

Afstudeerdossier

Rapporten



Dit document bevat de volgende onderdelen:

- Onderzoek rapportage
- Scrum rapportage
- Overige Rapportage

Versie:	1.0
Auteur:	Alvin Nutbeij
Studentnummer:	50128
Examinatoren:	Gerard Tuk, Robin Collard
Bedrijfsmentor:	Jorrit van der Ven
Datum:	31 mei 2012

Afstudeerdossier

Onderzoek

NCIM  GROEP

DE **H/AAGSE**
HOGESCHOOL

Dit document bevat de volgende onderdelen:

- Het vooronderzoek
- Het onderzoeksplan
- Het onderzoeksrapport

Versie:	1.0
Auteur:	Alvin Nutbeij
Studentnummer:	50128
Examinatoren:	Gerard Tuk, Robin Collard
Bedrijfsmentor:	Jorrit van der Ven
Datum:	31 mei 2012

NCIM-GROEP

Platform Onafhankelijke Applicaties

Vooronderzoek

Versie 1.1

Alvin Nutbeij

2-4-2012

NCIM  **GROEP**

Samenvatting

Op basis van dit vooronderzoek naar de mogelijkheden om platform onafhankelijk mobiele applicaties te ontwikkelen zijn enkele conclusies getrokken.

Momenteel zijn er vier mobiele platformen om rekening mee te houden:

- Android van Google
- iOS van Apple
- Windows Phone van Microsoft
- BlackBerry van Research In Motion

Android en iOS domineren de markt op het gebied van mobiele apparaten (smartphones en tablets). Het ziet er naar uit dat in de komende jaren Windows Phone ook een speler wordt om rekening mee te houden. Op dit moment neemt het aantal BlackBerry gebruikers met grote snelheid af. Bij het ontwikkelen van platform onafhankelijke applicaties zijn Android en iOS dus belangrijke vereiste, terwijl BlackBerry en Windows Phone als extra welkom zijn.

Er zijn vier methoden gedefinieerd om platform onafhankelijk te ontwikkelen:

- Mobiele web applicatie
- Hybride applicatie
- Geïnterpreteerde applicatie
- Gegenereerde applicatie

De keuze voor de methode kan situatie afhankelijk zijn. Verder onderzoek is nodig om te bepalen welke methode geschikt is voor welke situatie. Er zijn voor elk van de methodes veel verschillende tools en frameworks. Welke tools en frameworks het beste zijn per methode moet nog verder onderzocht worden.

De applicatiewinkels van de platformen hebben elk een aantal eisen waaraan applicaties moeten voldoen. In het kader van platform onafhankelijk ontwikkelen is voor de gebruikersinterface belangrijk om in de gate te houden. Zowel Microsoft (Windows Phone) als Apple (iOS) stellen hier specifieke eisen aan.

Inhoudsopgave

Samenvatting.....	1
1 Inleiding	3
2 Mobiele Platformen	4
3 Methoden	5
3.1 Mobiele web applicatie	5
3.2 Hybride applicatie.....	5
3.3 Geïnterpreteerde applicatie.....	6
3.4 Gegenereerde applicaties	6
3.5 Overzicht.....	7
4 Tools & Frameworks.....	8
5 Eisen van applicatiewinkels	9
Bibliografie	11
Figuren.....	14

1 Inleiding

Binnen dit vooronderzoek wordt georiënteerd op de mogelijkheden en aandachtspunten van het platform onafhankelijk ontwikkelen van mobiele applicaties. Binnen dit vooronderzoek wordt naar de volgende onderdelen gekeken:

- Welke mobiele platformen spelen een rol?
- Welke methoden bestaan er om platform onafhankelijk te ontwikkelen?
- Welke tools & frameworks zijn beschikbaar?
- Welke eisen stellen de applicatiewinkels van de platformen?

In het onderdeel mobiele platformen worden de belangrijkste spelers op de markt geïdentificeerd, ook wordt gekeken naar de verwachte ontwikkelingen in de toekomst.

De methoden worden uitgelegd en de voor en nadelen worden tegenover elkaar gezet. Wanneer welke methode de beste keuze is wordt niet binnen dit document besproken. Van de tools en frameworks worden een aantal basis eigenschappen, zoals methode en programmeertalen, gegeven. Ook voor de tools en frameworks geldt dat nog geen aandacht wordt besteed aan welke het beste zijn.

Ten slotte wordt gekeken naar welke eisen de applicatiewinkels van de platformen stellen aan de applicaties die aan worden geboden. Hierbij wordt vooral gekeken naar de eisen wat betreft de gebruikersinterface.

Op basis van de antwoorden op bovenstaande vragen wordt het onderzoeksplan opgesteld. De antwoorden dienen tevens als basis voor het verdere onderzoek.

2 Mobiele Platformen

Om tools te kunnen selecteren moet eerst duidelijk zijn welke mobiele platformen bestaan en wat hun marktaandelen zijn. Vanwege het grote aantal mobiele operating systems is een selectie gemaakt van de operating systems met het grootste marktaandeel [1]. De volgende operating systems zijn benoemd:

- iOS
- Android
- BlackBerry
- Symbian

Omdat Nokia stopt met het door ontwikkelen van Symbian en vanaf 2016 volledig stopt met de ondersteuning ervoor [2] is gekozen hier geen rekening meer mee te houden. Nokia toestellen worden vanaf nu uitgerust met Windows Phone [3].

De exacte marktaandelen van de operating systems zijn moeilijk vast te stellen. Op basis van verkopen zijn de marktaandelen van 2011 opgenomen in Tabel 1, daarnaast bevat tabel 1 ook de voorspelde marktaandelen voor 2015 [4].

Operating System	Marktaandeel 2011	Voorspeld marktaandeel 2015
Android	47,4%	58,1%
iOS	18,0%	16,6%
Windows Phone	1,9%	16,7%
Overige	32,7%	8,6%
Totaal	100%	100%

Tabel 1 Marktaandeel 2011 en voorspelling voor 2015 (bron: IHS iSuppli januari 2011)

Het aandeel van BlackBerry is op dit moment niet bekend. In augustus van 2011 was dit 11,7% terwijl dit een jaar eerder nog op 18,7% lag. Slechts 14% van de BlackBerry gebruikers is tevreden met hun toestel, de verwachting is dus ook dat het marktaandeel nog verder zal afnemen [5].

Op basis van de huidige markt en de voorspellingen voor 2015 zijn iOS en Android de belangrijkste operating systems om rekening mee te houden. Op termijn wordt ook Windows Phone belangrijk.

3 Methoden

Om een keuze te kunnen maken omtrent platform onafhankelijk ontwikkelen wordt eerst gekeken welke methoden er zijn om platform onafhankelijk te ontwikkelen. Het platformonafhankelijk ontwikkelen kan globaal worden ingedeeld in vier categorieën [6]:

- Mobiele web applicatie
- Hybride applicatie
- Geïnterpreteerde applicatie
- Gegenereerde applicatie

3.1 Mobiele web applicatie

Mobiele web applicaties worden geopend in de web browser van de smartphone. Hierbij wordt gebruik gemaakt van HTML(5), CSS(3) en JavaScript. Om een meer native look & feel aan de applicatie te geven kan gebruik worden gemaakt frameworks. De applicatie draait op een webserver in plaats van op de smartphone.

Het voordeel van dit soort applicaties is dat de browsers van alle moderne smartphones geschikt zijn voor het gebruik van HTML5, CSS3 en JavaScript [7]. Daardoor is het op vrijwel alle smartphones mogelijk om de applicatie te gebruiken. Omdat de applicatie op een web server draait hoeven updates niet gedistribueerd te worden en worden wijzigingen direct bij de gebruikers doorgevoerd.

Het nadeel hiervan is echter dat het niet mogelijk is om gebruik te maken van de telefoon functies. Ook kan data niet lokaal opgeslagen worden. Daardoor kan de gebruiker de applicatie niet openen wanneer er geen internet verbinding beschikbaar is. Een ander nadeel van web applicaties is dat gebruikers deze niet in de app stores van de operating systems kunnen vinden.

Voor iOS apparaten is het wel mogelijk om lokaal data op te slaan middels 'HTML5 client-side caching and storage' [8]. Het is echter nog steeds niet mogelijk om functionaliteiten van de telefoon, zoals GPS, te gebruiken. Bij web applicaties is de gebruikersinterface op alle platformen hetzelfde is, terwijl de native gebruikersinterface van de platformen juist verschillend is.

3.2 Hybride applicatie

Hybride applicaties zijn in feite web applicaties die van een extra framework gebruik maken om de native functionaliteiten van een telefoon aan te roepen, meestal via een JavaScript API. Met behulp van het framework in combinatie met een bijbehorende tool wordt een native applicatie gegenereerd die slechts een webview, een geïntegreerd browser scherm, bevat. Daarnaast worden alle webbestanden ook in de applicatie meegenomen waardoor de applicatie ook offline bruikbaar is.

Het voordeel is dus dat alle applicaties nog steeds één code basis hebben terwijl ze wel op meerdere platformen uitgebracht kunnen worden. De applicaties zijn dan dus wel in de app stores van de platformen te vinden. Ook kunnen deze applicaties gebruik maken van de functionaliteiten van de smartphone, zoals de GPS.

Het nadeel van de hybride applicaties is dat de gebruikersinterface nog steeds gelijk is aan de gebruikersinterface van web applicaties. Vooral Microsoft is streng in het selecteren op native gebruikersinterface [9], maar ook Apple kan applicaties weren die niet aan de native gebruikersinterface voldoen [10] [11]. Ook zijn de, in applicatie geïntegreerde, browsers niet zo snel met betrekking tot animaties als native applicaties, dit is merkbaar voor de gebruikers.

3.3 Geïnterpreteerde applicatie

Geïnterpreteerde applicaties nemen de frameworks zoals bij hybride applicaties een stap verder. De tools die gebruikt worden om geïnterpreteerde applicaties te schrijven genereren native code op basis van de enkele code basis. Middels een API worden gebruikersinterface elementen aangemaakt, zo zijn deze in de enkele code basis hetzelfde maar wordt per platform andere code gegenereerd.

Het voordeel hiervan is dat de geïnterpreteerde applicaties echte native applicaties zijn. Problemen met animatie snelheden e.d. zijn dus weg en ook de gebruikersinterface is op de, voor dat platform, gebruikelijke manier opgebouwd.

Het nadeel is echter dat de programmeur nog meer dan anders gebonden is door functionaliteiten die het framework aanbiedt. Het zou bijvoorbeeld mogelijk zijn dat een framework geen API bevat om een lijst te genereren. In dat geval is het voor de ontwikkelaar dus niet mogelijk om gegevens in een lijst weer te geven, ondanks dat dit op smartphones wel gebruikelijk is. Ook is het nadeel van de API afhankelijkheid dat wanneer een platform een update krijgt het een tijd kan duren voordat deze ondersteund wordt, zo kunnen mogelijk nieuwe functionaliteiten pas veel later benut worden.

3.4 Gegenereerde applicaties

Gegenereerde applicaties worden gemaakt in één programmeertaal, bijvoorbeeld C++, en worden vervolgens omgezet naar native code voor de verschillende platformen. Ook in dit geval worden native gebruikersinterface elementen aangeroepen middels een API. Het is echter ook mogelijk om deze API's links te laten liggen en een eigen gebruikersinterface te maken. Deze is dan wel weer hetzelfde voor alle platformen, dit is vooral handig bij het maken van spelletjes of andere intensieve applicaties.

Het voordeel van deze methode is dat de code per platform gegenereerd wordt. Voordat de applicatie wordt uitgebracht kunnen dus ook eventuele platform specifieke wijzigingen doorgevoerd worden. Ook is het met programmeertalen als C++ beter mogelijk om complexe en zware applicaties, zoals spelletjes, te maken dan met bijvoorbeeld HTML en CSS.

Het grootste nadeel van deze methode is dat het over het algemeen een hoog prijskaartje heeft. Daarnaast is dienen ontwikkelaars mogelijk een nieuwe programmeer taal te leren, terwijl bij de andere oplossingen standaarden als HTML, CSS en JavaScript gebruikt kunnen worden.

3.5 Overzicht

In Tabel 2 wordt een overzicht gegeven van de voor –en nadelen van de in dit hoofdstuk behandelde methoden.

Methode	Voordelen	Nadelen
Mobiele web applicatie	<ul style="list-style-type: none"> • Werken met bekende web standaarden • Updates hoeven niet gedistribueerd te worden 	<ul style="list-style-type: none"> • Geen native gebruikersinterface • Wordt niet opgenomen in de app stores • Geen lokale dataopslag • Geen gebruik van de functionaliteiten van de smartphone • Gecompliceerde animaties trager dan native applicaties
Hybride applicatie	<ul style="list-style-type: none"> • Werken met bekende web standaarden • Distributie via app stores • Gebruik van smartphone functionaliteiten 	<ul style="list-style-type: none"> • Geen native gebruikersinterface • Animaties trager dan native animaties
Geïnterpreteerde applicatie	<ul style="list-style-type: none"> • Native gebruikersinterface • Distributie via app stores • Native animaties 	<ul style="list-style-type: none"> • Afhankelijk van API
Gegenereerde applicatie	<ul style="list-style-type: none"> • Native gebruikersinterface • Applicatie specifieke code • Complexe applicaties mogelijk 	<ul style="list-style-type: none"> • Gedeeltelijk afhankelijk van API • Mogelijk een nieuwe programmeertaal leren • Duur

Tabel 2 Overzicht voor -en nadelen van platform onafhankelijke ontwikkel methoden

4 Tools & Frameworks

Er bestaan heel veel tools om platform onafhankelijk te ontwikkelen. De tijdens dit vooronderzoek gevonden tools & frameworks zijn per methode te vinden in Tabel 3. Vanwege het grote aanbod zullen deze tools niet allemaal uitvoerig behandeld kunnen worden binnen dit onderzoek. Ook bestaat de kans dat later in dit onderzoek nieuwe tools gevonden worden, mogelijk worden deze dan ook behandeld.

Tool	Methode	Programmeertalen	Ondersteunde platformen
PhoneGap	Hybride applicatie	HTML5, CSS3, JavaScript [12]	iOS, Android, BlackBerry, webOS, Windows Phone, Symbian, Bada
Appcelerator	Geïnterpreteerde applicatie	HTML5, CSS3, JavaScript, Ruby, Rails, PHP, Python [13]	iOS, Android, BlackBerry (beta)
Corona	Geïnterpreteerde applicatie	Lua [14]	iOS, Android
Rhodes	Hybride applicatie ¹	HTML, CSS, JavaScript, Ruby [15]	BlackBerry, Windows Mobile, Android, iOS
MoSync	Gegenereerde applicatie	HTML5, CSS3, JavaScript, C, C++ [16]	Java ME, Windows Mobile, Symbian, Android, iOS, Moblin/MeeGo, BlackBerry
Marmalade	Gegenereerde applicatie	C++ [17]	iOS, Android, Symbian, Bada, webOS
Codename One²	Gegenereerde applicatie	Java [18]	iOS, Android, Windows Phone (in planning), BlackBerry, J2ME
Wink Toolkit	Web applicatie	HTML, CSS, JavaScript [19]	iOS, Android, BlackBerry, Bada, Windows Phone
jQuery mobile	Web applicatie	HTML, CSS, JavaScript [20]	Alle
Sencha Touch	Web applicatie	HTML, CSS, JavaScript [21]	Alle
The-M-Project	Web applicatie	HTML5, CSS3, JavaScript [22]	Alle
jQTouch	Web applicatie	HTML5, CSS3, JavaScript [23]	Alle
Jo	Web applicatie	HTML5, CSS, JavaScript [24]	Alle

Tabel 3 Tools & Frameworks om platform onafhankelijk te ontwikkelen

- 1 Rhodes bestaat uit een webview met daarin de applicatie zoals deze door de ontwikkelaar is geschreven. Rhodes bevat ook een API om smartphone functionaliteiten aan te roepen. In dat opzicht is het een hybride applicatie. Echter zijn daarnaast ook enkele native gebruikersinterface objecten beschikbaar dus heeft het ook iets weg van een geïnterpreteerde applicatie. [25]
- 2 Codename One is momenteel in gesloten bèta. Het klinkt als een veelbelovend project, echter zal het niet op tijd publiek worden om in dit onderzoek meegenomen te kunnen worden.

5 Eisen van applicatiewinkels

Om te kijken of tools & frameworks geschikt zijn voor het maken van native applicaties is het belangrijk te weten wat de eisen aan deze applicaties zijn. Als applicaties niet aan de eisen van de applicatiewinkel voldoen kunnen ze geweigerd worden. In dat geval kan de tool / het framework nog steeds nuttig zijn voor applicaties die ad hoc worden uitgebracht en dus niet via een applicatiewinkel lopen.

Alle applicatiewinkels hebben in hun voorwaarden een clausule die ze in staat stelt om applicaties te verwijderen. BlackBerry en Android hebben slechts beperkte voorwaarden. BlackBerry zegt slechts dat ze het recht hebben om applicaties te controleren en eventueel tegen te houden [26]. Android heeft uitgebreider beschreven waar de grenzen liggen, echter gaat dit alleen over de geschiktheid van content en niet over de gebruikerservaring [27].

Apple en Microsoft hebben hun voorwaarden een stuk uitgebreider beschreven. In het geval van Apple zijn de voorwaarden alleen zichtbaar voor geregistreerde ontwikkelaars [10]. In het geval van platform onafhankelijke applicaties zijn de regels omtrent de gebruikersinterface de belangrijkste. Apple stelt hieraan de volgende eisen:

10. User interface

10.1 Apps must comply with all terms and conditions explained in the [Apple iOS Human Interface Guidelines](#)

10.2 Apps that look similar to apps bundled on the iPhone, including the App Store, iTunes Store, and iBookstore, will be rejected

10.3 Apps that do not use system provided items, such as buttons and icons, correctly and as described in the [Apple iOS Human Interface Guidelines](#) may be rejected

10.4 Apps that create alternate desktop/home screen environments or simulate multi-app widget experiences will be rejected

10.5 Apps that alter the functions of standard switches, such as the Volume Up/Down and Ring/Silent switches, will be rejected

10.6 Apple and our customers place a high value on simple, refined, creative, well thought through interfaces. They take more work but are worth it. Apple sets a high bar. If your user interface is complex or less than very good, it may be rejected

In de genoemde *Apple iOS Human Interface Guidelines* zijn alle details waarop een applicatie geweigerd kan worden te vinden, zoals te zien in punt 10.1 en punt 10.3 van de *App Store Review Guidelines*. Dit gaat voornamelijk over het gebruik van de standaard iOS elementen, zoals bijvoorbeeld de navigatie [11].

Microsoft heeft, nog meer dan Apple, uitgestippeld aan welke eisen een applicatie moet voldoen [28]. Dit zorgt voor een groot probleem tijdens het ontwikkelen voor meerdere platformen vanuit een enkele code basis. Android en BlackBerry stellen geen eisen aan de gebruikersinterface. Microsoft en Apple verwachten echter dat de applicaties voor hun mobiele apparaten voldoen aan de grafische richtlijnen. Regels zoals 10.6 bij Apple zorgen ervoor dat elke applicatie geweigerd kan worden, immers kan “minder dan erg goed” niet objectief vastgesteld worden en kan dus altijd als reden voor de weigering gegeven worden.

Deze regels zorgen er dus in elk geval voor dat wanneer een applicatie voor alle vier de platformen ontwikkeld wordt en in de applicatie winkels geplaatst moet worden het mogelijk moet zijn om verschillende gebruikersinterfaces te maken.

Bibliografie

- [1] G. Lyons, „2012 Mobile Market Share,” iCrossing, 2 februari 2012. [Online]. Available: http://connect.icrossing.co.uk/2012-mobile-market-share-infographic_7962. [Geopend 9 februari 2012].
- [2] R. Schoemaker, „Nokia stopt met Symbian in Amerika,” Webwereld, 10 augustus 2011. [Online]. Available: <http://webwereld.nl/nieuws/107557/nokia-stopt-met-symbian-in-amerika.html>. [Geopend 10 februari 2012].
- [3] E. Rekers, „Nokia stapt over op Windows Phone 7,” Elsevier, 11 februari 2011. [Online]. Available: <http://www.elsevier.nl/web/Nieuws/Internet-Gadgets/289116/Nokia-stapt-over-op-Windows-Phone-7.htm>. [Geopend 10 februari 2012].
- [4] W. Lam, „Lumia 900 Introduction to Trigger Smartphone Renaissance for Nokia and Microsoft,” iSuppli, 18 januari 2012. [Online]. Available: <http://www.isuppli.com/Mobile-and-Wireless-Communications/News/Pages/Lumia-900-Introduction-to-Trigger-Smartphone-Renaissance-for-Nokia-and-Microsoft.aspx>. [Geopend 9 februari 2012].
- [5] C. v. Hoek, „Aandeel Blackberry gaat verder dalen,” Nu.nl, 24 oktober 2011. [Online]. Available: <http://www.nu.nl/gadgets/2649121/aandeel-blackberry-gaat-verder-dalen.html>. [Geopend 9 februari 2012].
- [6] H. Behrens, „Cross-Platform App Development for iPhone, Android & Co - A Comparison | Presented at Mobile TechCon 2010,” 11 oktober 2010. [Online]. Available: <http://heikobehrens.net/2010/10/11/cross-platform-app-development-for-iphone-android-co-%E2%80%94a-comparison-i-presented-at-mobiletechcon-2010/>. [Geopend 9 februari 2012].
- [7] A. Deveria, „When can I use...,” CanIUse, 1 februari 2012. [Online]. Available: <http://caniuse.com/>. [Geopend 10 februari 2012].
- [8] Apple, „Safari Client-Side Storage and Offline Applications Programming Guide,” Apple, 21 september 2011. [Online]. Available: <https://developer.apple.com/library/safari/#documentation/iPhone/Conceptual/SafariJSDatabaseGuide/Introduction/Introduction.html>. [Geopend 10 februari 2012].
- [9] Microsoft, „Best practices for application marketing,” Microsoft, [Online]. Available: http://create.msdn.com/en-US/home/about/app_submission_walkthrough_application_marketing. [Geopend 10 februari 2012].
- [10] Apple, „App Store Review Guidelines,” Apple, 2012. [Online]. Available: <https://developer.apple.com/appstore/resources/approval/guidelines.html#user-interface>. [Geopend 2 februari 2012].

- [11] Apple, „iOS Human Interface Guidelines,” Apple, 12 oktober 2011. [Online]. Available: https://developer.apple.com/library/ios/#documentation/UserExperience/Conceptual/MobileHIG/Introduction/Introduction.html#//apple_ref/doc/uid/TP40006556-CH1-SW1. [Geopend 10 februari 2012].
- [12] Adobe Systems, „About PhoneGap,” Adobe Systems, [Online]. Available: <http://phonegap.com/about>. [Geopend 7 februari 2012].
- [13] Appcelerator, „Titanium Studio,” Appcelerator, [Online]. Available: <http://www.appcelerator.com/products/titanium-studio/>. [Geopend 7 februari 2012].
- [14] Anscamobile, „App Development Tool,” Anscamobile, [Online]. Available: <http://developer.anscamobile.com/content/introduction>. [Geopend 8 februari 2012].
- [15] Rhomobile, „Rhodes,” Rhomobile, [Online]. Available: <http://docs.rhomobile.com/rhodes/introduction>. [Geopend 10 februari 2012].
- [16] MoSync, „HTML5, JavaScript and CSS,” MoSync, [Online]. Available: <http://www.mosync.com/content/html5-and-javascript-programming-mosync>. [Geopend 7 februari 2012].
- [17] IdeaWorks3D, „Desktop C++ Development,” IdeaWorks3D, [Online]. Available: <http://www.madewithmarmalade.com/marmalade/features/desktop-c-development>. [Geopend 7 februari 2012].
- [18] Codename One, „FAQ,” Codename One, [Online]. Available: <http://www.codenameone.com/faq.html>. [Geopend 8 februari 2012].
- [19] Dojo Foundation, „Wink toolkit,” Dojo Foundation, [Online]. Available: <http://www.winktoolkit.org/>. [Geopend 10 februari 2012].
- [20] The jQuery Project, „jQuery Mobile Overview,” The jQuery Project, [Online]. Available: <http://jquerymobile.com/demos/1.0.1/docs/about/intro.html>. [Geopend 7 februari 2012].
- [21] Sencha, „Mobile JavaScript Framework for HTML5 Web App Development,” Sencha, [Online]. Available: <http://www.sencha.com/products/touch>. [Geopend 10 februari 2012].
- [22] Panacoda, „The-M-Project,” Panacoda, [Online]. Available: <http://the-m-project.org/>. [Geopend 10 februari 2012].
- [23] D. Kaneda, „README.md at master from senchalabs/jQTouch,” Sencha Labs, 17 december 2011. [Online]. Available: <https://github.com/senchalabs/jQTouch/blob/master/README.md>. [Geopend 10 februari 2012].
- [24] J. Dave Balmer, „Jo HTML5 Mobile App Framework,” [Online]. Available: <http://joapp.com/>. [Geopend 8 februari 2012].

- [25] Rhomobile, „Rhodes Application User Interface,” Rhomobile, [Online]. Available: <http://docs.rhomobile.com/rhodes/ui>. [Geopend 10 februari 2012].
- [26] Research In Motion, „BlackBerry App World Vendor Agreement,” [Online]. Available: http://us.blackberry.com/legal/BlackBerry_App_World_Vendor_Agreement_cl_EN.pdf. [Geopend 10 februari 2012].
- [27] Google, „Android Market Developer Program Policies,” Google, [Online]. Available: <http://www.android.com/us/developer-content-policy.html>. [Geopend 10 februari 2012].
- [28] Microsoft, „Windows Phone Development,” Microsoft, 15 december 2011. [Online]. Available: [http://msdn.microsoft.com/en-us/library/ff402535\(v=vs.92\).aspx](http://msdn.microsoft.com/en-us/library/ff402535(v=vs.92).aspx). [Geopend 10 februari 2012].

Lijst van Figuren

Tabel 1 Marktaandeel 2011 en voorspelling voor 2015 (bron: IHS iSuppli januari 2011.....	4
Tabel 2 Overzicht voor -en nadelen van platform onafhankelijke ontwikkel methoden	7
Tabel 3 Tools & Frameworks om platform onafhankelijk te ontwikkelen	8

Platformonafhankelijke Applicaties

Onderzoeksplan

Versie 1.0

Alvin Nutbeij

13-2-2012

NCIM  **GROEP**

Inhoudsopgave

1 Aanleiding.....	2
2 Probleemomschrijving.....	2
2.1 Probleemstelling.....	2
2.2 Doelstelling.....	2
2.3 Deelvragen.....	2
2.4 Begripsafbakening.....	3
3 Onderzoeksontwerp.....	3
4 Tijkpad	3

1 Aanleiding

Naar aanleiding van het vooronderzoek is gekozen om verder onderzoek te verrichten naar de mogelijkheden om platformonafhankelijk mobiele applicaties te ontwikkelen. Vanwege het grote aantal frameworks en tools dat te vinden is worden niet alle frameworks en tools uitgebreid behandeld. Hiervan wordt in een later stadium een selectie gemaakt.

2 Probleemomschrijving

Binnen dit hoofdstuk worden de probleemstelling en de doelstelling beschreven. Daarnaast worden enkele deelvragen opgesteld en worden een aantal begrippen afgebakend.

2.1 Probleemstelling

Wat is de beste manier om platformonafhankelijke mobiele applicaties te ontwikkelen?

2.2 Doelstelling

De opdrachtgever wil advies hebben over de beste manier om platformonafhankelijke mobiele applicaties te maken. Naast dit advies wil de opdrachtgever ook, doormiddel van een Proof of Concept, zien hoe de applicaties gemaakt moeten worden.

2.3 Deelvragen

Om de probleemstelling te kunnen beantwoorden dienen de volgende deelvragen beantwoord te worden:

1. Met welke tools/frameworks wordt geëxperimenteerd?
Vanwege het grote aantal beschikbare tools & frameworks die zijn gevonden tijdens het vooronderzoek kunnen deze niet allemaal binnen dit onderzoek gebruikt worden. Daarom moet worden bepaald met welke tools/frameworks geëxperimenteerd moet worden.
2. Welke functionaliteiten moet het Proof of Concept bevatten?
Voor elke gekozen tool/framework wordt een Proof of Concept gemaakt. Om te kunnen bepalen welke tools/frameworks geschikt zijn moeten een aantal minimale eisen worden opgesteld.
3. Hoe belangrijk is gebruikerservaring?
Om te bepalen of een tool/framework geschikt is speelt de gebruikerservaring een belangrijke rol. Deze kan verschillen per gebruikt framework. Deze deelvraag is ontstaan nadat de experimenten waren uitgevoerd.
4. Welke toekomstige ontwikkelingen worden verwacht?
De huidige generatie telefoons zijn nog niet altijd krachtig genoeg om alle functionaliteiten goed weer te kunnen geven. Daarnaast zijn ook nog niet alle functionaliteiten in de frameworks geïmplementeerd. Deze deelvraag is ontstaan nadat de experimenten waren uitgevoerd.

2.4 Begripsafbakening

Voor dit onderzoek geldt dat 'platformonafhankelijkheid' betekend dat de applicatie geschikt is voor de in het vooronderzoek bepaalde platformen:

- Android
- iOS
- Windows Phone
- BlackBerry

Daarbij hebben Android en iOS de prioriteit omdat deze platforms het belangrijkste zijn voor de opdrachtgever. Daarnaast is Windows Phone een relatief nieuw platform waardoor nog weinig ondersteuning verwacht kan worden.

Met 'gebruikerservaring' wordt binnen dit onderzoek de belevenis die de gebruik ervaart tijdens het gebruik van de applicatie bedoeld.

3 Onderzoeksontwerp

Om een selectie te maken van de tools en frameworks waarmee geëxperimenteerd wordt, worden de tools en frameworks die in het vooronderzoek naar voren kwamen geanalyseerd op basis van specificaties es.

Om te bepalen welke oplossing het meest geschikt is wordt voor elk van de geselecteerde oplossingen eenzelfde Proof of Concept gebouwd. Aan de hand van dit Proof of Concept kan dan bepaald worden welke oplossing het meest geschikt is. Daarbij wordt ook rekening gehouden met toekomstige ontwikkelingen en het belang van de gebruikerservaring.

4 Tijdspad

In Figuur 1 is het tijdspad voor het onderzoek te zien. Hierbij geldt dat Week 1 start op 13 februari 2012. De invulling van de Proof of Concepts is bepaald nadat deelvraag 1 beantwoord is.

	Week 1	Week 2	Week 3	Week 4	Week 5
Tools/Frameworks selecteren					
Proof of Concept ontwerpen					
PhoneGap + jQuery Mobile PoC					
Appcelerator Titanium PoC					
MoSync PoC					
Onderzoeksrapport opstellen					

Figuur 1 Tijdspad onderzoek

Platformonafhankelijke Applicaties

Onderzoeksrapport

Versie 0.6

Alvin Nutbeij

12-3-2012

NCIM  **GROEP**

Samenvatting

In dit rapport wordt antwoord gegeven op de vraag wat de beste manier is om platformonafhankelijke applicaties te ontwikkelen. Om de beste tool te bepalen wordt een selectie gemaakt van de beschikbare tools. Vervolgens wordt met de geselecteerde tools een Proof of Concept gebouwd.

De tools die geëvalueerd worden zijn:

- PhoneGap in combinatie met jQuery Mobile
- Appcelerator Titanium
- MoSync

Op basis van de gemaakte Proof of Concepts kan geconcludeerd worden dat de tools nog niet ver genoeg zijn om platformonafhankelijke applicaties te ontwikkelen. Voor lichtere applicaties kunnen web applicaties gemaakt worden en voor spellen kunnen gegenereerde applicaties gebruikt worden.

Door de snelle ontwikkeling van de tools is het echter goed mogelijk dat de tools in de nabije toekomst voor alle situaties geschikt zullen zijn.

Inhoudsopgave

Samenvatting.....	1
1 Inleiding	4
1.1 Achtergrond.....	4
1.2 Aanleiding.....	4
1.3 Probleemstelling.....	4
1.4 Deelvragen.....	5
1.5 Hypothese	5
2 Uitvoering onderzoek	6
2.1 Welke tools & frameworks worden onderzocht?	6
2.1.1 Mobiele web applicatie	6
2.1.2 Hybride applicatie.....	7
2.1.3 Geïnterpreteerde applicatie.....	7
2.1.4 Gegenerateerde applicatie	7
2.2 Welke functionaliteiten moet het Proof of Concept bevatten?	8
2.3 Gebruikerservaring	10
2.4 Toekomstige ontwikkelingen	10
3 Experiment	11
3.1 PhoneGap & jQuery Mobile	11
3.1.1 PhoneGap introductie	11
3.1.2 jQuery Mobile introductie.....	11
3.1.3 Leercurve	11
3.1.4 Uitvoering.....	12
3.1.5 Geheugengebruik	13
3.1.6 Conclusie	14
3.2 Appcelerator Titanium	15
3.2.1 Leercurve	15
3.2.2 Uitvoering.....	15
3.2.3 Geheugengebruik	16
3.2.4 Conclusie	17
3.4 MoSync.....	18
3.4.1 Leercurve	18
3.4.2 Uitvoering.....	18
3.4.3 Conclusie	20

4	Conclusie en aanbevelingen.....	21
4.1	Beperkingen.....	21
4.2	Conclusie	21
4.3	Aanbeveling naar aanleiding van het onderzoek	22
4.3.1	Algemeen.....	22
4.3.2	MOFLAS	24
4.3.3	Restaurant App.....	24
5	Bibliografie.....	25
6	Lijst van figuren	27

1 Inleiding

In dit onderzoeksrapport worden de uitvoer en het resultaat van het onderzoek naar platformonafhankelijke applicatie ontwikkeling besproken. Allereerst wordt in dit hoofdstuk de situatie en de probleemomschrijving geschetst. Daarna worden de deelvragen beantwoord en worden de experimenten uitgevoerd. Tot slot wordt aan de hand van de experimenten en de deelvragen een conclusie getrokken en worden enkele aanbevelingen gedaan.

1.1 Achtergrond

In 2011 zijn de PC verkopen voor het eerst ingehaald door de smartphone verkopen [1]. Van de PC verkopen nemen tablets ongeveer 15% in beslag. De applicaties voor deze mobiele apparaten zijn inmiddels een miljarden business. Volgens een schatting van ABI Research [2] werd in 2011 wereldwijd \$8,5 miljard omgezet. Naar hun schatting groeit dit in 2016 tot \$46 miljard.

Om meer (potentiele) klanten te bereiken en mogelijk de omzet te verhogen is het voor bedrijven aantrekkelijk om de stap naar de wereld van mobiele applicaties te zetten. Het probleem hierbij zijn de verschillende mobiele platforms. De grote spelers op dit moment zijn:

- Android van Google
- iOS van Apple
- Windows Phone van Microsoft
- BlackBerry van Research In Motion

Om een zo groot mogelijk publiek te bereiken is het dus van belang om applicaties voor al deze platformen uit te brengen. Een applicatie voor elk platform schrijven zou echter betekenen dat dezelfde applicatie meerdere keren geschreven moet worden in verschillende programmeertalen (Java voor Android en BlackBerry, Objective-C voor iOS en C# voor Windows Phone).

1.2 Aanleiding

Gezien de ontwikkelingen in de mobiele markt wil de opdrachtgever zich ook gaan richten op het ontwikkelen van mobiele applicaties. Dit zou echter betekenen dat voor elk platform medewerkers opgeleid moeten worden. Daarom is de opdrachtgever op zoek naar de beste methode om platformonafhankelijke mobiele applicaties te ontwikkelen.

1.3 Probleemstelling

Wat is de beste manier om platformonafhankelijke mobiele applicaties te ontwikkelen?

1.4 Deelvragen

Om de probleemstelling te kunnen beantwoorden dienen de volgende deelvragen beantwoord te worden:

1. Met welke tools/frameworks wordt geëxperimenteerd?
Vanwege het grote aantal beschikbare tools & frameworks die zijn gevonden tijdens het vooronderzoek kunnen deze niet allemaal binnen dit onderzoek geëvalueerd worden. Daarom moet worden bepaald met welke tools/frameworks geëxperimenteerd moet worden.
2. Welke functionaliteiten moet het Proof of Concept bevatten?
Voor elke gekozen tool/framework wordt een Proof of Concept gemaakt. Om te kunnen bepalen welke tools/frameworks geschikt zijn moeten een aantal minimale eisen worden opgesteld.
3. Hoe belangrijk is gebruikerservaring?
Om te bepalen of een tool/framework geschikt is speelt de gebruikerservaring een belangrijke rol. Deze kan verschillen per gebruikt framework. Deze deelvraag is ontstaan nadat de experimenten waren uitgevoerd.
4. Welke toekomstige ontwikkelingen worden verwacht?
De huidige generatie telefoons zijn nog niet altijd krachtig genoeg om alle functionaliteiten goed weer te kunnen geven. Daarnaast zijn ook nog niet alle functionaliteiten in de frameworks geïmplementeerd. Deze deelvraag is ontstaan nadat de experimenten waren uitgevoerd.

1.5 Hypothese

De verwachting is dat gegenereerde applicaties het beste resultaat op zullen leveren. Doordat deze applicaties code werkelijk omzetten naar native code kan hierbij waarschijnlijk het beste resultaat gehaald worden op het gebied van performance en de native look & feel.

2 Uitvoering onderzoek

In dit hoofdstuk wordt de uitvoering van het onderzoek besproken aan de hand van de opgestelde deelvragen.

2.1 Welke tools & frameworks worden onderzocht?

Vanwege de beperkte tijd die beschikbaar is voor dit onderzoek kunnen niet alle tools onderzocht worden. Tijdens het vooronderzoek zijn een aantal methoden voor platformonafhankelijke ontwikkeling naar voren gekomen:

- Mobiele web applicatie
- Hybride applicatie
- Geïnterpreteerde applicatie
- Gegenereerde applicatie

Om de beste manier te bepalen om platformonafhankelijke applicaties te ontwikkelen wordt per methode een tool geselecteerd.

2.1.1 Mobiele web applicatie

Tijdens het vooronderzoek zijn de volgende frameworks voor mobiele web applicaties naar voren gekomen:

- Wink Toolkit
- jQuery mobile
- Sencha Touch
- The-M-Project
- jQTouch
- Jo

Om te bepalen welk framework gebruikt wordt voor het experiment is van elk van de frameworks een demo applicatie van de ontwikkelaars geprobeerd op een iPad en een Samsung Galaxy S. Opvallend was dat elk framework traag was als het gaat om animaties en het inladen van pagina's. Vaak flitsen elementen over het scherm wat een chaotische indruk geeft.

De frameworks die het minst last hadden van schokkerige animaties en andere grafische bugs waren jQuery mobile en Sencha Touch. Uiteindelijk is gekozen voor jQuery mobile om twee redenen. Ten eerste hebben jQuery mobile applicaties een (voor iOS) native look & feel, terwijl Sencha Touch applicaties dat niet hebben. Daarnaast is jQuery één van de meest gebruikte javascript frameworks [3]. Dit zorgt ervoor dat bij problemen snel hulp gevonden kan worden. jQuery mobile is slechts een van de vele extensies op het jQuery framework, waardoor de werking hetzelfde is.

2.1.2 Hybride applicatie

Tijdens het vooronderzoek zijn de volgende tools voor hybride applicaties naar voren gekomen:

- PhoneGap
- Rhodes

Omdat voor het experiment het in paragraaf 2.1.1 gekozen framework gebruikt gaat worden in combinatie met het in deze paragraaf gekozen tool is gekozen voor PhoneGap. PhoneGap kan een bestaande web applicatie direct verpakken tot een native applicatie. Omdat de PhoneGap API verder met javascript werkt is deze gemakkelijk te implementeren binnen de web applicatie. Rhodes werkt echter met Ruby waardoor de volledige applicatie opnieuw moet worden geschreven. Doordat PhoneGap meer gebruikt wordt dan Rhodes is het ook gemakkelijker om hulp te vinden wanneer een probleem zich voor doet [4].

2.1.3 Geïnterpreteerde applicatie

Tijdens het vooronderzoek zijn de volgende tools voor geïnterpreteerde applicaties naar voren gekomen:

- Appcelerator
- Corona

Op basis van de specificaties van het Proof of Concept (paragraaf 2.2) is de keuze gevallen op Appcelerator. Het grootste voordeel van Appcelerator is dat met de tool applicaties met een volledige native interface worden gemaakt. Corona daarentegen bevat vrijwel geen native interface elementen, maar is meer geschikt voor het maken van spellen.

2.1.4 Gegeneerde applicatie

Tijdens het vooronderzoek zijn de volgende tools voor gegeneerde applicaties naar voren gekomen:

- MoSync
- Marmalade
- Codename One

Omdat Codename One nog in de bèta fase zit, valt deze af voor het experiment. Zowel MoSync als Marmalade zijn interessante opties. Echter, op basis van het Proof of Concept ontwerp (paragraaf 2.2), is gekozen om met MoSync verder te gaan.

Voor het Proof of Concept wordt zoveel mogelijk met native gebruikersinterface elementen gewerkt. Marmalade is slechts kort geleden begonnen met de implementatie van native elementen, wel staat een uitgebreidere implementatie hiervan in de planning [5]. Zoals ook te zien in de showcases [6] wordt Marmalade momenteel vooral gebruikt om spellen te maken [7].

2.2 Welke functionaliteiten moet het Proof of Concept bevatten?

Het Proof of Concept moet dienen als een indicatie van de geschiktheid van een tool/framework om platformonafhankelijke mobiele applicaties te maken. Het Proof of Concept moet dus de functionaliteiten bevatten die bij veel applicaties gebruikt worden.

Het Proof of Concept is een applicatie bestaande uit vier hoofdschermen. Deze worden bereikt via tabbladen. Per tabblad worden de benodigde functionaliteiten geïmplementeerd. De tabbladen zien er als volgt uit:

1. Een lijst met personen waarbij geldt:
 - In de lijst worden de voornaam en achternaam weergegeven.
 - Personen kunnen aan de lijst worden toegevoegd.
 - Door een persoon te selecteren kunnen de details van het gekozen persoon worden bekeken.
 - De details van de personen kunnen worden gewijzigd.
 - Personen kunnen worden verwijderd.
 - Bij het verwijderen wordt om bevestiging gevraagd.
 - De personen worden lokaal opgeslagen.
2. Een (Google Map) kaart waarbij geldt:
 - De kaart begint op een locatie in de Verenigde Staten.
 - Een knop om naar de huidige locatie te gaan.
 - Op de huidige locatie staat een aangepast plaatje.
 - Wanneer het plaatje geselecteerd wordt verschijnt een pop-up.
3. Een scrollend scherm waarbij geldt:
 - Wanneer het apparaat beweegt wordt een regel toegevoegd aan het scherm.
 - Wanneer het scherm vol is scrolt het automatisch mee wanneer nieuwe tekst wordt toegevoegd.
4. Aanmeldformulier waarbij geldt:
 - De gebruiker vult een e-mail adres in.
 - Dit e-mail adres wordt naar een externe server gestuurd.
 - Op de externe server wordt het e-mail adres gevalideerd, wanneer het geldig is wordt het toegevoegd aan een database.
 - Op basis van het resultaat van de validatie en het toevoegen wordt een response teruggestuurd naar de applicatie.
 - De applicatie vangt de response van de server af en geeft een aangepast bericht op basis van de response.

Naast de functionaliteiten binnen de applicatie moet het ook mogelijk zijn om push berichten te versturen. Push berichten zijn kleine berichten die vanaf een server naar de telefoon *gepusht* worden. In het geval van het Proof of Concept moet een push bericht verstuurd worden wanneer iemand een e-mail adres aanmeldt.

Middels het hiervoor beschreven Proof of Concept worden dus de volgende functionaliteiten getest:

- (Google) Map met een aangepaste *overlay*
- Beweging detectie
- Tabbladen
- Menu navigatie
- GPS
- Transitie animaties
- *TableView*
- *ScrollView*
- Pop up dialoog
- Invoervelden / formulieren
- Lokale database
- Communicatie met een externe server (PHP)
- Push notificaties

Om de e-mailadressen op te slaan en de push notificaties te kunnen versturen moet ook een server ingericht worden. Op deze server staat een kleine database, bestaande uit twee tabellen:

- mailinglist
Deze tabel bevat slechts één kolom waarin de e-mailadressen worden opgeslagen.
- Notifications
Deze tabel bevat twee kolommen. In de ene kolom wordt de device token, ter identificatie van het apparaat, opgeslagen. In de tweede kolom wordt een code voor het besturingssysteem van het apparaat opgeslagen, zodat bepaald kan worden van bij welke push service de device token hoort.

Naast de database bevat de server twee PHP pagina's. De eerste pagina registreert nieuwe apparaten wanneer deze zich aanmelden voor de push service. De tweede pagina registreert e-mailadressen en verstuurt de push notificaties wanneer een nieuw e-mail adres zich aanmeldt.

2.3 Gebruikerservaring

Bij mobiele applicaties is de gebruikerservaring belangrijker dan ooit [8]. Dit wordt mede aangetoond door de eisen die de applicatiewinkels aan de gebruikerservaring stellen. Voornamelijk Apple (iOS) en Microsoft (Windows Phone) stellen specifieke eisen aan de gebruikerservaring [9] [10]. Dit doen ze omdat ze willen dat applicaties een verlengstuk worden van het platform.

Ook de gebruikers verwachten dat applicaties als verlengstuk van het platform gezien kunnen worden. Ze verwachten dat de applicatie werkt zoals de andere dat doen. Dit betekent dus dat gebruikers de look & feel behorende bij het platform verwachten. Wanneer dit afwijkt bestaat de kans dat de gebruiker ontevreden is over de applicatie.

Johannes Fahrenkrug heeft een interessant artikel geschreven over dit onderwerp [11]. Hierin vergelijkt hij het platformonafhankelijk ontwikkelen voor smartphones met het platformonafhankelijk ontwikkelen voor PCs. Hij refereert hiernaar om aan te geven dat geen enkele methode geslaagd is in het maken van platformonafhankelijke applicaties op één methode na; internet. Dat applicaties op het internet op elk platform worden geaccepteerd, ook al past de website zich niet aan het platform aan, komt wederom door de verwachting van de gebruikers. Op internet zijn gebruikers gewend geraakt aan verschillende gebruikerservaringen op verschillende websites.

2.4 Toekomstige ontwikkelingen

Op dit moment ondersteunen de mobiele browsers nog lang niet alle HTML5 functionaliteiten [12]. Daarnaast zijn ook nog lang niet alle HTML5 functionaliteiten volledig gespecificeerd. Naar verwachting zal HTML5 volledig gespecificeerd zijn in 2014 [13].

In een recent interview van ZDNet België met Jeff Prosise [14], zelfstandige in dienst van Microsoft, komt naar voren dat bij Windows 8, en dus ook Windows Phone 8, het internet meer geïntegreerd zit in het besturingssysteem. De ontwikkeling zal dan ook voornamelijk plaats vinden in HTML5, ondersteund door Javascript, waardoor applicaties direct uitwisselbaar zijn tussen de mobiele en de desktop versie.

Wel komt ook in het interview naar voren dat HTML5 nog niet ver genoeg ontwikkeld is om als standaard te dienen. Momenteel ontbreken nog de tools om goed te kunnen ontwikkelen, testen en debuggen. Ook de performance is nog niet evengoed op de verschillende platforms, zo is iOS op de iPhone 4s momenteel nog een stuk sneller dan Android telefoons [15].

Strategy Analytics heeft onderzoek gedaan naar de verwachte verkoop van HTML5 telefoons in de komende jaren [16]. Helaas is het rapport niet vrij inzichtelijk, maar in de samenvatting komt naar voren dat in 2013 naar verwachting één miljard HTML5 telefoons verkocht gaan worden. Ook hier wordt gemeld dat HTML5 nog in de kinderschoenen staat en dat het nog enkele jaren zal duren voordat het echt gebruikt kan worden om platformonafhankelijke applicaties te maken.

3 Experiment

Tijdens het experiment wordt het Proof of Concept (paragraaf 2.2) gebouwd met behulp van de geselecteerde tools en frameworks (paragraaf 2.1).

3.1 PhoneGap & jQuery Mobile

Binnen het eerste experiment wordt het Proof of Concept gebouwd met behulp van een combinatie van PhoneGap en jQuery Mobile. PhoneGap wordt binnen dit experiment gebruikt om van een website een native applicatie te maken.

3.1.1 PhoneGap introductie

PhoneGap is een ontwikkelplatform waarbij mobiele applicaties gemaakt kunnen worden met behulp van HTML, CSS en Javascript. PhoneGap ondersteund applicaties voor iOS, Android, BlackBerry, Windows Phone, WebOS en Symbian.

Voor elk platform heeft PhoneGap een aparte tool, plugin of framework, afhankelijk van de standaardontwikkelomgeving van het platform. Dit betekent dus dat het project voor elk platform in een andere omgeving moet worden opgezet. Omdat de projecten in HTML, CSS en Javascript worden gemaakt kunnen dezelfde bestanden voor elke omgeving gebruikt worden. De enige afwijking is dat het Javascript bestand, met daarin de PhoneGap abstractie laag, voor elk platform anders is. Het enige verschil tussen projecten voor verschillende platforms is dus dit Javascript bestand.

Omdat PhoneGap zelf slechts een *native wrapper* om een website heen zet moet de website dus het gevoel geven een echte mobiele applicatie te zijn. Om dit makkelijker te maken wordt binnen dit experiment naast PhoneGap ook jQuery Mobile gebruikt

3.1.2 jQuery Mobile introductie

Het jQuery Mobile framework is een uitbreiding op het standaard jQuery framework. jQuery wordt al veel gebruikt door web ontwikkelaars. Het jQuery Mobile framework voegt slechts functies toe aan het bestaande jQuery framework. Voor web ontwikkelaars verandert er dus weinig wanneer ze de stap naar mobiele websites nemen.

Doordat jQuery Mobile gebruik maakt van standaard Javascript werkt het op elk modern platform, dus ook in de desktop browsers. Alle platforms die door PhoneGap worden ondersteund worden dus ook door jQuery Mobile ondersteund. Door de applicatie ook online te zetten kunnen de apparaten die niet worden ondersteund door PhoneGap alsnog van de applicatie gebruik maken met behulp van hun webbrowser.

3.1.3 Leercurve

Doordat de combinatie PhoneGap en jQuery Mobile gebruik maken van HTML, CSS en Javascript kunnen webdevelopers zonder extra kennis gemakkelijk een mobiele applicatie maken. Omdat deze zelfde talen gebruikt worden voor het maken van websites zijn deze uitvoerig gedocumenteerd. Ook voor programmeurs zonder web ervaring is het dus niet moeilijk om te beginnen met het maken van mobiele applicaties.

3.1.4 Uitvoering

De functionaliteiten zijn op push notificaties na allemaal geïmplementeerd. Dankzij het jQuery Mobile framework was het gemakkelijk om de navigatiebalk (boven) en de tab balk (beneden) aan te maken.

Scherf 1: Personenlijst

Doordat vaste posities middels CSS3 slechts door enkele van de nieuwste mobiele apparaten ondersteund worden [17] verdwijnen de navigatie balken tijdens het scrollen. Dit is later, doormiddel van een plugin, opgelost (zie Scherm 3: Console).

Data opslag

Momenteel implementeren nog niet alle browsers de nieuwe standaard voor het lokaal opslaan van data. Voorheen was de standaard Web SQL storage, echter heeft W3C dit laten vallen als standaard. In plaats daarvan wordt nu IndexedDB als W3C standaard ondersteund. De ondersteuning is echter nog niet geïmplementeerd in de meeste (mobiele) browsers.

Om te zorgen dat de personenlijst in alle browsers opgeslagen kan worden is voor dit experiment gebruik gemaakt van local storage. In plaats van een database is dit slechts een dictionary waarin strings, gekoppeld aan een key, kunnen worden opgeslagen. Om dit te kunnen gebruiken als database is een tussenlaag gemaakt in Javascript waar de objecten omgezet worden in een JSON string. Wanneer objecten uit de database opgehaald worden, wordt deze JSON string geparsed en kan door de objecten geïtereerd worden.

Wisselen van schermen

Wanneer op de knop toevoegen werd geklikt werd een tweede HTML pagina geopend. Bij het opslaan wordt de ingevoerde persoon aan de database toegevoegd en wordt de lijst pagina weer geopend. De opslaan functie moet worden gedefinieerd in de 'pageinit' of in de 'pageshow' functie van jQueryMobile. Deze functies werden echter telkens bij weergave aangeroepen, terwijl de opslaan functie nog gecached was. Daardoor werd de opslaan functie steeds een extra keer aangeroepen. Bijvoorbeeld bij de derde persoon die ingevoerd wordt, wordt ook de opslaan functie drie keer aangeroepen. Daardoor wordt de toegevoegde persoon dus ook drie keer opgeslagen.

Dit is opgelost door de complete applicatie in één HTML file te schrijven. Door achter de URL een hashtag met de geopende pagina mee te geven selecteert het jQuery Mobile framework automatisch de juiste pagina. De opbouw van de pagina is te zien in Figuur 3.1

```
<body>
  <div class=page id=namen>
    ...
  </div>
  <div class=page id=toevoegen>
    ...
  </div>
</body>
```

```
<body>
  <div class=page id=namen>
    ...
  </div>
  <div class=page id=toevoegen>
    ...
  </div>
</body>
```

Figuur 3.1 Pagina's middels div tags

Door in dit geval “index.html#namen” aan te roepen wordt de namen pagina weergegeven. Door “index.html#toevoegen” aan te roepen wordt de toevoegen pagina weergegeven. Op deze manier worden de pageinit methodes slechts één keer aangeroepen.

Scherf 2: Kaart

Doordat Google een uitgebreide API heeft om kaarten weer te geven is de kaart gemakkelijk geïmplementeerd. Het grootste probleem met de kaart is dezelfde plugin die de oplossing bracht bij de personenlijst en de console. Doordat deze plugin de pagina divs automatisch omzet naar scrollviews reageren zowel de kaart als de achtergrond op scrol bewegingen. Wanneer de gebruiker nu over de kaart probeert te scrollen, scrolt tevens de achtergrond mee. Helaas is de enige oplossing hiervoor de plugin handmatig aan te passen en het stuk te vervangen waar pagina's automatisch scrollviews worden. Hiervoor in de plaats zou dan handmatig aangegeven moeten worden welke pagina's moeten kunnen scrollen. Dit betekend dus dat dit voor elke applicatie apart gedaan zou moeten worden. Ook moet bij een eventuele update van de plugin dit voor elke applicatie weer opnieuw gedaan worden.

De PhoneGap API bevat geen mogelijkheid om een native kaart weer te geven, waardoor dus altijd de API van Google gebruikt moet worden. Dit zorgt ervoor de kaart trager inlaadt. Soms laadt de kaart zelfs niet helemaal in, waardoor grijze vlakken in beeld blijven.

Om naar de huidige locatie te gaan wordt gebruik gemaakt van de PhoneGap API.

Scherf 3: Console

Standaard bevat jQuery Mobile geen mogelijkheid om een scrollview te maken. Middels een plugin is het echter wel mogelijk om inline scrollen mogelijk te maken. Door dit scrollview tussen de menubalken te plaatsen blijven deze dus staan wanneer inline in het scrollview gescrold wordt.

De meeste browsers hebben nog geen ondersteuning voor beweging sensoren, daarom is voor deze functionaliteit gebruik gemaakt van de PhoneGap API.

Scherf 4: Mailinglist

Middels het jQuery framework kan een POST request naar een externe PHP pagina gestuurd worden. De inhoud van die pagina kan vervolgens afgevangen worden zodat op basis daarvan een melding weergegeven kan worden.

Push Notificaties

PhoneGap bevat momenteel geen API om een apparaat te registreren voor push notificaties, of om push notificaties te ontvangen. Er bestaan wel externe systemen die helpen bij het implementeren hiervan, maar ook met deze plugins moeten de push notificaties native geïmplementeerd worden.

Het is dan ook makkelijk om deze geheel handmatig te implementeren dan via deze systemen werken.

Omdat het binnen dit onderzoek gaat om platformonafhankelijke ontwikkeling zijn de push notificaties niet native geïmplementeerd, gezien dat juist hetgeen is dat voorkomen moet worden.

3.1.5 Geheugengebruik

Op iOS ligt de top van het geheugengebruik net boven de 10 MB. Daarbij is geen geheugen gelekt. Vooral wanneer de Google map in beeld is stijgt het geheugen gebruik. Wanneer de kaart weer uit beeld verdwijnt daalt het geheugengebruik met ongeveer 3 MB. Op Android ligt de top van het geheugengebruik op ongeveer 30 MB. Hierbij is niet bekend hoeveel geheugen gelekt is.

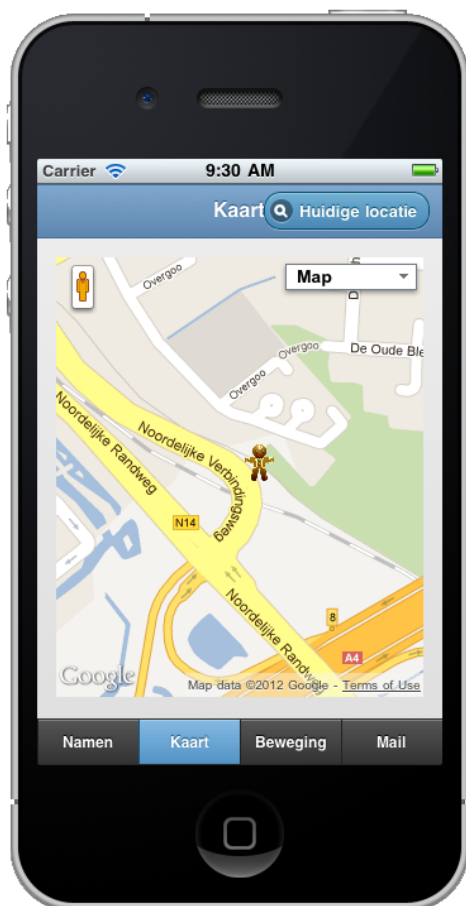
3.1.6 Conclusie

Helaas is het momenteel nog erg duidelijk te zien dat de applicatie slechts een webview met daarin een webpagina is. De animaties zijn traag en schokkerig en soms verspringen de pagina's heel even. Mobiele browsers worden steeds sneller, en ook de specificaties van de apparaten worden steeds beter, waardoor de applicaties in de toekomst waarschijnlijk soepeler zullen lopen.

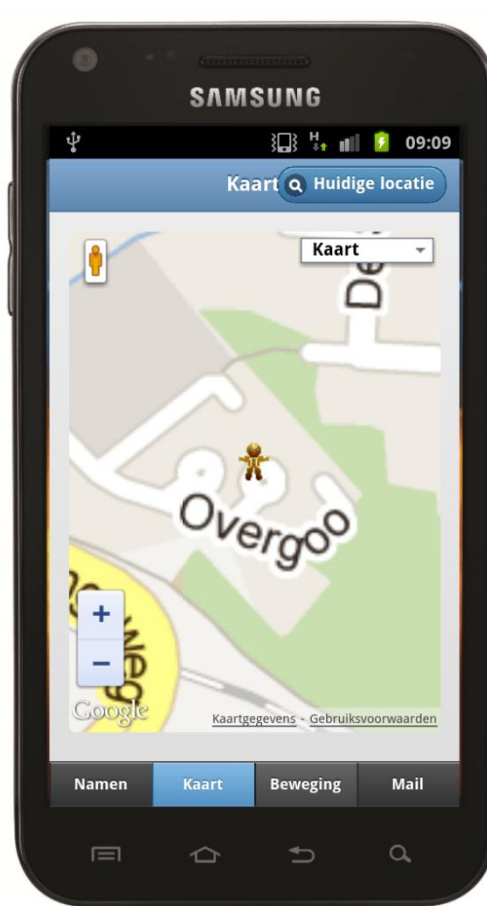
Het gebruik van de lokale opslag van de browsers is riskant. Wanneer de gebruiker de privégegevens wist worden ook de lokale databases weggegooid. Hierdoor raken alle gegevens verloren.

Doordat de applicatie toch duidelijk een website is, is het wellicht beter om de gehele website online te draaien en deze dan eventueel aan te roepen binnen PhoneGap, zodat de applicatie wel verpakt kan worden als native applicatie. Door de applicatie online te draaien kunnen ook complexere applicaties ontwikkeld worden, door bijvoorbeeld PHP te gebruiken voor de applicatie en de HTML, CSS en Javascript slechts te gebruiken voor de weergave.

Het voordeel van PhoneGap is tegelijkertijd ook het grootste nadeel; de applicatie is op elk platform hetzelfde (Figuur 3.2 en Figuur 3.3). Dit lijkt in eerste instantie een groot voordeel, maar in de praktijk kan dit gebruikers afschrikken. Gebruikers verwachten dat de applicatie exact zo werkt als de rest van hun applicaties, waaronder de standaard applicaties van het platform. Hier wordt dieper op ingegaan in paragraaf 2.3.



Figuur 3.2 PhoneGap op iOS



Figuur 3.3 PhoneGap op Android

3.2 Appcelerator Titanium

Voor het tweede experiment wordt het Proof of Concept gebouwd met behulp van Appcelerator Titanium. Met Appcelerator Titanium worden applicaties gemaakt met Javascript. Door de uitgebreide Javascript API kunnen native gebruikersinterface elementen worden aangemaakt. Appcelerator Titanium ondersteunt momenteel iOS, Android en BlackBerry, hoewel officieel nog in bèta is.

3.2.1 Leercurve

Doordat Appcelerator Titanium gebruik maakt van Javascript is het voor web ontwikkelaars makkelijk om over te stappen. Javascript is daarnaast een gemakkelijke taal om te leren, dus ook mensen zonder Javascript ervaring zullen weinig moeite hebben met het gebruik van Appcelerator Titanium.

Appcelerator Titanium heeft een uitgebreide API waar gemakkelijk mee geprogrammeerd kan worden. De auto aanvul functie van de meegeleverde IDE maakt het gemakkelijk om wegwijs te worden.

Leercurve is redelijk laag, gezien het javascript betreft....Ook framework (API) is redelijk makkelijk te begrijpen, doordat de IDE niet altijd goede autocorrections heeft moet wel vaak in de api docs gezocht worden -> deze zijn niet altijd goed gedocumenteerd

3.2.2 Uitvoering

Het experiment met Appcelerator Titanium begon al direct met een probleem. Om een project voor meerdere platforms te maken moeten deze platforms bij het aanmaken van het project geselecteerd worden. Het is echter niet mogelijk om iOS applicaties te initiëren op een Windows computer, terwijl het dan weer niet mogelijk is om BlackBerry applicaties te initiëren op een Mac. Voor dit experiment is het project geïnitieerd op de Mac en is BlackBerry niet meegenomen, mede doordat dit nog in bèta is.

Daarnaast viel als snel op dat de auto aanvul functie van de IDE niet goed werkt wanneer objecten aangemaakt zijn. Functies en eigenschappen van deze objecten moeten dus steeds opgezocht worden in de API reference. Hierin zijn eigenschappen en functies gemakkelijk te vinden, echter is de uitleg daarbij erg mager waardoor het ontwikkelen geremd wordt.

Platform afhankelijke code

Binnen het framework zijn sommige functionaliteiten wel beschikbaar op beide platforms, andere zijn weer platform specifiek. Daardoor gebeurt het regelmatig dat doormiddel van if-statements code moet worden aangeroepen afhankelijk van het huidige platform.

Een voorbeeld hiervan is de navigatie. Binnen iOS wordt gebruik gemaakt van een navigatiebalk aan de bovenkant van het scherm. Hierin kan bijvoorbeeld een knop zitten om nieuwe personen toe te voegen. Binnen Android wordt geen gebruik gemaakt van de navigatiebalk. In plaats daarvan hebben Android toestellen een menu knop, wanneer daarop gedrukt wordt verschijnen de menu opties in beeld.

Het gebeurt ook dat sommige events anders werken op de verschillende platforms. Zo wordt een event afgevuurd wanneer de gebruiker van tabblad wisselt. Echter op Android wordt dit event niet afgevuurd zoals het zou moeten. De event afhandeling moest dus specifiek voor iOS gemaakt worden, terwijl voor Android een tijdelijke oplossing nodig was.

Scherf 1: Personenlijst

Middels de Appcelerator Titanium API kan een lijst object worden aangemaakt. Dit lijst object wordt door Appcelerator Titanium automatisch vertaald naar een native lijst afhankelijk van het platform waarvoor wordt gecompileerd.

De data wordt opgeslagen in een SQLite database. Deze maakt het mogelijk om standaard SQL query's uit te voeren. Schermen worden aangemaakt in een 'constructor'. Voor elk scherm wordt een apart bestand gemaakt met daarin de constructor van dit scherm. Hierin wordt een 'self' variabele aangemaakt waarin een Appcelerator Titanium window object wordt gestopt. Hieraan worden de elementen van het scherm en de event handlers aan toegevoegd. Daarna wordt het scherm teruggegeven.

Scherf 2: Kaart

Appcelerator Titanium maakt gebruik van de kaart API van het platform waarvoor gecompileerd wordt. Daardoor kan een native kaart worden gebruikt, welke sneller werkt dan een online kaart. Het toevoegen van annotaties is simpel, een tap op de annotatie af vangen is echter weer platform specifiek. Via de API is het namelijk niet mogelijk om bij iOS een tap op de annotatie te detecteren, iets wat in native code wel kan. Wel is het mogelijk om een titel en twee knoppen te laten verschijnen wanneer de gebruiker de annotatie selecteert. Als oplossing is nu van deze mogelijkheid gebruik gemaakt door een knopje toe te voegen en de verdere afhandeling daaraan te hangen.

Scherf 3: Console

De beweging detectie met een scrollview waren gemakkelijk aangemaakt. Echter was ook hier weer een stukje platforms specifieke code nodig. De positie van het scrollview wordt voor iOS bepaald door het bovenste punt, terwijl dit bij Android juist bepaald wordt door het onderste punt. De berekening moest dus per platform verschillend.

Scherf 4: Mailinglist

Het framework bevat de mogelijkheid om een POST request te sturen en het antwoord op te vangen. De aanmeldfunctie was dus geen probleem.

Push notificaties

Appcelerator Titanium bevat gedeeltelijk ondersteuning voor push notificaties. Het is wel mogelijk om deze voor iOS te maken, echter is er nog geen API om push notificaties voor Android te implementeren. Het implementeren van de push notificaties voor iOS is verder zonder problemen verlopen.

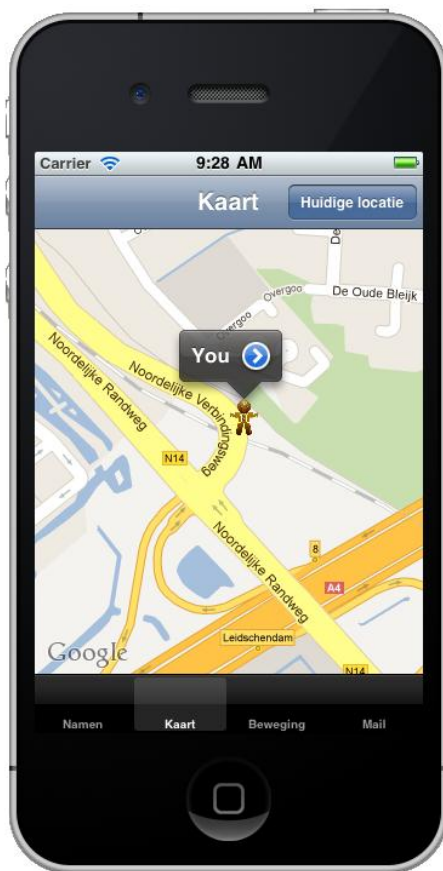
3.2.3 Geheugengebruik

Op iOS ligt de top van het geheugengebruik op ongeveer 4,75 MB. Daarbij zijn 250 bytes gelekt. Na nog vijf minuten gebruik is het geheugen niet gestegen en ook zijn er ook niet meer bytes gelekt. Opvallend is dat er nooit geheugen wordt vrijgegeven. Op Android ligt de top van het geheugengebruik op ongeveer 27,75 MB. Hierbij is niet bekend hoeveel geheugen gelekt is.

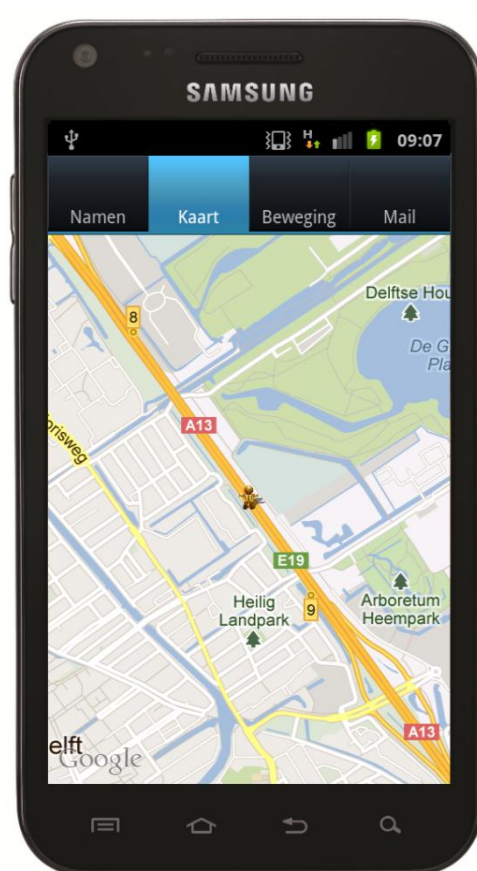
3.2.4 Conclusie

Dankzij het gebruik van standaard interface elementen creëert Appcelerator Titanium een native look & feel (Figuur 3.4 en Figuur 3.5). Daardoor zijn de applicaties niet te onderscheiden van native applicaties. De platform specifieke look & feel zorgt er echter wel voor dat sommige code op meerdere manieren geschreven moet worden. Dit zorgt dus wel weer voor opsplitsing van code waardoor de applicatie minder platformonafhankelijk wordt.

De grootste zorg is echter het geheugengebruik. Ondanks dat op iOS het geheugengebruik mee viel, viel het wel op dat er geen geheugen vrijgegeven werd wanneer schermen werden gesloten. Voor een kleine applicatie als deze is dat geen probleem, echter bij grotere applicaties kan dit geheugenproblemen en daarmee crashes veroorzaken. De geheugen lekken vallen mee, op basis van klachten [18] zag het er naar uit dat Appcelerator Titanium veel geheugen zou lekken. Het is natuurlijk wel mogelijk dat bij grotere applicaties meer geheugen lekt.



Figuur 3.4 Appcelerator Titanium op iOS



Figuur 3.5 Appcelerator Titanium op Android

3.4 MoSync

Voor het derde experiment wordt het Proof of Concept gebouwd met MoSync. Binnen MoSync zijn veel verschillende mogelijkheden om applicaties te schrijven. De volgende (combinaties van) talen zijn mogelijk:

- C Basic Project
- C Newlib Project
- C NativeUI Project
- Empty Project
- C++ Moblet Project
- C++ STL Project
- C++ NativeUI Project
- C++ OpenGL Project
- HTML5/JS/C++ Hybrid Project
- HTML5 JS NativeUI Project
- HTML5/JS WebUI Project

Vanwege de beperkte tijd voor dit onderzoek kon slechts één van deze mogelijkheden gebruikt worden voor dit experiment. Omdat gekozen is zoveel mogelijk gebruik te maken van native gebruikersinterface elementen is gekozen voor “C++ NativeUI Project”.

3.4.1 Leercurve

De leercurve voor is veel hoger dan voor PhoneGap en Appcelerator. Onder andere omdat C++ een lastigere taal is dan bijvoorbeeld javascript. Een groter probleem is echter de beperkte documentatie. Het is bijvoorbeeld niet altijd duidelijk welke headers geïmporteerd moeten worden. Ook blijkt dat de standaard build settings aangepast moeten worden, dit komt niet terug in de MoSync documentatie.

3.4.2 Uitvoering

Tijdens de uitvoering van het experiment kwamen al snel een aantal grote problemen naar voren. Zo kan wel een navigatiebalk worden toegevoegd aan de schermen, echter kan hierin geen knop aan de rechter bovenkant worden toegevoegd. Ook kan de kleur van de balk niet worden veranderd. Daardoor is het gebruik van kleuren voor de gehele applicatie ook direct beperkt, omdat andere kleuren wel bij de navigatiebalkkleur moeten passen.

Strings

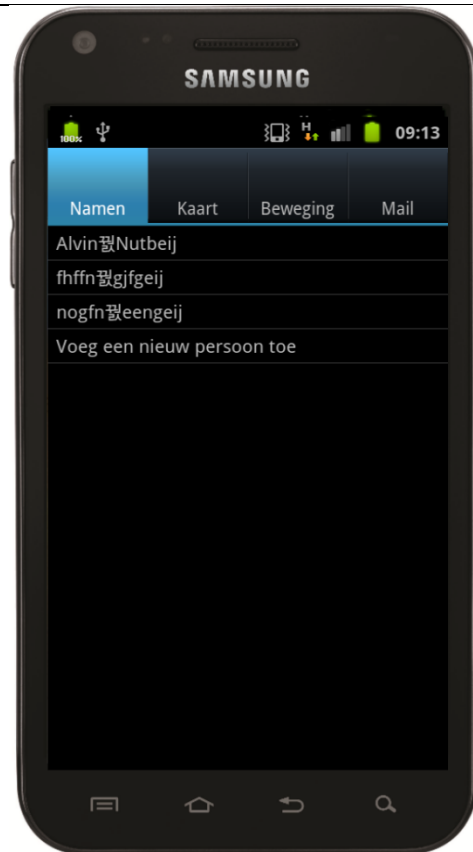
Het MoSync framework bevat een eigen datatype voor strings. Dit datatype wordt voor veel functies gebruikt als input parameter of als return type, dus om het framework te gebruiken moet ook het datatype gebruikt worden. Wanneer een enkele string variabele vaker gebruikt wordt geeft dit echter een probleem. Bijvoorbeeld de loop waarin namen aan de namenlijst worden toegevoegd.

Deze loop gaf meerdere problemen. Het eerst probleem is een vreemd teken dat over de spatie tussen de voor en achternaam wordt gezet, zoals te zien in Figuur 3.6. In eerste instantie leek een verborgen karakter achter de voornaam te staan. Toen echter de lengte van de voornaam aan de totale string werd toegevoegd (Figuur 3.7) verdween het vreemde teken. Wanneer dit aantal weer werd verwijderd kwam het teken weer terug.

```

List<Person *> people = db->getAllPeople();
List<Person*>::ListIterator it = people.begin();
Person *currentPerson;
ListViewItem *item;
String text;
String lastName;
while(it.hasNext()){
    currentPerson = it.next();
    item = new ListViewItem();
    text = currentPerson->getFirstName();
    lastName = currentPerson->getLastName();
    text.append(" ",1);
    text.append(lastName.c_str(), lastName.length());
    item->setText(text);
    lijst->addChild(item);
}

```



Figuur 3.6 MoSync vreemd teken bij strings in een loop code & resultaat

```

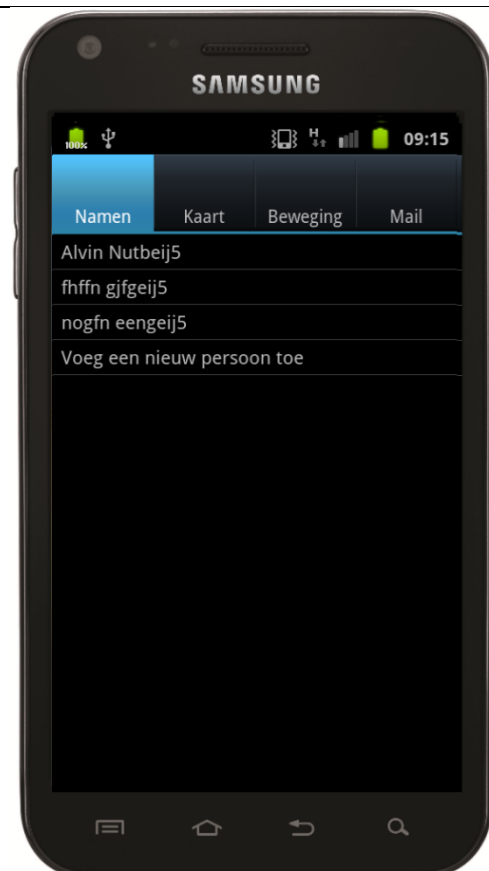
List<Person *> people = db->getAllPeople();
List<Person*>::ListIterator it = people.begin();
Person *currentPerson;
ListViewItem *item;
String text;
String lastName;
while(it.hasNext()){

    currentPerson = it.next();
    item = new ListViewItem();
    text = currentPerson->getFirstName();
    lastName = currentPerson->getLastName();
    text.append(" ",1);
    text.append(lastName.c_str(), lastName.length());

    text.append(
        integerToString(
            currentPerson->getFirstName().length()
        ).c_str(),
        integerToString(
            currentPerson->getFirstName().length()
        ).length()
    );

    item->setText(text);
    lijst->addChild(item);
}

```



Figuur 3.7 MoSync strings in loop code & resultaat met voornaam lengte

Naast de vreemde tekens is in Figuur 3.6 en Figuur 3.7 nog een fout te zien, de laatste letters van de eerste achternaam staan ook bij de rest van de achternamen. Het ziet er naar uit dat dit ontstaat doordat bij het ophalen van de naam de string eerst leeg wordt gemaakt, maar gewoon wordt overschreven. Wanneer een nieuwe string dus korter is dan de oude blijft het restant van de oude string gewoon staan. De oplossing hiervoor leek simpel, de string leeg maken voordat deze opnieuw een waarde wordt toegekend. De code hiervan is te zien in Figuur 3.8. De clear functie van het string datatype veranderd echter niets aan de uitvoer.

```
List<Person *> people = db->getAllPeople();
List<Person*>::ListIterator it = people.begin();
Person *currentPerson;
ListViewItem *item;
String text;
String lastName;
while(it.hasNext()){
    lastName.clear();
    text.clear();
    currentPerson = it.next();
    item = new ListViewItem();
    text = currentPerson->getFirstName();
    lastName = currentPerson->getLastName();
    text.append(" ",1);
    text.append(lastName.c_str(), lastName.length());
    item->setText(text);
    lijst->addChild(item);
}
```

Figuur 3.8 MoSync reset string waarde in loop code

Op basis van de eerder genoemde problemen in combinatie met de gebrekkige documentatie is besloten het experiment met MoSync te stoppen.

3.4.3 Conclusie

Op basis van het experiment met C++ NativeUI is MoSync nog niet geschikt om te gebruiken als tool voor platformonafhankelijke applicatieontwikkeling. Op dit moment heeft MoSync teveel beperkingen en een te slechte documentatie. Mogelijk dat één of meerdere van de andere type projecten beter geschikt zijn.

4 Conclusie en aanbevelingen

In dit hoofdstuk worden de conclusie en de aanbevelingen naar aanleiding van het onderzoek besproken. Daarnaast wordt aandacht besteed aan de beperkingen van dit onderzoek.

4.1 Beperkingen

Vanwege de beperkte tijd die voor dit onderzoek beschikbaar was zijn niet alle tools en frameworks aan bod gekomen. Tijdens het onderzoek zijn alleen jQuery mobile, PhoneGap, Appcelerator Titanium en een klein deel van MoSync aan bod gekomen.

Daarnaast ontwikkelen de tools en frameworks voor platformonafhankelijke applicatie ontwikkeling zich dusdanig snel dat een aantal tools tijdens dit onderzoek met belangrijke updates zijn gekomen. Hierop wordt dieper ingegaan in paragraaf 4.3, Aanbeveling naar aanleiding van het onderzoek.

4.2 Conclusie

Op basis van dit onderzoek is te concluderen dat de ontwikkelingen op het gebied van platformonafhankelijke applicatie ontwikkeling veel belovend zijn. De experimenten hebben laten zien dat voor verschillende situaties verschillende tools de beste oplossing zijn.

Een combinatie van een mobiel web framework en PhoneGap is bijvoorbeeld zeer bruikbaar wanneer een bestaande (mobiele) website omgezet wordt naar een app. Het gebruik van de website voelt dan aan als een applicatie, echter is het gewoon een website waardoor het ook parallel aan de normale website kan draaien. Daarmee dus gebruik makend van dezelfde serverside code.

Het nadeel van een PhoneGap applicatie is echter dat het zeer aannemelijk is dat de applicatie geweigerd wordt door Apple en nog aannemelijker dat deze geweigerd wordt door Microsoft omdat de gebruikerservaring niet native is. Voor Android zal het echter geen problemen opleveren. Daarnaast zijn de huidige web standaarden nog niet geschikt om zwaardere (grafische) applicaties te maken.

Wanneer het wel een echte native applicatie betreft, maar met weinig ingewikkelde logica en zonder al te veel grafische eisen, is het goed mogelijk om deze te maken met behulp van Appcelerator Titanium. Momenteel is dit nog beperkt tot Android en iOS maar door slechts gebruik te maken van javascript kan wel een volledig native applicatie neergezet worden, inclusief de native gebruikersinterface elementen.

Het nadeel van een Appcelerator Titanium applicatie is echter dat de afhankelijkheid van het javascript framework zeer groot is. Daardoor komen al snel beperkingen kijken zoals het niet aan kunnen passen van kleurtinten van de native gebruikersinterface elementen.

MoSync is duidelijk de meest krachtige oplossing die binnen de experimenten naar voren is gekomen. Doordat met een taal als C++ geprogrammeerd kan worden kan dit volledig object georiënteerd gedaan worden. Echter, door de gebrekkige documentatie en de beperkingen aan de native gebruikersinterface is het nog niet geschikt om alledaagse applicaties te schrijven. Wel zal deze tool geschikt kunnen zijn om applicaties te maken die niet afhankelijk zijn van deze native gebruikersinterface, zoals spellen.

4.3 Aanbeveling naar aanleiding van het onderzoek

Op basis van het onderzoek kunnen een aantal algemene aanbevelingen gedaan worden. Omdat de opdrachtgever momenteel twee mobiele applicaties in ontwikkeling heeft zal voor deze applicaties apart een advies worden gegeven.

4.3.1 Algemeen

Momenteel zijn de tools en frameworks om platformonafhankelijke applicaties te maken nog niet ver genoeg om volledig over te stappen. Het is dan ook aan te bevelen om applicaties in native code te schrijven.

De tools en frameworks bieden momenteel nog niet de gebruikerservaring die juist zo belangrijk is bij mobiele applicaties [11]. Ongeacht hoe goed een applicatie in elkaar zit is de kans dat de gebruiker de applicatie naast zich neer legt groot wanneer de gebruikerservaring niet optimaal is.

Uitzondering hierop zijn websites en spellen. Wanneer voor een applicatie platformonafhankelijkheid vereist is kan een web applicatie uitkomst bieden. Deze applicatie kan dan online draaien en bereikt worden via iedere browser. De applicatie omzetten naar een native applicatie met behulp van een tool als PhoneGap is echter af te raden in verband met de verwachtingen van de gebruiker (zie 2.3 Gebruikerservaring).

Op het gebied van spellen lijkt Marmalade de beste keuze te zijn. Gezien spellen niet behandeld zijn binnen dit onderzoek is hier verder onderzoek voor vereist. Ook de keuze voor een web applicatie framework kan nog dieper onderzocht worden.

Verder onderzoek

Door de hoeveelheid tools die in het vooronderzoek naar voren zijn gekomen was het voor dit onderzoek niet mogelijk om alle tools te evalueren. Daarnaast gaat de ontwikkeling van deze tools zo snel dat een aantal tools tijdens het onderzoek met (grote) updates zijn gekomen (Figuur 4.1). In de tabel is Corona niet opgenomen omdat hier geen grote releases worden gedaan, echter worden vrijwel dagelijks updates doorgevoerd [19].

Tool / Framework	Van versie	Naar versie	Verandering
PhoneGap	1.4.1	1.5.0	<ul style="list-style-type: none"> • Voornamelijk veel kleine updates • iOS geheugenfouten • Android API versimpeld
Appcelerator Titanium	1.8.1	1.8.2	<ul style="list-style-type: none"> • Kleine updates • Aankondiging van versie 2.0
MoSync	2.7	3.0	<ul style="list-style-type: none"> • Windows Phone 7 ondersteuning • Wormhole JS Library – PhoneGap applicaties in MoSync • Javascript native gebruikersinterface • Lokale/push notificaties • Vele andere veranderingen [20]
Marmalade	5.2	5.2.3	<ul style="list-style-type: none"> • Geen belangrijke updates tot nog toe, echter staat versie 6.0 nog steeds in de planning voor het eerste kwartaal van 2012 [5].
Codename One	Private Beta	Open Beta	<ul style="list-style-type: none"> • Kan vanaf nu vrij gedownload worden
Sencha Touch	1.1	2.0	<ul style="list-style-type: none"> • De snelheid van laden en reageren op een verandering van oriëntatie versneld [21] • Vele andere veranderingen [22]
jQuery Mobile	1.0.1	1.1.0 RC1	<ul style="list-style-type: none"> • Navigatiebalken op vaste posities • Soepelere en nieuwe animaties • Ondersteuning voor jQuery 1.7.1 • Vele andere veranderingen [23]

Figuur 4.1 Updates gedurende het onderzoek

Afhankelijk van het doel van een applicatie zouden verschillende oplossingen geschikt kunnen zijn, zoals al werd geconcludeerd in paragraaf 4.2. Om echter zeker te kunnen stellen zou er afhankelijk van het doel van de applicatie dus verder onderzoek gedaan moeten worden naar de tools die gebruikt kunnen worden.

Op het gebied van web applicaties zijn jQuery Mobile en Sencha Touch de meest actief onderhouden frameworks. Tijdens het selecteren van de tools was de performance van jQuery Mobile beduidend beter dan die van Sencha Touch. Echter door de grote update van Sencha Touch kan dit mogelijk veranderd zijn.

4.3.2 MOFLAS

Op basis van het onderzoek is het voorlopig aan te bevelen om MOFLAS in native-code te blijven schrijven. Naast het gebruik van native gebruikersinterface elementen worden ook zware bewerkingen uitgevoerd. Met een 3D functionaliteit in de planning is het simpelweg niet mogelijk om een web applicatie te maken.

Ook is de MOFLAS applicatie in eerste instantie gericht om te laten zien wat mogelijk is met een mobiele applicatie. Door naar een platformonafhankelijke oplossing over te stappen wordt ten aller tijden ingeleverd op native functionaliteiten. Om het maximale uit de applicatie te halen is het advies dus native code te schrijven.

4.3.3 Restaurant App

Op basis van het onderzoek zijn goede mogelijkheden te zien om de restaurant app platformonafhankelijk te ontwikkelen. Momenteel is de applicatie geschreven in native Objective-C code. Daarnaast is een variant geschreven met behulp van jQuery Mobile, deze is echter nog niet volledig up-to-date met de native applicatie.

Het doel van deze applicatie is het bestellen in restaurants te vergemakkelijken. Om hiermee zoveel mogelijk gebruikers aan te spreken is het belangrijk deze applicatie op meerdere platformen uit te brengen. Doordat de applicatie al gedeeltelijk in jQuery Mobile gemaakt is kost het relatief weinig moeite om deze variant uit te breiden en om te zetten naar een native applicatie met behulp van PhoneGap.

Voor iOS kan dan de native applicatie gebruikt blijven worden terwijl alle andere platformen de web variant draaien. Daarnaast maakt de jQuery Mobile versie het ook mogelijk om een online versie te draaien, waardoor gebruikers slechts een browsers nodig hebben om te bestellen. De online versie kan dan onder andere als back-up dienen in het geval dat Microsoft de PhoneGap variant weigert op te nemen in de Windos Phone Marketplace.

5 Bibliografie

- [1] Canalys, „Smart phones overtake client PCs in 2011,” 3 februari 2012. [Online]. Available: <http://www.canalys.com/newsroom/smart-phones-overtake-client-pcs-2011>. [Geopend 28 februari 2012].
- [2] ABI Research, „Mobile Application Business Models,” 16 februari 2012. [Online]. Available: <http://www.abiresearch.com/press/3847-In-App+Purchases+to+Outpace+Pay-Per-Download+Revenues+in+2012>. [Geopend 28 februari 2012].
- [3] I. Elliot, „jQuery is the top JavaScript Framework,” I Programmer, 7 augustus 2011. [Online]. Available: <http://www.i-programmer.info/news/167-javascript/2857--jquery-is-the-top-javascript-framework.html>. [Geopend 15 februari 2012].
- [4] „Stackoverflow - zoeken met de termen 'Rhodes' en op 'PhoneGap',” [Online]. Available: <http://www.stackoverflow.com>. [Geopend 15 februari 2012].
- [5] Ideaworks 3D, „Releases and Roadmap,” Ideaworks 3D, 2012. [Online]. Available: <http://www.madewithmarmalade.com/marmalade/releases-and-roadmap>. [Geopend 16 februari 2012].
- [6] Ideaworks 3D, „Showcases,” Ideaworks 3D, 2012. [Online]. Available: <http://www.madewithmarmalade.com/marmalade/showcase>. [Geopend 9 februari 2012].
- [7] how, „Is Marmalade SDK more relevant or commonly used for Game Development?,” Stackoverflow, 11 december 2011. [Online]. Available: <http://stackoverflow.com/questions/8415387/is-marmalade-sdk-more-relevant-or-commonly-used-for-game-development>. [Geopend 7 maart 2012].
- [8] K. Bjoran, „User Expectations with Mobile Apps - Catching up with EffectiveUI,” 21 december 2010. [Online]. Available: <http://www.uxbooth.com/blog/12207/>. [Geopend 28 februari 2012].
- [9] Apple, „iOS Human Interface Guidelines,” Apple, 12 oktober 2011. [Online]. Available: https://developer.apple.com/library/ios/#documentation/UserExperience/Conceptual/MobileHIG/Introduction/Introduction.html#//apple_ref/doc/uid/TP40006556-CH1-SW1. [Geopend 10 februari 2012].
- [10] Microsoft, „Windows Phone Development,” Microsoft, 15 december 2011. [Online]. Available: [http://msdn.microsoft.com/en-us/library/ff402535\(v=vs.92\).aspx](http://msdn.microsoft.com/en-us/library/ff402535(v=vs.92).aspx). [Geopend 10 februari 2012].
- [11] J. Fahrenkrug, „Thoughts on Mobile UI Design,” Springenwerk, 12 september 2011. [Online]. Available: <http://www.springenwerk.com/2011/09/thoughts-on-mobile-ui-design.html>. [Geopend 27 februari 2012].
- [12] W3C, „Standards for Web Applications on Mobile: February 2012 current state and roadmap,”

- W3C, februari 2012. [Online]. Available: <http://www.w3.org/2012/02/mobile-web-app-state/>. [Geopend 27 februari 2012].
- [13] W3C, „FAQs - HTML5 Wiki,” W3C, 22 februari 2011. [Online]. Available: http://www.w3.org/html/wiki/FAQs#When_will_HTML5_be_done.3F. [Geopend 28 februari 2012].
- [14] J. v. Oost, „Toekomst is aan HTML5-vormgevers/programmeurs,” ZDNet, 21 februari 2012. [Online]. Available: <http://www.zdnet.be/techzone/136580-1/-toekomst-is-aan-html5-vormgevers-programmeurs/>. [Geopend 28 februari 2012].
- [15] J. Joraanstad, „HTML5 and the Future of Mobile Apps and Gaming,” techerator, 1 februari 2012. [Online]. Available: <http://www.techerator.com/2012/02/html5-and-the-future-of-mobile/>. [Geopend 27 februari 2012].
- [16] N. Shah, „Global HTML5 Handset Sales Forecast,” Strategy Analytics, 7 december 2011. [Online]. Available: <http://www.strategyanalytics.com/default.aspx?mod=pressreleaseviewer&a0=5145>. [Geopend 27 februari 2012].
- [17] M. Firtman, „Mobile HTML5,” [Online]. Available: <http://mobilehtml5.org/>. [Geopend 15 februari 2012].
- [18] A. Dallera, „Why you should stay away from Appcelerator's Titanium,” 14 juni 2011. [Online]. Available: <http://usingimho.wordpress.com/2011/06/14/why-you-should-stay-away-from-appcelerators-titanium/>. [Geopend 21 februari 2012].
- [19] AnscA, „Corona Daily Builds Summary,” AnscA, 10 maart 2012. [Online]. Available: <http://developer.anscamobile.com/corona-daily-builds/summary>. [Geopend 12 maart 2012].
- [20] C. Hughes, „What's New in MoSync 3.0,” MoSync, 22 februari 2012. [Online]. Available: <http://www.mosync.com/documentation/release-notes/whats-new-mosync-30>. [Geopend 12 maart 2012].
- [21] J. Avins, „Sencha Touch 2.0 - Built for Amazing Apps,” Sencha, 6 maart 2012. [Online]. Available: <http://www.sencha.com/blog/announcing-sencha-touch-2/>. [Geopend 6 maart 2012].
- [22] Sencha, „Sencha Touch 2 Release Notes,” Sencha, 6 maart 2012. [Online]. Available: <http://dev.sencha.com/deploy/touch/release-notes.html>. [Geopend 12 maart 2012].
- [23] T. Parker, „Announcing jQuery Mobile 1.1.0 RC1,” jQuery, 28 februari 2012. [Online]. Available: <http://jquerymobile.com/blog/2012/02/28/announcing-jquery-mobile-1-1-0-rc1/>. [Geopend 12 maart 2012].

6 Lijst van figuren

Figuur 3.1 Pagina's middels div tags	12
Figuur 3.2 PhoneGap op iOS	14
Figuur 3.3 PhoneGap op Android	14
Figuur 3.4 Appcelerator Titanium op iOS	17
Figuur 3.5 Appcelerator Titanium op Android	17
Figuur 3.6 MoSync vreemd teken bij strings in een loop code & resultaat	19
Figuur 3.7 MoSync strings in loop code & resultaat met voornaam lengte	19
Figuur 3.8 MoSync reset string waarde in loop code	20
Figuur 4.1 Updates gedurende het onderzoek	23

Afstudeerdossier

Scrum Rapporten



Dit document bevat de volgende onderdelen:

- Product Backlog
- Sprint Backlogs
- Sprint Progress Reports

Versie:	1.0
Auteur:	Alvin Nutbeij
Studentnummer:	50128
Examinatoren:	Gerard Tuk, Robin Collard
Bedrijfsmentor:	Jorrit van der Ven
Datum:	31 mei 2012

Product Backlog

Project:		MOFLAS								
Product definition										
#	#	Type	Theme	As a	I want to	So that	Constraint / Note	Priority		
1	0	Epic	Navigate	Pilot	view my route in 3D	I can see where I'm flying		Must		
1	1	Story	Navigate	Pilot	view my position on a map in a 3D enviroment	I can see my current location		Must		
1	2	Story	Navigate	Pilot	see my airplane move across the map	I can follow my flight route		Must		
1	3	Story	Navigate	Pilot	predetermine my route by inserting waypoints	I know what route to follow	Waypoints are added by inserting their name into a text box. When multiple waypoints with the same name are found an option box should appear	Must		
1	4	Story	Navigate	Pilot	see in what direction I should fly within the 3D enviroment	I know what heading to follow	The route should be visualized by an arrow which the pilot should follow	Must		
1	5	Story	Navigate	Pilot	view my landing approach in a 3D enviroment	I can see exactly how I should approach the airstrip	Route should be displayed by hoops / squars through which the airplane must fly	Should		
1	6	Story	Navigate	Pilot	manage waypoints	I can alter or remove routes		Must		
1	7	Story	Navigate	Pilot	select previous routes	I don't have to create them each time I go flying		Must		
1	8	Story	Fly	Pilot	view my flight information	I have a quick overview	Flight info such as speed, location, heading etc.			
2	0	Epic	Live weather	Pilot	receive live weather information	I know what I can expect		Must		
2	1	Story	Live weather	Pilot	see the weather conditions in a 3D enviroment	I can see the weather conditions around me		Must		
2	2	Story	Live weather	Pilot	see whether I should fly over, under or through the clouds	I won't make the wrongdecision		Should		
2	3	Story	Live weather	Pilot	see different type of clouds	I can distinguish thunder clouds from regular clouds	Thunder clouds should have a red color to indictate danger	Should		
3	0	Epic	Demonstration	Sales Representative	be able to demonstrate the application to possible clients	we can show them what we can do and win them over		Must		
3	1	Story	Demonstration	Sales Representative	be able to turn off actual height	the airplane won't be on the ground when giving a demonstration		Must		
3	2	Story	Demonstration	Sales Representative	be able to turn off actual speed	the airplane won't stay on the same spot when giving a demonstration	It would be nice to set a custom speed	Must		
3	3	Story	Demonstration	Sales Representative	be able to turn off actual heading	the airplane can make turns when giving a demonstration	A light, constant turn is enough. It would be nice to also be able to steer	Must		

Sprint 24 Backlog

Product definition			Remaining points					
P#	S#	Description	Initial	After 1 day	After 2 days	After 3 days	After 4 days	After 5 days
		Target	69	55	41	28	14	0
		Remaining	69	69	60	54	38	30
1	0	Project aanmaken	1	1	0	0	0	0
1	1	3D kaart bestaand uit losse tiles implementeren	3	3	0	0	0	0
1	1	Object pool maken voor tiles	5	5	5	5	5	5
1	1	Vliegtuig toevoegen	3	3	3	3	3	3
1	1	Vliegtuig boven huidige locatie positioneren	3	3	3	3	3	3
1	2	Bewegen naar noorden	4	4	4	0	0	0
1	2	Kaart draaien op basis van richting	3	3	3	3	3	3
1	2	Hoogte van het vliegtuig aanpassen aan werkelijke hoogte	5	5	5	5	5	5
0	0	Deelvragen van onderzoek beschrijven in afstudeerverslag	5	5	0	0	0	0
0	0	PhoneGap + jQuery mobile beschrijven in afstudeerverslag	3	3	3	3	3	3
0	0	Appcelerator beschrijven in afstudeerverslag	3	3	3	3	3	3
0	0	MoSync beschrijven in afstudeerverslag	2	2	2	2	2	2
0	0	Conclusie en aanbevelingen beschrijven in afstudeerverslag	3	3	3	3	3	3
1	1	Oplossen probleem met tile textures	8	8	8	8	8	0
1	1	Fog toevoegen	2	2	2	0	0	0
1	2	Bewegen naar zuiden	4	4	4	4	0	0
1	2	Bewegen naar oosten	4	4	4	4	0	0
1	2	Bewegen naar westen	4	4	4	4	0	0
0	0	Plan van Aanpak project herschrijven	4	4	4	4	0	0

Sprint 25 Backlog

Product definition			Remaining points					
P#	S#	Description	Initial	After 1 day	After 2 days	After 3 days	After 4 days	After 5 days
		Target	76	61	46	30	15	0
		Remaining	76	43	33	31	31	19
1	2	Object Pool ontwerpen	4	0	0	0	0	0
1	2	Object Pool implementeren	4	0	0	0	0	0
0	0	Views loshalen van controller	4	0	0	0	0	0
0	0	Models loshalen van controller	4	0	0	0	0	0
1	2	Verplaatsen over de map 1 functie	6	0	0	0	0	0
1	1	Testen op iPad 2 of 3	1	0	0	0	0	0
1	1	Hoogte vliegtuig bepalen	6	6	0	0	0	0
1	2	Vliegtuig roteren	2	2	2	0	0	0
1	2	Camera met vliegtuig mee roteren	4	4	0	0	0	0
1	2	Richting obv echte gegevens	3	3	3	3	3	0
1	1	Locatie obv echte gegevens	3	3	3	3	3	0
1	1	Hoogte obv echte gegevens	3	3	3	3	3	0
1	2	Snelheid vliegtuig bepalen	6	0	0	0	0	0
1	2	Snelheid obv echte gegevens	3	3	3	3	3	0
1	1	Camera roteren dmv touch	8	8	8	8	8	8
0	0	PhoneGap + jQuery mobile beschrijven in afstudeerverslag	3	3	3	3	3	3
0	0	Appcelerator beschrijven in afstudeerverslag	3	3	3	3	3	3
0	0	MoSync beschrijven in afstudeerverslag	2	2	2	2	2	2
0	0	Conclusie en aanbevelingen beschrijven in afstudeerverslag	3	3	3	3	3	3
1	1	Vliegtuig toevoegen	4	0	0	0	0	0

Sprint 26 Backlog

Product definition			Remaining points					
P#	S#	Description	Initial	After 1 day	After 2 days	After 3 days	After 4 days	After 5 days
		Target	77	62	46	31	15	0
		Remaining	77	61	61	35	27	27
0	0	Afstudeerverslag: Experiment jQuery Mobile & PhoneGap	3	3	3	3	3	3
0	0	Afstudeerverslag: Experiment Appcelerator	3	3	3	3	3	3
0	0	Afstudeerverslag: Experiment MoSync	2	2	2	2	2	2
0	0	Afstudeerverslag: Onderzoek conclusies & aanbevelingen	3	3	3	3	3	3
0	0	Afstudeerverslag: Selectie 3D Engine	3	3	3	3	3	3
0	0	Afstudeerverslag: Selectie Map Framework	3	3	3	3	3	3
0	0	Onderzoeksrapport: Update conclusie en aanbevelingen	2	2	2	2	2	2
1	1	Rotate camera through touch	8	8	8	8	8	8
1	3	Invoerscherm waypoints	4	4	4	0	0	0
1	3	Navigatie (menu) in App	4	4	4	0	0	0
1	3	Waypoints database	4	4	4	0	0	0
1	3	Invoeren waypoints	4	4	4	0	0	0
1	3	Save route	2	2	2	0	0	0
1	3	Select previous route	2	2	2	2	0	0
1	1	Solve tile problem	16	0	0	0	0	0
1	4	Create route marker	4	4	4	0	0	0
1	4	Add route marker to map	4	4	4	0	0	0
1	4	Base route marker on actual direction	6	6	6	6	0	0

Sprint 27 Backlog

Product definition			Remaining points					
P#	S#	Description	Initial	After 1 day	After 2 days	After 3 days	After 4 days	After 5 days
		Target	60	48	36	24	12	0
		Remaining	60	42	39	39	28	28
0	0	Afstudeerverslag: PhoneGap + jQueryMobile	3	0	0	0	0	0
0	0	Afstudeerverslag: Appcelerator	3	0	0	0	0	0
0	0	Afstudeerverslag: MoSync	2	0	0	0	0	0
0	0	Afstudeerverslag: Conclusies onderzoek	3	0	0	0	0	0
0	0	Afstudeerverslag: 3D engine selectie	4	0	0	0	0	0
0	0	Afstudeerverslag: Aanpassing weergegevens	3	3	3	3	0	0
0	0	Afstudeerverslag: Laten vallen weergegevens	2	2	2	2	2	2
0	0	Onderzoeksrapport: Rewrite conclusies	3	0	0	0	0	0
0	0	Afstudeerverslag: Selectie map framework	3	3	0	0	0	0
0	0	Afstudeerverslag: 3D deel ontwerp	3	3	3	3	3	3
0	0	Afstudeerverslag: Navigatie deel ontwerp	5	5	5	5	5	5
0	0	Afstudeerverslag Navigatie deel ontwikkeling, inclusief tile problemen	8	8	8	8	8	8
1	6	Finish CRUD functionality for routes	4	4	4	4	4	4
0	0	Memory issues	8	8	8	8	0	0
1	4	Detect when waypoints are reached	6	6	6	6	6	6

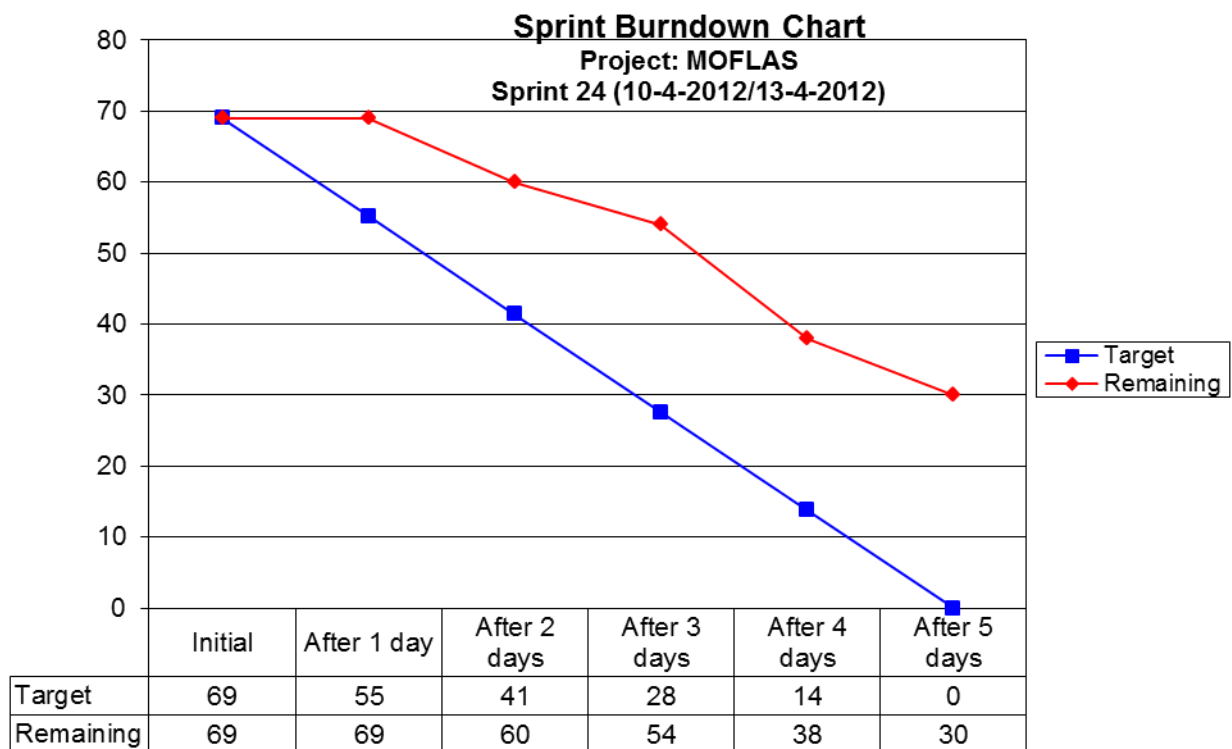
Progress Report

Project	MOFLAS
Sprint	24
Date	16-04-2012

Metrics

Note that the sprint consisted of only four days, during which a major impediment was encountered, therefore not all of the items planned items are finished.

	Planning	Actual
Hours	32	29
Points	69	39
Speed [pts/hr]	2,156	1,345



Major Impediments

Description	Priority	Required action	Owner
3D engine not fully compatible with the iPad1	High	Acquire a new iPad (version2 or 3)	
The iOS simulator doesn't handle the 3D graphics as well	High	Acquire a new iPad (version 2 or 3) for testing purposes	

as a real device does			
-----------------------	--	--	--

Functional Demonstration

Description	Importance	Required action	Owner

Release Overview

The overall release is behind schedule. The current version is not stable enough to demonstrate, however an earlier version can be used to demonstrate an airplane in a 3D environment.

Software Process

Description	Priority	Required action	Owner
None.			

Project Process

Description	Priority	Required action	Owner
An e-mail has been sent to Toon requesting a newer version of the iPad for testing purposes.	High		

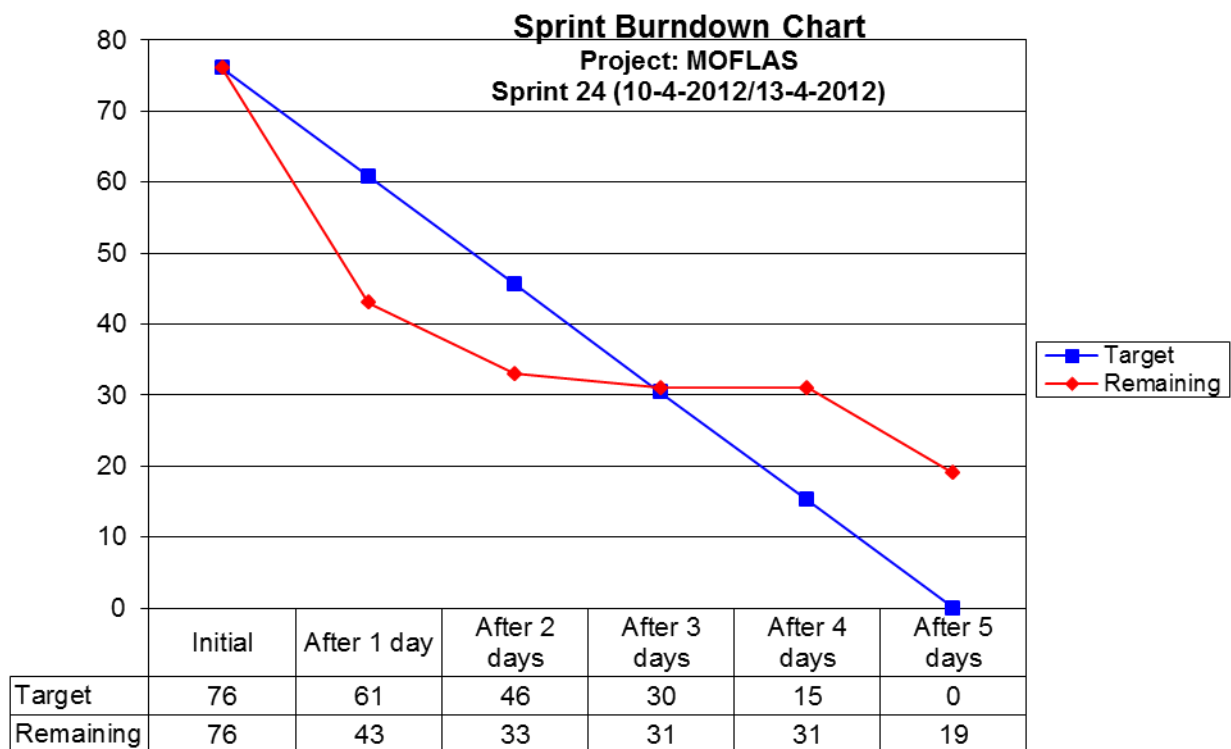
Progress Report

Project	MOFLAS
Sprint	25
Date	23-04-2012

Metrics

The SocialSOS project had a demonstration on Friday but didn't have enough material to demonstrate. I've helped them for at least 1,5 days which is why the actual hours are much lower than the planned hours

	Planning	Actual
Hours	40	28
Points	76	57
Speed [pts/hr]	1,9	2,0357



Major Impediments

Description	Priority	Required action	Owner
The iPad doesn't display the tiles as it should, which slowed this week's progress	High	Keep communicating with the engine's creator to find a solution + keep trying to find the solution	

--	--	--	--

Functional Demonstration

Description	Importance	Required action	Owner

Release Overview

The overall release is behind schedule. The current version is not stable enough to demonstrate, however an earlier version can be used to demonstrate an airplane in a 3D environment.

Software Process

Description	Priority	Required action	Owner
None.			

Project Process

Description	Priority	Required action	Owner
An iPad 2 has been acquired, which makes testing faster and ensures device availability			

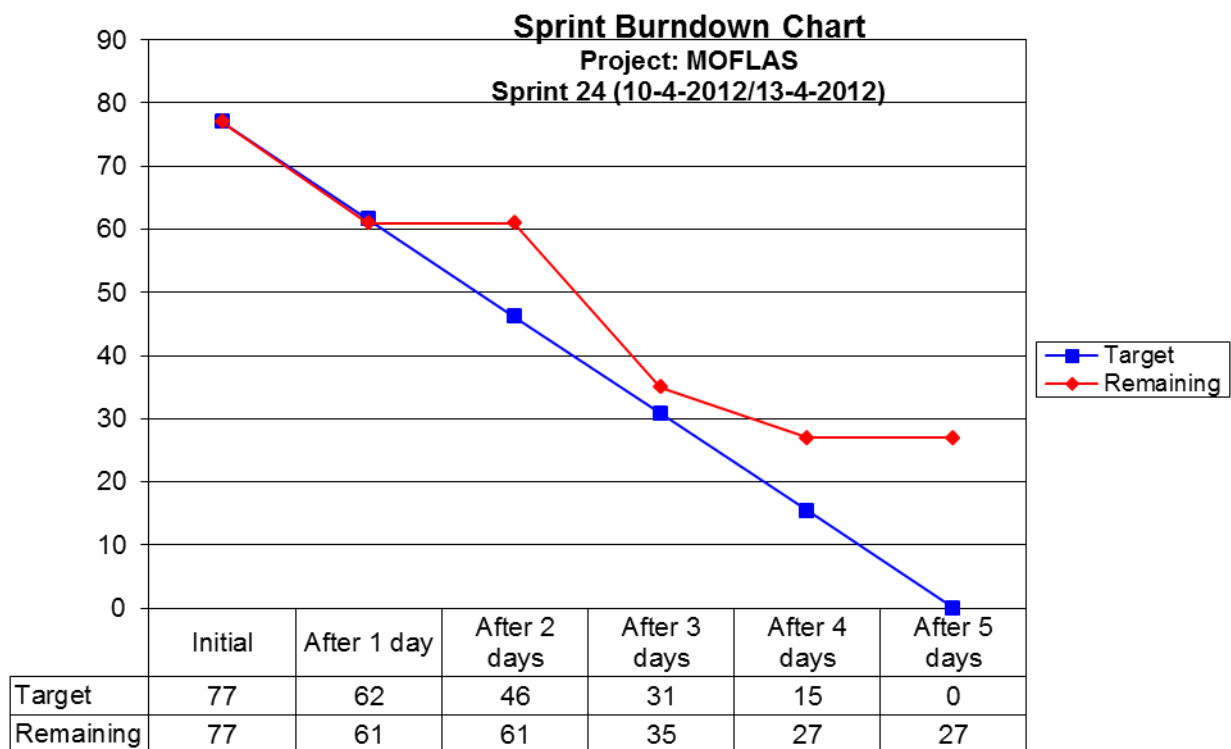
Progress Report

Project	MOFLAS
Sprint	26
Date	01-05-2012

Metrics

The SocialSOS project is nearing it's deadline and during the week completely turned around to focus on a different approach. Therefor I've spent about 2 full days helping them (spread during the week)

	Planning	Actual
Hours	40	24
Points	77	27
Speed [pts/hr]	1,925	2,083



Major Impediments

Description	Priority	Required action	Owner

Functional Demonstration

Description	Importance	Required action	Owner

Release Overview

The overall release is behind schedule. The current release is stable enough for internal demonstration, some memory errors (threading related) make this version unstable enough for external demonstration.

Software Process

Description	Priority	Required action	Owner
None.			

Project Process

Description	Priority	Required action	Owner

Progress Report

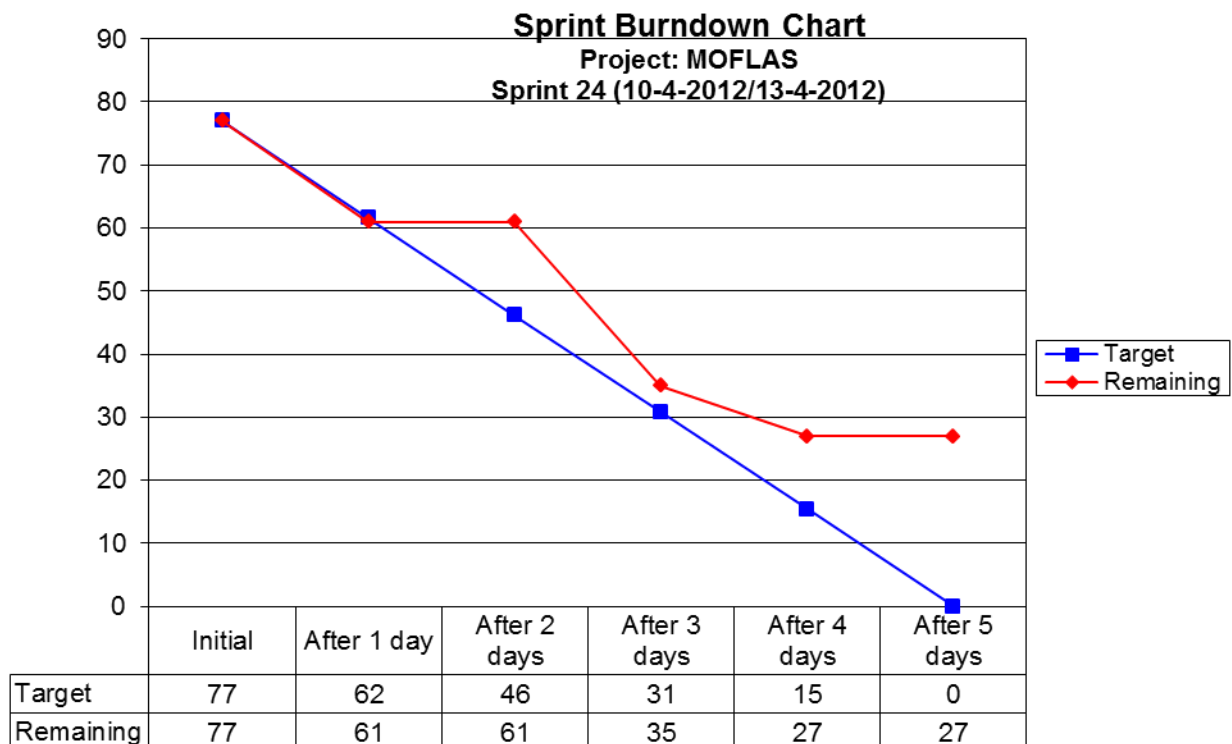
Project	MOFLAS
Sprint	27
Date	07-05-2012

Metrics

It's only one week until the SocialSOS deadline. Due to a change of programming language they needed less help than before, however the RestaurantApp has been continued by two new people who also required help.

During the sprint's planning some wrong choices were made concerning the tasks. Therefore a lot of time was spent on tasks not in the sprint planning.

	Planning	Actual
Hours	40	30
Points	60	28
Speed [pts/hr]	1,75	0,9375



Major Impediments

Description	Priority	Required action	Owner

Functional Demonstration

Description	Importance	Required action	Owner

Release Overview

The overall release is behind schedule. The current release is stable enough for internal demonstration, some memory errors (threading related) make this version unstable enough for external demonstration.

Software Process

Description	Priority	Required action	Owner
None.			

Project Process

Description	Priority	Required action	Owner

Afstudeerdossier

Overige documenten

NCIM  **GROEP**

DE HAAGSE
HOGESCHOOL

Dit document bevat de volgende onderdelen:

- Het goedgekeurde afstudeerplan
- Het Plan van Aanpak
- Het voortgangsverslag
- De tussentijdse beoordelingen

Versie:	1.0
Auteur:	Alvin Nutbeij
Studentnummer:	50128
Examinatoren:	Gerard Tuk, Robin Collard
Bedrijfsmentor:	Jorrit van der Ven
Datum:	31 mei 2012

Afstudeerplan

Informatie afstudeerder en gastbedrijf *(structuur niet wijzigen)*

Afstudeerblok: 2012-1.1 (start uiterlijk 6 februari 2012)

Startdatum uitvoering afstudeeropdracht: 6 februari 2012

Inleverdatum afstudeerdossier volgens jaarrooster: 1 juni 2012

Studentnummer: 50128

Achternaam: dhr Nutbeij

Voorletters: A

Roepnaam: Alvin

Adres: Wilgenlaan 27

Postcode: 2612TJ

Woonplaats: Delft

Telefoonnummer: 015 88 757 66

Mobiel nummer: 06 211 17 405

Privé emailadres: anutbey@gmail.com

Opleiding: Informatica

Locatie: Den Haag

Variant: voltijd

Naam studieloopbaanbegeleider: Merie Heijne

Naam begeleidend examiner: G.M. Tuk

Naam tweede examiner: R. Collard

Naam bedrijf: NCIM-Groep

Afdeling bedrijf: Operations

Bezoekadres bedrijf: Overgoo 11

Postcode bezoekadres: 2266JZ Leidschendam

Postbusnummer:

Postcode postbusnummer:

Plaats: Leidschendam

Telefoon bedrijf: 070-3501554

Telefax bedrijf: 070-3545515

Internetsite bedrijf: <http://www.ncim-groep.nl>

Achternaam opdrachtgever: Kasperink

Voorletters opdrachtgever: H.

Titulatuur opdrachtgever: dhr.

Functie opdrachtgever: Directeur Operationele Zaken

Doorkiesnummer opdrachtgever: -

Email opdrachtgever: h.kasperink@ncim.nl

Achternaam bedrijfsmentor: Van der Ven

Voorletters bedrijfsmentor: J.

Titulatuur bedrijfsmentor: Dhr.

Functie bedrijfsmentor: Software Engineer

Doorkiesnummer bedrijfsmentor: -

Email bedrijfsmentor: j.van.der.ven@ncim.nl

NB: bedrijfsmentor mag dezelfde zijn als de opdrachtgever

Doorkiesnummer afstudeerder:

Functie afstudeerder (deeltijd/duaal):

Titel afstudeeropdracht:

Onderzoek naar de mogelijkheden om platform onafhankelijke mobiele applicaties te maken en het uitbreiden van de Mobile Flight Assistant Applicatie op basis van de resultaten uit het onderzoek.

Opdrachtomschrijving**1. Bedrijf**

De NCIM-Groep is gespecialiseerd in het detacheren van goed opgeleide professionals. Ook worden intern een aantal projecten uitgevoerd. Het bedrijf richt zich op opdrachtgevers in de sectoren Defensie en Veiligheid, Energie en utilities, Telecommunicatie, transport en media, en Technische automatisering. Momenteel zijn meer dan 200 mensen in dienst bij de NCIM-Groep.

Het bedrijf is ISO 9001:2008 gecertificeerd, Oracle Partner en Microsoft Gold Partner. Iedere NCIM-professional is opgeleid aan een Nederlandse Universiteit of Hogeschool en is daarnaast gecertificeerd in zijn of haar specialisatie. Het bedrijf bestaat sinds 1988 en heeft vestigingen in Eindhoven, Brussel, Camberly (Verenigd Koninkrijk) en Fort Worth (Verenigde Staten). Het hoofdkantoor is gevestigd in Leidschendam.

Het bedrijf is recentelijk partnerschappen aangegaan met Sitecore Nederland en NFI. Daarnaast is het bedrijf uitgeroepen tot een van de Top ICT Employers van 2011 door het onafhankelijke CRF Institute.

In verband met de groeiende mobiele applicatie markt wil de NCIM-Groep haar klanten ook mobiele applicaties gaan aanbieden.

2. Probleemstelling

De NCIM-Groep wil zich gaan richten op de mobiele applicatie markt. Om de mogelijkheden binnen deze markt te onderzoeken zijn ze bezig met de ontwikkeling van de iPad applicatie "Mobile Flight Assistant". Dit is een applicatie die sportpiloten helpt tijdens de vlucht.

Deze applicatie bevat momenteel slechts beperkte functionaliteiten en is geschreven in native Objective-c code. Deze applicatie is al eerder als demo op een beurs getoond en heeft veel positieve response gehad. De NCIM-Groep wil deze applicatie dus verder ontwikkelen.

Om aan de wensen van klanten te kunnen voldoen moeten applicaties echter op verschillende platforms werken. Om de applicaties voor elk platform te maken kost veel tijd en daarmee geld. De NCIM-Groep is daarom op zoek naar een mogelijkheid om applicaties zo veel mogelijk platform onafhankelijk te maken.

De Mobile Flight Assistant is momenteel nog niet uitgebreid genoeg om daadwerkelijk klanten binnen te kunnen halen. Daarom moet de applicatie verder uitgebreid worden.

3. Doelstelling van de afstudeeropdracht

Om te zorgen dat de applicatie voor meerdere platforms geschikt is wil de NCIM-Groep weten of het mogelijk is om de bestaande applicatie om te zetten naar een platform onafhankelijke applicatie en toekomstige applicaties en functionaliteiten ook platform onafhankelijk te maken. De NCIM-Groep is op zoek naar een framework om het bouwen van platform onafhankelijke applicaties gemakkelijker te maken. Dit framework moet gecreëerd of, mocht het al bestaan, gevonden worden. De NCIM-Groep is in eerste instantie geïnteresseerd in het gebruik van HTML5 voor platform onafhankelijke applicaties. Zij willen dus in elk geval weten wat precies wel en niet mogelijk is met HTML5. Mocht HTML5 niet gebruikt kunnen worden wordt gezocht naar andere oplossingen.

Om de bevindingen te ondersteunen wordt een functioneel Proof of Concept gemaakt. Dit Proof of Concept bestaat uit een kleine demonstratie applicatie. Deze heeft geen doel anders dan het aantonen dat de gevonden methode voor het platform onafhankelijk ontwikkelen werkt.

Momenteel is de Mobile Flight Assistant nog zeer beperkt. Het uiteindelijke doel van de applicatie is het demonstreren van de mogelijkheden van een mobiele applicatie en daarmee de NCIM-Groep te promoten. Om aan dit doel te kunnen voldoen is een uitgebreidere applicatie nodig. Momenteel zijn er twee belangrijke uitbreidingen:

- Weergegevens
- Flight planning

Weergegevens

De bedoeling is dat de app realtime weergegevens laat zien. Deze worden uitgezonden door een ADSB ground station. Hier moet het signaal dus opgevangen, gedecodeerd en weergegeven worden.

Flight Planning

De gebruiker moet waypoints kunnen invoeren, welke worden weergegeven (3D) inclusief verscheidene controles, zoals niet door no-flight zones heen vliegen.

De uitbreidingen worden gemaakt op basis van de uitkomsten van het onderzoek ongeacht wat deze uitkomsten zijn.

4. Resultaat

De NCIM-Groep beschikt over nieuwe kennis over het maken van platform onafhankelijke mobiele applicaties. Deze kennis wordt in de toekomst gebruikt om nieuwe applicaties te ontwikkelen en bestaande applicaties uit te breiden of om te zetten. Daarnaast is de Mobile Flight Assistant applicatie uitgebreid met de nieuwe functionaliteiten:

- Weergegevens
- Flightplanning

De applicatie kan vervolgens vaker op beurzen in worden gezet om de mobiele applicatie ontwikkeling van de NCIM-Groep te promoten. De applicatie moet dus zorgen dat de NCIM-Groep nieuwe klanten aantrekt. Naast het promoten zal de applicatie ook daadwerkelijk door sport piloten gebruikt worden.

5. Uit te voeren werkzaamheden, inclusief een globale fasering, mijlpalen en bijbehorende activiteiten

Het eerste onderdeel van de opdracht is het onderzoeken naar de mogelijkheden om platform onafhankelijke mobiele applicaties te maken. De globale planning voor het onderzoek ziet er als volgt uit:

- Onderzoeksplan maken & Vooronderzoek (5 dagen)
- Onderzoek uitvoeren (10 dagen)
- Bouwen Proof of Concept uitvoeren (10 dagen)
- Onderzoeksrapport opstellen (5 dagen)

Bij het tweede deel van de opdracht wordt de Mobile Flight Assistant applicatie uitgebreid op basis van de bevindingen binnen het onderzoek. De NCIM-Groep maakt gebruik van de SCRUM ontwikkelmethode, deze methode wordt ook voor deze opdracht gehanteerd.

Per iteratie worden dezelfde werkzaamheden dus steeds opnieuw uitgevoerd, maar dan voor een ander systeemdeel. Op dit moment is het voor mij nog niet duidelijk hoe de iteraties precies verdeeld worden. Mogelijk wordt dit per uitbreiding gedaan, maar mogelijk worden de uitbreidingen ook weer opgedeeld in meerdere iteraties.

De volgende werkzaamheden dienen bij iedere iteratie uitgevoerd te worden (voor dit afstudeerplan ben ik er van uit gegaan dat per uitbreiding een iteratie plaatsvindt):

- Plan van Aanpak & vaststellen en analyseren van de requirements (3 dagen)
- Systeemdeel ontwerpen (3 dagen)
- Systeemdeel bouwen (10 dagen)
- Systeemdeel testen en implementeren (4 dagen)

Het tweede deel wordt ongeacht de uitkomsten van het onderzoek uitgevoerd. In het geval dat het onderzoek of het bouwen van het Proof of Concept vertraging op loopt wordt een beperkter Proof of Concept opgeleverd zodat de uitbreidingen volledig werkend kunnen worden opgeleverd. Daarnaast worden 15 dagen gereserveerd voor het opbouwen van mijn afstudeerdossier.

6. Op te leveren (tussen)producten

De volgende producten worden opgeleverd:

- Onderzoeksrapport met daarin de bevindingen & aanbevelingen
- Code Proof of Concept
- Documentatie uitbreidingen
 - UML Diagrammen
 - SCRUM Documenten
- Code uitbreidingen

7. Te demonstreren competenties en wijze waarop

De volgende beroepstaken worden behaald:

- **1.4 Uitvoeren analyse door definitie van requirements**

Middels interviews wordt bepaald wat de requirements zijn wat betreft de uitbreidingen. Hieruit volgt een lijst met geprioriteerde functionele en niet-functionele requirements.

- **3.2 Ontwerpen systeemdeel**

De uitbreidingen op de Mobile Flight Assistant applicatie worden ontworpen. Bij het ontwerpen worden design patterns gebruikt om de uitbreidbaarheid en de herbruikbaarheid van het ontwerp. Hieruit volgen een aantal producten:

- UML diagrammen
- User Interface beschrijving
- User Interaction beschrijvingen

- **3.3 Bouwen systeemdeel**

De uitbreidingen van de Mobile Flight Assistant applicatie worden gebouwd. Bij het bouwen wordt gebruik gemaakt van design patterns om de uitbreidbaarheid en de herbruikbaarheid van de code te vergroten. Deze beroepstaak wordt zowel bij het Proof of Concept als bij het bouwen van de uitbreidingen aangetoond.

Plan van Aanpak

Onderzoek naar de mogelijkheden om platformonafhankelijke mobiele applicaties te maken en het uitbreiden van de Mobile Flight Assistant Applicatie op basis van de resultaten uit het onderzoek

Versie 1.1

Alvin Nutbeij

12-4-2012

NCIM  GROEP

DE HAAGSE
HOGESCHOOL

Inhoudsopgave

1	Inleiding	2
2	Opdrachtoomschrijving	3
2.1	Kader van de opdracht	3
2.2	Probleemstelling.....	3
2.3	Doelstelling.....	3
3	Afbakening.....	4
4	Risicoanalyse	5
5	Benodigde middelen	7
6	Op te leveren producten	8
7	Planning.....	9
	Bijlage I	10

1 Inleiding

Deze opdracht is een initiatief van de NCIM Groep. De opdracht wordt uitgevoerd door Alvin Nutbeij als afstudeeropdracht voor de opleiding Informatica aan de Haagse Hogeschool.

In dit document wordt de aanpak van de opdracht omschreven. De volgende onderdelen zijn opgenomen binnen dit document:

- **Opdrachtoomschrijving**
Binnen de opdrachtoomschrijving worden het kader van de opdracht, de probleemstelling en de doelstelling beschreven.
- **Afbakening**
Hier worden de afbakeningen omtrent de opdracht omschreven.
- **Risicoanalyse**
In de risicoanalyse worden een aantal risico's benoemd. Deze risico's worden gecategoriseerd en vervolgens wordt per risico beschreven wat de oorzaak is, wat de gevolgen zijn, hoe de risico's opgelost kunnen worden en hoe deze te voorkomen zijn.
- **Benodigde middelen**
Hier worden de middelen die nodig zijn voor de uitvoer van de opdracht beschreven.
- **Op te leveren producten**
Hier worden de producten die na de uitvoer van de opdracht worden opgeleverd beschreven.
- **Planning**
Hier worden de globale planning en de mijlpalen besproken.

2 Opdrachtomschrijving

In dit hoofdstuk wordt de opdracht omschreven. Allereerst wordt het kader van de opdracht geschetst, vervolgens worden de probleemstelling en de doelstelling besproken.

2.1 Kader van de opdracht

De opdrachtgever en algemeen directeur van de NCIM Groep, Toon Akkermans, bezit een sportvliegtuig waar hij regelmatig in vliegt. De handboeken, met daarin frequenties en dergelijke, zijn zeer groot. Daarom kwam hij op het idee om deze te vervangen door een iPad applicatie.

Nadat een demo applicatie, genaamd Mobile Flight Assistant (MOFLAS), gemaakt was heeft de NCIM Groep hiermee op een beurs gestaan waar deze applicatie enthousiast werd ontvangen. Zodoende wil de NCIM Groep zich meer gaan richten op het ontwikkelen van mobiele applicaties en willen ze meer demonstratie materiaal hebben om te laten zien wat ze kunnen.

2.2 Probleemstelling

De NCIM Groep wil zich, naast de markten waarin het actief is, gaan richten op de mobiele markt. Om klanten te kunnen overtuigen voor de NCIM Groep te kiezen is echter meer nodig dan de demo applicatie die tot nu toe gemaakt is. De applicatie moet dus uitgebreid worden zodat hiermee wederom op beurzen gestaan kan worden om klanten te laten zien wat de NCIM Groep kan.

Tegelijkertijd wil de NCIM Groep zich gaan richten om meer platformen dan alleen iOS, het operating system voor de mobiele apparaten van Apple. De NCIM Groep wil weten wat de mogelijkheden zijn om applicaties platformonafhankelijk te maken met slechts één codebase.

2.3 Doelstelling

Om applicaties platformonafhankelijk te kunnen ontwikkelen moet een geschikt framework gezocht worden. Ook moeten de minimale eisen aan dit framework nog worden vastgesteld. In eerste instantie is de NCIM Groep geïnteresseerd in het gebruik van HTML5 als vervanging van native code, mocht dit geen uitkomst bieden kunnen ook andere methoden onderzocht worden.

Om de bevindingen van het onderzoek te ondersteunen dient een Proof of Concept applicatie gemaakt te worden om de mogelijkheden te laten zien. Dit Proof of Concept bestaat slechts uit een kleine demo applicatie die geen werkelijke functie hoeft te hebben.

De NCIM Groep wil naast de kennis over platformonafhankelijke ontwikkeling ook de MOFLAS applicatie uitgebreid zien worden met als doel deze in te kunnen zetten op beurzen en zodoende klanten te overtuigen voor de NCIM Groep te kiezen. Daarnaast wil de opdrachtgever de applicatie ook gebruiken tijdens het vliegen.

De functionaliteiten die geïmplementeerd moeten worden zijn:

- Flight planning
De gebruiker moet middels het invoeren van waypoints een route kunnen vaststellen. Vervolgens moet de opgegeven route in 3D worden weergegeven.
- Live weergegevens
De bedoeling is dat de applicatie realtime weergegevens laat zien. Deze worden uitgezonden door een ADS-B grondstation. Het signaal moet dus opgevangen, gedecodeerd en weergegeven worden.

3 Afbakening

Om het ontwikkelen van de uitbreidingen niet in gevaar te brengen is slechts vijf weken tijd voor het onderzoek. Om dit te realiseren wordt binnen het onderzoeksplan verder afgebakend.

4 Risicoanalyse

Op basis van de opdrachtomschrijving in hoofdstuk 2 zijn een aantal risico's benoemd. Deze zijn terug te vinden in Tabel 4.1. Elk risico heeft een niveau. Deze niveaus zeggen niets over de kans van de risico's, maar zeggen iets over hoe ernstig de gevolgen van de risico's zijn. De volgende drie niveaus zijn benoemd:

- **Laag**
Risico's van laag niveau brengen weinig tot geen gevolgen met zich mee. Mogelijk is het niet nodig om de genoemde actie te ondernemen.
- **Middel**
Risico's van middel niveau kunnen enige gevolgen met zich meebrengen. Deze gevolgen kunnen doorwerken op de planning maar zullen de algemene haalbaarheid van het project niet direct beïnvloeden.
- **Hoog**
Risico's van hoog niveau brengen ernstige gevolgen met zich mee. Deze gevolgen werken direct door in de planning en beïnvloeden direct de haalbaarheid van het project.

Risico	Beschrijving
Onvoldoende documentatie <i>Niveau: Laag</i>	Oorzaak De personen die eerder aan deze opdracht gewerkt hebben, hebben mogelijk niet genoeg gedocumenteerd. Gevolg Mogelijk is de werking van de huidige applicatie en die van de server niet geheel duidelijk. Daardoor kan het implementeren van functionaliteiten vertraging oplopen. Actie In contact komen met de originele makers van de applicatie en de server om onduidelijkheden op te helderen. Voorkoming Vooraf de documentatie controleren en eventueel vooraf om opheldering vragen bij iemand die al eerder aan het project gewerkt heeft.
Beperkingen mobiele platform <i>Niveau: Middel</i>	Oorzaak Programmeren voor mobiele platformen werkt via een SDK waarin grenzen zijn gesteld aan de mogelijkheden die de programmeur krijgt. Gevolg Sommige functionaliteiten kunnen niet worden geïmplementeerd. Actie Een alternatief zoeken en dit voorstellen aan de opdrachtgever. In overleg kan dan een alternatief geïmplementeerd worden. Voorkoming Voordat een functionaliteit gemaakt wordt eerst de haalbaarheid bekijken. Mocht een functionaliteit niet mogelijk zijn kan eerder actie ondernomen worden.

Onvoldoende kennis 3D programmeren <i>Niveau: Hoog</i>	Oorzaak 3D Programmeren voor iOS blijkt moeilijker dan vooraf gedacht. Gevolg De functionaliteit moet aangepast worden of loopt, mogelijk ernstige, vertraging op. Actie Met de opdrachtgever in gesprek over de haalbaarheid van dit onderdeel en eventueel de andere uitbreiding laten vallen of inperken. Voorkoming Tutorials volgen omtrent 3D programmeren en een geschikte 3D engine selecteren.
Weergegevens kunnen niet worden ontvangen <i>Niveau: Hoog</i>	Oorzaak Dit risico kan meerdere oorzaken hebben, zoals het niet aanwezig zijn van de benodigde apparatuur of het niet beschikbaar zijn van het signaal. Gevolg Weergegevens kunnen niet real-time worden weergegeven, deze uitbreiding kan dus niet volgens de originele omschrijving worden geïmplementeerd. Actie Een alternatief bedenken en mogelijk het live signaal simuleren totdat de oorzaak opgelost kan worden. Voorkoming Met de opdrachtgever in gesprek gaan over de haalbaarheid van deze uitbreiding.

Tabel 4.1 Project risico's

5 Benodigde middelen

Voor het uitvoeren van de opdracht zijn de volgende middelen nodig:

- Een Mac of een Macbook
iOS applicaties moeten op een Mac ontwikkeld worden.
- Xcode
De IDE waarin iOS applicaties ontwikkeld worden
- Office tools
Ten behoeven het schrijven van de documentatie
- Een iPad
Om gemaakte applicatie(delen) te testen

6 Op te leveren producten

Aan het einde van de opdracht worden de volgende producten opgeleverd aan de NCIM Groep:

- De resultaten van het vooronderzoek
- Het onderzoeksplan
- Het onderzoeksrapport
- Het onafhankelijke Proof of Concept en de daarbij behorende code
- De documentatie behorende bij de uitbreidingen, waaronder:
 - UML Diagrammen
 - Bijgewerkte versie van het product backlog
 - Sprint backlogs
 - Item designs
 - Bijgewerkte versie van het high level design
 - Bijgewerkte versie van het project start document
- De uitgebreide applicatie en de daarbij behorende code

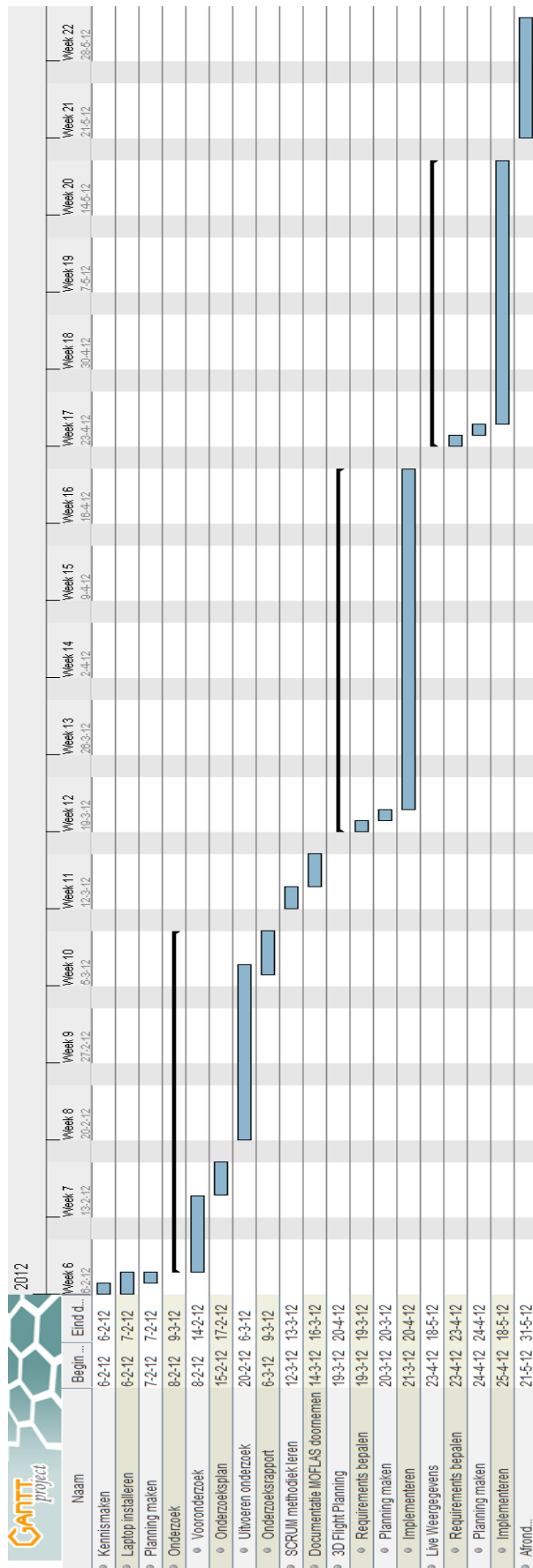
7 Planning

Omdat de opdracht uit twee delen bestaat wordt in dit Plan van Aanpak een zeer globale planning opgenomen. In het onderzoeksplan wordt een planning voor het onderzoek opgenomen. De planning van de uitbreidingen wordt van week tot week bekeken middels SCRUM sprints. De planning is te vinden in Bijlage I. De planning bevat een aantal mijlpalen, zoals te zien in Tabel 7.1.

Oplevering	Mijlpaal	Omschrijving
Week 5	Onderzoeksrapport	Het onderzoeksrapport met daarin mijn bevindingen en aanbevelingen omtrent de ontwikkeling van platformonafhankelijke mobiele applicaties
Week 11	3D Flightplanning	De 3D Flight planning uitbreiding is geïmplementeerd in de MOFLAS applicatie
Week 15	Live Weergegevens	De live weergegevens uitbreiding is geïmplementeerd in de MOFLAS applicatie
Week 17	Afstudeerdossier	Het afstudeerverslag wordt samen met alle overige documentatie ingeleverd

Tabel 7.1 Mijlpalen

Bijlage I



1 Inleiding

Middels dit voortgangsverslag wil ik u up-to-date houden omtrent de voortgang van mijn afstudeeropdracht. Gezien het bedrijfsbezoek wat later plaats vond dan in eerste instantie gepland was, is er nog niet veel veranderd in de tussentijd. Daarom geeft dit verslag voornamelijk een overzicht van de algehele voortgang, dus ook die besproken is tijdens het bedrijfsbezoek.

2 Onderzoek

De eerste weken van de opdracht stonden in het teken van het onderzoek naar platformonafhankelijke mobiele applicatie ontwikkeling. Alvorens aan dit onderzoek te beginnen heb ik een stukje vooronderzoek gedaan. Daarin heb ik onder andere tools op kunnen delen in vier categorieën.

Tijdens het daadwerkelijke onderzoek heb ik met behulp van een tool uit elk van deze categorieën een zeer simpele applicatie gemaakt om te kunnen bepalen in hoeverre en hoe gemakkelijk het gewenste resultaat gehaald kan worden.

Helaas heb ik aan de hand van dit onderzoek moeten concluderen dat er nog geen ‘perfecte’ oplossing is om applicaties platformonafhankelijk te ontwikkelen. Echter heb ik op basis van de geprobeerde tools wel kunnen bepalen in welke situatie de ene tool meer geschikt is dan de ander. Inmiddels is ook een nieuw app project gestart waarbij in eerste instantie het doel was op alle platformen te richten. Hierin werd mijn advies gevraagd en heb ik een van de tools aangeraden. Uiteindelijk bleek de opdrachtgever echter meer geïnteresseerd in een iPhone app waardoor nu alsnog platformafhankelijk ontwikkeld wordt.

Voor mijn eigen opdracht heb ik echter moeten concluderen dat het niet mogelijk is om deze om te zetten naar een platformonafhankelijke applicatie. De applicatie vraagt dusdanig veel van het apparaat dat het des te belangrijker is om in native code te programmeren. Op deze manier kan zo efficiënt mogelijk met de resources om worden gegaan.

3 3D Flight Planning

Het maken van een 3D applicatie (onderdeel) blijkt een stuk lastiger dan van te voren verwacht. Direct in OpenGL programmeren lijkt geen optie te zijn, gezien een simpele kubus tekenen al extreem veel (ingewikkelde) code vereist. Daarom heb ik gekozen om een 3D engine te gebruiken. Voor iOS zijn echter een stuk minder geschikte engines beschikbaar dan ik vooraf had gedacht.

De betere 3D engines zijn allemaal gericht op het maken van games. Deze werken eigenlijk allemaal met een externe tool waarin de game ontworpen kan worden, alvorens deze te exporteren naar een iOS applicatie. Dit maakt het in de meeste gevallen wel makkelijker om

voor meerdere platformen te ontwikkelen, echter is het puur geschikt voor games. Gezien dit onderdeel verwerkt moet worden in een 'normale' iOS applicatie kan het niet in zo'n engine ontwikkeld worden.

Over de engines die wel geïmplementeerd kunnen worden is helaas weinig te vinden. Daarom ben ik na een aantal probeersels overgestapt op Cocos3D, dit is een uitbreiding op het populaire Cocos2D framework waar een redelijke community achter zit. Echter is het bleek het Cocos3D framework een één mans project te zijn, welke momenteel pas p versie 0.6.5 zit. Ondanks dat is het toch een veelbelovend framework.

Cocos3D leek dan ook het beste framework tot ik NinevehGL tegen kwam. Ook dit framework bevindt zich nog in een bèta fase, momenteel versie 0.9.3. Het is echter een stuk volwassener. Het werkt prettig en vrij gemakkelijk, echter is er wederom zeer weinig over te vinden. De makers van het framework zelf hebben een aantal voorbeeldprojecten en tutorial filmpjes gemaakt, echter deze behandelen slechts de absolute basis.

4 Planning

Doordat het maken van een 3D onderdeel zoveel lastiger is dan verwacht heeft dit onderdeel wel wat vertraging opgelopen. Deze week heb ik echter een afspraak met Toon Akkermans, de opdrachtgever, om de requirements voor de flight planning uitbreiding te bespreken. Daarnaast blikken we tijdens dat gesprek ook al vooruit op de weergegevens, gezien momenteel nog geen ontvanger aanwezig lijkt te zijn om deze te kunnen ontvangen.

Ik ben verhuisd naar een andere projectruimte omdat ik alleen kwam te zitten (i.v.m. detachering van de overige collega's). Ook op deze kamer wordt momenteel aan een iOS app gewerkt (zie hoofdstuk 2). De SCRUM borden die hier aan de muur hangen zijn echter al in gebruik door andere projecten waardoor voor mij geen SCRUM bord beschikbaar is. In hoeverre volgens SCRUM werken logisch is voor één persoon moet ik ook nog bekijken, maar ik ben wel bezig met het leren van de benodigdheden en heb via collega's al wat templates voor documenten (zoals product & sprint backlogs) bemachtigd. Om toch gebruik te kunnen maken van een SCRUM bord heb ik een account aangemaakt op trello.com, waar een SCRUM bord kan worden nagemaakt (zie Figuur 4.1).



Figuur 4.1 Trello SCRUM bord

Bespreking concept	Tussentijds assessment	Eerste beoordeling
---------------------------	-------------------------------	---------------------------

Formulier bespreking concept afstudeerdossier

Student: Alvin Nutbeij

Studentnummer: 50128

Datum: 17 april 2012

Tijdens de bespreking is het volgende geconstateerd:		ja	nee
a	<i>Het voortgangsverslag is ontvangen</i>	X	
b	<i>Het afstudeerdossier is digitaal beschikbaar</i>	X	
c	<i>Het afstudeerdossier is opgebouwd conform de richtlijnen</i>	X	
d	<i>Het goedgekeurde afstudeerplan is aanwezig</i>	X	
e	<i>Het plan van aanpak is aanwezig</i>	X	
f	<i>Reeds geleverd commentaar is aanwezig</i>		X
g	<i>Het afstudeerdossier geeft voldoende inzicht in de stand van zaken</i>	X	
h	<i>De afstudeeropdracht is tot nu toe naar behoren uitgevoerd</i>	X	

Verbeterpunten:

Opmerkingen:

Het (voor)onderzoek ziet er goed uit, maar de conclusies ontbreken nog. Wat er nu ligt geeft goede hoop op een goede voortzetting en afronding, maar er is nog niet veel te zeggen over de uitbreiding van de MOFLAS-applicatie.

Het ziet ernaar uit dat de tegenslag in het ontwikkelwerk goed kan worden opgevangen door het beperken van de scope van het werk.

Naam begeleidend examiner: G.M. Tuk

Datum: 17 april 2012

Dit formulier wordt door de begeleidend examiner digitaal ingevuld en per email naar de student verstuurd met een cc naar de coördinator van ICT & Media @ Work (A.M.Schipper@hhs.nl). Het formulier dient door de student te worden opgenomen in het afstudeerdossier.

Bespreking concept	Tussentijds assessment	Eerste beoordeling
--------------------	------------------------	--------------------

Formulier tussentijds assessment

Student: Alvin Nutbeij

Studentnummer: 00050128

Datum: 7 mei 2012

eerste / tweede TTA: eerste

Tijdens het tussentijds assessment is het volgende geconstateerd:		ja	nee
a	Het voortgangsverslag is ontvangen	X	
b	Het afstudeerdossier is digitaal beschikbaar	X	
c	Het afstudeerdossier is opgebouwd conform de richtlijnen	X	
d	Het goedgekeurde afstudeerplan is aanwezig	X	
e	Het plan van aanpak is aanwezig	X	
f	Reeds geleverd commentaar is aanwezig	X	
g	Het afstudeerdossier geeft voldoende inzicht in de stand van zaken		X
h	De afstudeeropdracht is tot nu toe naar behoren uitgevoerd		X

Aanpak	O	T	V	G
Passend			X	
Theoretisch verantwoord	X			
Samenhang uitvoering beroepstaken	X			

Beroepstaken op afgesproken niveau uitgevoerd?		O	T	V	G
1	1.4 Uitvoeren analyse door definitie van requirements		X		
2	3.2 Ontwerpen systeemdeel		X		
3	3.3 Bouwen systeemdeel		X		
6					
7					
8					

Producten	O	T	V	G
<i>Tussenproducten</i>			X	
<i>Eindproducten</i>		X		

Effectief communiceren	O	T	V	G
<i>Binnen afstudeerbedrijf</i>		X		
<i>Afstudeerdossier</i>			X	

Reflectie	O	T	V	G
<i>Inzicht in eigen functioneren</i>	X			
<i>Inzicht in eigen leerproces</i>	X			

Toelichting per beoordelingscriterium

Aanpak
Uit verhaal duidelijk, niet correct opgeschreven.

Beroepstaken op afgesproken niveau uitgevoerd?
Allen twijfelachtig door onvolledige onderbouwing in doc.

Producten
Tussenproducten zijn opgeleverd (nog niet af). Eindproduct is nog niet opgeleverd.

Effectief communiceren
Binnen organisatie niet voldoende beschreven, afstudeerdossier goed gecommuniceerd.

Reflectie

Nog niet beschreven

Advies

<input checked="" type="checkbox"/>	Positief (bindend advies)
<input type="checkbox"/>	Verlengen (vrijblijvend advies)
<input type="checkbox"/>	Negatief (vrijblijvend advies)

Besluit student

Aankruisen welke beslissing de student heeft genomen (alleen na vrijblijvend advies)

<input type="checkbox"/>	Afstudeerdossier wordt op afgesproken datum ingeleverd Inleverdatum: 1 juni 2012
<input type="checkbox"/>	Afstudeerperiode wordt verlengd Inleverdatum:
<input type="checkbox"/>	Student stopt met afstudeeropdracht

Naam begeleidend examiner: G.M. Tuk
Naam tweede examiner: R. Collard
Datum: 7 mei 2012

Dit formulier wordt door de tweede examiner digitaal ingevuld, waarna de begeleidend examiner het per email verstuurt naar de student met een cc naar de coördinator van ICT & Media @ Work (A.M.Schipper@hhs.nl). Het formulier dient door de student te worden opgenomen in het afstudeerdossier.