

Denoi gertatu zaigu noizbait. Twitterren lasai asko zabiltzala behar sozial bat badagoela jabetzen zara:

Por favor, que alguien cree un bot que responda "Cálmese, señora" a cada tuit de Rosa Díez.

— Jordan (@jordansiedad) [November 2, 2017](#)

Teknologia behar sozialei erantzuteko dagoenez, bota nola egin ikusiko dugu gaur.

Baino guk etzibota egingo dugu. Izan ere, goiko txioak proposatzen duen [bota egin dute iada:](#)

Cálmese, señora.

— Rosa Díez exaltada (@RosaExaltada) [November 6, 2017](#)



Beraz, jar gaitezen etzibotarekin.

Aurrekoan [Telegrameko bota nola sortu](#) ikusi genuen. Gaurko honetan Twitter sare sozialean [@etzipm](#) erabiltzailearen txioak nola automatizatu ikusiko dugu.

Bot honek etzi.pm webak ea albiste berririk duen begiratuko du orduro eta baldin badu txiokatu egingo du albistea.

## Twitter aplikazioa sortu

Twitterrekin lotura egiteko token batzuk eskatu behar dizkiogu twitterri. Hortarako <http://apps.twitter.com>-era joango gara. Logeatu ondoren aplikazio berria sortu botoiari eman eta bete eskatzen dituen parametroak.

The image displays two screenshots of the Twitter Application Management interface. The left screenshot shows the 'Twitter Apps' overview page with a 'Create New App' button and the 'Etzibota' app listed. The right screenshot shows the 'Etzibota' app details page with options for 'Test OAuth', 'Details', 'Settings', 'Keys and Access Tokens', and 'Permissions'.

Consumer key eta consumer tokenak emango dizkigu berehala.

Ondoren beste bi token eskatu behar dira: Access token eta Access token secret-a

The screenshot shows the Twitter Application Management interface for an application named 'Etzibota'. The page is divided into several sections:

- Application Management Header:** Includes the Twitter logo, the application name 'Etzibota', and a 'Test OAuth' button.
- Navigation Menu:** Contains links for 'Details', 'Settings', 'Keys and Access Tokens' (which is highlighted), and 'Permissions'.
- Application Settings:** A section with a warning: 'Keep the "Consumer Secret" a secret. This key should never be human-readable in your application.' It lists the 'Consumer Key (API Key)' as '27a2Tb80hcESweSwhf' and the 'Consumer Secret (API Secret)' as 'm70IZ05XE93iHvHhVw4ZeyJLHK'.
- Your Access Token:** A section providing details for the current access token:
 

Access Token	2389312916-MCayVThtfEYppU06Pr
Access Token Secret	IDJP53Avve3uVE25urAGu6
Access Level	Read and write
Owner	etzipm
Owner ID	2389312916
- Token Actions:** A section with two buttons: 'Regenerate My Access Token and Token Secret' and 'Revoke Token Access'.

## Bot-aren kodea

Zerbitzaria sortzen hasteko fitxategi bat sortuko dugu proiektuarentzat:

```
$ mkdir etzi-bot
```

Proiektua jarriko dugun fitxategira sartu:

```
$ cd etzi-bot
```

Npm instalatu eta proiektua hasieratu:

```
$ sudo apt-get install npm
$ npm init
```

Exekutatzeko behar ditugun dependenziak instalatu:

```
$ npm install --save twit xml2js request path fs
$ touch azkena.txt
```

Twitterreko Tokenak sartzeko config.js fitxategia sortuko dugu.

```
$ sudo nano config.js
```

Bertan hau idatziko dugu: (Gure tokenekin, noski!)

```
/** TWITTER APP CONFIGURATION
 * consumer_key
 * consumer_secret
 * access_token
 * access_token_secret
 */
module.exports = {
  consumer_key: 'LK6wgfdgewfwdflalwmdk',
  consumer_secret: 'Sq5km213123ewfwe1231231kqD8F45t45tdfsdaA2',
  access_token: '11111278-Vpd0sdgs7ZVFnwAaQVYs41R',
  access_token_secret: 'YQ9twxUq321fd1321rdqwfwe2wqwr2312e12e'
}
```

konfigurazio fitxategia sortuko dugu giltzak beste fitxategi batean edukitzeko

```
$ sudo nano index.js
```

*index.js* dokumentuan hau sartuko dugu:

```
"use strict";

const request = require("request");
const parseString = require("xml2js").parseString;
const stripPrefix = require("xml2js").processors.stripPrefix;
const fs = require("fs");
const path = require("path");
////twit
const twit = require("twit");
const config = require("./config.js");

var registered = [];
var lastTitle = "";
var twitErab = "";
```

```
//////////twitter konfigurazioa
const Twitter = new twit(config);

////////// ea azkena.txt badagoen

const AZKENA_FILE = process.env.DB_PATH
  ? path.join(process.env.DB_PATH, "azkena.txt")
  : "azkena.txt";

//////////irakurri piztean azken titulua
var lastTitle = fs.readFileSync(AZKENA_FILE).toString();

//////////
//24 orduro exekutatzen den lan bat jartzen dugu martxan hemen
setInterval(() => {
  console.log("setInterval begiratu berririk baden:");
  request("https://etzi.pm/feed/", (err, res, body) => {
    parseString(body, { tagNameProcessors: [stripPrefix] }, (err, result) =>
    {
      let newest = result.rss.channel[0].item[0];
      if (lastTitle === newest.title[0]) {
        console.log("Ez dago albiste berririk: " + lastTitle);
        return;
      }
      //
      fs.unlinkSync(AZKENA_FILE);
      //console.log('azkena gorde'+newest.title[0]);
      fs.appendFileSync(AZKENA_FILE, newest.title[0]);
      //

      lastTitle = newest.title[0];
      twitErab = newest.creator;

      // twit
      Twitter.post(
        "statuses/update",
        {
          status: twitErab + " - " + lastTitle + ": " + newest.link + "
#etziBot"
        },
        function(err, data, response) {
          if (err) {
            console.log("Error in tweeting azken posta");
          } else {
            console.log("Azken posta shown successfully");
          }
        }
      );
    }
  });
});
```

```
    });  
  });  
}, 1000 * 60 * 24); //24 orduro begiratzen du
```

## Zerbitzaria martxan jarri

Sortutako kodea nahi den zerbitzarian jar dezakegu martxan. Edonork zerbitzari propioa ez dugunez, now zerbitzua erabiliko dugu:

```
$ sudo npm install -g now
```

#package.json sortu edo gainidatzi:

```
$ nano package.json
```

Gure package.json horrela geratuko da:

```
{  
  "name": "twittols-bot",  
  "version": "1.0.0",  
  "description": "etzibot",  
  "main": "index.js",  
  "author": "tolosa",  
  "scripts": {  
    "test": "echo \"Error: no test specified\" && exit 1",  
    "start": "node index.js"  
  },  
  "repository": {  
    "type": "git",  
    "url": "git+https://github.com/utolosa002/twittols-bot.git"  
  },  
  "license": "ISC",  
  "bugs": {  
    "url": "https://github.com/utolosa002/twittols-bot/issues"  
  },  
  "homepage": "https://github.com/utolosa002/twittols-bot#readme",  
  "dependencies": {  
    "fs": "0.0.1-security",  
    "path": "^0.12.7",  
    "request": "^2.83.0",  
    "twit": "^2.2.9",  
    "xml2js": "^0.4.19"  
  }  
}
```



Hasierako script-a gehitu ondoren, exekutatu honako komandoa proiektua dagoen karpetan

bertan:

```
$ now #zerbitzua zerbitzarian martxan jarriko du
```

Zerbitzua ondo jarri badu, martxan horrelako zerbait erantzungo du:

```
> Deploying ~/etzi-bot  
> Using Node.js 8.9.0 (default)  
> Ready! https://etzi-bot-zzzzz.now.sh  
> You are on the OSS plan. Your code will be made public.
```

Dena ondo egin badugu gure Bot-a martxan edukiko dugu eta artikulu berriren bat dagoenean [automatikoki txiokatuko du](#).

Kode guztia: [github](#)

