# Afstudeerscriptie

Sieger Falkena

Inhoud

[Afstudeerscriptie 1](#_Toc448739134)

[Inleiding 1](#_Toc448739135)

[Hoe maak ik een complexe webapplicatie inzichtelijk op mobiele apparaten? 1](#_Toc448739136)

[Wat zijn de eigenschappen van gebruiksvriendelijkheid? 1](#_Toc448739137)

[PACMAD 1](#_Toc448739138)

[Wat zijn de limitaties van een mobiel apparaat t.o.v. een desktop? 2](#_Toc448739139)

[Klein scherm 2](#_Toc448739140)

[Klein toetsenbordje 2](#_Toc448739141)

[Netwerk 2](#_Toc448739142)

[Hardware 2](#_Toc448739143)

[Memory leaks 2](#_Toc448739144)

[Hoe kunnen deze limitaties worden gelimiteerd? 2](#_Toc448739145)

[Responsive layout 2](#_Toc448739146)

[Autofill 2](#_Toc448739147)

[Dynamic loading 2](#_Toc448739148)

[Object lifecycles 2](#_Toc448739149)

[Ondiepe functieboom 2](#_Toc448739150)

[Wat zijn de voordelen van een mobiel apparaat ten opzichten van een desktop? 2](#_Toc448739151)

[Touchscreen 2](#_Toc448739152)

[Microfoon 2](#_Toc448739153)

[Camera 2](#_Toc448739154)

[Tactile 2](#_Toc448739155)

[Hoe kunnen we deze voordelen gebruiken om data inzichtelijk te maken? 2](#_Toc448739156)

[Gestures / swipe 2](#_Toc448739157)

[Barcode readers 2](#_Toc448739158)

[text readers 2](#_Toc448739159)

[Hoe kunnen we de webapplicatie structureren om de mobiele toegankelijkheid te vergroten? 3](#_Toc448739160)

[Doorklikdiepte minimaliseren 3](#_Toc448739161)

[Details abstraheren 3](#_Toc448739162)

# Inleiding

Dit onderzoeksrapport is geschreven naar aanleiding van mijn afstudeeronderzoek bij Diract-it.

Diract-it wil weten of het mogelijk is enige nieuwe webapplicatie ook in te zetten voor mobiele platformen, als smartphones of tablets.

## Concentrator

De concentrator is een product dat gebruikt word voor het importeren en exporteren van productinformatie naar fysieke winkels en webshops.

Dit importeren en exporteren gaat aan de hand van business-rules die voor de klant worden geprogrammeerd.

Producten bevatten veel informatie zoals maat, naam en kleur. Deze informatie is beschikbaar in verschillende contexten zoals taal, import- of exportlocatie.

Notes **kwantificeren door bijbehorende referenties op te zoeken**

# Hoe maak ik een complexe webapplicatie bruikbaar op mobiele apparaten?

# Wat zijn de eigenschappen voor mobiele gebruiksvriendelijkheid?

## PACMAD

People at the centre of mobile application development

Mobiele gebruiksvriendelijkheid is anders dan gebruiksvriendelijkheid voor desktopsystemen, en veel problemen worden vaak over het hoofd gezien. Harrison et. Al. Hebben een metaonderzoek gedaan naar andere onderzoeken die eigenschappen voor mobiele gebruiksvriendelijkheid klassificeerden. (Harrison, et al., 2013)

Volgens harrison zijn de volgende factoren van invloed op mobiele gebruiksvriendelijkheid.

* De gebruiker
* De taken die de gebruiker wil ondernemen
* De context in welke de gebruiker deze taken wil uitvoeren

In het geval van Diract-IT is de gebruiker een *product manager* of *order manager*. Zij zijn respectievelijk verantwoordelijk voor het bijhouden van productinformatie en order afhandeling. Van beide kan een basiskunde van computers worden verwacht. Tussen *product managers* en *order managers* is een overlap in het takenpakket, en de functies worden in sommige organisaties door dezelfde werknemer gedaan.

De taken voor beide gebruikers ligt in het onderhouden en beheren van het informatiesysteem, ondersteund door professionals van Diract-IT. Voorbeelden zijn het inzien en aanpassen van productinformatie, het activeren en deactiveren van producten en het te hulp schieten van mislukte orders.

De nieuwe concentrator kent verschillende contexten; de applicatie kan door de huidige opzet momenteel alleen makkelijk met een desktop worden gebruikt. De organisatie wil dat de concentrator ook dynamisch op de werkvloer kan worden gebruikt met een tablet of smartphone.

In het PACMAD model hebben deze factoren invloed op de volgende attributen; Effectiviteit, Efficientie, Bevrediging, Leerbaarheid, Onthoudbaarheid, Fouten en Cognitieve Ballast. De eerste drie (Effectiviteit, Efficientie en bevrediging ) zijn het best te objectief te beoordelen en het meest van invloed op gebruikerservaring.

De andere attributen; (leerbaarheid, onderhoudbaarheid fouten en cognitieve ballast) zijn in de meeste studies alleen subjectief gemeten. Ondanks het gebrek aan informatie over deze attributen is het wel mogelijk om hier rekening mee te houden.

# Wat zijn mobiele limitatie en hoe kunnen deze worden gecompenseerd?

De limitaties van mobiele apparaten tegenover het traditionele desktop met toetsenbord en muis zijn significant.

## Visueel

Visueel hebben mobiele apparaten een over het algemeen kleiner beeldscherm en een hogere resolutie. Dit maakt dat elementen (knoppen en tekst) die als goed bruikbaar op desktop worden ervaren, als te klein worden ervaren op mobile devices. Vice versa kunnen tekst en knoppen voor mobile devices als te groot worden ervaren.

De oplossing komt uit het gebruik van Responsive Web Design. Door in plaats van pixels (resolutie) of meeteenheden (afmeting) een percentage te gebruiken is het mogelijk dezelfde content op meerdere platformen goed aan te bieden.

Er zijn meerdere frameworks waarmee RWD makkelijk te implementeren is. Voornaamste spelers zijn (Twitter bootstrap en Foundation). Bootstrap is veruit het populairs en makkelijkst om toe te passen, terwijl foundation meer configuraties en keuzevrijheid bied. Aangezien de requirements vroegen om een functioneel ontwerp, en ik geen ontwerper ben, is er voor het proof-of-concept gekozen bootstrap te gebruiken.

Het is op een kleiner beeldscherm moeilijker een overzicht te tonen. Het opdelen van functionaliteit is moeilijker te realiseren dan een langere

## Invoer

Mobiele apparaten beschikken over een touchscreen als primair invoerapparaat. Het touchscreen functioneert beide als muis en toetsenbord. Dit hoofdstuk behandeld alleen de zwaktes van het touchscreen t.o.v. muis en toetsenbord; de sterktes worden in een volgend hoofdstuk behandeld.

|  |  |  |
| --- | --- | --- |
|  | **Mouse** | **Fingers** |
| **Precision** | High | Low |
| **Number of points specified** | 1 | usually 1 2–3 with multi-touch |
| **Number of controls** | 3: left/right button, scroll wheel | 1 |
| **Homing time?** | Yes | No |
| **Signal states** | Hover, mouse-down, mouse-up | Finger-down, finger-up |
| **Accelerated movements** | Yes | No |
| **Suitable for use with big desktop monitors (30-inch or more)** | Yes, because of acceleration | No: arm fatigue |
| **Visible pointer/cursor** | Yes | No |
| **Obscures view of screen** | No, thus allowing for continuous visual feedback | Yes |
| **Suitable for mobile** | No | Yes: nothing extra to carry around |
| **Ease of learning** | Fairly easy | Virtually no learning time |
| **Direct engagement with screen and "fun" to use** | No: an indirect pointing device | Yes |
| **Accessibility support** | Yes | No |

Bron: <https://www.nngroup.com/articles/mouse-vs-fingers-input-device/>, 10-4-2012

**Touchscreen als muis**

Het touchscreen doet niet onder de muis, maar is wel sterk anders in gebruik. Vingers zijn minder precies dan een muiscursor, en nemen daarnaast het scherm in beslag. Daartegen is de touchscreen wel intuïtief in gebruik, omdat ze de meest natuurlijke impuls weet te gebruiken.

**Touchscreen als toetsenbord**

Een mobiele telefoon is minder geschikt voor de invoer van lange stukken tekst, voornamelijk omdat de tactiele sensatie van een toetsendruk ontbreekt en omdat de knoppen over het algemeen kleiner zijn dan die van een fysiek toetsenbord. Beginnende touchscreen typisten halen een gemiddelde van 10 wpm op een (klein) touchscreen, vergeleken met 41,2 wpm voor een normaal toetsenbord.

Typefouten zijn daarnaast makkelijk te maken op een touchscreen, voornamelijk vanwege de imprecisie van de vinger.

De oplossing voor slechte tekstinvoer is het automatisch aanvullen van gedeeltelijke informatie. Door de gebruiker automatisch aanvullingsopties te geven hoeven er minder toetsen te worden aangeslagen.

De tweede oplossing is het gebruik van de sterke eigenschappen van de telefoon, en wordt in volgende hoofdstukken behandeld.

## Netwerk

Een snelle internetconnectie is voor mobiele apparaten minder vanzelfsprekend dan een desktop. Telefoons zijn afhankelijk van snelle draadloze netwerken die niet altijd en overal beschikbaar zijn. Zelfs wifi kan overbezet worden door de bandbreedte van de radiofrequenties.

Een lage bandbreedte in combinatie met grote applicatiegrootte kan tot een trage gebruikerservaring leiden. Wachten op het laden van de applicatie kan voor frustraties zorgen. Dit is een grote frustraties voor *productmanagers* die efficiënt en effectief een aantal operaties willen uitvoeren.

Het dynamisch inladen van dependencies kan laadtijden verkorten. Voornamelijk dependencies met een significante grootte en situationeel gebruik (Quagga.js later meer) hoeven niet vanaf het begin ingeladen te worden.

Door dependencies vooraf te laden kunnen wachttijden worden verminderd naar 0. Door vanuit de navigatieboom alle dependencies van connected edges te laden hoeven te laden hoeft er tijdens een gebruikersactie niets te worden geladen omdat alles al geladen is. Ook bij zoekresultaten met verschillende pagina’s kan alvast de volgende pagina worden geladen.

Bepaalde onderdelen kunnen ook worden gecached in de browser, server en localstorage. De browser doet cached bepaalde objecten automatisch, maar bepaalde onderdelen kunnen niet makkelijk gecached worden. Zoekqueries met pagination kunnen alvast de tweede pagina laden.

## Hardware

Mobiele telefoons beschikken over minder geheugen dan een computer voor dezelfde prijs. Onzorgvuldig gebruik van dit geheugen kan leiden tot frustratie van de gebruiker. Javascript is garbage-collected, maar een combinatie van lange sessies en slecht objectbeheer kunnen op langere duur het geheugen van elk apparaat volstouwen. Het is belangrijk een goed objectmanagement uit te voeren, en niet-relevante objecten uit het geheugen te laden.

Een goed framework zoals AngularJS of React kunnen een applicatie architectuur te forceren, en door een architectuur te forceren (MVVC) is het duidelijk welke dependencies (views, controllers en bijbehorende objecten) wel-of niet relevant zijn.

# Wat zijn de voordelen van een mobiel apparaat ten opzichten van een desktop?

## Touchscreen

## Microfoon

## Camera

## Tactile

# Hoe kunnen we deze voordelen gebruiken om data inzichtelijk te maken?

## Gestures / swipe

Hammer.js – Angular-Gestures

## Barcode readers

QuaggaJS

## text readers

OCRAD

## 

# Hoe kunnen we de webapplicatie structureren om de mobiele toegankelijkheid te vergroten?

## Doorklikdiepte minimaliseren

## Details abstraheren

# Bibliografie

**Harrison, Rachel, Flood, Derek en Duce, David. 2013.** *Usability of mobile applications.* sl : SpringerOpen, 2013.