

Twitterbot IOT project

Internet of things

Academiejaar 2019-2020

Groep 32: Kyle Vandeputte & Gillian Peralta

New Media Development

Bachelor in de grafische en digitale media

Arteveldehogeschool

Inhoudstafel

Discover	5
Define	6
Analyse	6
Noodzakelijke soft- en hardware	9
Inspiratie	9
Design	10
Develop	11
Code snippets	11
Schermafbeeldingen / foto's van de uitwerking	13
Deliverables	14
Handleiding	14
Timesheet	18

Discover

We maken via developer twitter een soort twitter account aan en verbinden die met behulp van twit. Twit is een Twitter API Client voor node. Dan verbinden we deze client aan de raspberry.

Op de raspberry zelf creëren we een database waar er informatie wordt opgehaald om te plaatsen zoals afbeeldingen en een bijkomende tekst met daarbij ook een link naar de informatie die geplaatst wordt. We zouden met node.js werken die dit dan doorgeeft naar Twitter en een post plaatst op ons gemaakt account. Via een interval kunnen we dan kiezen om de hoeveel tijd een afbeelding geplaatst wordt. Elke antwoord die we ontvangen op onze tweet wordt doorgestuurd naar de api van de printer van 2 collega's.

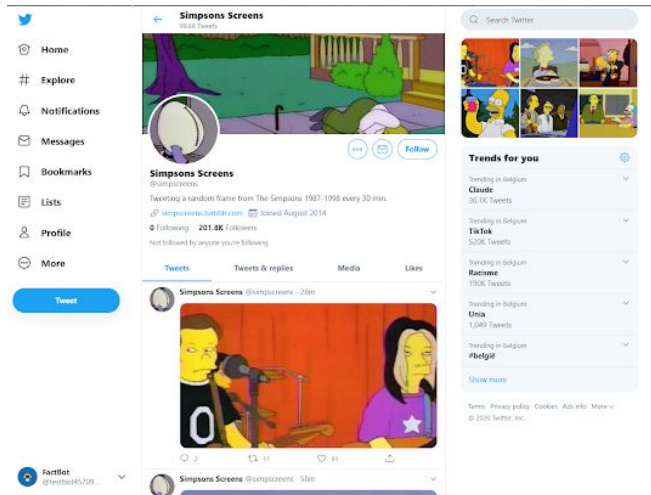
Define

Analyse

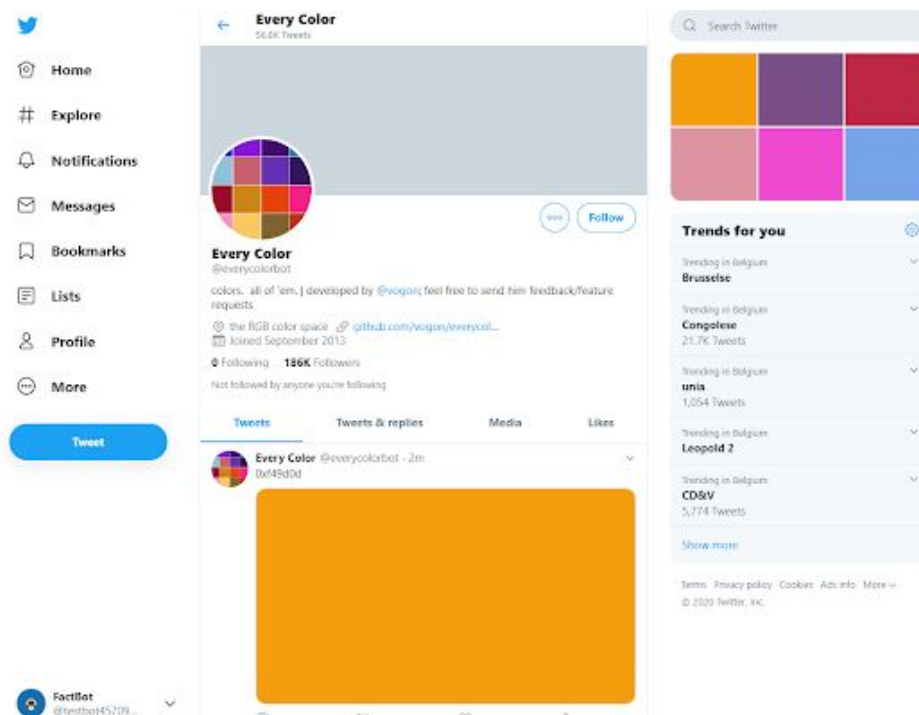
Concurrent 1: Actual Fact Bot 5337+1 op Facebook. Deze sociale media bot post ook random facts maar dit doet hij op Facebook terwijl wij ons focussen op Twitter.



Concurrent 2: Simpsons Screens op twitter. Deze sociale media bot post willekeurige frames van The Simpsons tussen 1987-1988 elke 30 min.



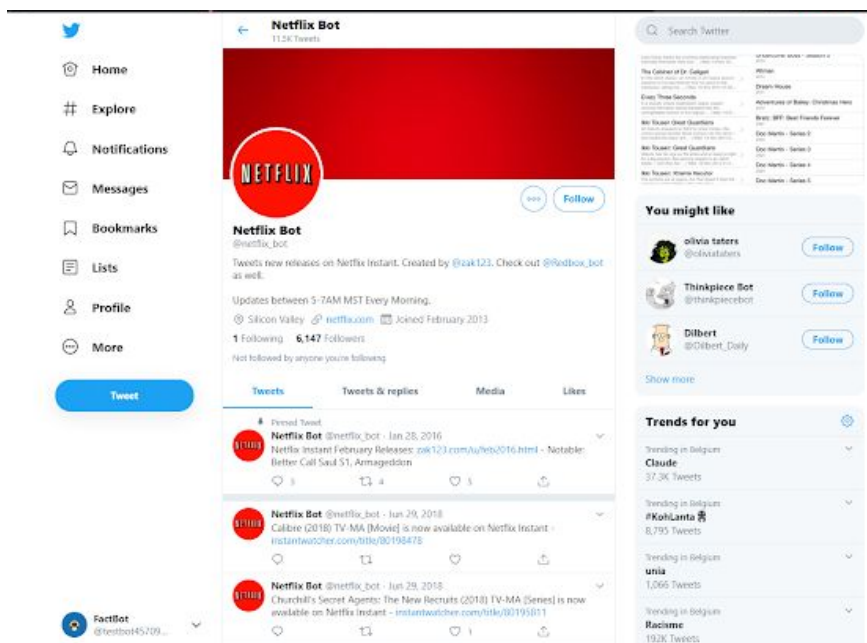
Concurrent 3: Every Color. Deze sociale media bot bevindt zich ook op Twitter. Hij post elk uur een willekeurige kleur gegenereerd door een willekeurige hex code.



Concurrent 4: Earthquake Robot. Deze sociale media bot post elke aardbeving groter dan een schaal van 5.0 terwijl ze aan het gebeuren zijn.



Concurrent 5: Netflix Bot. Deze sociale media bot tweet elke nieuwe release die uitkomt op netflix.



Noodzakelijke soft- en hardware

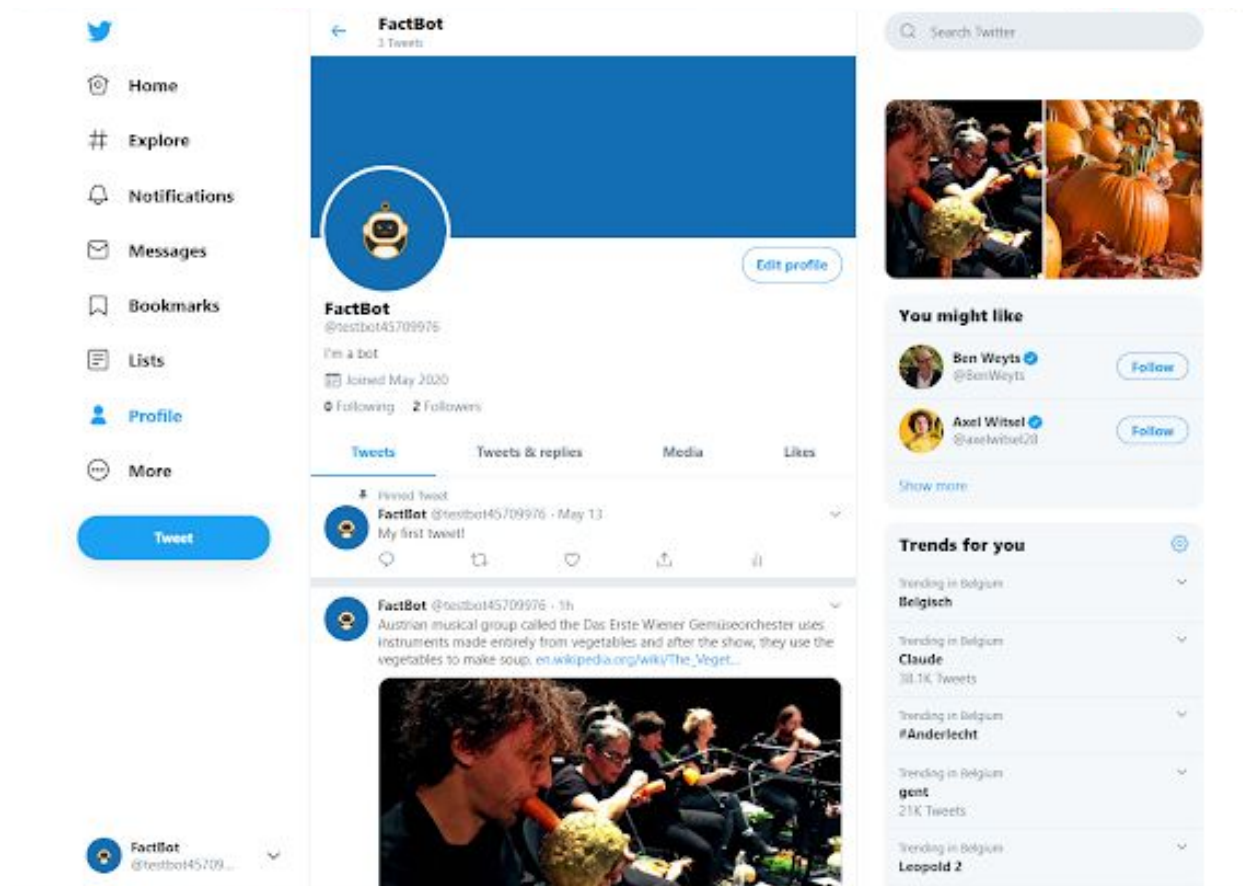
- Raspberry pi 4
- Node.js
- Javascript
- Twitter developer
- Twit (API)

Inspiratie

We zijn op dit idee gekomen doordat we allebei de Actual Fact Bot 5337+1 volgen op facebook en het leek ons een goed idee om dit eens na te maken.

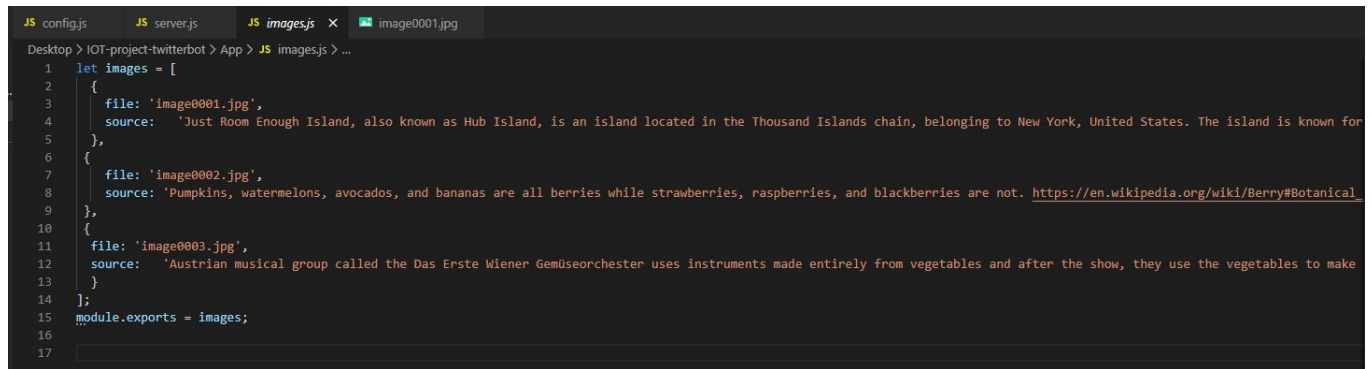
Tegenwoordig vinden mensen het ook zo leuk om dagelijks wat dingen te leren door middel van random weetjes op social media, dit vonden we dus zeer interessant om er een eigen versie van uit te werken om te zien hoe handig de raspberry pi eigenlijk is.

Design



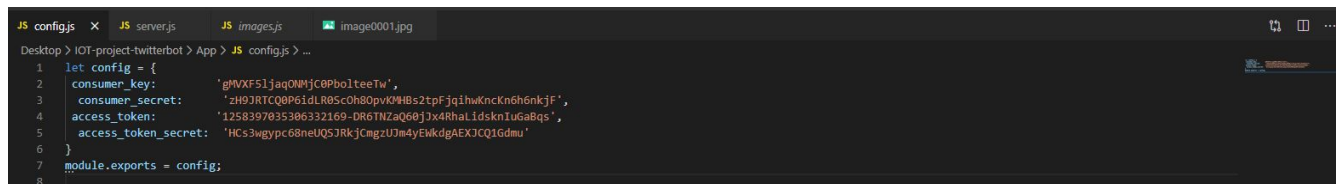
Develop

Code snippets



```
1 let images = [
2   {
3     file: 'image0001.jpg',
4     source: 'Just Room Enough Island, also known as Hub Island, is an island located in the Thousand Islands chain, belonging to New York, United States. The island is known for
5   },
6   {
7     file: 'image0002.jpg',
8     source: 'Pumpkins, watermelons, avocados, and bananas are all berries while strawberries, raspberries, and blackberries are not. https://en.wikipedia.org/wiki/Berry#Botanical
9   },
10  {
11    file: 'image0003.jpg',
12    source: 'Austrian musical group called the Das Erste Wiener Gemüseorchester uses instruments made entirely from vegetables and after the show, they use the vegetables to make
13  }
14 ];
15 module.exports = images;
16
17
```

Deze array zorgt ervoor dat we tekst en een bron kunnen linken aan onze afbeeldingen die getweet worden.



```
1 let config = {
2   consumer_key: 'gMVXFS1jaqQNMjC0PbolteeTw',
3   consumer_secret: 'zH9JRTCQ0P6idLR0Sc0h80pvKMHbs2tpFjqihwKncKn6h6nkjF',
4   access_token: '1258397035306332169-DR6TNZaQ60jJx4RhalLidskn1uGaBqs',
5   access_token_secret: 'HCS3wgypc68neUQSJRkjcmgzUJm4yEWkdGAEXJCQ1Gdmu'
6 }
7 module.exports = config;
8
```

Instellen van onze API keys van Twitter zodat onze bot toegang heeft tot Twitter.

```
JS config.js JS server.js X JS images.js image0001.jpg
Desktop > IOT-project-twitterbot > App > JS server.js > ...
22 function random_image_uploader(){
23   console.log('Afbeelding aan het zoeken...');
24   let image_path = path.join(__dirname, '/images/' + image.file),
25       b64content = fs.readFileSync(image_path, { encoding: 'base64' });
26
27   console.log('Afbeelding aan het uploaden...');
28
29   T.post('media/upload', { media_data: b64content }, function (err, data) {
30     if (err){
31       console.log('ERROR:');
32       console.log(err);
33     }
34     else{
35       console.log('Afbeelding geupload!');
36       console.log('Nu de afbeelding aan het tweeten...');
37
38       T.post('statuses/update', { status: (image.source),
39         media_ids: new Array(data.media_id_string)
40       },
41         function(err) {
42           if (err){
43             console.log('ERROR:');
44             console.log(err);
45           }
46           else{
47             console.log('Afbeelding gepost!');
48           }
49         }
50       );
51     }
52   });
53 }
54 fs.readdir(__dirname + '/images', function(err, files) {
55   if (err){
56     console.log(err);
57   }
58   else{
59     let images = [];
60     files.forEach(function(f) {
61       images.push(f);
62     });
63
64     setInterval(function(){
65       random_image_uploader(images);
66     }, 10000);
67   }
68 });
69
```

Deze functie zorgt ervoor dat er een willekeurige afbeelding uit de database wordt gehaald en wordt geüpload naar Twitter met een in te stellen interval.

Schermafbeeldingen / foto's van de uitwerking

Hieronder ziet u het proces en het eindresultaat wanneer de code wordt uitgevoerd.

Je ziet dat de afbeelding wordt gezocht en dan wordt geüpload, daarna wordt hij via twitter getweet/gepost.

```
pi@raspberrypi:~/Desktop/IOT-project-twitterbot/App $ node server.js
Afbeelding aan het zoeken...
Afbeelding aan het uploaden...
Afbeelding geupload!
Nu de afbeelding aan het tweeten...
Afbeelding gepost!
```



Deliverables

Handleiding

Stap 1:

Installeer Node.js op de Raspberry Pi

Stap 2:

Creëer de code om de twitterbot te maken:

Begin met een map aan te maken:

```
mkdir twitterbot
```

Dan verander je naar deze directory:

```
cd twitterbot
```

Maak hier een file server.js:

```
sudo nano server.js
```

Stap 3:

Installeer de Twit library met npm:

```
npm init
```

Installeer de Twit module van npm:

```
npm install twit --save
```

Wacht tot de files gedownload zijn.

Stap 4:

Open server.js en voeg de volgende code toe:

```
let fs = require('fs'),  
    path = require('path'),  
    Twit = require('twit'),  
    config = require(path.join(__dirname, 'config.js'));
```

Sla op en sluit af.

Stap 5:

Maak een Twitter app door te surfen naar developer.twitter.com (het verificatieproces kan enkele dagen duren)

Eenmaal aangemaakt kun je de gewenste consumer keys en acces tokens vinden die je nodig hebt om je bot toegang te geven tot Twitter.

Stap 6:

Maak config.js aan om deze keys en tokens in te steken.

```
sudo nano config.js
```

Hierin steek je het volgende.

```
let config = {  
  consumer_key: 'XXXXXX',  
  consumer_secret: 'XXXXXX',  
  access_token: 'XXXXXX',  
  access_token_secret: 'XXXXXX'  
}  
module.exports = config;
```

Waar XXXX staat vul je de gekregen API key details in.

Stap 7:

Maak je map met afbeeldingen aan.

Stap 8:

In server.js voeg je volgende code toe:

```
let T = new Twit(config);  
const image = random_from_array(images)  
  
function random_from_array(images) {  
  return images[Math.floor(Math.random() * images.length)];  
}
```

Deze code neemt een willekeurige afbeelding uit de database.

Wanneer dit gedaan is moet je een tweede function toevoegen:

```
function random_image_uploader(){
  console.log('Afbeelding aan het zoeken...');
  let image_path = path.join(__dirname, '/images/' +
image.file),
    b64content = fs.readFileSync(image_path, { encoding:
'base64' });

  console.log('Afbeelding aan het uploaden...');

  T.post('media/upload', { media_data: b64content }, function
(err, data) {
    if (err){
      console.log('ERROR:');
      console.log(err);
    }
    else{
      console.log('Afbeelding geupload!');
      console.log('Nu de afbeelding aan het tweeten...');

      T.post('statuses/update', { status: (image.source),
        media_ids: new Array(data.media_id_string)
      },
        function(err) {
          if (err){
            console.log('ERROR:');
            console.log(err);
          }
          else{
            console.log('Afbeelding gepost!');
          }
        }
      );
    }
  });
}

fs.readdir(__dirname + '/images', function(err, files) {
  if (err){
    console.log(err);
  }
}
```



```

    else{
      let images = [];
      files.forEach(function(f) {
        images.push(f);
      });

      setInterval(function(){
        random_image_uploader(images);
      }, 10000);
    }
  });

```

Deze functie neemt een willekeurige afbeelding uit de afbeelding database en daarna upload hij deze naar Twitter door middel van de API.

Onderaan staat het interval, deze zorgt ervoor dat je tijd kan steken tussen elke upload wat zeker aangeraden wordt.

Stap 9:

Tekst toevoegen aan de foto's met behulp van een array:

Begin met het aanmaken van images.js:

```
sudo nano images.js
```

Hierin voeg je de volgende code toe:

```

let images = [
  {
    file: 'image0001.jpg',
    source: 'Tekst + bron'
  },
  {
    file: 'image0002.jpg',
    source: 'Tekst + bron'
  },
  {
    file: 'image0003.jpg',
    Source: 'Tekst + bron'
  }
];

```

Dit is een array met twee elementen, file en source. Hierin sla je de filename en de URL op plus eventuele tekst die je bij source kan zetten.

Herhaal zoals nodig voor elke afbeelding.

Eindig dan het document met volgende code:

```
module.exports = images;
```

Sla op en doe het bestand dicht.

Dan open je server.js en voeg je volgende code toe onder const image:

```
const images = require(path.join(__dirname, 'images.js'));
```

Sla alles op en run dan server.js in de terminal en de Twitterbot is klaar voor gebruik!

Timesheet

Brainstormen: 45 min

Aanmaken van Twitter account: 10 min

Aanvraag van Twitter developer account: 2 dagen

Aanmaken van de App op twitter: 20 - 30 min

Installeren benodigde software: 5 min

Afbeeldingen + informatie opzoeken: 30 min

Research code: 120 min

Code schrijven: 3 u en 15 min