

# MindYourOwnBusiness

Global Coffin Enterprise

Daniël van Loon  
Richard Lieverse  
Ash Tokbay

# Inhoudsopgave

<b>1</b>	<b>Beschrijving</b>	<b>3</b>
1.1	Inleiding . . . . .	3
1.2	Productverantwoording . . . . .	3
1.3	Specificaties . . . . .	4
1.4	Functional Requirements . . . . .	5
1.5	Non-functional Requirements . . . . .	5
<b>2</b>	<b>Ontwerp</b>	<b>6</b>
2.1	Globaal Ontwerp . . . . .	6
2.2	Data ontwerp . . . . .	13
2.3	Detailontwerp . . . . .	14
2.4	Ontwerpverantwoording . . . . .	17
<b>3</b>	<b>Evaluatie</b>	<b>19</b>
3.1	Testdoelen . . . . .	19
3.2	Gebruikers(sub)groepen . . . . .	19
3.3	Scenario's . . . . .	19
3.4	Methodes en procedure . . . . .	21
3.5	Resultaten . . . . .	21
3.6	Conclusies . . . . .	24
<b>4</b>	<b>Reflectie</b>	<b>25</b>
4.1	Groep . . . . .	25
4.2	Richard Lieverse . . . . .	25
4.3	Daniël van Loon . . . . .	26
4.4	Aslı Tokbay . . . . .	26

# 1 Beschrijving

## 1.1 Inleiding

De rol van de gebruiker is de rol van een ondernemer. Het doel is om een succesvolle zakenman/vrouw te worden met behulp van succesvolle bedrijven of aandelen in andere bedrijven. Het is mogelijk om met collega's samen te werken en het is tevens ook mogelijk om ze tegen te werken indien gewenst. Dit samenwerkingsverband gaat met behulp van Facebook. De economische simulatie zal dagelijks rond middernacht plaats vinden.

## 1.2 Productverantwoording

### **Entertainment factor**

Elke handeling in de applicatie is geschikt om in beperkte tijd uit te voeren. Het is interactief door middel van de toevoeging van vrienden op Facebook. Het betrekken van vrienden werkt motiverend bij het opbouwen van een succesvol bedrijf.

### **Utility factor**

Indien de gebruiker een langere periode zal spelen, zal hij verbanden gaan zien en zo het economisch model beter begrijpen dat gebaseerd is op de echte wereld.

### **Usability factor**

Door middel van kleinschalig onderzoek bij onze doelgroep is gebleken dat onze applicatie gebruiksvriendelijk is.

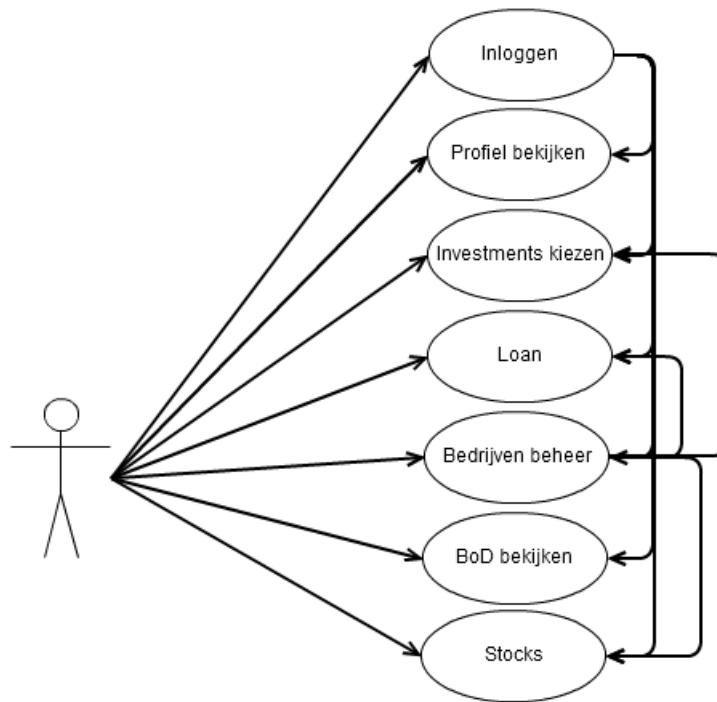
### **Functionality factor**

De applicatie maakt gebruik van automatische login met je Facebook-account via de Facebook-applicatie voor Android.

### **Maintenance factor**

Er is ruimte voor uitbreiding en deze is simpel te implementeren.

### 1.3 Specificaties



Figuur 1: UCD

## 1.4 Functional Requirements

### **Profiel bekijken:**

De use case geeft alle statistieken weer die van toepassing zijn op de huidige gebruiker.

### **Investments kiezen:**

De speler kan hier beslissen of hij in bepaalde onderdelen van het bedrijf wil investeren.

### **Bedrijven beheer:**

De speler kan hier een bepaald bedrijf selecteren waar hij aandelen van bezit of een nieuw bedrijf starten.

### **Loan:**

De speler kan hier een lening afsluiten.

### **Board of Directors bekijken:**

De speler kan hier zien hoeveel procent van de aandelen hij bezit van het bedrijf en kan hier stemmen om verschillende voorstellen goed of af te keuren.

### **Stocks:**

Spelers kunnen hier aandelen kopen en verkopen.

### **Inloggen:**

Hier kan de speler inloggen.

## 1.5 Non-functional Requirements

Voor onze applicatie zijn de belangrijkste Non-functional Requirements:

### **Scaleabilty:**

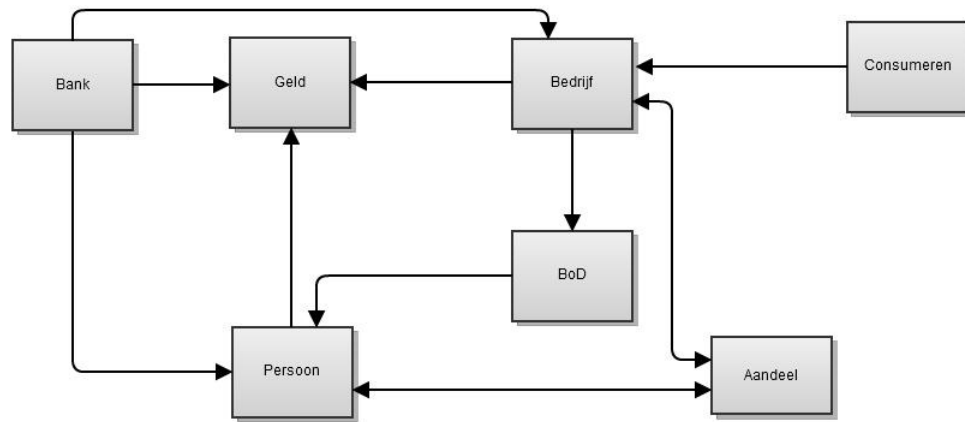
Het gebouwde systeem moet moeiteloos kunnen werken met tien gebruikers, maar ook met vijfhonderd of drieduizend gebruikers. Het systeem hoeft niet aangepast te worden als de hoeveelheid gebruikers stijgt, zelfs niet als dit bijvoorbeeld 100 keer zoveel zou worden.

### **Availability:**

Het gebouwde systeem moet altijd toegankelijke zijn voor gebruikers. Het zal dus 24 uur per dag beschikbaar moeten zijn.

## 2 Ontwerp

### 2.1 Globaal Ontwerp



Figuur 2: Globaal Ontwerp

<b>Nummer</b>	1
<b>Naam</b>	Profiel bekijken
<b>Omschrijving</b>	Laat de speler zijn gegevens zien.
<b>Primaire Actor</b>	Speler.
<b>Trigger</b>	De use case wordt getriggerd als de speler zijn profiel wil bekijken.
<b>Basic Flow</b>	
<b>Stap</b>	<b>Actie</b>
1	Het systeem laat het profiel zien.
<b>Alternative Flows</b>	
De speler heeft geen profiel.	
2	Het systeem maakt een profiel aan.
<b>Exception Flows</b>	
-	
<b>Precondities</b>	Het systeem heeft toegang tot internet.
<b>Postcondities</b>	Het systeem laat het profiel van de speler zien.
<b>Business rules</b>	-

<b>Nummer</b>	2
<b>Naam</b>	Investments kiezen
<b>Omschrijving</b>	Speler kan hier beslissen of hij in bepaalde onderdelen van het bedrijf wil investeren.
<b>Primaire Actor</b>	Speler.
<b>Trigger</b>	De use case wordt getriggerd als de speler naar investments gaat.
<b>Basic Flow</b>	
<b>Stap</b>	<b>Actie</b>
1a	Het systeem vraagt om het bedrag dat de speler wil investeren in Property.
1b	Het systeem vraagt om het bedrag die de speler wil investeren in Research.
1c	Het systeem vraagt om het bedrag die de speler wil investeren in Human Resources.
1d	Het systeem vraagt om het bedrag die de speler wil investeren in Marketing.
2a	De speler geeft het bedrag dat hij in Property wil investeren.
2b	De speler geeft het bedrag dat hij in Research wil investeren.
2c	De speler geeft het bedrag dat hij in Human Resources wil investeren.
2d	De speler geeft het bedrag dat hij in Marketing wil investeren.
<b>Alternative Flows</b>	
-	
<b>Exception Flows</b>	
3	De speler geeft ongeldige invoer. Het systeem geeft een waarschuwing.
<b>Precondities</b>	Het systeem heeft toegang tot internet. De speler moet aandelen hebben in het bedrijf.
<b>Postcondities</b>	De speler heeft een investeringsvoorstel gedaan. Alle aandeelhouders worden op de hoogte gebracht dat er een voorstel is.
<b>Business rules</b>	-



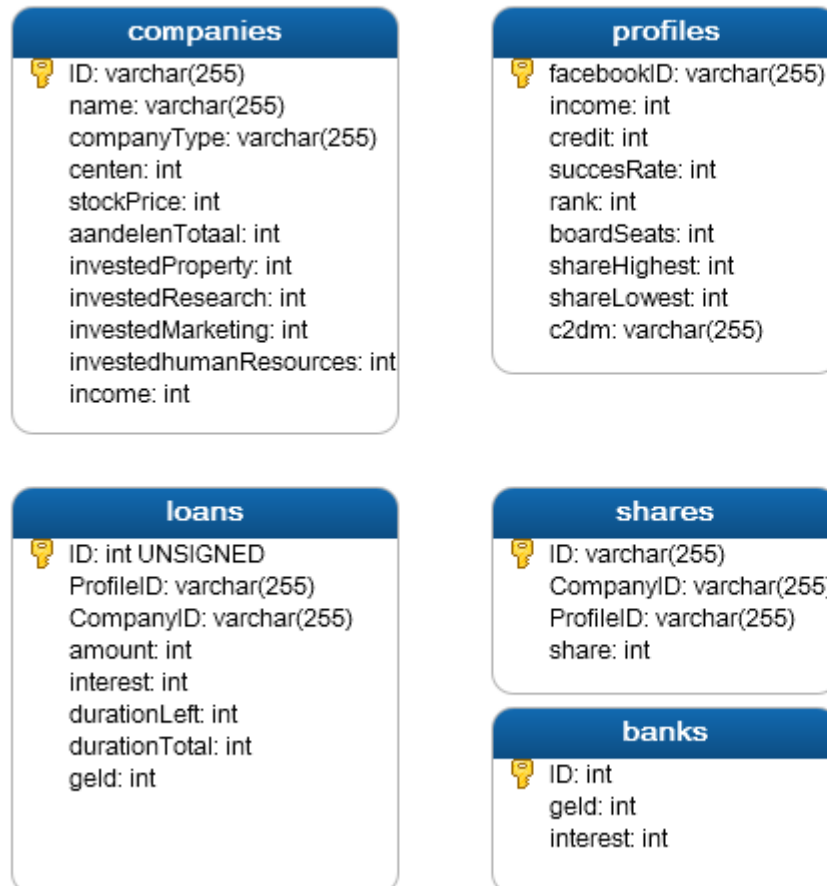
<b>Nummer</b>	3
<b>Naam</b>	Bedrijvenbeheer
<b>Omschrijving</b>	De speler kan hier een bepaald bedrijf selecteren waar hij aandelen van bezit. Hij kan hier ook een nieuw bedrijf beginnen.
<b>Primaire Actor</b>	Speler.
<b>Trigger</b>	De use case wordt getriggerd als de speler Bedrijven selecteert.
<b>Basic Flow</b>	
<b>Stap</b>	<b>Actie</b>
1	Het systeem laat alle bedrijven waar de speler een aandeel van bezit zien.
2	De speler selecteert een bedrijf.
3	Het systeem laat de bedrijfsresultaten zien.
4	Het systeem laat alle opties zien.
5	De speler selecteert de optie Board of directors.
<b>Alternative Flow</b>	
6	De speler selecteert de optie Loan.
7	De speler selecteert de optie Investments.
8	De speler selecteert de optie Stocks voor zijn bedrijf.
9	De speler selecteert de optie Stocks voor zichzelf.
<b>Exception Flow</b>	
10	De speler heeft geen aandelen in zijn bezit.
<b>Precondities</b>	Het systeem heeft toegang tot internet.
<b>Postcondities</b>	De speler heeft een optie geselecteerd en deze zal worden gestart door het systeem.
<b>Business rules</b>	-

<b>Nummer</b>	4
<b>Naam</b>	Loan
<b>Omschrijving</b>	De speler kan hier een lening afsluiten.
<b>Primaire Actor</b>	Speler.
<b>Trigger</b>	De speler kiest de optie My Loans in het hoofdmenu. De speler kiest optie Loan in bedrijven.
<b>Basic Flow</b>	
<b>Stap</b>	<b>Actie</b>
1	Het systeem laat het maximaal te lenen bedrag zien.
2	Het systeem vraagt het te lenen bedrag.
3	De speler geeft het te lenen bedrag aan.
<b>Alternative Flows</b>	
Maximaal te lenen bedrag is 0	
4	Het systeem zegt dat de speler niet meer kan lenen.
5	De speler kiest de optie Buy money.
<b>Exception Flows</b>	
6	De speler geeft onjuiste invoer.
<b>Precondities</b>	Het systeem heeft toegang tot internet.
<b>Postcondities</b>	De speler heeft een lening afgesloten voor zijn bedrijf of voor zichzelf.
<b>Business rules</b>	De rente voor de afgesloten lening blijft constant.

<b>Nummer</b>	5
<b>Naam</b>	Board of Directors bekijken
<b>Omschrijving</b>	De speler kan hier zien hoeveel procent van de aandelen hij bezit van het bedrijf en kan een speler uitnodigen.
<b>Primaire Actor</b>	Speler.
<b>Trigger</b>	De speler kiest de optie Board of Directors.
<b>Basic Flow</b>	
<b>Stap</b>	<b>Actie</b>
1	Het systeem laat de aandeelhouders zien met de bijbehorende percentages.
2	Het systeem laat het voorstel zien en vraagt om een antwoord.
3	De speler geeft een antwoord.
<b>Alternative Flows</b>	
4	De speler selecteert een aandeelhouder met het verzoek om hem uit te nodigen.
<b>Exception Flow</b>	
	-
<b>Precondities</b>	Het systeem heeft toegang tot internet.
<b>Postcondities</b>	-
<b>Business rules</b>	-

<b>Nummer</b>	6
<b>Naam</b>	Inloggen
<b>Omschrijving</b>	Hier kan de applicatie of speler inloggen.
<b>Primaire Actor</b>	Speler en applicatie.
<b>Trigger</b>	Opstarten applicatie.
<b>Basic Flow</b>	
<b>Stap</b>	<b>Actie</b>
1	Systeem kijkt of de telefoon reeds een geldige Facebook autorisatie heeft.
<b>Alternative Flows</b>	
Geen geldige Facebook autorisatie gevonden	
2	Het externe Facebook systeem vraagt om e-mail en wachtwoord.
3	De gebruiker geeft e-mail en wachtwoord.
<b>Exception Flow</b>	
4	De speler geeft een niet bestaande combinatie van e-mail en wachtwoord.
5	Het systeem vraagt opnieuw om een gebruikersnaam en wachtwoord.
<b>Precondities</b>	Het systeem heeft toegang tot internet. De speler heeft een Facebook account.
<b>Postcondities</b>	De speler is ingelogd en heeft toegang tot andere functies.
<b>Business rules</b>	-

## 2.2 Data ontwerp



Figuur 3: Data Ontwerp

## 2.3 Detailontwerp



Company Object	Invest Object	FriendList Object			
<ul style="list-style-type: none"> <li>- UUID ID</li> <li>- String name</li> <li>- String companyType</li> <li>- Geld geld</li> <li>- int stockPrice</li> <li>- int aandelenTotaal</li> <li>- int investedProperty</li> <li>- int investedResearch</li> <li>- int investedMarketing</li> <li>- int investedhumanResources</li> <li>- int income</li> <li>- LinkedList boardOfDirectors</li> </ul>	<ul style="list-style-type: none"> <li>- String companyID</li> <li>- String profileID</li> <li>- int property</li> <li>- int research</li> <li>- int humanResources</li> <li>- int marketing</li> </ul>	<ul style="list-style-type: none"> <li>+ LinkedList Friends</li> <li>+ String toTransfer()</li> </ul>			
<ul style="list-style-type: none"> <li>+ String toSqlInsert()</li> <li>+ String toSqlUpdate()</li> <li>+ String toSqlAppendCash(int, String)</li> <li>+ String toSqlSelect(String)</li> <li>+ String toSqlSelectCompanyName(String)</li> <li>+ String toSqlSelectFacebookID(String)</li> <li>+ String toSqlSelectFacebookIDAndCompanyID(String, String)</li> <li>+ String toSqlSelectAllShares(String)</li> <li>+ String toTransfer()</li> <li>- void init(String, LinkedList, CompanyType)</li> <li>+ boolean addOrEditDirector(Profile, int, int)</li> <li>+ LinkedList getBoardOfDirectors()</li> <li>- boolean aandelenBeschikbaar(int)</li> <li>+ int getAandelenBezeten()</li> <li>+ String getName()</li> <li>+ String getCompanyType()</li> <li>+ int getStockPrice()</li> <li>+ int getAandelenTotaal()</li> <li>+ int getInvestedProperty()</li> <li>- void addInvestedProperty(int)</li> <li>+ int getInvestedResearch()</li> <li>- void addInvestedResearch(int)</li> <li>+ int getInvestedMarketing()</li> <li>+ int getIncomeCenten()</li> <li>+ String getIncome()</li> <li>- void addInvestedMarketing(int)</li> <li>+ int getInvestedhumanResources()</li> <li>- void addInvestedhumanResources(int)</li> <li>+ String getID()</li> <li>+ Geld getGeld()</li> <li>+ void addInvestment(Invest)</li> </ul>	<ul style="list-style-type: none"> <li>+ String toTransfer()</li> <li>+ int getTotalInvestment()</li> <li>+ String getCompanyID()</li> <li>+ String getProfileID()</li> <li>+ int getProperty()</li> <li>+ int getResearch()</li> <li>+ int getHumanResources()</li> <li>+ int getMarketing()</li> </ul>	<table border="1"> <thead> <tr> <th>Friend Object</th> </tr> </thead> <tbody> <tr> <td> <ul style="list-style-type: none"> <li>- String facebookID</li> <li>- String name</li> </ul> </td> </tr> <tr> <td> <ul style="list-style-type: none"> <li>+ String toTransfer()</li> <li>+ String getFacebookID()</li> <li>+ String getName()</li> </ul> </td> </tr> </tbody> </table>	Friend Object	<ul style="list-style-type: none"> <li>- String facebookID</li> <li>- String name</li> </ul>	<ul style="list-style-type: none"> <li>+ String toTransfer()</li> <li>+ String getFacebookID()</li> <li>+ String getName()</li> </ul>
Friend Object					
<ul style="list-style-type: none"> <li>- String facebookID</li> <li>- String name</li> </ul>					
<ul style="list-style-type: none"> <li>+ String toTransfer()</li> <li>+ String getFacebookID()</li> <li>+ String getName()</li> </ul>					
	<table border="1"> <thead> <tr> <th>Geld Object</th> </tr> </thead> <tbody> <tr> <td> <ul style="list-style-type: none"> <li>- int centen</li> </ul> </td> </tr> <tr> <td> <ul style="list-style-type: none"> <li>+ String toString()</li> <li>- void put(int)</li> <li>+ boolean take(Geld, int)</li> <li>+ int getCenten()</li> </ul> </td> </tr> </tbody> </table>	Geld Object	<ul style="list-style-type: none"> <li>- int centen</li> </ul>	<ul style="list-style-type: none"> <li>+ String toString()</li> <li>- void put(int)</li> <li>+ boolean take(Geld, int)</li> <li>+ int getCenten()</li> </ul>	
Geld Object					
<ul style="list-style-type: none"> <li>- int centen</li> </ul>					
<ul style="list-style-type: none"> <li>+ String toString()</li> <li>- void put(int)</li> <li>+ boolean take(Geld, int)</li> <li>+ int getCenten()</li> </ul>					
	<table border="1"> <thead> <tr> <th>CompanyList Object</th> </tr> </thead> <tbody> <tr> <td> <ul style="list-style-type: none"> <li>+ LinkedList Companies</li> </ul> </td> </tr> </tbody> </table>	CompanyList Object	<ul style="list-style-type: none"> <li>+ LinkedList Companies</li> </ul>		
CompanyList Object					
<ul style="list-style-type: none"> <li>+ LinkedList Companies</li> </ul>					
		<table border="1"> <thead> <tr> <th>Loan Object</th> </tr> </thead> <tbody> <tr> <td> <ul style="list-style-type: none"> <li>- int ID</li> <li>- String profileID</li> <li>- String companyID</li> <li>- int amount</li> <li>- int interest</li> <li>- int durationLeft</li> <li>- int durationTotal</li> <li>- Geld geld</li> </ul> </td> </tr> <tr> <td> <ul style="list-style-type: none"> <li>+ String toTransfer()</li> <li>+ String toSqlInsert()</li> <li>+ String toSqlSelect(String, String)</li> <li>+ String toSqlSelectCompanyID(String)</li> <li>+ String toSqlSelectCount(String, String)</li> <li>+ String toSqlUpdate()</li> <li>+ void checkCicle(Geld)</li> <li>+ String getProfileID()</li> <li>+ String getCompanyID()</li> <li>+ int getAmount()</li> <li>+ int getInterest()</li> <li>+ int getDurationLeft()</li> <li>+ int getDurationTotal()</li> </ul> </td> </tr> </tbody> </table>	Loan Object	<ul style="list-style-type: none"> <li>- int ID</li> <li>- String profileID</li> <li>- String companyID</li> <li>- int amount</li> <li>- int interest</li> <li>- int durationLeft</li> <li>- int durationTotal</li> <li>- Geld geld</li> </ul>	<ul style="list-style-type: none"> <li>+ String toTransfer()</li> <li>+ String toSqlInsert()</li> <li>+ String toSqlSelect(String, String)</li> <li>+ String toSqlSelectCompanyID(String)</li> <li>+ String toSqlSelectCount(String, String)</li> <li>+ String toSqlUpdate()</li> <li>+ void checkCicle(Geld)</li> <li>+ String getProfileID()</li> <li>+ String getCompanyID()</li> <li>+ int getAmount()</li> <li>+ int getInterest()</li> <li>+ int getDurationLeft()</li> <li>+ int getDurationTotal()</li> </ul>
Loan Object					
<ul style="list-style-type: none"> <li>- int ID</li> <li>- String profileID</li> <li>- String companyID</li> <li>- int amount</li> <li>- int interest</li> <li>- int durationLeft</li> <li>- int durationTotal</li> <li>- Geld geld</li> </ul>					
<ul style="list-style-type: none"> <li>+ String toTransfer()</li> <li>+ String toSqlInsert()</li> <li>+ String toSqlSelect(String, String)</li> <li>+ String toSqlSelectCompanyID(String)</li> <li>+ String toSqlSelectCount(String, String)</li> <li>+ String toSqlUpdate()</li> <li>+ void checkCicle(Geld)</li> <li>+ String getProfileID()</li> <li>+ String getCompanyID()</li> <li>+ int getAmount()</li> <li>+ int getInterest()</li> <li>+ int getDurationLeft()</li> <li>+ int getDurationTotal()</li> </ul>					

Profile Object
- String facebookID - int income - int succesRate - int rank - int boardSeats - int shareHighest - int shareLowest - Geld geld - UUID GeldID
- void init(String[]) + String toSqlSelect(String) + String toSqlSelectC2dm(String) + String toSqlSelectFacebookID() + String toSqlSelectC2dm() + String toSqlInsert() + String toSqlUpdate() + String toSqlAppendCash(int, String) + String getUUID() + String getFacebookID() + int getIncomeCents() + String getIncome() + String toTransfer() + String getCredit() + Geld getGeld() + int getBoardseats() + int getSuccesRate() + int getRank() + int getShareHighest() + int getShareLowest()



## 2.4 Ontwerpverantwoording

Waarom is de login functie van Facebook geïmplementeerd?

De alternatieven:

### **Eigen login**

Deze mogelijkheid hebben we niet gekozen om twee redenen. Het is gebruiksvriendelijker om gebruikers een reeds bestaande account te laten gebruiken. Daarnaast is het zo dat het ons dubbel werk zou opleveren aangezien Facebook reeds een bestaande login functie beschikbaar heeft gesteld voor ontwikkelaars.

### **Hyves**

Hoewel deze mogelijkheid simpeler te integreren is dan een eigen login, is deze ook afgevallen. Dit vanwege het feit dat onze doelgroep toch veelal bestaat uit studenten en middelbare scholieren. Onze doelgroep maakt vrijwel geen gebruik van Hyves.

### **Twitter**

Deze is afgevallen omdat Facebook een grotere binding geeft met vrienden. Dit is van belang, omdat dit onze applicatie een extra dimensie geeft en een extra stimulans om de applicatie te blijven gebruiken.

### **Google**

De gebruiker heeft een Google-account nodig om de applicatie te kunnen downloaden, echter ook hierbij geldt dat de binding ontbreekt die bij Facebook wel aanwezig is.

### **Facebook**

Facebook had alle voordelen, behalve dat het niet nodig is voor de Play-store. Daarnaast heeft het veel actieve gebruikers binnen onze doelgroep.

## Data opslag methodes:

### Java serialization

Deze methode kan problemen opleveren door middel van compatibiliteitsproblemen. Het kan zijn dat bepaalde context deel uitmaakt van de Dalvik virtual machine van Android en deze dus niet kan herstellen in een andere omgeving.

Het is versie gevoelig, dus zowel de server als het Android apparaat moeten dezelfde versie hebben van dit onderdeel van Java.

De prestatie is belangrijk, omdat de gebruiker dit direct zal merken. Er zijn kleine tests gedaan die uitwijzen dat het eerste object gemiddeld 1 seconde op zich laat wachten. Deze resultaten zijn gebaseerd op 15 tests waarbij 1000 objecten worden weggeschreven.

### Java Json

Java Json is succesvol geïmplementeerd in Android vanaf API 11. Gezien onze doelgroep kunnen we niet aan deze eis voldoen. Volgens de laatste Google peilingen maakt ongeveer 80% nog steeds gebruik van api 8 tot 12.

### Eigen implimentatie

We kunnen selectief data overbrengen, zodat de accu minder belast gaat worden. Hierdoor treden er geen fouten op van onnodige functionaliteiten die het werk vervolgens verder vertragen.

## 3 Evaluatie

### 3.1 Testdoelen

#### **Het bekijken van het profiel**

Deze taak heeft betrekking op de integratie met Facebook en het bekijken van de statistieken van jouw online personage.

#### **Persoonlijke lening afsluiten**

Dit is essentieel in onze applicatie om een succesvol bedrijf te starten, zodat er investeringen gemaakt kunnen worden.

#### **Aanmaken van een Bedrijf in een bepaalde sector**

Het is een belangrijke factor dat het makkelijk is om een bedrijf aan te maken en te selecteren om hier acties op uit te voeren.

#### **Het uitnodigen van een Facebook relatie**

Het motiverende element om te blijven spelen moet ook zorgvuldig getest worden. Deze test moet uitwijzen of het lukt om een Facebook relatie in je bedrijf te introduceren.

### 3.2 Gebruikers(sub)groepen

Mensen die gevraagd zijn om de applicatie te testen zijn personen die zelf in het bezit zijn van een Android-toestel. De applicatie is gemaakt voor mensen die al overweg kunnen met Android. Ze moeten al vaker een touchscreen gezien en gebruikt hebben en de bijbehorende knoppen. Wat ook belangrijk is, is de al eerder opgedane ervaring van de testers. Door deze voorwaarden komen er nadrukkelijk de volgende doelgroepen naar voren: Studenten en middelbare scholieren.

### 3.3 Scenario's

#### **Taken:**

##### **Taak 1:**

Start het spel op en bekijk jouw profiel

##### **Taak 2:**

Leen een bedrag van 1000 euro van de bank voor jezelf.

##### **Taak 3:**

Maak een nieuw bedrijf met de naam Maïs in de sector Agriculter en investeer 600 euro in Research.

##### **Taak 4:**

Nodig AshTokbay uit voor je nieuwe bedrijf.

## Vragenlijst

**1** Was er een bepaalde taak die moeilijk uit te voeren was? Zo ja welke taak/taken?

.....

**2** Wat vindt u van de vormgeving van de app?

- Zeer slecht, want .....
- Slecht, want .....
- Neutraal
- Goed
- Zeer goed

**3** Wat vindt u van de reactiesnelheid van de app?

- Zeer langzaam
- Langzaam
- Neutraal
- Snel
- Zeer snel

**4** Zou u de app in de toekomst nog willen downloaden?

- Ja, ga naar vraag 5
- Nee, ga naar vraag 6

**5** Denkt u dat u de app voor een langere periode te gaan gebruiken?

- Ja
- Nee, want .....

**6** Waarom zou u de app niet downloaden?

.....

Bedankt voor uw medewerking en tijd.

### 3.4 Methodes en procedure

Tijdens de usability-test wordt er gebruik gemaakt van hard op denken met observatie. Dit wordt gedaan zodat het duidelijk wordt of de structuur van de applicatie goed in elkaar steekt. Als gebruikers op een verkeerde knop klikken kan de observator gelijk vragen waarom de gebruiker dit deed.

Nadat de taken zijn uitgevoerd moet de gebruiker een vragenlijst invullen. Dit om te zien wat de gebruiker in het algemeen van de applicatie vindt.

Alle gebruikers hebben zelf hun Android-toestel bij zich en het spel zal aan het begin van de test worden geïnstalleerd. Dit wordt zo gedaan zodat de gebruikers het spel tijdens de test voor het eerst zien. Tevens mogen ze ook niet te veel voorkennis hebben van hoe het spel werkt.

Er wordt kort uitgelegd wat het spel inhoud en wat de gebruikers moeten doen. Vervolgens zal de gebruiker een lijst te zien krijgen met alle taken daarop. Er zal tijdens de hele test een observator bij zijn. De observator moet ervoor zorgen dat de gebruiker blijft praten en hele tijd meldt of hij ergens niet uitkomt. De observator mag geen tips geven ook mag niet tegen de gebruiker gezegd worden of de taak klaar is, de gebruiker moet zelf beslissen of hij een taak afgerond heeft.

### 3.5 Resultaten

#### Pien Walraven

##### Taak 1

Nou het zal dan wel het profiel zijn en opent het.

##### Taak 2

Hier opent ze alles goed. Ze voert een bedrag in en leent het bedrag.

##### Taak 3

Ze klikt gelijk op Companies en Start New Company. Hier gaat iets fout met de applicatie, het nieuwe aangemaakte bedrijf staat niet bij My companies. Er wordt gezegd dat ze verder moet gaan met een andere bedrijf.

##### Taak 4

Investeren doet ze op een reeds bestaand bedrijf. Dit doet ze in één keer goed.

##### Taak 5

Vriend uitnodigen lukt ook. Ze mist de confirmatie.

##### Vraag 1

Ze geeft op het eind aan dat het moeilijkste van de taken het investeren was.

##### Vraag 2

Vormgeving wordt als neutraal beoordeeld.

##### Vraag 3

De snelheid van de app wordt beoordeeld als zeer snel.

##### Vraag 4

Ja, als de bugs eruit zijn.

##### Vraag:5

Ja, als hij werkt.

##### Opmerkingen

Als laatste geeft ze aan dat een tutorial wel handig zou zijn omdat het spel haar best ingewikkeld lijkt.

## **Tom Sanders**

### **Taak 1**

Na te hebben gezegd: Ik denk dat het Profile is, opent hij zijn profiel.

### **Taak 2**

Hij zegt: ik neem aan dat ik eerst terug moet voor taak 2. Hierna klikt hij op de terug knop en dan op My Loans. Hij klikt gelijk verder op new loan en sluit een nieuwe lening af met de gegeven gegevens.

### **Taak 3**

Bedrijf aanmaken ging goed.

### **Taak 4**

Investments duurt langer, omdat het niet helemaal duidelijk was hoe hij bij het bedrijfsmenu moest komen.

### **Taak 5**

Vriend toevoegen lukt in een keer.

### **Vraag 1**

Hij vond Investments moeilijk te vinden, omdat het bedrijfsmenu moeilijk te vinden was volgens hem. Hij vond het onduidelijk dat je op de bedrijfsnaam moest klikken om in het bedrijfsmenu te komen.

### **Vraag 2**

Vormgeving is goed.

### **Vraag 3**

Reactiesnelheid van de app is zeer goed indien internet.

### **Vraag 4**

Ja.

### **Vraag 5**

Als het leuk blijft.

## **Sjors Clabbers**

### **Taak 1**

Nou dat zal wel Profile zijn. Vervolgens opent hij het profiel.

### **Taak 2**

Hierna gaat hij terug naar het hoofdmenu en opent hij My Loans. Hij leent geld. Hij maakt ook een opmerking over dat je wel heel veel geld kan lenen.

### **Taak 3**

Vervolgens start hij een nieuwe bedrijf op. Dit doet hij in een keer goed.

### **Taak 4**

Investeren ging goed. Hij had geen verdere opmerkingen.

### **Taak 5**

Hij gaat vervolgens Ash uitnodigen voor zijn bedrijf. Hij stuurt heel veel uitnodigingen in plaats van eentje. Hij merkt op dat je een speler meerdere keren kan uitnodigen voor het zelfde bedrijf. Hij zegt ook dat het niet duidelijk is of een invite wel is verstuurd.

### **Vraag 1**

Hij zegt: dat geen een taak echt moeilijk was, maar dat als hij moest kiezen dat het dan een nieuw bedrijf opstarten was.

**Vraag 2**

Hij vindt de vormgeving neutraal.

**Vraag 3**

Hij vindt de reactiesnelheid goed.

**Vraag 4**

Ja.

**Vraag 5**

Ligt er aan of het spel interessant blijft.

**Opmerkingen**

Na de taken geeft hij aan dat een tutorial wel handig zou zijn.

Hij meldt ook dat de back knop beter zou zijn als je in het bedrijfsmenu zit in plaats van een knop companies.

**Ben Brücker**

**Taak 1**

Ik neem aan dat het Profile is. Hierna opent hij het profiel.

**Taak 2**

Persoonlijke lening gaat ook goed. Hij mist alleen een bevestiging dat de lening gelukt is.

**Taak 3**

Nieuw bedrijf opstarten lukt ook.

**Taak 4**

Bij het investeren geeft hij aan de standaard ingevulde waarde moet verdwijnen zodra de gebruiker iets wil invullen. Hij meldt ook dat niemand results zal lezen om naar zijn investments te kijken.

**Taak 5**

Uitnodigen gaat ook goed.

**Vraag 1**

De moeilijkste taak was lenen, omdat het niet duidelijk was of je lening echt was gelukt.

**Vraag 2**

Na de taken zegt hij dat de vormgeving wel leuk is, maar wat minder zwart, meer kleur en meer plaatjes mooier zouden zijn. Titelbalk aan de bovenkant van de hoofdmenu mag weg omdat hij wel weet welke applicatie er draait

**Vraag 3**

Reactiesnelheid is goed.

**Vraag 4**

Ja.

**Vraag 5**

Weet ik nog niet.

**Opmerkingen**

De knop companies kan beter een Back knop worden.

### **3.6 Conclusies**

Alle gebruikers zijn tevreden over de snelheid van onze applicatie.

Gebruikers willen een tutorial en als ze een wijziging doorgeven willen ze bevestiging dat het is gelukt.

De knop Companies die in het bedrijfshoofdmenu zit wordt door de gebruikers ook als onhandig gezien.



## 4 Reflectie

### 4.1 Groep

Wij zijn tevreden over de communicatie in de groep. Er is bij het begin gelijk afgesproken als er iets is, dat je het dan gelijk moet laten weten. Door gebruik te maken van Dropbox en SVN had iedereen toegang tot alle bestanden en dit heeft er ook voor gezorgd dat het samenwerken makkelijker werd. Iedereen was ook bijna altijd bereikbaar, dus als een groepslid een vraag/suggestie had kon de rest daar snel op reageren.

We hebben een aantal deadlines net gehaald en sommige ook niet.

Dit vonden wij erg vervelend. We denken dat dit onder andere komt doordat onze applicatie te groot is. We hadden aan het begin het idee dat het wel zou lukken om de applicatie af te krijgen, maar toen we het prototype af moesten hebben waren we nog niet zo ver.

Volgende keer zouden we eerst naar de hoeveelheid tijd die we hebben kijken en naar de hoeveelheid uren voordat we een project ontwerpen. Nu was er ook sprake van het feit dat Richard heel veel van het programmeerwerk heeft gedaan. Volgende keer gaan we eerst kijken naar hoeveel iedereen kan bijdragen.

### 4.2 Richard Lieverse

Als ik terug denk aan Research & Development 1 dan moet ik denken aan

1. Verantwoording van je product
2. Presenteren van je product
3. Realiseren van je product
4. Middelen om dat allemaal te bereiken

Persoonlijk ben ik gegroeid in de manier van denken over het ontwikkelproces van een product. Voordat je begint met de realisatie kun je eigenlijk al met redelijke zekerheid bepalen of de applicatie überhaupt wel een succes kan gaan worden.

Mijn programmeer kennis met betrekking tot Android is enorm vergroot. Presenteren is zo veel makkelijker, omdat je eigenlijk nu altijd met je product bezig bent en niet zo zeer met het proces. Voor mij is het dan ook makkelijker om te presenteren, omdat je immers sterk in je schoenen staat na redelijk veel onderzoek.

### 4.3 Daniël van Loon

**Inhoud:**

Voor deze cursus wist ik niet dat je ook kunt verantwoorden door alle keuzes te geven en aan te geven waarom je juist die ene keuze gemaakt hebt, dat vond ik dus een vrij grote openbaring. Wat mij ook weer opgevallen is, is dat ik liever werk aan een project waar iteratie voorkomt in plaats van dat er gewerkt wordt aan de hand van het waterval principe.

**Persoonlijk:**

Ik heb vooral op persoonlijk gebied de laatste paar weken een groei bij mezelf vastgesteld. Ik ben naar mijn eigen mening beter gaan presenteren. Eerder was ik nog zenuwachtig geweest voor en tijdens een presentatie en de laatste paar weken ben ik alleen zenuwachtig voorafgaand aan de presentatie en niet meer tijdens.

### 4.4 Aslı Tokbay

Ik heb de volgende dingen bij deze cursus geleerd:

Programmeren met Android: Ik had nog nooit met Android geprogrammeerd, dus moest ik nog leren hoe je met Android programmeert.

Verslagen maken: Ik heb een betere methode geleerd om verslagen te maken. Je krijgt al heel snel te horen dat je heel veel dingen droog moet beschrijven en dat je alleen maar bij bepaalde dingen mag interpreteren.

Ik kan sneller werken met LaTeX dan voor deze cursus.

Wat ik heb gemerkt is dat ik heel veel moeite heb met presenteren. Ik word dan heel zenuwachtig en dit duurt de hele presentatie lang. Dit is iets wat ik zal meenemen zodat ik er in de toekomst aan kan werken.