

ALSET

Julien Cambier



Index

Briefing	5
Analyse	7
Planning	8
Sitemap	10
Wireframes	11
Wireflow	12
Style Guide	13
Visual Designs	14
Code Snippets	16-17
Schermafbeeldingen	18
Handleiding	19
Deployment Guide	21

Discover

- A L S E T -

Briefing

"Hoe kan een CMS-gebaseerd softwaresysteem het delen van voertuigen faciliteren?"

Voor het olod CMS Development moet ik rond deze probleemstelling een CMS applicatie bouwen. De Back-end werkt met Drupal en de front werkt via een API met vue.js.

Define

- A L S E T -

Analyse

Database:

Auto:

- Model
- Name
- Age
- Current mileage
- Power Output (in kW)
- Range (km)
- Top Speed
- Acceleration (0-100km/h)
- Number of Doors
- Number of Seats
- Photo
- Price per day
- Pick-up Location
- Availability (max 2 two days in a row, only week-ends, ...)
- Owner
- Rating
- Reviews
- General state of the car

User:

- name
- email
- pass
- image
- address
- city
- country
- Phone

Reservation:

- User
- Car
- Start Date
- End Date
- Price/Day
- Total Price

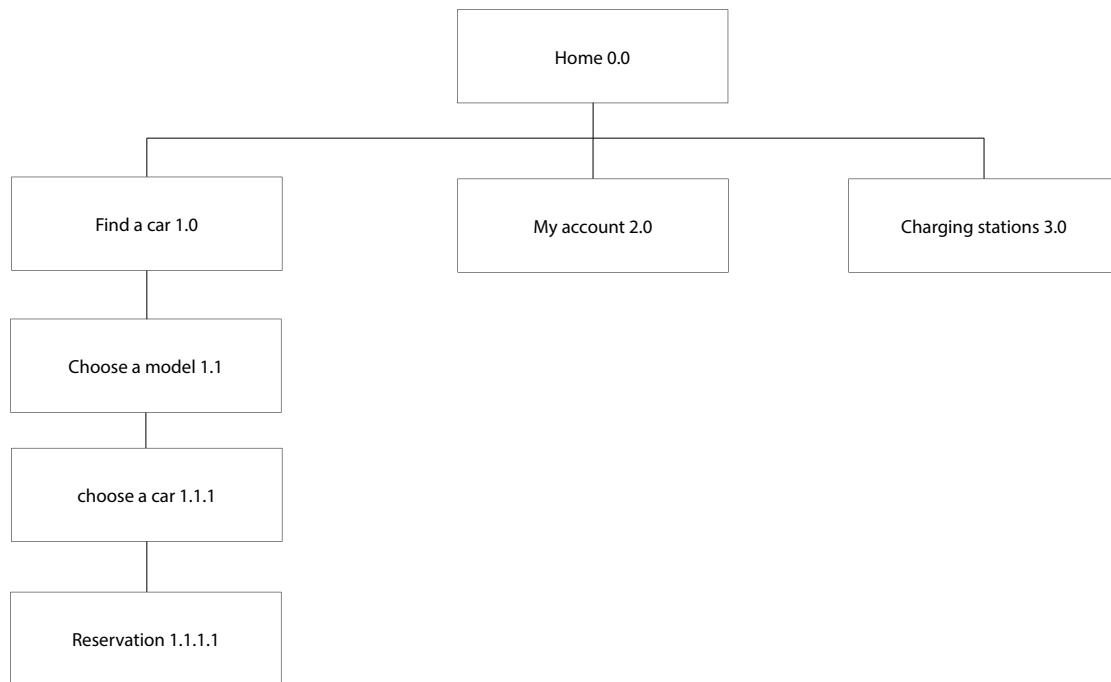
Planning

1. Installeren Drupal
2. Database opstellen met Drupal
3. API opstellen
4. Mobile Web App
 - 4.1. Vue.js project opstarten
 - 4.2. Templates maken
 - 4.3. Templates opvullen met data via API
 - 4.4. Aanmaken van nieuwe data via API
 - 4.5. Aanpassen van data via API
 - 4.6. Verwijderen van data via API

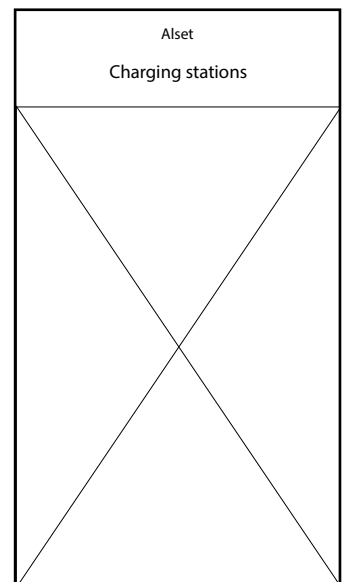
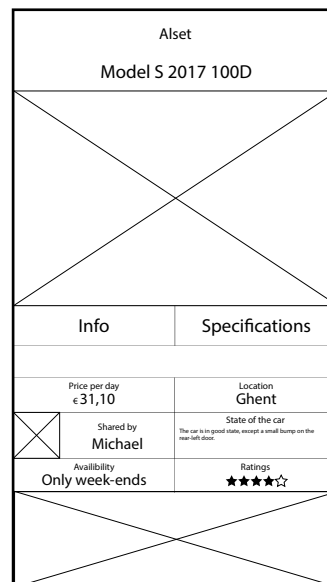
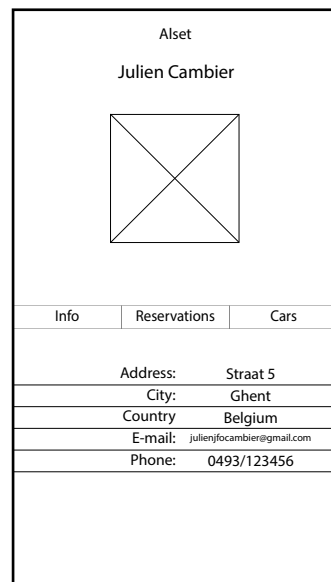
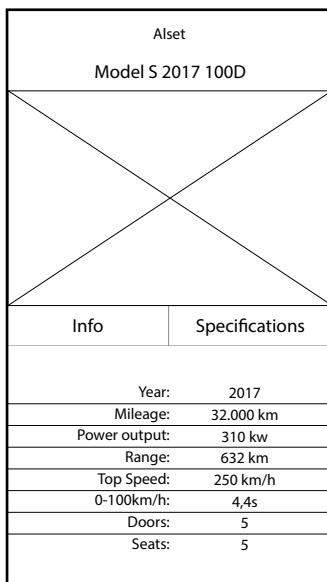
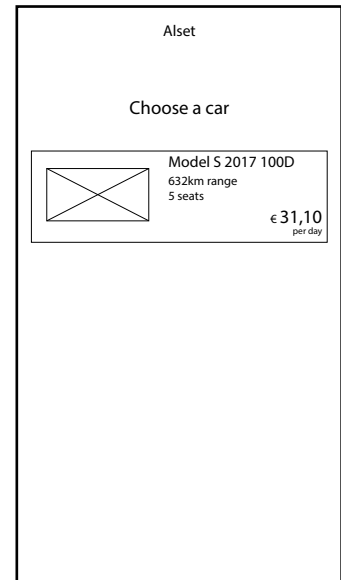
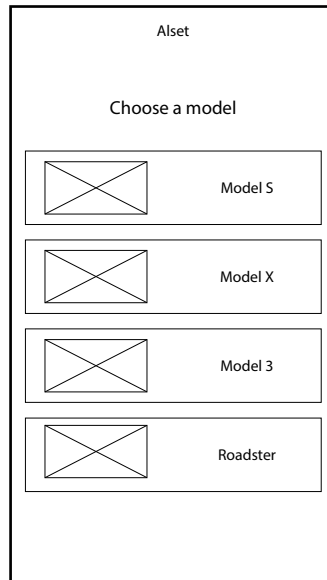
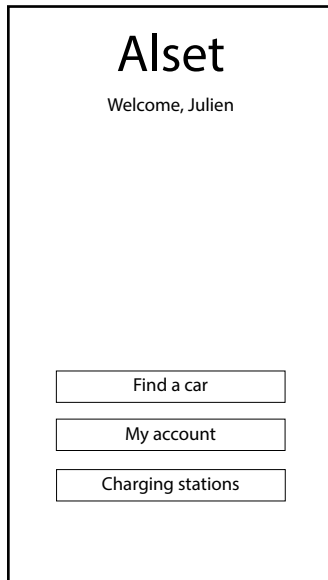
Design

- A L S E T -

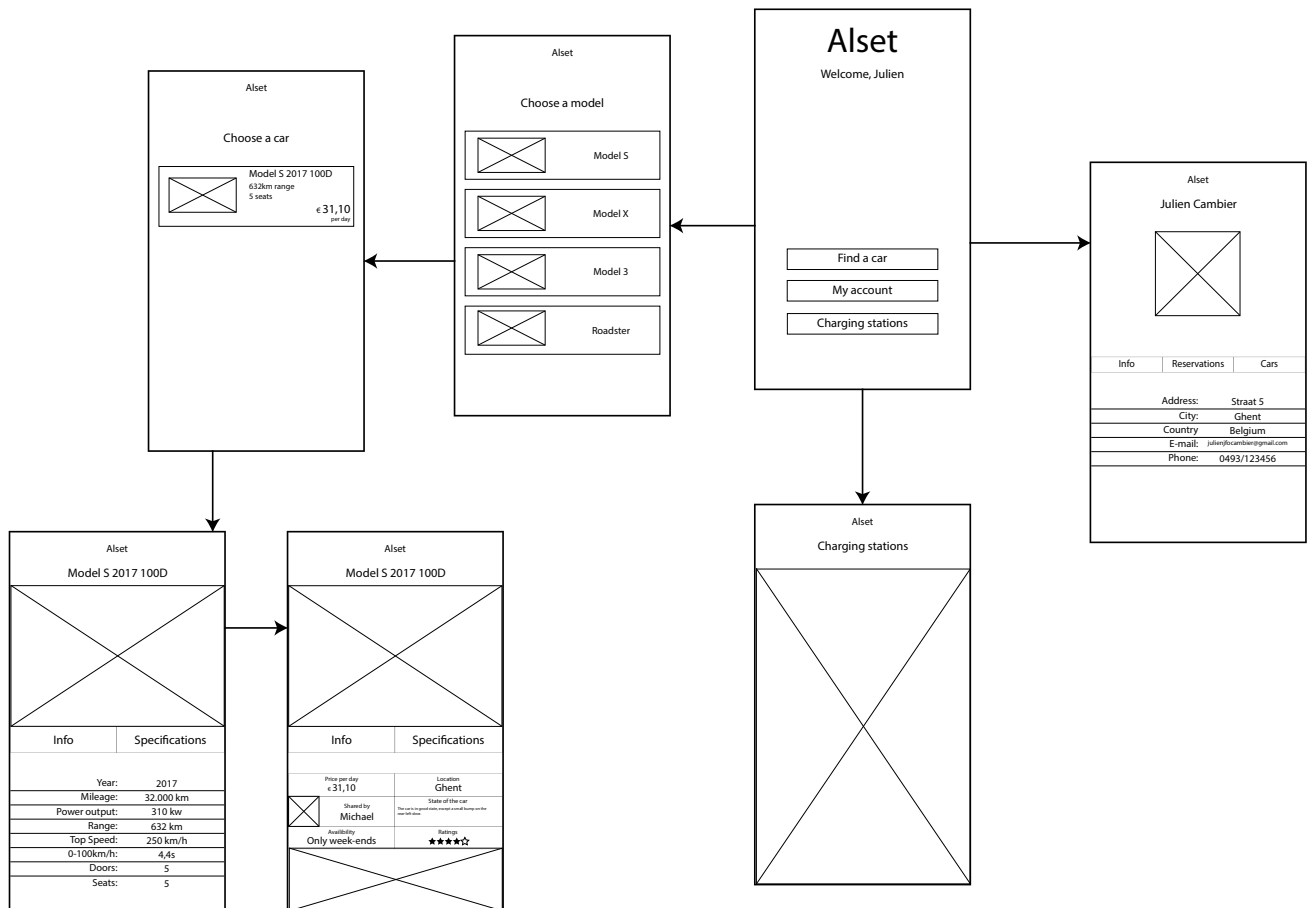
Sitemap



Wireframes



Wireflow



Style guide

Logo:



Typografie:

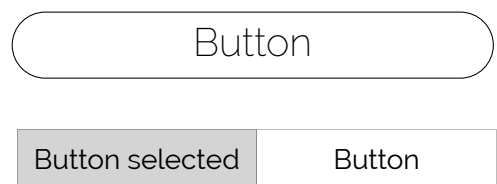
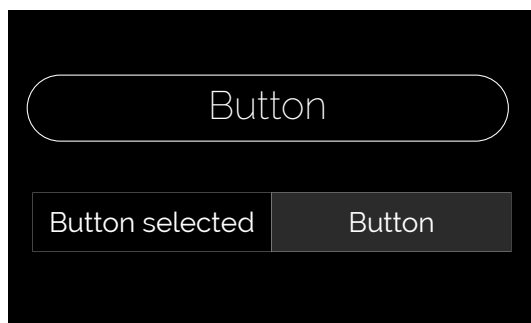
ALSET
zelda - 46pt

Alset
Raleway ExtraLight - 30pt

Alset
Raleway SemiBold - 18pt

Alset
Raleway Regular - 12pt

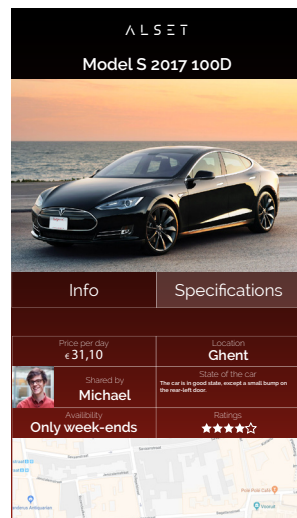
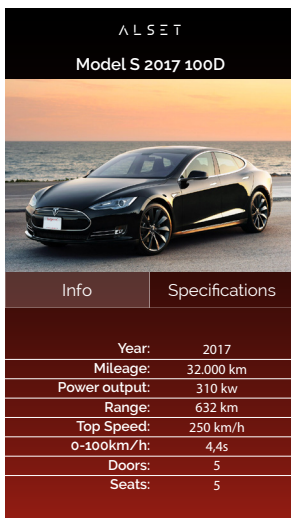
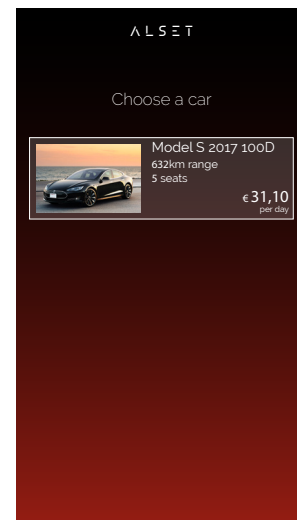
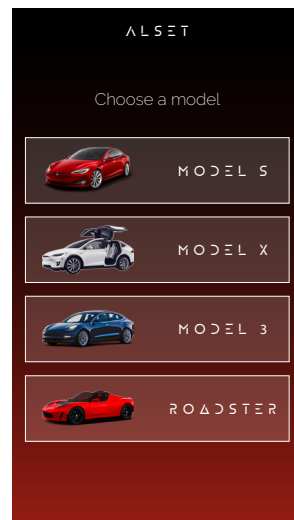
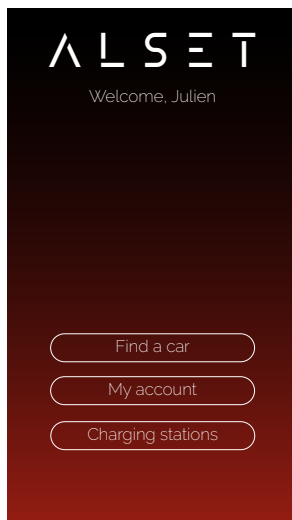
Componenten:



Kleuren:



Visual Designs



Develop
- A L S E T -

Code snippets

[Vue file]

Hier haal ik de verschillende automodellen via de API. Dan geraken die in 'results'.

Daarna wordt de data ingevuld in de template. Als er op een model geklikt wordt gaat men naar een andere template met het 'id' van het model.

```
1  <template>
2    <div class="search">
3      <h1 class="logo">Alset</h1>
4      <h3 class="subtitle">Choose a model</h3>
5      <router-link v-for="model in results" :key="model.id" :to="{ name: 'Cars', params: { id: model.id }}">
6        <div class="card" :id="model.id">
7          
8          <h2>{{model.name}}</h2>
9        </div>
10     </router-link>
11   </div>
12
13 </template>
14
15 <script>
16   import axios from 'axios'
17
18   export default {
19     name: 'models',
20     data () {
21       return {
22         results: [],
23         APIurl: window.APIurl
24       }
25     },
26     mounted () {
27       axios({ method: 'GET', url: window.APIurl + '/models' }).then(result => {
28         this.results = result.data
29         console.log(this.results[0].field_image)
30       }, error => {
31         console.error(error)
32       })
33     }
34   }
35 </script>
```

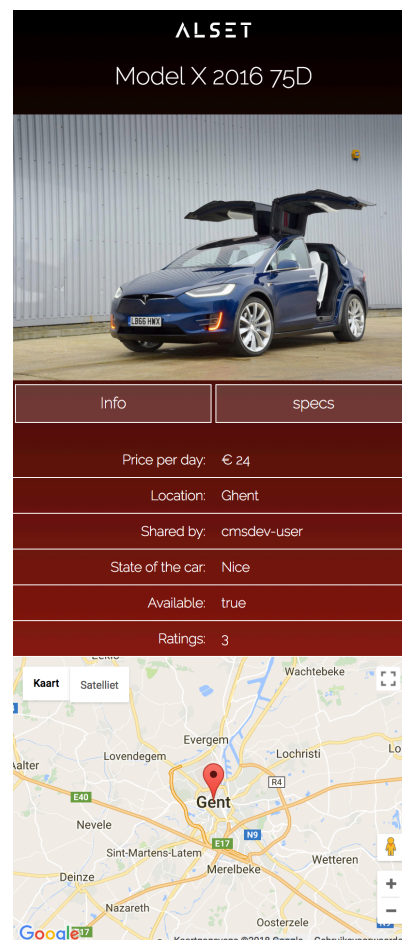
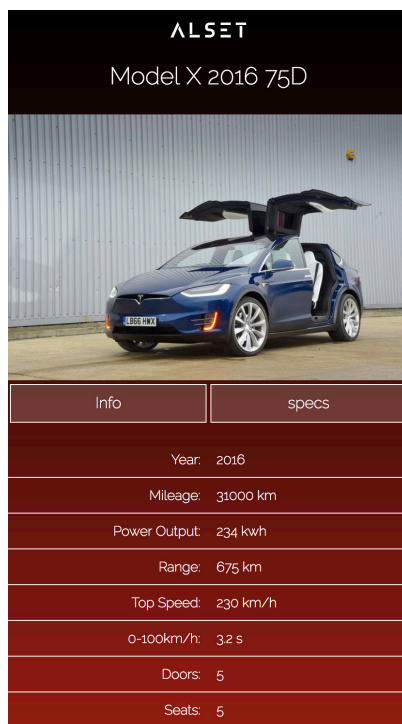
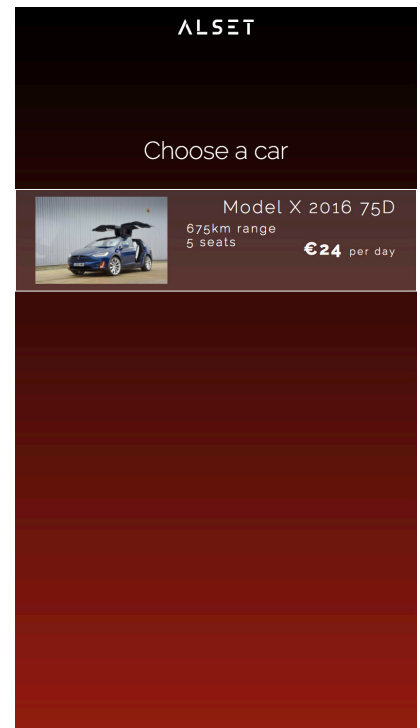
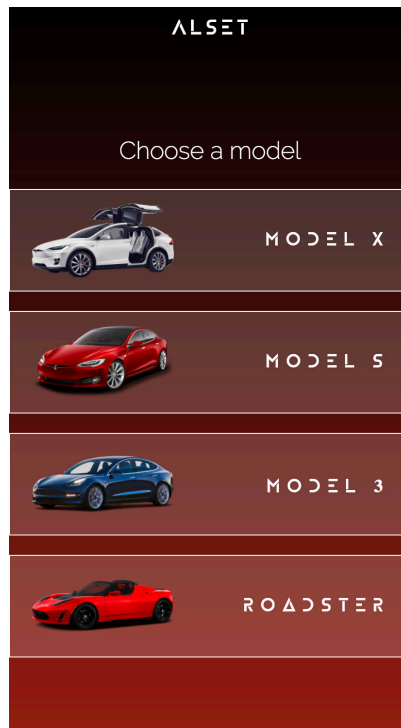
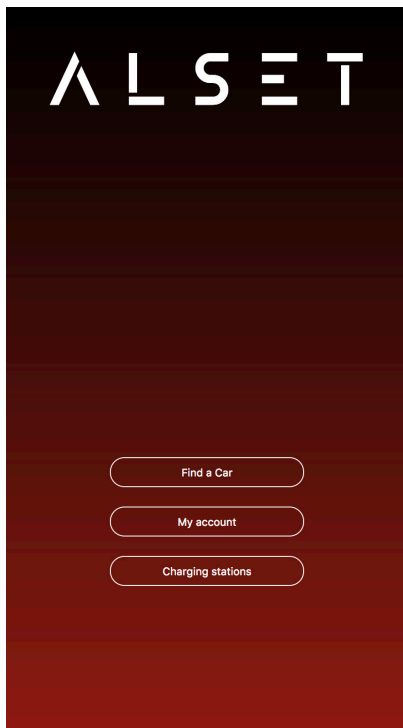

[Vue file]

Hier gebeuren er meerdere dingen:

- We halen het id van de gekozen auto.
- We halen via de API en met het id de data van de auto.
- We halen informatie via de API van de eigenaar van de auto.
- We halen de locatie van de auto en geven die door aan de Google Maps Geocode API om de coördinaten te krijgen. De coördinaten steken we in 'center' en 'markers' om die te tonen op een Google Maps kaart.

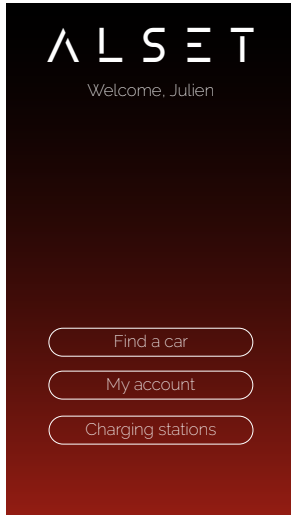
```
94 export default {
95   name: 'detailsview',
96   data () {
97     return {
98       car: [],
99       APIurl: window.APIurl,
100       tab: false,
101       user: [],
102       center: {lat: 10.0, lng: 10.0},
103       markers: [{
104         position: {lat: 10.0, lng: 10.0}
105       }]
106     }
107   },
108   mounted () {
109     // this.modelId = this.$route.params.id
110     this.modelId = this.$route.params.id
111     axios({ method: 'GET', url: window.APIurl + '/cars?id=' + this.modelId }).then(result => {
112       this.car = result.data[0]
113       axios({
114         method: 'GET',
115         headers: {'Authorization': 'Basic Y21zZGV2LXVzZXI6Y21zZGV2LXBhc3M='},
116         url: window.APIurl + '/user/' + this.car.field_owner[0].target_id + '?_format=json'
117       }).then(result => {
118         this.user = result.data
119         axios({
120           method: 'GET',
121           url: 'https://maps.googleapis.com/maps/api/geocode/json?address=' + this.car.field_city[0].value + '&key='
122         }).then(result => {
123           this.markers[0].position.lat = result.data.results[0].geometry.location.lat
124           this.markers[0].position.lng = result.data.results[0].geometry.location.lng
125           this.center.lat = result.data.results[0].geometry.location.lat
126           this.center.lng = result.data.results[0].geometry.location.lng
127         }, error => {
```

Schermafbeeldingen



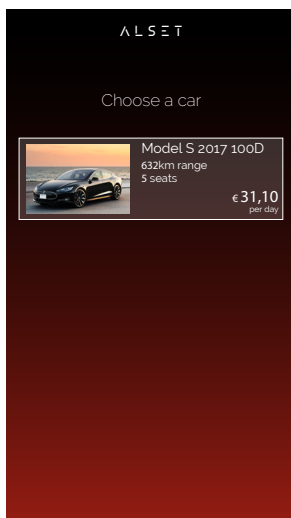
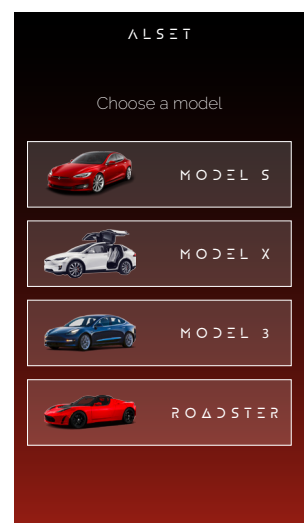
Handleiding

- ALSET -



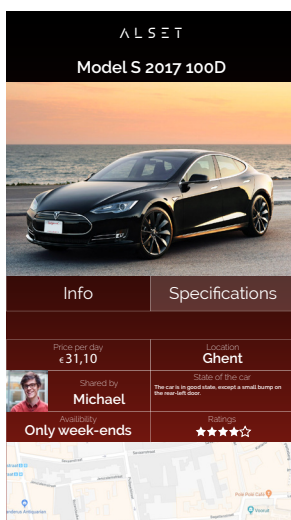
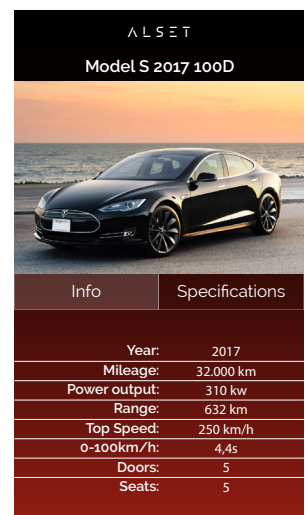
Voor een auto te vinden moet men op 'Find A Car' klikken

Hier ziet men alle beschikbare modellen. Klik op een model.

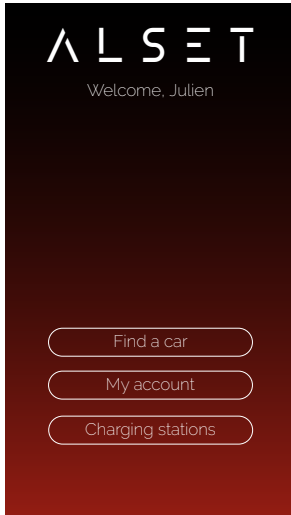


Dit zijn alle beschikbare auto's van dit bepaald model. Klik op een auto.

Alle specificaties van de auto zijn weergegeven, maar als men op Info klikt ziet men ook de algemene informatie

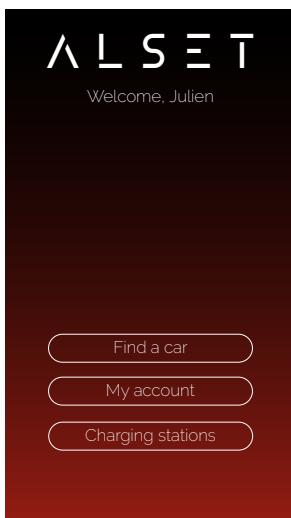


Hier ziet men de prijs, eigenaar, locatie, enz...



Voor ons account te zien klikt men op 'My Account'

Hier ziet men alle onze info, onze res-
ervaties en onze auto's.



Voor ons de herlaadpunten te vinden
klikt men op 'Charging Stations'

Dit zijn alle herlaadpunten rond de
gebruiker.



Deployment Guide

- A L S E T -

Drupal:

URL: cmsdev.localhost

Poort: 8080

Login: cmsdev-user

Wachtwoord: cmsdev-pass

Mobile Web App:

Poort: 8090