# App werking/Voortgang

## App

### Login

Als de gebruiker de app opent komt hij/zij in aanraking met het login scherm. Dit is omdat de app verwacht dat je het alleen kan gebruiken als je een account hebt. Dit is ook een laag beveiliging. Als de gebruiker inlogt komt er een loading spinner tevoorschijn aangevend dat de app aan het werken is. Dit is een ook een handigheid voor de gebruiker om dat hij/zij weet dan dat de app bezig is en niet vastgevroren

We hebben de volgende test gebruikers:  
**Gebruikersnaam**: test0, **Password**: secret

**Gebruikersnaam:** test1, **Password:** secret

**Gebruikersnaam:** test2, **Password:** secret

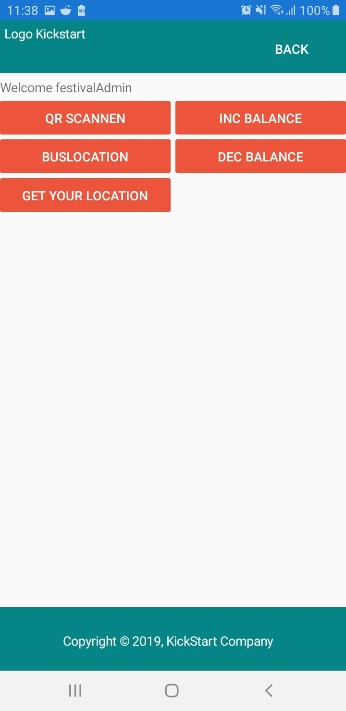
**Gebruikersnaam:** test3, **Password:** secret

**Gebruikersnaam:** test4, **Password:** secret

**Gebruikersnaam:** festivalAdmin, **Password:** admin

We hebben ook nog de volgende foutmeldingen

**HttpStatusCode** : Dit betekent dat er iets is mis gegaan aan de Api kant. We gaan dit nog veranderen naar een betere foutmelding code. Een die de gebruiker kan begrijpen.

**Error:** En we hebben nog een normale afhandeling voor als iets fout gaat dit kan van alles zijn en wordt aan de gebruiker getoond als een alert

### Start Page

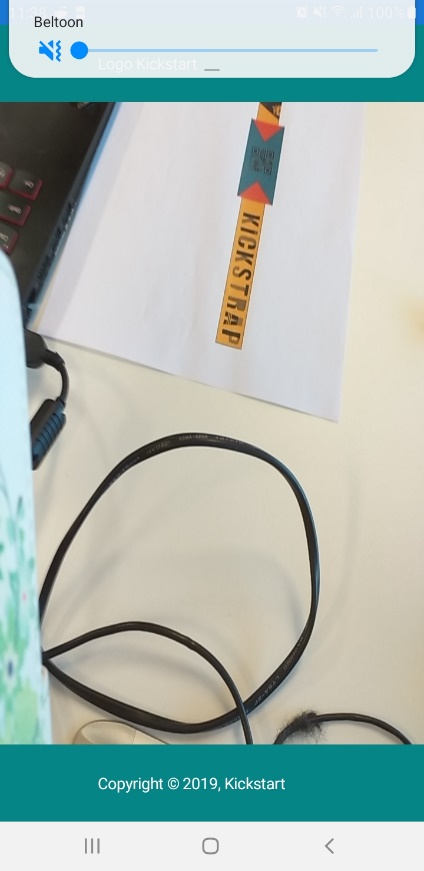
Als de gebruiker is ingelogd dan komt de gebruiker ter recht op de eerste pagina van de app. Genaamd de begin pagina of start page. Hier heeft de gebruiker keuze om verschillende dingen te doen

We geven de gebruiker de keuze om zijn/haar balans op het bandje aanpassen dit kan door er geld van af te halen en terug op je rekening te storten. Of er van je rekening geld bij te storten.

Qr scanner is er alleen voor mensen van het festival. Dit is omdat een gebruiker geen Qr code hoeft te scannen. Medewerkers moeten de Qr codes kunnen scannen om te kijken of een bezoeker het terrein op mag. Dan bij de bar moet nog kunnen om ervoor te zorgen dat je daar met je bandje kan betalen.

Bussen zijn een groot hulpmiddel voor festival bezoekers. De bussen vinden is vaak een opgave en daarnaast ook nog je eigen bus vinden wordt al helemaal erg. Daarom kan je nu via de app de bus locatie openen. Hier mee krijg je een pad via google maps van jouw positie naar de positie van de bus. Dit zou frustratie moet verhelpen en te laat komen bij de bus

Er is ook nog een locatie systeem. Het systeem checkt eerst of er bij de positie van de gebruiker op bij de coördinaten 0 staat. Als dit zo is dan gaat het systeem er van uit dat het de eerste keer is en wordt er aan de gebruiker gevraagd om een locatie toe te voegen. Als het niet waar is dan wordt google maps geopend op de positie van de gebruiker. Een gebruiker kan nog via een button in de header zijn/locatie aanpassen

Vrienden bestaan ook in de app. Gebruikers kunnen elkaar toevoegen. Vervolgens als vrienden dat willen kunnen ze elkaar aan een lijst toevoegen. Via de lijst kan je dan elkaars locatie op het terrein zien. Hierdoor kan je vrienden nooit meer kwijt raken. Of kan je rustig een biertje halen/hapje eten

Als laatste idee hebben we om een soort van systeem te maken waarin de gebruiker kan zien hoe druk het is bij een bar. Dit wordt bepaald door 2 manieren. We hebben het idee om transactie bij te houden van een bepaalde tijd. Ook is het idee om door punten neer te zetten en daar drukte in te meten. Hierdoor hopen we de druktes een beetje te verdelen over de baren.

### Qr scanner

De Medewerkers en eigenaren van festivals hebben toegang tot een Qr scanner in de app. De Qr scanner is bedoeld om de bandjes te kunnen scannen. De Qr codes moeten voor de bezoekers toegang geven tot het festival en betalingen bij de kramen

Een medewerker opent de Qr scanner en krijgt het scherm hiernaast getoond. Een header en footer verrijk komen het scherm met een camera beeld er tussen in

Omdat er geen indicatie is of je de Qr scanner goed scant hebben we ingebouwd dat er een trilling van 2 seconde plaats vindt als je de code hebt gescaned. Voor nu stoppen we de user id in de Qr code om mee te werken

### Locatie

Het locatie systeem is nog niet helemaal af. Momenteel werkt het op tot het openen op google maps. Verder werking staat onder het kopje startpage.

## Api

### Database

Ik weet niet uit mijn hoofd of het werkt maar als je de database wilt bereiken moet je naar het volgende adres

<https://phpmyadmin.hera.fhict.nl/>

Username: dbi403879

Password: Dropzone8

### Api acties

De Api kan via het volgende adres worden benaderd

<http://i403879.hera.fhict.nl/api>

Er zijn de volgende methoden

### Login

<http://i403879.hera.fhict.nl/api/login>

Dit is een post methode. De login methode verwachte 2 parameters. Een username en een password. Hier mee wordt dan gekeken in de database. De username en password moeten worden mee gegeven als Form-urlencoded aan de call

### Users

<http://i403879.hera.fhict.nl/api/users>

De user endpoints hebben de volgende methoden beginnend met de URL hierboven. Deze methode is een get manier. Hiermee krijg je allen users terug die in de database staan. ER is niks extra`s voor nodigen

[http://i403879.hera.fhict.nl/api/users/{id}](http://i403879.hera.fhict.nl/api/users/%7bid%7d)

We hebben tot nu 2 methoden voor users/id

### Get

Hetzelfde geldt voor de alle users endpoints als de gene met een id. Deze methode heeft geen extra headers nodig om te werken en haalt een user op met het id dat je aangeeft. Als er geen user bestaat met het ID geeft ie een 404 not found error

### PUT

Deze method vereist iets meer aandacht maar kan aan worden geroepen met dezelfde UR. Alleen er zijn een aantal extra headers nodig om te slagen

Beginnend hebben we 3 parameters nodig:

Latitude, Longitude, username

string username, double latitude, double longitude

Vervolgens maak je de webclient aan

WebClient webClient = new WebClient();

//Replace the comma in the Longitude and Latitude to avoid Json errors

string Latitude\_Less = latitude.ToString().Replace(',', '.');

string Longitude\_Less = longitude.ToString().Replace(',', '.');

Vervolgens converten we de username naar json om errors te voorkomen

En maken we een json string aan om mee te geven aan de call

string jsonUsername = JsonConvert.SerializeObject(username);

string json = "{\"username\":" + jsonUsername + ","

+ "\"latitude\":" + Latitude\_Less + ","

+ "\"longitude\":" + Longitude\_Less + "}";

De put methode kan voor meerdere doel einden worden gebruikt. We hebben nu ook een methode die de profiel page van een user update alleen het verschil is nu dat alleen de email en username en e.v.t password moet worden geüpgraded.

string jsonUsername = JsonConvert.SerializeObject(username);

string jsonEmail = JsonConvert.SerializeObject(email);

string json;

if (password == "")

{

json = "{\"username\":" + jsonUsername + ","

+ "\"Email\":" + jsonEmail + "}";

}

else

{

json = "{\"username\":" + jsonUsername + ","

+ "\"Email\":" + jsonEmail + ","

+ "\"Password\":" + password + "}";

}

Zo valt op te merken dat het enige verschil is in de request de json data die mee wordt gestuurd. Zo kan je er mee om heen spelen en hoef je niet te verplichten aan een verplichte soort

En dan voeren we de call uit

//Make the call

string webUri = API\_Address + $"users/{id}"

webClient.Headers.Add("Content-Type", "application/json");

## string reply = webClient.UploadString(webUri, "PUT", json);

## Bronnen

<https://github.com/justinvandelaar/PT12-PO6> --> Code statsh

<http://i403879.hera.fhict.nl/> -> Api locatie