



# Cloud

Betere slaapgewoontes  
voor middelbareschool studenten

Door Basiel Smitz

New Media Development - Multimedia Productie

Docent: Frederick Roegiers

2018-2019



# Table of contents

---

## Contents

---

<b>Table of contents</b>	<b>2</b>
<b>Discover</b>	<b>3</b>
Briefing	3
<b>Define</b>	<b>4</b>
Analyse	4
<b>Design</b>	<b>5</b>
Screenesigns en wireflow	5
Styleguide	14
<b>Develop</b>	<b>15</b>
Code snippets.	15
Screenshots	31
<b>Deliver</b>	<b>35</b>
Handleiding	35
<b>Deploy</b>	<b>37</b>



# Discover

---

## Briefing

In het kader van een project rond gezonde slaapgewoontes bij jongeren, wordt vanuit de onderzoeksgroep 'Gezondheidsbevordering' aan de UGent momenteel een interventie ontwikkeld die gezonde slaap wil promoten. Momenteel nemen 7 Vlaamse scholen deel aan dit project (3 als interventieschool, 4 als controleschool). De inhoud van de interventie ter promotie van gezonde slaapgewoontes, wordt samen met de doelgroep bepaald, er wordt in dit project namelijk gedoeld om op maximale wijze participatief te werk te gaan.

Van de jongeren werd vernomen dat ze heel graag een app zouden ontwikkelen, die hen kennis over slaap bijbrengt maar ook stimuleert om hun smartphone/tablet minder te gebruiken voor het slapengaan.

Dit wordt gerealiseerd doormiddel van een Angular Progressvie Web App die gevoed wordt door een Drupal 8 backend.



# Define

---

## Analyse

### Requirements:

- Gebruiker
  - Login & registratie
  - Gebruikers profiel met foto, naam, school
  - Slaapdoel opstellen
  - Type slaper bepalen via quiz
- Vrienden
  - Lijst van vrienden op een interactive map weergeven
  - Vrienden profiel met openbare gegevens
- Slaap tracker
  - Dagelijkse mogelijk tot ingeven
    - Hoe laat gaan slapen/opgestaan
    - rating van 0 - 5 omtrent diepte van de slaap
    - rating van 0 - 5 omtrent hoe uitgerust je bent
  - Dagelijkse melding om die meld wanneer je moet gaan slapen
  - statistieken
- Tips en Tools
  - verschillende categorieën
  - filmpjes om tot rust te komen
  - tips om beter te slapen
- Fotowedstrijd
  - sorteermogelijkheid
  - overzicht alle foto's
  - upload van foto met hastag
  - mogelijkheid om foto terug te verwijderen
  - foto kan geliked worden
  - aantal likes moet zichtbaar zijn



# Design

Screenesigns en wireflow

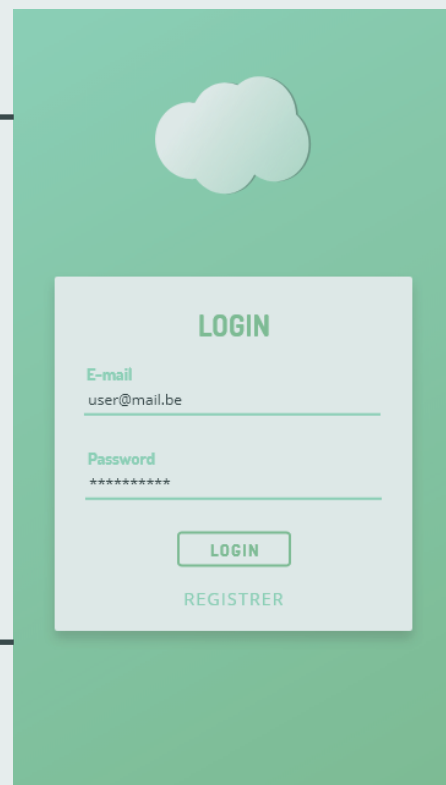


# Home/Login

0.0 loading page



0.1 Login page



1.0 Welcome screen



0.2 Register

1.1 User

1.2 Friends


1.3 Sleep tracker

1.4 Tips & Tools

1.5 Contest

# Register

0.2.0 Register name



Welcome bij Cloud!

**REGISTREER**


Uw voornaam  
Jane

Uw naam  
Doe

→

LOGIN

0.2.1 Register school



Welcome bij Cloud!


**REGISTREER**

Waar gaat u naar school?  
Arteveldehogeschool Gent

← →

LOGIN

0.2.3 Register password



Welcome bij Cloud!

**REGISTREER**


Wachtwoord  
\*\*\*\*\*

Wachtwoord herhalen  
\*\*\*\*\*

← **VERZEND**

LOGIN

0.2.2 Register Adress



Welcome bij Cloud!

**REGISTREER**

Straat	Nr
Veldstraat	33

Gemeente  
9000 Gent

← →

LOGIN

1.0 Welcome screen

## 1.1 User page



**Jane Doe**

**School**

Arteveldehogeschool Gent

**Adres**

Veldstraat 33  
9000 Gent

**Type slaper** Doe de Quiz!

Schone slaper

**Slaap doelstelling**

9 uur / dag & 63 uur / week

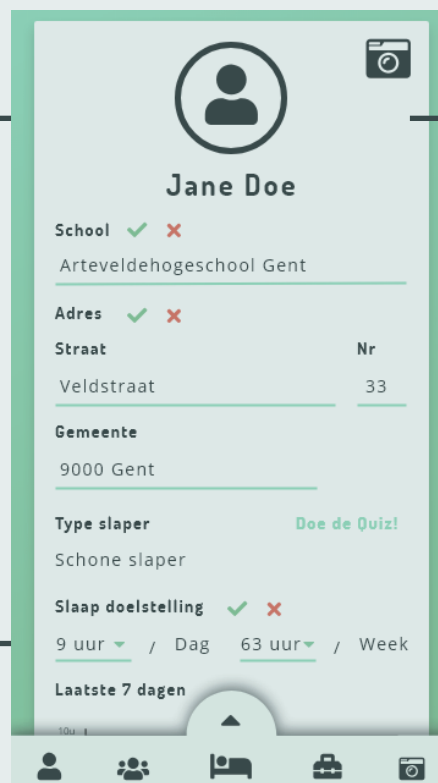
**Laatste 7 dagen**



10u  
9u  
8u  
7u  
6u  
5u  
4u

X.1 Quiz

## 1.1 User page (edit)



**Jane Doe**

**School**

Arteveldehogeschool Gent

**Adres**

**Straat** Nr

Veldstraat 33

**Gemeente**

9000 Gent

**Type slaper** Doe de Quiz!

Schone slaper

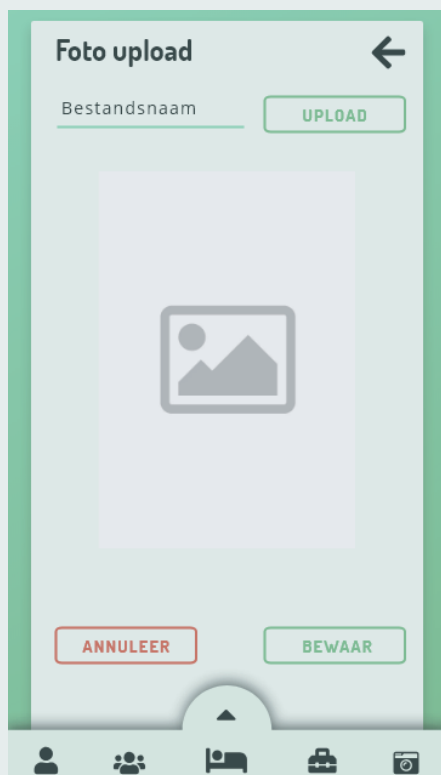
**Slaap doelstelling**

9 uur / Dag 63 uur / Week

**Laatste 7 dagen**


10u  
9u  
8u  
7u  
6u  
5u  
4u

## X.0 Picture upload



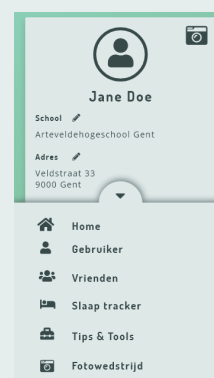
**Foto upload**

Bestandsnaam **UPLOAD**



**ANNULEER** **BEWAAR**

Expanded menu



**Jane Doe**

**School**

Arteveldehogeschool Gent

**Adres**

Veldstraat 33  
9000 Gent

Home

Gebruiker

Vrienden

Slaap tracker

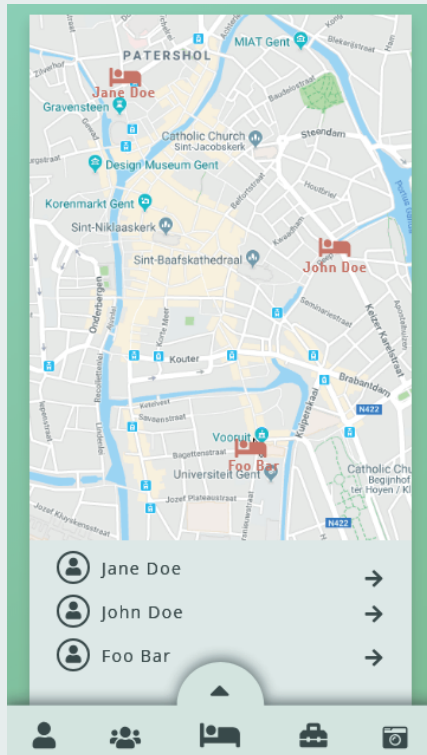
Tips & Tools

Fotowedstrijd



# Vrienden

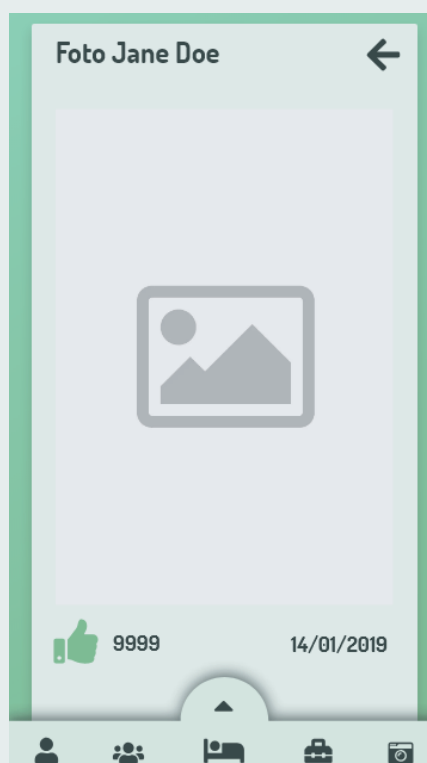
## 1.2.0 Friends



## 1.2.1 Friend detail



## 1.2.2 Friend picture



# Slaap tracker

## 1.3 Sleep tracker (input)

### Slaap tracker

Vandaag

M	D	W	D	V	Z	Z
30	31	1	2	3	4	5
6	7	8	9	10	11	12
13	14	15	16	17	18	19
20	21	22	23	24	25	26
27	28	29	30	1	2	3

Wanneer bent u gaan slapen? 23 : 50

Wanneer bent u opgestaan? 7 : 20

Hebt u diep geslapen? Nee

Voelt u zich uitgerust? Ja

Beoordeel uw nacht:

0 1 2 3 4 5

SAVE

## 1.3 Sleep tracker (result)

### Slaap tracker

Januari

M	D	W	D	V	Z	Z
30	31	1	2	3	4	5
6	7	8	9	10	11	12
13	14	15	16	17	18	19
20	21	22	23	24	25	26
27	28	29	30	1	2	3

U hebt in de nacht van 16 januari 2019  
7 uur en 30 minuten geslapen.

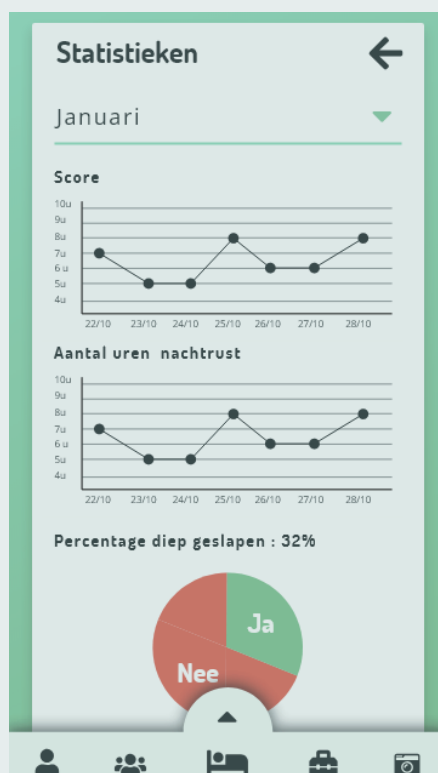
Dat is 1 uur en 3 minuten minder dan  
uw dagelijkse doelstelling.

Verder hebt u niet diep geslapen maar  
voelde u zich wel uitgerust in de  
ochtend.

U gaf deze nacht een score van 3!

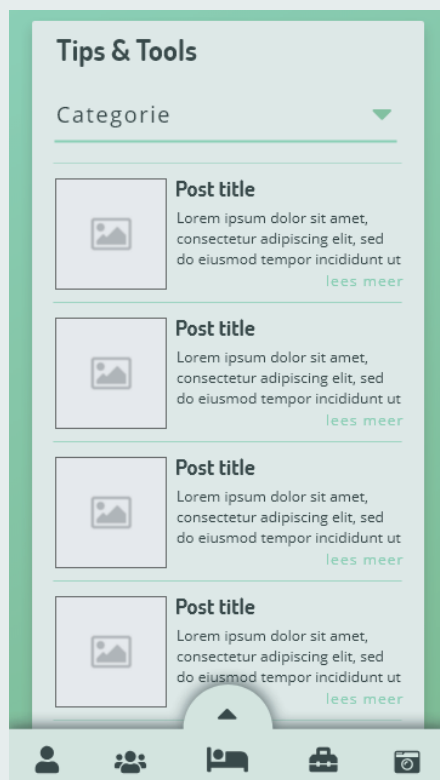
[Statistieken bekijken](#)

## 1.3.0 Statistics



# Tips and Tools

## 1.4 tips and tools

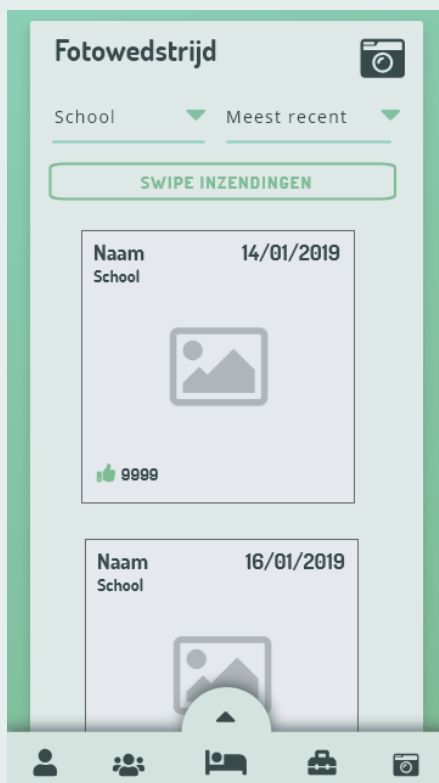


## 1.4.1 tips/article detail

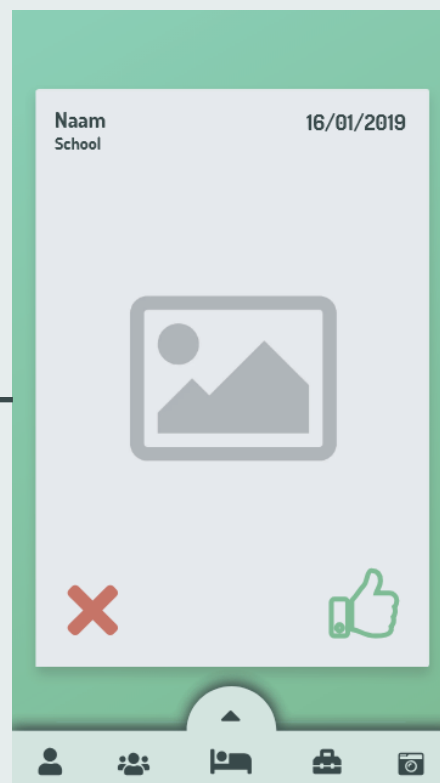


# Contest

1.5 Contest overview



1.5.0 Contest swipe



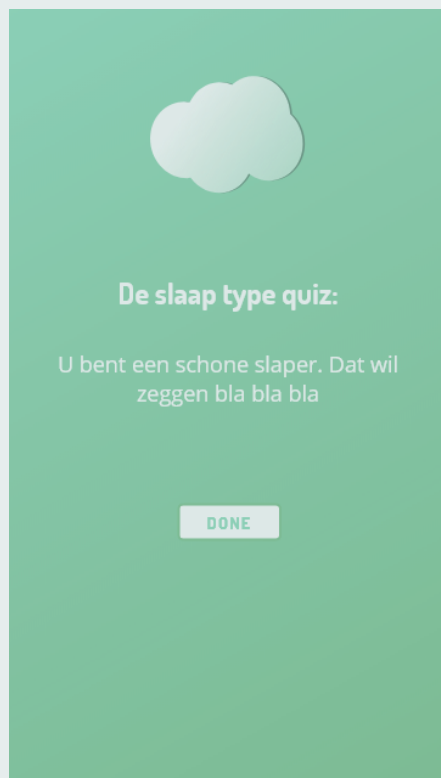
X.2.0 Quiz start



X.2.1 Quiz slider



X.2.3 Quiz end



X.2.2 Quiz Yes/No



## Cloud : styleguide

### Typography

**H1: Dosis - bold 24pt**

**H2: Dosis - bold - 18pt**

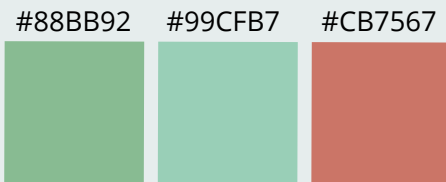
**H3: Dosis - bold 16 pt**

Reguliere text: Open-Sans 12pt  
Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris.

**Link: Open Sans bold 12 pt**

**Link visited: Open Sans bold 12 pt**

### Colors



### Inputs

Regular Open Sans 12 pt

Valid input

Wrong input

Error message input

Select closed



Select open



Option  
Option  
Option  
Option  
Option

☒ Option ☐ Option

☒ True

☐ False

✓ Positive message

✗ Negative message



# Develop

Code snippets



## Contest: Image upload

```
9  export class AddComponent implements OnInit {
10
11      public selectedFile = null;
12      public previewSrc = null;
13      private newImageId = null;
14      public canPost = false;
15
16      constructor(
17          | private imageService: ImageService,
18          ) { }
19
20      ngOnInit() {
21      }
22      upload() {
23          const _this = this;
24          this.imageService.upload(this.selectedFile).then(response => {
25              | console.log(response);
26              | this.newImageId = response.data.uuid[0].value;
27              | _this.canPost = true;
28          }).catch(error => console.log(error));
29
30      }
```





```
31 post () {
32   this.imageService.post({
33     data: {
34       type: 'node--picture_post',
35       attributes: {
36         title: 'new',
37       },
38       relationships: {
39         field_image: {
40           data: {
41             id: this.newImageId,
42             type: 'file--file',
43           }
44         },
45         field_school: {
46           data: {
47             type: 'taxonomy_term--schools',
48             id: 'c09a9739-b033-46ae-88f8-744be6efcda7'
49           }
50         },
51         field_contest: {
52           data: {
53             type: 'taxonomy_term--contest',
54             id: '2df0a02c-ed66-40c1-ba99-f79205e3afcf'
55           }
56         }
57       }
58     }
59   }).then(response => {
60     console.log(response);
61   }).catch(error => {
62
63   })
64 }
65
66 onNewImage(event) {
67   this.selectedFile = <File>event.target.files[0];
68   const reader = new FileReader();
69   reader.onload = (e) => {
70     // put next line in comment or first time building will fail. uncomment and second build succeeds.
71     this.previewSrc = e.target.result;
72   };
73   reader.readAsDataURL(this.selectedFile);
74 }
75
76 }
```



## Contest: Swipe component

```
export class SwipeComponent implements OnInit {

  private allEntries = JSON.parse(localStorage.getItem('pictureEntries'));
  private filteredEntries = [];
  private disliked = [];
  public currentEntry = null;
  private userId = JSON.parse(localStorage.getItem('currentUser')).data.id;

  constructor(
    private contestService: ContestService,
    private router: Router,
  ) { }

  ngOnInit() {
    this.filterEntries();
  }

  filterEntries() {
    console.log(this.allEntries);
    const filter1 = this.allEntries.filter(entry => entry.user.uuid !== this.userId);
    console.log(filter1);
    const filter2 = filter1.filter(entry => entry.likes === this.hasLiked(entry.likes));
    this.filteredEntries = filter2;
    console.log(this.filteredEntries);
    this.currentEntry = this.filteredEntries[0];
    console.log(this.currentEntry);
  }

  hasLiked(likes) {
    const filtered = likes.filter(like => like.id === this.userId);
    if (filtered.length < 1) {
      return likes;
    }
    return null;
  }

  next() {
    this.currentEntry = this.filteredEntries[this.filteredEntries.indexOf(this.currentEntry) + 1];
    if (!this.currentEntry) {
      this.router.navigate(['/contest']);
    }
  }
}
```

# Tips and Tools

```
9  export class TipsandtoolsComponent implements OnInit {
10
11      constructor(private tipsAndToolsService: TipsAndToolsService) { }
12      private allTipsAndTools = [];
13      public result = [];
14      private filtered = [];
15      public options = [
16          {name: 'All', value: 'all'},
17          {name: 'Videos', value: 'node--video'},
18          {name: 'Tips', value: 'node--tip'},
19          {name: 'Articles', value: 'node--article'},
20          {name: 'Memes', value: 'node--meme'}
21      ];
22      public selected = this.options[0].value;
23      private videosDone = false;
24      private tipsDone = false;
25      private articlesDone = false;
26      private memesDone = false;
27      private tempAllTipsAndTools = [];
28      ngOnInit() {
29          const localTipsAndTools = JSON.parse(localStorage.getItem('tipsAndTools'));
30          if (localTipsAndTools) {
31              this.allTipsAndTools = localTipsAndTools;
32              this.filterResult();
33          }
34          this.getVideos();
35          this.getTips();
36          this.getArticles();
37          this.getMemes();
38      }
39      getVideos() {
40          const _this = this;
41          this.tipsAndToolsService.getVideos().then(response => {
42              const videos = response.data.data;
43              videos.forEach((video, index) => {
44                  _this.tempAllTipsAndTools.push(video);
45                  if (index === videos.length - 1) {
46                      _this.videosDone = true;
47                      _this.checkDone();
48                  }
49              });
50          });
51      }
```



```
64  getArticles() {
65      const _this = this;
66      this.tipsAndToolsService.getArticles().then(response => {
67          const articles = response.data.data;
68          articles.forEach((article, index) => {
69              _this.tempAllTipsAndTools.push(article);
70              if (index === articles.length - 1) {
71                  _this.articlesDone = true;
72                  _this.checkDone();      }
73          });
74      });
75  }
76  getMemes() {
77      const _this = this;
78      this.tipsAndToolsService.getMemes().then(response => {
79          const memes = response.data.data;
80          memes.forEach((meme, index) => {
81              _this.tempAllTipsAndTools.push(meme);
82              if (index === memes.length - 1) {
83                  _this.memesDone = true;
84                  _this.checkDone();      }
85          });
86      });
87  });
88  }
89  filterResult() {
90      if (this.selected !== 'all') {
91          this.result = this.allTipsAndTools.filter(element => element.type === this.selected);
92          this.result.sort((a, b) => b.attributes.created - a.attributes.created);
93      } else {
94          this.result = this.allTipsAndTools;
95          this.result.sort((a, b) => b.attributes.created - a.attributes.created);
96      }
97  }
98  checkDone() {
99      if (this.videosDone && this.tipsDone && this.memesDone && this.articlesDone) {
100          this.allTipsAndTools = this.tempAllTipsAndTools;
101          this.filterResult();
102          localStorage.setItem('tipsAndTools', JSON.stringify(this.allTipsAndTools));
103      }
104  }
105  getThumbnailImage(video) {
106      const fullVideoUrl = video.attributes.field_video_url.uri;
107      if (fullVideoUrl.includes('youtube')) {
108          const id = fullVideoUrl.split('?v=')[1];
109          return `https://img.youtube.com/vi/${id}/0.jpg`;
110      }
111  }
112 }
```

# Sleep tracker

```
12 export class SleeptrackerComponent implements OnInit {
13
14     constructor(
15         private sleepLogService: SleepLogService,
16         private dateService: DateService,
17     ) { }
18     private user = JSON.parse(localStorage.getItem('currentUser'));
19     private newDate = new Date();
20     private today = {
21         data: null,
22         date: this.newDate.getDate(),
23         day: this.newDate.getDay(),
24         month: this.newDate.getMonth() + 1,
25         year: this.newDate.getFullYear(),
26         fullDate: this.newDate
27     };
28     public month = this.newDate.getMonth();
29     public year = this.newDate.getFullYear();
30     private firstOfMonth = new Date(this.newDate.getFullYear(), this.month + 1, 1);
31     private firstDayOfMonth = this.firstOfMonth.getDay();
32     public date = this.newDate.getDate();
33     public logs = [];
34     public calendar = JSON.parse(localStorage.getItem('calendar'));
35     private calendarArray = [];
36     public calendarReady = false;
37     public currentMonth = null;
38     public selectedMonth = this.today.month;
39     public availableMonths = JSON.parse(localStorage.getItem('availableMonths'));
40     public userGoal = JSON.parse(localStorage.getItem('currentUser')).data.attributes.field_goal_hours;
41     public selectedLog = null;
42     public dateIsToday = false;
43     ngOnInit() {
44         if (JSON.parse(localStorage.getItem('logs'))) {
45             this.logs = JSON.parse(localStorage.getItem('logs'));
46             this.createCalendar();
47         }
48         this.getLogs();
49         if (this.availableMonths) {
50             this.setMonthList();
51         } else {
52             this.availableMonths = [];
53         }
54         this.isToday();
55     }
56 }
```



```
56 isToday() {
57
58     const strippedToday = this.today.fullDate.setHours(0, 0, 0, 0);
59     this.logs.forEach(log => {
60         const date = new Date(log.data.field_datum);
61         date.setHours(0, 0, 0, 0);
62     });
63     const tempToday = this.logs.filter(log => log === this.checkDate(log, strippedToday));
64     if (tempToday.length < 1) {
65         this.dateIsToday = true;
66     } else {
67         this.dateIsToday = false;
68         this.selectLog(tempToday);
69     }
70 }
71 checkDate(log, today) {
72     const stripDate = new Date(log.data.field_datum);
73     stripDate.setHours(0, 0, 0, 0);
74     const stripNumber = stripDate.getTime();
75     if (stripNumber === today) {
76         return log;
77     }
78     return null;
79 }
80 setMonthList() {
81     const months = JSON.parse(localStorage.getItem('availableMonths'));
82     const filtered = months.filter(m => m.value === this.today.month);
83     if (filtered.length < 1) {
84         const month = {
85             value: this.today.month,
86             name: this.dateService.getMonthDetail(this.today.month - 1).name + ' ' + this.today.year,
87             year: this.today.year,
88         };
89         months.unshift(month);
90         localStorage.setItem('availableMonths', JSON.stringify(months));
91     }
92     this.availableMonths = months;
93 }
94 getDayName(day: number) {
95     return this.dateService.getDayName(day);
96 }
97 }
```



```
98  getLogs() {
99      const tempLogs = [];
100     const _this = this;
101     this.sleepLogService.getUserLogs(this.user.data.attributes.uid).then(response => {
102         const allLogs = response.data.data;
103         allLogs.forEach((log, index) => {
104             const logDate = new Date(log.attributes.created * 1000);
105             logDate.setHours(0, 0, 0, 0);
106             const newLog = {
107                 data: log.attributes,
108                 date: logDate.getDate(),
109                 day: logDate.getDay(),
110                 month: logDate.getMonth() + 1,
111                 year: logDate.getFullYear(),
112                 fullDate: logDate
113             };
114             tempLogs.push(newLog);
115             const month = {
116                 value: newLog.month,
117                 name: _this.dateService.getMonthDetail(newLog.month - 1).name + ' ' + newLog.year,
118                 year: newLog.year,
119             };
120             const filtered = _this.availableMonths.filter(m => m.value === month.value);
121             if (filtered.length === 0) {
122                 _this.availableMonths.push(month);
123                 localStorage.setItem('availableMonths', JSON.stringify(_this.availableMonths));
124             }
125             if (allLogs.length - 1 === index) {
126                 localStorage.setItem('logs', JSON.stringify(tempLogs));
127                 _this.logs = _this.logs.splice(0, _this.logs.length);
128                 _this.logs = tempLogs;
129                 this.createCalendar();
130             }
131         });
132     });
133 }
```



```
134 createCalendar() {
135   if (this.calendar) {
136     this.calendar = this.calendar.splice(0, this.calendar.length);
137   }
138   const currentDate = new Date(this.newDate.getFullYear(), this.newDate.getMonth() + 1, 0);
139   currentDate.setHours(0, 0, 0, 0);
140   const lastLog = new Date(this.logs[this.logs.length - 1].fullDate);
141   const lastDate = new Date(lastLog.getFullYear(), lastLog.getMonth(), 1);
142   lastDate.setHours(0, 0, 0, 0);
143   let lastIndex = 0;
144   while (currentDate.getTime() >= lastDate.getTime()) {
145     const lastLogDate = new Date(this.logs[lastIndex].data.created * 1000);
146     lastLogDate.setHours(0, 0, 0, 0);
147     if (lastLogDate.getTime() === currentDate.getTime()) {
148       this.calendarArray.unshift(this.logs[lastIndex]);
149       if (lastIndex < this.logs.length - 1) {
150         lastIndex++;
151       }
152     } else {
153       this.calendarArray.unshift({
154         data: null,
155         date: currentDate.getDate(),
156         day: currentDate.getDay(),
157         month: currentDate.getMonth() + 1,
158         year: currentDate.getFullYear(),
159         fullDate: new Date(currentDate.getTime()),
160       });
161     }
162   }
163   if (currentDate.getTime() === lastDate.getTime()) {
164     this.calendar = this.calendarArray;
165     this.calendarReady = true;
166     this.getMonth(this.today.month);
167     localStorage.setItem('calendar', JSON.stringify(this.calendar));
168     this.setMonthList();
169     this.isToday();
170     break;
171   }
172   currentDate.setDate(currentDate.getDate() - 1);
173 }
174 }
```





```
175 getMonth(monthId: number) {
176   const filteredMonth = this.calendar.filter(date => date.month === monthId);
177   let start = this.calendar.indexOf(filteredMonth[0]);
178   let end = this.calendar.indexOf(filteredMonth[filteredMonth.length - 1]);
179   --start;
180   ++end;
181   if (start > 0) {
182     while (this.calendar[start].day >= 1) {
183       filteredMonth.unshift(this.calendar[start]);
184       start--;
185       if (start === 0) {
186         filteredMonth.unshift(this.calendar[start]);
187         break;
188       }
189     }
190   } else {
191     let date = filteredMonth[0];
192     while (date.day > 1 || date.day === 0) {
193       const newDate = new Date(date.fullDate);
194       newDate.setDate(newDate.getDate() - 1);
195       const newEntry = {
196         data: null,
197         date: newDate.getDate(),
198         day: newDate.getDay(),
199         month: newDate.getMonth() + 1,
200         year: newDate.getFullYear(),
201         fullDate: new Date(newDate.getTime()),
202       };
203       date = newEntry;
204       filteredMonth.unshift(newEntry);
205       this.calendar.unshift(newEntry);
206     }
207   }
208   if (end < this.calendar.length) {
209     while (this.calendar[end].day <= 6 && this.calendar[end].day !== 1) {
210       filteredMonth.push(this.calendar[end]);
211       if (end === 6) {
212         end = 0;
213       } else {
214         end++;
215       }
216       if (end === this.calendar.length - 1) {
217         if (filteredMonth[filteredMonth.length - 1].day === 6) {
218           filteredMonth.push(this.calendar[end]);
219         }
220         break;
221       }
222     }
223   }
```



```
224 } else if (this.calendar.length === end) {
225   let date = filteredMonth[filteredMonth.length - 1];
226   while (date.day > 0 && date.day <= 6) {
227     const newDate = new Date(date.fullDate);
228     newDate.setDate(newDate.getDate() + 1);
229     const newEntry = {
230       data: null,
231       date: newDate.getDate(),
232       day: newDate.getDay(),
233       month: newDate.getMonth() + 1,
234       year: newDate.getFullYear(),
235       fullDate: new Date(newDate.getTime()),
236     };
237     date = newEntry;
238     filteredMonth.push(newEntry);
239     this.calendar.push(newEntry);
240     if (date.day === 0) {
241       break;
242     }
243   }
244 }
245 this.currentMonth = filteredMonth;
246 }
```



```
251 selectLog(date) {
252   if (date) {
253     if (date.data) {
254       this.selectedLog = date.data;
255       this.dateIsToday = false;
256       console.log('oke');
257     }
258     const logDate = new Date(date.fullDate);
259     const logWT = new Date(date.fullDate);
260     const todayDate = new Date(this.today.fullDate.getTime());
261     todayDate.setHours(0, 0, 0, 0);
262     if (logDate.getTime() === todayDate.getTime() && !date.data) {
263       this.dateIsToday = true;
264       this.selectedLog = null;
265     } else if (logDate.getTime() === todayDate.getTime()) {
266     }
267   }
268 }
269 }
270 setClass(date) {
271   if (this.selectedMonth === date.month) {
272     if (date.data) {
273       const wakeUp = new Date(date.data.field_wake_up_time);
274       const goSleep = new Date(date.data.field_sleeping_time);
275       const hoursSlept = Math.abs(wakeUp.getTime() - goSleep.getTime()) / 36e5;
276       if (hoursSlept >= this.userGoal) {
277         return 'goodLog';
278       }
279       return 'badLog';
280     }
281     const logDate = new Date(date.fullDate.getTime());
282     const todayDate = new Date(this.today.fullDate.getTime());
283     todayDate.setHours(0, 0, 0, 0);
284     if (logDate.getTime() === todayDate.getTime() && !date.data) {
285       return 'todayLog';
286     }
287     return 'emptyLog';
288   }
289   return 'unavailabeLog';
290 }
291 }
```



# Http service

```
8 export class HttpService {
9   public instance = axios.create();
10  constructor(private authenticationService: AuthenticationService) {
11    const _this = this;
12    this.instance.defaults.headers.common['Authorization'] = `Bearer ${localStorage.getItem('access_token')}`;
13    this.instance.interceptors.request.use(function (config) {
14      const originalRequest = config;
15      const date = new Date();
16      const time = date.getTime();
17      const expiration = +localStorage.getItem('token_expiration');
18      console.log((expiration - time) / 1000);
19      console.log(time);
20      console.log(expiration);
21      const requestingToken = JSON.parse(localStorage.getItem('requestingToken'));
22      if (time >= expiration && !requestingToken) {
23        localStorage.setItem('requestingToken', 'true');
24        return authenticationService.refresh().then((response) => {
25          authenticationService.setLocalStorage(response.data);
26          originalRequest['Authorization'] = `Bearer ${localStorage.getItem('access_token')}`;
27          localStorage.setItem('requestingToken', 'false');
28          return Promise.resolve(originalRequest);
29        });
30      }
31      return config;
32    }, function (error) {
33      // Do something with request error
34      return Promise.reject(error);
35    });
36  }
37 }
```

## Sleep log srvice

```
8 export class SleepLogService {
9
10   constructor(private httpService: HttpService) { }
11   private baseUrl = 'http://slaapapp.local';
12   private logs = this.baseUrl + '/jsonapi/node/sleep_log';
13   getLatestUserLogs(uid) {
14     const _this = this;
15     const date = new Date();
16     date.setHours(0, 0, 0, 0);
17     date.setDate(date.getDate() - 7);
18     const time = date.getTime() / 1000;
19     console.log(time);
20
21     return this.httpService.instance.get(`${this.logs}?filter[uid]=${uid}` +
22       `&filter[created-filter][condition][path]=created` +
23       `&filter[created-filter][condition][operator]=%3E` +
24       `&filter[created-filter][condition][value]=${time}` +
25       `&page[limit]=7&sort=-created`,
26       {
27         headers: {
28           'Content-Type': 'application/vnd.api+json',
29           'Authorization': `Bearer ${localStorage.getItem('access_token')}`,
30         },
31       });
32   }
33   getUserLogs(uid) {
34     return this.httpService.instance.get(`${this.logs}?filter[uid]=${uid}` +
35       `&sort=-created`,
36       {
37         headers: {
38           'Content-Type': 'application/vnd.api+json',
39           'Authorization': `Bearer ${localStorage.getItem('access_token')}`,
40         },
41       });
42   }
43   saveLog(data) {
44     console.log('saving log');
45     return this.httpService.instance.post(`${this.logs}`, {
46       data: {
47         type: 'node--sleep_log',
48         attributes: data,
49       },
50     }, {
51       headers: {
52         'Content-Type': 'application/vnd.api+json',
53         'Authorization': `Bearer ${localStorage.getItem('access_token')}`,
54       },
55     });
56   }
57 }
```



# Contest Service

```
8 export class ContestService {
9
10   constructor(private httpService: HttpService) { }
11   private baseUrl = 'http://slaapapp.local';
12   private contests = this.baseUrl + '/jsonapi/node/picture_post';
13   getLatestByUser(uid) {
14     return this.httpService.instance.get(`${this.contests}?filter[uid]=${uid}&page[limit]=1&sort=-created`,
15     {
16       headers: {
17         'Content-Type': 'application/vnd.api+json',
18         'Authorization': `Bearer ${localStorage.getItem('access_token')}`,
19       },
20     });
21   }
22   getAllFromScool(id) {
23     return this.httpService.instance.get(`${this.contests}` +
24     `?filter[field_school.uuid]=${id}&filter[status][value]=1` +
25     `&include=uid,field_image,field_contest`,
26     {
27       headers: {
28         'Content-Type': 'application/vnd.api+json',
29         'Authorization': `Bearer ${localStorage.getItem('access_token')}`,
30       },
31     });
32   }
33   castLike(data, id) {
34     console.log(id);
35     console.log(data);
36     const currentLikes = data.relationships.field_likes.data;
37     currentLikes.push({
38       type: 'user--user',
39       id: id,
40     });
41     console.log(currentLikes);
42     return this.httpService.instance.patch(`${this.contests}/${data.id}`, {
43       data: {
44         type: 'node--picture_post',
45         id: data.id,
46         relationships: {
47           field_likes: {
48             data: currentLikes
49           }
50         }
51       }
52     }, {
53       headers: {
54         'Content-Type': 'application/vnd.api+json',
55         'Authorization': `Bearer ${localStorage.getItem('access_token')}`,
56       }
57     });
```



# Develop

## Screenshots

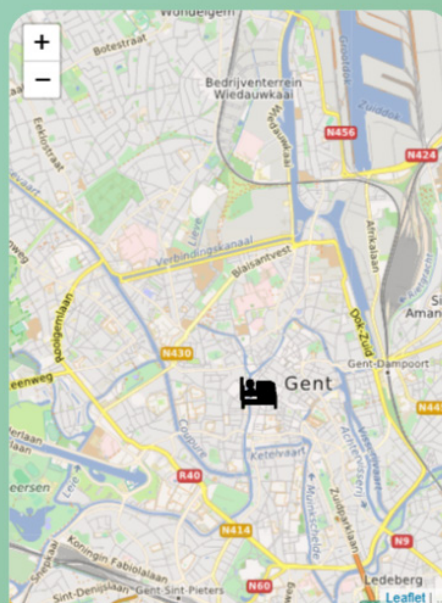


## Welcome bij Cloud Basiel Smitz!

Het is donderdag en u zit deze week 8 uur voor met slapen op uw doel

Uw laatste #iwokeuplikethis foto heeft ondertussen 0 likes.

HomeUserFriendsSleep TrackerTips & ToolsContest



Fons De Spons

Bob Bouwer

Holle Bolle

berta flap

HomeUserFriendsSleep TrackerTips & ToolsContest



### Basiel Smitz

School

Het Kwadrant

Adres

Molenaarsstraat 21

Gent 9000

Type slaper

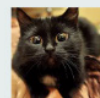
Schone slaper

Slaap doelstelling

7 uur / nacht



HomeUserFriendsSleep TrackerTips & ToolsContest



### Fons De Spons

School

Het Kwadrant

Adres

Veldstraat 3

9000 Gent

Type slaper

Schone slaper

Slaap doelstelling

55 uur / nacht

HomeUserFriendsSleep TrackerTips & ToolsContest





## Slaap tracker

januari 2019

M	D	W	D	V	Z	Z
31	1	2	3	4	5	6
7	8	9	10	11	12	13
14	15	16	17	18	19	20
21	22	23	24	25	26	27
28	29	30	31	1	2	3

Wanneer bent u gaan slapen? --:--

Wanneer bent u opgestaan? --:--

in welke mate hebt u diep geslapen?

in welke mate voelt u zich uitgerust?

Beoordeel uw nacht?

HomeUserFriendsSleep TrackerTips & ToolsContest

M	D	W	D	V	Z	Z
31	1	2	3	4	5	6
7	8	9	10	11	12	13
14	15	16	17	18	19	20
21	22	23	24	25	26	27
28	29	30	31	1	2	3

Wanneer bent u gaan slapen? --:--

Wanneer bent u opgestaan? --:--

in welke mate hebt u diep geslapen?

in welke mate voelt u zich uitgerust?

Beoordeel uw nacht?

save

Statistics

HomeUserFriendsSleep TrackerTips & ToolsContest

## Slaap tracker

januari 2019

M	D	W	D	V	Z	Z
31	1	2	3	4	5	6
7	8	9	10	11	12	13
14	15	16	17	18	19	20
21	22	23	24	25	26	27
28	29	30	31	1	2	3

U hebt in de nacht van 12 januari 2019 11 uur geslapen.

Goed! U hebt 4 uur meer dan uw dagelijkse slaapdoel

Verder hebt u diep geslapen en voelde u zich uitgerust in de ochtend.

U gaf deze nacht een score van 2/5!

Statistics

HomeUserFriendsSleep TrackerTips & ToolsContest



## Tips and Tools

All ▼

nieuwe tip

long article

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Duis vitae ipsum eu massa pharetra dictum. Duis tempus congue fringilla. Sed dapibus, lacus in feugiat porttitor, leo velit dapibus velit, sit amet aliquam turpis leo ut arcu. Vestibulum id sapien nunc. Vivamus rutrum felis massa, a ultricies mi pulvinar id. Quisque eu tortor nisl. Aenean sed congue justo. Suspendisse lacinia tincidunt dolor, nec tristique nisi ultricies sit amet. Integer et laoreet ante. Donec ut lacus id nulla cursus euismod quis sed velit. Duis facilisis dui metus, id congue ex lacinia quis. Donec placerat ante porttitor metus fringilla sollicitudin. Integer dignissim cursus neque, at rhoncus metus vulputate in. Interdum et malesuada fames ac ante ipsum primis in faucibus.

new video



HomeUserFriendsSleep TrackerTips & ToolsContest□

Swipe Inzendingen

Doe mee



Basiel Smitz

0



Basiel Smitz

1



HomeUserFriendsSleep TrackerTips & ToolsContest□



long article

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Duis vitae ipsum eu massa pharetra dictum. Duis tempus congue fringilla. Sed dapibus, lacus in feugiat porttitor, leo velit dapibus velit, sit amet aliquam turpis leo ut arcu. Vestibulum id sapien nunc. Vivamus rutrum felis massa, a ultricies mi pulvinar id. Quisque eu tortor

HomeUserFriendsSleep TrackerTips & ToolsContest□



no

yes

HomeUserFriendsSleep TrackerTips & ToolsContest□



# Deliver

Handleiding



## **Registratie**

De gebruiker deelt naam, locatie school en wachtwoord mee. Na de registratie begint de Quiz om de gebruiker zijn slaaprofiel vast te stellen.

Alle gegevens binnen de applicatie zullen gelinkt zijn aan de school die hier werd opgegeven.

## **Login**

Na het inloggen komt de gebruiker op een welkomstscherf. Daar wordt meegedeeld hoe de afgelopen nacht verliep en hoeveel likes zijn deelname voor de wedstrijd ondertussen heeft.

Onderaan staat een menu waarmee de rest van de toepassing te bereiken is

## **User**

Op de gebruikerspagina vindt de gebruiker zijn gegevens terug. Hier kan hij bepaalde gegevens wijzigen alsook de quiz opnieuw nemen voor het slaap profiel opnieuw in te stellen.

## **Vrienden**

Op de vriendenpagina krijgt de gebruiker een lijst met bijbehorende map waarop die hun locaties te zien zijn. Hier kan men doorklikken naar een lijst van medestudenten om nieuwe vrienden toe te voegen.

Elke vriend heeft zijn eigen detailpagina.

## **Slaap tracker**

Via de slaap tracker heeft de gebruiker dagelijks details over zijn slaap in. de tijd wanneer deze is gaan slapen alsook de tijd van het opstaan.

Hier wordt ook door middel van sliders scores gegeven omtrent de diepte van de slaap, in welke mate de gebruiker is uitgerust en een algemene score van de nacht.

## **Tips & Tools**

De tips & tools pagina is een verzameling van artikelen, tips filmpjes en memes

## **Wedstrijd**

Op de wedstrijdpagina worden alle wedstrijd inzendingen weergegeven met de gebruiker en aantal likes. Van hieruit kan men naar het swipe onderdeel van de wedstrijd waar gebruikers hun stem kunnen uitbrengen. Als ook een optie om een nieuwe foto aan de wedstrijd toe te voegen.



# Deploy



1. Maak via het administratie paneel van uw hosting een nieuwe database aan
2. Update de gegevens in "sites/default/settings naar die van de online database
3. Upload alle inhoud van de public folder via ftp naar de root van de web server.
4. Exporteer de lokale database via drupal console of drush
5. Importeer deze export in de online database.
6. Pas alle API requests in het Angular project aan aan de nieuwe url
7. Run "ng build" en host de volledig gegenereerde dist folder op een webserver.