äldre versioner och nyare; även mediaspelare har använts. Eftersom utredningen ska ta reda på hur långt tillbaka det är möjligt att gå i mobilversioner för att kunna utföra en testsession behövs även nyare mobiltelefoner.

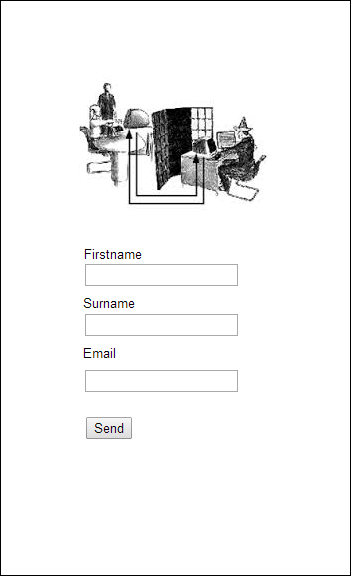
De enheter som användes under utredningen listas i tabell 1.

**Tabell 1 De enheter som användes under utredningen**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Modell | Operativsystem | Webbläsare | År | WiFi | Touch |
| Samsung Galaxy S4 | Andriod 4.2.2 | Google Chrome v35 | 2013 | Ja | Ja |
| HTC Hero | Android 2.2 |  | 2009 | Ja | Ja |
| iPhone 5s | iOS 7.0 | Safari 7.0 | 2013 | Ja | Ja |
| Nokia E71 | Symbian 9.2 | - | 2008 | Ja | Nej |
| iPad | iOS 7.0 | Safari 7.0 | 2012 | Ja | Ja |
| Sony Ericsson K300i | - | - | 2004 | Nej | Nej |
| Sony Ericsson W380i | - | - | 2007 | Nej | Nej |
| iPod Touch | iOS 4.2.1 | Safari 5.0.2 | 2008 | Ja | Ja |
| Samsung GT-i9023 | Andriod 4.1.2 | Google Chrome v34 | 2010 | Ja | Ja |

Dessa mobiltelefoner valdes då de flesta av dem fanns tillgängliga vid användarlabbet medan en del var i min ägo (tyvärr fanns ingen Nokia 1100 att tillgå). Mobiltelefonerna befinner sig i ett intervall mellan åren 2004-2013. De har olika typer av operativsystem och skärmstorlekar.

För att kunna ta reda på vilka av enheterna som klarar av att utföra en testsession med Ozlab har ett skal skapats. Skalet innehåller ett formulär med fyra fält som ger TP möjlighet att fylla i text, en knapp som TP ska kunna interagera med samt en bild visas för testpersonen; se figur 2. Bilden har även tilldelas beteendet **ObjectMovableForTP/TL** för att kontrollera om det är möjligt att flytta ett objekt. Beteendet byts även ut till **ObjectInvisibleMoveForTP/TL** eftersom det inte går att kombinera dessa beteenden.



Figur 2 Testskal för mobiltelefon

### 3.1.1 Krav för godkänd enhet

För att kunna bestämma om en enhet är kompatibel med Ozlab-systemet behöver mobiltelefonerna uppfylla vissa krav under testsession för att bli godkända. De krav som listas nedan har tagits fram för att om en enhet uppfyller dessa är det fullt möjligt att genomföra ett test mot den enheten. Däremot är det inte sagt att det inte kan uppstå andra problem. Testet omfattar även om det är möjligt att flytta bilden och om det syns ifall testledaren flyttar ett objekt, men detta anses inte som ett krav för att enheten ska bli godkänd men är av intresse vid prototyptillverkning.

* K1: Ansluta till testsessionen.
* K2: Mata in text i textrutorna.
* K3: Klicka på knapp, ska ge notifikation åt TL.
* K4: Bild ska visas.

### 3.1.2 Testgenomförande

All testning skedde vid användarlabbet på Karlstads Universitet. Det första som gjordes var att sätta upp en testsession i Ozlab där mobiltelefonerna skulle ansluta till. Därefter skulle alla enheter ansluta till [ozlabservern](http://ozlab-hhk.kau.se) via enhetens webbläsare för att sedan ansluta till Ozlab testsessionen. Vid utfall av att enheten inte klarade av att ansluta till testsessionen räknades testet som avslutat för den enheten. Klarade enheten att ansluta till testsession utfördes uppgifter för att testa om kraven K2, K3 och K4 gick att uppfylla.

Alla enheter som användes under testets gång använde sig av Karlstads universitets WiFi-nätverk förutom de båda modellerna från Sony Ericsson, de använde sig av 2G nätverk för att ansluta mot Ozlab med Telia som operatör.

### 3.1.3 Testresultat

Totalt klarade 4 av 9 mobiltelefoner att ansluta mot Ozlab och genomföra samtliga krav för att anses som godkänd; se tabell 2. Däremot klarade ingen av dessa enheter att dra och flytta objekt, för båda beteendena men när objektet flyttades från testledaren syntes objektet och placerades på rätt position. De enheter som hade tillgång till WiFi men som inte kunde ansluta till Ozlab fastnade vid laddningssidan av Ozlab; se figur 3. De två enheterna som använde sig av 2G visade aldrig startsidan för Ozlab där det ges möjlighet att ansluta till testsessionen.



Figur 3 Ozlab laddningssida

**Tabell 2 resultat från testgenomförande**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Modell** | **K1** | **K2** | **K3** | **K4** |
| Samsung Galaxy S4 | Ja | Ja | Ja | Ja |
| HTC Hero | Nej | - | - | - |
| iPhone 5s | Ja | Ja | Ja | Ja |
| Nokia E71 | Nej | - | - | - |
| iPad | Ja | Ja | Ja | Ja |
| K300i | Nej | - | - | - |
| W380i | Nej | - | - | - |
| iPod Touch | Nej | - | - | - |
| Samsung GT-i9023 | Ja | Ja | Ja | Ja |

### 3.1.4 Sencha EXT JS - webbläsarstöd

Enligt utvecklarnas hemsida sträcker sig stödet för olika typer av webbläsare långt bak. De listar webbläsare som har stöd och den äldsta versionen som stödjer Sencha EXT JS (Sencha 2014):

* Internet Explorer 6+
* Firefox 3.6+ (PC, Mac)
* Safari 4+
* Chrome 10+
* Opera 11+ (PC, Mac)

### 3.1.5 Websockets

Vidare har systemet undersökts och vid diskussion med utvecklarna av systemet har de berättat att Ozlab använder websockets för att skapa kommunikation mellan de båda enheterna. Alla webbläsare stödjer inte tekniken websockets, enligt caniuse.com som har sammanställt vilka webbläsare som har stöd eller inte för websockets. Av de telefoner som testats och inte klarat av att ansluta till Ozlab har en webbläsare som inte stödjer denna teknik. Noterbart är att webbläsaren Opera Mini saknar stöd helt och hållet (Can I use 2014), Opera Mini används i studien av Gitau (2010) som presenteras i avsnitt 2.5.

Av de mobiltelefoner som klarade av att ansluta till Ozlab användes Google Chrome och Safari som webbläsare. Där den äldsta versionen av Google Chrome var 34, för Safari var den äldsta och den enda som användes 7.0. Google Chrome stödjer delvis websockets från version 4.0 och full support vid version 14.0. Safari klarar av websockets vid version 5.0 men har fullt stöd vid 6.0. Dock klarade inte en av enheterna att ansluta trots att den använde sig av Safari 5.0.2.

## 3.2 Installation av portabel ozlabmiljö

För att kunna utföra fälttester i uppkopplingslösa miljöer behövs en portabel version av Ozlab som inte är beroende av strömförsörjning, internetuppkoppling och inte heller av ett nätverk. Idén med den portabla miljön är relativt enkel:

Systemet ligger för närvarande på en av Karlstad universitets servrar och för att få åtkomst till systemet måste enheterna ha tillgång antingen till universitetets nätverk eller till internet. Tanken är då att använda sig av likadan serverprogramvara som universitets server använder sig av och installera det på en bärbar dator sam implementera Ozlab lokalt på datorn. För att göra det möjligt för mobiltelefonen att ansluta till den lokala versionen av Ozlab på datorn är det tänkt att datorns trådlösa nätverk ska delas ut. Tidigare studier har visat att detta är möjligt:

“The experimenters also ran a Momento server on the laptop and configured the laptop for peer-to-peer wireless networking. In this way, the mobile devices could connect directly to the server running on the laptop […]” (Carter, Mankoff, Heer 2007, s.8)

Denna lösning gör att Ozlab inte blir internet- eller nätverks-beroende och inte heller ström-beroende, förutom när den bärbara datorn behöver laddas.

### 3.2.1 Idén med den portabla ozlabmiljön

Idén är att den bärbara datorn ska agera likt servern på universitetet med undantaget för nätverks begränsningar. Den bärbara datorn ska dela ut sitt nätverk som mobiltelefonen sedan kan ansluta till och därefter ansluta till Ozlab-systemet; se figur 4 .



Figur 4 Illustration av portabel Ozlabmiljö

### 3.2.2 Installation av virtuellmiljö och implementering av Ozlab-systemet

För att få en korrekt installation av systemet togs hjälp av Jonas Kihlström som jobbar på Motillo. De har utvecklat den senaste versionen av Ozlab och har därför kunskap om vad som krävs för att utföra installationen.

Datorn och mjukvaran som användes för den portabla miljön listas nedan:

* VMware Player version 6.0.1 (används för att skapa en virtuell miljö)
* Windows Server 2012 professional
* Dator: HP EliteBook 8460p

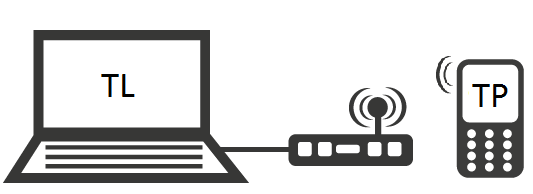
Då Ozlab idag ligger på en server med Windows Server 2012 krävs det att datorn som den portabla miljön ska implementeras på också har det operativsystemet. Det första som installerades var programvaran för att skapa den virtuella miljön, VMware Player. Sedan installerades Windows-servern i den virtuella miljön. Därefter lämnades datorn till Kihlström som skulle implementera Ozlab-systemet i den virtuella miljön.

### 3.2.3 Testning av den portabla miljön

När allt var installerat och datorn var tillbaka efter implementation behövdes ett test utföras för att se om idén med den portabla miljön var möjlig. Testet skulle kontrollera om det var möjligt att ansluta mobiltelefonen mot den lokala versionen av Ozlab som var installerad på datorn. Tyvärr möttes testet av komplikationer innan det ens hade påbörjats. Det gick nämligen inte att använda datorns WiFi i den virtuella miljön. Jag har försökt att få till att datorn ska kunna dela ut sitt nätverk. Dock fick inte ens IT-avdelning på universitet det att fungera.

### 3.2.4 Nödlösning p.g.a. WiFi-komplikationen

Eftersom det inte gick att använda sig av det trådlösa nätverkskortet i den virtuella miljön, gick det inte att slippa använda sig av en extern åtkomstpunkt för enheterna. I samråd med prof. Pettersson användes en extern router istället där routern fungerar som en länk mellan den virtuella miljön och mobiltelefonen; se figur 5. Tyvärr blir den portabla miljön lite mindre portabel då den blir beroende av strömförsörjning till routern.



Figur 5 Illustration av portabel Ozlabmiljö med router

### 3.2.5 Testning av den routerbaserade lösningen

Testningen av miljön gjordes hemmavid efter det att miljön och uppsättning av systemet var gjort. Prof. Pettersson hade använt den routerbaserade nödlösningen i demo av nya Ozlab vid M4D 2014 med hjälp av iPhone 5 samt den i tabell 1 listade Samsung GT-i9023. Jag använde mig av min personliga mobiltelefon (Samsung Galaxy S4) och trådlösrouter (Netgear wnr3500). Jag anslöt den bärbara datorn med nätverkssladd till routern och kopplade mobiltelefonen mot det trådlösa nätverket. Nätverket hade inte tillgång till internet.

Därefter provade jag att starta ett test i Ozlab-systemet och ansluta telefonen mot datorns IP-adress för att se om det gick att genomföra tester likadant som mot systemet som ligger på universitets server. Allt gick utan komplikationer.

## 3.3 Uppsättning av litet fälttest i labbmiljö

I avsnitt 2.2 förklaras att en observation av testledare och testpersoner som utför tester med Ozlab mot mobiltelefoner är nödvändig för att kunna besvara undersökningsfråga 3. Under testsessionen kommer Observatören (jag) observera testledaren (TL) samt testpersonen (TP). Dock kommer det vara svårt för observatören att observera båda deltagarna och därför kommer det att användas två TL vid varje testsession. Där den icke deltagande TL observerar TP och Observatören observerar den deltagande TL. TL med erfarenhet behövdes för att kunna få relevant data om deras upplevelse under testkörning. Totalt användes fyra TL med tidigare erfarenhet av Ozlab, dock hade ingen av dem kört tester mot mobiltelefoner och inte heller tester i fält. Optimalt hade naturligtvis varit att använda sig av TL med tidigare erfarenheter av testkörningar mot mobiltelefoner (emulerat på pekskärm i gamla Ozlabsystemet) och även fälttester. Tyvärr kunde jag inte få tag i någon med dessa erfarenheter men ansåg att resultatet inte skulle påverkas särskilt mycket av detta – de mobillösningar jag kom att testa saknar motsvarighet i gamla Ozlab.

Eftersom det inte är möjligt att testa Ozlab mot äldre mobiltelefoner, se kapitel 3.0, bestämdes, vid rådgivning med Pettersson, att illustrera en äldre mobiltelefon i Ozlab. Prototypen skulle innehålla en skärmyta, en illustration av menyknappar och en bild för textinmatning. Enkla uppgifter åt TP skulle användas då fokus låg på att få reda på hur TL upplevde testerna.

Rubriken på detta avsnitt kan verka vara självmotsägande, då testet ej utfördes i fält utan i en kontrollerad miljö. Men uppsättningen är den samma som skulle användas vid (internetlöst) fältarbete även om lokalen som användes var användarlabbet på universitetet, som förklaras nedan i avsnitt 3.3.5. Internetlöshet var enkelt att simulera genom att stänga av TL-datorns wifi. Det är dessutom enklare för oerfarna TL att genomföra testsessioner samt samla in data i en mer kontrollerad och familjärmiljö. Försöket var ju inte avsett att testa TL-deltagarnas fältarbeteskapacitet utan enbart att testa Ozlabs möjligheter att simulera enkla mobiltelefoner.

### 3.3.1 Design av prototyp

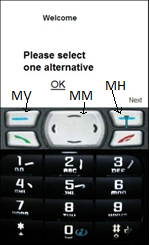
**Layout 1**

Prototypen är uppdelad i två delar, den övre delen illustrerar skärmytan för mobiltelefonen och den nedre illustrerar mobiltelefonens tangenter; se figur 6.

Hjälptexter användes för att ge TP tips om hur denne skulle gå vidare. Det gjordes för att undvika kommunikation och avbrott under testsessionen i den mån som gick.   
Prototypen är anpassad för att passa skärmupplösningen för Samsung Galaxy S4. Valet av mobiltelefon gjordes på grund av storleken på skärmen som gav plats till prototypen utan att påverka designen, den ska trots allt illustrera en äldre mobiltelefon. Men fram för allt med vetskapen om att Ozlab fungerar mot mobiltelefonen, se även här kapitel 3.0.

Menyknapparna användes för att kunna ge TP navigeringsmöjligheter. Knappen i mitten (MM) användes för att kunna välja bland de olika alternativen i prototypen. För att kunna gå vidare eller gå tillbaka i prototypen är TP tvungna att använda sig av de menyknapparna högst upptill höger (MH) respektive vänster (MV). Inga förvalda funktioner i Ozlab användes vid navigering, detta för att ge TL mer uppgifter under testsessionen.

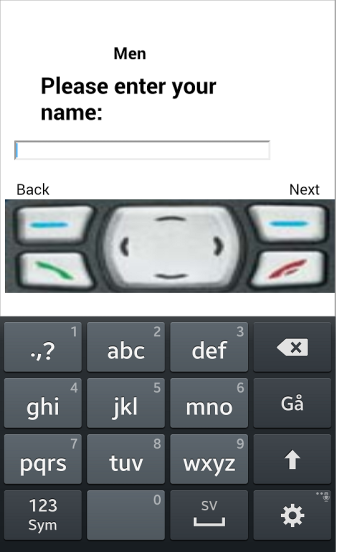
Vid textinmatning användes en knappsats med totalt tolv knappar, var av nio av knapparna gav möjligheten för TP att mata in bokstäver.



Figur 6 Ozlab-prototyp av äldre mobiltelefon.  
(Prototypen visas som en skärmbild på en smartphone.)

**Layout 2**

Den enda skillnaden från layout 1 är att tangentbordet är utbytt mot tangentbordet som finns i mobiltelefonen det vill säga smartphonen som testerna ska utföras mot; se figur 7. Idén med att använda sig av tangentbordet för mobiltelefonen var att ge TP en förbättrad upplevelse och samtidigt underlätta för TL vid inmatning av TP namn.



Figur 7 Ozlab prototyp med Android-tangentbord.

### 3.3.2 Pilottest – val av layout

Ett pilottest utfördes för att välja bland de två olika layouterna men samtidigt för att kontrollera att alla delar i testet kunde utföras som planerat i testplanen. Det framgick ganska snabbt att layout 1 var bättre anpassad att utföra de riktiga testerna på.

**Layout 1**

TL tyckte det var svårt att mata in TP namn, eftersom det var besvärligt att följa varje interaktion från TP.

Layout 2

Gav inga indikationer till TL när TP trycker på knapparna, däremot var det enklare för TP att fylla i sitt namn då inmatningen av en bokstav sker med motsvarande knapp som TP trycker på.

Ytterligare en nackdel var att få fram tangentbordet åt TP, då det inte finns någon möjlighet för TL att trycka fram tangentbordet åt TP och samtidigt markera textrutan så att TP kan mata in sitt namn. För att kunna genomföra detta är TP tvingad att interagera med ett tangentbod som verkligen finns i TP-enheten. Vilket förstör idén med testet eftersom prototypen ska illustrera en äldre mobiltelefon utan touchskärm. Eftersom knapparna skulle illustrera fysiska knappar.

### 3.3.3 Pilottest åtgärder

Det första som kom på tal efter pilottestet var att underlätta för TL när TPs namn skulle matas in. Det bestämdes då att en mikrofon skulle användas i testrummet och TPs namn skulle sägas tydligt av observatören. För att få in namnet på ett bra sätt, utan att ge TP föraningar om att det finns någon bakom spegelglaset som styr, skulle observatören berätta TPs namn innan testet påbörjades i stil med "Då så, [namn], du kan nu börja med att...".

### 3.3.4 testplan

Idén med testet var att observera TL för att senare få deras synpunkter på hur det var att utföra en testsession med det webbteknikbaserade Ozlab mot mobiltelefoner. Samtidigt var det en möjlighet att försöka ge TP en känsla av att utföra ett test på en äldre mobiltelefon. Uppgifterna som TP skulle utföra är enligt mig mindre avancerade uppgifter. Nedan listas uppgifterna i turordning:

* Ta sig vidare från en uppmaning: ”Please select one alternative, OK”. För att ta sig vidare skulle TP klicka på menyknappen MH (se figur 6 för ”MH”).
* Välja ett av tre alternativ: Female, Male, I do not want to answer. Valet görs med hjälp av menyknappen MM och bekräftas med MH.
* Fylla i sitt namn i textrutan. Inmatningen av namnet görs med tangentbordet. Bekräftas med MH.

När namnet har bekräftas tackas TP av prototypen. Prototypens delar och flöde visas i bilaga 2.

TP får inte använda sig av skärmytan för att interagera med prototypen utan de är tvungen att använda sig av menyknapparna och knappsatsen för interaktion med prototypen. För att förhindra att TP skulle av misstag trycka på skärmen och ange ett alternativ blockerades inmatning för TP med Ozlab-systemet's alternativ Block input, som förklaras i avsnitt 2.3.5.

Innan en testsession startades skulle en genomgång med TL utföras och även ställas intervjufrågor om deras erfarenhet av Ozlab. Genomgången planerades att vara en kort sammanfattning av vad som ska hända när TP interagerar med prototypen. Även ett pilottest genomfördes för att ge TL en uppfattning om vad de behöver göra.

Frågor till TL före testsessionerna:

* **FTL 1**: Har du tidigare använt dig av Ozlab, om ja har du tidigare agerat testledare?
* **FTL** **2**: Har du tidigare använt dig av Ozlab, eller något annat prototyping-verktyg, för att utföra tester mot mobiltelefoner?
* **FTL 3**: Har du tidigare erfarenheter av fälttester med prototyping-verktyg?

Uppgifter för TL, i turordning:

* Dölja och flytta uppmaningsrutan åt TP.
* Markera ett av alternativen i turordning när TP klickar på MM.
* Byta till motsvarande scen till alternativet TP valt efter det att TP bekräftat valet.
* Mata in en bokstav åt gången när TP interagerar med tangentbordet, sudda ut en bokstav om TP klickar på MV.
* När TP fyllt i sitt namn och klickat på MH ska TL byta till slutscen.

Efter det att genomgången med TL är gjord ska TP informeras om testet, i vilket syftet testet används till och vilken typ av data som kommer att samlas in. De blir sedan tillfrågade att skriva under medgivande (Rubin & Chisnell 2008, s.173). I medgivandet står det vilken typ av data som kommer att samlas in och att TP har blivit informerad om testets syfte, se bilaga 1. Här tydliggörs även att det inte TP i sig som testats utan prototypen (Rubin & Chisnell 2008:161).

Sist innan själva testutförande startas ställs en fråga till TP:

* **FTP1**: Har du tidigare använt dig av en äldre mobiltelefon med tangentlayout T9 som hade tillgång till internet?

Som tidigare nämnts är det TL som har störst fokus, men det är därmed inte sagt att TP reaktioner och beteende kring testet är irrelevant. Under hela testsessionen antecknades reaktioner och beteende från både TL och TP. Observationen av TP gjordes av den TL som inte kontrollerar ozlabskalet och observationen av TL görs av Observatören.

När testet är slutfört ställs efterfrågor till TP:

* **FTP2**: Hur upplevde du testet?
* **FTP3**: Upplevdes något som svårt under testets gång?
* **FTP4**: Förstod du testuppgifterna?
* **FTP5**: Förstod du att det var en prototyp som testet utfördes mot?
* **FTP6**: Trodde du att det var en prototyp som var programmerad? Trodde du att det var någon bakom ridån och styrde prototypen?

Efter att alla TP har utfört testet tackades TP för sitt deltagande och sedan fördes en diskussion med TL utifrån deras upplevelser och reaktioner kring testet.

### 3.3.5 Testmiljö

Alla tester utfördes i Karlstads universitets användarlabb. Även om testmiljön inte påminner om extrem uppkopplingslös miljö eller fältliknande miljö finns det fördelar med att använda sig av användarlabbet. Det finns nämligen en spegelvägg som avgränsar rummen, kontrollrummet där TL sitter och testrummet där TP befinner sig. Detta ger möjlighet för Observatören att observera både TL och TP, i den mån som Observatören hinner med. En annan fördel är att användarlabbet är placerat på campus vilket underlättar att få tag i flera TP. Se figur 8 och 9 för testmiljön.



Figur 8 Kontrollrummet vid användarlabbet Figur 9 Testrummet vid användarlabbet  
Fotograf: Malin Wik Fotograf: Malin Wik

## 3.4 Testresultat

Nedan presenteras den data som samlades in från testsessionerna, data presenteras i tabeller.

### 3.4.1 Sammanställning av intervjufrågor för TL

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Testledare** | **FTL 1** | **FTL 2** | **FTL 3** | **FTL 4** |
| **TL 1** | Ja, vid två kurstillfällen vid Karlstads universitet. Har använt mig av den äldre versionen av Ozlab och den webbaserade. Jag var testledare under båda kurstillfällena. Aldrig utfört tester mot mobiltelefoner. | Nej | Nej | Nej |
| **TL 2** | Ja, vid två kurstillfällen. Har använt mig av båda versionerna av Ozlab. Jag var testledare vid det senaste kurstillfället, alltså var jag testledare med den senaste versionen av Ozlab. Aldrig utfört tester mot mobiltelefoner. | Nej | Nej | Nej |
| **TL 3** | Ja, under två kurser vid Karlstads universitet. Har använt mig av den äldre och senaste versionen av Ozlab. Var testledare med den senaste versionen av Ozlab. Aldrig utfört tester mot mobiltelefoner. | Nej | Nej | Nej |
| **TL 4** | Ja under två kurser vid Karlstads universitet. Har bara varit testledare en gång, vid den senaste versionen av Ozlab. Aldrig utfört tester mot mobiltelefoner. | Nej | Nej | Nej |

### 3.4.2 observation av TL

|  |  |
| --- | --- |
| **Testledare** | **Kommentar** |
| **TL 1** | Blev uppjagad/stressad i början av testsessionen, hade svårt att hänga med när TP klickade på skärmen. Efter ett tag in i testet kom TL 1 mer in i sin roll som testledare och stressen minskade. Däremot var TL 1 upptagen med att hålla koll på vart TP klickade nästa gång. |
| **TL 2** | Liknande för TL 1, stressad i början och kunde efter tag slappna av mer då TL 2 förstod att det var denne som styrde över testsessionen. |
| **TL 3** | Märkbart stressad, mindre än föregående TL's. Kändes mer bekväm i sin roll som TL. |
| **TL 4** | Stressad, svårt att hänga med i vart TP tryckte när denne skulle mata in sitt namn. Blev allt mer stressad när TP ville radera det som TP 4 matade in i textrutan. |

### 3.4.3 Diskussion med TL

Samtliga TL ansåg att det var svårt att hänga med när TP interagera med prototypen, de kände sig stressade. Eftersom de inte visste alls vart nästa interaktion skulle komma, då det enbart syns när TP trycker på skärmen. Det underlättade för samtliga TL att de visste vilka steg som skulle utföras av TP och även att de hade fått göra en testkörning. Ingen av TL ansåg att de hann med att hålla koll på loggen som visar interaktionen från TP. De hade fullt upp med att hålla koll på vart TP skulle trycka nästa gång.

Alla TL ansåg att det var svårt att hålla koll på valet som TP skulle göra när de väljer kön. Eftersom TL enbart ser första markeringen av radiobuttons men inte andra gången då alternativet byts. Men samtidigt konstaterar de att de enkelt kunde hålla koll på vilket alternativ som TP bör ha valt genom att de visste namnet och kunde se TP genom spegelväggen.

Vid inmatning av namn från TP tyckte alla TL att om det inte hade fått reda på namnet i förväg hade det varit betydligt svårare att fylla i rätt namn. Även fast de fick namnet i förväg, var det inte alltid lätt att komma ihåg namnet och eller hur namnet stavas.

En av TL konstaterar att med mer flera testkörningar och mer övning med Ozlab skulle det bli enklare att hålla koll på flera saker samtidigt. Samtidigt som de andra TL anser att de skulle behöva en observatör som kan hjälpa till att hålla koll på antalet klick som TP gör och samtidigt föra anteckningar.

### 3.4.4 Sammanställning av intervjufrågor för TP

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Testperson** | **FTP 1** | **FTP 2** | **FTP 3** | **FTP 4** | **FTP 5** | **FTP 6** | **Övrigt** |
| **TP 1** | Ja, utan internet. | Ingen respons när jag tryckte. Upplevde det som segt. | Det var svårt att skriva, träffa rätt tangent. | Ja, inga konstigheter | Nu när du säger det, annars trodde jag nog enbart att det var väldigt segt. | Det trodde jag inte. Trodde det att den programmerad faktiskt. | - |
| **TP 2** | Ja, utan internet. | Långsamt, långa laddningstider. | Nej. | Ja, det tycker jag nog. | Ja, eftersom det inte var så snyggt och det kändes segt. | Oj, det hade jag ingen aning om. | - |
| **TP 3** | Ja, utan internet. | Det kändes föråldrat och gammalt. | Nej, inget var svårt. Tycker dock att det tog lång tid att gå vidare. | Ja | Ja | Nej det trodde jag inte. Trodde det var jag själv som matade in texten och valen. | - |
| **TP 4** | Ja, utan internet. | Det var långsamt, hade förväntat mig snabbare övergångar efter valen. Fick heller ingen feedback när jag tryckte på skärmen. | Nej, inget direkt. | Ja | Ja, jag trodde det enbart var en bild som byttes fram och tillbaka | Hade mina föraningar, eftersom jag inte trodde något var programmerat utan enbart var bilder. Trodde dock inte att ni kunde se vart jag klickade. | - |
| **TP 5** | Ja, utan internet. | Tyckte att testet flöt på bra, tog lite lång tid att komma in i T9 igen. | Nej, var tillräckligt tydligt för att kunna utföra uppgifterna. | Ja | Kanske, men tänkte det nog mer som en halvfärdig produkt som skulle kunna vidareutvecklas. | Nej det trodde jag inte alls. Var helt säker på att det var jag som tryckte fel. | Trodde det var möjligt att fylla i med "snabb" T9. Där jag enbart behövde trycka på de knapparna som mitt namn innehåller och så skulle namnet dyka upp. |

### 3.4.5 Observation av TP

|  |  |
| --- | --- |
| **Testperson** | **Kommentar** |
| **TP 1** | Otålig, ville fort fram under testsessionen. Klickade frekvent på mobiltelefonen. Klickade rätt i samtliga steg. |
| **TP 2** | Svårt att använda sig av knappsatsen för inmatning av sitt namn. Kan ha berott på att denne inte såg vad som stod på knapparna. Klickade rätt i samtliga steg. |
| **TP 3** | Klickade upprepade gånger på fel knapp för att ta sig vidare. Tryckte ganska hårt vid interaktion. |
| **TP 4** | Otålig, klickade frekvent. Klickade rätt i samtliga steg. |
| **TP 5** | Klickade oerhört snabbt vid inmatning av sitt namn. Klickade rätt i samtliga steg. |

# 4.0 Analys & Slutresultat

Som anges i avsnitt 1.2 så är syftet med uppsatsen att undersöka och ta reda på om det är möjligt att använda sig av det webbteknikbaserade Ozlab-systemet i fält med mobiltelefonersom TP-enhet .

Speciellt har data insamlats för att kunna besvara tre frågor, nämligen:

U1: Vad är viktigt att tänka på när ”Wizard of Oz”-metoden används i utveckling för tjänster till mobiltelefoner i utvecklingsländer?

U2: Ozlab, som nu är utformad som en webbtjänst, hur kan den köras ”lokalt”, speciellt mot mobiltelefoner? (Vad är systemkraven för att köra en demo eller ett test med Ozlab i fält där Internetuppkoppling inte finns?)

U3: Vad är för- och nackdelar med den webbaserade Ozlab-tekniken mot mobiltelefoner?

För den första frågan så har metoden i huvudsak varit att analysera tidigare studier. I avsnitt 4.1 görs en sammanvävning av det som tagits upp i kapitel 2 och 3, som berör denna fråga.

För den andra frågan har tillvägagångssättet varit att ta fram en portabel Ozlab-miljö för att på så sätt åskådliggöra systemkraven för mobiltelefoner som ska kunna använda sig av Ozlab tagits. Detta beskrevs i avsnitt 3.1 och 3.2 och i avsnitt 4.2 nedan analyseras resultaten av detta arbete. Analysen är ganska rakt på eftersom det var rent det praktiska arbetet med installation och funktionskontroll.

Slutligen för fråga 3 har jag kunnat nyttja svaren på de ovanstående båda frågorna men också gjort användartester för att få ”mjuka” data som också är en del av svaret på den tredje frågan. Testsessioner mot mobiltelefoner har observerats och data i form av användarupplevelse samlats in, från både testledare och testperson. Resultat av testsessionerna återfinns i avsnitt 3.3 och i här i avsnitt 4.3 förs en diskussion kring dessa resultat för att kunna argumentera för de för- och nackdelar som listas i slutet av det avsnittet.

## 4.1 U1: Vad är viktigt att tänka på när ”Wizard of Oz”-metoden används i utveckling för tjänster till mobiltelefoner i utvecklingsländer?

Vad som genomsyrar teorigenomgången är vikten av prototyping och tester vid systemutveckling överlag, helst ska detta ske i ett så tidigt skede som möjligt. Det är viktigt att komma ihåg varför prototyper och tester ska användas. Tester ska inte utföras bara för att det anses bra eller som "best practice" utan därför att det viktigt att kunna involvera slutanvändaren och förstå bakgrunden till problemet som ska lösas. Det är ett sätt för att kunna utveckla en mer optimal lösning; se till exempel avsnitt 2.6.

Med hjälp av WOz-metoden går det att enkelt testa nya idéer och interaktionstekniker för att sedan kunna utesluta eller förbättra dem. enkelheten kommer sig av att ingen programmering behövs i praktiken för att framställa en WOz-prototyp (undantag finns naturligtvis; se avsnitt 2.2). Det går då att prova en tänkt lösning på ett problem och tidigt få information av slutanvändaren hur de upplever lösningen.

För att kunna använda sig av WOz-metoden behöver någon form av nätverks-kommunikation mellan verktyget och enheten upprättas för att kunna spegla vyn från mobiltelefonen. Då de typer av mobiltelefoner som cirkulerar i afrikanska utvecklingsländer är så kallade "feature phones", behövs ett sätt att kunna utföra tester mot dessa typer av mobiltelefoner. Ozlab saknar stöd för att utföra tester mot dessa typer av mobiltelefoner, men kan däremot illustrera en "feature phone", till exempel på en smartphone. Trots att det är möjligt, kan det skapa komplikationer då slutanvändaren kanske aldrig har använt sig en smartphone tidigare. Den insamlade datan över hur slutanvändaren upplever lösningen kan då vara avvikande från verkligheten ifall de hade fått utföra tester mot en verklig "feature phone". Samtliga testpersoner som genomförde testsessionen upplevde prototypen som långsam. Det kan påverka resultatet i en negativ form, men oavsett om testet hade utförts mot en "feature phone" hade stegen mellan de olika vyerna tagit minst lika lång tid om inte längre eftersom det är TL, testledaren, som styr övergångarna mellan vyerna. Det skulle vara möjligt att påskynda denna process om TL hade fått mer övning innan testsessionen utfördes.

Den portabla Ozlab-lösningen som är upprättad under arbetet med denna rapport kräver en strömkälla för att kunna utföra tester. Det är därför viktigt att veta var testsessionerna kommer att äga rum, att känna till platsen. Men även om strömkälla finns, bör en förståelse om problemet vara av största vikt. Toyama berättar i sin artikel om problemet med att utföra HCI-projekt bara för sakens skull; det är oerhört viktigt att veta om förhållanden och hur tekniken är bäst lämpad att användas på av slutanvändaren (2013, s.66).

I avsnitt 2.5.2 behandlas problematiken angående läs- och skrivsvårigheter. Flera av testpersonerna i de rapporter som Ndiaye och Zouinar presenterar behöver hjälp med att läsa/skriva mindre komplicerade texter. När man då ska utföra en prototyp kan det vara bra att ha detta i bakhuvudet, för att kunna genomföra bra tester. Med Ozlab är det möjligt att utföra små ändringar om detta skulle bli ett problem. Istället för att använda texter kan det vara möjligt att använda symboler. Vilket även det skulle kunna ge information och data på ifall de symboler man använder i prototypen representerar det som man vill.

**Slutsatser**

Det går inte att ge några direkta råd om vad som är viktigt att tänka på för användning av WOz-metoden vid utveckling av mobiltjänster. Det berör egentligen alla utvecklingsmetoder. Men eftersom nätverks-möljligheter är i stor utsträckning ett problem i utvecklingsländer bör detta funderas över innan tester utförs på den valda platsen med användning av Ozlab. Däremot är det viktigt att verkligen förstå problemet som ska lösas.

Innan tester ska utföras anser jag att det är viktigt att verkligen besöka platsen, förstå bakgrunden till problemet. Det är som Toyama berättar att det inte alltid är nödvändigt att använda sig av komplicerade lösningar, när den bästa lösningen kanske enbart är att använda sig av samtal. Men med det sagt går det även att utesluta vissa lösningar med WOz-tekniken relativt snabbt eftersom man verkligen får snabb feedback från slutanvändaren vid testsessioner.

Ytterligare något som är viktigt att tänka på är att vara väl förberedd som TL. Testsessionerna flyter på bättre och det ger användaren en bättre upplevelse. Vilket kan leda till bättre reslutat för projektet, eftersom slutanvändaren då kan ge bättre feedback istället för att påpeka att prototypen är långsam och eller att den innehåller designmissar.

Även om det är möjligt att utföra tester mot en illustrerad "feature phone", saknar den fortfarande viktiga aspekter från en faktiskt "feature phone". Som t.ex. feedback vid interaktion med mobiltelefonen i form av knapptryck. För att verkligen få en rättvis bild av prototypen kanske det behöver utvecklas ett WOz-verktyg som klarar av att utföra testsessioner mot s.k. "feature phones".

## 4.2 U2: Ozlab, som nu är utformad som en webbtjänst, hur kan den köras ”lokalt”, speciellt mot mobiltelefoner? (Vad är systemkraven för att köra en demo eller ett test med Ozlab i fält där Internetuppkoppling inte finns?)

Att göra Ozlab-systemet portabelt innebär att det är möjligt att utföra WOz-tester på platser som saknar tillgång till internetuppkoppling och/eller nätverkskopplingar till Karlstads universitet. Till exempel skulle det vara möjligt att utföra tester i utvecklingsländer där det ofta saknas bra internetförbindelser.

**Portabel Ozlab-miljö**

För att Ozlab-systemet ska kunna köras lokalt krävs det att man har en dator som klarar av att köra Windows Server 2012, då Ozlab använder funktionalitet som det operativsystemet stödjer. För att göra systemet portabelt behöver man en bärbar dator, vilket underlättar vid förflyttning. Självklart går det att använda sig av en stationär dator, men frågan är då om det går att kalla för portabelt.

Vill man inte installera över sitt nuvarande operativsystem går det att använda sig av en virtuell maskin för att installera miljön. Vid denna installation användes WMware Player version 6.0.1; den klarar av Windows Server 2012.

Vid den slutgiltiga uppsättningen av miljön krävdes en extern router vilket inte var planerat. Tidigare studier har visat att det ska vara möjligt att skapa ett WiFi-nätverk utan extern router, se avsnitt 3.2. Men efter flera försök att få nätverksdelningen att fungera och även med hjälp från Karlstads universitets IT-avdelning gick det inte att lösa.

**Vilka mobiltelefoner klarar av Ozlab**

För att kunna utföra tester mot mobiltelefoner krävs det att telefonen klarar av tekniken websockets. Vid de tester som utfördes klarade 4 av 9 enheter att ansluta mot Ozlab. Dessa enheter är från 2010 och framåt och använde sig av webbläsaren Google Chrome version 34 och uppåt eller Safari 7.0. Google Chrome och Safari har full support av websockets vid version 14.0 respektive 6.0. Däremot ska Safari klara av att använda sig av websockets redan vid version 5. Trots det var det en enhet som använde sig av Safari version 5.0.2 men som inte kunde ansluta till testsessionen.

Enheterna behöver även ha tillgång till WiFi då de måste kunna koppla upp sig mot den externa routern. Utan den specifikationen kan inte enheten kommunicera med den portabla Ozlab-miljön. Däremot är det möjligt att använda 3G eller liknande ifall Ozlab är tillgängligt publikt utåt.

**Slutsatser**

Det är möjligt att sätta upp en lokal miljö av Ozlab-systemet, däremot blir den inte fullt så portabel som det först var tänkt. Trots att uppsättningen kräver en router fungerar den utan komplikationer. Det är dock viktigt att komma ihåg att routern kräver en extern strömkälla för att enheter ska kunna ansluta till den lokala miljön.

Det var enbart de senare modellerna av enheterna som klarade av att utföra tester mot Ozlab-systemet. Detta beror troligtvis på att det enbart var dessa enheter som stödjer websockets, till skillnad från de äldre enheterna. Mobiltelefoner utan WiFi-möjligheter vid fälttester i utvecklingsländer bör räknas bort utifrån det som presenteras i kapitel 2.5.4 då dessa länder ofta saknar täckning för 3G.

Mobiltelefoner som ska användas vid testsessioner mot den portabla Ozlab-miljön bör ha fullt stöd för websockets. De behöver även ha tillgång till WiFi för att kunna koppla upp sig mot den externa routern som används i den portabla miljön.

## 4.3 U3: Vad är för- och nackdelar med den webbaserade Ozlab-tekniken mot mobiltelefoner?

Under de testsessioner som har genomförts under uppsatsarbetet har det kommit fram intressant data. Ozlab-systemet har blivit mindre plattformsberoende i och med att det är webbaserat, det är möjligt att utföra tester mot olika enheter. Men systemet saknar stöd för äldre mobiltelefoner eller så kallade "feature phones" på grund av den teknik som används vid utveckling av Ozlab, webbsockets.

Däremot är det möjligt att utforma och illustrera äldre mobiltelefoner i Ozlab-prototyper som tester körs mot. Men det ska dock påpekas att 2 av de 5 testpersonerna saknade feedback vid interaktion mot prototypen som användes i testsessionen.

**Intervjufrågor med TP**

Samtliga TP, totalt 5, hade någon gång tidigare använt sig av en äldre mobiltelefon med T9, däremot hade ingen haft tillgång till internet. Alla TP uppfattade prototypen som långsam och att den inte gav någon respons vid interaktion. En av TP tyckte att det var svårt att träffa rätt på tangenterna när denne skulle mata in sitt namn, annars ansåg alla TP att testet gick att förstå och hade inga svårigheter med att genomföra testet. 3 TP förstod direkt att det var en prototyp de 2 andra förstod det först efter att frågan var ställd. De 2 som inte först förstod att det var en prototyp trodde att det var en halvfärdig utvecklad produkt som var väldigt långsam. 4 av 5 TP trodde det var prototypen som skötte navigeringen och inmatningen av texter och hade ingen aning om att det satt en person som styrde och såg deras interaktioner med prototypen. Den TP som inte trodde att prototypen skötte interaktionen, hade föraningar om att det var någon annan person som skötte det, eftersom TP trodde det enbart var bilder och att inget var programmerat i prototypen. Som även nämns ovan sakande 2 av 5 testpersoner feedback vid interaktion med prototypen.

Utformningen av prototypen kan självklart förbättras. Till exempel skulle varje knapp som TP kunde interagera med tilldelas beteende Vibrate, som på så vis skulle kunna ge feedback att mobiltelefoner vibrerar. Vad gällande långsamheten under testet och att TP tryckte flertal gånger för att utföra samma uppgift, så beror troligtvis denna på oerfarenheten hos TL, vilket med mer övning skulle kunna förbättra upplevelsen för TP.

**Observation av TP**

Observationen skedde under testets gång av samtliga TP. Vad som märktes väl från observationen var att TP var otåliga när det gäller byte av scen och de klickade flera gånger för att utföra samma uppgift.

Detta berodde delvis på att TL inte var varm i kläderna och hade fullt upp med att inte göra fel. En ytterligare faktor, som nämnts tidigare, kan vara som en av TP nämnde var att de inte fick någon feedback när de interagerat med prototypen. TP kan då bli osäker på ifall denne verkligen tryckte på rätt knapp och eller träffade den ytan de ville interagera med. Även om prototypen skulle illustrera en äldre mobiltelefon ger fortfarande en äldre mobiltelefon någon form av feedback när man interagerar med knapparna på telefonen.

**Intervjufrågor och diskussion med TL**

Samtliga TL hade någon gång tidigare varit i kontakt med båda versionerna av Ozlab när de studerat vid Karlstads universitet. En av TL hade agerat testledare med båda versionerna av Ozlab de två andra hade varit testledare enbart med den senaste versionen av Ozlab. Ingen av TL hade tidigare använt sig av Ozlab eller något annat prototypingverktyg för att utföra tester mot mobiltelefoner. De hade heller inga erfarenheter från fälttester.

Trots att ingen av TL hade utfört testsessioner mot mobiltelefoner förut klarade de av att utföra en testsession utan större problem. Självklart hjälpte det att de hade använt sig av Ozlab tidigare och agerat TL. Tester mot mobiltelefoner jämfört med tester mot en dator skiljer sig inte speciellt mycket. Den stora skillnaden är att TL inte vet vart TP kommer att interagera kontra mot dator där TL kan följa TP's muspekare.

**Observation av TL**

Det märktes tydligt på samtliga TL att de blev stressade när testet kördes igång. Det var mycket att hålla reda på samtidigt som de var tvungna att veta var TP trycker på skärmen. Inte ens jag som observatör kunde hänga med i loggen när TP tryckte på skärmen. Längre in i testet märkts att TL blev lugnare när de vet om att det är de som styr testet. De kan sakta ner testsessionen och låta TP trycka en extra gång utan att förstöra testet.

Trots att samtliga TL inte var erfarna testledare gick testsessionen att utföra utan större förhinder. Bristen i att inte kunna veta var TP trycker har stor inverkan i större prototyper, i detta fallet fanns det endast ett fåtal interaktionsmöjligheter i prototypen för TP vilket underlättade för TL. Däremot i större och mer komplicerade prototyper kommer det vara viktigt att använda sig av mer erfarna TL än under denna testsession. En av detta tests TL menade att med mer övning skulle testsessionen flyta på bättre.

**Slutsatser**

Användning av Ozlab vid tester mot mobiltelefoner skiljer sig inte mycket från att utföra tester mot datorer ur testledarens perspektiv. Största skillnaden och en stor nackdel är att testledaren har svårt att följa var testpersonen befinner sig i prototypen. Det är avsevärt lättare för testledaren att förutspå var testpersonen kommer att interagera när denne kan följa en muspekare. Ytterligare en nackdel är att testledaren inte kan se vilken radiobutton som är markerad efter första interaktionen med knapparna, men denna bugg i Ozlab-systemet måste väl lätt gå att åtgärda.

Från testpersonens enhet går det inte att dra och flytta objekt. Däremot syns det när testledaren flyttar objekten i testpersonen vy.

Att kunna använda sig av flera olika typer av mobila enheter med det webbaserade Ozlab måste ses som en stryka hos Ozlab. De andra verktygen som denna uppsats berörde klarade enbart att utföra tester mot Android-enheter. På så vis kan man utföra tester på olika typer av enheter, med samma prototyp, utan att behöva anpassa prototypen till de olika enheterna.

Trots att Ozlab saknar stöd till "feature phones" går det att illustrera dessa telefoner. Däremot blir det svårt att få till "känslan" att trycka på en verklig knapp – även om det går att tilldela objekt beteendet Vibrate blir det inte samma sak.

Man bör ha i åtanke att vid skapandet av en prototyp, som ska illustrera en applikation, kan testpersoner få uppfattningen att applikationen är en webbtjänst då testningen sker i en webbläsare och adressfältet syns. Om detta är en stor nackdel låter jag vara osagt då denna uppsats inte har behandlat den aspekten.

## 4.4 Förslag på fortsatt utvecklingsarbete

Framför allt är det två förslag på fortsatt utvecklingsarbete som har kommit fram genom denna studie:

1. För att kunna utföra vidare studier om användning av WOz-tekniken mot mobiltelefoner bör man tänka över ett verktyg som klarar av tester mot "feature phones".
2. För fortsatt utveckling av den portabla miljön bör problemet med nätverksdelningen åtgärdas vilket skulle kunna göra den mer portabel.

Naturligtvis finns det ytterligare aspekter på hur väl Ozlab fungerar i ett verkligt scenario där testning sker mot befolkning i utvecklingsländer. Men dessa får kanske sin rätta belysning först när man börjar pröva Ozlab i fält och då bör ovanstående två saker vara lösta.

# Källförteckning

Alce, G., Hermodsson, K., Wallergård, M. (2013), WozARd: A Wizard of Oz Tool for Mobile AR, MobileHCI 2013.

Ardito, C., Buono, P., Costabile, M. F., Lanzilotti, R. & Piccinno, A. (2009). A tool for Wizard of Oz studies of multimodal mobile systems. HSI 2009, Catania, Italy, May 21-23, sid. 344-347.

Beaudouinn-Lafon, M., Mackay, W., (2003), Prototyping tools and techniques. Kapitel i The Human-Computer Interaction Handbook. Fundamentals, evolving technologies, and emerging applications. Eds. Jacko & Sears. Lawrence Erlbaum Associates. Sid 1006 – 1031.

Boots, M., Farmer, L. (2014). Barriers and Solutions in using M4D: Connecting Directly to Citizens for Scalable Impact. Niang, Scharff & Wamala eds., Proceedings of 4th International Conference on M4D Mobile Communication for Development, M4D 2014, General Tracks. Karlstad University Studies 2014:26. Sid. 208-212.

Buxton, B. (2007), Sketching user experience: getting the design right and the right design. Elsevier/Morgan Kaufmann, Amsterdam.

Can I use (2014), [Can I use](http://caniuse.com/) Web Sockets?, Tillgänlig: <http://caniuse.com/websockets> [2014-05-31]

Carter, S., Mankoff, J., Heer, J. (2007), Momento: Support for Situated Ubicomp Experimentation, CHI 2007. ACM.

Dahlbäck, N., A. Jönsson & L. Ahrenberg (1993), Wizard of Oz Studies – Why and How. Knowledge-Based Systems 6(4), sid. 258-266.

Economist (2011), Makers of mobile devices see a new growth market, Tillgänglig: http://www.economist.com/node/18529875?story\_id=18529875 [2014-05-29]

Gebregziabher, S. Krauss, K.E.M. (2014). The Bi-directional influence of Technology and Society: How M-PESA is shaping and is being shaped by business in Kenya. Niang, Scharff & Wamala eds., Proceedings of 4th International Conference on M4D Mobile Communication for Development, M4D 2014, General Tracks. Karlstad University Studies 2014:26. Sid 281-292.

Gitau, S., Marsden, G., Donner, J., (2010). After Access - Challenges Facing Mobile-Only Internet Users in the Developing World.Proceedings of the 28th international conference on human factors in computing systems (CHI 2010). ACM. Sid. 2603-2606.

[GSMA Mobile for Development](http://www.gsma.com/developmentfund/" \t "_blank) (2013), Countries, Tillgänlig: https://mobiledevelopmentintelligence.com/countries [2014-05-28]

Houde, S. & Hill, C. (1997), What do prototypes prototype? Kapitel i Handbook of Human-Computer Interaction. Second, completely revised edition (eds. Helander, Landauer & Prabhu). Elsevier Science B.V. Sid. 367-381.

Kelley, J.F. (March 1984), "An iterative design methodology for user-friendly natural language office information applications", ACM Transactions on Office Information Systems.

Kumar, V. & Svensson, J. (ed. 2012) Proceedings of M4D 2012 28-29 February 2012 New Delhi, India. Karlstad University Studies, 2012:3.

Larsson, U., (2006), Co-design as Proposals, Assessments and Decisions – Stakeholder Interaction in Information Systems Development. Presenterad vid Fourteenth International Conference on Information Systems Development (ISD´2005), Karlstad, August 2005. Proceedings published in Advances in Information Systems Development, eds. A.G. Nilsson et al., Springer-Verlag. Sid. 143-154.

Linnell, N., Bareiss, R., Pantic, K. (2012), A Wizard of Oz Tool for Android. MobileHCI'12, San Francisco, USA, September 21-24. ACM. Sid. 65-70.

Ndiaye, M. A., Zouinar, M., (2014). The Usage of Mobile Phones by Low-literate Users in Senegal: An Ethnigraphic Study. Niang, Scharff & Wamala eds., Proceedings of 4th International Conference on M4D Mobile Communication for Development, M4D 2014, General Tracks. Karlstad University Studies 2014:26.

Niang, I., Scharff, CH. & Wamala, C. (eds. 2014), Proceedings of 4th International Conference on M4D Mobile Communication for Development, M4D 2014, General Tracks. Karlstad University Studies 2014:26.

Nilsson, J & J. Siponen (2006), Challenging the HCI concept of fidelity by positioning Ozlab prototypes. Presented at ISD´2005, Karlstad. Published in Andvances in Information Systems Development, eds. A.G. Nilsson et al., Springer-Verlag, 2006. Sid. 349-360.

Patel R. & Davidson B. (2003), Forskningsmetodikens grunder. Att planera, genomföra och rapportera en undersökning. Studentlitteratur

Pettersson, J.S. (2003), Ozlab – a system overview with an account of two years of experiences. In HumanIT 2003. Karlstad University Studies 2003:26, ed. J.S. Pettersson. Karlstad University. Sid. 159-185.

Pettersson, J.S. (ed. 2008), Proceedings of The 1st International Conference on M4D Mobile Communication Technology for Development (M4D 2008, General Tracks): 11–12 December, 2008, Karlstad University, Sweden. Karlstad University Studies 2008:61.

Pettersson, J.S. & J. Nilsson (2011), Effects on software quality of early user-testing (testing before programming). Data from a case study. Presented at The 18th International Conference on Information Systems Development (ISD2009), September 16-19, Nanchang, China. Published in Information Systems Development. Asian Experiences. Springer, New York. Pp 499-510. ISBN: 978-1-4419-7205-7

Rubin, J. & Chisnell, D. (2008), Handbook of Usability Testing, Second Edition: How to Plan,Design, and Conduct Effective Tests, Wiley Publishing, Inc.

Sencha (2014), Announcing Public Beta of Ext JS, Tillgänglig: http://www.sencha.com/blog/announcing-public-beta-of-ext-js-5/ [2014-05-14]

Sencha (2014), Sencha Ext JS JavaScript Framework for Rich Desktop Apps, Tillgänglig: <http://www.sencha.com/products/extjs/> [2014-05-14]

Svenson, J & Wicander, G. (ed. 2010) Proceedings of the 2nd International conference on M4D Mobile Communication Technology for Development M4D 2010: 10-11 November 2010 Kampala, Uganda. Karlstad University Studies 2010:31.

The Next Web (2011), Report: Smartphones account for just 27% of all mobile phones worldwide, Tillgänglig: <http://thenextweb.com/mobile/2011/11/29/report-smartphones-account-for-just-27-of-all-mobile-phones-worldwide/> Tillgänglig: [2014-05-31]

Time (2011), Kenya's Banking Revolution. Artikel publicerad: 2011-01-31. http://content.time.com/time/magazine/article/0,9171,2043329,00.html Tillgänglig:[2014-05-31]

Vetensskapsrådet (2012). Forskningsetiska principer inom humanistisk-samhällvetenskaplig forskning. [Elektronisk]. Tillgänglig: http://www.codex.vr.se/texts/HSFR.pdf. Tillgänglig :[2014-05-13]

Warfel, T. Z.(2009), Prototyping A Practitioner's Guide. Rosenfeld Media Brooklyn, New York.

World Bank (2013), Energy, the facts Tillgänglig: <http://go.worldbank.org/6ITD8WA1A0> [2014-05-29]

Wikipedia (2014), Feature Phone, Tillgänglig: http://en.wikipedia.org/wiki/Feature\_phone [2014-05-26]

# Bilagor

## Bilaga 1 Medgivande för testperson

## Medgivande

Data kommer vara i form av intervjufrågor och reflektioner från test. Inga personliga uppgifter kommer att publiceras, enbart resultat från tester.

- Jag bekräftar att jag fått denna skriftliga samt annan muntlig information om forskningsstudien.  
- Jag ger mitt samtycke till att delta i studien och vet att mitt deltagande är helt frivilligt.  
- Jag är medveten om att jag när som helst och utan förklaring kan avsluta mitt deltagande.

 Namn Underskrift Datum

-------------------- -------------------- ----------------

## Bilaga 2 Testsessions prototyp och dess navigering

