



INTERACTIVE WALL

Sander Van Damme
Ilona Zancaner

New Media Development
Web Of Things

Bachelor in de Grafische & Digitale Media
Arteveldehogeschool 2017-18

INHOUDSTABEL

1. Discover.....	5
1.1. Briefing.....	5
2. Define.....	6
2.1. Analyse.....	6
2.2. Planning.....	7
2.3. Ideaboads.....	8
3. Design	13
3.1. Sitemap Mobile Web App	13
3.2. Wireframes Website	14
3.3. Wireflow Mobile Web App	17
3.4. Style Guide	21
3.5. Visual Designs.....	22
4. Develop	28
4.1. Code Snippets.....	28
4.2. Schermafbeeldingen.....	34
5. Deliver.....	34
5.1. Handleiding	34
6. Deploy.....	35
6.1. Deployment Guide	35

1. DISCOVER

1.1. Briefing

De opdracht is om een interactieve muur te ontwikkelen. Dit wordt gedaan met behulp van de Raspberry Pi. De beweging van een fel object dat de gebruiker vasthouwt, wordt opgevangen met de Raspberry Pi camera. Op die manier kan er geschilderd worden op een groot canvas. De data wordt doorgestuurd via PubNub zodat andere gebruikers die niet schilderen de livestream kunnen bekijken. Wanneer iemand klaar is met zijn kunstwerk kan deze worden opgeslagen in de database.

Het doel

Het doel is om de gebruiker een interactieve en leuke belevening te bieden.

Het doelpubliek

De leeftijd van de eindgebruiker zou liggen tussen 6 en 50 jaar. Omdat deze applicatie waarschijnlijk beschikbaar zal staan op de opendeurdag zullen er personen van verschillende leeftijden aanwezig zijn. Zowel mannen als vrouwen behoren tot onze doelgroep. Opleidingsniveau varieert maar heeft in dit geval weinig invloed. Omdat we hoogstwaarschijnlijk niet internationaal gaan zal deze applicatie enkel in België beschikbaar zijn. Daarom zal onze doelgroep enkel Belgen bevatten.

Informatie

Er is geen informatie beschikbaar. Dit is een volledig nieuw concept dus we starten zonder informatie, zowel contextueel als layout/huisstijl.

Het budget

Er is geen budget van toepassing.

Timing

De deadline voor dit project is 12 januari 2018.

2. DEFINE

2.1. Analyse

Wat zullen we de klant bieden?

Een responsive webapplicatie met een gebruiksvriendelijke interface. De backend is makkelijk te beheren en nieuwe administrators kunnen er direct mee aan de slag.

Wat gaan we doen?

Een nieuwe applicatie ontwikkelen

Specificaties

Technische specificaties

- MongoDB
- ExpressJS
- Yarn of NPM
- ES6 of TypeScript
- Sass
- Gulp
- VueJS
- OAuth0

Functionele specificaties

Admin:

- Kunstwerken bekijken, editeren en verwijderen in de backend
- Livestream bekijken

Gebruiker

- Kunstwerken aanmaken en bekijken in de frontend
- Livestream bekijken

Niet-functionele specificaties

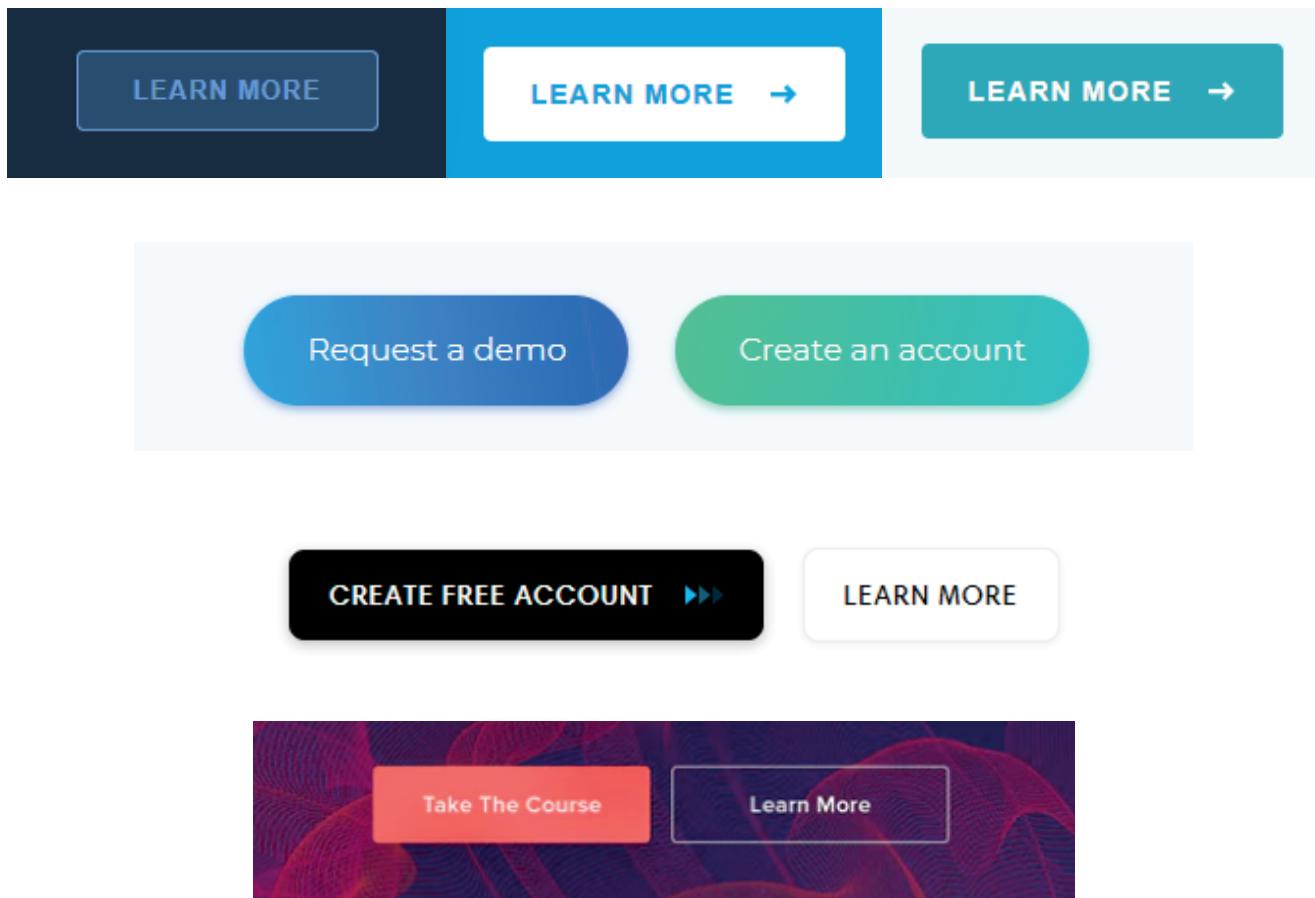
- Responsive
- Esthetisch verantwoord
- Gebruiksvriendelijk

2.2. Planning

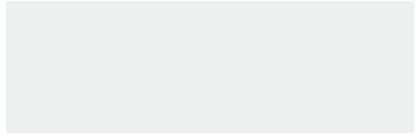
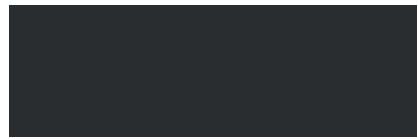
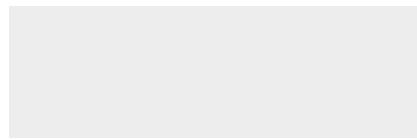
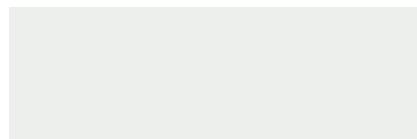
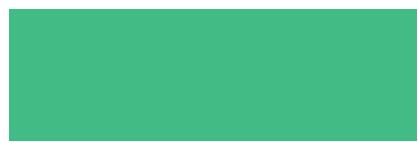
- 21/11/2017 Productiedossier: discover
- 24/11/2017 Productiedossier: define
- 28/11/2017 Milestone 1: productiedossier discover & define klaar**
- 30/11/2017 Styleguide & Sitemap
- 02/12/2017 Wireframes
- 04/12/2017 Visual designs
- 05/12/2017 Milestone 2: productiedossier design klaar**
- 11/12/2017 Styleguide code
- 12/12/2017 Milestone 3: Styleguide (code) en data klaar**
- 15/12/2017 CRUD tekeningen
- 19/12/2017 Backend interface & prototype canvas
- 21/12/2017 Frontend interface
- 26/12/2017 Read & Create vanuit frontend
- 10/01/2018 Productiedossier aanvullen en afwerken
- 11/01/2018 Presentatie voorbereiden
- 12/01/2017 Milestone 4: Presentatie + app indienen**

2.3. Ideaboard

Buttons



Kleur



Productivity Boost



4th December, 2012 • 2 Comments • 2 Shares

Elliot Jackson ([@ eej](#)) created a wonderful blog post about productivity, which I shameless stole.

Productivity is something that we all strive for but can never seem, to fully achieve it (at least that's my case). I have been studying my workflow recently and have noticed a few of things that I was losing a lot of time on. I have also come across apps that have sped me up tenfold. Here are a few tips that I have picked up.

The Air Theme

Lore ipsum dolor sit amet, consectetuer adipiscing elit, sed diam nonummy nibh euismod tincidunt ut laoreet dolore magna aliquam erat volutpat.

[Archive](#) [RSS](#) [Theme](#) [Twitter](#) [Instagram](#)

Photographica

[PHOTO](#) [TEXT](#) [QUOTE](#) [LINK](#) [CHAT](#) [AUDIO](#) [VIDEO](#)

Photographica is a portfolio Tumblr theme for photographers and creative professionals. It's versatile and easy to customize.

BOUGENVILLE

A Beauty & Casual Tumblr Theme



• [Home](#) • [About](#) • [Typography](#) • [Download](#) • [Ask Me](#)

Mimesi.

A bold and yet professional Wordpress portfolio theme for creative folks.

Navigatie

ULTRALINX LINXSPIRATION ARCHIVE RANDOM RSS



Home



Projecten



Diensten



Over mij



Aanpak



Contact



MINIMALISM

ABOUT

INSTALL

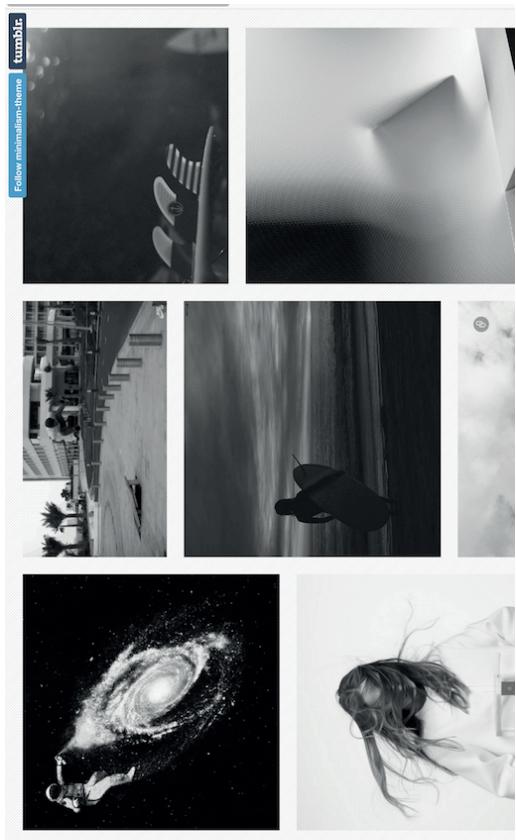
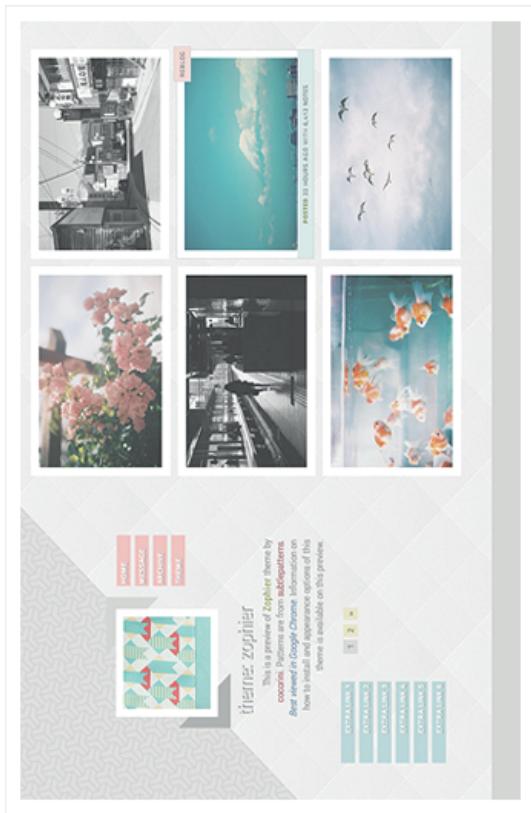
INSTAGRAM

ASK

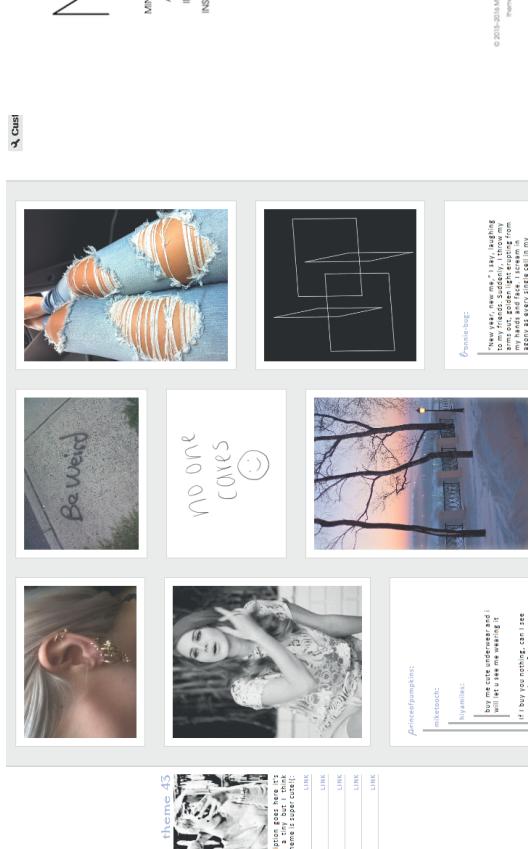
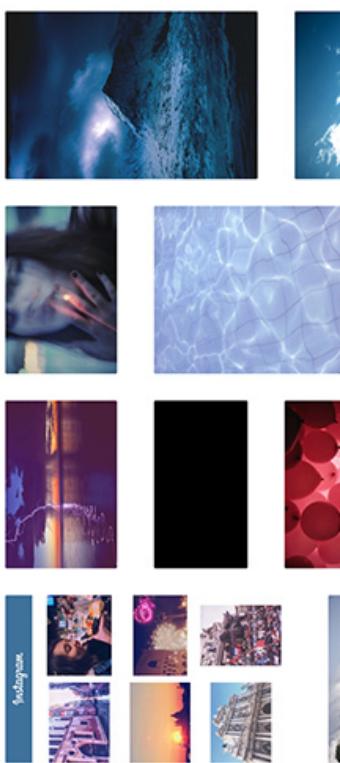


© 2015-2016 Minimalism — Minimalism
theme by BETADELI

Layout



THEMES LTD.



3. DESIGN

3.1. Sitemap Mobile Web App



3.2. Wireframes Website

The wireframe shows a user interface for an 'Interactive Wall' application. On the left is a sidebar with a dark grey header containing a profile picture placeholder and the text 'Interactive Wall'. Below this are five menu items: 'Home' (house icon), 'Interactive Wall' (document icon), 'Paintings' (camera icon), 'Account' (person icon), and 'Roles' (shield icon). The main content area has a light grey header with the title 'Dashboard' and a subtitle 'Paintings'. It features a table with three rows of data:

id	thumbnail	title	artist	actions
1		Mona Lisa	Da Vinci	
2		Het Ordeel van Paris	Rubens	
3		De Sterrennacht	Van Gogh	

Annotations with circled numbers are present: '1' is at the top right of the sidebar header; '2' is at the far right end of the delete icon's row.

Tabel overzicht

Overzicht van alle items in een tabel. De user kan zelf een nieuw item toevoegen of een item bekijken, updaten of verwijderen (soft-delete)

1 Ingelogde user wordt weergegeven. Onder de link zit een dropdown met link naar Profiel en om uit te loggen.

2 Item bekijken - Item bewerken - Item verwijderen

 Interactive Wall

Hello, randomuser! 1

-  Home
-  Interactive Wall
-  Paintings
-  Account
-  Users
-  Roles

Dashboard

Paintings



De Sterrennacht

Painting details

Title	De Sterrennacht
Artist	Van Gogh
URL	Republicans
Created at:	2017-05-14
Updated at:	2017-05-14

 edit  delete

Copyright

Detail

De detailpagina van een item. De user kan de data bekijken en eventueel aanpassen of verwijderen.

1

Ingelogde user wordt weergegeven. Onder de link zit een dropdown met link naar Profiel en om uit te loggen.

The screenshot shows the 'Interactive Wall' application interface. On the left is a sidebar with icons and labels: Home (house), Interactive Wall (circle), Paintings (camera), Account (person), Users (people), and Roles (shield). The main area is titled 'Dashboard' and shows a card for a painting: 'Paintings - "De Sterrennacht"'. It has edit buttons for title, artist, and URL, and an 'Edit' button at the bottom.

1 Hello, randomuser!

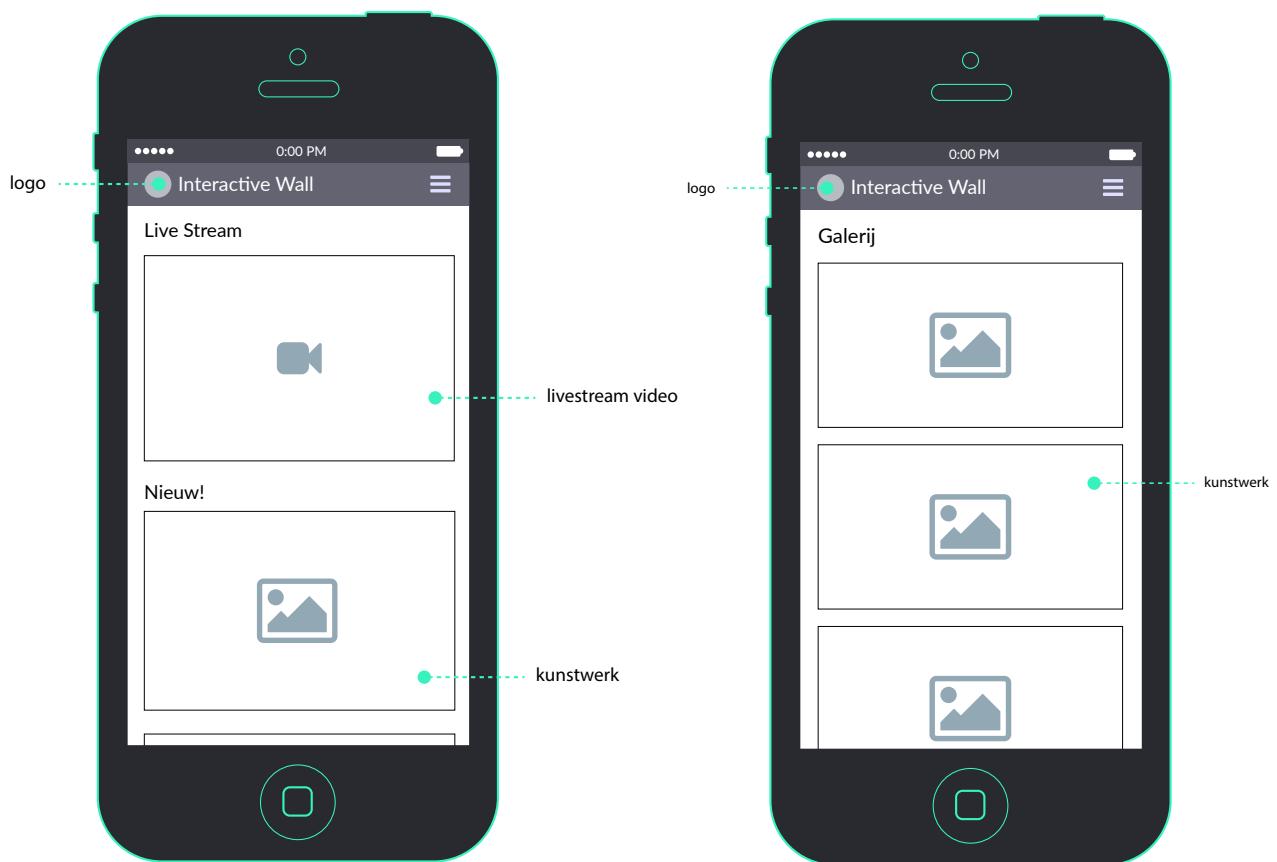
Item bewerken

Formulier om een bestaand item te bewerken.

1 Ingelogde user wordt weergegeven. Onder de link zit een dropdown link naar Profiel en om uit te loggen.

3.3. Wireflow Mobile Web App

Mobile

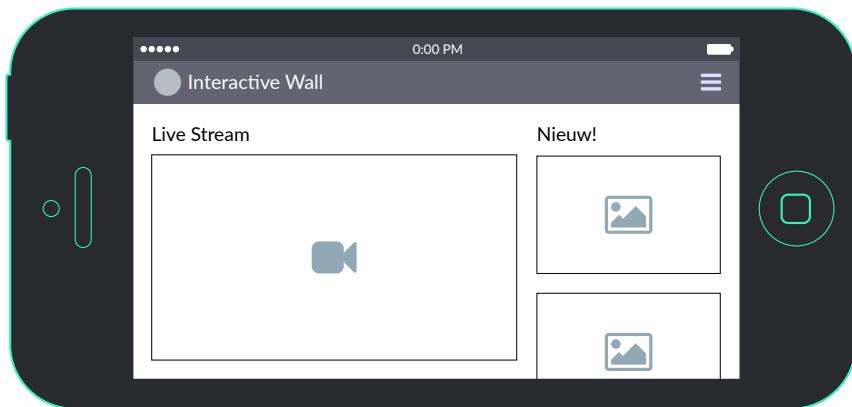


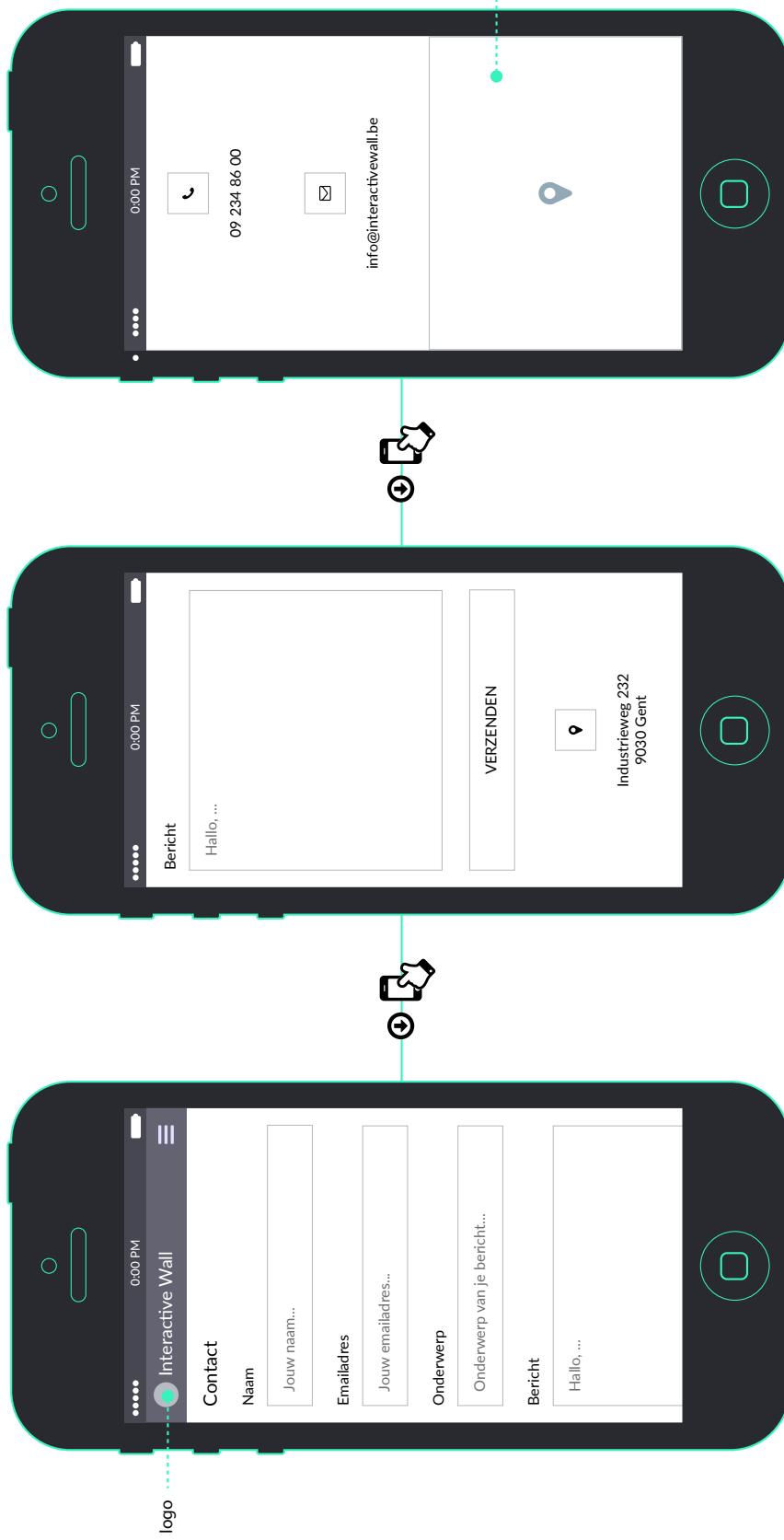
Homepagina met livestream

Bezoekers kunnen op de homepagina de livestream bekijken. De meest recente kunstwerken worden ook getoond.

Galerij

Alle kunstwerken die gemaakt zijn worden hier gedisplayed. Wanneer er op een kunstwerk geklikt wordt, verschijnt de titel en de artiest.





Contactpagina

Op de contactpagina vindt de bezoekers de contactgegevens terug en kan hij een bericht verzenden naar de admins.

Desktop

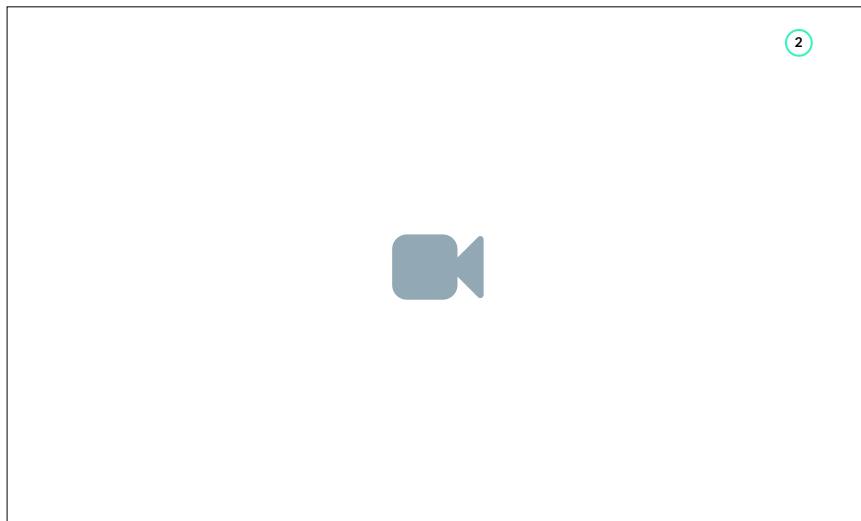


1



Interactive Wall

Live Stream



2

Homepagina met livestream

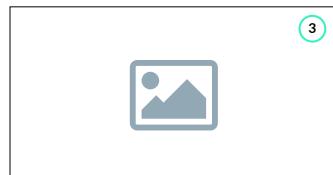
Bezoekers kunnen op de homepage de livestream bekijken. De meest recente kunstwerken worden ook getoond.

1 Dichtgeklapte navigatie

2 Livestream video

3 Kunstwerk

Nieuwste kunstwerken



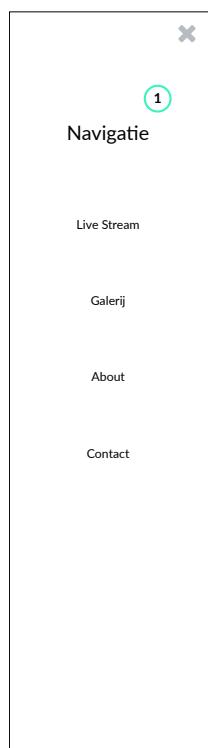
3



3



3

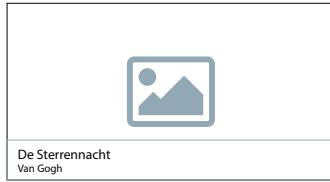
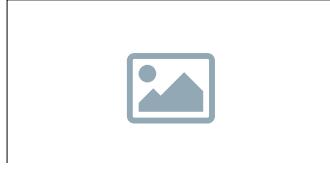


Interactive Wall

Galerij



2

De Sterrennacht
Van Gogh

Galerij

1 Uitgeklapte navigatie

2 Kunstwerk

Alle kunstwerken die gemaakt zijn worden hier gedisplayed. Wanneer er over een kunstwerk gehovered wordt, verschijnt de titel en de artiest.



Contact

Naam



Emailadres



09 234 86 00



info@interactivewall.be

Onderwerp

Bericht



Contactpagina

Op de contactpagina vindt de bezoekers de contactgegevens terug en kan hij een bericht verzenden naar de admins.

1 Dichtgeklapte navigatie**2** Google Maps met marker op het adres

3.4. Style Guide

Logo



Kleuren

primaire kleur



#35e5b3

tekst



#313133

grijze tint voor
borders



#c7d1cd

Typografie

BELANGRIJKE TITELS (H1)

Grote belangrijke titels zijn geschreven in Lato in Heavy, een spatiëring van 1px en hebben een grootte van 40px en hebben als tekenkleur de tekst kleur (#313133)

Ondertitels (h2)

Iets kleinere titels hebben het lettertype Lato in Heavy, een spatiëring van 1px hebben een lettergrootte van 24px en krijgen de primaire kleur (#35e5b3).

Kleine titels voor alinea's en sections (h3)

De kleinste titels hebben het lettertype Lato in Heavy, lettergrootte 18px en hebben dezelfde kleur als de tekst (#313133)

Doorlopende tekst heeft een lettergrootte van 16px, line-height 150%, de tekenkleur is #313133 en het lettertype is Lato in Regular.

Knoppen

Primaire knop



3.5. Visual Designs

Backend

INTERACTIVE WALL

Hello, randomuser!

Home

INTERACTIVE WALL

Paintings

ACCOUNT

Users

Roles

DASHBOARD

Paintings

id	thumbnail	title	artist			
1		Mona Lisa	Da Vinci			
2		Het Ordeel van Paris	Rubens			
3		De Sterrenacht	Van Gogh			

INTERACTIVE WALL

Hello, randomuser!

Home

INTERACTIVE WALL

Paintings

ACCOUNT

Users

Roles

DASHBOARD

Paintings

DE STERRENNACHT

Painting details

Title	De Sterrenacht
Artist	Van Gogh
URL	http://imgur.com
Created at:	2017-05-14
Updated at:	2017-05-14

EDIT **DELETE**

INTERACTIVE WALL

Hello, randomuser!

 Home

INTERACTIVE WALL

 Paintings

ACCOUNT

 Users

 Roles

DASHBOARD

Paintings - "De Sterrennacht"

 Edit item "De Sterrennacht"

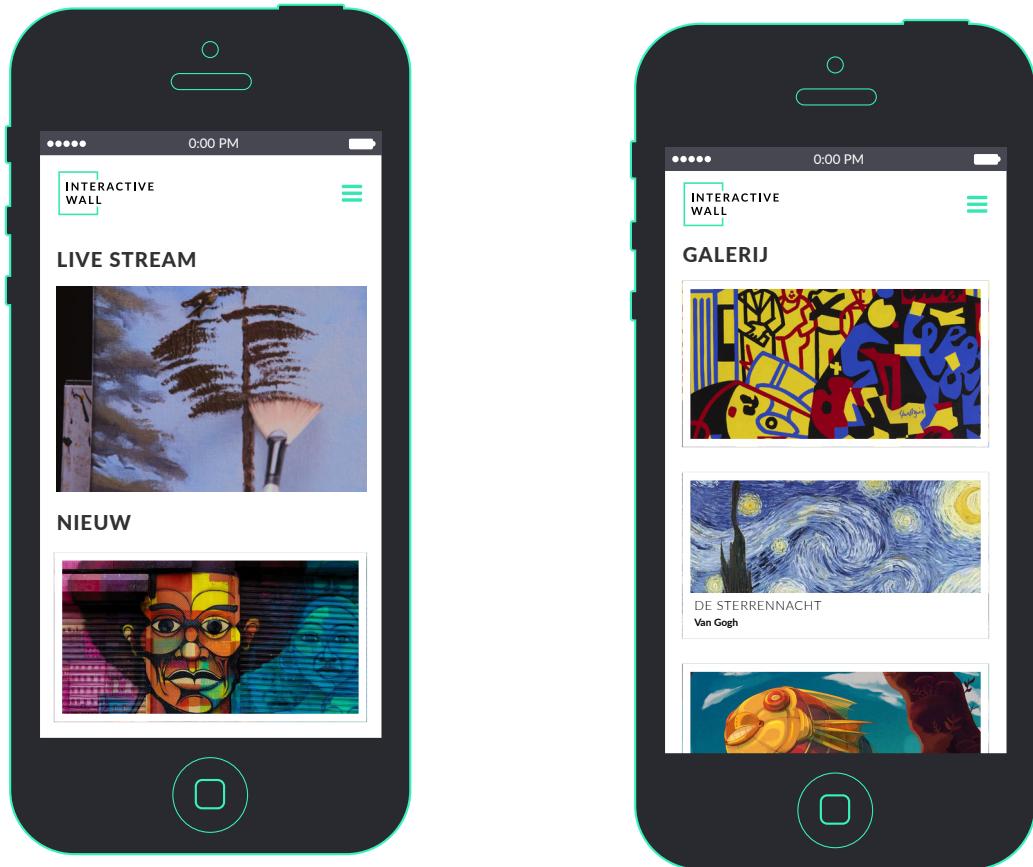
Title

Artist

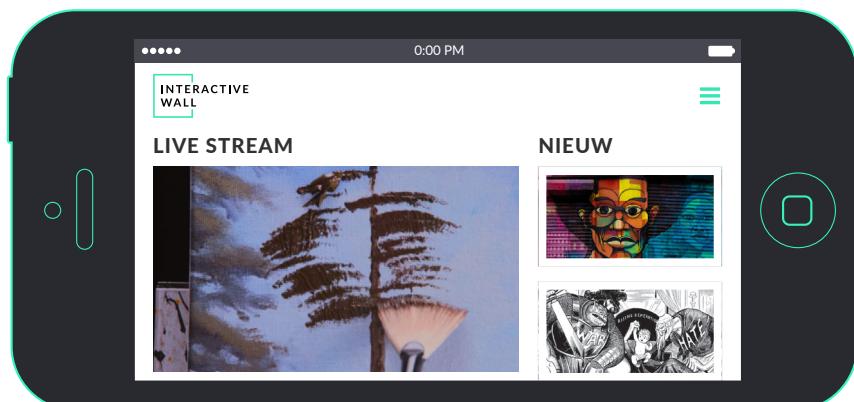
URL

EDIT ITEM

Mobile App - Mobile



C





Mobile App - Desktop



INTERACTIVE
WALL

LIVE STREAM



NIEUWSTE KUNSTWERKEN

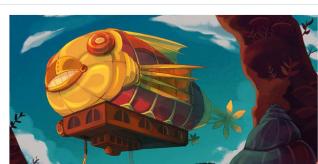


X

NAVIGATIE

- Live Stream
- Galerij
- About
- Contact

GALERIJ





CONTACT

Naam

Emailadres

Onderwerp

Bericht

VERZENDEN



Industrieweg 232
9030 Gent



09 234 86 00



info@interactivewall.be



4. DEVELOP

4.1. Code Snippets

```
JS PaintingsController.js ✘
1  'use strict';
2
3
4  var mongoose = require('mongoose'),
5      Painting = mongoose.model('Painting');
6
7  exports.paintingindex = function(req, res) {
8      Painting.find({}, function(err, task) {
9          if (err)
10              |  res.send(err);
11          res.json(task);
12      });
13  };
14
15  exports.paintingshow = function(req, res) {
16      Painting.findById(req.params.id, function(err, task) {
17          if (err)
18              |  res.send(err);
19          res.json(task);
20      });
21  };
22
23  exports.paintingcreate = function(req, res) {
24      let new_painting = new Painting(req.body);
25      new_painting.save(function(err, task) {
26          if (err)
27              |  res.send(err);
28          res.json(task);
29      });
30  };
31
32  exports.paintingedit = function(req, res) {
33      let obj = req.body;
34      let id = req.params.id;
35      Painting.findByIdAndUpdate(req.params.id, obj, function(err, task){
36          if (err)
37              |  res.send(err);
38          res.json(task);
39      })
40  };
41
```

```

▼ PaintingEdit.Vue ✘
  6   <form method="PUT" @submit.prevent="onSubmit">
  7     <label for="title">Titel</label>
  8     <input type="text" id="title" name="title" placeholder="Titel van je kunstwerk..." v-model="painting.title">
  9     <label for="artist">Artiest</label>
 10    <input type="text" id="artist" name="artist" placeholder="Je naam/pseudo..." v-model="painting.artist">
 11    <label for="imageUrl">URL</label>
 12    <input type="text" id="imageUrl" name="imageUrl" placeholder="URL..." v-model="painting.imageUrl">
 13    <button class="btn smallbtn" type="submit">Wijzigen</button>
 14  </form>
 15 </div>
 16 </div>
 17 </template>
 18
 19
 20 <script>
 21 import axios from 'axios';
 22 import Vue from 'vue';
 23
 24 export default {
 25   name: 'PaintingEdit',
 26   data () {
 27     return {
 28       painting: {},
 29       layout: 'dashboard'
 30     }
 31   },
 32   mounted () {
 33     axios.get('http://localhost:3000/paintings/' + this.$route.params.id)
 34     .then(response => {
 35       console.log(response)
 36       this.painting = response.data;
 37     });
 38   },
 39   methods: {
 40     onSubmit(){
 41       let editpainting = this.painting
 42
 43       console.log(editpainting);
 44       axios.put('http://localhost:3000/paintings/' + this.$route.params.id, editpainting)
 45         .then((response) => {
 46           console.log(response);
 47         })
 48         .catch((error) => {
 49           console.log(error);
 50         })
 51     }
 52   }
 53 }
 54 </script>

```

```
JS server.js ×  
1  let express = require('express'),  
2  app = express(),  
3  port = process.env.PORT || 3000,  
4  mongoose = require('mongoose'),  
5  Painting = require('./api/models/Painting'),  
6  bodyParser = require('body-parser');  
7  
8  mongoose.Promise = global.Promise;  
9  mongoose.connect('mongodb://localhost/InteractiveWallDb');  
10 app.use(bodyParser.urlencoded({ extended: true }));  
11 app.use(bodyParser.json());  
12  
13 app.use(function (req, res, next) {  
14   res.setHeader('Access-Control-Allow-Origin', 'http://localhost:8081');  
15   res.setHeader('Access-Control-Allow-Methods', 'GET, POST, OPTIONS, PUT, PATCH, DELETE');  
16   res.setHeader('Access-Control-Allow-Headers', 'X-Requested-With,content-type');  
17   res.setHeader('Access-Control-Allow-Credentials', false);  
18   next();  
19 });  
20  
21 let routes = require('./api/routes/PaintingRoutes');  
22 routes(app);  
23 app.listen(port);  
24  
25 console.log('painting List started on:' + port);|
```

```

tracking.py x

33 while True:
34     frame = vs.read()
35     frame = imutils.resize(frame, width=500)
36
37     hsv = cv2.cvtColor(frame, cv2.COLOR_BGR2HSV)
38     mask = cv2.inRange(hsv, greenLower, greenUpper)
39     mask = cv2.erode(mask, None, iterations = 2)
40     mask = cv2.dilate(mask, None, iterations = 2)
41
42     # (x, y) center of the ball
43     cnts = cv2.findContours(mask.copy(), cv2.RETR_EXTERNAL, cv2.CHAIN_APPROX_SIMPLE)[-2]
44     center = None
45
46     if len(cnts) > 0 :
47         c = max(cnts, key = cv2.contourArea)
48         ((x, y), radius) = cv2.minEnclosingCircle(c)
49         M = cv2.moments(c)
50         center = (int(M["m10"] / M["m00"]), int(M["m01"] / M["m00"]))
51
52         if radius > 5:
53             cv2.circle(frame, (int(x), int(y)), int(radius), (0, 255, 255), 2)
54
55             cv2.circle(frame, center, 5, (0, 0, 255), -1)
56
57     pts.appendleft(center)
58
59     # loop over the set of tracked points
60     for i in xrange(1, len(pts)):
61         if pts[i - 1] is None or pts[i] is None:
62             continue
63
64         thickness = int(np.sqrt(args["buffer"] / float(i + 1)) * 2.5)
65         cv2.line(frame, pts[i - 1], pts[i], (0, 0, 255), thickness)
66
67     # flip the frame
68     frame = cv2.flip(frame, 1)
69
70     # show coords
71     if 'x' in locals():
72         cv2.putText(frame, "X: " + str(x), (10, 10), cv2.FONT_HERSHEY_SIMPLEX, 0.35, (0, 0, 255), 1)

```

```

  80 },
  81     mounted() {
  82         // ===== scaling =====
  83         // Raspberry: 500 * 370
  84         // 500 -> 100%
  85         // 5 -> 1%
  86         // 370 -> 74% (-> 370 / 5)
  87
  88         var canvasWidth = document.getElementById('canvas').clientWidth;
  89         this.scaling = (((canvasWidth - this.canvassize.width) / 5) / 100) + 1;
  90
  91         // Make it happen when people resize
  92         window.addEventListener('resize', () => {
  93             var canvasWidth = document.getElementById('canvas').clientWidth;
  94             this.scaling = (((canvasWidth - this.canvassize.width) / 5) / 100) + 1;
  95         });
  96
  97         // =====
  98
  99         this.$pubnub.subscribe({
100             channels: ['canvas-data', 'controller-data']
101         });
102
103         this.$pubnub.addListener({
104             message: message => {
105                 if (message.channel == 'controller-data')
106                 {
107                     this.controller = message.message;
108
109                     if (this.controller.drawing && this.controller.erasing)
110                     {
111                         // Not allowed
112                         this.controller.drawing = null;
113                         this.controller.erasing = null;
114                     }
115
116                     if (this.controller.direction != null)
117                     {
118                         switch(this.controller.direction)
119                         {
120                             case 'in':

```

```
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
```

```
    var canvas = document.getElementById('canvas');
    var context = canvas.getContext('2d');

    // Show the position of the balloon on canvas for references sake
    this.brush.position = [this.reverse(message.message[0][0]), message.message[0][1]];
    this.brush.cursorstyle = {
        'width': ((this.brush.size * 1.5) * this.scaling) + 'px',
        'height': ((this.brush.size * 1.5) * this.scaling) + 'px',
        'left': (this.brush.position[0] * this.scaling) + 'px',
        'top': (this.brush.position[1] * this.scaling) + 'px',
        'margin-top': '-' + (((this.brush.size * 1.5) / 2) * this.scaling) + 'px',
        'margin-left': '-' + (((this.brush.size * 1.5) / 2) * this.scaling) + 'px',
    };

    // If drawing or erasing, draw the line
    if (this.controller.drawing || this.controller.erasing)
    {
        var coords1 = [this.reverse(message.message[0][0]), message.message[0][1]];
        var coords2 = [this.reverse(message.message[1][0]), message.message[1][1]];

        context.beginPath();
        context.strokeStyle = this.colors[this.activecolor[0]][this.activecolor[1]];
        context.lineWidth = this.brush.size;

        if (this.controller.erasing)
        {
            context.strokeStyle = this.canvasbackground;
        }

        if (this.prevcoords != null)
        {
            context.moveTo(this.prevcoords[0], this.prevcoords[1]);
            context.lineTo(coords1[0], coords1[1]);
        }

        context.moveTo(coords1[0], coords1[1]);
        context.lineTo(coords2[0], coords2[1]);

        context.lineCap = 'round';
        context.stroke();
    }
}
```

4.2. Schermafbeeldingen

5. DELIVER

5.1. Handleiding

Backoffice

Authenticatie

Om toegang te krijgen tot de backoffice moet je eerst inloggen. De admins zijn opgeslagen in de database dus enkel deze accounts kunnen inloggen.

Create - Read - Update - Delete

Items in de database wijzigen is heel vanzelfsprekend. In de tabel op de overview pagina kun je klikken op een van de iconen naast de details.

Een item bekijken

Op de detail pagina kan je de informatie van het gekozen item bekijken. Er zijn knoppen verzen om het item te bewerken of te verwijderen.

Een item wijzigen

Om een item te updaten, wijzig je de informatie die in de velden van het formulier worden getoond. Daarna klik je op "Wijzigen" en de nieuwe informatie wordt opgeslagen.

Een item verwijderen

Wanneer je klikt op dit icoon wordt er gevraagd naar een bevestiging om dit item te verwijderen. Klik je op "Ja" dan wordt deze definitief uit de database verwijderd.

6. DEPLOY

6.1. Deployment Guide

Backoffice

Frontend

1. In console

```
$ cd vue
```

```
$ npm install
```

```
$ npm run build
```

2. Kopieer index.html en /dist/ folder in de root van je website directory op de server