

Bot Agency Dashboard

Niels Cappelle
Ismail Kutlu

Web Of Things
Academiejaar 2017-18

Multimediatproductie - New Media Development
Bachelor in de grafische en digitale media
Arteveldehogeschool

Inhoudsopgave

Discover	4
Briefing	4
Specifiek	4
Define	5
Analyse	5
Planning	7
Inspiration	8
Moodboard	8
Design	9
Wireframes Website	9
Wireflow	10
Style Guide	10
Visual Designs	10
Develop	11
Code snippets	11
Schermafbeeldingen	14
Deliver	18
Handleiding	18
Deploy	19
Deployment Guide	19

1. Discover

1.1. Briefing

Hoe kan een softwaresysteem realtime communicatie realiseren tussen de gebruikers, software en hardware en dit op een gebruiksvriendelijke en attractieve manier?

De bedoeling van de eindopdracht is om een softwaresysteem te ontwikkelen in een groep van 2 studenten dat een antwoord biedt op bovenstaande probleemstelling. Hiervoor moeten we beroep doen op de technologieën die tijdens de colleges aan bod kwamen.

Het softwaresysteem bestaat uit 4 delen:

- Frontoffice
- Backoffice
- Cloud services
- Hardware

Daarnaast wordt er een productiedossier, presentatie en een timesheet per student opgeleverd.

1.2. Specifiek

We gaan een bot ontwikkelen waarmee we kunnen communiceren via Slack. Deze bot zorgt ervoor dat we informatie kunnen ophalen van API's en stuurt deze informatie realtime door naar een dashboard waar we alle informatie gaan weergeven.

2. Define

2.1. Analyse

2.1.1. Algemeen

Beide eindproducten zijn esthetisch verantwoord en gebruiksvriendelijk.

2.1.2. App

Niet-functionele specificaties

De content moet echt zijn. Daarom doen we beroep op content van API's van betrouwbare bronnen.

Voor de frontoffice gaan we een dashboard maken die bestaat uit een zelfgemaakt structuur aan de hand van Vue.js en CSS Grid. Deze zullen we opmaken aan de hand van SCSS.

Als backoffice kiezen we voor de chatapplicatie Slack. Slack is een bekend medium waarvan heel wat webbureaus gebruik maken. We maken een bot waarmee het team kan communiceren. Door bepaalde zaken te zeggen zullen er zaken veranderen op het dashboard die behoort tot de frontoffice.

We willen niet enkel onze frontoffice beïnvloeden door te communiceren met Slack. Daarom doen we beroep op de Raspberry Pi. We zullen alle emoticons die worden gebruikt tijdens het chatten, weergeven op een LED-scherm die in verbinding staat met de Raspberry Pi.

Functionele specificaties

- Slack API + DialogFlow
 - Chatten
 - Dialogs
 - Action buttons
 - Slash commands
- API's
 - Spotify API
 - Nummer, album en afspeellijst zoeken
 - Profiel weergeven
 - Afspeellijst aanmaken
 - Favoriete afspeellijsten weergeven
 - Aangemaakte afspeellijsten weergeven
 - Nummers toevoegen aan zelfgemaakte afspeellijsten
 - Favoriete albums weergeven
 - Tracklist opvragen van een afspeellijst
 - Tracklist opvragen van een album

- Spotify player bedienen
 - Nummer afspelen
 - Nummer pauzeren
 - Volgend nummer afspelen
 - Vorig nummer afspelen
- Yahoo Weather API
 - Weer opvragen per stad
- Google Maps Directions API
- Google Maps Geolocation API
- Emoticons weergeven op LED-screen

Technische specificaties

- Npm
- ES6
- Sass
- Vue.js
- Node.js
- Express.js
- OAuth
- API's
 - Spotify API
 - Yahoo Weather API
 - Slack API
 - Google Maps API
- Firebase
- Heroku Cloud
- DialogFlow (API.AI)
- Webhooks

2.2. Planning

2.2.1. Milestone 1 - Deadline: 28-11-2017

Productiedossier

- Discover
 - Briefing
- Define
 - Analyse
 - Planning
 - Inspiration
 - Ideaboard

2.2.2. Milestone 2 - Deadline: 05-12-2017

Productiedossier

- Design
 - Sitemap Website
 - Wireframes Website
 - Wireflow
 - Style Guide (Adobe CC)
 - Visual Designs (Adobe CC)

2.2.3. Milestone 3 - Deadline: 12-12-2017

Productiedossier + Code

- Develop
 - Style Guide (Webtechnology - Sass)
 - Data

2.3. Inspiration

2.3.1. Moodboard

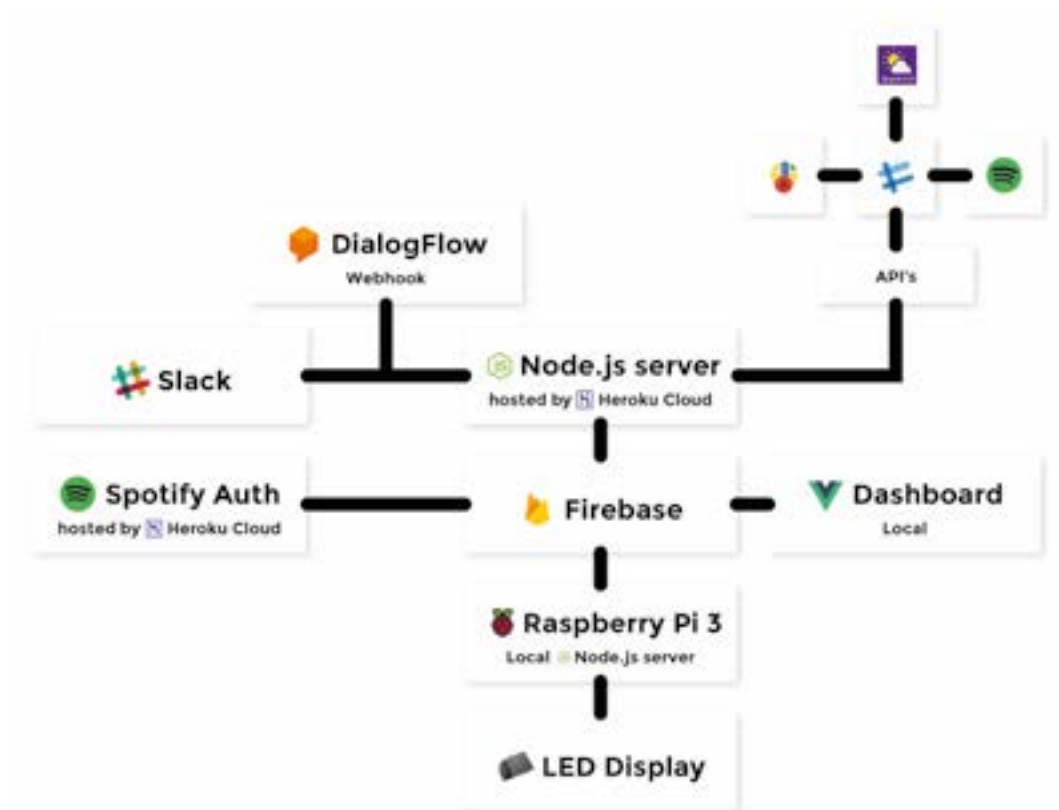


3. Design

3.1. Wireframes Website



3.2. Wireflow



3.3. Style Guide

De style guide zit in de bijlage als Adobe XD.

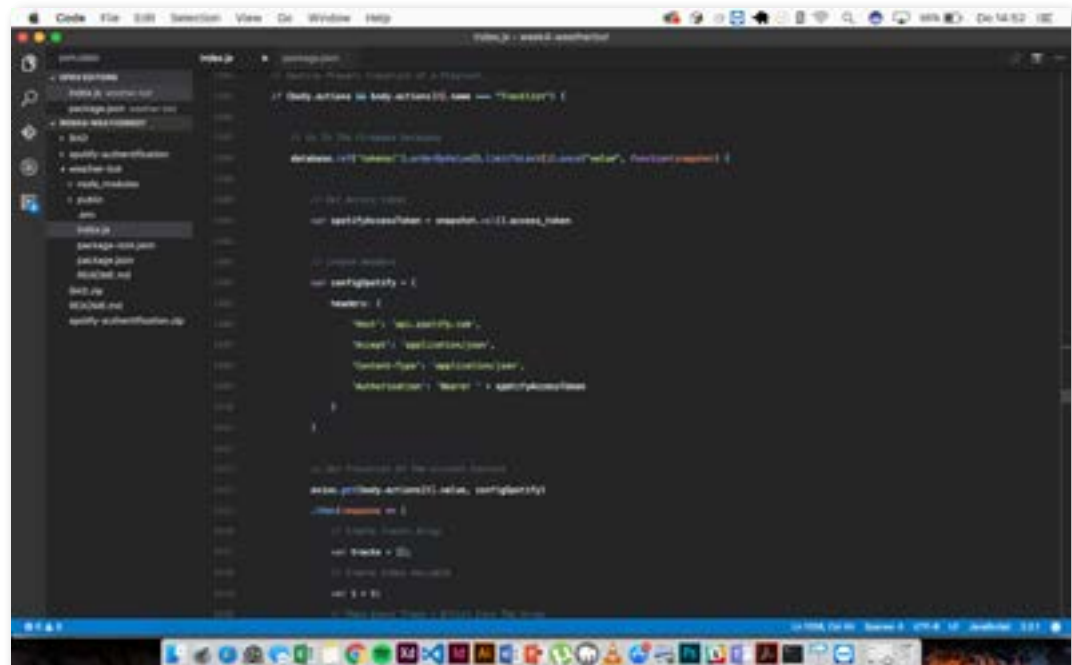
3.4. Visual Designs

De visual designs zitten in de bijlage als Adobe XD.

4. Develop

4.1. Code snippets

Chatbot 01



```
import express
import jwt
import bcrypt
import config

app.post('/login', async (req, res) => {
  // Get the username and password
  const { username, password } = req.body

  // Find the user in the database
  const user = await User.findOne({ username })

  // Check if the user exists
  if (!user) {
    return res.status(401).json({ message: 'User not found' })
  }

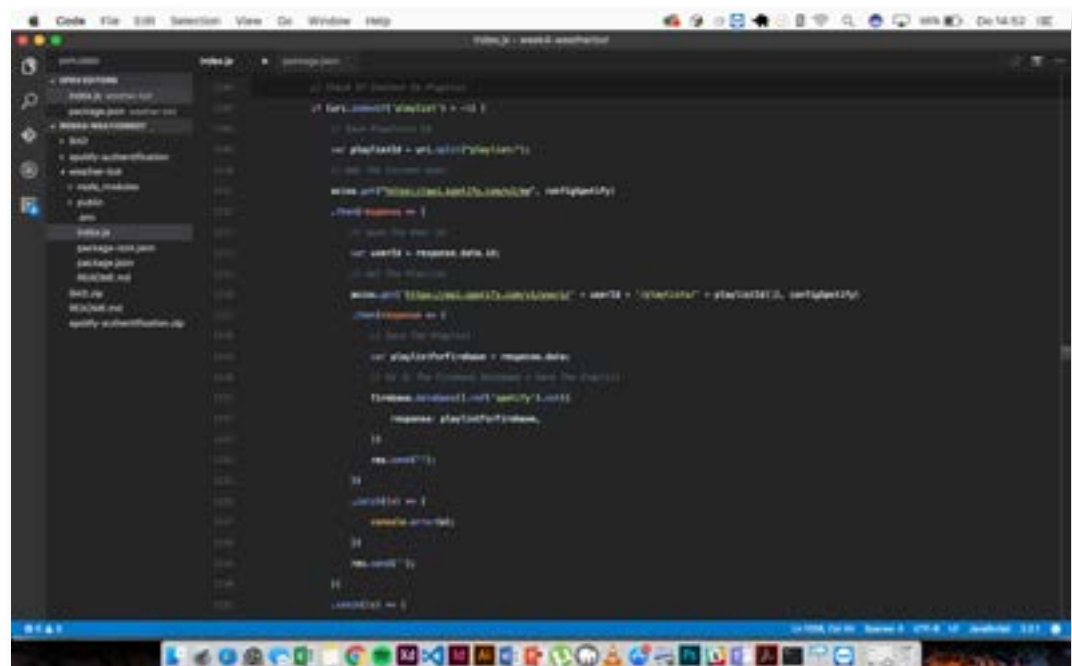
  // Compare the password
  const isMatch = await bcrypt.compare(password, user.password)

  // If the password is correct, generate a token
  if (isMatch) {
    const token = jwt.sign({ username }, config.jwtSecret, { expiresIn: '1h' })

    // Return the token and user information
    return res.json({ token, user })
  }

  // If the password is incorrect, return an error
  return res.status(401).json({ message: 'Invalid password' })
})
```

Chatbot 02



```
import express
import axios
import config

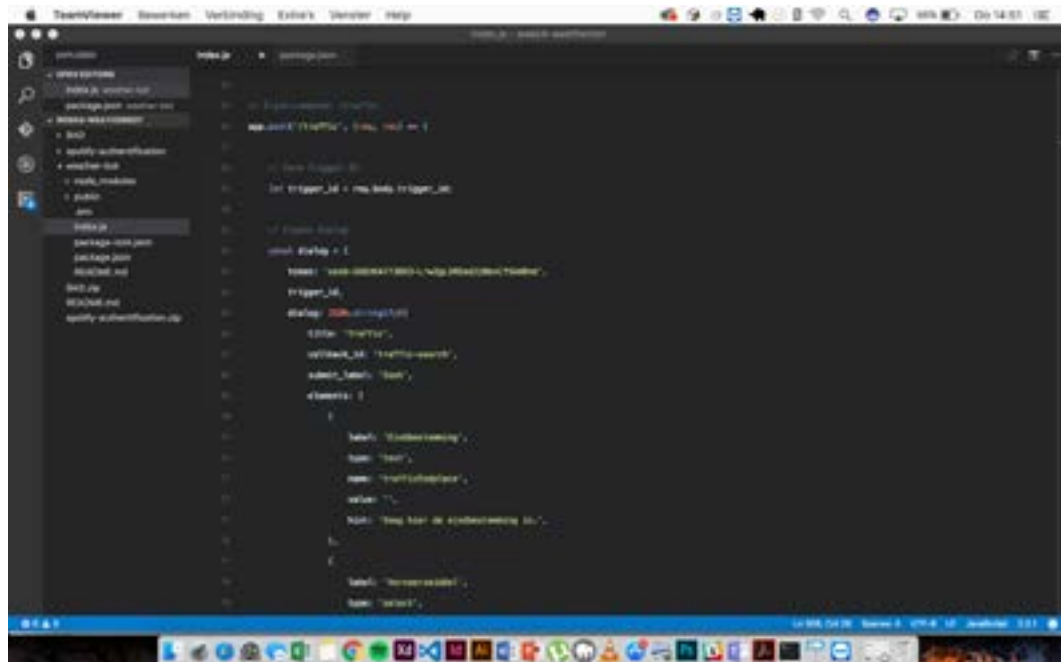
app.get('/chat', async (req, res) => {
  // Get the message from the user
  const { message } = req.query

  // Call the OpenAI API
  const response = await axios.post(
    'https://api.openai.com/v1/chat/completions',
    {
      model: 'gpt-3.5-turbo',
      messages: [
        { role: 'system', content: 'You are a helpful assistant.' },
        { role: 'user', content: message }
      ],
    },
    {
      headers: {
        'Authorization': `Bearer ${config.openAIAPIKey}`,
      },
    }
  )

  // Get the response from the API
  const { choices } = response.data

  // Return the response
  return res.json({ message: choices[0].text })
})
```

Chatbot 03



```

{
  "name": "chatbot",
  "version": "1.0.0",
  "description": "A chatbot application",
  "main": "index.js",
  "scripts": {
    "start": "node index.js"
  },
  "dependencies": {
    "express": "^4.17.1",
    "body-parser": "^1.19.0",
    "morgan": "^1.10.0",
    "helmet": "^3.12.1",
    "cors": "^2.8.5",
    "dotenv": "^8.2.0",
    "socket.io": "^2.3.0"
  },
  "devDependencies": {
    "nodemon": "^1.19.1"
  },
  "engines": {
    "node": "12.x"
  }
}

```

Chatbot 04



```

{
  "name": "chatbot",
  "version": "1.0.0",
  "description": "A chatbot application",
  "main": "index.js",
  "scripts": {
    "start": "node index.js"
  },
  "dependencies": {
    "express": "^4.17.1",
    "body-parser": "^1.19.0",
    "morgan": "^1.10.0",
    "helmet": "^3.12.1",
    "cors": "^2.8.5",
    "dotenv": "^8.2.0",
    "socket.io": "^2.3.0"
  },
  "devDependencies": {
    "nodemon": "^1.19.1"
  },
  "engines": {
    "node": "12.x"
  }
}

```

Spotify Login 01

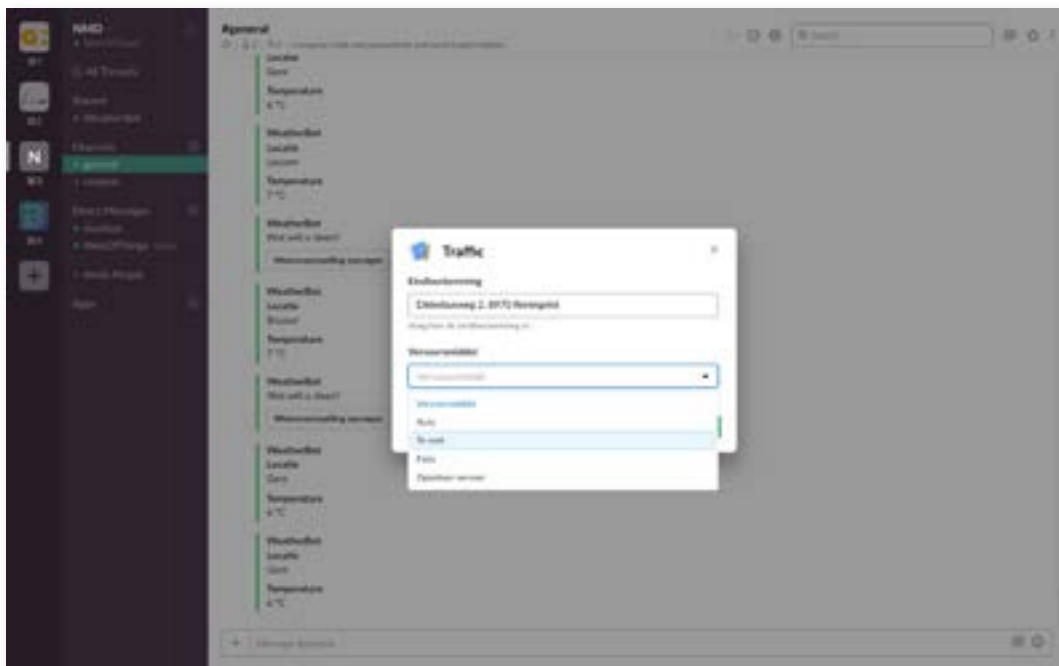
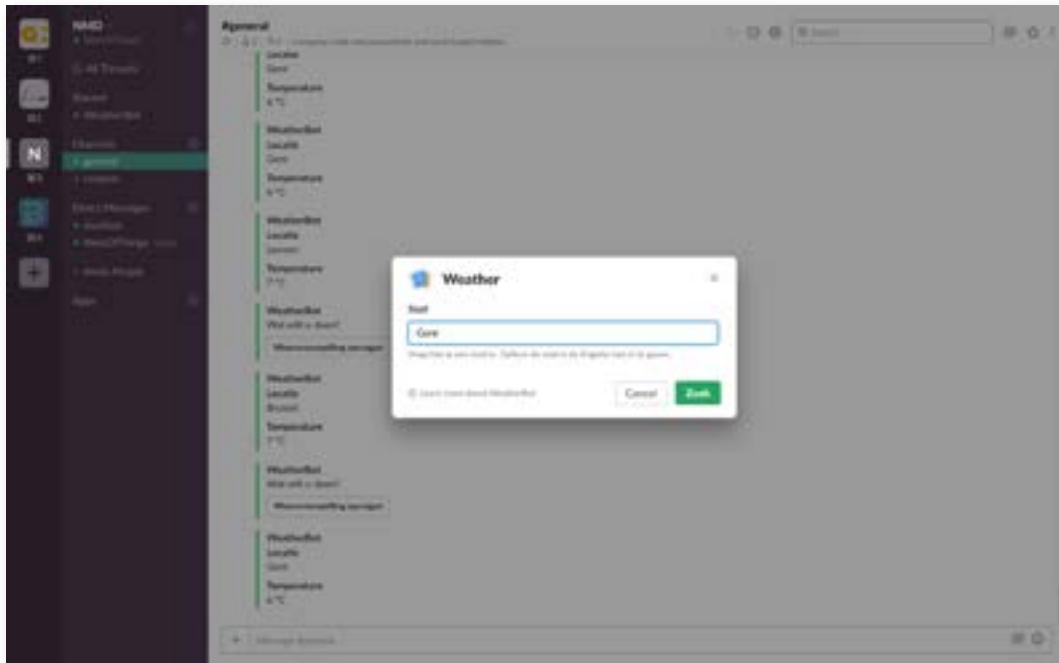
```

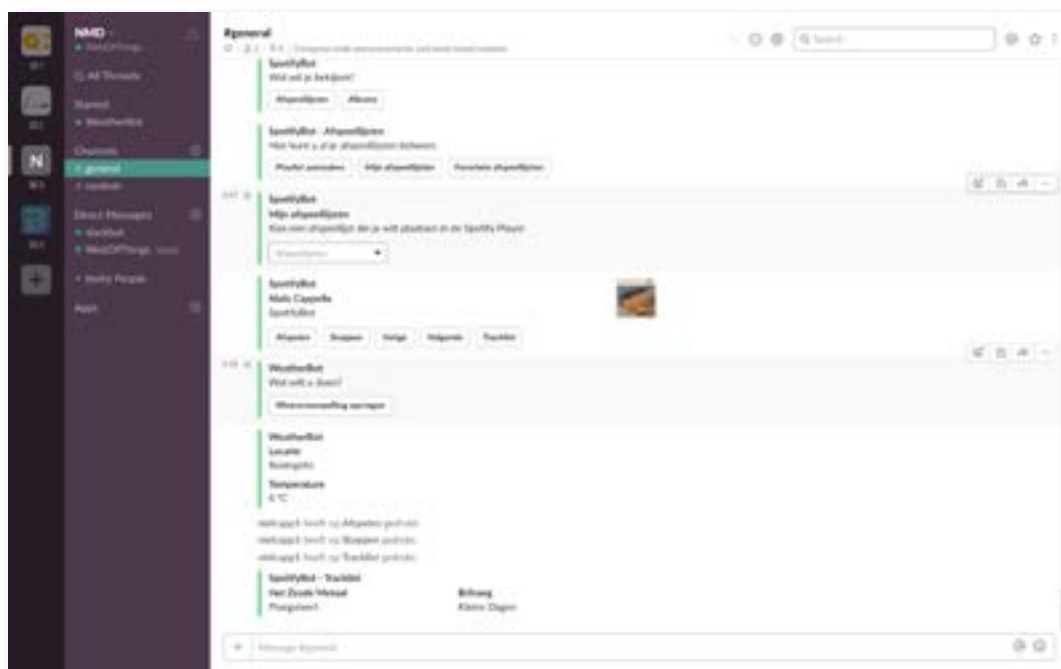
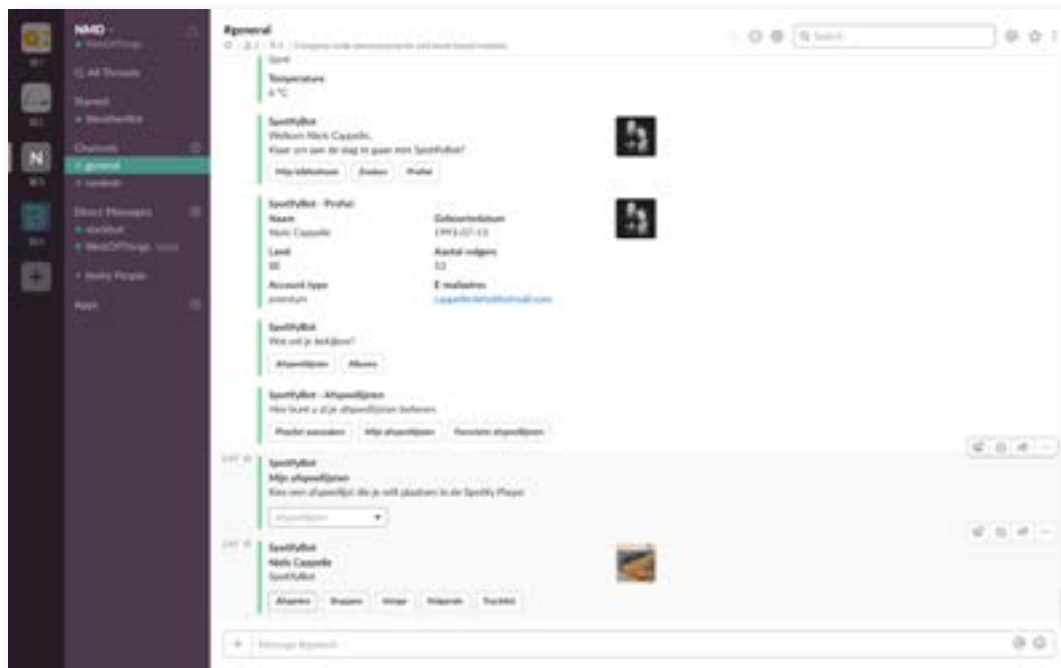
40 req.get('realIp').replace(/, /g, ' ');
41
42 // user information requests format and access token
43 // After checking the state parameter
44
45 var state = req.query.state || null;
46 var state = req.query.state || null;
47 var stateState = req.cookies ? req.cookies[stateState] : null;
48
49 if (state == null || state != stateState) {
50   res.redirect('/?err = '
51     + encodeURIComponent(
52       'error: "state_mismatch"')
53     + '&');
54 } else {
55   req.clearCookie(stateState);
56   var options = {
57     url: 'https://connect.qq.com/oauth2/authorize',
58     form: {
59       code: code,
60       redirect_uri: redirect_uri,
61       grant_type: 'authorization_code'
62     },
63     headers: {
64       Authorization: 'Basic ' + (new Buffer('client_id:' + client_id + ':' + client_secret)).toString('base64')
65     },
66     json: true
67   };
68
69   request.post(options, function(err, response, body) {
70     if (err || response.statusCode != 200) {
71
72       // access_token = body.access_token,
73       // refresh_token = body.refresh_token;
74
75       var options = {
76         url: 'https://api.qq.com/oauth2/authorize',
77         headers: {
78           'Authorization': 'Basic ' + (new Buffer('client_id:' + client_id + ':' + client_secret)).toString('base64')
79         },
80         json: true
81       };
82
83       // console.log('url:' + url);
84
85       // console.log('url:' + url);
86
87       // console.log('url:' + url);
88
89       // console.log('url:' + url);
90
91       // console.log('url:' + url);
92
93       // console.log('url:' + url);
94
95       // console.log('url:' + url);
96
97       // console.log('url:' + url);
98
99       // console.log('url:' + url);
100
101       // console.log('url:' + url);
102
103       // console.log('url:' + url);
104
105       // console.log('url:' + url);
106
107       // console.log('url:' + url);
108
109       // console.log('url:' + url);
110
111       // console.log('url:' + url);
112
113       // console.log('url:' + url);
114
115       // console.log('url:' + url);
116
117       // console.log('url:' + url);
118
119       // console.log('url:' + url);
120
121       // console.log('url:' + url);
122
123       // console.log('url:' + url);
124
125       // console.log('url:' + url);
126
127       // console.log('url:' + url);
128
129       // console.log('url:' + url);
130
131       // console.log('url:' + url);
132
133       // console.log('url:' + url);
134
135       // console.log('url:' + url);
136
137       // console.log('url:' + url);
138
139       // console.log('url:' + url);
140
141       // console.log('url:' + url);
142
143       // console.log('url:' + url);
144
145       // console.log('url:' + url);
146
147       // console.log('url:' + url);
148
149       // console.log('url:' + url);
150
151       // console.log('url:' + url);
152
153       // console.log('url:' + url);
154
155       // console.log('url:' + url);
156
157       // console.log('url:' + url);
158
159       // console.log('url:' + url);
160
161       // console.log('url:' + url);
162
163       // console.log('url:' + url);
164
165       // console.log('url:' + url);
166
167       // console.log('url:' + url);
168
169       // console.log('url:' + url);
170
171       // console.log('url:' + url);
172
173       // console.log('url:' + url);
174
175       // console.log('url:' + url);
176
177       // console.log('url:' + url);
178
179       // console.log('url:' + url);
180
181       // console.log('url:' + url);
182
183       // console.log('url:' + url);
184
185       // console.log('url:' + url);
186
187       // console.log('url:' + url);
188
189       // console.log('url:' + url);
190
191       // console.log('url:' + url);
192
193       // console.log('url:' + url);
194
195       // console.log('url:' + url);
196
197       // console.log('url:' + url);
198
199       // console.log('url:' + url);
200
201       // console.log('url:' + url);
202
203       // console.log('url:' + url);
204
205       // console.log('url:' + url);
206
207       // console.log('url:' + url);
208
209       // console.log('url:' + url);
210
211       // console.log('url:' + url);
212
213       // console.log('url:' + url);
214
215       // console.log('url:' + url);
216
217       // console.log('url:' + url);
218
219       // console.log('url:' + url);
220
221       // console.log('url:' + url);
222
223       // console.log('url:' + url);
224
225       // console.log('url:' + url);
226
227       // console.log('url:' + url);
228
229       // console.log('url:' + url);
230
231       // console.log('url:' + url);
232
233       // console.log('url:' + url);
234
235       // console.log('url:' + url);
236
237       // console.log('url:' + url);
238
239       // console.log('url:' + url);
240
241       // console.log('url:' + url);
242
243       // console.log('url:' + url);
244
245       // console.log('url:' + url);
246
247       // console.log('url:' + url);
248
249       // console.log('url:' + url);
250
251       // console.log('url:' + url);
252
253       // console.log('url:' + url);
254
255       // console.log('url:' + url);
256
257       // console.log('url:' + url);
258
259       // console.log('url:' + url);
260
261       // console.log('url:' + url);
262
263       // console.log('url:' + url);
264
265       // console.log('url:' + url);
266
267       // console.log('url:' + url);
268
269       // console.log('url:' + url);
270
271       // console.log('url:' + url);
272
273       // console.log('url:' + url);
274
275       // console.log('url:' + url);
276
277       // console.log('url:' + url);
278
279       // console.log('url:' + url);
280
281       // console.log('url:' + url);
282
283       // console.log('url:' + url);
284
285       // console.log('url:' + url);
286
287       // console.log('url:' + url);
288
289       // console.log('url:' + url);
290
291       // console.log('url:' + url);
292
293       // console.log('url:' + url);
294
295       // console.log('url:' + url);
296
297       // console.log('url:' + url);
298
299       // console.log('url:' + url);
300
301       // console.log('url:' + url);
302
303       // console.log('url:' + url);
304
305       // console.log('url:' + url);
306
307       // console.log('url:' + url);
308
309       // console.log('url:' + url);
310
311       // console.log('url:' + url);
312
313       // console.log('url:' + url);
314
315       // console.log('url:' + url);
316
317       // console.log('url:' + url);
318
319       // console.log('url:' + url);
320
321       // console.log('url:' + url);
322
323       // console.log('url:' + url);
324
325       // console.log('url:' + url);
326
327       // console.log('url:' + url);
328
329       // console.log('url:' + url);
330
331       // console.log('url:' + url);
332
333       // console.log('url:' + url);
334
335       // console.log('url:' + url);
336
337       // console.log('url:' + url);
338
339       // console.log('url:' + url);
340
341       // console.log('url:' + url);
342
343       // console.log('url:' + url);
344
345       // console.log('url:' + url);
346
347       // console.log('url:' + url);
348
349       // console.log('url:' + url);
350
351       // console.log('url:' + url);
352
353       // console.log('url:' + url);
354
355       // console.log('url:' + url);
356
357       // console.log('url:' + url);
358
359       // console.log('url:' + url);
360
361       // console.log('url:' + url);
362
363       // console.log('url:' + url);
364
365       // console.log('url:' + url);
366
367       // console.log('url:' + url);
368
369       // console.log('url:' + url);
370
371       // console.log('url:' + url);
372
373       // console.log('url:' + url);
374
375       // console.log('url:' + url);
376
377       // console.log('url:' + url);
378
379       // console.log('url:' + url);
380
381       // console.log('url:' + url);
382
383       // console.log('url:' + url);
384
385       // console.log('url:' + url);
386
387       // console.log('url:' + url);
388
389       // console.log('url:' + url);
390
391       // console.log('url:' + url);
392
393       // console.log('url:' + url);
394
395       // console.log('url:' + url);
396
397       // console.log('url:' + url);
398
399       // console.log('url:' + url);
400
401       // console.log('url:' + url);
402
403       // console.log('url:' + url);
404
405       // console.log('url:' + url);
406
407       // console.log('url:' + url);
408
409       // console.log('url:' + url);
410
411       // console.log('url:' + url);
412
413       // console.log('url:' + url);
414
415       // console.log('url:' + url);
416
417       // console.log('url:' + url);
418
419       // console.log('url:' + url);
420
421       // console.log('url:' + url);
422
423       // console.log('url:' + url);
424
425       // console.log('url:' + url);
426
427       // console.log('url:' + url);
428
429       // console.log('url:' + url);
430
431       // console.log('url:' + url);
432
433       // console.log('url:' + url);
434
435       // console.log('url:' + url);
436
437       // console.log('url:' + url);
438
439       // console.log('url:' + url);
440
441       // console.log('url:' + url);
442
443       // console.log('url:' + url);
444
445       // console.log('url:' + url);
446
447       // console.log('url:' + url);
448
449       // console.log('url:' + url);
450
451       // console.log('url:' + url);
452
453       // console.log('url:' + url);
454
455       // console.log('url:' + url);
456
457       // console.log('url:' + url);
458
459       // console.log('url:' + url);
460
461       // console.log('url:' + url);
462
463       // console.log('url:' + url);
464
465       // console.log('url:' + url);
466
467       // console.log('url:' + url);
468
469       // console.log('url:' + url);
470
471       // console.log('url:' + url);
472
473       // console.log('url:' + url);
474
475       // console.log('url:' + url);
476
477       // console.log('url:' + url);
478
479       // console.log('url:' + url);
480
481       // console.log('url:' + url);
482
483       // console.log('url:' + url);
484
485       // console.log('url:' + url);
486
487       // console.log('url:' + url);
488
489       // console.log('url:' + url);
490
491       // console.log('
```

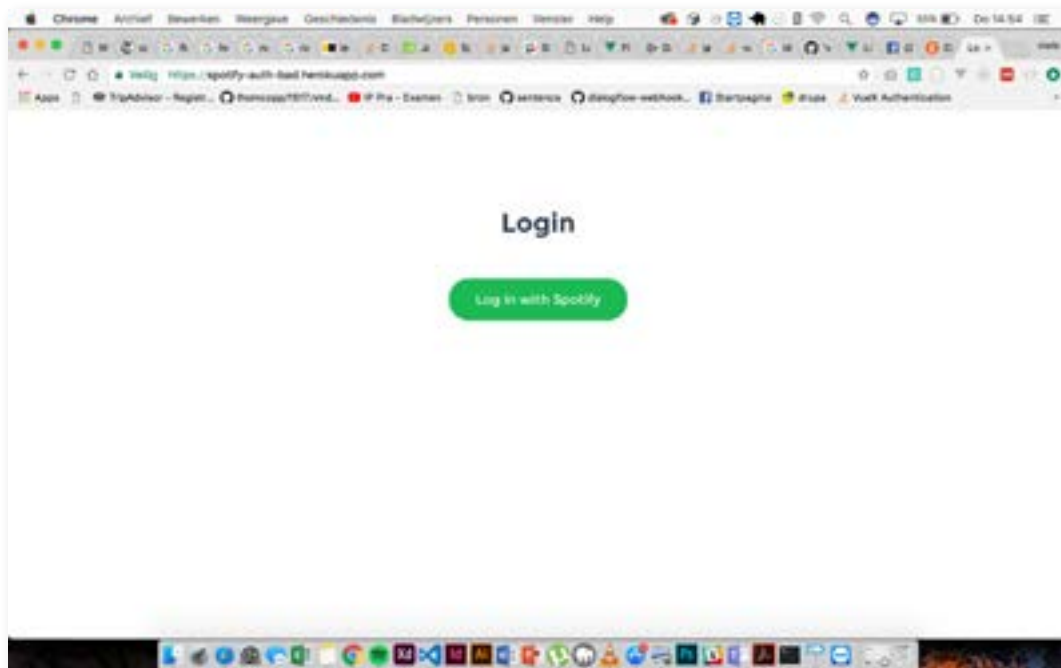
Dashboard 01

[illegible]

4.2. Schermafbeeldingen









5. Deliver

5.1. Handleiding

BAD staat voor Bot Agency Dashboard. Het is een chatbot die in verbinding staat met een dashboard. Het dashboard vangt alle informatie op die is verworven aan de hand van de chatbot.

Om aan de slag te gaan moet je beschikken over de volgende zaken:

- Device met de Slack App
- Raspberry PI
- LED matrix voor de Raspberry PI
- Computer met groot scherm voor dashboard

Nodig nu de bot uit op een kanaal in Slack via de volgende link:

```
https://slack.com/oauth/authorize?client_id=261064196212.267563655600&scope=bot,commands,incoming-webhook
```

Login op de Spotify Bot Login pagina:

```
https://spotify-auth-bad.herokuapp.com/
```

Laat het dashboard draaien op een computer met een groot scherm zoals in de Deployment Guide beschreven staat.

Je kan nu de bot aanspreken met de volgende Slack commands:

```
/Spotify
```

```
/Traffic
```

```
/Weather
```

Let op, je kan enkel '/Spotify' gebruiken met een Premium account.

Daarnaast kan je ook emoticons weergeven op de display door een emoticon te sturen naar WeatherBot (de naam van de Bot). Hiervoor moet je het Node.js script uitvoeren op de Raspberry PI uit de map LED-server. De LED-matrix moet je op zijn beurt ook connecteren met de de Raspberry PI.

6. Deploy

6.1. Deployment Guide

6.1.1. Bot

Dit is een node.js project die draait op de volgende link:

```
https://warm-tor-26768.herokuapp.com
```

6.1.2. Spotify Login

```
https://spotify-auth-bad.herokuapp.com/
```

6.1.3. Dashboard

```
npm install
```

```
npm run dev
```

6.1.4. Led Matrix

```
Node index.js
```