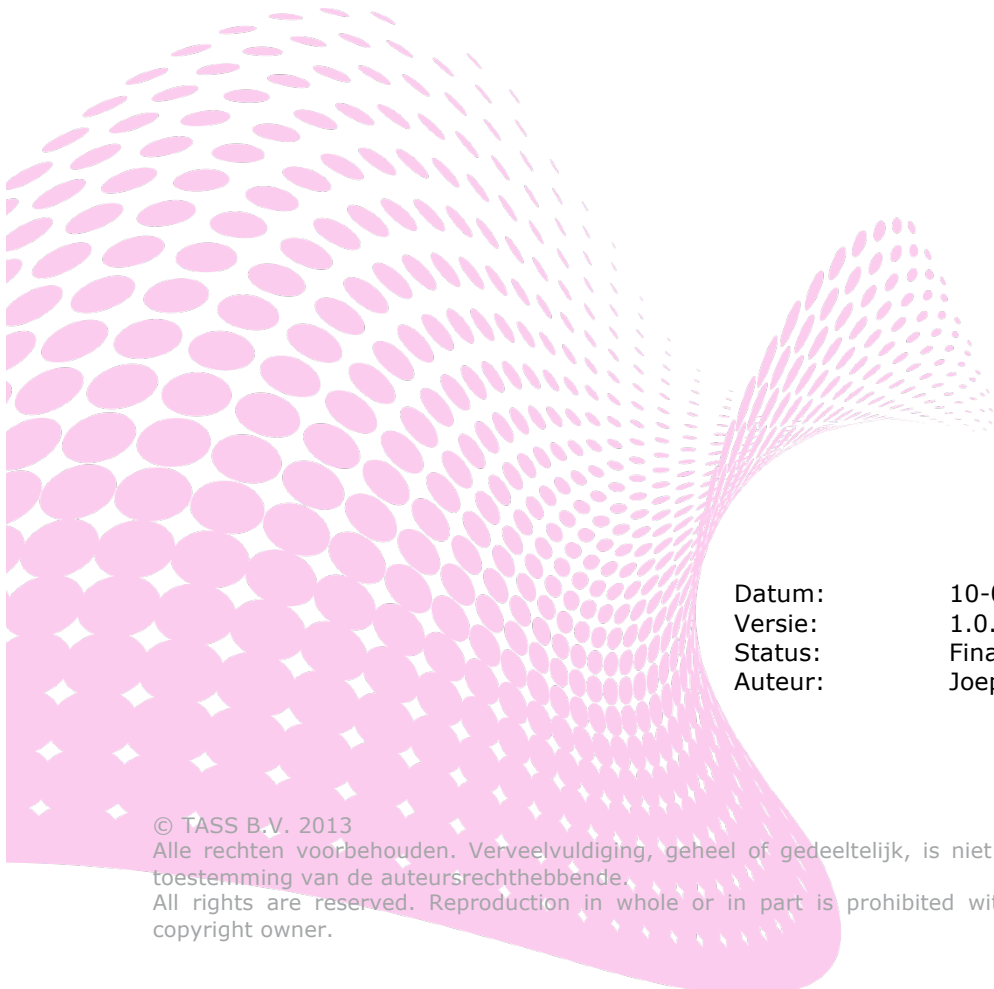


Afstudeerscriptie

Smartphones: develop once, run anywhere



Datum:	10-01-2013
Versie:	1.0.0
Status:	Final
Auteur:	Joep van der Heijden

© TASS B.V. 2013

Alle rechten voorbehouden. Verveelvuldiging, geheel of gedeeltelijk, is niet toegestaan dan met schriftelijke toestemming van de auteursrechthebbende.

All rights are reserved. Reproduction in whole or in part is prohibited without the written consent of the copyright owner.

Afstudeerverslag voor Fontys Hogeschool ICT

Gegevens student:

Naam: Joep van der Heijden
Studentnummer: 2135876
Afstudeerrichting: S (voltijd)
Afstudeerperiode: van 3 september 2012 t/m 18 januari 2013 (100 werkdagen)

Gegevens bedrijf:

Naam bedrijf: TASS Technology Solutions
Afdeling: Delivery
Plaats: Eindhoven
Bedrijfsbegeleider: Frits van de Water
Technisch begeleider: Jinmin Hu

Gegevens docentbegeleider:

Naam: Michael Franssen

Gegevens verslag:

Titel afstudeerverslag: Smartphones: develop once, run anywhere
Datum uitgifte: 10 januari 2013

Datum: 10 januari 2013



De bedrijfsbegeleider,
Frits van de Water

Documenthistorie

Versiebeheer

Versie	Auteur	Datum	Commentaar
0.0.1	Joep van der Heijden	14-12-2012	Eerste concept
0.1.0	Joep van der Heijden	04-12-2012	Concept uitbreiding, invulling van hoofdstukken, verwerking feedback
1.0.0	Joep van der Heijden	10-01-2013	Final versie

Distributielijst

Versie	Naam	Datum	Organisatie	Functie
0.0.1	Frits van de Water	14-12-2012	TASS	Bedrijfsbegeleider
0.0.1	Michael Franssen	14-12-2012	Fontys	Docentbegeleider
0.1.0	Frits van de Water	06-01-2013	TASS	Bedrijfsbegeleider
0.1.0	Jinmin Hu	06-01-2013	TASS	Technisch begeleider
0.1.0	Michael Franssen	06-01-2013	Fontys	Docentbegeleider
1.0.0	Michael Franssen	10-01-2013	Fontys	Docentbegeleider
1.0.0	Frits van de Water	10-01-2013	TASS	Bedrijfsbegeleider
1.0.0	Jinmin Hu	10-01-2013	TASS	Technisch begeleider

Voorwoord

Dit document is mijn afstudeerscriptie en is geschreven in het kader van het afstuderen bij TASS Technology Solutions voor de HBO opleiding Software Engineering aan de Fontys Hogeschool ICT te Eindhoven.

TASS ken ik van het evenement #devdate, een open en gratis evenement dat georganiseerd wordt door TASS. Toen ik TASS opbelde voor een mogelijke afstudeeropdracht kon ik meteen langskomen voor een gesprek. In dat gesprek kreeg ik een lijst voor mijn neus met redelijk wat opdrachten waar ik uit kon kiezen. Omdat het ontwikkelen voor smartphones en nieuwe technologieën tot mijn interesses behoren heb ik gekozen voor deze opdracht.

Mijn dank gaat uit naar alle mede-afstudeerders bij TASS waar ik een leuke tijd mee heb gehad, het was erg gezellig en iedereen kon elkaar altijd hulp bieden. Maar mijn dank gaat vooral uit naar mijn twee begeleiders van TASS. Allereerst Frits van de Water mijn bedrijfsbegeleider, hij zette mij altijd aan het denken en gaf mij de ruimte om mijn eigen invulling aan Scrum te geven. Ook stond hij altijd open voor mijn voorstellen van nieuwe functionaliteit. Jinmin Hu, mijn technische begeleider, bedank ik ook bij deze. Hij heeft mij laten inzien dat nieuwe technologieën, ook die nog niet af zijn, erg nuttig kunnen zijn.

Ik wens u veel leesplezier.

Joep van der Heijden

Eindhoven, januari 2013

Samenvatting	1
Summary	2
Verklarende woordenlijst	3
1 Inleiding	4
2 Over TASS	5
2.1 De organisatie	5
2.2 Plek van de afstudeerder	5
3 De opdracht	6
3.1 Achtergrond	6
3.2 Doelstellingen	6
3.3 Producten	7
3.4 Scope	7
3.5 Acceptatiecriteria	7
3.6 Aanpak	8
4 Onderzoek	12
4.1 Aanleiding & Doelstelling	12
4.2 Model	13
4.3 Vraagstelling	14
5 Methoden & Tools	15
5.1 Scrum	15
5.2 Behaviour Driven Development	18
5.3 Brackets	21
5.4 Versiebeheer	21
5.5 PhoneGap Builder	21
5.6 jQuery & jQuery Mobile	21
5.7 X-Tags	22
6 Realisatie	23
6.1 Sprint 0 – initialisatie	23
6.2 Sprint 1 – HTML5 webcomponent & inloggen	24
6.3 Sprint 2 – BDD & master-detail	26
6.4 Sprint 3 – detail-person & verbeteren testomgeving	28
6.5 Sprint 4 – detail-working & bewerkbare persoonsdata	30
6.6 Sprint 5 – stabiele versie	32
6.7 Sprint 6 – video	33
6.8 Sprint 7 – de afronding	35
7 Resultaten van het onderzoek	36
7.1 Wat zijn HTML5, CSS3 en JavaScript?	36
7.2 Wat voor apps worden er ontwikkeld door TASS?	37
7.3 Welke besturingssystemen ondersteunen HTML5?	38
7.4 Zijn er al apps die geschreven zijn in HTML5 en hoe worden deze ervaren?	39
7.5 Kan HTML5 dezelfde prestatie leveren als native?	40
8 Conclusies en aanbevelingen	41
Evaluatie	42

9 Literatuurlijst **43**

Bijlagen **44**

- Bijlage A: PID
- Bijlage B: Onderzoeksontwerp
- Bijlage C: Sprint notulen & Sprint Retrospectives
- Bijlage D: Document van bewijzen
- Bijlage E: Richtlijnen document
- Bijlage F: Onderzoeksverslag

Samenvatting

Tijdens de afstudeerperiode bij TASS Technology Solutions is er onderzoek uitgevoerd naar wat de technieken HTML5, CSS3 en JavaScript kunnen betekenen voor de app ontwikkelen van TASS.

TASS is een dienstverlener die vooral gespecialiseerd is in embedded software. Al sinds het begin van de smartphone ontwikkelt TASS apps voor de smartphone. Dit doet TASS door voor ieder besturingssysteem apart, in de programmeertaal die hiervoor bedoelt is, de app te ontwikkelen. Een zogenaamde native app. Deze programmeertalen zijn niet hetzelfde. Door dit te doen wordt er veel tijd besteed in het ontwikkelen van code die al in een andere programmeertaal geschreven is. De technieken HTML5, CSS3 en JavaScript zouden hier een oplossing voor kunnen bieden doordat deze technieken op alle besturingssystemen te gebruiken zijn.

Tijdens het project is gebruik gemaakt van de software ontwikkelmethode Scrum dit zorgde ervoor dat er iteratief aan het project gewerkt is. Hierdoor was er een goed beeld bij de opdrachtgever van wat hij kan verwachten.

In HTML5, CSS3 en JavaScript zijn componenten ontwikkeld die van toepassingen kunnen zijn bij andere projecten binnen TASS. Deze componenten zijn zo ontwikkeld dat ze makkelijk instelbaar zijn voor de ontwikkelaars zodat zij deze voor hun doel kunnen gebruiken. Om de kwaliteit van de code te kunnen garanderen is Behaviour Driven Development gebruikt om de functionaliteit te kunnen testen.

Met dit project is er een basis gelegd voor het ontwikkelen van herbruikbare componenten binnen projecten van TASS. De technieken HTML5, CSS3 en JavaScript kunnen veel betekenen voor TASS doordat er al veel mogelijk is. Wel moet er rekening gehouden worden met het inleveren van performance bij het ontwikkelen van een HTML5 app. Deze performance kan verbeterd worden door het gebruik van native plug-ins. Hierdoor worden HTML5, CSS en JavaScript technieken gebruikt in combinatie met native waardoor er een hybrid app ontstaat.

Summary

While completing this graduation project at TASS Technology Solutions research has been done to see how the standards HTML5, CSS3 and JavaScript can contribute to the app development of TASS.

TASS is a service provider. The company focuses on developing embedded software. From the birth of the smartphone TASS has been developing apps for smartphones. They do this by developing the app for each operating system in the programming language intended for this operating system. An app created this way is called a native app. The programming languages used to develop these apps are different for every operating system. The standards HTML5, CSS3 and JavaScript could change this because they are used on every current operating system.

During the execution of this project the agile software development method Scrum was used to have an iterative project. By doing this the participants knew very early what to expect from this project.

Using the standards HTML5, CSS3 and JavaScript components were developed that can be used in other projects by TASS. These components are developed in on a way they are easily configured for the developer needs. To insure the quality of the code Behaviour Driven Development was used for functional testing.

With this project a foundation is created for developing reusable components for projects within TASS. The standards HTML5, CSS3 and JavaScript can be very helpful to TASS because they are proven very versatile. But it does have a down side as well. The performance of HTML5 is not as good as then native standards are used. However, the performance could be improved using native plug-ins. By doing this the HTML5, CSS3 and JavaScript standards will be combined with native resulting in a hybrid app.

Verklarende woordenlijst

Term	Definitie
IDE	Integrated Development Environment. Een hulp programma voor het maken van software.
Scrum	De agile software ontwikkelmethode die is gebruikt tijdens dit project.
BDD	Behaviour Driven Development.
PID	Project Initiation Document.
PMP	Project Manage Plan.
iOS	Het besturingssysteem op de iPhone.
User story	Een beschrijving van functionaliteit door de ogen van een gebruiker.
JSON	JavaScript Object Notation, een standaard voor data formatting.
Sprint	Een iteratie voor het project.
HTML	HyperText Markup Language, een formaat voor de opmaak van documenten.
CSS	Cascading Style Sheet, een formaat voor de stijl van documenten.
JavaScript	Een programmeertaal die vooral gebruikt wordt in browsers.
NFC	Near Field Communication. Techniek voor het uitwisselen van data op korte afstand.

1 Inleiding

Een mobiele applicatie, ook wel app genoemd, maar één keer ontwikkelen en laten draaien op meerdere besturingssystemen. Dit is de droom van iedere programmeur die nu nog alle apps meerdere keren ontwikkelt zodat deze op ieder besturingssysteem draait. Is dit mogelijk? HTML5 kan wellicht deze dromen waarmaken.

TASS Technology Solutions, de opdrachtgever, ontwikkelt apps voor meerdere besturingssystemen. Dit doen zij voor ieder besturingssysteem anders. Om de time to market van de apps van TASS te kunnen verkorten zouden zij nu willen weten of dit anders kan.

Tijdens dit project is er gekeken of HTML5 daadwerkelijk iets kan betekenen voor de ontwikkeling van apps. Het proces van dit project is vastgelegd in deze scriptie.

In hoofdstuk 3 en 4 wordt de aanpak van de opdracht en onderzoek beschreven, daarna worden in hoofdstuk 5 alle methoden en tools die gebruikt zijn bij dit project benoemd en uitgelegd. Hoofdstuk 6 beschrijft het ontwikkelde product, een demo app bestaande uit een aantal componenten. Met hoofdstuk 7 en 8 wordt afgesloten met de resultaten van het onderzoek en de conclusie.

2 Over TASS

2.1 De organisatie

Zoals meer bedrijven in Eindhoven is ook TASS Technology Solutions vroeger een onderdeel van Philips geweest. In 2007 heeft Philips besloten dat TASS verkocht moest worden, het is toen onderdeel geworden van Total Specific Solutions, afgekort TSS.

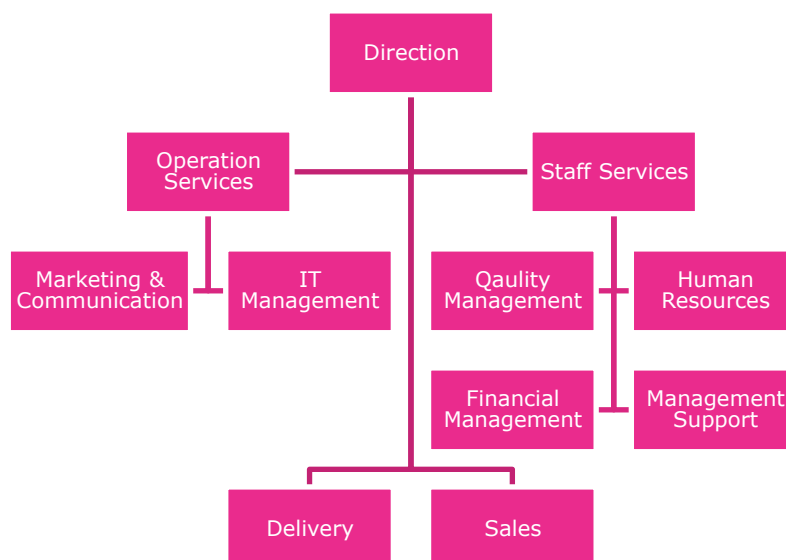
TASS is een dienstverlener en is vooral gespecialiseerd in embedded software. Ze heeft in Nederland twee vestigingen, naast het hoofdkwartier in Eindhoven is er ook een vestiging in Apeldoorn. In België heeft zij ook twee vestigingen gelegen te Gent en Leuven. TASS heeft iets minder dan 200 medewerkers in dienst waarvan het grootste deel gedetacheerd is, een veel kleiner deel werkt aan interne projecten.



Afbeelding 2-1 Het hoofdkantoor van TASS te Eindhoven

2.2 Plek van de afstudeerder

Onderstaande organogram is van toepassing binnen TASS. De afstudeerder bij TASS zitten bij de afdeling Delivery.



Afbeelding 2-2 Organogram van TASS

3 De opdracht



In dit hoofdstuk wordt de opdracht omschreven. Voor meer hierover zie het hoofdstuk projectdefinitie in het PID in bijlage A.

3.1 Achtergrond

De smartphone is de laatste jaren een apparaat geworden dat bijna iedereen gebruikt. Al sinds het begin van de smartphone ontwikkelt TASS apps voor de smartphone. Dit doet TASS door het ontwikkelen van zogenaamde native apps. Dit houdt in dat een app speciaal voor een specifiek besturingssysteem, zoals Android of iOS, ontwikkeld wordt. Als deze app dan ook op een ander besturingssysteem moet kunnen draaien betekent dit dat de app opnieuw geschreven moet worden voor het andere besturingssysteem. Door deze manier van ontwikkelen gaat er veel tijd zitten in het programmeren van code die er al is in andere programmeertalen.

Om te kunnen realiseren dat apps draaien op ieder besturingssysteem en maar een keer geschreven hoeven te worden kan er gebruik gemaakt worden van technieken die op ieder besturingssysteem bekend zijn. In dit project zal worden gekeken of de technieken HTML, JavaScript en CSS hier een goede oplossing voor kunnen bieden.

3.2 Doelstellingen

Dit project heeft als doelstelling om kennis op te doen hoe de time to market, de tijd die het kost voor het op de markt brengen, van apps ontwikkelt door TASS verkort kan worden door het gebruik van de technieken HTML5, JavaScript en CSS3.. Hierdoor worden er minder uren besteed door medewerkers van TASS aan onderdelen die al eens ontwikkeld zijn.

3.3 Producten

De volgende producten worden ontwikkeld:

- Het PID, Project Initiation Document; in dit document wordt het plan van aanpak uitgewerkt.
- PMP, Project Management Plan; in dit document wordt het plan van aanpak uitgewerkt in de stijl van TASS.
- Het onderzoeksontwerp; dit ontwerp heeft als doel om een goed beeld te scheppen over de aanpak van het onderzoek.
- Het onderzoeksverslag; de resultaten van het onderzoek worden hier in opgenomen.
- Document met bewijzen; een document waarin problemen en eventuele oplossingen staan genoteerd over de ontwikkeling.
- Richtlijnen; de documentatie over het gebruik van de gemaakte componenten.

3.4 Scope

Tijdens dit project worden er componenten ontwikkeld in HTML, JavaScript en CSS. De focus van de ontwikkeling van deze componenten liggen bij apps die al native ontwikkeld zijn of in de nabije toekomst ontwikkeld worden door TASS. Door het ontwikkelen van deze componenten wordt onderzocht of de functionaliteit gemaakt kan worden in HTML, JavaScript en CSS. Daarnaast wordt er onderzoek gedaan om de hoofdvraag te kunnen beantwoorden, zie hoofdstuk 4.

3.5 Acceptatiecriteria

De volgende criteria zijn opgesteld door de opdrachtgever om te kunnen beoordelen of de afstudeerstage een succes is:

Alle features die ontwikkeld gaan worden moeten beschreven worden in het document met richtlijnen en in het document met bewijzen moet hier meer technisch op ingegaan zijn. Hierin worden alle problemen en eventuele oplossingen opgesomd. Mocht een feature niet ontwikkeld kunnen worden moet dit beargumenteerd staan in de documenten.

3.6 Aanpak

Zoals beschreven in het PID wordt bij het afstuderen gebruik gemaakt van de methode competent afstuderen en stagelopen van Kempen en Keizer. In deze methode wordt het afstudeerproject opgedeeld in tien stappen en drie fases, beschreven in het hoofdstuk fasering. Voor deze methode is gekozen omdat er bij de aanvang slechts een vaag beeld is van wat het werk inhoudt en er geen idee is of de afgesproken opdracht in de beschikbare afstudeertijd past. De student mist de ervaring en het overzicht om dat goed te kunnen beoordelen.

Tijdens de realisatie en het uitvoeren van het onderzoek wordt Scrum toegepast, meer hierover is te lezen in hoofdstuk 5.1.

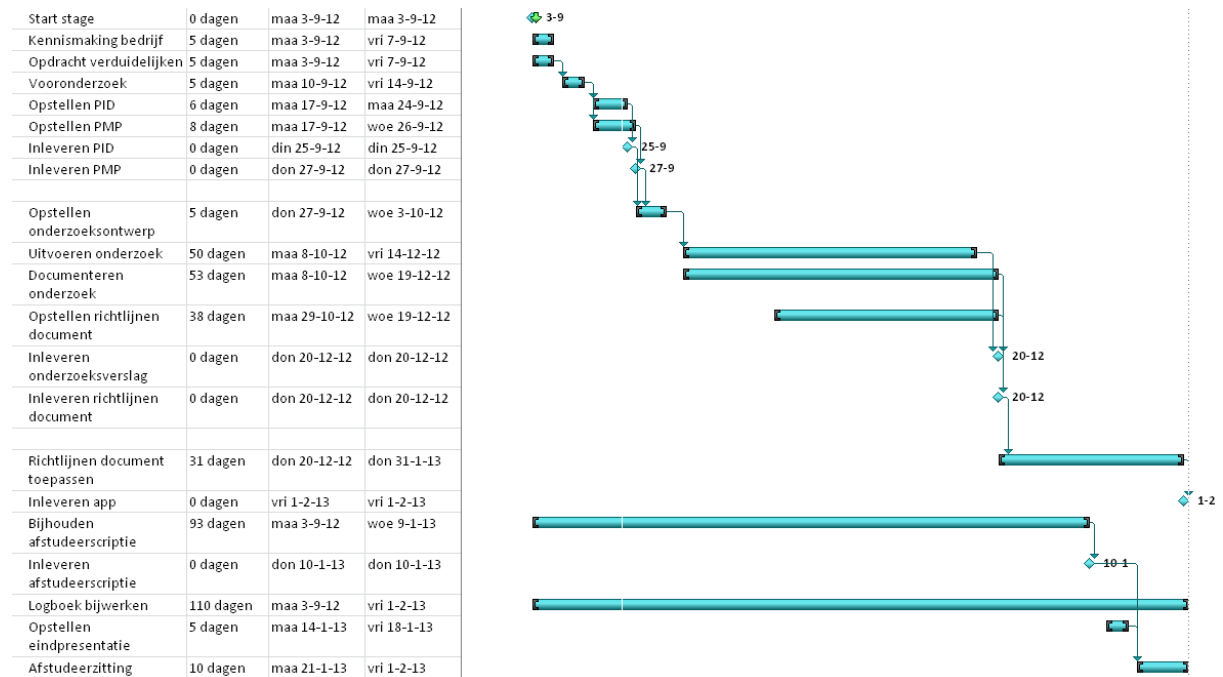
3.6.1 Fasering

In de onderstaande tabel wordt beschreven welke fases er worden uitgevoerd om de projectdoelstellingen te kunnen realiseren. Voor een gedetailleerde planning zie het hoofdstuk planning. Er is uitgebreider ingegaan op de activiteiten en producten in bijlage B van het PID.

Fase	Omschrijving	Planning
Oriëntatiefase	Oriëntatie op de afstudeeropdracht en een definitieve versie van het PID. Na het opstellen van het PID en goedkeuring door de opdrachtgever wordt er overgegaan naar de volgende fase.	03-09-2012 t/m 28-09-2012
Onderzoeks- en oplossingsfase	Aan het begin van deze fase wordt een onderzoeksontwerp gemaakt. Na goedkeuring door de opdrachtgever wordt het onderzoek uitgevoerd. Tijdens het uitvoeren van het onderzoek worden de bevindingen gedocumenteerd in het onderzoeksverslag. Na het onderzoek zal het document met richtlijnen worden opgesteld.	01-10-2012 t/m 21-12-2012
Invoeringsfase	De gerealiseerde documenten worden overgedragen aan de opdrachtgever. Toepassen van de richtlijnen op een app van TASS. Er wordt een presentatie gegeven aan de opdrachtgever. De scriptie wordt afgerond.	21-12-2012 t/m 01-02-2013

3.6.2 Planning

Zoals beschreven in bijlage B van het PID is de volgende planning opgesteld voor dit project



Afbeelding 3-1 De opgestelde planning aan het begin van het project

Milestones:

Start stage:	03-09-2012
Inleveren PID:	25-09-2012
Inleveren PMP:	27-09-2012
Inleveren onderzoeksverslag:	20-12-2012
Inleveren richtlijnen document:	20-12-2012
Inleveren afstudeerscriptie:	10-01-2013
Afstudeerzitting:	21-01-2013 – 01-02-2013
Inleveren app:	01-02-2013
Logboeken:	Wekelijks

3.6.3 Afwijkingen op de planning

De hieronder genoemde afwijkingen zijn gedaan op de planning zoals beschreven in het voorgaande hoofdstuk.

Toepassen richtlijnen op aparte app

Het richtlijnen document zal niet meer toegepast worden op een app, omdat de beschreven functionaliteit, die in de richtlijnen zijn beschreven, al toegepast wordt in de voorbeeld app. De componenten die zijn ontwikkeld zijn in één app gebundeld als een demo app.

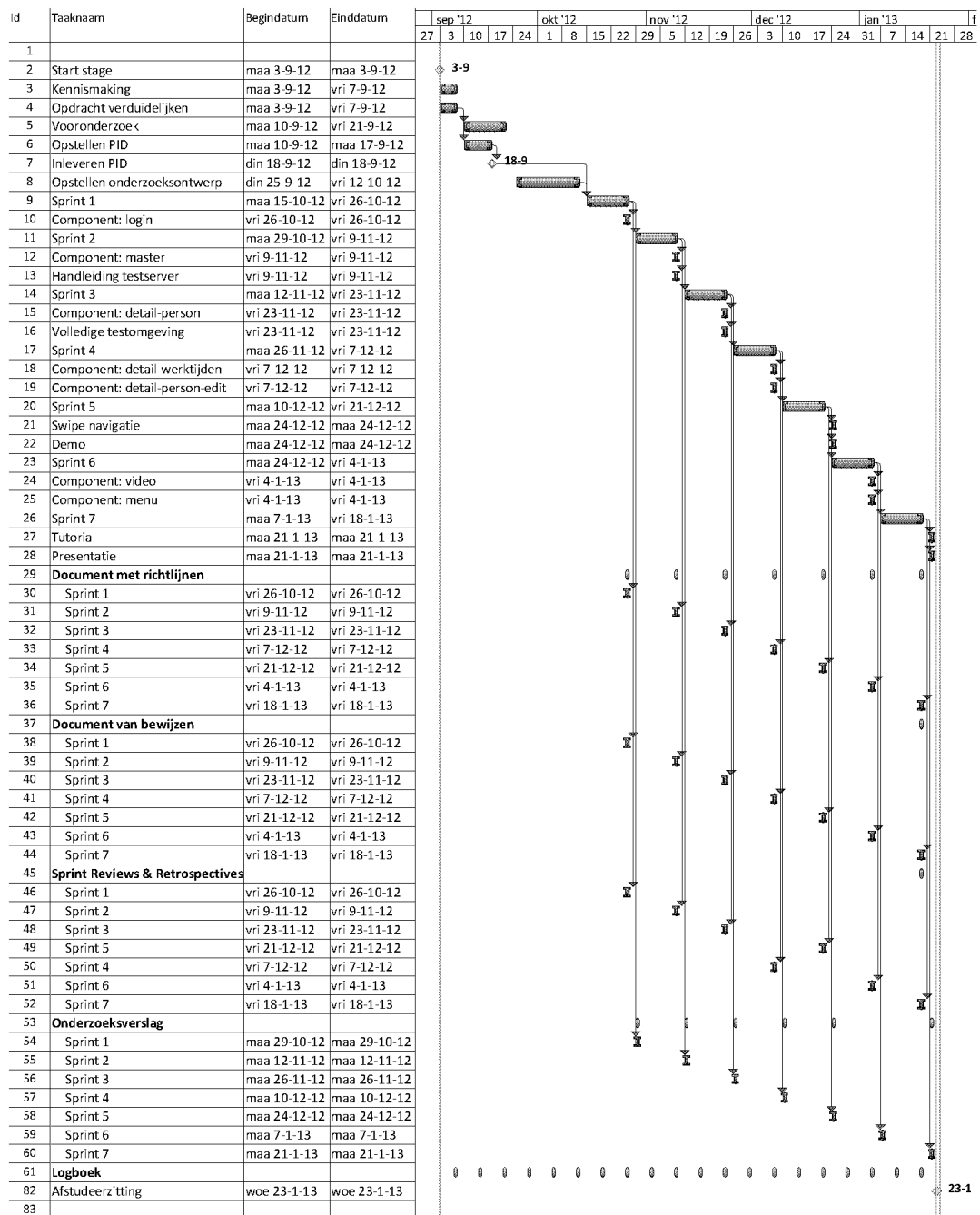
PMP

Er is in overleg met de opdrachtgever besloten dat het PMP niet verder uitgewerkt zal worden. Dit is besloten omdat in het PID alles wordt beschreven wat ook in het PMP van toepassing is.

Betere toepassing van Scrum in de planning

In de planning staan grote stukken gepland werk, dit is verdeeld over de gemaakte sprints zodat de planning een duidelijker beeld weergeeft over het ingeplande werk. De planning is er als volgt uit komen te zien:

Op de volgende pagina is de nieuwe planning in te zien.



Afbeelding 3-2 De nieuwe planning

4 Onderzoek



Het onderzoek dat zal worden uitgevoerd is gebaseerd op de methode van Verschuren en Doorewaard. Dit is gedaan omdat de afstudeerder bekend is met deze methode en omdat dit een goede houvast geeft aan het project. Het gehele onderzoeksontwerp is te vinden in bijlage B.

4.1 Aanleiding & Doelstelling

TASS ontwikkelt mobiele applicaties voor intern gebruik en voor klanten. Dit wordt nu nog gedaan op de native manier. Dit houdt in dat voor ieder besturingssysteem de applicatie in de programmeertaal die voor dat besturingssysteem bedoeld is geschreven moet worden. Voor iOS, het besturingssysteem van de iPhone, is dit de programmeertaal Objective-C. De native applicaties voor Android worden geschreven in Java.

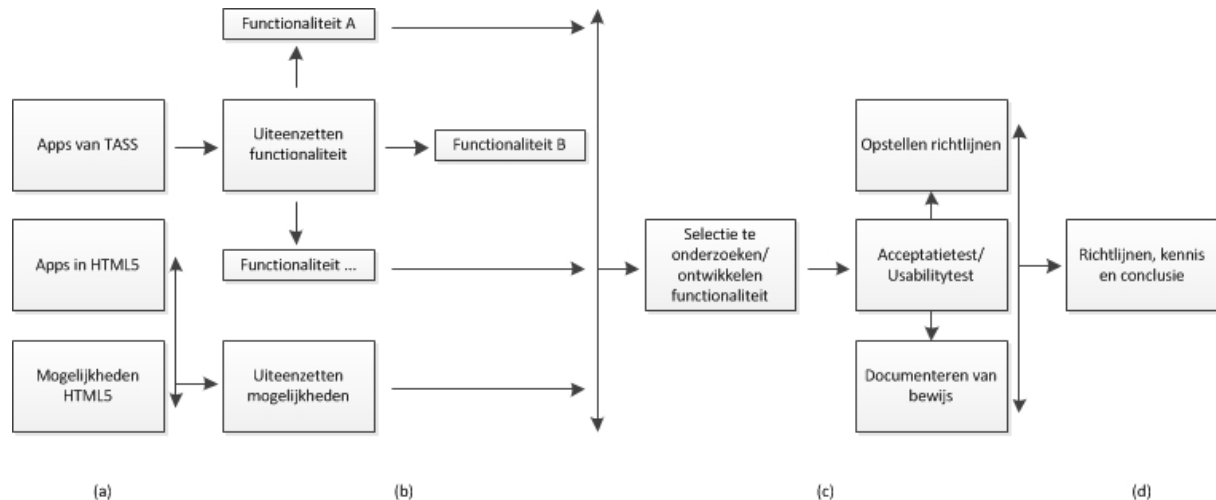
Door het uitvoeren van dit onderzoek wil TASS meer kennis opdoen over wat de technieken HTML5, JavaScript en CSS3 kunnen betekenen voor het ontwikkelen van mobiele applicaties voor de mobiele besturingssystemen.

Om tot een conclusie te kunnen komen wordt er gekeken naar wat voor apps TASS nu, en in de nabije toekomst gaat ontwikkelen. Deze functionaliteiten zullen dan ontwikkeld worden in technieken HTML5, JavaScript en CSS3. De functionaliteiten worden vervolgens door de opdrachtgever in een acceptatietest getest. Ook zullen er testen plaatsvinden onder de smartphone gebruikers om te kijken of zij merken dat de ontwikkelde functionaliteiten anders reageren dan dat zij gewend zijn.

De onderzoeksvraag, vat bovenstaande tekst samen en luidt als volgt: In hoeverre kan HTML5 de native apps van TASS vervangen?

4.2 Model

Het onderzoek dat wordt uitgevoerd is weergegeven in het onderstaande onderzoeksmodel, behorende bij het onderzoeksontwerp (bijlage B).



Afbeelding 4-1 Het onderzoeksmodel

In de eerste fase van het onderzoek wordt bekeken wat voor apps ontwikkeld worden door TASS en wat zij verwachten te ontwikkelen in de toekomst. Dit wordt gedaan door medewerkers en de opdrachtgever te interviewen. In deze fase wordt ook gekeken naar mogelijke oplossingen in HTML5, JavaScript en CSS3 (a).

Hierna wordt alle gevonden informatie uiteengezet (b) zodat hier een selectie van kan worden gemaakt, welke vervolgens ontwikkeld wordt. Dit wordt daarna getest door de opdrachtgever en smartphone gebruikers. De resultaten van deze test vormen het bewijs. Met de kennis die is opgedaan wordt een document met richtlijnen opgesteld(c). De resultaten van de testen worden opgenomen in de conclusie (d).

Omdat er gewerkt wordt met de Scrum ontwikkelmethode zullen per iteratie een aantal functionaliteiten het model doorlopen i.p.v. dat dit meteen met alle functionaliteit gedaan wordt. Voor meer over Scrum zie hoofdstuk 5.1.

4.3 Vraagstelling

Om de onderzoeksvraag, in hoeverre kan HTML5 de native apps van TASS vervangen, te kunnen beantwoorden worden eerst de volgende deelvragen beantwoord.

1. Wat zijn HTML5, CSS3 en JavaScript?

Bij het beantwoorden van deze vraag wordt gekeken wat er allemaal nieuw is vergeleken met de vorige HTML en CSS versie en wat JavaScript kan betekenen bij het ontwikkelen van een app.

2. Wat voor apps worden er ontwikkeld door TASS?

Hier wordt gekeken wat voor TASS nu al heeft ontwikkeld en wat voor apps zij in de toekomst gaan ontwikkelen. Door deze vraag te beantwoorden is er een duidelijker beeld wat een HTML5 app moet kunnen om iets te kunnen betekenen voor TASS.

3. Welke besturingssystemen ondersteunen HTML5?

Er wordt onderzocht of HTML5 ondersteund wordt op de besturingssystemen Android en iOS en of HTML5 ook wordt geaccepteerd.

4. Zijn er al apps die geschreven zijn in HTML5 en hoe worden deze ervaren?

Er wordt hier gezocht naar apps die geschreven zijn in HTML5 voor verschillende besturingssystemen. Er wordt gekeken naar reacties van gebruikers om een goed beeld te vormen hoe deze ervaren worden.

5. Kan HTML5 dezelfde prestatie leveren als native?

Hier wordt gekeken of de mogelijkheden in HTML5 ook een goede prestatie kunnen leveren, net zoals native. En zo ja, hoe dit dan wordt gedaan; zo nee, hoe dit opgelost zou kunnen worden.

5 Methoden & Tools

5.1 Scrum

Scrum is een agile software ontwikkelingsmethode, het is flexibel te beheren. Dit komt doordat er meerdere momenten zijn in het project dat er een deel van het product opgeleverd wordt. Er is voor deze methode gekozen omdat er veel kennis en support over deze methode is binnen TASS. Maar nog veel belangrijker omdat er meerdere momenten zijn voor een review op het gemaakte werk. Hierdoor kan er snel bijgestuurd worden mocht dit nodig zijn. Scrum kan op vele manieren toegepast worden. Hieronder wordt beschreven hoe het in dit project is toegepast.

Scrum is een iteratief proces. Dit betekent dat er meerder iteraties uitgevoerd worden om tot het resultaat te komen. Een iteratie wordt in Scrum een sprint genoemd. Aan het einde van een sprint wordt er een sprint retrospective gemaakt waarin verbeteringen afgesproken worden die toegepast gaan worden in de volgende sprint.

Om een overzicht te krijgen van wat er allemaal gedaan moet worden wordt er een product backlog bijgehouden waarin alle wensen staand vermeld voor het product. Dit wordt gedaan in de vorm van user stories, dit zijn korte feature beschrijvingen. De product backlog wordt aan de start van het project en tijdens de uitvoering van het project gevuld.

In een user story wordt verteld wie de gebruiker is, bijvoorbeeld een administrator, moderator of een standaard gebruiker. Daarnaast wat de gebruiker wil en hoe deze gebruiker dit doet. Een user story over bijvoorbeeld de feature om in te kunnen loggen ziet er als volgt uit:

In order to get access to the functionality of the app
As a user
I want to log in

Een user story heeft één of meerdere taken die uitgevoerd gaan worden door het projectteam. In dit project zijn alle scenario beschrijvingen die bij de user story horen als een taak geclassificeerd. Daarnaast zijn er nog andere taken zoals een test opzetten, een schets maken of de CSS van de pagina maken. De taak heeft een naam en een beschrijving, waar dieper ingegaan wordt wat er precies gedaan moet worden. Voor iedere taak wordt het aantal uren ingeschat dat het projectteam denkt nodig te hebben. Zo heeft de user story log in onder andere de volgende taken:

Naam: happy flow
Beschrijving: Login scherm wordt getoond
Gebruiker geeft zijn inloggegevens
Systeem checkt inloggegevens (simulatie)
Systeem slaat inloggegevens op
Doorsturen naar volgende scherm
Tijd: 3 uur

Naam: schets
Beschrijving: Voorstel voor schets
Tijd: 1 uur

Naam: opmaak
Beschrijving: Het maken van de CSS voor de pagina in TASS stijl
Tijd: 8 uur

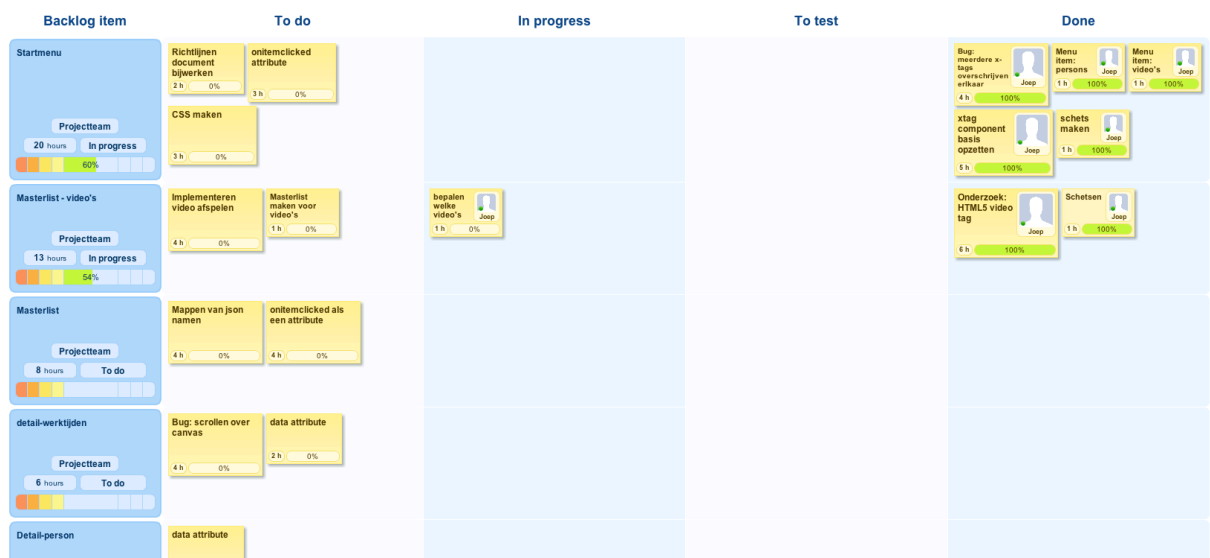
Als er een nieuwe sprint wordt gestart worden er user stories met de te maken taken in de sprint backlog gezet. Hierdoor is er inzicht in wat er in de sprint gedaan moet worden. Alle taken worden toegewezen aan een projectlid.

Als een projectlid begint met een taak wordt de status van *to do* naar *in progress* gezet. Vindt het projectlid dat de taak is afgerond dan krijgt de taak de status *to verify*. De scrummaster, dit is een persoon die het proces beheert, zet vervolgens de status op *done* of weer terug op *in progress*.

Een user story in een sprint is helemaal voltooid als alle taken op done staan en als aan alle punten is voldaan van de *definition of done*. Dit is een checklist met punten. Voor dit project ziet de definition of done er als volgt uit:

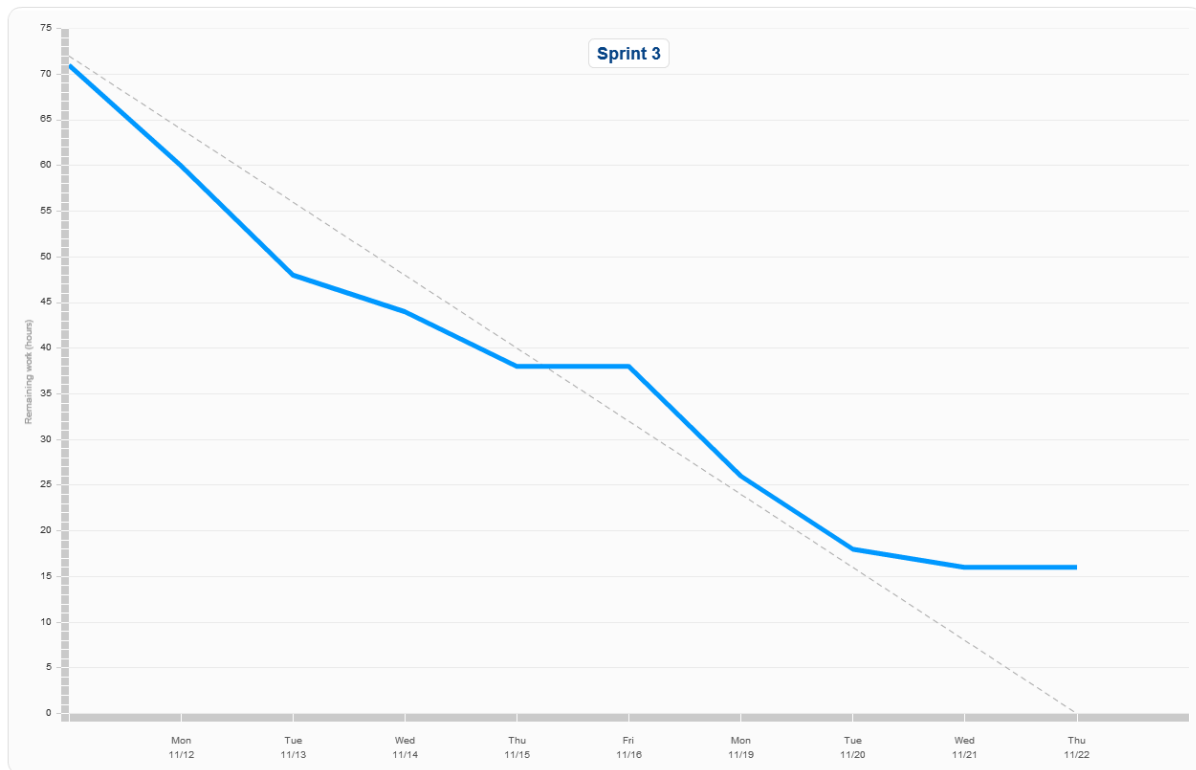
- **Al de code is geschreven**
- **De code is getest zoals beschreven in het testplan**
- **De code is ingecheckt in SVN**
- **De wiki is up-to-date**
- **Het document van bewijzen is bijgewerkt**
- **Het richtlijnen document is bijgewerkt**
- **Geen openstaande bevindingen**
- **De user story is geaccepteerd door de klant**

Dit alles wordt bijgehouden op het scrumboard en de burndownchart. Op het scrumboard staan vijf kolommen waarvan één voor de user story en vier kolommen die de status van een taak aangeven: *to do*, *in progress*, *to test* en *done*. Iedere user story heeft een rij met deze kolommen met daarin alle taken van deze user story.



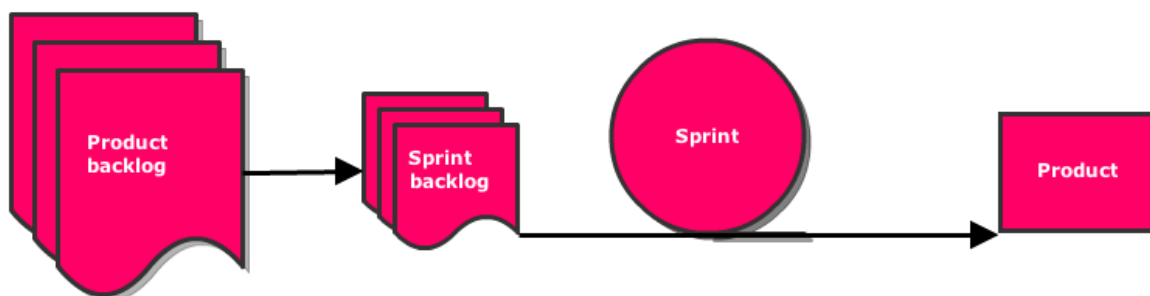
Afbeelding 5-1 Voorbeeld van een scrumboard

Op de burndownchart, hieronder weergegeven, is te zien hoeveel uur er nog gedaan moet worden voor de sprint. De gestippelde lijn geeft aan hoe een perfect ingeplande sprint en de uitvoering daarvan eruit zou zien. De doorgetrokken lijn geeft de daadwerkelijke aantal uren aan. Dit verschilt van de gestippelde lijn omdat geschatte tijden soms niet kloppen of omdat er soms een taak bijkomt waar nog niet over nagedacht was.



Afbeelding 5-2: de burndownchart van sprint 3

Het gehele scrumproces is afgebeeld in onderstaande afbeelding:



Afbeelding 5-3 Het scrumproces

5.2 Behaviour Driven Development

Behaviour Driven Development (BDD) is afgeleid van Test Driven Development (TDD). TDD is een methode die testers en ontwikkelaars gebruiken om klassen en methodes te testen door middel van unit testen. Dit heeft als gevolg dat het ware gedrag van het gehele systeem niet getest wordt. BDD is hier een oplossing voor. Bij deze methode worden bepaalde scenario's getest op het gehele systeem. Hier komt dan ook het voordeel bij kijken dat het makkelijker is om tests te schrijven op de BDD methode dan op de TDD methode, omdat bij TDD de classes en methodes al moeten bestaan. Bij BDD wordt juist het gedrag getest. Deze methode van ontwikkelen is erg goed toepasbaar bij Scrum omdat hier het gedrag en functionaliteit wordt beschreven in de user stories.

5.2.1 Tools

Om Behaviour Driven Development te kunnen implementeren in dit project wordt gebruik gemaakt van Cucumber. Cucumber is een Ruby Gem, een library of programma geschreven voor de programmeertaal Ruby, die zogenaamde feature bestanden uitvoert. In een feature bestand staat allereerst de feature beschreven zoals dit gedaan wordt in een user story (zie 5.1 Scrum). Daarna staan stappen die beschrijven wat het gedrag moet zijn van bepaalde scenario's. Een test ziet er dan als volgt uit:

Feature: log in

In order to get access to the app's functionality
As a user
I want to log in

Background:

Given I am on the /index.html#login page
Given I have not logged in yet

Scenario: happy flow

Given I am on the /index.html#login page
Then I should be on the /index.html#login page with content "Inloggen"
When I fill in "username" with "heijdvvdj@tass.nl"
And I fill in "password" with "mypwd"
And I click on "bt_login"
Then I should be on the /index.html#main page with content "Main"
And the username "heijdvvdj@tass.nl" and password "mypwd" are saved

Scenario: bad flow 1, Al een keer ingelogd

Given I am on the /index.html#login page
Then I should be on the login page with content "Inloggen"
When I fill in "username" with "heijdvvdj@tass.nl"
And I fill in "password" with "mypwd"
And I click on "bt_login"
Then I should be on the main page with content "Main"
Given I am on the /index.html#login page
Then I should be on the main page with content "Main"

Gherkin

De taal waarin deze tests geschreven zijn heet Gherkin (vertaal naar Nederlands: augurkje). Deze taal is ontstaan door het Cucumber project en de parser hiervoor is te downloaden via de Ruby Gems. De Gherkintaal bestaat uit maar weinig woorden en is mede daarom makkelijk te begrijpen. Een andere reden waarom deze taal makkelijk te begrijpen is is omdat de rest van de tests in menselijke taal geschreven is waardoor iedereen de tests zou kunnen schrijven.

Cucumber gebruikt zogenaamde step definitions om de Gherkin taal in het feature bestand te koppelen aan code. Dit zijn bestanden die de tests beschrijven. Als we uit de hierboven beschreven test de volgende stap pakken:

```
Given I am on the /index.html#login page
```

Wordt deze gekoppeld in het step definition bestand door de volgende code:

```
Given /^I am on the (.*) page$/ do |uri|  
  visit uri  
end
```

Capybara

In de bovenstaande code wordt de methode visit aangeroepen. Deze methode komt uit de Ruby Gem genaamd Capybara. Dit is een library die vaak in combinatie met Cucumber wordt gebruikt en komt met veel methodes die een simulatie geven hoe de gebruiker met een web applicatie om zal gaan. Dit zijn naast de methode visit, interactie methodes met de user interface zoals:

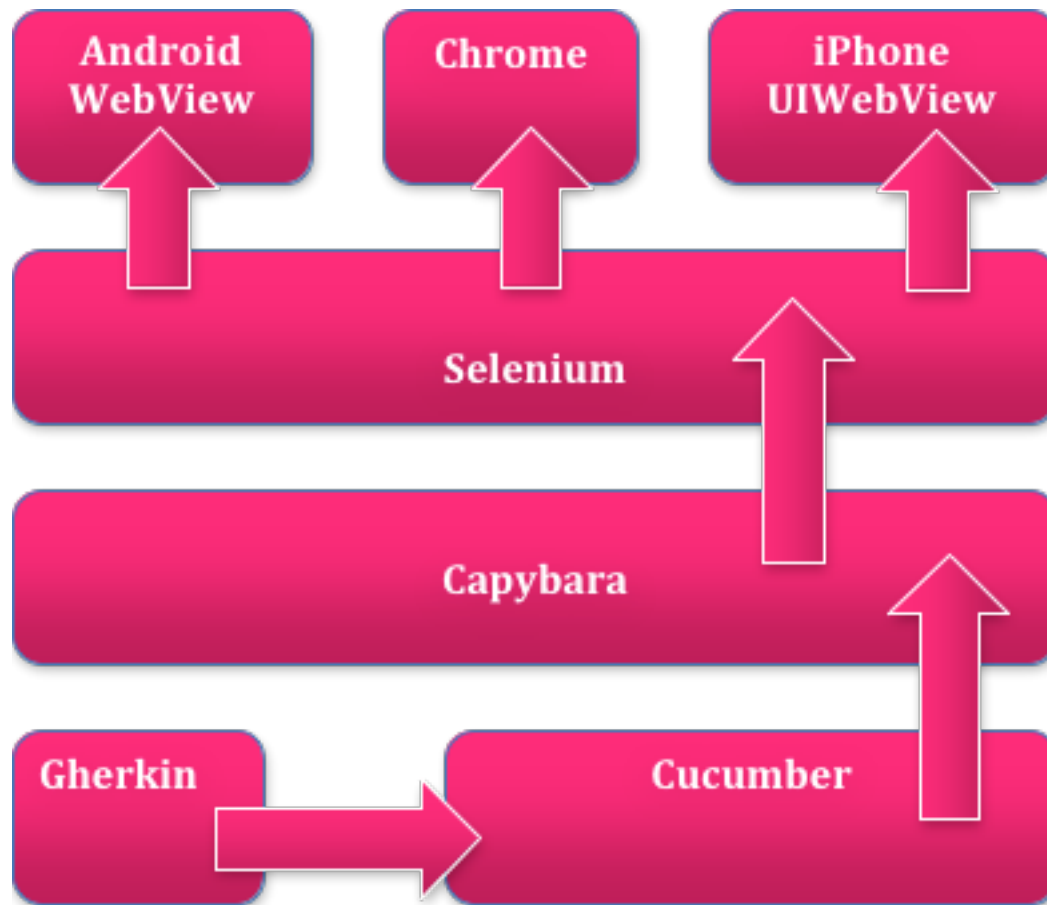
- click_on, zorgt ervoor dat er op een link of knop wordt gedrukt.
- fill_in, vult een veld in met de meegegeven waarde.
- have_content, kijkt of de meegegeven waarde voorkomt op de pagina.

Naast dit soort methodes biedt Capybara ook de mogelijkheid om JavaScript code uit te voeren, dit wordt gedaan met de methode execute_script.

Selenium

Capybara kan dit allemaal uitvoeren in een browser door gebruik te maken van Selenium. Dit is een driver die ervoor zorgt dat er gecommuniceerd kan worden naar een browser. Selenium kan ook van afstand aangesproken worden; er is een app beschikbaar voor iPhone en Android. Deze app maakt gebruik van het WebView component van het besturingssysteem.

Al de bovenstaande beschreven talen en technieken over Behaviour Driven Development die toegepast zijn binnen dit project zijn in het onderstaande diagram samengevat:



Afbeelding 5-4 Diagram van de testtool

5.3 Brackets

De IDE, Integrated Development Environment, die gebruikt wordt heet Brackets. Dit is een open-source project van Adobe. Deze applicatie is geheel geschreven in HTML, CSS en JavaScript. Een belangrijke functie in deze IDE is live development. Deze functie zorgt ervoor dat CSS aangepast kan worden en dat het resultaat meteen zichtbaar is in de browser. Ook wordt de pagina in de browser vernieuwd als een verandering opgeslagen wordt.



Afbeelding 5-5 Het logo van Brackets

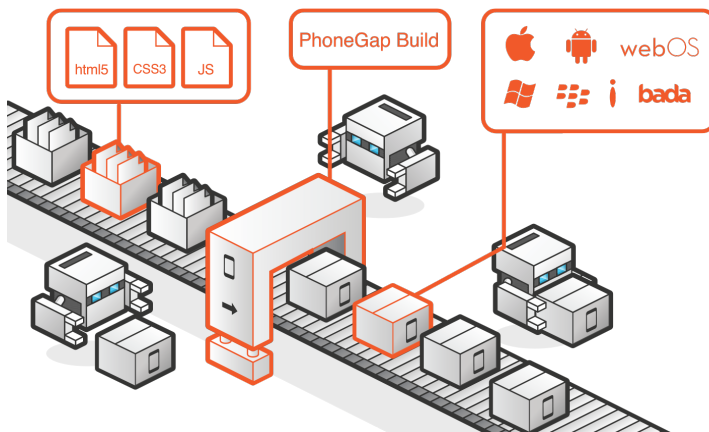
5.4 Versiebeheer

Voor het versiebeheer is gebruik gemaakt van SVN. Hier is voor gekozen omdat TASS hiervoor een server in gebruik heeft. TASS is bezig met het opzetten van een Git server, er zou in de toekomst hiernaar overgestapt kunnen worden.

Alle documentatie en code wordt opgeslagen in de trunk map. Na iedere sprint wordt er een kopie van deze map opgeslagen in de tags map.

5.5 PhoneGap Builder

Om de geschreven code van de app ook daadwerkelijk als een app op een Android en iPhone toestel te gebruiken wordt de dienst PhoneGap Builder gebruikt. Deze dienst build de gemaakte app in de cloud voor de mobile besturingssystemen iOS, Android, WindowsPhone, Blackberry, webOS, bada en Symbian. Daarnaast biedt PhoneGap de mogelijkheid om native plug-in te gebruiken. Deze plug-ins zijn geschreven in de native taal van het besturingssysteem en zijn aan te roepen vanuit JavaScript.



Afbeelding 5-6 Het build proces van PhoneGap afgebeeld

5.6 jQuery & jQuery Mobile

Tijdens het ontwikkelen van de functionaliteit is er gebruik gemaakt van jQuery en jQuery Mobile. Er is aangegeven door de opdrachtgever om dit te gebruiken. jQuery is een JavaScript Library die het makkelijker maakt voor de ontwikkelaar om de HTML te bewerken. jQuery Mobile is specifiek bedoeld voor mobiele platformen. jQuery Mobile zorgt er bijvoorbeeld voor dat de pagina's en de navigatie daartussen makkelijk te ontwikkelen is.

5.7 X-Tags

Om herbruikbare componenten te kunnen realiseren wordt gebruik gemaakt van X-Tag, in hoofdstuk 6.2.1 is uitgelegd waarom hiervoor gekozen is. Op de website van X-Tag leggen ze als volgt uit wat X-Tag precies is:

"X-Tag is a small JavaScript library, created and supported by Mozilla, that brings Web Components Custom Element capabilities to all modern browsers."

X-Tag zorgt er dus voor dat eigen componenten gemaakt kunnen worden en ook werken op alle moderne browsers. X-Tag zorgt er ook voor dat er eigen attributen en methodes gebruikt kunnen worden die bij dat specifieke component horen. Zo is er een X-Tag component dat in dit project ontwikkeld is genaamd master. Dit is een component dat een lijst weergeeft. Om een andere ontwikkelaar dit component te laten gebruiken moet hij de juiste bestanden in zijn HTML bestand zetten. Allereerst is dit de X-Tag library zelf, gevolgd door het JavaScript bestand waarin het mastercomponent gemaakt is. Om de CSS te gebruiken voor dit component moet het CSS bestand ook aan het HTML bestand worden toegevoegd. Daarnaast maakt het component ook nog gebruik van jQuery. In onderstaand voorbeeld wordt een mastercomponent gebruikt om een lijst weer te geven met namen. In het richtlijnen document, te vinden in bijlage E, staan de specificaties van dit component.

```
<!DOCTYPE HTML>
<html>
<head>
  <script src="/jQuery.js"></script>
  <script src="/x-tag.js"></script>
  <script src="/master.js"></script>
  <link rel="stylesheet" href="/master.css"/>
</head>
<body>
  <x-master data-json-src="people.json" data-name="true"
    data-json-name="name">
  </x-master>
</body>
</html>
```

Een component kan ook dynamisch aangemaakt worden in JavaScript en de methodes van het mastercomponent kunnen dan worden aangeroepen. Dit gebeurt als volgt:

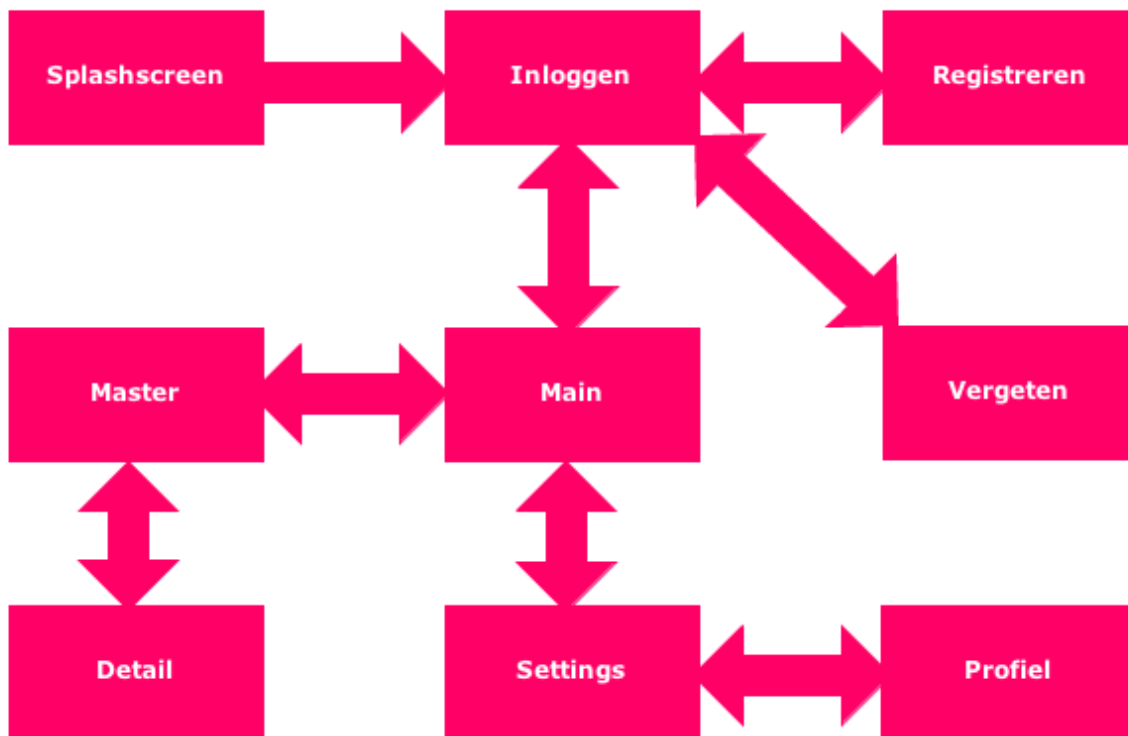
```
var master = document.createElement('x-master');
master.setSource('people.json');
master.setName(true);
master.setJSONName('name');
```

6 Realisatie

De volgende hoofdstukken geven een samenvatting van de belangrijkste werkzaamheden die tijdens de sprints zijn gemaakt. De sprint review notulen en de sprint retrospectives zijn te vinden in bijlage C. Hierin staat alles genoteerd wat er in een sprint is gedaan en of er eventuele verbeterpunten zijn.

6.1 Sprint 0 – initialisatie

In de eerste sprint van dit project is het PID gemaakt en het onderzoeksontwerp gemaakt. Verder is er gezocht naar onderzoeksmateriaal en is dit materiaal globaal doorgelezen en bewaard. De belangrijkste punten uit het PID en het onderzoeksontwerp zijn genoemd in hoofdstukken 3 en 4. Daarnaast is er gekeken naar wat voor apps TASS maakt. Samen met de opdrachtgever is onderstaand schema gemaakt. Hier worden belangrijke componenten genoemd die in veel apps terug zouden kunnen komen.



Afbeelding 6-1 De belangrijke componenten van een TASS app

6.2 Sprint 1 – HTML5 webcomponent & inloggen

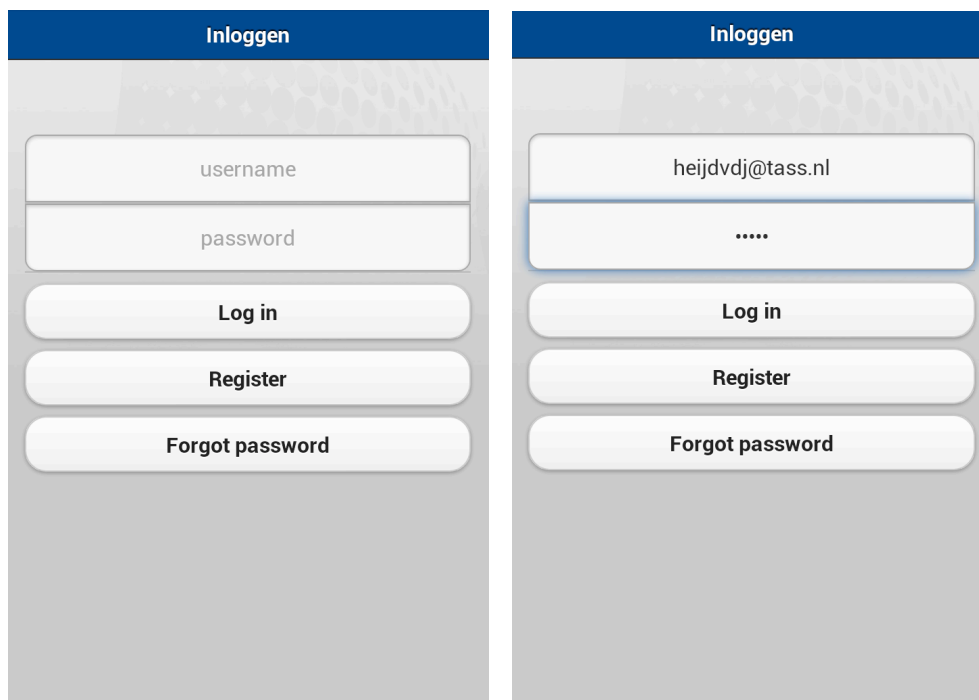
In sprint 1 is gekeken naar hoe herbruikbare webcomponenten gerealiseerd kunnen worden. Dit is op het begin gedaan zodat er in dit project op dezelfde manier componenten gemaakt kunnen worden. Deze componenten zullen dan inzetbaar zijn bij andere projecten.

6.2.1 Webcomponent & X-Tags

In de eerste sprint is gekeken hoe er herbruikbare componenten gemaakt kunnen worden. De nieuwe HTML5 feature zou hier een oplossing voor kunnen bieden. Helaas ondersteunen de browsers deze nieuwe feature van HTML5 nog niet. Dit omdat de specificaties van de webcomponent feature nog in behandeling zijn. Er is toen gezocht naar een mogelijke oplossing. Mozilla, de organisatie achter de webbrowser Firefox, heeft een open-source project genaamd X-Tags. X-Tags is een JavaScript library die ervoor zorgt dat zelfgemaakte HTML tags gebruikt kunnen worden en is gebaseerd op de W3 specificaties van webcomponents. Deze mogelijke oplossing is gecommuniceerd naar de opdrachtgever met het advies om dit te gaan gebruiken tot het moment dat webcomponents gebruikt kunnen worden. Dit advies is door de opdrachtgever overgenomen.

6.2.2 Het Logincomponent

Het eerste X-Tag component dat is gerealiseerd is het inlogcomponent. De richtlijnen van dit component zijn vastgelegd in het richtlijnen document te vinden in bijlage E.



Afbeelding 6-2 Screenshots van het login component. Rechts zijn de username en wachtwoord ingevuld.

6.2.3 Pagina navigatie met jQuery Mobile & CSS3 transitions

Verder is in deze sprint gekeken naar de navigatie tussen verschillende HTML pagina's. jQuery Mobile biedt een oplossing om te kunnen navigeren tussen HTML pagina's met een overgangseffect. Dit werkt alleen niet goed tussen twee losse HTML bestanden. Wel werkt het als beide HTML pagina's in één bestand staan. Dit wordt gedaan door een HTML element de waarde page mee te geven op de attribute data-role. Omdat jQuery Mobile op deze manier afdwingt om alle pagina's in één bestand te zetten, en zo het overzicht snel verloren kan gaan, is er gezocht naar een alternatief.

De alternatieve oplossing die is bedacht, is om twee HTML elementen te nemen en die als pagina's te gebruiken. Eén element geeft dan de huidige pagina weer en in het andere element wordt de toekomstige pagina geladen. Door gebruik te maken van de nieuwe CSS3 transition feature kan er tijdens de overgang tussen deze twee pagina's een animatie af worden gespeeld.

Nadat dit is gecommuniceerd naar de opdrachtgever heeft hij aangegeven toch gebruik te willen maken van de jQuery Mobile oplossing en dus alle HTML pagina's in één bestand te zetten. Hier is voor gekozen omdat de CSS3 transition erg schokkerig waren. Door het gebruik van x-tag hoeft alles niet meer in één bestand komen te staan dus kan het overzicht toch gehouden worden.

6.3 Sprint 2 – BDD & master-detail

In sprint 2 is een testserver opgezet om de tools voor BDD, genoemd in hoofdstuk 5.2, op te laten draaien. Daarnaast is er een nieuw component ontwikkeld, het master component. Ook is er een begin gemaakt voor een swipe gesture die gebruikt kan worden voor navigatie.

6.3.1 Testserver

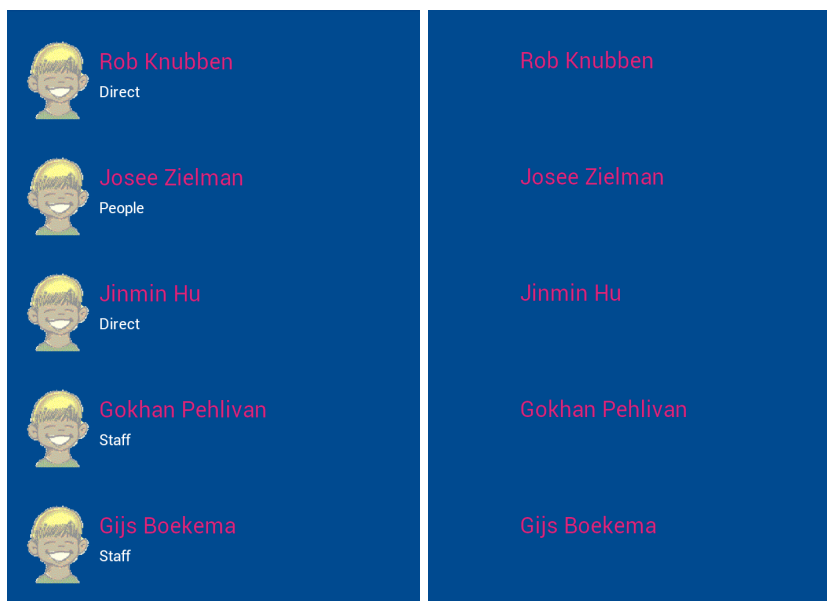
Bij TASS zijn ze startende met het gebruik van BDD. Er is een image beschikbaar van een server waar deze tools opstaan zodat een nieuwe server makkelijk op te zetten is. Helaas voldeed deze image niet aan de behoefte van dit project. Het besturingssysteem wat gebruikt wordt op de image is Ubuntu Server. Omdat deze versie van het besturingssysteem geen desktop omgeving heeft, konden de HTML bestanden niet getest worden in de webbrowser Chrome. Daarom is ervoor gekozen om een nieuwe virtuele machine op te zetten die Ubuntu Desktop draait. Op deze virtuele machine zijn toen de tools geïnstalleerd. Er is een handleiding gemaakt voor de installatie van deze virtuele machine en op de TASS wiki gezet.

6.3.2 Mastercomponent

Het doel van het mastercomponent is om een lijst te kunnen weergeven voor verschillende doeleinden. Als voorbeeld is gekozen om een lijst met werknemers weer te geven, per rij in deze lijst kan dan een foto, een naam en een beschrijving weer gegeven worden. Dit component is zo ontwikkeld dat andere ontwikkelaars die dit component bij TASS gaan gebruiken het zo kunnen configureren dat het lijstje ook alleen een naam of alleen een foto of beschrijving kan weergeven of een combinatie van de drie.

Als de gebruiker op een werknemer drukt in deze lijst wordt hij naar een nieuwe pagina genavigeerd. Op deze pagina zullen de details van de werknemer getoond worden. Zo'n constructie heet een master-detail pattern. Deze detail pagina is in sprint 3 ontwikkeld.

Om een "wow-effect" te krijgen voor dit lijstje is er een effect toegevoegd voor bij het scrollen door de lijst. Hier is stroll.js voor gebruikt, dit is een open source Chrome Experiment. Dit werkt goed als er gescrold wordt door tekst maar blijkt te haperen als er foto's in de lijst worden weergegeven. Er is daarom besloten dit effect toch niet te gaan gebruiken.



Afbeelding 6-3 Screenshots van het mastercomponent. Links een lijst met werknemers met foto, naam en afdeling. Rechts een lijst met werknemers met alleen de naam.

6.3.3 Swipe gesture

Om te kunnen realiseren dat op de iPhone de gebruiker naar de vorige pagina kan navigeren is er in deze sprint begonnen met het ontwikkelen van een swipe gesture, de gebruiker veegt met zijn vinger naar rechts. Dit bleek op de iPhone nog niet goed te werken, er is besloten om dit in een andere sprint terug te laten komen omdat dit nu een lage prioriteit heeft. Dit is gerealiseerd in sprint 5, zie hoofdstuk: 6.6.2

6.4 Sprint 3 – detail-person & verbeteren testomgeving

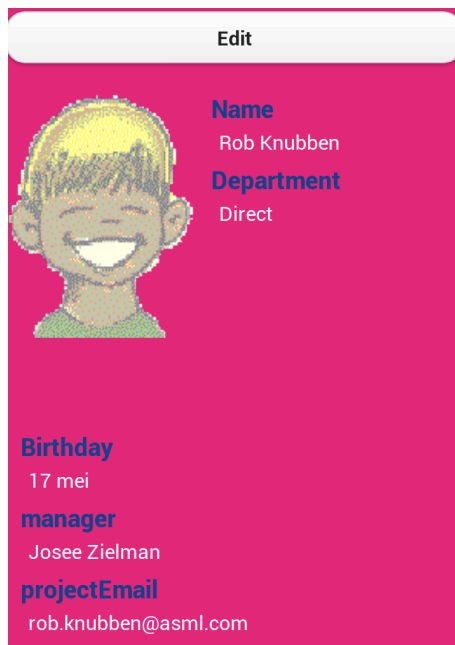
In de derde sprint is het detail-person component ontwikkeld. Ook is gekeken naar een animatie in het master component, de reactietijd van een knop en het draaien van de tests op de mobiele apparaten. Er is besloten om niet te gaan werken met WindowsPhone in deze sprint.

6.4.1 Detail-person component

Dit component laat de data zien van een persoon, het bestaat uit drie delen:

1. De foto van de persoon.
2. Belangrijke informatie, deze wordt getoond naast de foto.
3. Overige informatie, deze wordt getoond onder de foto.

De ontwikkelaar die dit component gebruikt kan de namen van de keys van het persoon dat meegegeven wordt aan dit component meegeven aan het component zodat alle data op de juiste plek getoond worden.



Afbeelding 6-4 Screenshot van het detail-person component

6.4.2 Reactietijd knop

Er is geconstateerd dat reactietijden van knoppen erg hoog zijn. Dit blijkt een bekende bug te zijn in jQuery Mobile. Er is een plug-in geïmplementeerd voor jQuery die ervoor zorgt dat er om deze bug heen gewerkt wordt. Deze code is gebaseerd op een artikel van Google ¹over het maken van snelle knoppen in mobiele web applicaties. De reactietijd van de knop was hoog omdat de WebView 300ms wacht na de eerste druk op de knop voor een dubbel klik actie.

¹ https://developers.google.com/mobile/articles/fast_buttons

6.4.3 Tests op mobiele apparaten



De tests die geschreven zijn voor Cucumber zijn vanaf deze sprint ook uit te voeren op de mobiele apparaten. Er is een app beschikbaar voor Android en iOS. Na het installeren van deze app kan in de configuratie van Cucumber het IP adres ingevoerd worden van het mobiele apparaat waarna de tests op dat apparaat uitgevoerd wordt. De Android app gebruikt hier het WebView component voor, iOS maakt gebruik van het UIWebView component. In deze componenten van het besturingssystemen kunnen HTML bestanden getoond worden.

6.4.4 Asynchrone tests

Omdat content dynamisch wordt ingeladen in de pagina moet er tijdens een test, die kijkt of de content in de pagina zit, wachten tot deze content ook daadwerkelijk op de pagina aanwezig is. Als er niet wordt gewacht op de content zal de testen falen omdat Selenium deze niet kan vinden. Voor deze sprint werd dat gedaan door aan te geven dat Selenium een aantal seconden moest wachten totdat er wordt gekeken of de content er is. Capybara heeft hier een techniek voor, welke al toegepast is bij het klikken op een knop. Daar wordt gezocht naar een knop waarop gedrukt moet worden, als deze er niet is wordt er opnieuw gezocht totdat de knop gevonden is of totdat de ingestelde `default_waiting_time` over is. Om dit bij het zoeken van content ook te doen wordt nu gebruik gemaakt van de code `page.should have_content('de te zoeken content')` i.p.v. `page.has_content('de te zoeken content')`.

6.4.5 WindowsPhone

De nieuwste versie van WindowsPhone, 8, is uitgekomen op 29 oktober. Er is in overleg met de opdrachtgever besloten om dat het gewenst is om een Nokia toestel aan te schaffen met dit besturingssysteem omdat Nokia en Microsoft, de maker van WindowsPhone, nauw met elkaar samenwerken. Dit toestel heeft te lang op zich laten wachten en er is daarom besloten om dit niet tijdens dit project aan te schaffen.

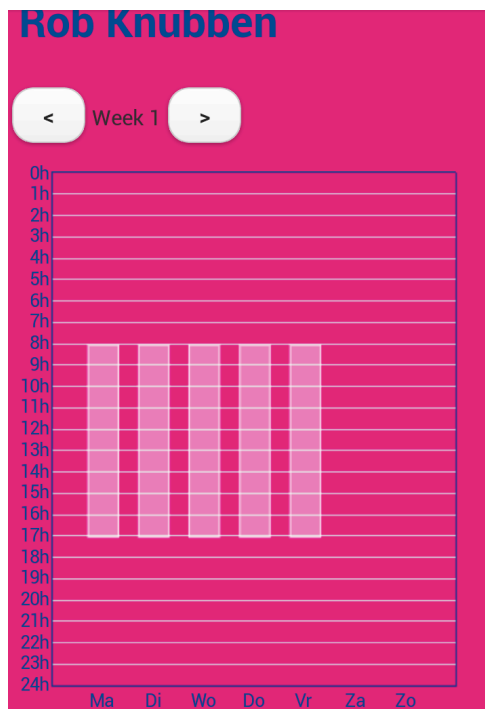
6.5 Sprint 4 – detail-working & bewerkbare persoonsdata

In deze sprint is er het nieuwe detail-working component ontwikkeld. Ook is er de eerder ontwikkelde detail-person component aangepast zodat de data die hierin weergegeven wordt bewerkbaar is, dit is gedaan via een knop die toegevoegd kan worden en de gebruiker doorstuurt naar het detail-person-edit component.

6.5.1 Detail-working component

Om dit component te bereiken is er een knop toegevoegd aan het detail-person component. De ontwikkelaar kan aangeven via een attribute of deze getoond wordt of niet.

Het component toont de naam van een werknemer en een grafiek waarin de uren die hij heeft gewerkt per week. Er kan tussen de weken genavigeerd worden door de knoppen om de week titel heen. Het HTML element waarop de grafiek getekend wordt is een canvas, dit is een HTML5 element die gebruikt kan worden voor het tekenen van 2D vormen en bitmap afbeeldingen. In de toekomst zal ook 3D ondersteund worden voor dit element. Voor het tekenen van de grafiek wordt de library flot2.0 gebruikt.

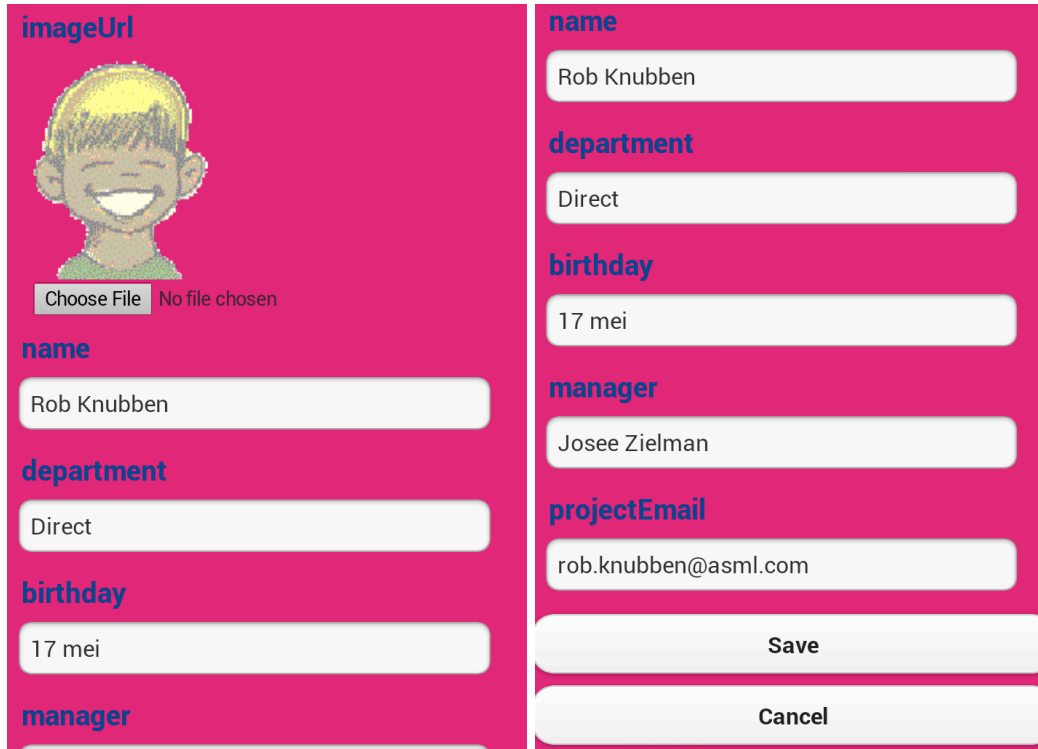


Afbeelding 6-5 Screenshot van het detail-working component. Deze is iets naar beneden gescrold zodat de grafiek in zijn geheel getoond kan worden.

6.5.2 Detail-person-edit component

Het detail-person-edit component laat de data van een persoon zien net als in het detail-person component. Alleen zijn hier de waardes bewerkbaar, de tekst kan doormiddel van textboxes aangepast worden. Er kan een nieuwe foto geüpload worden doormiddel van HTML input element met als type file, het besturingssysteem handelt dit het selecteren van een bestand af.

De wijzigingen die aangebracht zijn in de data van de persoon kunnen door de annuleer knop geannuleerd worden en door de opslaan knop opgeslagen. Er wordt dan een event getriggerd die de nieuwe persoon als data meegeeft met daarin eventuele een nieuwe URL naar een foto.



The screenshot displays two side-by-side panels of the 'detail-person-edit' component, illustrating different scroll positions. Both panels share the same data: name (Rob Knubben), department (Direct), birthday (17 mei), manager (Josee Zielman), and projectEmail (rob.knubben@asml.com). The left panel, representing a scroll to the top, includes an 'imageUrl' section with a photo of a boy and a 'Choose File' button. The right panel, representing a scroll to the bottom, shows the 'Save' and 'Cancel' buttons.

Afbeelding 6-6 Screenshot van het detail-person-edit component. Links is deze helemaal naar boven gescrold en rechts helemaal naar beneden.

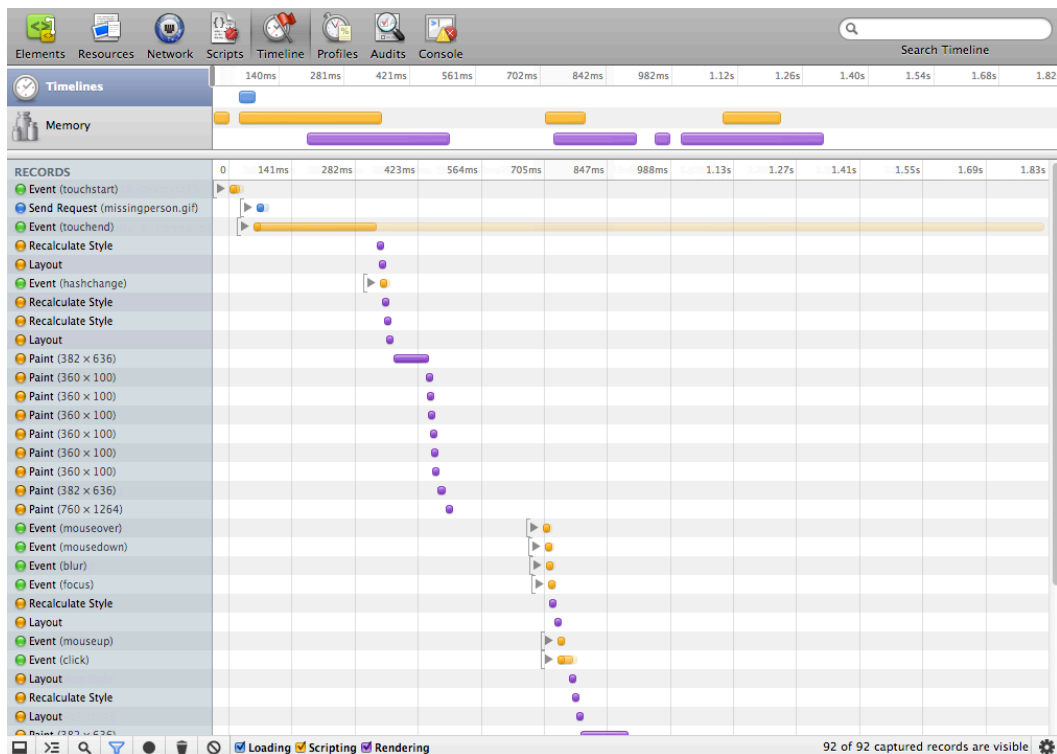
6.6 Sprint 5 – stabiele versie

Deze sprint is gedeeltelijk gebruikt om een stabiele versie te krijgen van de tot nu toe gemaakte functionaliteit, hiermee zijn kleine openstaande taken voltooid. Daarnaast is gekeken naar remote debuggen op iPhone en Android en is de swipe navigatie gerealiseerd waar in sprint 2 naar gekeken is.

6.6.1 Remote debuggen

Vanaf iOS 6, de huidige versie van het besturingssysteem op de iPhone, is het mogelijk om vanuit Safari, de browser van Apple, remote debuggen te gebruiken. Voor Android kan dit al wat langer en wordt de Android Debug Bridge, afgekort adb, gebruikt. De adb forward de TCP naar de computer waarmee het Android apparaat via USB mee is verbonden.

Het debuggen op Android en iPhone wordt in beide gevallen gedaan met Web Inspector. Dit is een tool waarmee de bron van de pagina bekeken kan worden, scripts kunnen worden uitgevoerd en een timeline ingekeken kan worden waarin te zien is wat er allemaal gebeurt en hoeveel tijd dit kost.



Afbeelding 6-7 Screenshot van de timeline. De data die getoond wordt is opgenomen van het drukken op een item in het mastercomponent t/m het laden van de detail-person component.

6.6.2 Swipe navigatie

De swipe gesture, een veeg met één vinger van links naar rechts op het scherm, om terug te gaan naar de vorige pagina is tijdens deze sprint goed geïmplementeerd. De gesture werkt alleen niet in de Chrome browser voor Android. Dit komt omdat Chrome zelf de gesture al gebruikt voor het navigeren tussen verschillende tabs in de browser. Omdat de Chrome browser alleen gebruikt wordt om te testen is dit geen probleem. Als de app gebouwd wordt met gebruik van PhoneGap zal deze in een WebView draaien en niet in de browser waardoor de swipe gesture niet meer gekaapt wordt door de browser.

6.7 Sprint 6 – video

In deze sprint is er gekeken naar het afspelen van een video. Dit wordt gedaan met de HTML5 video tag. Verder is in deze sprint een hoofdmenu gerealiseerd zodat er genavigeerd kan worden naar de lijst met werknemers en een lijst van video's.

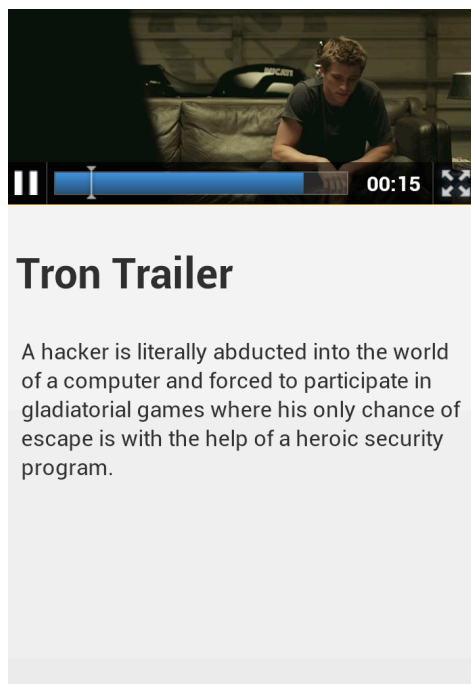
6.7.1 Meerdere x-tags

Om een overzicht te krijgen in de app van de af te spelen video's wordt het eerder gerealiseerde master component gebruikt om een lijst weer te geven van deze video's. Toen dit gedaan werd trad er een nieuw probleem op. De variabelen en functies in een component zijn niet in een object gedefinieerd waardoor deze variabelen de waardes hebben van het laatst ingevoegde master component. Dit probleem is opgelost door alle benodigde variabelen en functies in een nieuwe functie te zetten, in JavaScript kan er dan een object gemaakt worden van deze functie.

6.7.2 Videocomponent

HTML5 biedt een tag aan voor het tonen van een video. Dit is een video element waarin de bron van de video gezet kan worden die vervolgens afgespeeld kan worden. Verder kan dit element nog geconfigureerd worden door het meegeven van verschillende attributen. De attributen die gebruikt zijn zijn in het richtlijnen document beschreven onder het videocomponent.

Er is een videocomponent gemaakt waarin de ontwikkelaar een video element kan configureren een daarbij een nog een eventuele titel en omschrijving bij kan plaatsen.



Afbeelding 6-8 Screenshot van het videocomponent. Hier wordt een trailer van de film Tron afgespeeld.

6.7.3 Mapping van namen in een JSON bestand

Bij de detailcomponenten die zijn gemaakt wordt informatie weergegeven vanuit een JSON bestand. Dit wordt getoond door middel van een label met de key in een JSON bestand en eronder de waarde van deze key. Het kan vaak voorkomen dat de key niet de juiste tekst heeft om weer te geven in het component. Zo heeft bijvoorbeeld het JSON bestand dat gebruikt wordt bij het component detail-person een key die name heet. Het mappen van de keys naar nieuwe waarden voegt de functionaliteit toe dat bijvoorbeeld name als "Naam" getoond kan worden in het component. In onderstaande voorbeeld wordt de mapping attribute, genaamd data-name-mapping gebruikt.

```
<x-detail-person data-data="/json/p.json"
  data-maininfo-names="name,department"
  data-name-mapping="name=Naam,department=Afdeling">
</x-detail-person
```

6.7.4 Functies meegeven in attributen

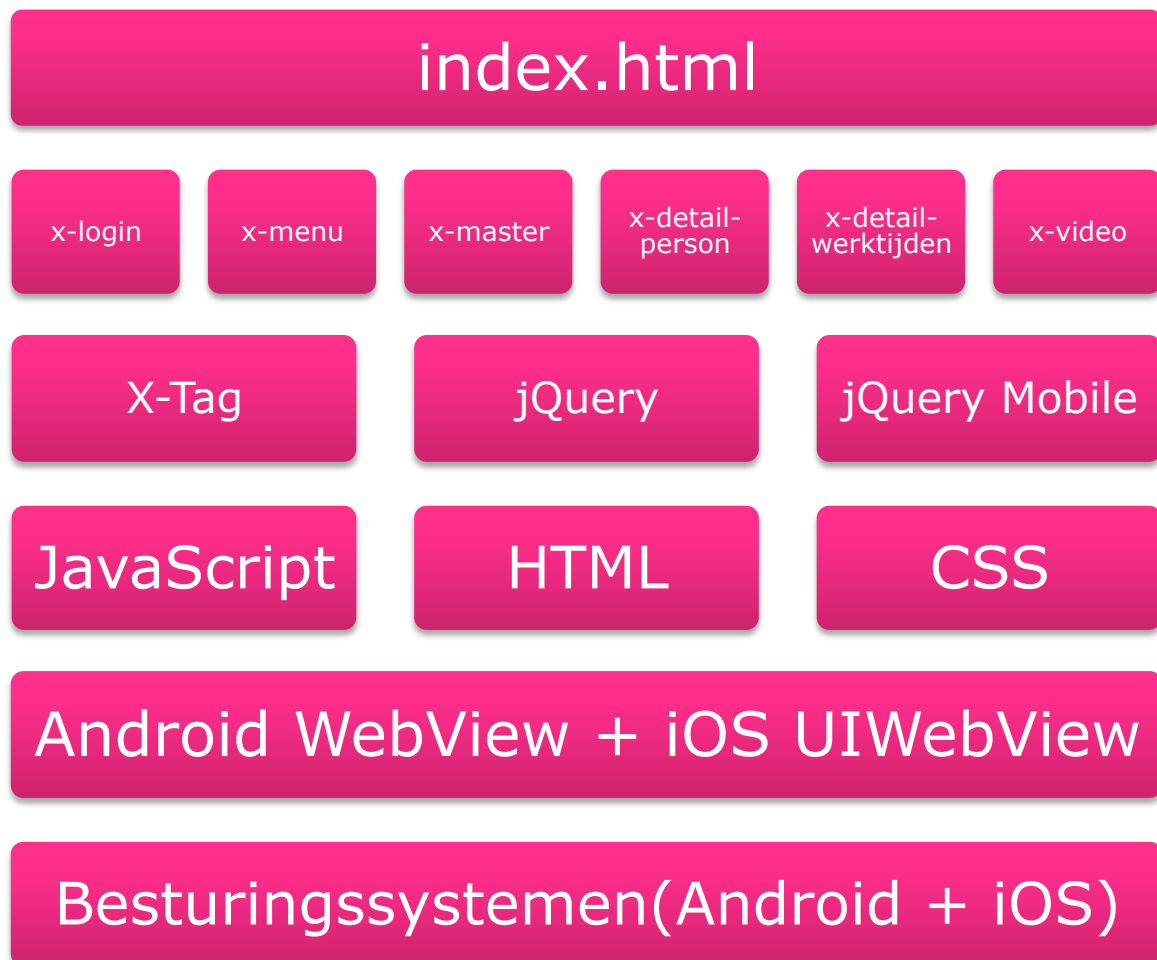
Om listeners aan events te kunnen zetten wordt gebruik gemaakt van de functie `addEventListener`. Het is meer gewenst om dit te doen in een attribute bij het definiëren van een component. Dit wordt ook al gedaan bij bijvoorbeeld knoppen in HTML. Een knop heeft dan een `onclick` attribute waarbij de functienaam die de ontwikkelaar als listener gebruikt mee kan geven. Dit is nu ook gerealiseerd voor alle gemaakte componenten. Zo kan er nu bijvoorbeeld in het mastercomponent de functienaam voor de listener op het event `onItemClicked` meegegeven worden in de attribute `data-onItemClicked`.

6.8 Sprint 7 – de afronding

De laatste sprint van dit project is gebruikt om alle documentatie op de wiki in orde te krijgen. Het document met richtlijnen is op de wiki gezet zodat deze makkelijk toegankelijk en bewerkbaar is. Er is een tutorial geschreven voor het gebruiken van een component vanuit een leeg project. Ook is voor deze tutorial een video gemaakt zodat er doormiddel van beeld en geluid een duidelijker beeld is.

Daarnaast zijn een aantal componenten dynamischer gemaakt door extra attributen toe te voegen en zijn alle eventlisteners, zoals in een vorige sprint, voor alle componenten nu toe te voegen via een attribute.

Onderstaand figuur beschrijft het gemaakte product.



7 Resultaten van het onderzoek

In dit hoofdstuk zijn de deelvragen beantwoord. De hoofdvraag wordt beantwoord in het volgende hoofdstuk.

7.1 Wat zijn HTML5, CSS3 en JavaScript?



Het W3C, World Wide Web Consortium, is het instituut die gaat over de HTML en CSS standaarden. Op de website van W3C wordt de volgende uitleg gegeven over HTML:

This specification defines the 5th major revision of the core language of the World Wide Web: the Hypertext Markup Language (HTML). In this version, new features are introduced to help Web application authors, new elements are introduced based on research into prevailing authoring practices, and special attention has been given to defining clear conformance criteria for user agents in an effort to improve interoperability.

HTML5 is dus simpelweg de volgende versie van HTML, waar nieuwe functionaliteit aan toegevoegd is.

HTML is al jarenlang het meest gebruikte middel om de opmaak van documenten mee te regelen. Met de nieuwe versie van HTML wordt meer gekeken naar wat HTML meer zou kunnen betekenen voor web applicaties i.p.v. alleen maar documenten.

CSS3 is de nieuwe versie van CSS dat staat voor Cascading Style Sheet en is een techniek om een stijl, bijvoorbeeld lettertypes en kleuren, toe te voegen aan een document.

JavaScript is een programmeertaal die gebruikt wordt om websites dynamisch te kunnen maken. Het is mogelijk om object georiënteerd te programmeren in JavaScript. Dit wordt op basis van prototypes gedaan i.p.v. classes. Dit verschil is belangrijk omdat veel ontwikkelaars voor iOS en Android gewend zijn te programmeren in object georiënteerde talen met classes. Het is belangrijk dat de ontwikkelaar van dit verschil afweet.

De combinatie van HTML5, CSS3 en JavaScript in de vorm van een app wordt vaak een HTML5 app genoemd. HTML, CSS en JavaScript zijn zulke belangrijke standaarden omdat deze gebruikt worden door browsers, die door ieder populair besturingssysteem gebruikt worden.

7.2 Wat voor apps worden er ontwikkeld door TASS?

Aan het begin van het project van het project is er een met de opdrachtgever om de tafel gezeten om antwoord te krijgen op deze vraag. Er is toen een diagram gemaakt die te zien is in hoofdstuk 6.1 (sprint 0). Uit dit schema kan geconcludeerd worden dat TASS apps maakt waar data in te zien is en te bewerken. Ook is TASS altijd bezig om te kijken wat nieuwe technologieën kunnen betekenen voor app ontwikkeling. Om mobiele apparaten zijn dit bijvoorbeeld functies zoals:

- NFC, Near Field Communication. Hiermee kunnen smartphone en andere apparaten die NFC ondersteuning hebben vanaf een erg kleine afstand, niet meer dan een paar centimeter, een kleine hoeveelheid data uitwisselen.
- Wi-Fi Direct. Communicaties tussen smartphones via Wi-Fi.
- Locatie. Deze kan bepaald worden door GPS en Wi-Fi.
- Accelerometer. Zorgt ervoor dat de bewegingen van de smartphone in alle drie de richtingen (X, Y en Z as) gemeten kan worden.
- Camera. Voor het maken van foto's en video's. Maar zou ook voor geavanceerdere doeleinden kunnen worden gebruikt zoals gezichtsherkenning.

Veel van deze functies kunnen al via HTML5 benaderd worden. Als dit (nog) niet kan biedt PhoneGap altijd de mogelijkheid om een native plug-in hiervoor te gebruiken.

Tijdens het gesprek met de opdrachtgever is er ook aangegeven dat TASS gebruikt maakt van apps voor het aansturen van apparaten. Hier wordt dan vaak data verstuurd naar een ander apparaat als de gebruiker een actie onderneemt.

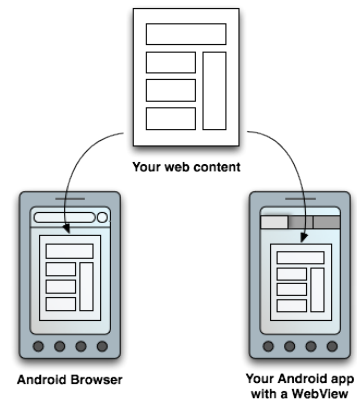
Op de planning staat een app waar de medewerkers van TASS hun uren in kunnen bijhouden. Ook zijn er plannen om de SharePoint omgeving naar de mobiele platformen te halen. Met het gebruik van NFC wil TASS in de toekomst een beveiligingssysteem ontwikkelen waarmee je het gebouw binnenkomt.

7.3 Welke besturingssystemen ondersteunen HTML5?

Op de officiële website voor ontwikkelaars van Android staat een artikel over web applicaties. Hierin wordt verteld dat er twee manier zijn om de content van de web applicatie aan de gebruiker op te dienen.

1. Via de Android browser.
2. Door een WebView in een Android app te implementeren.

De WebView die gebruikt wordt is dezelfde als die de standaard browser van Android gebruikt. Dit is niet Chrome. Chrome ondersteunt meer HTML5 functionaliteiten dan de standaard browser.



Tijdens het Google evenement: Google I/O 2012 is er aangekondigd dat Chrome binnenkort gebruikt zal gaan worden voor de WebView in Android. Wanneer dit dus wordt geïmplementeerd worden er meer HTML5 functies ondersteund.

De officiële website van Apple gaat niet specifiek in op het ontwikkelen van apps in HTML5. Wel wordt gezegd dat de standaarden ondersteunt worden in de webbrowser Safari:

Every new Apple mobile device and every new Mac — along with the latest version of Apple’s Safari web browser — supports web standards including HTML5, CSS3, and JavaScript.

In iOS wordt het UI component UIWebView gebruikt voor het tonen van HTML bestanden, deze is gelijk aan de Safari browser.

Op de website html5test.com staat een overzicht van iedere browser die zij getest hebben op de HTML5 functionaliteiten. Voor iedere functionaliteit kan een browser punten scoren, maximaal 500 punten.

Android scoort een score van 297 met de standaard browser, maar met Chrome wordt een hogere score gehaald, 390. iOS haalt een score van 386.

7.4 Zijn er al apps die geschreven zijn in HTML5 en hoe worden deze ervaren?

Op de website van PhoneGap zijn apps genoemd die geschreven zijn met gebruik van HTML5. Er is een analyse gedaan op vijf van deze apps, de zogenaamde featured apps van PhoneGap. Hieronder staan de apps opgesomd met de resultaten:

1. The Panasonic World Heritage Calendar

De app heeft vrij weinig functionaliteit, er worden foto's weergegeven en er kan tekst gelezen worden die bij de foto hoort. De app is gemaakt door het bedrijf BluSoft Inc's, zij hebben laten weten aan PhoneGap dat zij niet alleen in een recordtijd de app af wisten te krijgen maar dat de app ook veel makkelijker te onderhouden was:

"Not only could we go from prototyping to release in record time but maintenance time for future releases was also reduced because of the shared codebase."

In Japan heeft deze app tweede gestaan in de App Store van Apple in de lijst van 'Top Free Apps'.

In de Android Play Store is de app minder populair met maar 100 tot 500 downloads.

2. Completure

Deze app toont kleine verhaaltjes die de gebruikers in elkaar zetten en gebruikt hierbij de functionaliteit van de camera en locatie.

De app is alleen maar in de App Store van Apple beschikbaar en heeft de een uitstekende beoordeling, 5 sterren.

3. HealthTap

In deze app kunnen gebruikers dokters in de buurt om advies vragen bij medische problemen.

De app staat in de Android Play Store en de App Store van Apple. In beide gevallen scoort de app erg hoog met 4 en een halve ster.

4. Untappd

Deze app laat gebruikers berichten en foto's delen via zijn eigen sociale netwerk maar ook via Twitter, Facebook en Foursquare.

De app staat in de App Store en wordt beoordeeld met vijf sterren, in de Play Store wordt de app iets minder, maar nog steeds goed, beoordeeld met 4,1 ster.

5. Wikipedia

PhoneGap is gebruikt om deze app te ontwikkelen omdat Wikipedia overal aanwezig moet zijn volgens de director of mobile and special projects van het bedrijf achter Wikipedia, Wikimedia. De app heeft meer dan 5 miljoen actieve Android gebruikers.

De app wordt goed beoordeelt in de Play Store met 4 en een halve ster. De iPhone gebruikers zijn er minder enthousiast over en geven de app maar 2 sterren.

Bovenstaande apps worden redelijk goed beoordeeld maar er kan ook besloten worden om HTML5 niet gebruiken. Facebook had zijn app ontwikkeld in HTML5 maar is hier op terug te komen omdat ze toch een betere gebruikerservaring willen bieden aan hun gebruikers. Zuckerberg, de baas van Facebook, zegt over HTML5:

"Het platform was er nog niet klaar voor."

"We moeten het hoogste kwaliteitsniveau bereiken en dat kan alleen maar als we native gaan."

7.5 Kan HTML5 dezelfde prestatie leveren als native?

Tijdens het realiseren van functionaliteit in HTML5 voor dit project is geconstateerd dat HTML5 vaak niet dezelfde performance kan bieden zoals native dit wel kan. Dit is tijdens acceptatietesten bij iedere sprint geconstateerd. Er is wel de conclusie getrokken dat, al zou TASS liever dezelfde performance willen als bij native, dat dit vaak niet erg hinderlijk is.

Zoals in het antwoord op de vorige deelvraag ook gezegd wordt heeft Facebook gekozen voor het ontwikkelen van zijn app in native om een betere performance te halen.

De platformen zijn nog niet klaar om performance te bieden zoals native dit doet. Steeds meer functionaliteiten van HTML5 worden geïmplementeerd in de browsers. Het is een kwestie van tijd totdat HTML5 vergelijkbare performance gaat bieden.

8 Conclusies en aanbevelingen

Met dit project is een goede basis opgezet voor het maken van componenten die hergebruikt kunnen worden door de ontwikkelaars van TASS. De componenten moeten alleen nog door een designer de juiste opmaak krijgen voordat deze daadwerkelijk hergebruikt kunnen worden.

HTML5 is een erg groot onderwerp. In de project is niet naar alle functionaliteit van HTML5 onder de aandacht gekomen.

De vraag naar mobiele applicatie is de laatste jaren erg gegroeid. Android en iOS hebben duidelijk het grootste deel van de markt in handen maar er zijn steeds meer bedrijven die een gooi willen doen naar de mobiele markt. Zo komt Mozilla binnenkort met FireFox OS, Ubuntu met een mobiel besturingssysteem, wordt er gewerkt aan het mobiele besturingssysteem Tizen met de steun van The Linux Foundation die erg sturen naar HTML5 applicaties en natuurlijk is daar nog Windows Phone die steeds meer marktaandeel in handen krijgt. Om de mobiele applicaties te laten draaien op al deze besturingssystemen en rekening te houden met de toekomst is HTML5 vaak een goede keuze. HTML5 wordt steeds groter en beter ondersteund door de besturingssystemen, het belooft veel te worden. Het is dus belangrijk deze ontwikkelingen goed in de gaten te blijven houden.

Voor veel applicaties die TASS ontwikkelt kan HTML5 een goede keuze zijn omdat deze nog niet per se een performance moeten hebben zoals native deze heeft. HTML5 is dus een goede keuze omdat er erg veel tijd bespaard wordt door het maar één keer te hoeven ontwikkelen. Het blijft nog belangrijk om voor de start van het ontwikkelen van een nieuwe app te kijken welke functionaliteit deze app moet maken en of dit problemen zou kunnen geven als dit in HTML5 zou worden gedaan. Als de performance belangrijk is kunnen er native plug-ins ontwikkeld worden zodat er een hybrid app ontstaat.

Tijdens dit project is gewerkt met Scrum en Behaviour Driven Development. De tools die gebruikt zijn voor Behaviour Driven Development werken zeer goed en gemakkelijk. Binnen TASS zou het goed zijn dit, waar mogelijk, toe te passen om de kwaliteit van projecten beter te kunnen garanderen. De tool die gebruikt wordt voor Scrum, Scrumwise, mist functionaliteit om optimaal te kunnen werken met Scrum. Er is daarom ook een aanbeveling gedaan om te kijken naar een goede tool voor Scrum.

Evaluatie

Eén van de belangrijkste redenen dat ik bij TASS ben gaan zoeken naar een afstudeerstage is dat ik een kijkje wilde nemen in een wat groter software bedrijf, zodat ik zou weten hoe het daar aan toe gaat. Voorafgaand aan mijn afstudeerstage werkte ik bij een klein softwarebedrijf. Het verschil tussen deze twee bedrijven zit, naast het aantal medewerkers, in de projectaanpak. Bij TASS merk je dat er meer ervaring is en dat de projecten ook met een softwareontwikkelmethode aangepakt worden. Dit in tegenstelling tot het andere bedrijf waar de mentaliteit meer weg heeft van: "We gaan het gewoon maken."

In de tijd dat ik bij TASS aan mijn project gewerkt heb, heb ik veel kennis opgedaan van de ontwikkelmethode Scrum. Deze ontwikkelmethode heb ik één keer tijdens mijn opleiding gebruikt, nu kom ik tot de conclusie dat dat meer weg had van waterval op een scrumboard. Naast Scrum heb ik ook veel kennis opgedaan van Behaviour Driven Development en de tool Cucumber die daarbij gebruikt werd.

De technieken en talen die ik gebruikt heb waren allemaal nieuw voor mij. Van HTML5, CSS3 en JavaScript had ik erg weinig kennis toen ik begon met deze afstudeerstage. De programmeertaal Ruby had ik nog nooit gezien, ik wist alleen dat het bestond. JavaScript ben ik geweldig gaan vinden, in mijn eigen tijd ben ik gaan kijken naar node.js waar JavaScript op de server gedraaid kan worden.

De gebruikte tools Cucumber, PhoneGap, jQuery, jQuery Mobile en Brackets waren allemaal nieuw voor mij. Het heeft wel tijd gekost om gewend te raken aan deze tools, maar het zijn allemaal goede leermomenten geweest en ik bedank mijn begeleiders bij TASS mede daarom ook dat zij mij kennis hebben laten maken met deze tools.

Er hing een gezellige en open sfeer bij TASS, iedereen deelde zijn kennis en niemand was bang een ander hulp te vragen of deze aan te bieden. Zo kon ik ook een programmeur/systeemanalist te hulp schieten met het opstellen van zijn Cucumber tests.

Over mijn tijd bij TASS ben ik erg tevreden, niet alleen omdat ik zoveel geleerd heb maar ook omdat er nu een goede basis ligt aan kennis waaraan verder gewerkt kan worden. Hierdoor heb ik iets van waarde achtergelaten bij het bedrijf.

Het was een leerzame en leuke tijd.

9 Literatuurlijst

Wynne, M., & Hellesøy, A. (2012). *The Cucumber Book: Behaviour-Driven Development for Testers and Developers*. The Pragmatic Bookshelf.

W3C. (sd). Opgehaald van www.w3.org

Web Apps. (sd). Opgehaald van Android Developers: <http://developer.android.com/>

What is Scrum? (sd). Opgehaald van Scrum.org: www.scrum.org

X-Tags. (sd). Opgehaald van www.x-tags.org

Verschuren, P., & Doorewaard, J. (2007). *Het ontwerpen van een onderzoek*. Boom Lemma uitgevers.

Brackets. (sd). Opgehaald van www.brackets.io

Capbara. (sd). Opgehaald van Rubydoc: <http://rubydoc.info/>

Chrome Experiments. (sd). Opgehaald van www.chromeexperiments.com/

Google. (sd). *Creating Fast Buttons for Mobile Web Applications*. Opgehaald van www.code.google.com/mobile/articles/fast_buttons.html

Hakim's Projects. (sd). Opgehaald van scroll-effects: www.lab.hakim.se

Haring, R. (2011, februari). Behaviour Driven development. *Java Magazine* .

HTML5. (sd). Opgehaald van Apple: www.apple.com

HTML5 hoeft geen misstap te zijn. (sd). Opgehaald van <http://computerworld.nl/article/14001/html5-hoeft-geen-misstap-te-zijn.html>

Kempen, P., & Keizer, J. (2011). *Competent afstuderen en stagelopen*. Noordhoff Uitgevers bv.

Microsoft. (sd). *Building Apps with HTML5: What You Need to Know*. Opgehaald van <http://msdn.microsoft.com/en-us/magazine/hh335062.aspx>

PhoneGap. (sd). Opgehaald van PhoneGap: www.phonegap.com

Stackoverflow. (sd). Opgehaald van Stackoverflow: www.stackoverflow.com

Bijlagen

Bijlage A: PID

Bijlage B: Onderzoeksontwerp

Bijlage C: Sprint notulen & Sprint Retrospectives

Bijlage D: Document van bewijzen

Bijlage E: Richtlijnen document

Bijlage F: Onderzoeksverslag

Bijlage A

Project Initiation Document

PROJECT INITIATIE DOCUMENT



Datum voltooid: 25-09-2012

Auteur: Joep van der Heijden

Versie: 1.0.0

Status: Concept

1 Documenthistorie

1.1 Revisies

Versie	Status	Datum	Auteur:	Wijzigingen
0.0.1	concept	21-09-2012	Joep van der Heijden	Eerste concept
0.0.2	Concept	24-09-2-12	Joep van der Heijden	Volledige PID

1.2 Goedkeuring

Dit document behoeft de volgende goedkeuringen:

Versie	Datum goedkeuring	Naam	Functie	Paraaf
1.0.0	25-09-2012	Frits van de Water	Bedrijfsbegeleider	

1.3 Distributie

Dit document is verstuurd aan:

Versie	Datum verzending	Naam	Functie
0.0.1	21-09-2012	Frits van de Water	Bedrijfsbegeleider
0.0.2	24-09-2012	Frits van de Water	Bedrijfsbegeleider

2 Managementsamenvatting

Deze PID hoort bij het project "Smartphones, develop once run anywhere." Dit project wordt uitgevoerd binnen TASS als afstudeerproject.

2.1 Doel van dit document

Dit document heeft tot doel het project te definiëren, als basis te dienen voor het management ervan en de beoordeling van het succes van het project mogelijk te maken.

De twee belangrijkste redenen voor gebruik van dit document zijn:

- om er zeker van te zijn dat het project een gezonde basis heeft voordat de Stuurgroep gevraagd wordt zich aan het project te committeren;
- om te dienen als basisdocument op grond waarvan de Stuurgroep en de Projectmanager de voortgang en wijzigingen kunnen toetsen en bewaken en vragen omtrent geldigheid van het project tijdens de uitvoering ervan kunnen beoordelen.

2.2 Aanleiding

Sinds de komst van de smartphone is TASS zich bezig gaan houden met het ontwikkelen van apps voor alle verschillende besturingsystemen. Dit wordt gedaan door native apps te ontwikkelen, met native app wordt bedoeld dat iedere app voor een besturingsstelsel apart geschreven wordt.

Het ontwikkelen van native apps kost veel tijd omdat alles voor ieder besturingsstelsel opnieuw geschreven moet worden. HTML5 kan een oplossing bieden voor dit probleem, de vraag is in hoeverre dit kan. Door het ontwikkelen van componenten met technieken die bekend zijn op ieder besturingsstelsel, zoals HTML5, Javascript en CSS3, kan veel tijd worden bespaard.

2.3 Globale aanpak

Dit project maakt gebruik van het Tien Stappen Plan van Kempen en Keizer. Het onderzoek dat zal worden uitgevoerd is gebaseerd op een methode van Verschuren en Doorewaard.

Op 3 september 2012 is dit project gestart en zal tussen de 85 en 100 dagen duren. Op 1 februari 2013 zal dit project officieel beëindigd worden.

De belangrijkste vraag die tijdens dit project beantwoord zal moeten worden is:

"In hoeverre kan HTML5 de native apps, die ontwikkeld worden door TASS, vervangen?"

Naast dit document zullen de volgende producten worden gerealiseerd:

- Het onderzoeksverslag
- Een document met richtlijnen voor het ontwikkelen van apps binnen TASS
- Het document van bewijzen
- De afstudeerscriptie

3 Inhoudsopgave

1 INLEIDING	5
2 ACHTERGROND	6
3 PROJECTDEFINITIE	7
3.1 PROJECTDOELSTELLINGEN	7
3.2 GEKOZEN AANPAK	7
3.3 SCOPE VAN HET PROJECT	8
3.4 PROJECTBUDGET	8
3.5 BENODIGDE RESOURCES	8
3.6 UITSLUITINGEN	9
3.7 AFHANKELIJKHEDEN	9
3.8 RANDVOORWAARDEN	9
3.9 AANNAMES	9
4 PROJECTORGANISATIESTRUCTUUR	10
4.1 PROJECTROLLEN	10
5 PROJECTBEHEERSING	11
5.1 VOORTGANGSBEWAKING	11
5.2 TOLERANTIES	11
<i>Projecttolerantie</i>	<i>11</i>
<i>Fasetoleranties</i>	<i>11</i>
5.3 RISICOMANAGEMENT	11
5.4 KWALITEITSBEWAKING	11
5.5 ESCALATIEPROCEDURE	11
BIJLAGE A: COMMUNICATIEPLAN	12
BIJLAGE B: INITIËLE PROJECTPLANNING	14
BIJLAGE C: INITIEEL RISICOLOGBOEK	17
BIJLAGE D: KWALITEITSPLAN	18

1 Inleiding

Doel van dit document

Dit document is opgesteld om alle relevante basisinformatie en uitgangspunten van het project vast te leggen om het op de juiste wijze te kunnen besturen. Het heeft tot doel het project te definiëren, als basis te dienen voor het management ervan en de beoordeling van het succes van het project mogelijk te maken.

Dit Projectinitiatiedocument (of PID) behandelt de volgende fundamentele aspecten van het project:

- Wat beoogt men met het project te bereiken?
- Waarom is het belangrijk om deze doelstellingen te bereiken?
- Wie zijn er betrokken bij het managen van het project en wat zijn hun rollen en verantwoordelijkheden?
- Hoe en wanneer zullen de maatregelen die in dit PID besproken worden gerealiseerd worden?

Het document wordt gebruikt:

- om er zeker van te zijn dat het project een gezonde basis heeft voordat de Stuurgroep gevraagd wordt zich aan het project te committeren;
- om te dienen als basisdocument op grond waarvan de Stuurgroep en de Projectmanager de voortgang en wijzigingen kunnen toetsen en bewaken en vragen omtrent geldigheid van het project tijdens de uitvoering ervan kunnen beoordelen.

2 Achtergrond

De smartphone is de laatste jaren een apparaat geworden dat bijna iedereen gebruikt. Al sinds het begin van de smartphone ontwikkelt TASS apps voor de smartphone. Dit doet TASS door het ontwikkelen van zogenaamde native apps. Dit houdt in dat een app speciaal voor een besturingssysteem ontwikkeld wordt. Als deze app dan ook op een ander besturingssysteem moet kunnen draaien betekent dit dat de app opnieuw geschreven moet worden voor het andere besturingssysteem. Door deze manier van ontwikkelen gaat er veel tijd zitten in het programmeren van code die er al is in andere programmeertalen.

Om te kunnen realiseren dat apps draaien op ieder besturingssysteem en maar een keer geschreven hoeven te worden kan er gebruik gemaakt worden van technieken die op ieder besturingssysteem bekend zijn. In dit project zal worden gekeken of de technieken HTML, Javascript en CSS3 hier een goede oplossing voor kunnen bieden.

3 Projectdefinitie

3.1 Projectdoelstellingen

Dit project heeft als doelstelling om de time to market van apps, ontwikkeld door TASS, te verkorten. Hierdoor worden er minder uren besteed door medewerkers van TASS aan onderdelen die al eens gedaan zijn.

Er moet een document komen met richtlijnen die beschrijven hoe een app in HTML5 gerealiseerd moet gaan worden, maar ook hoe een app eruit moet komen te zien en hoe deze zich moet gedragen.

3.2 Gekozen aanpak

Bij het afstuderen wordt er gebruik gemaakt van de methode competent afstuderen en stagelopen van Kempen en Keizer. In deze methode wordt het afstudeerproject opgedeeld in tien stappen en drie fases. Voor deze methode is gekozen omdat er bij de aanvang slechts een vaag beeld is van wat het werk inhoudt en er geen idee is of de afgesproken opdracht in de beschikbare afstudeertijd past. De student mist de ervaring en het overzicht om dat goed te kunnen beoordelen.

In de onderstaande tabel wordt beschreven welke fases er worden uitgevoerd om de projectdoelstellingen te kunnen realiseren. Voor een gedetailleerde planning zie bijlage B.

Fase	Omschrijving	Planning
Oriëntatiefase	Oriëntatie op de afstudeeropdracht en een definitieve versie van het PID. Na het opstellen van het PID en goedkeuring door de opdrachtgever wordt er overgegaan naar de volgende fase.	03-09-2012 t/m 28-09-2012
Onderzoeks- en oplossingsfase	Aan het begin van deze fase wordt een onderzoeksontwerp gemaakt. Na goedkeuring door de opdrachtgever wordt het onderzoek uitgevoerd. Tijdens het uitvoeren van het onderzoek worden de bevindingen gedocumenteerd in het onderzoeksverslag. Na het onderzoek zal het document met richtlijnen worden opgesteld.	01-10-2012 t/m 21-12-2012
Invoeringsfase	De gerealiseerde documenten worden overgedragen aan de opdrachtgever. Toepassen van de richtlijnen op een app van TASS. Er wordt een presentatie gegeven aan de opdrachtgever. De scriptie wordt afgerond.	21-12-2012 t/m 01-02-2013

Tijdens de realisatie wordt de ontwikkelmethode SCRUM toegepast. Er is voor deze methode gekozen zodat er snel en gemakkelijk bijgestuurd kan worden en omdat er binnen TASS veel support is voor deze methode.

3.3 Scope van het project

Tijdens dit project wordt een document gerealiseerd met richtlijnen voor het ontwikkelen van apps binnen TASS in HTML5. De focus hierbij ligt bij apps die ontwikkeld zijn of in de nabije toekomst ontwikkeld worden door TASS.

3.4 Projectbudget

Aan dit project wordt fulltime gewerkt door de afstudeerder, dit zijn 85 tot 100 werkdagen. Iedere week worden de vorderingen besproken met de begeleider(s), zij geven de afstudeerder ook de nodige ondersteuning.

Er wordt een inventarisatie gedaan naar de apparaten die beschikbaar zijn bij TASS. Er is een standaard budget van 500 euro voor een afstudeerproject bij TASS. Mocht er toch nog iets aangeschaft moeten worden, wat niet binnen dit budget valt, wordt er een afweging gedaan of TASS hier profijt van zou hebben.

3.5 Benodigde Resources

De volgende hardware en software is nodig om dit project te kunnen uitvoeren.

Naam:	Aantal:	Toelichting:
PC	1	Een computer voor het realiseren van het project
Android smartphone	1	Testen van gemaakt product op Android
iOS smartphone	1	Testen van gemaakt product op iOS
WindowsPhone smartphone	1	Testen van gemaakt product op WindowsPhone
Brackets	1	Ontwikkeltool voor het bewerken van HTML, Javascript en CSS.
Cucumber	1	Testtool. Met deze tool wordt de gemaakte Javascript getest.
Scrumwise	1	Tool voor het beheren van SCRUM
JSunit	1	Test tool voor het maken van unittests voor Javascript
Trac	1	Ticketsysteem
Capybara	1	Test tool voor het testen van HTML
SVN	1	Versiebeheer
MS office of gelijkwaardig pakket	1	Gebruikt voor documentatie
Internetverbinding	1	Voor het vinden van informatie

3.6 Uitsluitingen

Het aanpassen van web applicaties en alle apps van TASS behoort niet tot het projectresultaat. Wel gaat het richtlijnen document toegepast worden op een app van TASS, dit zal gebeuren in de invoeringsfase.

3.7 Afhankelijkheden

Ontwikkelaars van apps binnen TASS moeten geïnterviewd worden over hun werkzaamheden.

Bij het opstellen van de richtlijnen moet rekening gehouden worden met de verplichtingen die zijn opgesteld door de makers van de besturingssystemen.

3.8 Randvoorwaarden

De benodigde resources die zijn beschreven zijn nodig om het project tot een goed einde te kunnen brengen.

3.9 Aannames

Er is een aanname gemaakt dat HTML5 in ieder geval een deel van de native ontwikkeling zou kunnen overnemen. Hoeveel dit is is nog niet bekend.

Verder zijn er geen aannames gemaakt.

4 Projectorganisatiestructuur

4.1 Projectrollen

Rol:	Naam:	Omschrijving:
Opdrachtgever	Dhr. F. van de Water	De eigenaar van de business case. Controleert de vooruitgang van het project.
Docent Fontys	Dhr. M. Franssen	Biedt ondersteuning aan de project manager mocht dit nodig zijn. Controleert de vooruitgang van het project.
Technisch begeleider	Dhr. J. Hu	Geeft advies aan de project manager, biedt ondersteuning aan de project manager.
Bedrijfsbegeleider	Dhr. F. van de Water	Geeft advies aan de project manager, biedt ondersteuning aan de project manager.
Project manager	Dhr. J. van der Heijden	De afstudeerder. Leidt het project.

De communicatie en contactgegevens van deze rollen worden beschreven in het communicatieplan, deze is te vinden in bijlage A.

5 Projectbeheersing

5.1 Voortgangsbewaking

Wekelijks zullen er voortgangsgesprekken zijn tussen de projectmanager en de opdrachtgever. In deze gesprekken wordt besproken hoe het project loopt en worden eventuele problemen en onduidelijkheden besproken.

Er wordt door de projectmanager een logboek bijgehouden zoals beschreven in het tien stappen plan. Dit document wordt doorgestuurd naar de bedrijfsbegeleider en de docent van de Fontys. De docent geeft hier eventueel feedback op.

De planning voor het project wordt beschreven in bijlage B.

5.2 Toleranties

Projecttolerantie

Afwijkingen in de planning zijn toegestaan als deze de deadline niet beïnvloeden. Als er een dreiging is voor een deadline zal dit besproken worden met de opdrachtgever. Uit dit gesprek zal een nieuwe plan komen om de deadline alsnog te kunnen halen of een nieuwe deadline.

Fasetoleranties

De onderstaande percentages zijn berekend met de gegevens van paragraaf 3.2.

Fase	Budget:	Doorlooptijd:
Oriëntatiefase	18%	Week
Onderzoeks- en ontwikkelfase	55%	Week
Invoeringsfase	27%	Week

5.3 Risicomanagement

In bijlage C is het risicologboek te vinden. Hierin wordt beschreven waar rekening mee gehouden moet worden en wat de ernst ervan is.

5.4 Kwaliteitsbewaking

In bijlage D wordt beschreven hoe de kwaliteit bewaakt wordt over dit project. Hierin wordt beschreven hoe de kwaliteit gegarandeerd wordt.

5.5 Escalatieprocedure

In het risicologboek (bijlage C) staan de risico's die zich voor kunnen doen. Mocht er iets misgaan waardoor deadline in gevaar komen wordt dit besproken met de opdrachtgever en wordt er een passende oplossing gevonden.

4 Bijlage A: Communicatieplan

Inleiding

Dit communicatieplan benoemt alle partijen die een (positief of negatief) belang hebben bij het project en de wijze waarop zij bij het project zullen worden betrokken en welke communicatievormen daarbij gebruikt worden. Het gaat hierbij om partijen en communicatie buiten de formele projectmanagementstructuur zoals beschreven in het PID.

A.1 Contactgegevens

Projectmanager

Joep van der Heijden

Telnr.: +31 623681072

E-mail: 212354@student.fontys.nl

joep.van.der.heijden@tass.nl

Docent Fontys

Michael Franssen

Telnr.: 0885089313

E-mail: michael.franssen@fontys.nl

Bedrijfsbegeleider

Frits van de Water

Telnr.: +31 623526736

E-mail: Frits.van.de.water@tass.nl

Technisch begeleider

Jin Hu

E-mail: jin.hu@tass.nl

A.2 Communicatiekanalen

Van:	Naar:	Informatie:	Medium:	Frequentie:
Projectmanager	Docent Fontys	Voortgang en planning	E-mail/Mondeling	Wekelijks
Projectmanager	Opdrachtgever	Voortgang en planning	E-mail/Mondeling	Wekelijks
Docent Fontys	Projectmanager	Terugkoppeling voortgang en planning	E-mail/Mondeling	Wekelijks
Opdrachtgever	Projectmanager	Terugkoppeling voortgang en planning	E-mail/mondeling	Wekelijks
Docent Fontys	Bedrijfsbegeleider	Controle project.	Mondeling	Begin en einde project

A.3 Incidenteel

Van:	Naar:	Informatie:	Medium:	Situatie:
Projectmanager	Bedrijfsbegeleider	Reden en verwachte duur	E-mail/ Mondeling/Telefoon	Absentie
Projectmanager	Opdrachtgever en Docent Fontys	Voortgang en planning	E-mail/Mondeling	Afwijking planning
Projectmanager	Opdrachtgever en Technisch begeleider	Aanvraag overleg	E-mail/Mondeling	Onvoorziene omstandigheden
Opdrachtgever	Projectmanager	Feedback op functioneren	Mondeling	Project loopt uit de hand

5 Bijlage B: Initiële Projectplanning

B.1 Projectfasering

Oriëntatiefase

In de oriëntatiefase van dit project wordt een definitieve versie van het PID opgesteld. Deze wordt goedgekeurd door de opdrachtgever. Ook wordt er een PMP opgesteld. Deze wordt goedgekeurd door de opdrachtgever.

Activiteiten:		Op te leveren producten:
Kennismaking bedrijf		Project Initiatie Document (PID)
Opdracht verduidelijken		Project Management Plan (PMP)
Opstellen PID		
Opstellen PMP		
Uitvoeren vooronderzoek naar HTML5		
Bijhouden afstudeerscriptie		

Onderzoeks- en oplossingsfase

Aan het begin van deze fase wordt een onderzoeksontwerp gemaakt. Na goedkeuring door de opdrachtgever wordt het onderzoek uitgevoerd.

Tijdens het uitvoeren van het onderzoek worden de bevindingen gedocumenteerd in het onderzoeksverslag.

Na het onderzoek zal het document met richtlijnen worden opgesteld.

Activiteiten:		Op te leveren producten:
Opstelling onderzoeksontwerp		Onderzoeksontwerp
Uitvoeren onderzoek		Onderzoeksverslag
Documenteren onderzoek		Richtlijndocument
Opstellen richtlijndocument		
Bijhouden afstudeerscriptie		

Invoeringsfase

In deze fase worden de gerealiseerde documenten overgedragen aan de opdrachtgever. De richtlijnen die zijn opgesteld in de vorige fase worden toegepast op een app van TASS. De resultaten worden gepresenteerd.

Activiteiten:		Op te leveren producten:
Opstellen afstudeerscriptie		Afstudeerscriptie
Opstellen eindpresentatie		Eindpresentatie
Richtlijnen document toepassen		Aangepaste app volgens de richtlijnen
Opleveren van het project		

B.2 Planning

Belangrijke data:

Start stage:	03-09-2012
Inleveren PID:	25-09-2012
Inleveren PMP:	27-09-2012
Inleveren onderzoeksverslag:	20-12-2012
Inleveren richtlijnen document:	20-12-2012
Inleveren app:	01-02-2013
Inleveren afstudeerscriptie:	10-01-2013
Afstudeerzitting:	21-01-2013 – 01-02-2013
Logboeken:	Iedere week

6 Bijlage C: Initieel Risicologboek

NR:	Omschrijving:	Aktie:	Kans:	Impact:	Risico:
1	Onvoldoende medewerking in interviews met medewerkers	Ander persoon zoeken om toch de benodigde informatie te krijgen.	L	G	L
2	Onvoldoende contactmomenten met de opdrachtgever	Aannames maken	L	H	G
3	Langdurige ziekteperiode afstudeerder	Afspraken maken met bedrijfsbegeleider voor mogelijke uitstel en thuiswerken.	L	H	G
4	Kortdurige ziekteperiode afstudeerder	Afspraken maken met bedrijfsbegeleider voor mogelijke uitstel en thuiswerken.	G	L	L
5	Planning wordt niet gehaald	Nieuwe afspraken maken en de planning hierop aanpassen	G	G	G

De risico's worden als volgt berekend: $\text{Risico} = \text{Kans} * \text{Impact}$

		Impact		
		Hoog	Gemiddeld	Laag
Kans	Hoog	Hoog	Hoog	Gemiddeld
	Gemiddeld	Hoog	Gemiddeld	Laag
	Laag	Gemiddeld	Laag	Laag

7 Bijlage D: Kwaliteitsplan

D.1 Definition of Done

Tijdens het ontwikkelen van de richtlijnen wordt SCRUM toegepast. Een taak is done als het voldoet aan de volgende eisen:

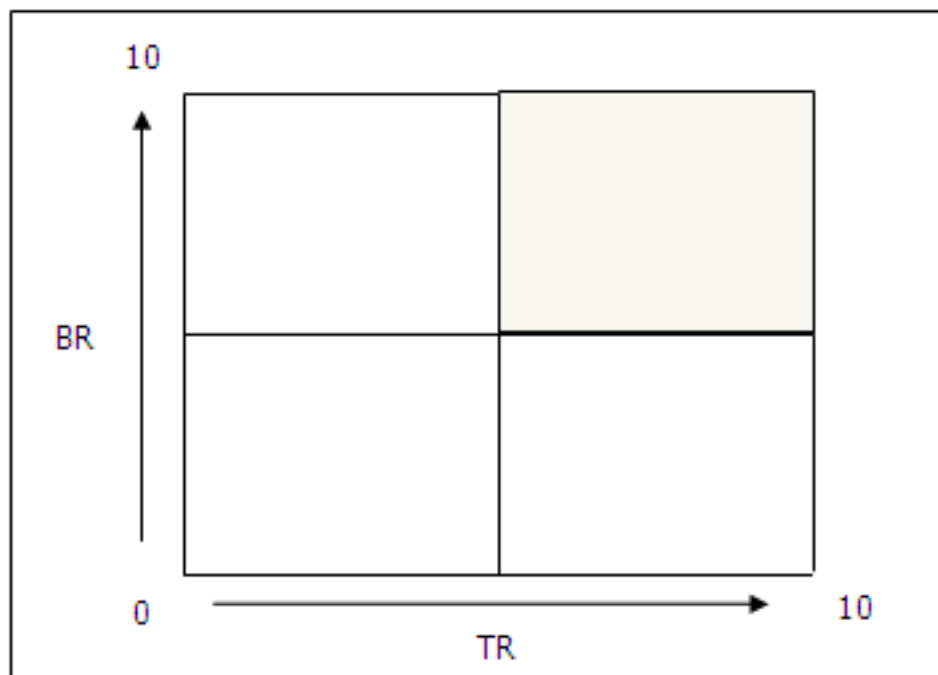
- Al de code is geschreven
- De code is getest zoals beschreven in het testplan
- De code is ingecheckt in SVN
- De wiki en dit document is up-to-date
- Geen openstaande bevindingen
- Trac is bijgewerkt
- De user story is geaccepteerd door de klant

D.2 Testplan

Dit testplan heeft betrekking op de code die ontwikkeld zal worden bij het opstellen van de richtlijnen.

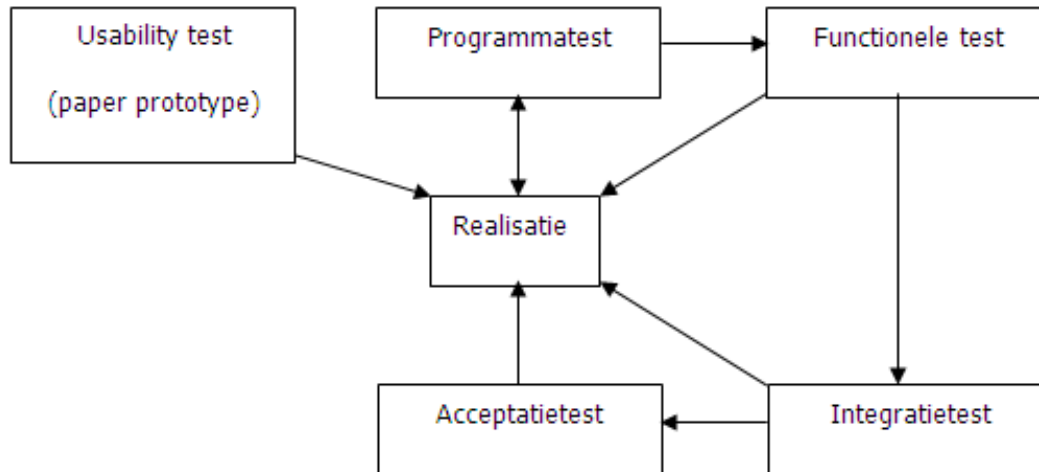
D.2.1 Scope

De taken die voortkomen uit een user story worden gewaardeerd met een Business Risk (BR) en een Technology Risk (TR). Deze waarde heeft een minimum van 1 en een maximum van 10. Als het gemiddelde van deze twee waardes 5 of hoger is (zie grijs vlak) is het testen van deze taak een must. De waarde voor TR worden geschat door het project team, de inschatting van de waarde voor BR wordt gedaan door de product owner.



D.2.2 Aanpak & testomgeving

Per sprint wordt onderstaand geïllustreerde proces doorlopen. Na iedere fase wordt gekeken of door kan gaan naar de volgende. Als dit niet het geval is wordt er teruggegaan naar de realisatie fase en volgen de test fases weer opnieuw.



In de volgende paragrafen zijn alle soorten testen die zullen worden uitgevoerd omschreven.

Usability test

Bij het maken van de user interface worden er paper prototypes gemaakt, deze worden getest op de gebruikers. Hier wordt getest of de user interface voldoet aan de verwachtingen en hoe lang het duurt tot de gebruiker een taak kan voltooien. Bij deze test zullen de volgende personen aanwezig zijn:

- Real Users, er worden steeds 5 gebruikers gebruikt bij het uitvoeren van een test om 85% van de usability problemen op te vangen.¹
- A facilitator, noteert de problemen die gevonden worden. Geeft de taken aan de gebruiker.
- A human computer, deze persoon beheert de paper prototype tijdens de test.
- Observers, het project team interpreteert de taken die de gebruiker uitvoert.

Programmatest

De gemaakte Javascript code en gebruikte code uit jQuery wordt getest door unittesten met gebruik van JSunit. De HTML wordt getest met gebruik van de tool Capybara.

Integratietest

In deze fase wordt er getest of de software op verschillende hardware zich hetzelfde gedraagt. Als dit niet het geval is moet de code worden aangepast of er moet native code geschreven worden. De verschillende hardware is beschreven in de paragraaf benodigde resources.

Functionele test

Er zullen functionele testen worden uitgevoerd door het gebruik van de testtool Cucumber. Hierin wordt gekeken of aan de user stories wordt voldaan.

¹ Why you only need to test with 5 people - <http://www.useit.com/alertbox/20000319.html>

D.3 Reviews

Aan het eind van iedere sprint zal er een sprint review meeting gehouden worden met de product owner. In deze meeting wordt besproken welke taken wel en niet zijn voltooid. De voltooide taken worden gepresenteerd. Er zal aan het einde van een sprint ook een sprint retrospective plaatsvinden, hierdoor wordt er gereflecteerd op de gemaakte sprint.

Voor iedere release wordt er een code review uitgevoerd door de product owner hierdoor wordt de gemaakte code bekeken door meer dan een persoon en kunnen mogelijke bugs gevonden worden voor een release.

Via een audit review wordt gecontroleerd of er gehouden wordt aan de afgesproken werkmethodes.

D.4 Versiebeheer

Het versienummer van documenten en software bestaat uit drie cijfers, startend met 0.0.0. Bij iedere aanpassing wordt het laatste cijfer met 1 verhoogd, bij iedere goedkeuring van een document door de product owner wordt het tweede nummer met 1 verhoogd en het derde cijfer weer terug op 0 gezet. Bij iedere release wordt het eerste cijfer met 1 verhoogd en het tweede en derde cijfer teruggezet op 0.

Bijlage B

Onderzoeksontwerp

Documenthistorie

Versie	Datum	Auteur	Omschrijving
0.0.1	02-10-2012	Joep van der Heijden	Eerste concept
0.0.2	08-10-2012	Joep van der Heijden	Vijfde deelvraag toegevoegd. Planning, onderzoeksstrategie uitgebreid.
1.0.0	12-10-2012	Joep van der Heijden	Document goedgekeurd.

Distributie

Versie	Datum	Naam:	Functie
0.0.1	02-10-2012	Frits van de Water	Opdrachtgever
0.0.2	08-10-2012	Frits van de Water	Opdrachtgever

Goedkeuring

Versie	Datum	Naam:	Functie
1.0.0	12-10-2012	Frits van de Water	Opdrachtgever

Inhoudsopgave

Conceptueel ontwerp	2
Doelstelling	2
Onderzoeksmodel	3
Vraagstelling	4
Technisch ontwerp	5
Onderzoeksstrategie	5
Onderzoeksmateriaal	6
Onderzoeksplanning	7

Conceptueel ontwerp

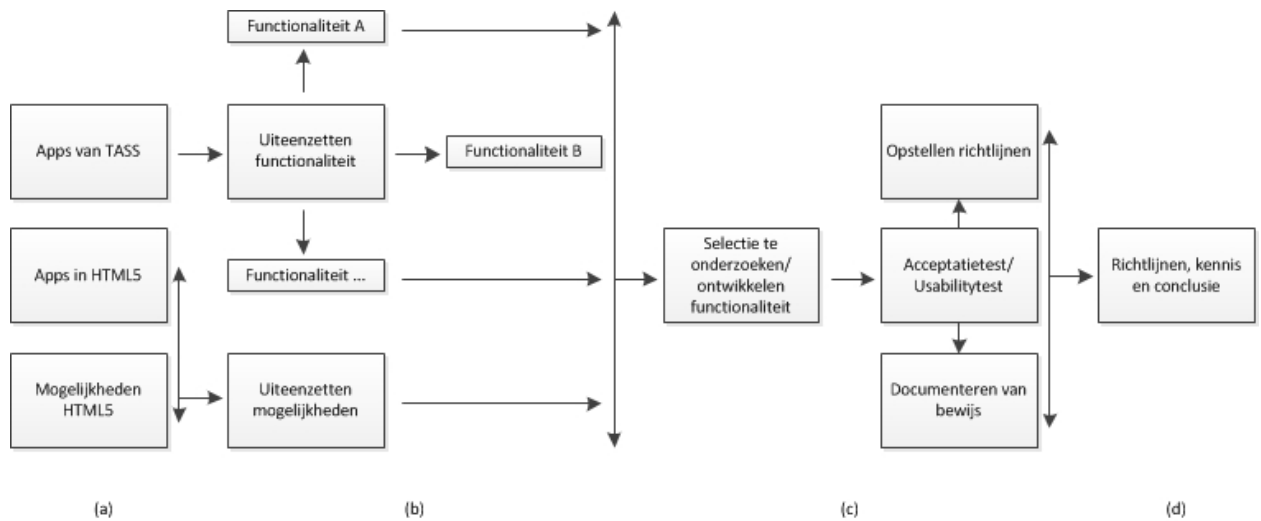
In het conceptueel ontwerp staat beschreven hoeveel, wat en waarom er onderzocht gaat worden.

Doelstelling

TASS ontwikkelt mobiele applicaties voor intern gebruik en voor klanten. Dit wordt nu nog gedaan op de native manier. Dit houdt in dat voor ieder besturingssysteem de applicatie in de programmeertaal die voor dat besturingssysteem bedoelt is geschreven moet worden. Voor iOS, het besturingssysteem van de iPhone, is dit de programmeertaal Objective-C. De native applicaties voor Android worden geschreven in Java.

Door het uitvoeren van dit onderzoek wil TASS meer kennis opdoen over wat de programmeertalen HTML5, Javascript en CSS3 kunnen betekenen voor het ontwikkelen van mobiele applicaties voor de mobiele besturingssystemen.

Onderzoeksmodel



In de eerste fase van het onderzoek wordt bekeken wat voor apps ontwikkeld worden door TASS en wat zij verwachten te ontwikkelen in de toekomst. Dit wordt gedaan door medewerkers en de opdrachtgever te interviewen. In deze fase wordt ook gekeken naar mogelijke oplossingen in HTML5, Javascript en CSS3 (a).

Hierna wordt alle gevonden informatie uiteengezet (b) zodat hier een selectie van kan worden gemaakt, deze selectie wordt dan ontwikkeld. Dit wordt daarna getest door de opdrachtgever en smartphone gebruikers. De resultaten van deze test vormen het bewijs, met de kennis die is opgedaan wordt een document met richtlijnen opgesteld(c).

De resultaten van de testen worden opgenomen in de conclusie (d).

Vraagstelling

Om de onderzoeksvraag, in hoeverre kan HTML5 de native apps van TASS vervangen, te kunnen beantwoorden worden eerst de volgende deelvragen beantwoord.

1. Wat zijn HTML5, CSS3 en Javascript?

Bij het beantwoorden van deze vraag wordt gekeken wat er allemaal nieuw is vergeleken met de vorige HTML en CSS versie en wat JavaScript kan betekenen bij het ontwikkelen van een app.

2. Wat voor apps worden er ontwikkeld door TASS?

Hier wordt gekeken wat voor TASS nu al heeft ontwikkeld en wat voor apps zij in de toekomst gaan ontwikkelen. Door deze vraag te beantwoorden is er een duidelijker beeld wat een HTML5 app moet kunnen om iets te kunnen betekenen voor TASS.

3. Welke besturingssystemen ondersteunen HTML5?

Er wordt onderzocht of HTML5 ondersteund wordt op de besturingssystemen Android en iOS en of HTML5 ook wordt geaccepteerd.

4. Zijn er al apps die geschreven zijn in HTML5 en hoe worden deze ervaren?

Er wordt hier gezocht naar apps die geschreven zijn in HTML5 voor verschillende besturingssystemen. Er wordt gekeken naar reacties van gebruikers om een goed beeld te vormen hoe deze ervaren worden.

5. Kan HTML5 dezelfde prestatie leveren als native?

Hier wordt gekeken of de mogelijkheden in HTML5 ook een goede prestatie kunnen leveren, net zoals native. En zo ja, hoe dit dan wordt gedaan; zo nee, hoe dit opgelost zou kunnen worden.

Technisch ontwerp

Onderzoeksstrategie

Om een antwoord te kunnen krijgen op het de vraag, in hoeverre HTML5 de native apps van TASS zou kunnen vervangen, wordt er van een aantal, nader te bepalen, functionaliteiten ontwikkeld. Deze functionaliteiten in HTML5, Javascript en CSS3 ontwikkeld voor verschillende besturingssystemen. De ontwikkelde functionaliteiten worden getest door de opdrachtgever in een acceptatietest. Mocht de opdrachtgever het nodig vinden, wordt er een usabilitytest uitgevoerd met de gebruikers.

Onderzoeksmateriaal

1. Wat is HTML5, CSS3 en JavaScript?

Bronnen	Ontsluiting
Media: video's (YouTube: "HTML5")	Inhoudsanalyse
Documenten: artikelen betreffende HTML5	Inhoudsanalyse

2. Wat voor apps worden er ontwikkeld door TASS?

Bronnen	Ontsluiting
Personen: medewerkers van TASS	Ondervraging (face-to-face)
Media: website van TASS	Inhoudsanalyse

3. Welke besturingssystemen ondersteunen HTML5?

Bronnen	Ontsluiting
Media: website van maker besturingssysteem	Inhoudsanalyse
Code: gemaakte code	

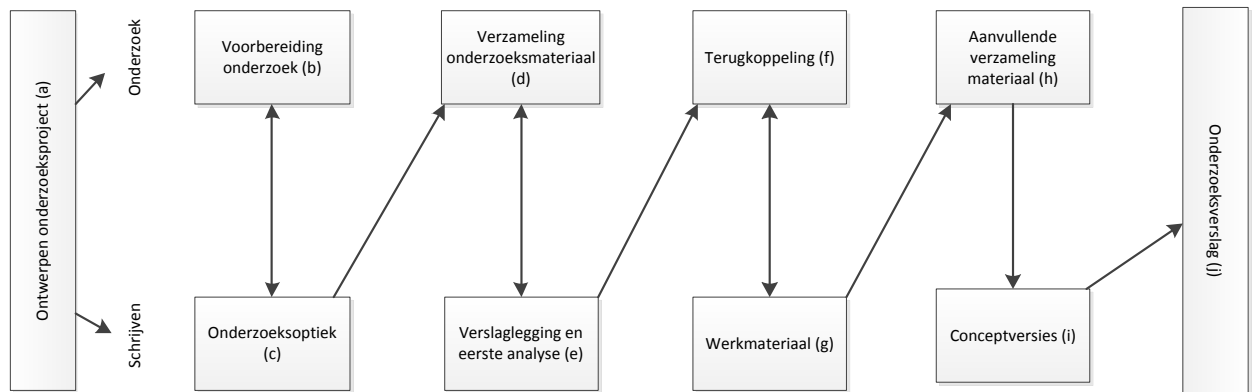
4. Zijn er al apps die geschreven zijn in HTML5 en hoe worden deze ervaren?

Bronnen	Ontsluiting
Media: opensource html5 apps, stores van besturingssystemen (reacties en waardering)	Inhoudsanalyse

5. Kan HTML5 dezelfde prestatie leveren als native?

Bronnen	Ontsluiting
Personen: gebruikers van de TASS apps	Ondervraging (face-to-face)
Tests	

Onderzoeksplanning



Bovenstaande planning van b tot en met e zal iedere sprint doorlopen worden per te ontwikkelen functionaliteit.

a. Ontwerpen onderzoek

Het maken van dit document. Het projectkader wordt hier verkend, er wordt duidelijk gemaakt wat de opdrachtgever precies als doel van het onderzoek ziet.

b. Voorbereiding onderzoek

Bepalen te ontwikkelen functionaliteiten. Er wordt overlegd met de opdrachtgever welke functionaliteiten hij graag ontwikkelt ziet worden. En er wordt gekeken of er andere applicaties gevonden kunnen worden die deze functionaliteiten ook hebben toegepast.

c. Uitwerking onderzoeksoptiek

De functionaliteit beschrijven. De testen worden geschreven. Schetsen indien nodig worden gemaakt.

De beschreven functionaliteit en de schetsen worden afgestemd met de opdrachtgever.

d. Verzameling onderzoeksmateriaal

Realisatie van de functionaliteiten.

e. Verslaglegging en eerste analyse

Document van bewijzen en het richtlijnen document wordt bijgewerkt.

f. Terugkoppeling

Sprintreview.

g. Ontwikkeling werkmateriaal

Het onderzoeksverslag wordt bijgewerkt met de opgedane kennis.

h. Aanvullend materiaal

Als er tijdens de vorige stappen is gebleken dat er meer gedaan moet worden om de vraagstelling te beantwoorden worden hier nieuwe taken aan de backlog toegevoegd.

Bijlage C

Sprint Notulen & Sprint Retrospectives

Sprint 1 Review Notulen

Aanwezigen: Frits van de Water, Jinmun Hu, Joep van der Heijden

Datum, tijd en locatie: Vrijdag 26 okt. 2012, 10:00-12:00, TASS hoofdkantoor.

Besproken: jQuery mobile, login, werkwijze, nieuwe sprint

jQuery mobile

In de eerste week van de sprint is geprobeerd de navigatie van pagina's te regelen met gebruik van jQuery mobile. Toen dit niet lukte is besloten dit zelf te schrijven, dit werkte wel maar de CSS transitions waren erg schokkerig. Er is toen besloten weer terug te gaan naar jQuery mobile en alle pagina's in de index te zetten. Dit werkt nu goed.

Login

Het component voor het inlogscherf is niet gemaakt in HTML5 webcomponent. Dit wordt nog niet ondersteund. Er is gebruik gemaakt van X-Tag om toch componenten te kunnen maken die later omgezet kunnen worden naar HTML5 webcomponenten wanneer dit ondersteund is.

Werkwijze

De manier van werken met Scrum was nog niet helemaal duidelijk. Wanneer er nu een taak gemaakt moet worden als het al in de definition of done staat was niet duidelijk.

De scenario's waren niet beschreven, dit moet wel bij behaviour-driven-development.

Er waren te veel zaken die gedaan moesten worden voor de eerste sprint waar de afstudeerder nog niet helemaal bekend mee is. Zo wordt er gebruik gemaakt van de scrumtool, SVN, trac, cucumber en wiki. Ook is de afstudeerder nog niet goed bekend met CSS waardoor er te veel tijd werd gestoken in de opmaak van het scherm.

Nieuwe sprint

In de nieuwe sprint worden de happy flow en de bad flow van het loginscherf gerealiseerd. Ook zal de happy flow van master-detail worden gerealiseerd.

Voordat er wordt gestart met ontwikkelen worden eerst de scenario's beschreven in Cucumber waarna de testen worden geschreven met behulp van Capybara.

Er zal gebruik worden gemaakt van de Chrome plugin tin.cr. Dit zorgt ervoor dat de CSS makkelijkere te ontwikkelen is.

Sprint 1 Retrospective

Start doing

- Voordat er ontwikkelt gaat worden moeten eerst de testen geschreven worden.
- Gebruik van tin.cr zodat het ontwikkelen van de CSS sneller en makkelijker gaat.

Stop doing

- Veel zelf programmeren i.p.v. jQuery en jQuery Mobile gebruiken.

Continue doing

- Gebruik van X-Tag om componenten te ontwikkelen.
- Gebruik van jQuery Mobile.

Sprint 2 Review Notulen

Aanwezigen: Frits van de Water, Jinmun Hu, Joep van der Heijden

Datum, tijd en locatie: Vrijdag 9 nov. 2012, 10:00-11:00&13:00-14:30, TASS hoofdkantoor.

Besproken: testen, master, detail, knop reactietijd, swipe navigatie, login, scrum, nieuwe sprint

Testen

In de sprint is Cucumber gebruikt om de applicatie te testen. Dit is goed verlopen maar er zijn wat problemen:

- De applicatie wordt nu getest in Chrome op de desktop en dus niet op de smartphones.
- Asynchrone calls worden niet goed getest. Ipv the wachten op een callback wordt er een seconde gewacht, dit was een snelle oplossing.

Master

Er is een master component gemaakt die een lijst kan weergeven. De namen van de attributen die meegegeven kunnen worden zijn te lang. De geïmplementeerde animatie in de lijst is goed.

Detail

Aan de detail component wordt alleen nog maar het jsonobject meegegeven. Dit is oké voor deze sprint.

Knop reactietijd

Er is een probleem met de touch feedback voor knoppen, dit gaat te langzaam of werkt niet goed. Er is veel tijd besteed om dit proberen op te lossen waardoor andere taken niet voltooid zijn. Het goed functioneren van de knop heeft een hoge prioriteit.

Swipe navigatie

De swipe navigatie werkt niet goed op iOS. Er is aangegeven dat de prioriteit hier niet hoog voor is.

Login

Het login component dat is gemaakt is oké. Alleen de scenario's voor geen response van de server en 3x geprobeerd in te loggen

Scrum

Het werken via Scrum is goed gegaan. Maar als een taak meer uren kost moet niet de estimated time worden aangepast maar de remaining time zodat de ingeschatte tijd altijd nog terug te vinden is.

Nieuwe sprint

De volgende verbeteringen worden meegenomen:

- De testen gaan voortaan draaien op Android en/of iPhone als dit mogelijk is.
- Testen met asynchrone functionaliteit moeten op de juiste manier afgehandeld worden.
- Meteor wordt gebruikt om sneller nieuwe versie van de applicatie te kunnen laden.

Er is afgesproken om verder te werken aan het master component en om het bestaande detail component te gebruiken als basis voor nieuwe detail componenten. Er zal een detail-personen component worden gemaakt die dit als basis neemt.

De problemen met de reactie tijd van de knop zal opgelost proberen te worden door een plugin te schrijven op jQuery.

De volgende scenario's gaan beschreven worden en het aantal uren om dit te realiseren zal worden geschat:

Voor het master component:

- Bad flow 1: een lijst van honderd elementen wordt in delen weergegeven.
- Bad flow 2: geen response bij ophalen lijst.
- Bad flow 3: geen data in de opgehaald lijst.
- Bad flow 4: de opgehaald lijst is geen JSON.
- Bad flow 5: afbeelding laden werkt niet.

Voor het detail-personen component:

- Happy flow: data wordt goed weergegeven op het scherm.

Sprint 2 Retrospective

Start doing

- Tests draaien op mobiele apparaten.
- Asynchrone tests goed implementeren.
- Gebruik van Meteor om veranderingen sneller op alle apparaten te laden.

Stop doing

- tin.cr werkte niet zoals verwacht, stop met dit te gebruiken.

Continue doing

- Het maken van tests in Cucumber.

Sprint 3 review notulen

Aanwezigen: Frits van de Water, Jinmun Hu, Joep van der Heijden

Datum, tijd en locatie: Vrijdag 23 nov. 2012, 10:00-11:00, TASS hoofdkantoor.

Besproken: testen, meteor, stroll.js, knop reactietijd, design, WindowsPhone, nieuwe sprint

Testen

In deze sprint is Cucumber weer gebruikt om de applicatie te testen. Dit is goed verlopen de verbeterpunten van de vorige keer zijn geïmplementeerd:

- Asynchroon testen.
- Testen op Android apparaat.
- Testen op iPhone apparaat.

Meteor

De tool Meteor gaat nu niet gebruikt worden. Het implementeren van deze tool kost te veel tijd omdat de code erdoor verandert moet worden.

Stroll.js

Stroll.js is een open source Chrome Experiment die gebruikt wordt om een mooi effect te geven aan het scrollen door meerdere elementen. Dit blijkt niet goed te werken als er foto's in de elementen staan. Dit is pas laat opgemerkt omdat tijdens de realisatie nog niet gewerkt werd met foto's. Er is nu besloten op dit moment niet verder te gaan met stroll.js zodat er meer tijd in het ontwikkelen van meerdere componenten gestoken kan worden.

Knop reactietijd

De reactietijd van de knoppen is opgelost door gebruik te maken van de fastClick extensie op jQuery. Deze extensie bleek, wanneer toegepast binnen een lijst, het scrollen onbruikbaar te maken. Voor het klikken op elementen in een lijst wordt nu een tap event gebruikt waarbij ook gekeken wordt of er tussen het moment van aanraken en het moment van loslaten gescrold is.

Design

Er is aangegeven dat er weer te veel tijd verloren is gegaan met het design te realiseren in CSS. Er is afgesproken om minder tijd te steken in het design. De tijd wordt meer gefocust op het ontwikkelen van componenten. Het is de bedoeling dat een designer de componenten later er goed laat uitzien

WindowsPhone

Er is in overleg met de opdrachtgever besloten om dat het gewenst is om een Nokia toestel aan te schaffen met dit besturingssysteem omdat Nokia en Microsoft, de maker van WindowsPhone, nauw met elkaar samenwerken. Dit toestel heeft te lang op zich laten wachten en er is daarom besloten om dit niet tijdens dit project aan te schaffen.

Nieuwe sprint

De volgende verbeteringen worden meegenomen:

- Minder aandacht besteden aan CSS.
- Kortere namen gebruiken voor de attributen van componenten.

De volgende taken worden meegenomen vanuit de user story "Masterlist/details – part 2" uit de vorige sprint:

- Data attribute Photo only.
- Bad flow 1: meer elementen.
- Bad flow 2: no response.
- Bad flow 3: geen data in json array.

De volgende scenario's gaan beschreven worden en het aantal uren om dit te realiseren zal worden geschat:

Voor het detail-personen component:

- Bad flow 1: data is aan te passen.

Voor het nieuwe component detail-werktijden:

- Happy flow: data wordt weergegeven op het scherm.

Het component detail-werktijden moet de werktijden van een werknemer kunnen weergeven. Hoe dit wordt weergegeven wordt in een nader te bepalen schets bepaald.

Sprint 3 Retrospectives

Start doing

- Minder aandacht besteden aan de CSS.
- Kortere namen gebruiken voor attributen.

Stop doing

- Gebruik van Meteor. Het implementeren van deze tool kost te veel tijd omdat de code erdoor verandert moet worden.

Continue doing

- Tests draaien op mobiele apparaten.
- Asynchrone tests goed implementeren.

Sprint 4 review notulen

Aanwezig: Frits van de Water, Joep van der Heijden

Datum, tijd en locatie: Vrijdag 8 dec. 2012, 10:00-11:00, TASS hoofdkantoor.

Besproken: detail-person-edit, detail-werktijden, design, nieuwe sprint

Detail-person-edit

Component is ontwikkeld, de data die is veranderd wordt via een event aan de ontwikkelaar gegeven. Dit is gewesnt.

Detail-werktijden

Component is ontwikkeld, de data over de werktijden wordt via JSON opgehaald. De attributen in het JSON bestand moeten nu nog de juiste namen hebben. Dit zou later eventueel veranderd kunnen worden.

Voor het tekenen van de grafiek wordt flotr2.0 gebruikt, dit werkt goed.

Design

Zoals tijdens de vorige sprint review afgesproken is ging er minder tijd besteed worden aan de CSS. Hier wordt nog steeds te veel tijd aan besteed.

Nieuwe sprint

De volgende verbeteringen worden meegenomen:

- Minder aandacht blijven besteden aan CSS.
- Swipe navigatie realiseren.
- Ontwikkelomgeving verbeteren, remote debuggen.
- Stabiele demo, alle ontwikkelde functionaliteit wordt gepresenteerd in een stabiele demo.

Sprint 4 Retrospective

Start doing

- Remote debugging

Continue doing

- Minder aandacht besteden aan CSS

Sprint 5 review notulen

Aanwezigen: Frits van de Water, Joep van der Heijden

Datum, tijd en locatie: Vrijdag 21 dec. 2012, 10:00-11:00, TASS hoofdkantoor.

Besproken: ontwikkelomgeving, stabiele demo, swipe navigatie, nieuwe sprint

Ontwikkelomgeving

Het debuggen op iPhone en Android werkt nu helemaal. Dit wordt gedaan door Chrome en Safari op de desktop.

Stabiele demo

Tijdens de review is een demo gegeven over de tot nu toe gemaakte componenten die allemaal in één app zitten. Deze demo is goed ontvangen.

Swipe navigatie

De navigatie door middel van een swipe gesture is gerealiseerd. Er kan nu ook op de iPhone genavigeerd worden naar de vorige pagina.

Nieuwe sprint

Er zullen in de nieuwe sprint twee nieuwe componenten worden ontwikkeld:

- Videocomponent, moet een video kunnen afspelen en tekst kunnen tonen over deze video.
- Menucomponent, dit wordt het component dat getoond wordt bij het starten van de app. Vanuit hier zal genavigeerd kunnen worden naar een lijst met video's en een lijst met medewerkers.

Verder worden de volgende taken uitgevoerd:

- Mapping van namen, als er een JSON bestand ingelezen moet worden op dit moment de attribute namen van dit JSON bestand meegegeven aan een component. Deze namen worden in het component getoond. De namen moeten aangepast kunnen worden zodat de gebruiker een andere tekst kan zien.
- Functions meegeven in attributen, in een attribute van een component moeten functions meegegeven kunnen worden.

Sprint 6 Review Notulen

Aanwezig: Frits van de Water, Joep van der Heijden, Jinmin Hi

Datum, tijd en locatie: Vrijdag 4 febr. 2013, 10:00-11:00, TASS hoofdkantoor.

Besproken: video, menu, mapping, functies in attributen, nieuwe sprint

Video

Het video component is ontwikkeld. Er kan een video worden afgespeeld en een titel en omschrijving meegegeven worden.

Het mastercomponent wordt gebruikt om een lijst weer te geven met videos.

Menu

Het start menu is heel simpel gehouden. Hierin wordt gelinked naar het lijstje met de videos en het lijstje met de werknemers.

Mapping

Het mappen van attributen in een JSON bestand is gelukt en toegepast op het detail-person component.

Functies

Functies kunnen nu meegegeven worden als listener in een attribute.

Nieuwe spint

Het project wordt afgerond in sprint 7. Er zal veel tijd worden besteed aan de scriptie. Daarna wordt de wiki bijgewerkt en wordt ervoor gezorgd dat alle document volledig zijn.

Bijlage D

Document van bewijzen

Doel van dit document

Om een meer technisch inzicht te krijgen in de constatering die gemaakt zijn is dit document gemaakt. In dit document staan ook problemen en eventueel gevonden oplossingen beschreven.

PhoneGap plug-ins

Tijdens het duidelijk krijgen wat voor apps TASS ontwikkelt is er functionaliteit genoemd die gebruikt kan worden met HTML5 functionaliteit of met native plug-ins. Er is gekeken hoe dit te realiseren kan zijn:

Locatie

De locatie van de gebruiker kan worden opgevraagd door de volgende code:

```
function onSuccess(position){
    //handle location
}

function onError(error){
    //handle error
}

//Opvragen van de locatie
navigator.geolocation.getCurrentPosition(onSuccess, onError);
```

Ook kan de locatie van de gebruiker in de gaten gehouden worden, dit wordt als volgt gedaan:

```
navigator.geolocation.watchPosition(onSuccess, onError);
```

NFC

Er is een native plug-in beschikbaar op GitHub, <https://github.com/chariotsolutions/phonegap-nfc>.

Deze kan gebruikt worden in een project en door middel van JavaScript aan geroepen worden.

Wi-Fi Direct

Op dit moment is er nog geen plug-in beschikbaar voor Wi-Fi Direct. Een andere afstudeerder is tijdens deze periode bezig geweest met het ontwikkelen van een Android applicatie die deze techniek gebruikt. Zijn code zou gebruikt kunnen worden bij het ontwikkelen van een eigen plug-in.

Accelerometer

De accelerometer kan opgevraagd worden door middel van de volgende code:

```
navigator.accelerometer.watchAcceleration(onSuccess, onError);
```

Camera

Door het gebruik van de PhoneGap plug-in kan de default camera app als volgt aangeroepen worden:

```
Navigator.camera.getPicture(onSuccess, onError);
```

Gerelateerd

<http://dev.w3.org/geo/api/spec-source>

<https://github.com/chariotsolutions/phonegap-nfc>

Configureren van PhoneGap

De WebView die PhoneGap gebruikt als de app gebuild wordt kan worden geconfigureerd door een bestand mee te sturen genaamd config.xml. De volgende configuraties zijn gedaan met PhoneGap:

- Op de iPhone is er een bounce effect als er gescrold wordt, dit kan uitgezet worden.
- De naam van de app
- De beschrijving van de app
- Access origin is aangepast. De pagina die geladen wordt in de WebView mag alleen maar calls maken naar zijn eigen origin. Omdat de app toegang moet kunnen hebben tot andere bronnen is dit aangezet.

Het config.xml bestand ziet er als volgt uit:

```
<?xml version="1.0" encoding="UTF-8" ?>
<widget xmlns = "http://www.w3.org/ns/widgets"
xmlns:gap = "http://phonegap.com/ns/1.0"
id        = "nl.tass.dora"
version   = "0.0.1">

    <name>DORA</name>
    <description>
        Een voorbeeld applicatie om de functionaliteit te testen.
    </description>
    <access origin="*" />

    <preference name="webviewbounce" value="false" />
</widget>
```

Viewport

Om ervoor te zorgen dat de app de juiste grootte heeft zodat er optimaal gebruik gemaakt wordt van de ruimte die de web view biedt wordt de viewport gebruikt. Dit wordt als volgt gedaan:

```
<meta name="viewport" content="width=device-width, initial-scale=1.0,  
    user-scalable=yes">
```

Om ervoor te zorgen dat de gebruiker niet kan inzoomen kan de user-scalable op no gezet worden of de maximum-scale op 1.0:

```
<meta name="viewport" content="width=device-width, initial-scale=1.0,  
    maximum-scale=1.0 user-scalable=no">
```

Navigatie binnen de app

Huidige situatie

Voor het ontwikkelen van de navigatie is gekeken hoe dit gedaan wordt op Android en iPhone waardoor de volgende vragen zijn beantwoord:

Hoe wordt er genavigeerd naar een nieuwe pagina?

Beide, Android en iPhone, maken simpelweg gebruik van knoppen en links om naar een nieuwe pagina te navigeren.

Hoe wordt er genavigeerd naar de vorige pagina?

Android heeft een back knop in het besturingssysteem geïntegreerd zitten. Als deze wordt ingedrukt wordt er genavigeerd naar de vorige pagina. Deze back knop bevindt zich onderaan in het scherm.

iPhone gebruikt een back knop die gemaakt moet worden door de ontwikkelaar, deze staat meestal gepositioneerd in de linksbovenhoek.

Nieuwe situatie

De Android back knop zal gebruikt worden om terug te kunnen navigeren naar de vorige pagina. Om niet de hele pagina te hoeven aanpassen op Android zodat op de iPhone ook terug genavigeerd kan worden wordt er gebruik gemaakt van een swipe gesture. Als er van de linker naar de rechter kant van het scherm geswiped wordt navigeert de applicatie terug naar de vorige pagina. Tussen het wisselen van de pagina wordt een slide animatie gebruikt.

Er is een JavaScript bestand ontwikkeld om ontwikkelaars te helpen met de navigatie genaamd tass-helper.js. In het document van richtlijnen staat beschreven hoe dit te gebruiken.

Voor het realiseren van de swipe gesture zijn de HTML5 touchevents gebruikt, deze zijn:

- touchstart, wordt getriggerd als de gebruiker het scherm met de vinger aanraakt.
- touchend, wordt getriggerd als de gebruiker het scherm met de vinger los laat.
- touchmove, wordt getriggerd als de gebruiker zijn vinger over het scherm beweegt.

In de eventhandler van touchstart worden de coördinaten van de aanraking opgeslagen in een variabele die ook gedeeld wordt met de andere eventhandlers. De touchmove eventhandler controleert of er de juiste kant op beweegt wordt, als dit het geval is, wordt naar een instelbare afstand, een boolean op true gezet die aangeeft dat er een legale swipe is gedaan om terug te gaan naar de vorige pagina. Als de gebruiker zijn vinger dan loslaat van het scherm controleert deze eventhandler of de boolean op true staat en navigeert dan een pagina terug, daarna verwijdert de eventhandler alle geregistreerde eventhandlers voor deze actie.

Het registreren en inlezen van een touchevent:

```
var page = document.getElementById('my-page');
function onTouchStart(e){
    e.pageX; //uitlezen van de X coördinate
    e.pageY; //uitlezen van de Y coördinate
}
page.addEventListener('touchstart', onTouchStart);
```

Het wisselen van pagina wordt hieronder beschreven. In dit voorbeeld zijn er twee pagina's:

```
<div id="page-one" data-role="page">
</div>
<div id="page-two" data-role="page">
</div>
<script>
    $.mobile.changePage('page-two', {
        transition : 'slide'
    }
);
</script>
```

Het slide effect dat gebruikt wordt is officieel nog niet ondersteund in iOS en Android. Om dit toch kunnen gebruiken in jQuery Mobile moet er iets aangepast worden. jQuery Mobile checkt of het effect ondersteunt wordt door, dit gebeurt in de function `_maybeDegradeTransition`. Door de volgende wijziging wordt de controle niet meer uitgevoerd:

```
//Oud:
//$.mobile._maybeDegradeTransition = $.mobile._maybeDegradeTransition ||
//    function( transition ) {
//        return transition;
//    };
//Nieuw:
$.mobile._maybeDegradeTransition = function( transition ) {
    return transition;
};
```


Problemen en oplossingen

Probleem

Bij het navigeren naar andere schermen in de app door nieuwe HTML pagina's in te laden met behulp van jQuery mobile worden sommige effecten genegeerd of werken erg traag.

Issue: <https://github.com/jquery/jquery-mobile/issues/3680>

Voorstel

Er is door de afstudeerder in Javascript een eigen navigatiesysteem gemaakt zodat de gebruiker door de verschillende schermen kan navigeren. Dit is gerealiseerd door twee containers in de hoofdpagina te hebben. In container 1 staat de, als aanwezig, vorige pagina, deze pagina is niet zichtbaar. In container 2 staat de huidige pagina die wordt weergegeven. Als er naar een nieuwe pagina genavigeerd moet worden wordt de vorige pagina de nieuwe huidige pagina en de oude pagina de vorige pagina. Dit wordt in één bestand gedaan zodat er een animatie tussen de schermwisseling afgespeeld kan worden.

Al de HTML code die als een pagina diende werd opgeslagen in een apart bestand. Deze pagina werd dan geladen in een container in het index bestand door de jQuery functie load() aan te roepen.

Voor dit voorstel is niet gekozen omdat de animatie tussen het wisselen van de pagina's niet goed werkt.

Oplossing

Er wordt gebruik gemaakt van de jQuery mobile attribute data-role="page" en van de functie changePage. Hierdoor blijven alle pagina's wel in één bestand staan. Dit is gecommuniceerd naar de opdrachtgever en hij heeft aangegeven dat dit geen probleem is.

Probleem

Sommige Android devices, zoals de Galaxy Nexus, hebben een bug in Chrome. Deze bug zorgt ervoor dat het touchmove event niet goed werkt.

Mogelijke oplossing

Dit probleem kan worden opgelost door een touchmove event te registreren op het document. In de handler wordt het normale gedrag uitgeschakeld:

```
document.addEventListener('touchmove', function(event) {  
    event.preventDefault();  
})
```

Door dit toe te passen ontstaat er een nieuw probleem: er kan niet meer gescrold worden in de pagina.

Het probleem is minder groot dan gedacht, omdat de bug alleen in Chrome zit. De WebView van Android wordt gebruikt, op Chrome wordt alleen getest voor toekomstig gebruik.

Openstaande problemen

Tijdens de eerste pagina wisseling op een Android apparaat is de animatie fade terwijl deze slide moet zijn. Dit werkt op Chrome wel de eerste keer, dit probleem lost zich waarschijnlijk vanzelf sinds Google Chrome wil gaan gebruiken voor de WebView.

Inloggen

Huidige situatie

Dit verschilt nu per app. Zo wordt in de field management app de username en password vergeleken met een database op het device zelf. Er is aangegeven door de opdrachtgever dat er op dit moment gebruik gemaakt kan worden van een simulatie.

Bij TASS zijn ze bezig om een inlogsysteem te maken met gebruik van OAuth, dit is nog niet gerealiseerd.

Nieuwe situatie

Er wordt een gebruikersnaam en wachtwoord veld getoond aan de gebruiker. Deze worden gestuurd naar de simulatie als de gebruiker op de inlog knop drukt.

Verder is in dit scherm een forgot password knop en een registreren knop aanwezig die de gebruiker naar de juiste pagina stuurt.

Openstaande taken

Het implementeren van OAuth.

Webcomponenten

Problemen en oplossingen

Probleem

Door het gebruik van webcomponent kunnen we eigen herbruikbare componenten maken. Dit wordt nog niet ondersteund.

Oplossing

Er wordt gebruik gemaakt van de X-Tag JavaScript bestand om de componenten toch te kunnen gebruiken. Dit kan later, als webcomponenten ondersteund worden, worden aanpast.

Gerelateerd

De website van X-Tag: <http://x-tags.org>

Data opslaan

In de app moet de mogelijkheid zijn om data op te slaan. Bij het inlogcomponent wordt dit bijvoorbeeld gebruikt voor de username en password. Dit wordt gedaan door het gebruik van localStorage, de username en password worden in een object opgeslagen zodat deze in JSON format weggeschreven kunnen worden:

```
var cred = {};  
cred.username = 'username';  
cred.password = 'password';  
try{  
    localStorage.credentials = JSON.stringify(cred);  
} catch(e){  
    console.log(e);  
}
```

Problemen en oplossingen

Probleem

W3C heeft aangekondigd dat WebSQL geen standaard meer is.

Oplossing

Gebruik maken van webStorage en localStorage.

Probleem

Het opslaan van localStorage werkt niet en laat een functie crashen als dit niet wordt ondersteund, zoals bij incognito-mode.

Oplossing

Try catch gebruiken zodat de data niet wordt opgeslagen maar de functie niet crasht.

Gerelateerd

W3C over webSQL: <http://www.w3.org/TR/webdatabase/>

W3C webstorage: <http://dev.w3.org/html5/webstorage/>

Algemeen: testen

Problemen en oplossingen

Probleem

async data laden in pagina's laat de tests mislukken doordat een stap uit de test al wordt uitgevoerd als de data nog niet geladen is.

Oplossing

Veel functies in Capybara proberen al opnieuw een element te zoeken als deze niet gevonden is. Door de `default_waiting_time` hoger te zetten worden veel stappen succesvol opgelost.

Om te controleren of er data op een bepaalde pagina staat wordt er nu gebruik gemaakt van de functie:

```
page.should have_content "data"
```

Deze functie blijft zoeken naar de data totdat deze gevonden is of totdat de `default_waiting_time` op is.

Image source

Problemen en oplossingen

Probleem

Als de source van een image niet geladen kan worden moet er een standaard plaatje getoond worden. Dit wordt als volgt gedaan:

```
var $img = $('img');  
$img.error(function() {  
    $(this).attr('src', 'missingperson.gif');  
})  
.attr('src', urlToMyPhoto);
```

Op de iPhone werkt dit niet, de .error function wordt niet aangeroepen bij een 403/401 error. Op Chrome werkt dit wel maar in de oude browser van Android, die Phonegap gebruikt, niet.

Oplossing

De situatie blijft zoals het nu is. Als een gebruiker geen toegang heeft tot een afbeelding wordt deze simpel weg niet getoond.

Stroll.js

Problemen en oplossingen

Probleem

Om een weergegeven lijst een mooi effect te geven tijdens het scrollen wordt er gebruikt gemaakt van stroll.js. Dit is een open source Chrome Experiment. Als er in een lijst afbeeldingen getoond worden neemt het paint event om deze afbeelding te tekenen te veel tijd in beslag waardoor het scrollen erg traag en haperend gaat.

Oplossing nog te doen. Er is gekozen om het scroll effect niet te implementeren, dit i.v.m. tijd.

Gerelateerd

De source van stroll.js <https://github.com/hakimel/stroll.js>

Scherm met bewerkbare data

Huidige situatie

Voor het bepalen van de flow van dit scherm is gekeken naar Facebook, LinkedIn en de Google People app.

Facebook laat maar een bepaald aantal data bewerkbaar zijn via mobiel. Facebook laat data in groepen zien. Sommige groepen zijn wel en sommige groepen zijn niet bewerkbaar. Als een groep wel bewerkbaar is wordt dit getoond door een edit knop boven deze groep. Als hierop gedrukt wordt worden de tekstlabels textboxes en is er een save knop om de wijzigingen op te slaan en een cancel knop om te annuleren.

LinkedIn heeft een edit knop in het profiel van de gebruiker staan, als hierop gedrukt wordt word een nieuw scherm geladen waarin alle data bewerkt kan worden. Er kan worden opgeslagen door op done te drukken.

De Google People app doet hetzelfde als LinkedIn, alleen heeft de app de knop achter een uitklapbaar menu zitten.

Nieuwe situatie

In het component waar de details worden weergegeven komt een edit knop te staan. Als op deze knop gedrukt wordt word er een scherm getoond met dezelfde details alleen dan in inputvelden. Op dit scherm staat een Save en Cancel knop, beide knoppen sturen de gebruiker terug naar de vorige pagina.

Als de gebruiker een foto wil uploaden wordt dit gedaan door een input veld met type file. Dit wordt door het onderliggende besturingssysteem geregeld.

Gerelateerd

X-Tag component data-person-edit

jQuery Mobile button

Problemen en oplossingen

Probleem button verbergen

jQuery mobile zet zijn eigen style om een button heen. Dus i.p.v. het input element met type="button" een style te geven verbergt hij dit. En zet er een div omheen met de juiste style. Hierdoor werkt onderstaande code niet meer.

```
<input id="myinput" type="button" value="Edit"/>
$('#myinput').hide();
```

Boven staande button wordt als volgt weergegeven in de bron:

```
<div data-corners="true" data-shadow="true" data-iconshadow="true"
    data-wrapperels="span" data-icon="null" data-iconpos="null"
    data-theme="c" class="ui-btn ui-btn-up-c ui-shadow ui-btn-corner-all"
    aria-disabled="false">
  <span class="ui-btn-inner ui-btn-corner-all">
    <span class="ui-btn-text">Edit</span>
  </span>
  <input id="mypinput" type="button" value="Edit" class="ui-btn-hidden"
    aria-disabled="false">
</div>
```

Om de button nu te kunnen verbergen moet de div verborgen worden. Dit wordt op de volgende manier gedaan:

```
$('#mypinput').closest('.ui-btn').hide();
```

Het probleem hierbij is dat de div nog niet om de input heen zit als de page nog niet getoond is.

Oplossing

Het plaatsen van een container om de button en deze te verbergen. Doordat de container om de button heen zit blijft de container ook om de nieuwe HTML code heen staan. Als de container dus wordt verbergt wordt de knop ook verbergt.

Gerelateerde files

Dit wordt toegepast in de detail-person component om de edit knop te verbergen als de attribute data-editable niet bestaat.

Grafiek tekenen

In het X-Tag detail-werktijden wordt een grafiek getekend. Dit wordt gedaan op het HTML5 canvas element met gebruik van de JavaScript library flot2.0. Deze library heeft een Flotr object met de draw methode waarin een object aan meegeven kan worden.

Het HTML5 canvas element is bedoelt voor het tekenen van bijvoorbeeld grafieken, vormen en games. Op dit moment kan er op het element alleen 2D tekeningen worden getekend, in de toekomst zal ook 3D ondersteund gaan worden.

Aan de X-Tag component kan een URL naar een JSON bestand worden meegestuurd. Op dit moment moet dit bestand de volgende format hebben:

```
{
  "week1": [
    {
      "day": 1,
      "start": 8,
      "eind": 17
    },
    {
      "day": 2,
      "start": 8,
      "eind": 17
    },
    {
      "day": 3,
      "start": 8,
      "eind": 17
    },
    {
      "day": 4,
      "start": 8,
      "eind": 17
    },
    {
      "day": 5,
      "start": 8,
      "eind": 17
    }
  ]
}
```

Er wordt een array meegegeven met weken, daarin staan de dagen met begin en eind uren.

Gerelateerd

<http://www.w3.org/wiki/HTML/Elements/canvas>

<http://www.humblesoftware.com/flotr2>

Video

Huidige situatie

Video's worden fullscreen afgespeeld.

Nieuwe situatie

HTML5 heeft een tag voor video's, de <video> tag. Hierin kan de source worden gezet met de <source> tag, waarin weer verschillende bronnen opgegeven kunnen worden. Dit kunnen er meer zijn zodat er een fallback systeem ontstaat, mocht een bestandtype niet afgespeeld kunnen worden dan wordt de volgende geprobeerd.

Op Android wordt netjes de video binnen de video tag afgespeeld. Op de iPhone wordt de video geopend in een videospeler en daar geopend.

De volgende video formaten worden voornamelijk gebruikt:

- .mp4 (H.264 + AAC)
- .ogg/.ogv (Theora + Vorbis)
- .webm (VP8 + Vorbis)

Er is gekozen om H.264 te gebruiken omdat deze standaard de meest gebruikte standaard lijkt te gaan worden.

De video tag heeft de volgende attributen die ook gebruikt worden in het ontwikkelde video component:

- autoplay, geeft aan of de video meteen moet beginnen met afspelen.
- controls, geeft aan of de video controls moet hebben.
- loop, geeft aan of de video opnieuw moet beginnen met afspelen wanneer deze eindigt.
- muted, geeft aan of de video geluid mag produceren of niet.

Ook kan via JavaScript de functions play en pause aangeroepen worden:

```
$('#video')[0].play();  
$('#video')[0].pause();
```

Events voor om play en pause mee op te vangen zijn ook aanwezig:

```
$('#video').on('play', function(e){  
    //handle  
});  
$('#video').on('pause', function(e){  
    //handle  
});
```

Gerelateerde files

<http://www.w3.org/wiki/HTML/Elements/video> De documentatie van de video tag.

Bijlage E

Document met Richtlijnen

Inhoudsopgave

Doel van dit document	2
Gebruik van X-Tag componenten	3
Component: login	5
Component: master	8
Component: detail-person	11
Component: detail-person-edit	14
Component: detail-working	15
Component: video	16
Javascript: tass-helper.js	18

Doel van dit document

In dit document is de documentatie te vinden van alle X-Tag componenten. In het volgende hoofdstuk, gebruik van X-Tag componenten, wordt uitgelegd hoe deze componenten gebruikt kunnen worden. Er staat een voorbeeld in die duidelijk aangeeft hoe het component master moet worden gebruikt.

Bij ieder component in dit document staat alle attributen, events en methodes die dit component heeft.

In het laatste hoofdstuk wordt de documentatie getoond van een JavaScript helper bestand, genaamd tass-helper.js, die de ontwikkelaar kan helpen met het navigeren tussen pagina's.

Gebruik van X-Tag componenten

Een X-Tag component is een JavaScript bestand. Dit bestand moet geïmporteerd worden voordat jQuery en zijn. Een X-Tag component heeft de X-Tag JavaScript bestand als afhankelijkheid. Importeer deze dus voor de X-Tag componenten.

Zoals hierboven wordt beschreven moeten de JavaScript bestanden van X-Tag en jQuery geïmporteerd worden. Als er gebruik gemaakt wordt van jQuery Mobile moet dit ook eerst gedaan worden. Hieronder staat een voorbeeld hoe het component master gebruikt moet worden voor het volgende JSON bestand:

```
[
  {
    "birthday": "17 mei",
    "department": "Direct",
    "imageUrl": "missingperson.gif",
    "internEmail": "rob.knubben@tass.nl",
    "name": "Rob Knubben",
  },
  {
    "birthday": "13 juli",
    "department": "People",
    "imageUrl": "missingperson.gif",
    "internEmail": "josee.zielman@tass.nl",
    "name": "Josee Zielman",
  },
  {
    "birthday": "28 april",
    "department": "Direct",
    "imageUrl": "missingperson.gif",
    "internEmail": "Jin.Hu@tass.nl",
    "name": "Jinmin Hu",
  },
  {
    "birthday": "13 februari",
    "department": "Staff",
    "imageUrl": "missingperson.gif",
    "internEmail": "heijdvdj@tass.nl",
    "name": "Joep van der Heijden",
  }
]
```

```

<html>
<head>
<script src="/js/jquery.js"></script>
<script src="/js/jqueryMobile.js"></script>
<script src="/js/xtag.js"></script>
<script src="/js/master.js"></script>
<script>
    function handle(data){
        //log the jsonobject
        console.log(data.jsonobject);
    }
</script>
<link href="css/master.css"/>
<link href="css/jqueryMobile.css"/>
</head>
<body>
    <div data-role="page">
        <x-master data-src="json/people.json" data-name
            data-name-name="name" data-details
            data-details-name="department" data-photo
            data-photo-name="imageUrl" data-onitemclicked="handle">
        </x-master>
    </div>
</body>
</html>

```


Component: login

Dit component toont een veld voor de gebruikersnaam en een veld voor het wachtwoord. Naast de login knop biedt dit component de mogelijkheid om een registreerknop en wachtwoord vergeten knop te tonen.

Tagnaam: x-login

Attributen

`data-hint-username`

De tekst die weergegeven moet worden als hint in het username veld.

Default: username

`data-hint-password`

De tekst die weergegeven moet worden als hint in het password veld.

Default: password

`data-bt-register`

Boolean. Geeft aan of de register knop weergegeven moet worden. Als de attribute aanwezig is zonder waarde, met lege waarde of met waarde true wordt de knop weergegeven.

Default: true

`data-bt-forgot`

Boolean. Geeft aan of de forgot knop weergegeven moet worden. Als de attribute aanwezig is zonder waarde, met lege waarde of met waarde true wordt de knop weergegeven.

Default: true

`data-bt-text-login`

De tekst van de login knop.

Default: Login

`data-bt-text-register`

De tekst van de register knop.

Default: Register

`data-bt-text-forgot`

De tekst van de forgot knop.

Default: Forgot password

data-onlogin

De naam van de listener. Deze wordt aangeroepen als er op de login knop gedrukt wordt.

Default: undefined

data-onfailed

De naam van de listener. Deze wordt aangeroepen als het inloggen mislukt is.

Default: undefined

data-onregister

De naam van de listener. Deze wordt aangeroepen als er op de register knop gedrukt wordt.

Default: undefined

data-onforgot

De naam van de listener. Deze wordt aangeroepen als er op de forgot knop gedrukt wordt.

Default: undefined

Events

onLogin(event)

De gebruiker zijn inloggegevens zijn correct.

Parameters

Event, standaard event object.

onFailed(event)

De gebruiker zijn inloggegevens zijn niet correct.

Parameters

event, standaard event object.

onRegister(event)

Er wordt op de registreer knop gedrukt.

Parameters

event, standaard event object.

onForgot(event)

Er wordt op de wachtwoord vergeten knop gedrukt.

Parameters

event, standaard event object.

Methodes

`setHintUsername(hint)`

Toont de meegegeven waarde in de username veld als hint.

Parameters:

hint, de nieuwe waarde van de hint in het username veld.

`setHintPassword(hint)`

Toont de meegegeven waarde in de password veld als hint.

Parameters:

hint, de nieuwe waarde van de hint in het password veld.

`setBtRegister(value, text)`

Toont de register knop of verbergt deze en toont de juiste tekst op de knop.

Parameters:

value, geeft aan of de knop getoond moet worden(true) of verborgen(false).

text, de nieuwe tekst voor de knop.

`setBtForgot(value, text)`

Toont de forgot knop of verbergt deze en toont de juiste tekst op de knop.

Parameters:

value, geeft aan of de knop getoond moet worden(true) of verborgen(false).

text, de nieuwe tekst voor de knop.

`setBtLogin(text)`

Toont de nieuwe tekst op de login knop.

Parameters:

text, de nieuwe tekst voor de knop.

Component: master

Geef een lijst weer van de JSON data die is opgehaald uit een JSON bestand. Een item in de lijst kan een naam, details en een foto weergeven.

Tagnaam: x-master

Attributen

data-photo

Boolean. Geeft aan of er een foto weergegeven moet worden. Als de attribute aanwezig is zonder waarde, met lege waarde of met waarde true wordt er een foto weergegeven.

Default: false

data-details

Boolean. Geeft aan of er details weergegeven moet worden. Als de attribute aanwezig is zonder waarde, met lege waarde of met waarde true worden de details weergegeven.

Default: false

data-name

Boolean. Geeft aan of de naam weergegeven moet worden. Als de attribute aanwezig is zonder waarde, met lege waarde of met waarde true wordt de naam weergegeven.

Default: false

data-src

Een URL naar een JSON bestand.

Default: undefined

data-photo-name

De naam van de photo attribute in het JSON bestand.

Default: undefined

data-details-name

De naam van de details attribute in het JSON bestand.

Default: undefined

data-name-name

De naam van de name attribute in het JSON bestand.

Default: undefined

data-onitemclicked

De naam van de listener. Deze wordt aangeroepen als er op een listitem wordt gedrukt.

Default: undefined

Events

`onItemClicked(event)`

Er wordt op een item geklikt. Het JSON object wordt meegestuurd.

Parameters

event, standaard event object met extra attribute genaamd jsonobject.

Voorbeeld

```
el_master.addEventListener('onItemClicked', function(event){  
    data.jsonobject; //het jsonobject  
});
```

Methodes

`getContent(callback)`

Haalt de gegevens op uit het JSON bestand die is meegegeven als dit nog niet is gedaan.

Parameters

callback, function die aangeroepen wordt als de content geladen is.

`refresh(callback)`

Haalt de gegevens op uit het JSON bestand die is meegegeven.

Parameters

callback, function die aangeroepen wordt als de content geladen is.

`setSource(URL)`

Zet de source van het JSON bestand voor het component. Laadt de gegevens nog niet in het component.

Parameters

URL, de locatie van het JSON bestand.

`setPhoto(show, key)`

Geeft aan of er een foto weergegeven moet worden.

Parameters:

show, boolean die aangeeft of de foto getoond moet worden.

key, de naam van de photo attribute in het JSON bestand.

`setDetails(show, key)`

Geeft aan of er details weergegeven moeten worden.

Parameters:

show, boolean die aangeeft of de details getoond moet worden.

key, de naam van de detail attribute in het JSON bestand.

`setName(show, key)`

Geeft aan of er naam weergegeven moeten worden.

Parameters:

show, boolean die aangeeft of de naam getoond moet worden.

key, de naam van de naam attribute in het JSON bestand.

Component: detail-person

Geeft de details weer van de persoon. Deze worden doorgegeven via een JSON object.

Tagnaam: x-detail-person

Attributen

data-photo-name

De naam van de key van de foto in het JSON bestand.

Default: undefined

data-maininfo-names

Komma gescheiden waardes. De namen van de keys in het JSON bestand die rechts van de foto getoond moeten worden.

Default: undefined

data-otherinfo-names

Komma gescheiden waardes. De namen van de keys in het JSON bestand die onder de foto getoond moeten worden.

Default: undefined

data-data

De URL naar het JSON bestand.

Default: undefined

data-namemapping

String. De keys uit het JSON bestand die gemapped moeten worden aan waardes die getoond worden in het component.

Default: undefined

Voorbeeld:

```
<x-detail-person data-data="/json/p.json"
  data-maininfo-names="name,department"
  data-name-mapping="name=Naam,department=Afdeling">
</x-detail-person>
```

data-editable

Boolean. Geeft aan of de edit knop getoond moet worden. Als de attribute aanwezig is zonder waarde, met lege waarde of met waarde true wordt de knop weergegeven.

Default: false

data-bt-text-edit

De tekst van de edit knop.

Default: edit

data-onedit

De naam van de listener. Deze wordt aangeroepen als er op de edit knop gedrukt wordt.

Default: undefined

data-working

Boolean. Geeft aan of de werktijden knop getoond moet worden. Als de attribute aanwezig is zonder waarde, met lege waarde of met waarde true wordt de knop weergegeven.

Default: false

data-onworking

De naam van de listener. Deze wordt aangeroepen als er op de werktijden knop gedrukt wordt.

Default: false

Events

onEditRequest(event)

Er wordt op de edit knop gedrukt.

Parameters

event, standaard event object.

onWorkingRequest(event)

Er wordt op de werktijden knop gedrukt.

Parameters

event, standaard event object.

Methodes

`setData(data)`

Verwerkt de meegegeven data en toont dit in het component.

Parameters

data, de data van de persoon die weergegeven moet worden.

`clear()`

Leegt de informatie die getoond wordt in het component.

`setPhotoName(name)`

Geeft de naam door van de key van de foto in het object en toont deze in het component.

Parameters:

name, de waarde van de key.

`setMainInfoNames(names)`

Geeft de namen door van de keys van de main infos in het object en toont deze in het component.

Parameters:

names, de waardes van de key gescheiden door een komma.

`setOtherInfoNames(names)`

Geeft de namen door van de keys van de other infos in het object en toont deze in het component.

Parameters:

names, de waardes van de key gescheiden door een komma.

`setEditable(value, text)`

Toont de edit knop als er true wordt meegegeven, als false wordt deze verborgen.

Parameters:

value, boolean om aan te geven of de edit knop getoond moet worden.

text, de tekst die getoond moet worden op de knop.

`setWorking(value, text)`

Toont de werktijden knop als er true wordt meegegeven, als false wordt deze verborgen.

Parameters:

value, boolean om aan te geven of de werktijden knop getoond moet worden.

text, de tekst die getoond moet worden op de knop.

`setNameMap(key, label)`

Mapped een key met een waarde die weergegeven moeten worden in het component.

Parameters:

key, de naam van de key

label, de naam die getoond moet worden in het component

Component: detail-person-edit

Laat de gebruiker de details van een persoon bewerken. Deze worden doorgegeven via een JSON object.

Tagnaam: x-detail-person-edit

Attributen

`data-names`

De namen van de values in de jsonfile gescheiden met een komma.

Default: empty array

`data-onsave`

De naam van de listener. Deze wordt aangeroepen als er op de save knop wordt gedrukt.

Default: undefined

`data-uncancel`

De naam van de listener. Deze wordt aangeroepen als er op de cancel knop wordt gedrukt.

Default: undefined

Events

`onCancelRequest(event)`

Wordt opgeroepen als de gebruiker op de annuleer knop drukt.

Parameters:

event, standaard event object.

`onSaveRequest(event)`

Wordt opgeroepen als de gebruiker op de opslaan knop drukt.

Parameters:

event, standaard event object met als extra attribute person. Als er voor een nieuwe foto is gekozen wordt deze al seen JavaScript File type megegeven.

Methodes

`setData(data)`

Laat de data zien in het component die wordt meegegeven.

Parameters:

data, het object dat een person representeert.

`clear()`

Verwijderd alle data uit het component

Component: detail-working

Toont werktijden per week in een grafiek uit een JSON bestand.

Tagnaam: x-detail-working

Methodes

`setData(data)`

Tekent de grafiek met de data die is meegegeven.

Parameters:

data, het object dat de werktijden representeert.

`setName(name)`

Toont de meegegeven naam in het component.

Parameters:

name, de naam die getoond moet worden.

Afhankelijkheden

Flotr2.0

Component: video

Toont een video aan de gebruiker met eventuele titel en omschrijving.

Tagnaam: x-video

Attributen

data-source

De URL naar de video.

Default: undefined

data-title

De titel van de video

Default: undefined

data-description

De omschrijving van de video

Default: undefined

data-autoplay

Boolean. Geeft aan of de video meteen moet afspelen.

Default: false

data-controls

Boolean. Geeft aan of de controls (play, pause, muted, timeline) getoond moeten worden.

Default: false

data-loop

Boolean. Geeft aan of de video moet lopen(true) of niet(false).

Default: false

data-muted

Boolean. Geeft aan of volume van de video gedempt is(true) of niet(false)

Default: false

Methodes

`setSource(src)`

Laad de meegegeven URL in de videospeler.

Parameters

src. De URL naar de af te spelen video.

`setTitle(title)`

Toont de titel die wordt meegegeven.

Parameters

title. De titel van de video.

`setDescription(descr)`

Toont de omschrijving die wordt meegegeven.

Parameters

descr. De omschrijving van de video.

`play()`

Speelt de video af. Hervat als deze gepauseerd is.

`pause()`

Pauzeert de video.

`setAutoplay(value)`

Geeft aan of de video meteen afgespeeld moet worden of moet wachten tot de gebruiker dit aangeeft.

Parameters

value. True of false.

`setControls(value)`

Geeft aan of de controls van de video getoond moeten worden of niet.

Parameters

value, true of false.

`setLoop(value)`

Geeft aan of de video in een loop moet afspelen of niet.

Parameters

value, true of false.

`setMuted(value)`

Geeft aan of de video gedempt moet afspelen of niet

Parameters

value, true of false.

Javascript: tass-helper.js

Dit bestand heeft de functionaliteit om van pagina's te wisselen.

Voorbeeld, een tass-helper.js object wordt als volgt aangemaakt:

```
var helper = new TassHelper();
```

Methodes

`goToPage(id)`

Navigeert naar de meegegeven pagina met een slide effect.

Parameters

id, het id van de pagina.

`enableBackSwipe(id)`

Zet de backswipe aan op de meegegeven pagina.

Parameters

id, het id van de pagina.

`disableBackSwipe(id)`

Zet de backswipe uit op de meegegeven pagina

Parameters

id, het id van de pagina.

Bijlage F

Onderzoeksverslag

Wat zijn HTML5, CSS3 en JavaScript?



Het W3C, World Wide Web Consortium, is het instituut die gaat over de HTML en CSS standaarden. Dit instituut is opgericht door Tim Berners-Lee, de bedenker van het World Wide Web.

Op de website van W3C wordt de volgende uitleg gegeven over HTML:

This specification defines the 5th major revision of the core language of the World Wide Web: the Hypertext Markup Language (HTML). In this version, new features are introduced to help Web application authors, new elements are introduced based on research into prevailing authoring practices, and special attention has been given to defining clear conformance criteria for user agents in an effort to improve interoperability.

HTML5 is dus simpelweg de volgende versie van HTML, waar nieuwe functionaliteit aan toegevoegd is.

HTML is al jarenlang het meest gebruikte middel om de opmaak van documenten mee te regelen. Met de nieuwe versie van HTML wordt meer gekeken naar wat HTML zou kunnen betekenen voor web applicaties i.p.v. alleen maar de opmaak van documenten.

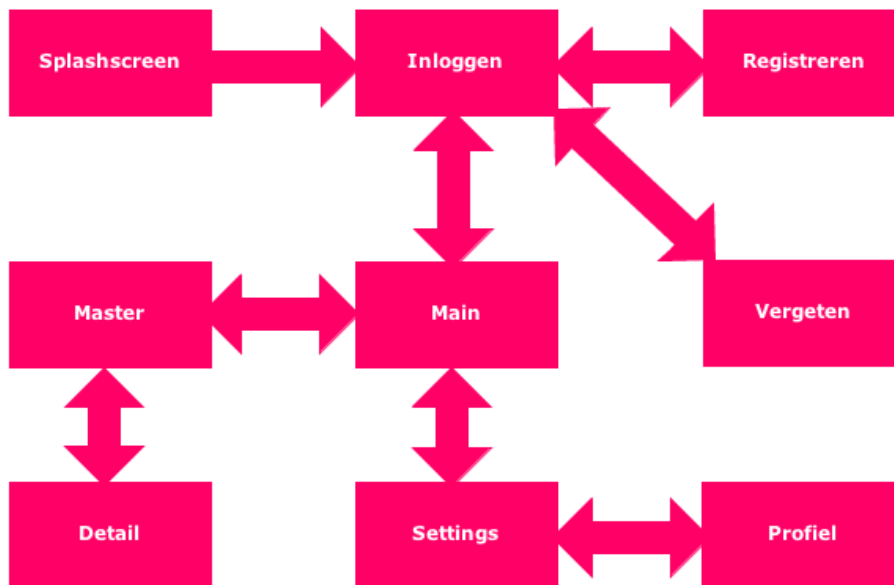
CSS3 is de nieuwe versie van CSS dat staat voor Cascading Style Sheet en is een techniek om een stijl toe te voegen aan een document bijvoorbeeld lettertypes en kleuren.

JavaScript is een programmeertaal die gebruikt wordt om websites dynamisch te kunnen maken. Het is mogelijk om object georiënteerd met te programmeren in JavaScript. Dit wordt op basis van prototypes gedaan i.p.v. classes. Dit verschil is belangrijk omdat veel ontwikkelaars voor iOS en Android gewend zijn te programmeren in object georiënteerde talen met classes. Het is belangrijk dat de ontwikkelaar van dit verschil afweet.

De combinatie van HTML5, CSS3 en JavaScript in de vorm van een app wordt vaak een HTML5 app genoemd. HTML, CSS en JavaScript zijn zulke belangrijke standaarden omdat deze gebruikt worden door browsers, die ieder populair besturingssysteem gebruikt.

Wat voor apps worden er ontwikkeld door TASS?

Aan het begin van het project van het project is er een met de opdrachtgever om de tafel gezeten om antwoord te krijgen op deze vraag. Er is toen een diagram gemaakt die hieronder te zien is.



Uit dit schema kan geconcludeerd worden dat TASS apps maakt waar data in te zien is en te bewerken. Ook is TASS altijd bezig om te kijken wat nieuwe technologieën kunnen betekenen voor app ontwikkeling. Om mobiele apparaten zijn dit bijvoorbeeld functies zoals:

- NFC, Near Field Communication. Hiermee kunnen smartphone en andere apparaten die NFC ondersteuning hebben vanaf een erg kleine afstand, niet meer dan een paar centimeter, een kleine hoeveelheid data uitwisselen.
- Wi-Fi Direct. Communicaties tussen smartphones via Wi-Fi.
- Locatie. Deze kan bepaald worden door GPS en Wi-Fi.
- Accelerometer. Zorgt ervoor dat de bewegingen van de smartphone in alle drie de richtingen gemeten kan worden.
- Camera. Voor het maken van foto's en video's. Maar zou ook voor geavanceerdere doeleinden kunnen worden gebruikt zoals gezichtsherkenning.

Veel van deze functies kunnen al via HTML5 benaderd worden. Als dit (nog) niet kan biedt PhoneGap altijd de mogelijkheid om een native plug-in hiervoor te gebruiken.

Tijdens het gesprek met de opdrachtgever is er ook aangegeven dat TASS gebruikt maakt van apps voor het aansturen van apparaten. Hier wordt dan vaak data verstuurd naar een ander apparaat als de gebruiker een actie onderneemt.

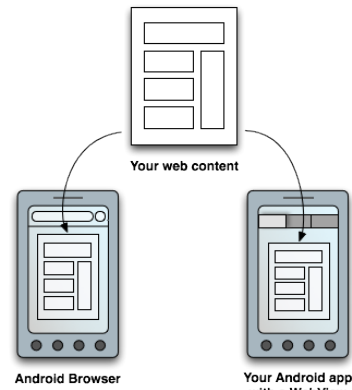
Op de planning staat een app waar de medewerkers van TASS hun uren in kunnen bijhouden. Ook zijn er plannen om de SharePoint omgeving naar de mobiele platformen te halen. Met het gebruik van NFC wil TASS in de toekomst een beveiligingssysteem ontwikkelen waarmee je het gebouw binnen komt.

Welke besturingssystemen ondersteunen HTML5?

Op de officiële website voor ontwikkelaars van Android staat een artikel over web applicaties. Hierin wordt verteld dat er twee manieren zijn om de content van de web applicatie aan de gebruiker op te dienen.

1. Via de Android browser.
2. Door een WebView in een Android app te implementeren.

De WebView die gebruikt wordt is dezelfde als die de standaard browser van Android gebruikt. Dit is niet Chrome. Chrome ondersteunt meer HTML5 functionaliteiten dan de standaard browser.



Tijdens het Google evenement: Google I/O 2012 is er aangekondigd dat Chrome binnenkort gebruikt zal gaan worden voor de WebView in Android. Wanneer dit dus wordt geïmplementeerd worden er meer HTML5 functies ondersteund.

De officiële website van Apple gaat niet specifiek in op het ontwikkelen van apps in HTML5. Wel wordt gezegd dat de standaarden ondersteunt worden in de webbrowser Safari:

Every new Apple mobile device and every new Mac — along with the latest version of Apple's Safari web browser — supports web standards including HTML5, CSS3, and JavaScript.

In iOS wordt het UI component UIWebView gebruikt voor het tonen van HTML bestanden, deze is gelijk aan de Safari browser.

Op de website html5test.com staat een overzicht van iedere browser die zij getest hebben op de HTML5 functionaliteiten. Voor iedere functionaliteit kan een browser punten scoren, maximaal 500 punten.

Android scoort een score van 297 met de standaard browser, maar met Chrome wordt een hogere score gehaald, 390. iOS haalt een score van 386.

Zijn er al apps die geschreven zijn in HTML5 en hoe worden deze ervaren?

Op de website van PhoneGap zijn apps genoemd die geschreven zijn met gebruik van PhoneGap. Er is een analyse gedaan op vijf van deze apps, de zogenaamde featured apps van PhoneGap. Hieronder staan de apps opgesomd met de resultaten:

1. The Panasonic World Heritage Calendar

De app heeft vrij weinig functionaliteit, er worden foto's weergegeven en er kan tekst gelezen worden die bij de foto hoort. De app is gemaakt door het bedrijf BluSoft Inc's, zij hebben laten weten aan PhoneGap dat zij niet alleen in een recordtijd de app af wisten te krijgen maar dat de app ook veel makkelijker te onderhouden was:

"Not only could we go from prototyping to release in record time but maintenance time for future releases was also reduced because of the shared codebase."

In Japan heeft deze app tweede gestaan in de App Store van Apple in de lijst van 'Top Free Apps'.

In de Android Play Store is de app minder populair met maar 100 tot 500 downloads.

2. Completure

Deze app toont kleine verhaaltjes die de gebruikers in elkaar zetten en gebruikt hierbij de functionaliteit van de camera en locatie.

De app is alleen maar in de App Store van Apple beschikbaar en heeft de een uitstekende beoordeling, 5 sterren.

3. HealthTap

In deze app kunnen gebruikers dokters in de buurt om advies vragen bij medische problemen.

De app staat in de Android Play Store en de App Store van Apple. In beide gevallen scoort de app erg hoog met 4 en een halve ster.

4. Untappd

Deze app laat gebruikers berichten en foto's delen via zijn eigen sociale netwerk maar ook via Twitter, Facebook en Foursquare.

De app staat in de App Store en wordt beoordeeld met vijf sterren, in de Play Store wordt de app iets minder, maar nog steeds goed, beoordeeld met 4,1 ster.

5. Wikipedia

PhoneGap is gebruikt om deze app te ontwikkelen omdat Wikipedia overal aanwezig moet zijn volgens de director of mobile and special projects van het bedrijf achter Wikipedia, Wikimedia. De app heeft meer dan 5 miljoen actieve Android gebruikers.

De app wordt goed beoordeeld in de Play Store met 4 en een halve ster. De iPhone gebruikers zijn er minder enthousiast over en geven de app maar 2 sterren.

Bovenstaande apps worden redelijk goed beoordeeld maar er kan ook besloten worden om HTML5 niet gebruiken. Facebook had zijn app ontwikkeld in HTML5 maar is hier op terug te komen omdat ze toch een betere gebruikerservaring willen bieden aan hun gebruikers. Zuckerberg, de baas van Facebook, zegt over HTML5:

"Het platform was er nog niet klaar voor."

"We moeten het hoogste kwaliteitsniveau bereiken en dat kan alleen maar als we native gaan."

Kan HTML5 dezelfde prestatie leveren als native?

Tijdens het realiseren van functionaliteit in HTML5 voor dit project is geconstateerd dat HTML5 vaak niet dezelfde performance kan bieden zoals native dit wel kan. Dit is tijdens acceptatietesten bij iedere sprint geconstateerd. Er is wel de conclusie getrokken dat, al zou TASS liever dezelfde performance willen als bij native, dat dit vaak niet erg hinderlijk is.

Zoals in het antwoord op de vorige deelvraag ook gezegd wordt heeft Facebook gekozen voor het ontwikkelen van zijn app in native om een betere performance te halen.

De platformen zijn nog niet klaar om performance te bieden zoals native dit doet. Steeds meer functionaliteiten van HTML5 worden geïmplementeerd in de browsers. Het is een kwestie van tijd totdat HTML5 vergelijkbare performance gaat bieden.