

Jouw mobiele universiteit!

Dré Hendriks s4130626
Lore Veugen s0815772
Simon Brugman s4151437

29 juni 2012

Inhoudsopgave

Voorwoord

Voor u ligt het eindverslag van de cursus Research en development 1 van het groepje CrApp. Behalve dit verslag zijn er nog andere documenten beschikbaar die meer inzicht bieden in het ontwikkelproces van onze Radboud Universiteits App, zoals een powerpointpresentatie (met visuele voorbeelden) en verschillende andere tekstdocumenten.

Tijdens het schrijven van dit verslag en de dagen die hiervoor volgden hebben we de effectiviteit van de app al zelf kunnen ervaren. Wanneer is vandaag het tentamen? Wat was ons cijfers voor R&D? De cijfers van Wiskunde 1b staan online, al 2 uur ná het college! We hopen ook over een tijdje dit soort verhalen ook van andere gebruikers te horen, en zijn persoonlijk erg blij met het resultaat.

In de rest van dit document kunt u lezen hoe de ontwikkeling van de app is verlopen, en toelichtingen waar nodig.

CrApp,
Dré Hendiks,
Lore Veugen,
Simon Brugman

Hoofdstuk 1

Beschrijving

1.1 Inleiding

Wat we zelf misten, en graag wilden hebben, hebben we gemaakt, namelijk: een Radboud universiteitsapp. Het idee hierachter is dat alles met betrekking tot je studie wat je snel wilt weten, makkelijk en snel te vinden is. Denk hierbij aan dingen zoals je rooster en je behaalde resultaten. Het belangrijkste aan onze app wordt dat het heel makkelijk in gebruik is en dat je alles binnen enkele drukken op je telefoon kunt vinden. We hebben voor dit onderwerp gekozen omdat we het zelf heel irritant vinden om op alle verschillende websites de hele tijd opnieuw in te loggen in de browser op je telefoon.

1.2 Productverantwoording

Wij geloven erin dat onze app een succes gaat worden. We hebben zelf al eerste versies van de app op onze telefoon staan en we gebruiken ze ook best geregeld. Ook denkt de management faculteit dat een dergelijk idee een goed idee zou zijn. Dat blijkt wel uit de posters waar ze de hele campus mee vol gehangen hebben. Niet alleen het idee van de app is goed, we hebben er denk ik ook een goed product van gemaakt.

De sleutel tot een succesvolle app is omschreven in 5 woorden:

1. Make it super useful; Dit is bij onze app zeker het geval. Het is heel fijn om even snel je rooster op te zoeken om te kijken waar je college hebt of een melding te krijgen als je tentamen is nagekeken.
2. Make it interactive; Onze app is super makkelijk in het gebruik en je kunt alle basis informatie vinden binnen 2 drukken op de knop. Als je wat meer informatie wilt. Zul je helaas nog een derde keer moeten klikken.
3. Make it entertaining; Dit is bij onze app minder het geval, maar onze app compenseert hierin doordat het super functioneel is, super gebruiksvriendelijk is. Daardoor is onze app heel efficiënt en tijdbesparend.
4. Make it a mixture; Onze app is vooral een mix van osiris en blackboard en andere bronnen. Dit bespaard veel tijd in het opzoeken.
5. Make it free; Onze app is helemaal gratis, lekker Hollands.

Vergelijkbare apps

Er zijn niet veel apps die hetzelfde idee hebben als die van ons, eigenlijk maar één. Dit is

een Universiteit Antwerpen app. Deze is echter slechts bedoelt om nieuws over de universiteit op te zoeken of highlights van de campus te leren kennen. Hij is dus niet geschikt voor studenten om hun gegevens in te zien. Kortom we hebben een heel origineel idee gehad. Voor de Radboud universiteit zelf is geen app beschikbaar. Onafhankelijk van de kwaliteit van de Antwerpen app zal onze doelgroep alleen een Radboud app willen, daar studeren ze immers.

1.3 Specificaties

1.3.1 Functionele eisen

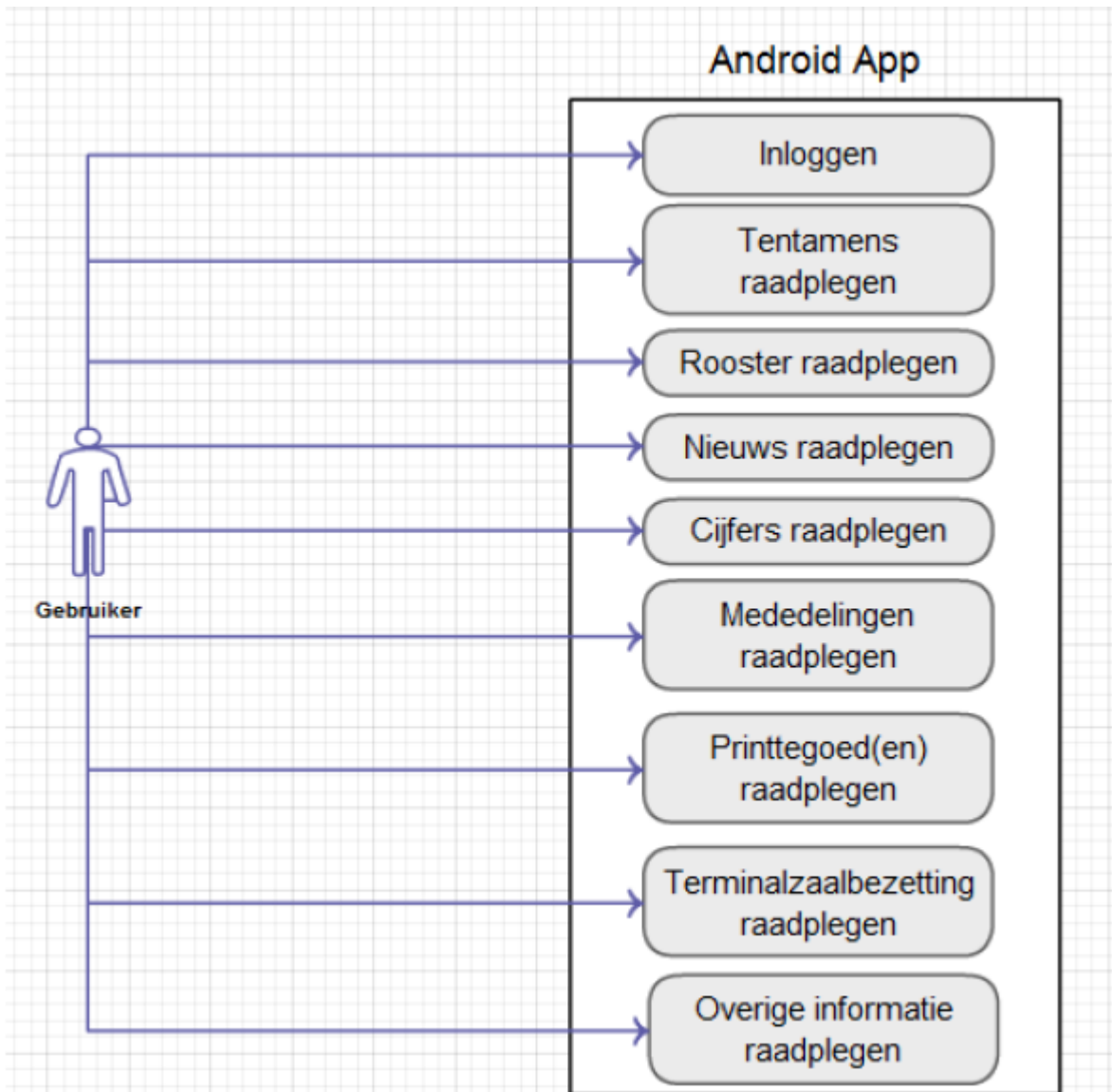
Met de app moet je kunnen:

- Kunnen inloggen
- Tentamenuitslagen kunnen raadplegen (Osiris)
- Het rooster kunnen bekijken
- Nieuws gerelateerd aan de RU te bekijken
- Resultaten van Blackboard bekijken
- Announcements van Blackboard bekijken
- Printtegoeden bekijken (Peage en Science/DHZ)
- Terminalkamerbezetting bekijken
- Informatie kunnen vinden over het gebruik (instructies, voorwaarden etc.)

1.3.2 Niet-Functionele eisen

- Gegevens binnen de app moeten veilig worden verstuurd
- De app moet altijd beschikbaar zijn (netwerk of geen netwerk)
- Laden van gegevens moet binnen redelijke termijn gebeuren
- Het gebruik van de app moet eenvoudig zijn en niet veel uitleg behoeven
- De app moet berekend zijn op meerdere gebruikers (1-1000)
- De app moet zo weinig mogelijk processorkracht van het android apparaat gebruiken
- De app moet commerciëel interessant zijn

1.3.3 Use Case Diagram



1.3.4 Uitgewerkte use case

We willen één use case laten zien omdat dit eigenlijk een typische use case is. Het toont gegevens die in de achtergrond worden opgehaald, over het hoe en wat wordt in de use case (nog) niet gesproken.

UC#	UC03
Naam	Rooster Raadplegen
Omschrijving	Deze Use case stelt de gebruiker in staat zijn rooster te raadplegen
Primaire actor	Student (Gebruiker)
Trigger	De gebruiker geeft aan zijn rooster te willen raadplegen
BCoE	<ol style="list-style-type: none"> 1. De gebruiker selecteert “rooster raadplegen” 2. De systeem toont het rooster van vandaag
Alternate paths	2a. Het systeem toont geen rooster als het weekend is
Alt Path	<ol style="list-style-type: none"> 2b. De gebruiker geeft aan het rooster van een andere datum dan die van vandaag te willen bekijken. 3b. Het systeem toont het rooster van de geselecteerde datum.
Pre-conditions	De gebruiker is ingelogd
Post-conditions	-
Business Rules	1. De gebruiker heeft een “persoonlijkrooster.ru.nl’-account
Opmerkingen	-

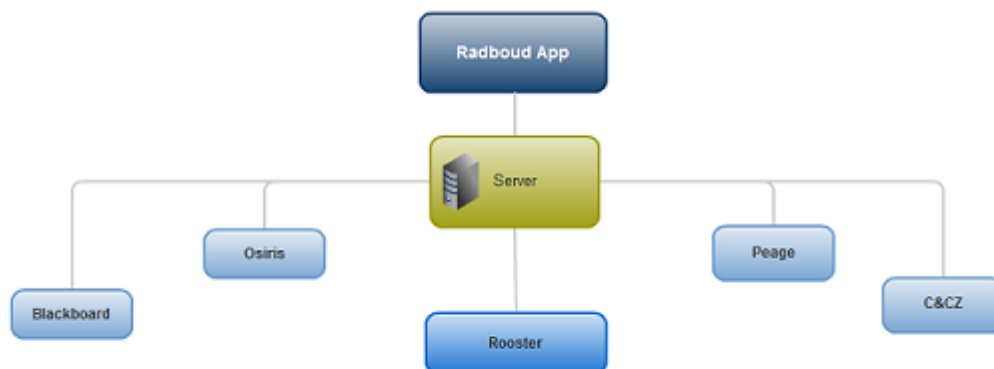
Hoofdstuk 2

Ontwerp

2.1 Globaal ontwerp

De app kan worden gezien als een aantal losse onderdelen, namelijk:

- Updater: het onderdeel dat verantwoordelijk is voor het verkrijgen van de juiste actuele data.
- Authenticator: het onderdeel dat verantwoordelijk is voor het koppelen van de juiste gebruiker aan de juiste data.
- User interface: het onderdeel dat de gegevens overzichtelijk weergeeft en ordent naar de onderdelen zoals hierboven aangegeven.



2.2 Detailontwerp

We zullen hier de uitgewerkte use case in detail bespreken. Wanneer de gebruiker aangeeft het rooster te willen bekijken, komt er een scherm met het rooster van die dag. Wanneer de dag vandaag of zaterdag of zondag is, krijgt de gebruiker een melding dat deze dat op een zondag valt. Bij het weergeven van het rooster wordt de naam van de cursus, de locatie en de tijden van het college getoond. Het kan natuurlijk zijn dat een gebruiker het rooster van eergister, morgen of volgende week wil bekijken. Dit kan door aan te geven welke datum de gebruiker

graag wil bekijken. In het uiteindelijke product is dit geïmplementeerd als een *swipe*; waar bewegen naar links staat voor een dag verder en bewegen naar rechts staat voor een dag terug. Ook kan er via het menu makkelijk van datum worden gewisseld.

2.3 Ontwerpverantwoording

Wat voor een gebruiker onder meer belangrijk is, is dat een app niet te veel batterij en niet veel bandbreedte gebruikt. Dit heeft een belangrijke rol gespeeld in ons ontwerp. Onze eerste opzet was dat de app direct communiceerde met de bronnen. Dit bleek in strijd te zijn met het hiervoor genoemde: er werd veel overbodige data opgehaald, de vele verschillende verbindingen zorgden voor veel verbruik van batterij en wachtwoorden moesten in de app zelf worden opgeslagen (wat niet erg veilig is). De oplossing voor deze problemen is geweest om er een server tussen te plaatsen. Deze server haalt alle gegevens op, filtert deze en maakt deze (in compacte vorm) beschikbaar aan de app. Zo wordt elke geheugen-intensieve operatie verplaatst naar de server, waardoor de app eigenlijk niks anders wordt dan een grafische gebruikersinterface.

De server, ookal valt deze buiten de focus van de app, zal ik kort toelichten: we gebruiken een server die draait met PHP en MySQL. De communicatie tussen de app en de server verloopt via XML. De server verzamelt alle gegevens van bronnen zoals Blackboard, Osiris, rooster, C&CZ en Peage en bewaart alleen de relevante gegevens. Wanneer de app controleert of er nieuwe gegevens zijn, zal de server de *update code* van de app vergelijken met die van het nieuwe bestand. Pas wanneer het bestand is vernieuwd, stuurt de server de gegevens door zodat de app deze kan overnemen. Het mooie van deze aanpak is dat alleen gegevens worden verstuurd wanneer dit echt nodig is, en zo minimaal mogelijk. Dit bespaart batterij en zorgt voor een lager verbruik van bandbreedte.

Hoofdstuk 3

Evaluatie

3.1 Testdoelen

- Intuïtief gebruik
- Bereikbaarheid van de functies
- Snelheid van het gebruik

3.2 Gebruikers(sub)groepen

- Bètastudenten aan de Radboud Universiteit Nijmegen
- Niet-bètastudenten aan de Radboud Universiteit Nijmegen

3.3 Scenario's

3.3.1 Rooster bekijken

Je zit op de fiets, op weg naar de campus. Helaas weet je alleen nog maar dat je college hebt, maar je weet niet meer in welk gebouw en/of welk lokaal. Je weet dus nog niet waar je naartoe aan het fietsen bent. Je wilt even snel je rooster zien om te kijken waar je college hebt. Hiervoor open je de app en kies je het rooster in het menu.

3.3.2 Cijfers raadplegen

Je staat op de gang te wachten tot je college gaat beginnen. Dan vertelt een van je medestudenten dat een ander vak is nagekeken en dat hij de uitslagen van het tentamen op osiris had zien staan. Je wilt graag kijken wat voor punt jij hebt gehaald voor dat vak.

3.4 Methoden voor gegevensverzameling

We hebben subjectieve gegevens verzameld via vragenlijsten en we hebben ook objectieve gegevens gemeten. Alvorens de echte test stelden we kort wat vragen over wat de verwachtingen zijn van de app. We hebben geprobeerd deze verwachtingen zo objectief mogelijk te observeren

en alles wat door ons te observeren was hebben we door middel van een vragenlijst na de test behandeld.

3.4.1 Pre-test

Als je een RU App in de market ziet staan, zonder verdere informatie, wat verwacht je dan dat hij kan?

Als je je rooster via een app kunt bekijken. In welke situatie verwacht je dat dan te gebruiken?

Hoeveel tijd zou het opzoeken van het rooster maximaal mogen kosten?

Als je je punten via een app kunt bekijken. In welke situatie verwacht je dit te gebruiken?

Hoeveel tijd zou het opzoeken van punten maximaal mogen kosten?

3.4.2 Test

Test bestaat uit 2 kleine testjes, het opzoeken van het rooster en het opzoeken van punten, daarna heeft de persoon nog even te tijd om rond te kijken. Hierbij wordt ook genoteerd hoe vlot beide tests worden uitgevoerd.

3.4.3 Post-test

We hebben je je rooster laten opzoeken. Geef een punt van 1 tot 10 over hoe soepel het ging. Denk hierbij aan hoeveel tijd het kostte en of je het meteen kon vinden.

Heb je verder nog opmerkingen over het opzoeken van je rooster?

We hebben je je punten laten opzoeken. Geef een punt van 1 tot 10 over hoe soepel het ging. Denk hierbij aan hoeveel tijd het kostte en of je het meteen kon vinden.

Heb je verder nog opmerkingen over het opzoeken van je punten?

Wat heb je in je vrije tijd? nog eens bekeken?

Heb je nog opmerkingen/bevindingen over dingen die je in je vrije tijd bent tegen gekomen. Denk hierbij aan dingen die je lastig vond.

Zou je onze app gebruiken als hij al in de market verkrijgbaar zou zijn?

Heb je verder nog opmerkingen over de app in zijn geheel.

3.4.4 opgeleverde data

3.5 Procedure

We hebben op de universiteit twee personen de app laten testen, op de manier die hierboven is beschreven. De app werd getest op de telefoon van Simon. Verder hebben we de antwoorden op de vragen zorgvuldig bekeken en serieus genomen.

3.6 Resultaten

3.6.1 Test 1

Pre-test

Als je een RU App in de market ziet staan, zonder verdere informatie, wat verwacht je dan dat hij kan? Dat zal dan vast een app zijn waar je informatie over de RU op kan vinden.

Als je je rooster via een app kunt bekijken. In welke situatie verwacht je dat dan te gebruiken? Als ik even niet weet wat voor les ik heb of als ik naar de uni toe moet.

Hoeveel tijd zou het opzoeken van het rooster maximaal mogen kosten? Als ik naar de uni toe moet ongeveer 30 seconden maar als ik het even snel wil checken wil ik het binnen 10 seconden weten.

Als je je punten via een app kunt bekijken. In welke situatie verwacht je dit te gebruiken? Als er net een toets is nagekeken?

Hoeveel tijd zou het opzoeken van punten maximaal mogen kosten? Ik denk 2 minuten ofzo, zolang het maar sneller is dan via de browser.

Test

Alvorens de test kwamen we erachter dat er een error zat in het inloggen. Dit hebben we dus niet in de test meegenomen. De testpersoon scrollde eerst naar beneden en daarna weer omhoog en klikte het rooster aan. De testpersoon klikte daarna gelijk de resultaten van osiris aan. In zijn vrije tijd is de persoon naar de informatie over de app gegaan. Dit zijn echter alleen maar dingen die we nog moeten invullen en hiermee is verder dus niet veel getoond.

Post-test

We hebben je je rooster laten opzoeken. Geef een punt van 1 tot 10 over hoe soepel het ging. Denk hierbij aan hoeveel tijd het kostte en of je het meteen kon vinden. 9

Heb je verder nog opmerkingen over het opzoeken van je rooster? Het stond nog net niet op de app toen ik hem opende maar wel de deur ernaast.

We hebben je je punten laten opzoeken. Geef een punt van 1 tot 10 over hoe soepel het ging. Denk hierbij aan hoeveel tijd het kostte en of je het meteen kon vinden. 9

Heb je verder nog opmerkingen over het opzoeken van je punten? -

Wat heb je in je vrije tijd? nog eens bekeken? Ik heb de informatie bekeken, maar dat was een onzin verhaaltje dus daar had ik niet zoveel aan.

Heb je nog opmerkingen/bevindingen over dingen die je in je vrije tijd bent tegen gekomen. Denk hierbij aan dingen die je lastig vond. -

Zou je onze app gebruiken als hij al in de market verkrijgbaar zou zijn? Waarschijnlijk wel.

Heb je verder nog opmerkingen over de app in zijn geheel. -

3.6.2 Test 2

Pre-test

Als je een RU App in de market ziet staan, zonder verdere informatie, wat verwacht je dan dat hij kan? Ik verwacht dat het een app is met informatie erin over de RU en dat je misschien ook je rooster kan opzoeken.

Als je je rooster via een app kunt bekijken. In welke situatie verwacht je dat dan te gebruiken? Altijd als ik mijn rooster moet checken en niet achter de computer zit, is makkelijker dan helemaal je laptop opstarten.

Hoeveel tijd zou het opzoeken van het rooster maximaal mogen kosten? Op mijn laptop kost het ongeveer 10 seconden maar ik zou al blij zijn met 20 seconden.

Als je je punten via een app kunt bekijken. In welke situatie verwacht je dit te gebruiken? Als iemand me heeft verteld dat een bepaald vak is nagekeken.

Hoeveel tijd zou het opzoeken van punten maximaal mogen kosten? Een minuut ofzo.

Test

In de test klikt hij gelijk het rooster aan en bekijkt het ook met de osiris resultaten, dit gaat gelijk goed. In zijn vrije tijd bekijkt hij de terminal beschikbaarheid. Na het laden van 2 seconden ongeveer staat hij erop en probeert hij er een aan te klikken voor meer informatie, dit is echter niet mogelijk.

Post-test

We hebben je je rooster laten opzoeken. Geef een punt van 1 tot 10 over hoe soepel het ging. Denk hierbij aan hoeveel tijd het kostte en of je het meteen kon vinden. 10

Heb je verder nog opmerkingen over het opzoeken van je rooster? Je hoeft het alleen maar aan te klikken.

We hebben je je punten laten opzoeken. Geef een punt van 1 tot 10 over hoe soepel het ging. Denk hierbij aan hoeveel tijd het kostte en of je het meteen kon vinden. 10

Heb je verder nog opmerkingen over het opzoeken van je punten? Je hoeft het alleen maar aan te klikken.

Wat heb je in je vrije tijd? nog eens bekeken? De terminal kamers, zag er chique uit.

Heb je nog opmerkingen/bevindingen over dingen die je in je vrije tijd bent tegen gekomen. Denk hierbij aan dingen die je lastig vond. -

Zou je onze app gebruiken als hij al in de market verkrijgbaar zou zijn? Ja, zeker weten!

Heb je verder nog opmerkingen over de app in zijn geheel. -

3.7 Conclusie

De basisfunctionaliteit werkt prima. De resultaten waren 9 en 10. Ook waren ze allebei ruim binnen de tijd die ze hadden gehoopt voor het checken van rooster en punten. Het verbeterpunt wat we nog hebben is het verder opvullen van de app. We moeten nog even wat stukjes schrijven van over de app e.d.. Ook is het misschien mooi om zodra je de app opent een pagina standaard te openen (bijvoorbeeld rooster).

Hoofdstuk 4

Reflectie

4.1 Groep

We hebben als groep de samenwerking als prettig ervaren. Dit noemen we expres als eerste omdat dit volgens ons het belangrijkste is. Als de samenwerking namelijk slecht is of zelfs helemaal ontbreekt zal het project als geheel ook floppen. Wanneer er iemand een drukke periode had dan werd het werk voor een deel door de anderen opgevangen zodat die persoon een stukje meer ruimte kreeg voor andere dingen. Maar we hebben ook veel dingen over het proces zelf en binnen het programmeren geleerd. Zo was het voor ons een hele frisse blik op de zaak dat we nu een app maakten en dat gebruiksvriendelijkheid als belangrijkste peiler overeind bleef staan. Ook hebben we geleerd dat wat jij logisch vindt niet altijd logisch gevonden wordt door de omgeving waar je in het werken bent. Zo heeft de android omgeving een heleboel frustraties opgeleverd. Je zou het zomaar kunnen vergelijken met het goed krijgen van de opmaak van een MS Office Word document.

4.2 Individueel

4.2.1 Dré

Als informatiekundige persoon in onze groep heb ik (gelukkig) niet zo veel te maken gehad met de daadwerkelijke code van het product. Dit is niet alleen gelukkig voor mij, omdat ik er niet zo van houdt en omdat dit waarschijnlijk ook geen goede resultaten zou opleveren voor het product. Ik heb wel geoefend in het presenteren. Dit is voor mij wel van groot belang, ik heb namelijk zelf wel het idee dat ik hier nog wel het een en ander aan kan verbeteren. Ook heb ik leren werken met LaTeX en dat gaat nu zelfs best soepel. Voor de uitleg heb ik al geprobeerd er wat mee te spelen, dit lukte frustrerend genoeg echter helemaal niet. Maar het belangrijkste wat ik geleerd heb is denk ik een stukje teamwork. Het natuurlijk nooit mis om samen met anderen ergens aan te werken. Het geeft toch een iets realistischer beeld van hoe het later in de boze buitenwereld zal gaan, daar kun je ook niet alles in je eentje doen en zul je zelfs de informatiekundige dingen ook nog met een team moeten zien te voltooien op een efficiënte manier.

4.2.2 Lore

4.2.3 Simon

Werken met android was compleet nieuw voor mij. Vooral de hele opbouw van een app is vernieuwend. Het werken van intents, broadcasts en ander soort zaken had ik snel onder de knie, maar het opstellen van een veilig en compact communicatieprotocol tussen de server en app was een erg intensief proces. Ik heb veel werk gestoken in de app, om te zorgen dat het gewoon een goed product is (mede omdat we hem zelf gebruiken). Veel problemen worden pas zichtbaar nadat de code is geïmplementeerd én de app in gebruik is. Android probeert bijvoorbeeld zoveel mogelijk services uit te zetten als het lijkt dat deze te veel geheugen gebruiken, of niet meer nodig zijn. Hierdoor worden gegevens niet meer up-to-date gehouden. Om dit op te lossen moest de hele code van het onderdeel údatergrotendeels worden aangepast, wat de nodige vertraging heeft opgeleverd. Dit soort problemen zijn we (veel te) vaak tegengekomen. Concluderend vind ik het een geslaagd en leerzaam project, door trail-and-error ben ik veel zaken tegengekomen die ik in het vervolg direct zal inzien.