Technische documentatie

|  |  |
| --- | --- |
| Projectnaam : | Goodlookz |
| Bedrijfsnaam : | 4People |
| Soort document : | Technische documentatie |
| Datum : | 22/5/18 |
| Naam student : | R. Broens |
| Naam assessor : | J. van Amsterdam |
| Naam bpv-begeleider : | J. van Amsterdam |
| Naam praktijkopleider : | J. Franssen |
| Versie document : | 1.00 |

Goodlookz

# Voorwoord

Dit document neemt onderdeel uit aan het eindassessment van de opleiding applicatieontwikkelaar in het derde leerjaar van de opleiding. Het doel van dit document is om verdere informatie te bieden over de uiteindelijke versie van de applicatie. Er zijn enige aanpassingen toegebracht aan de applicatie, hierdoor is het van belang dat er een technische documentatie opgesteld wordt. Deze zal de veranderingen tot de oorspronkelijke documentatie in kaart brengen. Verder is dit document geschreven door Rebecca Broens, die momenteel stageloopt bij het bedrijf 4People in Nuenen.

Goodlookz is een mobiele applicatie die meiden uitnodigt om hun kleding te delen met vrienden. Zij kunnen hun kleding kast online zetten en deze delen met hun vrienden. Deze applicatie is ontwikkeld door Furkan Demirci, een eerdere stagiaire van 4People. Momenteel is dit al mogelijk, maar er is behoefte aan een uitbreiding van het programma.

# Versiebeheer

Hieronder ziet u welke aanpassingen er plaatsgevonden hebben over diverse versies van het document:

|  |  |
| --- | --- |
| Versie | Aanpassingen |
| V0.01 | Opzet document, structuur document, hoofdstuk 1 toegevoegd, hoofdstuk 2 opgesteld |
| V0.02 | Hoofdstuk 2 uitbreiden |
| V0.03 | Hoofdstuk 2 verder uitwerken, hoofdstuk 2 indeling veranderen en hoofdstuk 4 toevoegen |
| V0.04 | Hoofdstuk 2 en 4 uitbreiden |
| V0.05 | Hoofdstuk 2 uitbreiden, hoofdstuk 4 afronden |
| V0.06 | Hoofdstuk 2 uitbreiden |
| V0.07 | Hoofdstuk 2 afgerond, datamodellen toevoegen |
| V0.08 | Gehele datamodel toegevoegd, mappenstructuur omschreven |
| V1.00 | Spellingscontrole |

Inhoud

[Voorwoord 1](#_Toc514746206)

[Versiebeheer 1](#_Toc514746207)

[Inleiding 3](#_Toc514746208)

[Hoofdstuk 1: Ontwikkelomgeving 4](#_Toc514746209)

[Hoofdstuk 2: Deelproducten 6](#_Toc514746214)

[Leen functies 6](#_Toc514746215)

[Verkoop functies 8](#_Toc514746216)

[Instelling functies 11](#_Toc514746217)

[In Your Colour koppelling (Colourshealth) 15](#_Toc514746218)

[Spiegel functie 15](#_Toc514746219)

[Kledingkast uitbreidingen 16](#_Toc514746220)

[Notificaties 20](#_Toc514746221)

[Profiel functies 21](#_Toc514746222)

[Wachtwoord vergeten optie 24](#_Toc514746223)

[Hoofdstuk 3: Gegevensmodel 28](#_Toc514746224)

[Datamodel 28](#_Toc514746225)

[Mappenstructuur 31](#_Toc514746226)

[Navigatiestructuur 32](#_Toc514746227)

[Hoofdstuk 4: GUI 35](#_Toc514746228)

[Leen functies 35](#_Toc514746229)

[Verkoop functies 36](#_Toc514746230)

[Instelling functies 37](#_Toc514746231)

[In Your Colour koppelling (Colourshealth) 39](#_Toc514746232)

[Spiegel functie 40](#_Toc514746233)

[Kledingkast uitbreidingen 41](#_Toc514746234)

[Notificaties 43](#_Toc514746235)

[Profiel functies 44](#_Toc514746236)

[Wachtwoord vergeten optie 46](#_Toc514746237)

[Nawoord 48](#_Toc514746238)

[Bijlages 49](#_Toc514746239)

[Bijlage 1: Datamodel vergrote versies 49](#_Toc514746240)

[Bijlage 2: Gehele datamodel 53](#_Toc514746241)

[Bijlage 3: Gehele navigatiestructuur 54](#_Toc514746242)

# Inleiding

Dit document is gericht op lezers die algemene kennis van ICT gerelateerde onderwerpen begrijpen. Het doel is om duidelijkheid te scheppen over de uiteindelijke versie van de applicatie. Met hierbij de aanpassingen die ontstaan zijn ten opzichte van de oorspronkelijke documentatie.

In hoofdstuk 1 staat de ontwikkelomgeving omschreven die gebruikt is bij ontwikkeling. Daarnaast is omschreven of er enige aanpassingen zijn gemaakt en wat deze aanpassingen inhouden.

Informatie over alle deelproducten is te vinden in hoofdstuk 2. Hierin wordt uitgebreide informatie weergegeven over hoe de deelproducten in elkaar zitten, welke functionaliteiten tot het deelproduct behoren en hoe de database tabel in elkaar zit.

Hoofdstuk 3 geeft weer hoe de uiteindelijke databasestructuur eruitziet, welke mappen er gebruikt worden en hoe de navigatiestructuur in elkaar zit. Hier zijn verschillende schematische weergaves voor gemaakt om duidelijkheid te schetsen.

Daarnaast staat in hoofdstuk 4 verschillende schermafbeeldingen van de applicatie weergegeven. Hier wordt uitgelegd hoe de uiteindelijke GUI in elkaar zit en hoe deze tot stand is gekomen. Tot slot komt het nawoord met nog enige bijlages.

# Hoofdstuk 1: Ontwikkelomgeving

De oorspronkelijke ontwikkelomgeving en de huidige ontwikkelomgeving komen overeen met wat er in het technisch ontwerp is omschreven. Dit betekent dat er geen aanpassingen zijn gemaakt binnen de ontwikkelomgeving. Er wordt nog steeds gebruik gemaakt van dezelfde software, hardware, talen en ontwikkelmethodiek.

Hieronder zal de ontwikkelomgeving opnieuw omschreven worden (deze informatie is uit het technisch ontwerp gehaald en indien nodig aangepast):

## Programmeertalen

Bij het realiseren van de applicatie is gebruik gemaakt van Xamarin, Xamarin maakt het mogelijk om met één programmeertaal nieuwe applicaties te ontwikkelen die op verschillende platformen kunnen draaien, zoals: iOS, Windows Phone, UWP en Android.

Dit betekend dat de backend van de applicatie geschreven is in C# en de front-end in XAML. Daarnaast is er gebruik gemaakt van een webserver om te kunnen communiceren met de database. Om gegevens op te halen, te wijzigen of op te slaan wordt er gebruik gemaakt van PHP en JSON.

## Ontwikkelmethodiek

Er is merendeels gewerkt via de watervalmethodiek tijdens het ontwikkelen van de applicatie, maar aangezien de opdrachtgever niet bekend is met ontwikkelmethodieken zijn er enkele uitzonderingen ontstaan. Enkele aspecten van de scrummethodiek zijn opgenomen bij het realiseren van dit product.

De voortgang van het project is opgeslagen op de website “[Trello](https://trello.com/)”, een website die als tool gebruikt wordt bij een doorlopend project. Dit zorgt ervoor dat de voortgang van het project duidelijk weergegeven worden voor de opdrachtnemer en opdrachtgever. De website maakt gebruik van verschillende borden waar gebruikers kaarten op kunnen plaatsen.

## Gebruikte software/websites

Bij het ontwikkelen van de applicatie wordt het programma “Visual Studio 2015” gebruikt. Deze is tevens ook gebruikt bij het testen van de functionaliteiten. De bestanden, die zich op de webserver bevinden, zijn gemaakt en aangepast met “Visual Studio Code”.

Om de webserver te bereiken en bestanden aan te passen is [FileZilla](https://filezilla-project.org/index.php), een gratis ftp-software tool, gebruikt. Daarnaast zijn er enkele websites die gebruikt zijn bij de ontwikkeling van de applicatie, zoals: [MySQL Syntax controleren](https://nl.piliapp.com/mysql-syntax-check/), [Trello](https://trello.com/), [PHP validator](https://phpcodechecker.com/), [W3Schools](https://www.w3schools.com/), [JSON validator](https://jsonlint.com/), enz.

## Hardware informatie ontwikkelsysteem

De applicatie is gebouwd op een werkcomputer die de opdrachtgever heeft geleverd. Dit systeem is niet alleen gebruikt bij de ontwikkeling van het product, maar ook bij het testen. Hieronder staan de hardwarecomponenten van dit systeem:

|  |  |
| --- | --- |
| Component | Benaming |
| Processor | Intel® Core™ i7-2600 CPU @ 3.40GHz |
| Geïnstalleerd geheugen (RAM) | 8 GB |
| Type systeem | 64-bits besturingssysteem, x64-processor |
| Opslagcapaciteit | Samsung SSD 850 EVO 250 GB, Intel Raid 0 Volume 1 TB |
| Videokaart | AMD Radeon HD 5800 Series |

# Hoofdstuk 2: Deelproducten

Heironder staan de deelproducten binnen de applicatie omschreven. Deze zijn in verschillende alinea’s gesplitst en hierbij staan de volgende punten: omschrijving van het deelproduct, wat voor aanpassingen er ontstaan zijn en welke gegevens er in de database worden opgeslagen.

## Leen functies

#### Leenverzoek versturen

**Omschrijving**

Gebruikers kunnen van elkaar kleding lenen, dit kunnen ze doen door een leenverzoek naar elkaar te versturen. Wanneer een gebruiker een kledingstuk heeft gevonden die hij/zij wilt lenen, dan moet er een datum opgegeven worden wanneer het item geleend wordt, het aantal dagen dat de gebruiker het item wilt lenen en kan de gebruiker ervoor kiezen om een bericht erbij te versturen. Het laatste is optioneel.

De applicatie voegt het verzoek toe in de database, die vervolgens aankomt bij de eigenaar van het item. Hierna kan de eigenaar het accepteren of afwijzen en een contactformulier invullen om met elkaar in contact te komen.

Hierlangs is een flowchart te zien die de stappen doorloopt van dit proces.

**Database tabel**

De tabel waar de leenverzoeken in op worden geslagen heet lend. Lend\_id is de primary key in deze database. Daarnaast heb je het borrow\_users\_id en owner\_users\_id met hierin users\_id’s van de twee personen rondom het verzoek. Zoals de titel al omschrijft, gaat borrow\_users\_id over degene die het item wilt lenen en owner\_users\_id is degene die de eigenaar is van het item. Vooraf heette deze twee rijen users\_id1 en users\_id2, aangezien het lastig is om te bepalen welke de eigenaar en welke de lener is, zijn er aanpassingen gemaakt.

Verder wordt het type van het kledingstuk in type opgeslagen en het id van in item\_id. Dan heb je nog informatie over het verzoek. In date komt de datum waarop het item geleend zal worden, in days komen het aantal dagen, in message komt het optionele bericht en tot slot heb je de rij lending. Lending laat zien of het leenverzoek geaccepteerd is of nog niet.

#### Contactformulier

**Omschrijving**

Wanneer gebruikers contact met elkaar willen maken, werden eerst de contactgegevens doorgestuurd. Dit is veranderd binnen de applicatie. De gebruiker zal een contactformulier in moeten vullen waarbij zij hun naam en telefoonnummer of email moeten opgeven. Daarnaast kunnen zij aangeven of ze liever per telefoon of mail gecontacteerd willen worden.

Hierna zal er een mail gestuurd worden naar de ontvanger met de contactgegevens hierin. Hierdoor kunnen gebruikers contact opnemen met elkaar en zal Good Lookz de gebruikers een stapje in de goede richting lijden.

**Database tabel**

Bij deze functie zit een database tabel aan vast. Gebruikers, die het contactformulier invullen en opsturen, zullen akkoord moeten gaan. Hierdoor is er zekerheid dat zij wetende hun gegevens doorsturen naar een ander en zorgt dit ook voor bewijsmateriaal.

Inde tabel accept\_contact worden deze gegevens opgeslagen. Het users\_id wordt gebruikt om het terug te kunnen koppelen met de gebruiker. Daarnaast is er de rij accept, als deze op 1 staat is het geaccepteerd. En tot slot komt accept\_date die aangeeft wanneer de gebruiker dit heeft geaccepteerd.

## Verkoop functies

#### Verzoek versturen

**Omschrijving**

Gebruikers hebben nu de optie om een bericht te sturen bij hun aanbod, daarnaast is te zien uit welke plaats het kledingstuk vandaan komt. Gebruikers krijgen de optie om een bod in te vullen of leeg te laten, hetzelfde geld over het berichtje die zij erbij kunnen sturen.

Hierna zal de applicatie via de webserver de gegevens opslaan in de database met een insert query. Hierlangs is te zien hoe het proces zich verloopt.

**Database tabel**

In de tabel sale\_requests wordt alle informatie over het verzoek opgeslagen. Hierbij wordt de requests\_id gebruikt als primary key om functies uit te voeren. De sale\_id linkt het request terug naar het sale item. Daarnaast worden users\_id1 en users\_id2 gebruikt om de gebruikers met elkaar te linken, in de database zijn er opmerkingen geplaats om deze twee uit elkaar te houden.

Binnen de rij users\_id1 worden de id’s van de bidders geplaatst, en in users\_id2 komen de id’s van de eigenaren. Daarnaast wordt de gebruikersnaam van de bidder opgeslagen in username.

Vervolgens komt er informatie over het bod, hoeveel de bidder heeft geboden wordt in bidding opgeslagen. Binnen comments worden de opmerkingen opgeslagen. Tot slot komt typeCloth, hierin staat een int die betrekking heeft tot het type kleding.

#### Aanpassen verkoop item

**Omschrijving**

Gebruikers hebben de mogelijkheid gekregen om kledingstukken die te koop staan aan te passen. De applicatie haalt allereerst de gegevens van het item op en stopt ze in verschillende invoervelden. Wanneer een gebruiker zijn wijzigingen wilt opslaan, moeten zij op edit drukken. Hierna voert de applicatie een update query uit en zal het opgeslagen zijn.



**Database tabel**

Vele rijen die momenteel in deze tabel zitten zijn overgenomen van de oude versie van de database. Ondanks enkele van deze rijen aangepast kunnen worden, is dit niet gebeurd a.d.h.v. de impact die het zou hebben op de applicatie. Er zijn enkele functies die afhankelijk zijn van deze velden, vandaar dat ze momenteel nog in de tabel zitten.

Sale\_id wordt gebruikt om een sale request te linken aan een sale item. Users\_id linkt terug naar de gebruiker die de eigenaar is van het kledingstuk die verkocht wordt. Hierna komt het item id van het kledingstuk, in een van de volgende rijen zal deze staan: head\_id, top\_id, bottom\_id, feet\_id. De andere rijen zullen null zijn.

Verder komt de picture van het item, deze wordt opgeslagen om het te kunnen weergeven binnen de applicatie. Daarnaast wordt de prijs van het kledingstuk opgeslagen in price. Gebruikers moeten overigens ook een omschrijving toevoegen, deze staat in desc. De plaatsnaam van de locatie komt te staan in sale\_city en tot slot het type.

## Instelling functies

#### Notificatievoorkeuren

**Omschrijving**

Binnen de instellingen kunnen gebruikers aangeven welke notificaties hij/zij aan of uit wilt hebben. Dit wordt doormiddel van switches gedaan. De applicatie slaat de voorkeuren op in de database en in de applicatie zelf. Wanneer er notificaties geladen worden, wordt er ook gekeken naar de voorkeuren van de gebruiker.

Als de pagina geopend wordt, haalt de applicatie de notificatie instellingen van de gebruiker op uit de database en zet de switches aan of uit (a.d.h.v. de opgehaalde instellingen).



**Database tabel**

De voorkeuren worden opgeslagen in de tabel: notification\_settings. En bestaat uit de volgende kolommen: notify\_id, notify\_new\_friend, notify\_new\_bid, notify\_new\_lend en users\_id.

Notify\_id wordt gebruikt om rijen uit de tabel te verwijderen of aan te passen, daarnaast geven notify\_new\_friend, notify\_new\_bid en notify\_new\_lend aan of deze notificaties uit of aan staan. Wanneer er een “1” staat, betekent dat de notificatie aan staat. Notificaties zullen standaard aan staan wanneer een gebruiker voor de eerste keer inlogt. De kolom users\_id wordt gebruikt om de voorkeuren te linken bij het juiste account.

#### Wachtwoord wijzigen

**Omschrijving**

Wanneer een gebruiker zijn/haar wachtwoord wilt wijzigen moet hij/zij als allereerst het huidige wachtwoord opgeven. Daarnaast moet er een nieuw wachtwoord ingevoerd worden en deze wordt tevens ook herhaalt ter controle.

Wanneer de gebruiker op “wijzig wachtwoord” klikt, zal er eerst gekeken worden of het ingevoerde wachtwoord overeenkomt met het huidige wachtwoord. Daarnaast wordt er ook gekeken of nieuwe en herhaalde wachtwoord hetzelfde zijn.



**Database tabel**

Er is geen specifieke tabel voor deze functie, wel wordt er gebruik gemaakt van de users tabel. De wachtwoorden staan hierin opgeslagen. Er wordt gekeken of het ingevoerde oude wachtwoord overeenkomt met het wachtwoord die opgeslagen staat op de database. Als alles klopt, wordt het wachtwoord in de database gewijzigd.

#### Persoonlijke gegevens aanpassen

**Omschrijving**

Als allereerst worden de persoonlijke gegevens van de gebruiker opgehaald en in de invoervelden gestopt, zoals de voor- en achternaam. Daarnaast worden ook de opgeslagen maten van de gebruiker opgeslagen die de gebruiker kan aanpassen. Via pickers worden de regio en geslacht gekozen. Daarna worden de maten in de drie onderste pickers gezet. Wanneer een gebruiker een regio of geslacht kiest, veranderen de maten mee.

Gebruikers kunnen de invoervelden aanpassen en kunnen andere maten kiezen doormiddel van de pickers. Wanneer ze de veranderingen willen opslaan, moeten ze op “save” drukken. Hierna stuurt de applicatie een update query naar de verschillende tabellen. Over welke tabellen het gaat is te zien in de volgende alinea. De applicatie krijgt vervolgens een resultaat, wanneer het mislukt is komt er een foutmelding.



**Database tabel**

Er zijn twee verschillende tabellen die geüpdatet worden bij deze functie: users en users\_size. De rijen first\_name en last\_name worden gewijzigd in de tabel users. Daarnaast worden de gender, region, topNr, botNr en feetNr gewijzigd in de users\_size tabel.

#### E-mail veranderen

**Omschrijving**

Naast het wijzigen van wachtwoorden of persoonlijke gegevens kunnen gebruikers ook hun e-mail aanpassen om er zeker van te zijn dat mailtjes goed binnen komen. De applicatie haalt allereerst het huidige email op via een class en stop deze als een placeholder in het eerste invoerveld.

Naast een nieuwe email, moet de gebruiker ook het huidige wachtwoord opgeven als validatie. Deze wordt gecheckt wanneer de gebruiker op “change email” drukt. En wanneer hij juist is zal de update query uitgevoerd worden, mits de email nog niet in de database staat.



**Database tabel**

De tabel users wordt gebruikt bij het wijzigen van deze gegevens. De enige kolom die gewijzigd wordt, is de email kolom. Daarnaast wordt het users\_id gebruikt om te kijken bij welke user het email adres gewijzigd moet worden.

## In Your Colour koppelling (Colourshealth)

**Omschrijving**

Ondanks de opdrachtgever de documentatie doorgenomen heeft met de opdrachtnemer, is er een misverstand ontstaan. De koppeling met “Colourshealth” moest een koppeling met “In Your Colour” zijn. Dit is naderhand aangepast wanneer de functionaliteit al gerealiseerd was, het concept blijft enigszins hetzelfde.

Naast het kunnen bekijken van een gevoelskleur, is er ook een knop toegevoegd die de gebruiker naar een webview van de pagina brengt. Hierdoor kan de gebruiker meer informatie opdoen over In Your Colour. Ook worden de gevoelens nu opgehaald vanuit de database, zodat deze koppeling uitgebreid kan worden als hier behoefte aan is.

**Database tabel**

De tabel die gebruikt wordt voor dit deelproduct is: inyourcolour. Er zijn drie verschillende kolommen binnen deze tabel: colourID, colourTitle en colourHex. colourID is het id van een gevoel, deze is altijd uniek en wordt vooral gebruikt wanneer een kleur verwijdert moet worden. Iedere kleur heeft een titel en een hex code. De titel wordt opgeslagen in colourTitle en de hex in colourHex. De titel is terug te zien in de applicatie en de hex wordt gebruikt om het knopje de juiste kleur te geven.

## Spiegel functie

Er zijn geen wijzigingen gemaakt bij het realiseren van de spiegel functie.

## Kledingkast uitbreidingen

#### Filterfunctie

**Omschrijving**

Om een filterfunctie uit te voeren zal de applicatie allereerste de gegevens van de gebruiker moeten ophalen. De applicatie laat alleen de mogelijkheden zien die van toepassing zijn voor de gebruiker. Als de gebruiker alleen maar broeken heeft opgeslagen in zijn kledingkast, zal alleen de waarde “bottom” weergegeven worden in de picker.

Nadat een gebruiker een type kleding heeft gekozen, haalt de applicatie op welke verschillende kleuren er beschikbaar zijn. Wanneer de gebruiker alleen maar zwarte broeken heeft toegevoegd, zal de applicatie alleen de optie zwart weergeven. Verder worden kledingstukken waarvan geen kleur opgegeven is, gezien als “varied colours”.

Hierna kunnen gebruikers op filter klikken en zal de applicatie de filteropties opslaan in een static class en stuurt de gebruiker terug naar de voorafgaande pagina. Hierbij leest de applicatie of er een filter is toegepast en als dit het geval is, zal hij alleen de kledingstukken laten zien die passend zijn bij de filter opdracht.

Wanneer de gebruiker de filters reset, worden de waardes weggehaald in de static class en zal de applicatie alle kledingstukken laten zien. Hieronder staan de flowcharts van dit proces te zien. De rechter flowchart laat zien hoe de filter opties worden opgehaald en de linker laat het proces zien binnen de wardrobepage.



**Database tabel**

Bij het ophalen van de types en kleuren wordt gekeken naar de head, top, bottom of feet tabel. Hierbij wordt allereerst gekeken of er enige kledingstukken opgeslagen zijn per type en hierna wordt er gekregen naar de color rij. Er zijn geen wijzigingen in de database gemaakt voor deze functionaliteit. Verder worden er alleen gegevens opgehaald vanuit de database, niet aangepast.

#### Kleding toevoegen met maat

**Omschrijving**

De applicatie haalt allereest de regio en geslacht van de gebruiker op, deze staan opgeslagen in de database. Nadat deze gegevens zijn opgehaald worden de juiste waardes geselecteerd in de pickers. Hierna kunnen gebruikers dezelfde stappen doorlopen als bij de oude kleding toevoegen functie.

De enige toevoeging hierbij is dat gebruikers nu een maat kunnen kiezen uit een picker, i.p.v. het invullen van een invoerveld.



**Database tabel**

Om de maten op te halen van de gebruiker wordt gebruik gemaakt van een tabel. Binnen de tabel, users\_size, bevinden zich een aantal rijen, zoals: users\_id, gender, region, topnr, botnr en feetnr. Users\_id wordt gebruikt om de users tabel met de user\_size tabel te linken. Iedere gebruiker heeft een record binnen de user\_size tabel waarbij duidelijk wordt gemaakt van welke regio hij/zij komt en wat het geslacht is. Daarnaast wordt het index nummer gegeven van de maat.

#### Maten opslaan

**Omschrijving**

Wanneer gebruikers voor de eerste keer inloggen moeten zij verschillende stappen doorlopen om uiteindelijk bij de menupagina te komen. Ze moeten hun winkelvoorkeuren opgeven en hun profiel bewerken, daarnaast krijgen ze ook een formulier waar gevraagd wordt naar hun maten. Deze informatie is voor belang om kleding die te klein of te groot is te filteren uit de kledingkast van vrienden.

Allereerst wordt de gebruiker gevraagd om hun geslacht en regio te kiezen in de pickers. Wanneer deze twee waardes geselecteerd zijn, worden de maten geladen in de drie verschillende pickers. Gebruikers moeten een maat voor een top, bottom en feet selecteren voordat ze hun maten kunnen opslaan.



**Database tabel**

De tabel users\_size wordt gebruikt om de maten in op te slaan, hierbij worden ook de regio en geslacht opgeslagen van de gebruiker. Binnen de rijen topNr, botNr en feetNr staan de maatnummers in. De gebruiker krijgt alleen de werkelijke maat te zien, maatnummers worden gebruikt om de maten van verschillende regio’s en geslachten te vergelijken. Verder wordt het users\_id gebruikt om de gegevens te linken aan de gebruiker, het spreekt voor zich.

## Notificaties

**Omschrijving**

Binnen de applicatie zullen gebruikers notificaties binnen krijgen, deze zijn terug te vinden op het menu. Rechtsboven op de menupagina zal een notificatie icoontje staan. Langs dit icoontje staat het aantal binnengekregen notificaties.

Daarnaast kunnen gebruikers hun notificaties bekijken wanneer ze op het icoontje klikken. De applicatie opent een nieuwe pagina met hierin een lijst. Hierbij staan een aantal gegevens: de profiel foto van de afzender, gebruikersnaam van de afzender en een passende tekst die aangeeft over wat voor soort notificatie het gaat.

Wanneer een gebruiker op een notificatie klikt, wordt hij/zij doorgestuurd naar de juiste pagina. Hieronder staat een flowchart die dit proces visueel in kaart brengt:



**Database tabel**

Notificaties worden op de database opgeslagen binnen de notifications tabel. De notif\_id wordt opgeslagen, het users\_id van de ontvanger wordt opgeslagen, notif\_type gaat over welk type notificatie het gaat. Er zijn drie soorten notificaties: leenverzoeken, vriendenverzoeken en koopverzoeken.

Daarnaast wordt er ook het id van het item opgeslagen in de rij notif\_relation\_id. Hier kan dus een lend\_id, request\_id of friends\_id staan. Dit wordt gebruikt om ervoor te zorgen dat als een vriendenverzoek geaccepteerd wordt, deze ook niet meer te zien is bij de notificaties.

Verder wordt er een notif\_username op geslagen met hierin de gebruikersnaam van de verstuurder. Notif\_message bepaald wat voor soort tekst achter de notificatie staat en de notificaties worden op datum gesorteerd, daarvoor wordt notif\_date gebruikt.

## Profiel functies

De opdrachtnemer heeft tijdens de realisatiefase aangegeven dat hij graag profielfoto’s wou toevoegen om de notificaties persoonlijker te maken. Hier is over na gedacht en zijn de onderstaande functionaliteiten voor gemaakt.

De volgende functionaliteiten gebruiken allemaal dezelfde tabel binnen de database.

**Database tabel**

De tabel die gebruikt wordt heet: “profile” en beschikt over 5 verschillende kolommen. Iedere gebruiker zal gegevens in deze tabel hebben staan over hun profiel.

De volgende kolommen bevinden zich in de tabel: profile\_id, users\_id, username, picture en description. Profile\_id wordt gebruikt als een primary key, users\_id en username linken de accounts aan een profiel, binnen de rij picture komt de link te staan naar de afbeelding en in de description komt een omschrijving die de gebruiker heeft opgegeven.

#### Profiel opslaan

**Omschrijving**

Wanneer een gebruiker voor de eerste keep inlogt, zijn er een aantal stappen die zij moeten doorlopen. Winkelvoorkeuren moeten aangegeven worden, er wordt gevraagd welke maten de gebruiker gebruikt en er wordt een profiel opgeslagen.

De gebruiker zal een foto kunnen maken met zijn telefoon of kiezen uit zijn galerij en kan vervolgens een omschrijving aangeven over zichzelf. Daarnaast kan de gebruiker ervoor kiezen om deze laatste stap over te slaan en gelijk door te gaan naar het menu.

De afbeeldingen worden via een php bestandje op formaat gemaakt en opgeslagen op de good-lookz server. Ze zullen niet in de database worden opgeslagen.

Op de volgende pagina zijn enkele flowcharts te zien die deze functionaliteit omschrijven.

Profiel opslaan:

Profiel opslaan overslaan:



#### Profielpagina aanpassen

**Omschrijving**

Binnen de instellingen kunnen gebruikers ervoor kiezen om hun profiel aan te passen. Hierbij kunnen ze de foto en omschrijving wijzigen.

De foto aanpassen kan via twee verschillende manieren: een foto maken of een foto kiezen. Bij het maken van de foto opent de applicatie de camera en bij het kiezen van een foto opent de galerij.

Het aanpassen van de omschrijving is te doen door simpelweg het invoerveld aan te klikken en te typen. Gebruikers moeten wel op “save” drukken, anders worden de wijzigingen niet opgeslagen.

Na het opslaan stuurt de applicatie de gebruiker terug naar de vorige pagina.

Deze functie werkt op een soortgelijke manier als het opslaan van een account. De enige aanpassing is in de MYSLQI-query. In plaats van een insert query wordt hier een update query gebruikt. De bovenstaande flowchart is dus ook van toepassing voor deze functionaliteit.

#### Profielpagina van vrienden bekijken

**Omschrijving**

Binnen de applicatie is het nu mogelijk om de profielpagina’s van je vrienden te bekijken en hierdoor ook naar hun kledingkast te navigeren. Van tevoren kon je hier al naartoe navigeren, maar dit was vanuit de wardrobe zelf. Wanneer de gebruiker op de vrienden pagina bevindt, kan er geklikt worden op een vriend.

Hierbij wordt er een nieuwe pagina geopend met hierin de volgende gegevens: gebruikersnaam, profiel foto en profiel beschrijving van de geselecteerde vriend.

Daarnaast is er een knop aanwezig waardoor de gebruiker naar de kledingkast van de vriend kan navigeren. Dit zorgt voor een snellere en makkelijkere manier om de kleding van vrienden te bekijken.

## Wachtwoord vergeten optie

**Omschrijving**

De opdrachtgever heeft later aangegeven dat hij graag een wachtwoord vergeten functie in de applicatie wilt hebben, deze is hierdoor zonder documentatie toegevoegd aan de applicatie. Het is van belang dat hier alsnog een stukje documentatie over gemaakt wordt.

Gebruikers die in willen loggen, kunnen hun wachtwoord vergeten zijn. Zij krijgen de kans om een code aan te vragen, doormiddel van hun e-mail in te voeren. Mits de e-mail herkend is binnen het systeem, wordt er een e-mail verstuurd met hierin een willekeurige code gegenereerd (8 tekens lang, letters en nummers, hoofdlettergevoelig) die de gebruiker vervolgens kan gebruiken.

Wanneer de gebruiker de code heeft ontvangen zal de code voor maximaal 30 minuten geldig zijn en zal hierna verlopen. De gebruiker zal naar de volgende pagina moeten navigeren waarbij zij opnieuw hun e-mail invoeren en de code die zij hebben ontvangen.

Mits de code niet overeenkomt met de werkelijke code, zal de applicatie een foutmelding geven. Iedere recovery code heeft 5 kansen. Nadat de gebruiker de code 5 keer fout heeft ingevoerd, wordt de code geblokkeerd. Gebruikers kunnen opnieuw een code aanvragen en de oude code wordt verwijderd.

Als de code al een gebruikt is, zal dit in de database weergegeven staan. Onder de kolom “recovery\_active” zal “used” staan. De applicatie zal een foutmelding geven.

Daarnaast is er nog een uitzondering die kan ontstaan bij het invoeren van de e-mail en code. Wanneer de e-mail onjuist is, zal de applicatie dit ook aangeven.

Even kort samenvatten, dit zijn de uitkomsten die zich kunnen voordoen bij het invoeren van de e-mail en code:

|  |  |
| --- | --- |
| Type | Omschrijving |
| Active | E-mail en code zijn goedgekeurd |
| Blocked | Code is geblokkeerd, er zijn meer dan vijf kansen gebruikt |
| Used | Recovery code is gebruikt en niet meer geldig |
| Expired | Code is ouder dan 30 minuten, en hierdoor verlopen |
| Invalid e-mail | Ingevoerde e-mailadres is onjuist |
| Invalid code | Code is onjuist ingevoerd |

Nadat de e-mail en code overeenkomt met de database, kan de gebruiker zijn/haar wachtwoord wijzigen. De applicatie vraagt de gebruiker een nieuw wachtwoord in te voeren en deze te herhalen voor de zekerheid. Wanneer dit correct gedaan is, zal het nieuwe wachtwoord gewijzigd worden.

Hieronder staan enkele flowcharts die dit proces kort zullen samenvatten:





**Database tabel**

Bij deze functionaliteit hoort de tabel: recovery\_emails, deze kunt u hierlangs bekijken. Iedere recovery heeft een recovery\_id, users\_id, recovery\_time, recovery\_tries en een recovery\_active. Het users\_id wordt gebruikt om een koppeling te maken met de users en recovery\_emails tabellen. Verder wordt in de kolom recovery\_code de willekeurige code opgeslagen die gebruikers moeten gebruiken, recovery\_time geeft aan wanneer het verzoek verstuurd is en recovery\_active laat zien of de code op active of used staat.

Iedere gebruiker kan maar één recovery\_code opgeslagen hebben in de database. Wanneer een gebruiker twee keer een code opvraagt, zal de oude code verwijderd worden en een nieuwe opgeslagen worden.

# Hoofdstuk 3: Gegevensmodel

Hieronder is de nieuwe versie van het datamodel uitgewerkt en is de uiteindelijke versie van de mappenstructuur te zien. Tevens is er ook een nieuwe versie van de navigatie gemaakt met hierin de nieuwe pagina’s die ontwikkeld zijn.

## Datamodel

Aangezien de database over vele diverse tabellen beschikt, is het datamodel opgesplitst in verschillende onderdelen om overzicht te behouden. De onderstaande diagrammen zijn verkleind, grotere versies van deze diagrammen zijn terug te vinden in de bijlages. Daarnaast is er ook een gehele versie gemaakt van de datatabellen, deze is terug te zien in bijlage 2.

Om de leenfunctionaliteiten werkend te krijgen wordt gebruik gemaakt van de onderstaande tabellen met relaties. De borrow\_users\_id en owner\_users\_id komen beide vanuit de users tabel. Uiteraard zullen hier niet twee keer dezelfde id’s in staan.

Daarnaast kan het item\_id bestaan uit een head\_id, top\_id, bottom\_id of feet\_id. Dit ligt eraan welk kledingstuk is gekozen.

Ook de notificatie tabel zit aan het leen tabel vast. Wanneer een leenverzoek verstuurd wordt, zal er ook een notificatie aangemaakt worden. En om deze te linken met elkaar, komt het lend\_id van toepassing.



Bij het versturen van de contactgegevens worden gebruikers vereist om hier mee akkoord te gaan. Dit wordt binnen de accept\_contact tabel opgeslagen en de users\_id wordt gelinkt.

Bij de instellingen zijn er vele verschillende aanpassingen een gebruiker kan maken, zoals: aanpassen van het profiel, veranderen van voor- en achternaam, wijzigen maten, wachtwoord aanpassen of de email wijzigen. Al deze verschillende tabellen zijn uiteindelijk gelinkt door het users\_id.



Bij het versturen van een verkoop verzoek wordt gebruik gemaakt van de onderstaande tabellen. Allereerst staat een sale item opgeslagen in de sale tabel, met hierbij de benodigde informatie. Het users\_id linkt het item met de gebruiker.

Om te kijken welke sale\_requests behoren tot een sale item, is hier een relatie gemaakt tussen de twee tabellen. Wanneer een item verwijderd wordt van de sale, zullen de sale\_requests ook verwijdert moeten worden.

Daarnaast wordt er ook een notificatie gecreëerd. Binnen de notif\_relation\_id wordt het requests\_id gezet. Wanneer het verzoek verwijderd of geaccepteerd is, zal de notificatie ook verwijderd worden. Daarom is het van belang dat deze twee tabellen met elkaar verbonden zijn.

Gevoelens worden in de database opgeslagen en vervolgens opgehaald door de applicatie, dit zorgt ervoor dat er ruimte is gemaakt voor het uitbreiden van deze gevoelens.

De tabel heeft geen relaties met andere tabellen en heeft ook niets te maken met andere functies.

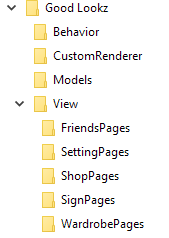


Bij de wachtwoord vergeten functie wordt gebruik gemaakt van de tabel recovery\_emails. Deze tabel heeft een relatie met de users tabel.

Hiervoor wordt het users\_id gebruikt. Daarnaast is het zo dat er per gebruiker maar één record in de recovery\_emails kan staan.

## Mappenstructuur

De mappenstructuur van de applicatie en de API zijn enigszins anders dan oorspronkelijk gepland was. Enige mappen zijn uiteindelijk toegevoegd of verwijderd. Lees het technisch ontwerp voor uitgebreide informatie over de mappenstructuur.

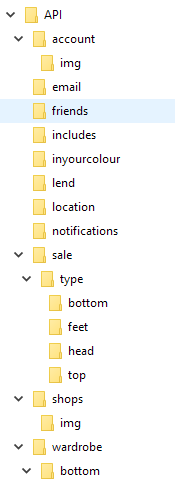
**De applicatie**

Hierlangs is de mappen structuur te zien van de uiteindelijke applicatie. Wanneer deze mappenstructuur vergeleken worden met de mappenstructuur van het technisch ontwerp, dan is er te zien dat er twee verschillen zijn. Er is een map toegevoegd, CustomRenderer, en een map is weggehaald, BackgroundPages.

Binnen de CustomRenderer map worden de renderers opgeslagen voor enige controls, zoals: de camera en ronde afbeeldingen. Deze renderers zijn nodig geweest om de huidige controls aan te passen en eigen controls te maken. Hiervoor moet voor zoals iOS als Android aparte classes gemaakt worden.

BackgroundPages is uiteindelijk weggehaald, omdat dit een misverstand is geweest in de documentatie.

**De API**

Binnen de mappenstructuur van de API is er maar een verandering ontstaan ten opzichte van het technisch ontwerp. Er is een nieuwe map toegevoegd, inyourcolour. Deze map ontstaat uit twee PHP-scripts die gegevens ophalen vanuit de database of nieuwe gevoelens opslaat in de database. Uitleg over de rest van de mappen is te vinden in het technisch ontwerp.



## Navigatiestructuur

Pagina’s binnen deze navigatiestructuur zijn alleen de pagina’s met de functionaliteiten die betrekking hebben tot het project, niet iedere pagina van de applicatie zit in deze structuur. Aangezien er binnen de navigatiestructuur vele verschillende pagina’s bevinden, is de structuur opgesplitst in een aantal verschillende categorieën. De gehele navigatiestructuur is terug te vinden in de bijlages.

**Eerste login**

In het technisch ontwerp heette dit onderdeel “Maat opslaan”, maar aangezien er ook een andere functie bij is gekomen is deze naam gewijzigd. Er is een nieuw onderdeel toegevoegd: het opslaan van je profiel. Dit kunnen gebruikers doen bij de eerste login. Daarnaast zijn er nog een aantal andere zaken die zij bij de eerste inlog opslaan, zoals: winkelvoorkeuren en de kleding van de gebruiker. Hieronder is de structuur te zien van wanneer een gebruiker voor het eerst inlogt op zijn account:

Vanaf de LoginPage zal de gebruiker naar ChoseShopPage pagina navigeren, deze is overigens ontwikkeld door F. Demirci. Hierna komt de gebruiker bij SaveSizePage, hierbij worden ze gevraagd om hun kledingmaten aan te geven. Dit is van belang om alleen kleding op te halen die de gebruiker wellicht zou passen. Vervolgens kan de gebruiker zijn profielfoto en omschrijving aanpassen en opslaan, daarnaast is er ook de optie om dit over te slaan. Tot slot komt de gebruiker bij de MenuPage. Bij een normale inlog zal de gebruiker vanuit de LoginPage naar de MenuPage gaan.

**Menu navigatie**

Binnen het menu zijn er een aantal wijzigingen gemaakt in ten opzichte van de structuur die in het technisch ontwerp omschreven staat. Hieronder staat de structuur aangegeven:

De FriendsPage was al aanwezig in de applicatie, maar aangezien er een nieuwe functie erbij is gekomen, wordt deze nu ook weergegeven binnen de structuur. Gebruikers kunnen het profiel van hun vrienden bekijken en hierdoor ook hun kledingkast.

Daarnaast is de pagina “ColoursHealthPage” verandert naar “InYourColourPage”, verder is er ook een webview toegevoegd die de pagina van In Your Colour weergeeft binnen de applicatie. Verder zijn er geen wijzigingen gemaakt.

**Kledingkast uitbreidingen**

De onderstaande structuur is een uitgebreide versie van de structuur van het leensyteem en filterfunctie. Daarnaast is de benaming aangepast om het globaler te maken. Er zijn vele aanpassingen gemaakt tegenover de structuur die te zien is in het technisch ontwerp.



In het technisch ontwerp is ervan uitgegaan dat LendPage de pagina was met verschillende leningen, dit bleek uiteindelijk de LendingListPage te zijn. Daarnaast zijn ook een aantal pagina’s van het verkoopsysteem toegevoegd aan de structuur, aangezien er hier enige functionaliteiten aan toegevoegd zijn. Daarnaast is er een nieuwe pagina, WardrobeSelectedSaleItem, ontwikkeld. Op deze pagina kunnen gegevens van een sale item aangepast worden.

Om contact op te kunnen nemen moeten gebruikers hun gegevens invoeren en deze opsturen naar de ontvanger, dit doen ze op de pagina WardrobeContact. Zoals te zien is, kunnen gebruikers deze pagina op verschillende manieren bereiken.

**Notificatiescherm**

Er zijn geen wijzigingen plaatsgevonden bij de navigatie van de notificaties.

**Instellingen**

Tijdens de realisatie van de applicatie zijn enige aanpassingen plaatsgevonden bij de structuur van de instellingen. Er zijn enige pagina’s toegevoegd, zoals: ChangeProfile, PP en TOC.

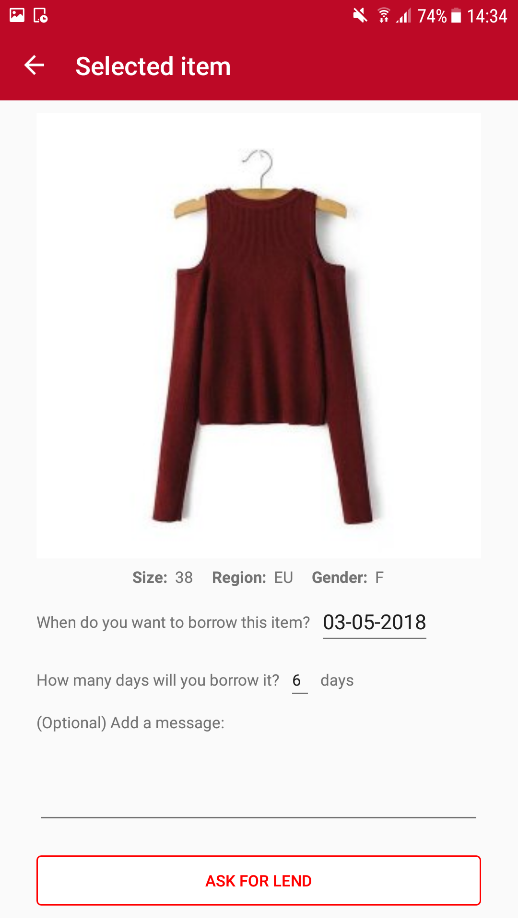
Daarnaast is er een pagina, ChangeBackground, verwijdert. En is de benaming van ChangeUsername veranderd naar ChangeMail.

Op PP is een webview te zien met hierin de Privacy Policy van Good Look en op TOC bevindt zich een webview met de Terms of Conditions.

# Hoofdstuk 4: GUI

Binnen dit hoofdstuk staan de schermafbeeldingen van de applicatie Good Lookz. De afbeeldingen zijn genomen door een Samsung Galaxy s6 Edge Plus.

## Leen functies



**Leenverzoek versturen**

Via een datepicker kunnen gebruikers aangeven op welke datum zij het kledingstuk willen lenen. Ze kunnen vanaf de huidige datum tot 6 maanden een item lenen, andere datums kunnen niet geselecteerd worden.

Daarnaast moeten gebruikers aangeven hoelang zij het item willen lenen, dit wordt met een entry gedaan.

Tot slot krijgt de gebruiker de optie om een berichtje toe te voegen bij het verzoek.

**Afbeelding met schermafbeelding

Beschrijving is gegenereerd met zeer hoge betrouwbaarheid**

**Contactformulier**

Gebruikers zijn verplicht om hun naam in te vullen bij het contactformulier. Daarnaast krijgen ze de optie om een telefoonnummer of een emailadres in te voeren waarmee de ontvanger contact kan opnemen met ze.

Ook kan een communicatie voorkeur opgegeven worden. Als de voorkeur uit gaat naar telefonisch contact, dan moet het telefoonnummer wel ingevoerd zijn.

Het form bestaat uit drie entries en een picker. Daarnaast is er een behaviour toegevoegd op de email entry, om de email te valideren.

## Verkoop functies

**Afbeelding met schermafbeelding

Beschrijving is gegenereerd met hoge betrouwbaarheid**

**Verzoek versturen**

Er zijn enige wijzigingen ontstaan bij de oorspronkelijke pagina en de huidige pagina. Een van de grootste veranderingen is de informatie die erbij is gekomen.

Gebruikers kunnen nu zien in welke plaats het item zich bevindt en ze kunnen een bericht bij het aanbod versturen.

Afbeelding met schermafbeelding

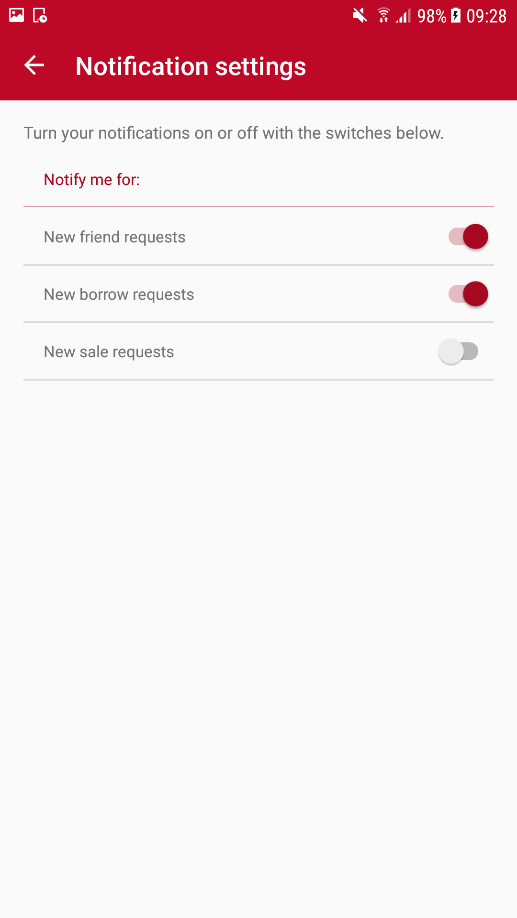
Beschrijving is gegenereerd met zeer hoge betrouwbaarheid

**Aanpassen verkoop item**

Gebruikers hebben de mogelijkheid om de items die zij te koop hebben staan, aan te passen. Ze kunnen de prijs, de plaats en de omschrijving wijzigen als zij hier behoefte aan hebben. De gegevens worden vanuit de database opgehaald en in de invoervelden gestopt.

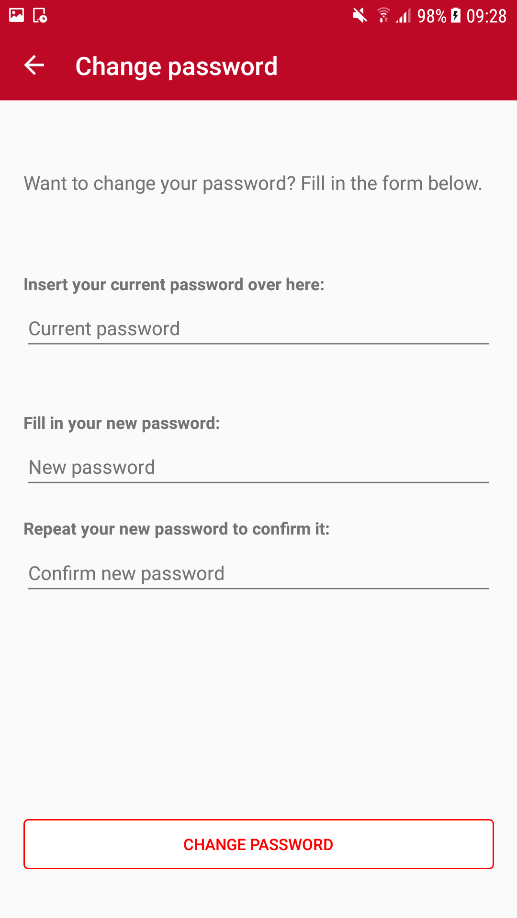
Wanneer een gebruiker de wijzigingen wilt opslaan, zal hij of zij op edit moeten drukken.

## Instelling functies

****

**Notificatievoorkeuren**

Hierlangs is een schermafbeelding te zien van de uiteindelijke GUI voor het instellen van notificaties. De gebruiker kan de switches aan of uit zetten. Er is een TableView aangemaakt met drie SwitchCells. Er is een SwitchCell voor: vriendenverzoeken, leenverzoeken en koopverzoeken.

****

**Wachtwoord wijzigen**

Aan de rechterzijde van deze pagina is de uiteindelijke GUI te zien van het wachtwoord veranderen. Zoals u ziet wordt eerst het oude wachtwoord opgevraagd, hierna een nieuw wachtwoord en er wordt gevraagd of het nieuwe wachtwoord herhaalt kan worden. Hierna kunnen gebruikers op “change password” klikken om hun wachtwoord definitief te wijzigen.

Afbeelding met schermafbeelding

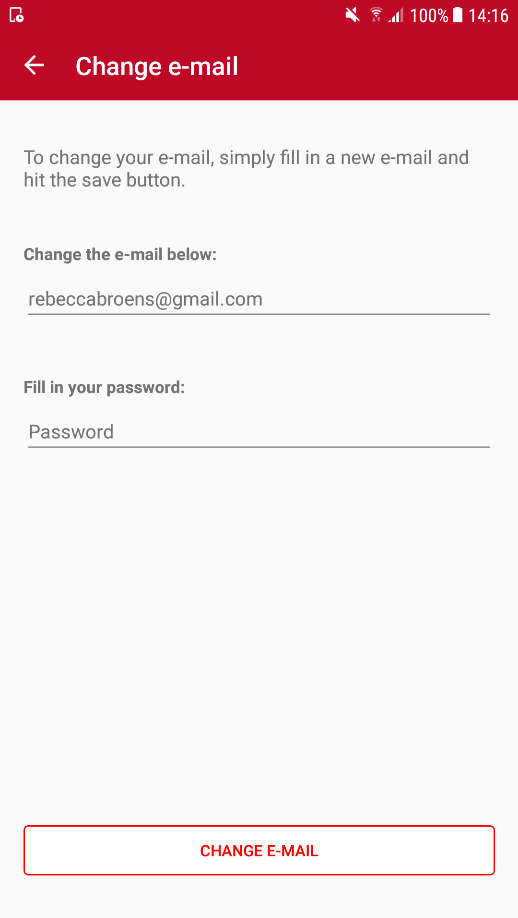
Beschrijving is gegenereerd met zeer hoge betrouwbaarheid

**Persoonlijke gegevens wijzigen**

Binnen het technisch of functioneel ontwerp zijn er geen schetsen gemaakt over hoe deze pagina eruit zal gaan zien.

De gebruikers kunnen via entries hun gegevens wijzigen. Ieder invoerveld is verplicht. Daarnaast kunnen gebruikers hun regio en geslacht wijzigen door een andere waarde te kiezen bij de picker.

Verder kunnen de maten van de gebruiker ook aangepast worden. Dit is te doen door andere items te kiezen in de pickers en op save te drukken.

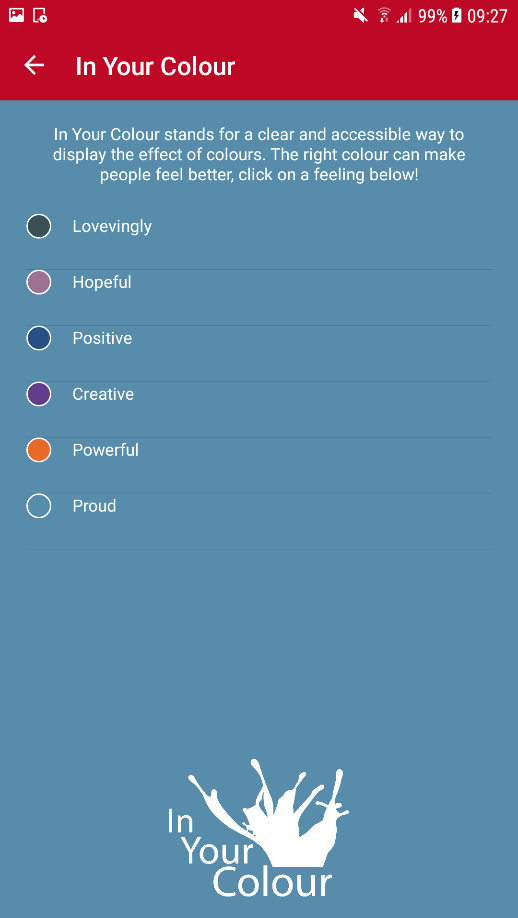
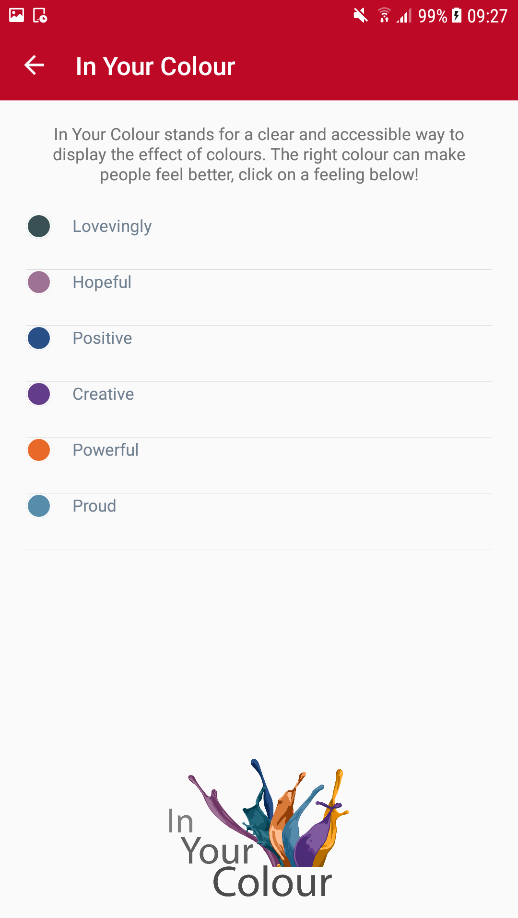
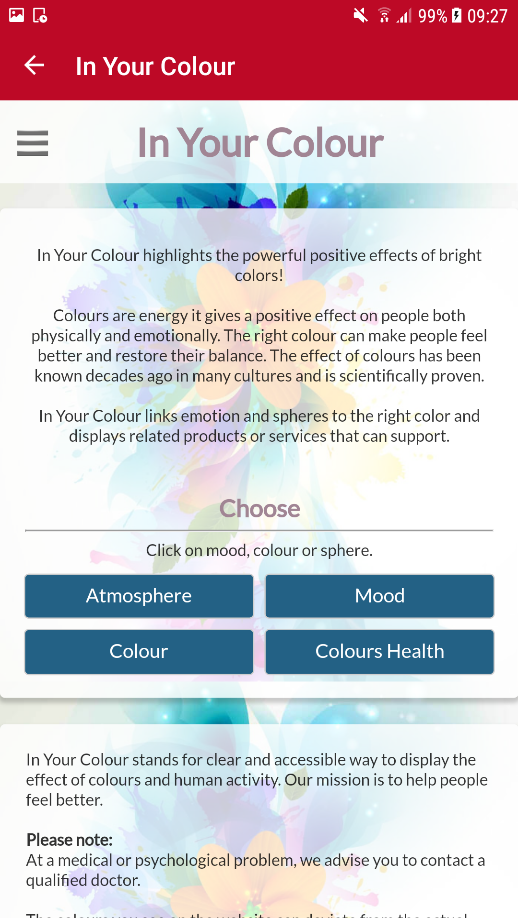


**Email veranderen**

Als allereerst wordt de huidige email weergegeven als een placeholder. Er zijn twee invoervelden die de gebruiker verplicht moet invoeren voordat hij het email wilt wijzigen.

## In Your Colour koppelling (Colourshealth)

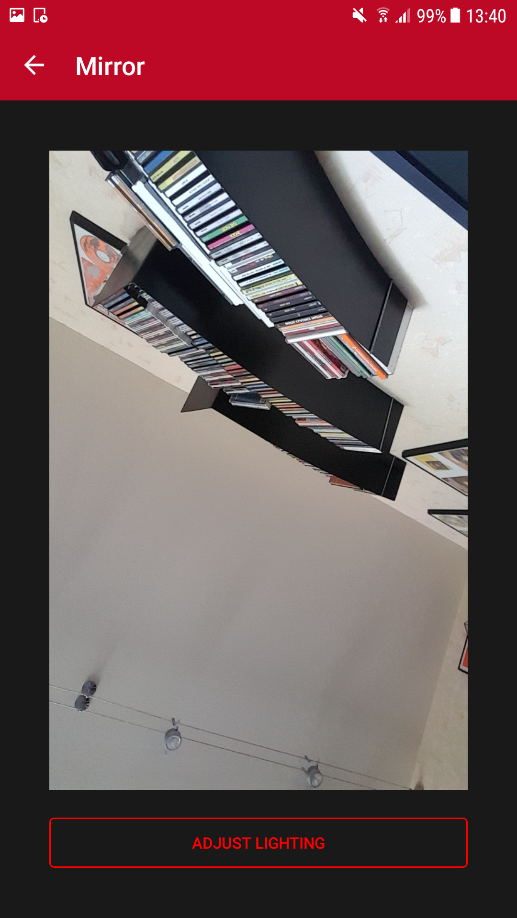
Hieronder zijn de schermafbeeldingen van de In Your Colour functie. Zoals te zien is, verandert de tekst en logo kleur wanneer er een gevoel is gekozen. Hierdoor blijft de tekst goed leesbaar voor de gebruikers. Wanneer er op het logo wordt gedrukt, komt de gebruiker op de webpagina van In Your Colour, waarbij zij meer informatie kunnen vinden over het concept.

****

## Spiegel functie

Hieronder zijn de schermafbeeldingen te zien van de spiegel functie. De twee rechter afbeeldingen laten de spiegel met de lichtgevende rand zien en op de linker afbeelding staat de pagina waar het licht in te stellen is. Via de slider wordt bepaald hoe fel de achtergrond moet zijn.

Afbeelding met muziek

Beschrijving is gegenereerd met hoge betrouwbaarheid  
Afbeelding met schermafbeelding

Beschrijving is gegenereerd met zeer hoge betrouwbaarheid

## Kledingkast uitbreidingen

**Afbeelding met schermafbeelding

Beschrijving is gegenereerd met zeer hoge betrouwbaarheid**

**Filterfunctie**

Er is een kleine wijziging gemaakt met de plaats van de knoppen ten opzichte van de GUI die in het technisch ontwerp zit.

Wanneer gebruikers een filter willen instellen, worden alleen de opties weergegeven die voor de gebruiker van toepassing zijn. Als de gebruiker geen broeken opgeslagen heeft, zal dit ook geen optie zijn voor de gebruiker. Hierdoor voorkomen we enige fouten en maken we het makkelijker voor de gebruiker.

De pickers vullen zichzelf automatisch aan de hand van wat er in de database staat en wat er gekozen is door de gebruiker.

Daarnaast kunnen de filters ook gereset worden, wanneer de gebruiker zijn gehele kledingkast wilt zien.

**Afbeelding met schermafbeelding

Beschrijving is gegenereerd met zeer hoge betrouwbaarheid**

**Kleding toevoegen met maat**

De kleding toevoegen was al een functie die binnen de applicatie zat. Om ervoor te zorgen dat de gebruiker alleen kleding te zien krijgt die dichtbij zijn of haar maat zit, is het nodig geweest om deze functie een klein beetje aan te passen.

De region en gender van de gebruiker worden opgehaald en in de bovenste pickers geplaatst, deze kan de gebruiker ook wijzigen voor het geval dat de kleding uit een ander land komt.

Hierna kunnen zij de maten opgeven van het kledingstuk met een picker die de juiste maten ophaalt.

Afbeelding met schermafbeelding

Beschrijving is gegenereerd met zeer hoge betrouwbaarheid

**Maten opslaan**

Hierlangs is de opmaak te zien van het maat opslaan bij de eerste inlog. Er wordt hierom gevraagd naar het geslacht en de regio van de gebruiker. Aan de hand van de gekozen items, komen er maten in de onderste pickers.

De gebruikers zijn vereist om deze pickers in te vullen en op save te drukken voordat ze naar de volgende pagina gaan.

## Notificaties

Rechtsboven in het menu is een toolbaritem aangemaakt met hierin een icoontje en count voor de notificaties. Wanneer de gebruiker een notificatie binnen heeft gekregen, zal hij of zij dit weten doordat er een nummertje langs het icoontje staat. Wanneer er geen berichten binnen zijn gekomen, zal het nummer niet zichtbaar zijn maar het icoontje wel.

Wanneer een gebruiker op het icoontje klikt, opent de applicatie de notificatielijst. Hierin krijgt de gebruiker de profielfoto, gebruikersnaam van de afzender en een omschrijving te zien, zodat hij weet wat voor soort notificatie het is. Wanneer er op een notificatie geklikt wordt, opent de applicatie de juiste pagina die te maken heeft met de notificatie. Bijvoorbeeld: AliciO wants to be your friend navigeert naar FriendRequests pagina.

Afbeelding met schermafbeelding

Beschrijving is gegenereerd met zeer hoge betrouwbaarheid

## Profiel functies

**Afbeelding met schermafbeelding

Beschrijving is gegenereerd met zeer hoge betrouwbaarheid**

**Profiel opslaan**

Na het opslaan van de maat, zal de gebruiker op deze pagina komen. Hier kunnen zij kiezen om hun profielpagina aan te passen, doormiddel van een foto en omschrijving toe te voegen.

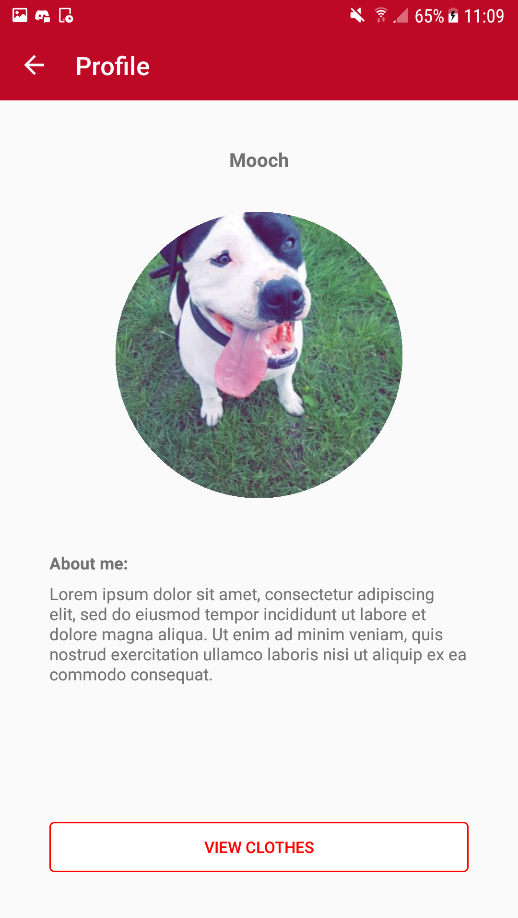
Wanneer de gebruiker dit wilt overslaan, kunnen zij dit doen door rechtsboven op de tekst te klikken.

**Afbeelding met schermafbeelding

Beschrijving is gegenereerd met hoge betrouwbaarheid**

**Profielpagina aanpassen**

Hierlangs is een schermafbeelding te zien van de pagina. Er is te zien dat er twee verschillende knoppen zijn voor het aanpassen van de foto: pick a picture en take a picture. Hierboven staat de afbeelding, deze wordt allereerst opgehaald en is hierna aan te passen. Daarnaast is er een custom renderer gebruikt om afbeeldingen in een ronde vorm weer te geven.



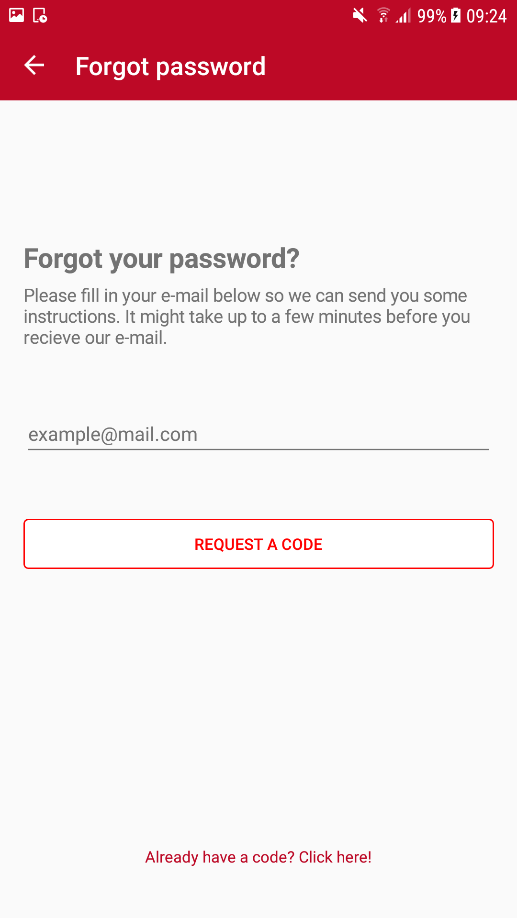
**Profielpagina van vrienden bekijken**

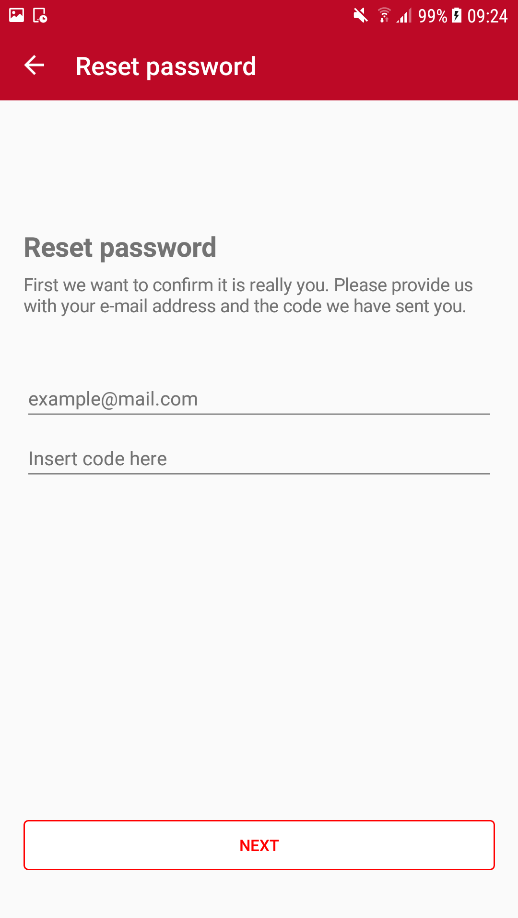
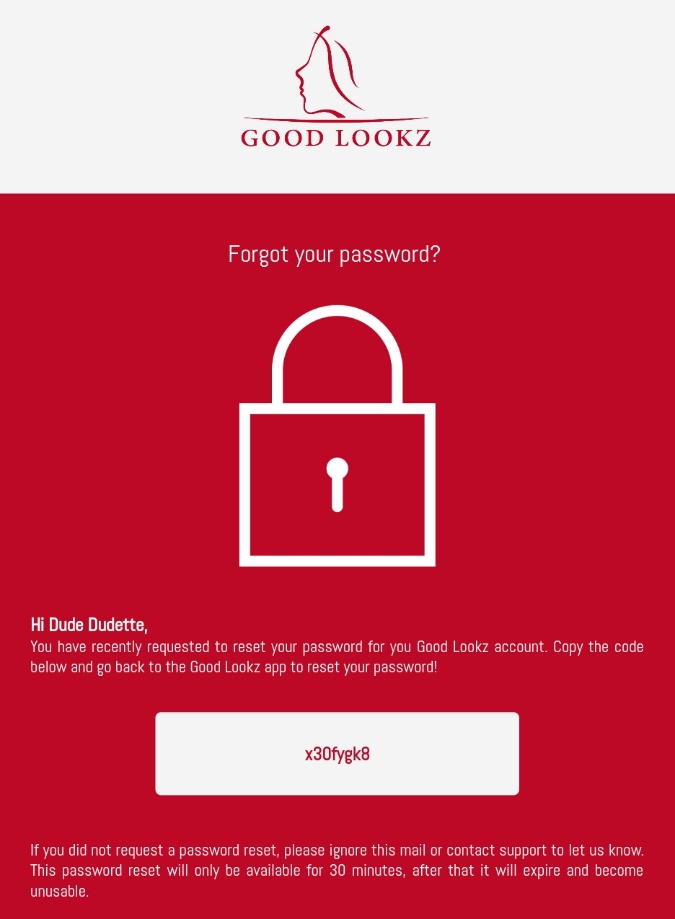
Net zoals bij de voorgaande pagina’s wordt er hier ook gebruik gemaakt van een custom renderer die ervoor zorgt dat de afbeelding een ronde rand heeft. Daarnaast laat de pagina de omschrijving van het profiel zien in een label. Tot slot staat er nog een knop (view clothes) hiermee kunnen gebruikers navigeren naar de kleding kast van een vriend.

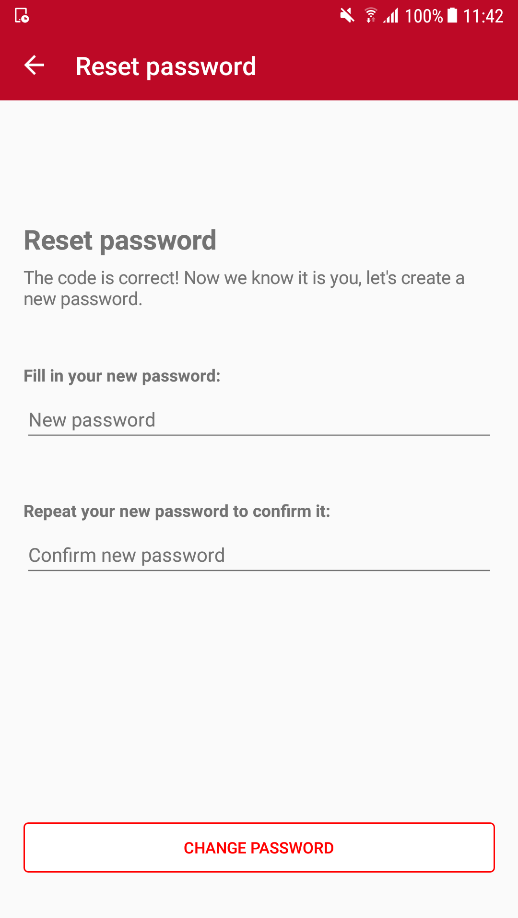
## Wachtwoord vergeten optie

Op de inlogpagina staat een label die de gebruiker kan aanklikken, deze stuurt de gebruiker door naar afbeelding 2. Op deze pagina wordt om een e-mail gevraagd. Er kan op twee verschillende manieren genavigeerd worden naar afbeelding 3. De gebruiker kan zijn e-mail invullen en op “request a code” klikken of op de label “Already have a code? Click here!”. Beide manieren zijn correct. Wanneer de gebruiker zijn e-mail invoert en op “request a code” klikt, krijgt hij een e-mail bericht binnen, deze is te zien op afbeelding 4.

Op afbeelding 3 vraagt de applicatie om een e-mail en code aan de gebruiker. Wanneer deze juist zijn ingevoerd komt de gebruiker op afbeelding 4 terecht. Hier kan het wachtwoord eindelijk veranderd worden.

Afbeelding 1: Afbeelding 2:

Afbeelding 3: Afbeelding 4:

Afbeelding 5

# Nawoord

Hopelijk heeft dit document u voldoende geïnformeerd over de technische documentatie. Als u nog enige opmerkingen en/of vragen heeft, dan kunt u deze doormailen naar het onderstaande mailadres:

* [rebeccabroens@gmail.com](mailto:rebeccabroens@gmail.com)

# Bijlages

## Bijlage 1: Datamodel vergrote versies









## Bijlage 2: Gehele datamodel



## Bijlage 3: Gehele navigatiestructuur