



Figure 1: GitHub

# Contents

<b>Voorwoord</b>	<b>2</b>
Over dit boek . . . . .	2
<b>Processing opstarten op cursuslaptop</b>	<b>3</b>
Terminal starten . . . . .	3
Processing starten . . . . .	7
Processing slimmer starten . . . . .	7
Updaten . . . . .	8
Eindppdracht . . . . .	8
<b>GitHub</b>	<b>8</b>
<b>:exclamation: Waarschuwing :exclamation:</b>	<b>8</b>
GitHub aanmaken . . . . .	10
Ga naar GitHub . . . . .	10
Maak jezelf lid . . . . .	11
Kies je account . . . . .	11
Vrienden maken . . . . .	13
Bevestig je emailadres . . . . .	13
Een eigen GitHub website aanmaken . . . . .	13
Maak een nieuwe GitHub website . . . . .	13
Stel de nieuwe GitHub website in . . . . .	13
Zet dingen op je eigen GitHub website . . . . .	16
<b>GitHub Pages</b>	<b>19</b>
GitHub Page aanmaken . . . . .	19
GitHub Page repository maken . . . . .	20
HTML . . . . .	20
De homepage maken . . . . .	22
<b>Processing.js</b>	<b>25</b>
Processing.js downloaden . . . . .	25
Processing sketch uploaden . . . . .	28
<b>GitHub chat</b>	<b>30</b>
Wat is een chatroom? . . . . .	30
Gedragsregels . . . . .	30
Website . . . . .	31
IRC inloggen . . . . .	31
IRC server login . . . . .	31
IRC chatroom . . . . .	31
<b>git, GitHub samenwerken</b>	<b>37</b>
git, GitHub . . . . .	37
Sprookje . . . . .	37

De GitHub op je computer zetten . . . . .	37
Samenwerken . . . . .	38
Merge conflicts . . . . .	38
<b>Voorwoord</b>	<b>39</b>
Over dit boek . . . . .	39

## Voorwoord



Figure 1: Het logo van De Jonge Onderzoekers



Figure 2: Het logo van Codestarter

Dit is het Processing boek van de Dojo. Processing is een programmeertaal. Dit boek leert je die programmeertaal.

## Over dit boek

Dit boek heeft een CC-BY-NC-SA licensie.



Figure 3: De licensie van dit boek

(C) Dojo Groningen 2016

Het is nog een beetje een slordig boek. Zo staat bijvoorbeeld het plaatje dat eigenlijk op de kaft moet staan op pagina twee. Er zitten tiepvauten in en de opmaak is niet altijd even mooi.

Daarom staat dit boek op een GitHub. Om precies te zijn, op <https://github.com/richelbilderbeek/Dojo>. Hierdoor kan iedereen die dit boek te slordig vindt minder slordig maken.

## Processing opstarten op cursuslaptop

We programmeren in Processing. Processing is een programma. Dit programma moeten we dus eerst opstarten.

### Terminal starten

Eerst moeten we een terminal starten.

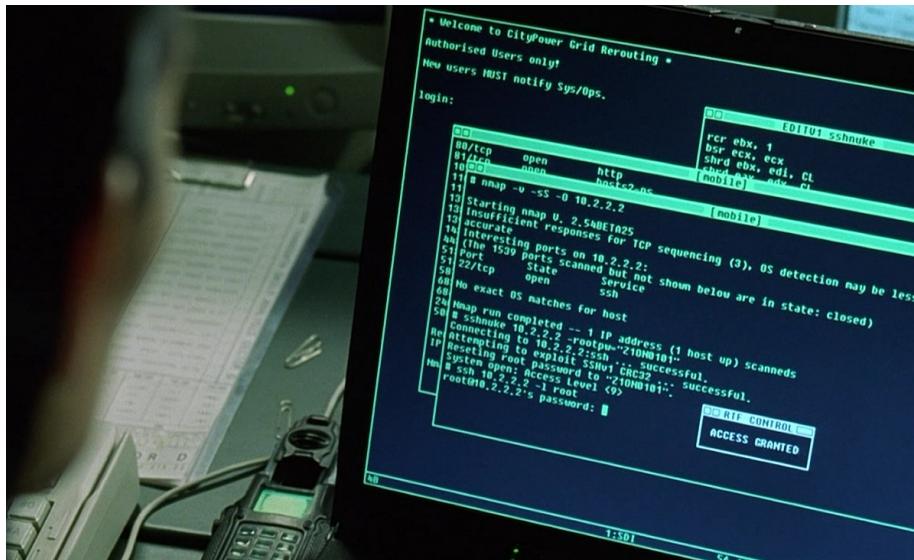


Figure 4: Hackers werken met een terminal

Wij hebben twee soorten laptops:

- Debian (sprek uit ‘Debie-jen’)
- Lubuntu (sprek uit ‘Loe-boen-toe’)

Start een Terminal:

- Debian: druk op de WIN toets (deze zit linksonder, tussen CTRL en ALT). Type dan ‘terminal’ en dan ENTER
- Lubuntu: druk op CTRL + ALT + T

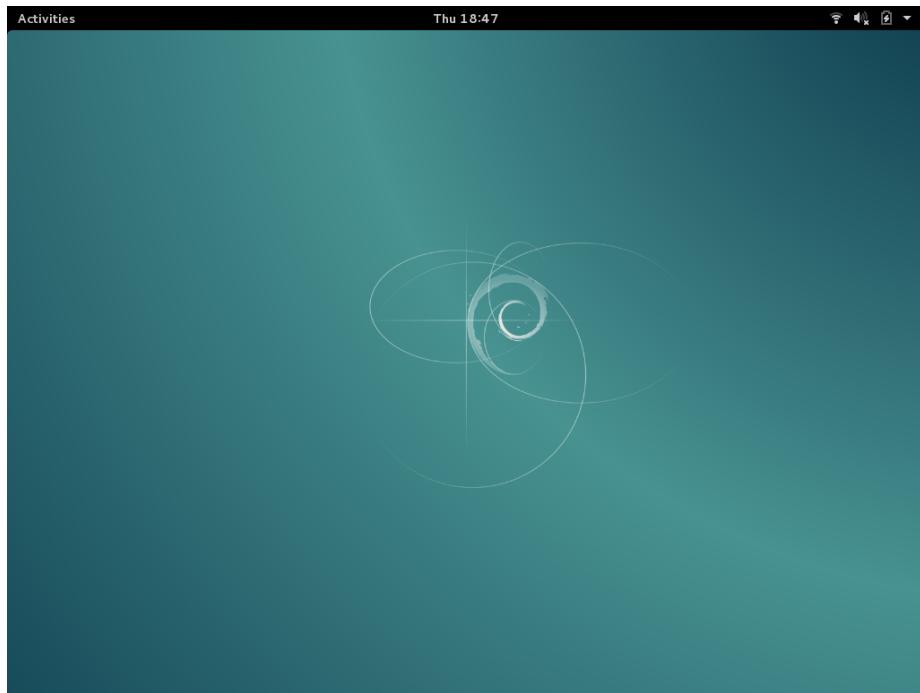


Figure 5: Debian: druk op WIN



Figure 6: De WIN toets zit tussen CTRL en ALT

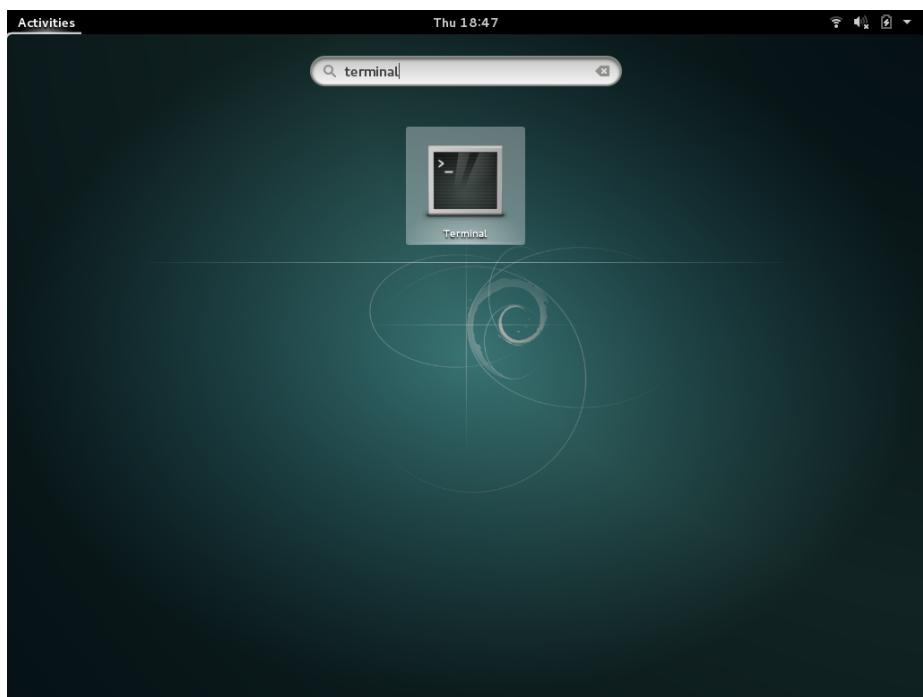


Figure 7: Debian: druk op WIN en type dan ‘terminal’

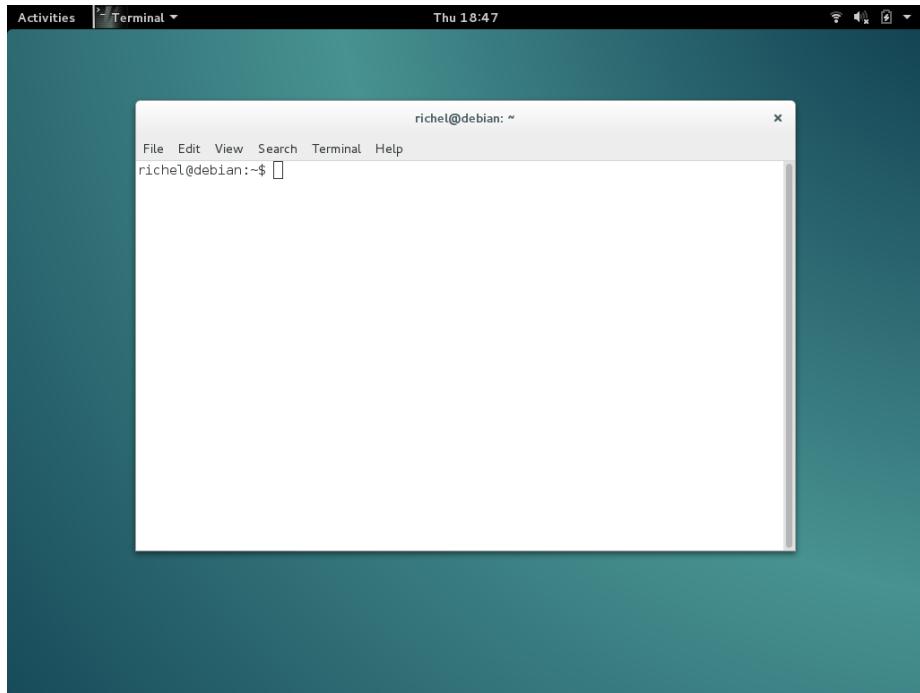


Figure 8: Debian: een terminal

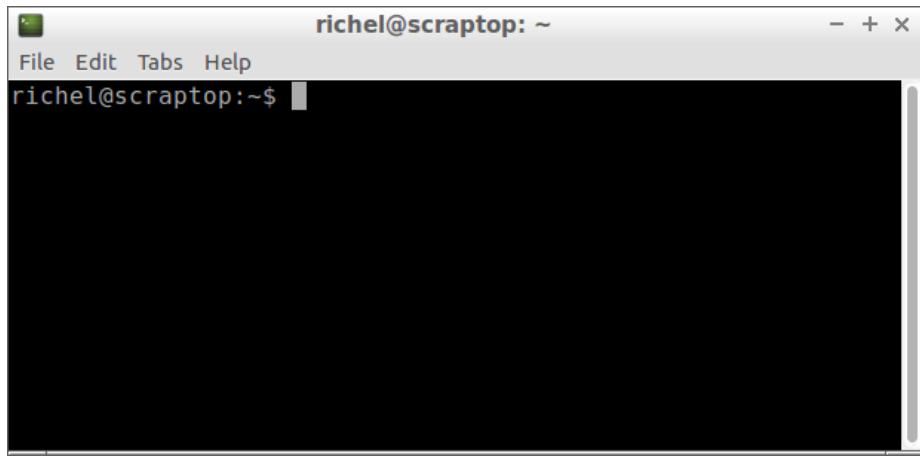


Figure 9: Lubuntu: een terminal

## **Processing starten**

In de terminal, type:

```
cd Programs/processing-3.2.1
```

En type dan:

```
./processing
```

Klaar!

## **Processing slimmer starten**

Je hoeft niet alles helemaal te typen. Een terminal kan je ook helpen. Als je in een terminal op Tab drukt, maakt deze het woord af.

In de terminal, type:

```
cd Progr
```

en druk dan op ‘Tab’.

Nu wordt staat er op je scherm:

```
cd Programs/
```

Maak ervan

```
cd Programs/pr
```

en druk dan op ‘Tab’.

Nu staat er:

```
cd Programs/processing-3.2.1
```

druk dan op Enter.

Doe nu:

```
./p
```

en druk dan op Tab. Nu staat er:

```
./processing
```

Druk op Enter.

Klaar!

## Updaten

Het kan gebeuren dat er een venster komt met een Update. Het venster vraagt of je een nieuwe versie van Processing wilt installeren. Klik op ‘No’. ‘No’ is Engels voor ‘nee’.



Figure 10: Processing Update venster, klik op ‘No’

## Eindppdracht

- Start Processing helemaal zonder hulp

## GitHub

GitHub is een site waar programmeurs samen aan programma's werken. Dit doen ze, omdat ze dan nooit meer hun code kwijtraken.

Wij gaan vandaag GitHub gebruiken om:

- Nooit meer code kwijt te raken
- Altijd bij onze code te kunnen

Wij gaan later GitHub gebruiken om:

- Samen te kunnen werken
- Onze games online te kunnen zetten

## :exclamation: Waarschuwing :exclamation:

GitHub is openbaar. Iedereen kan zien wat je doet. Doe dus alleen slimme dingen op GitHub:

- Blijf altijd vriendelijk

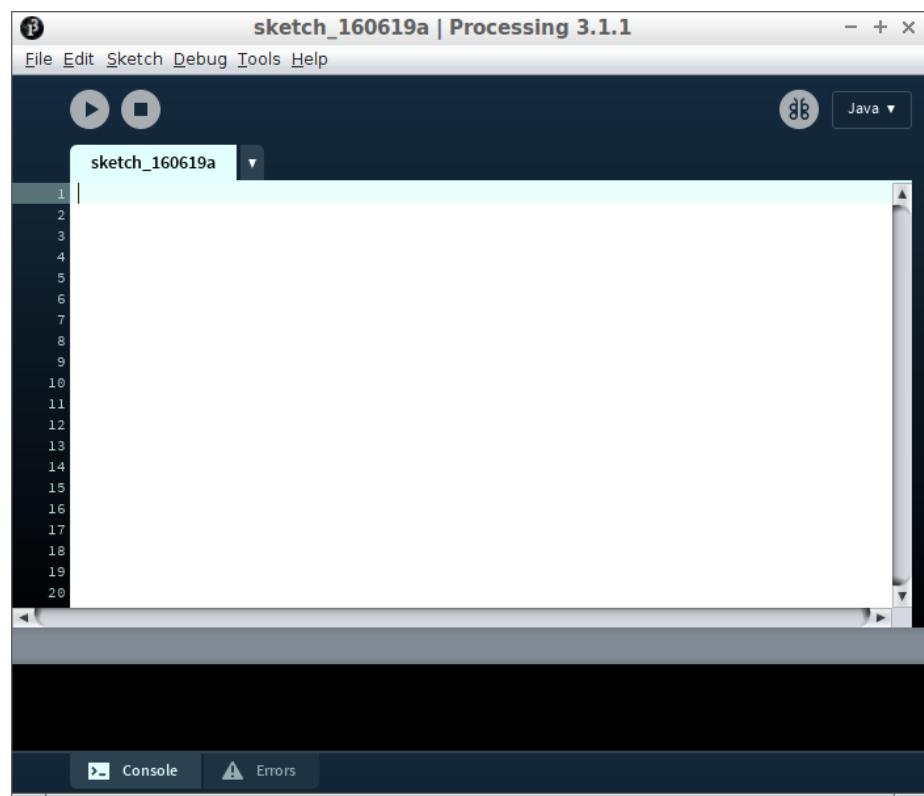


Figure 11: Processing zonder code

- Blijf altijd netjes
- Grappen zijn prima, maar nooit als iemand zich beleidg voelt

## GitHub aanmaken

In deze les maken we een GitHub account en gaan we onze code er op zetten.

### Ga naar GitHub

- Start een webbrowser, zoals Firefox, Chromium, Chrome, Edge, Internet Explorer, Opera
- Ga naar [www.github.com](https://www.github.com)

Je komt nu op de GitHub homepage:

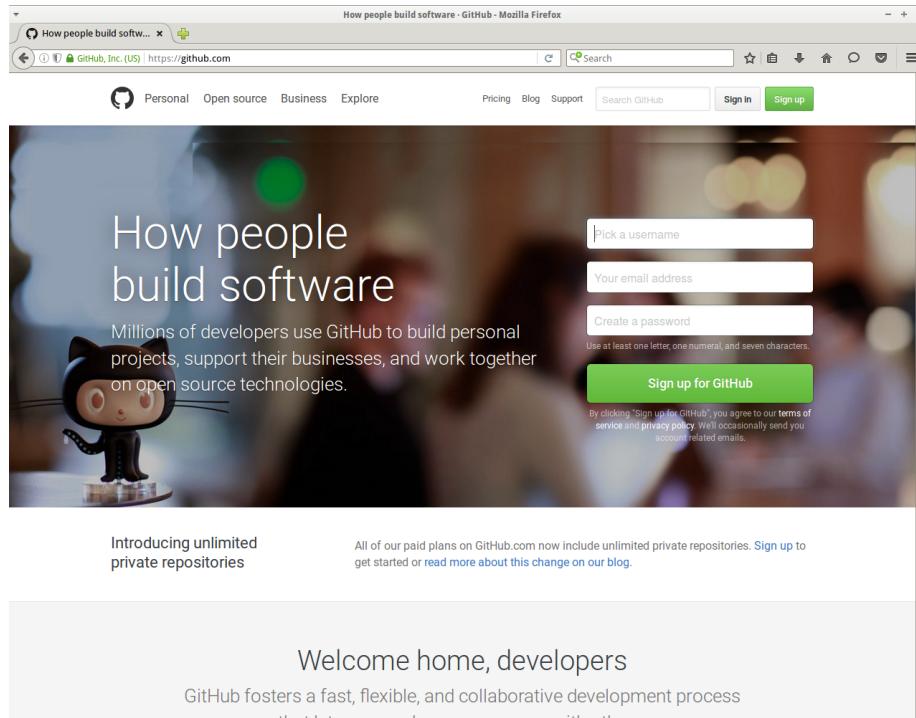


Figure 12: GitHub homepage

De kat heet Octocat en is de mascotte van GitHub.

## Maak jezelf lid

- Klik op Sign Up (NL: Inschrijven)

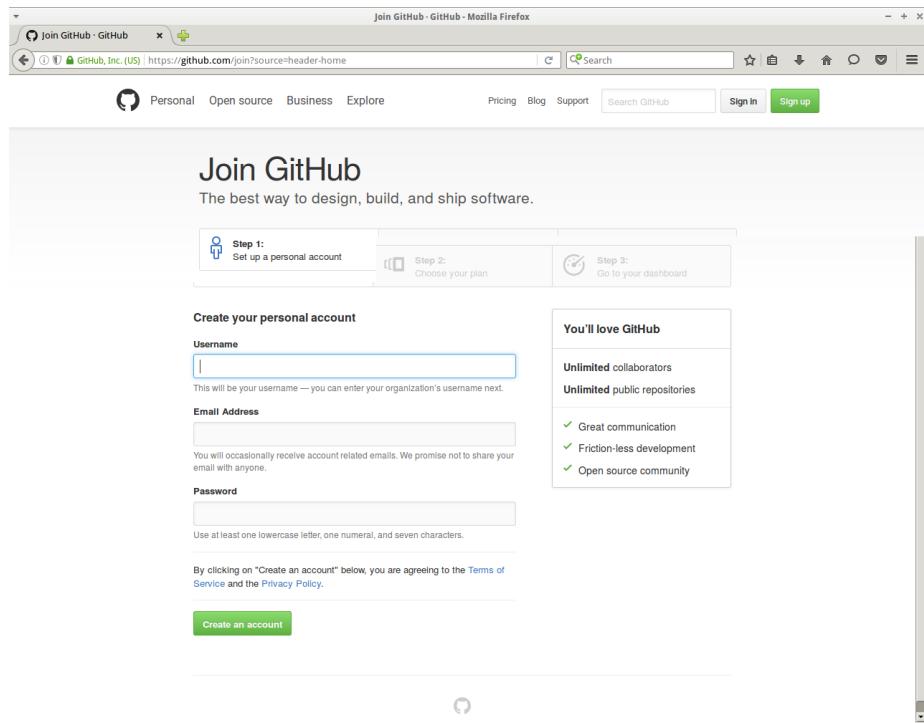


Figure 13: GitHub inschrijven

Vul hier je gegevens in:

- **Username:** je gebruikersnaam. Kies geen gebruikersnaam waar je later spijt van gaat krijgen
- **Email address:** je emailadres. Dit moet je echte emailadres zijn
- **Password:** je wachtwoord. Deze moet een cijfer bevatten, bijvoorbeeld `iloverichel4ever` is een goede
- Klik **Create an account** om een account aan te maken.

## Kies je account

In het volgende scherm moet je je soort account kiezen.

Wij gebruiken gewoon een gratis account

- Klik **Finish sign up** en je hebt een GitHub account!

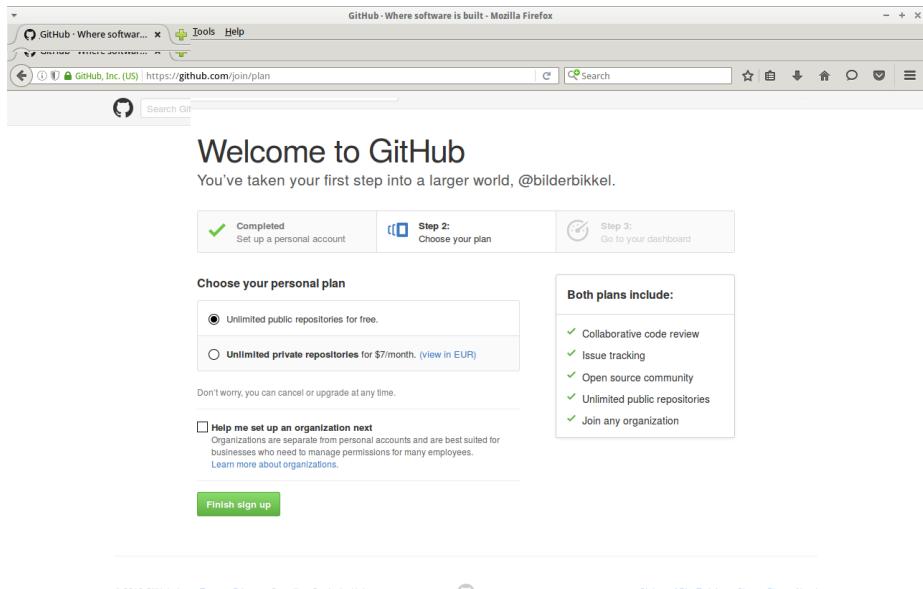


Figure 14: GitHub account kiezen

## Vrienden maken

Nu je een account hebt, kun je vrienden maken

- Ga naar <https://github.com/richelbilderbeek> en klik op Follow om Richel te volgen
- Ga naar <https://github.com/thijsvb> en klik op Follow om Thijs te volgen
- Ga naar <https://github.com/LJK1991> en klik op Follow om Lucas te volgen
- Ga naar <https://github.com/Modanung> en klik op Follow om Frode te volgen

## Bevestig je emailadres

- Ga naar je email toe
- Open het maitje van GitHub
- Klik op Confirm om te bevestigen dat je een GitHub account hebt

## Een eigen GitHub website aanmaken

:exclamation: Dit kan alleen als je je emailadres bevestigd hebt

### Maak een nieuwe GitHub website

In bijna elk scherm, is er een kruis rechts bovenaan:

- Klik op Create New Repository om een nieuwe GitHub webpagina aan te maken

### Stel de nieuwe GitHub website in

Vul hier in:

- Repository name: de naam van je GitHub website, bijvoorbeeld Mijncraft
- Description: omschrijving van je GitHub website, bijvoorbeeld Minecraft-achtig spel, gemaakt in Processing
- Kies Public
- Vink aan: Initialize this repository with a README
- Kies bij Add .gitignore: Processing
- Kies bij Add a license: GNU General Public License v3.0

Nu heb je je eigen GitHub website!

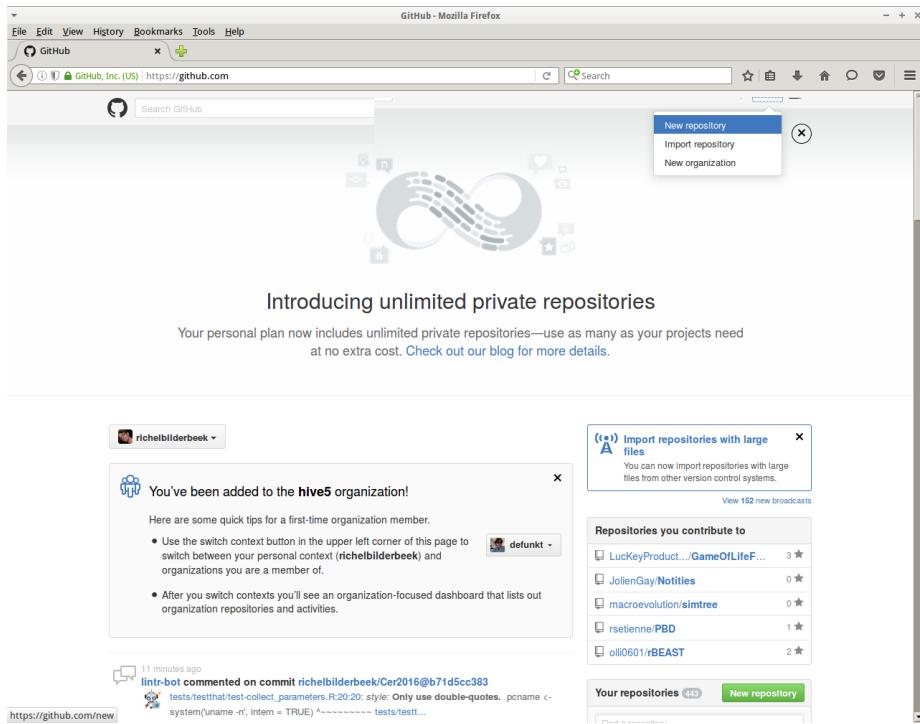


Figure 15: Maak een GitHub

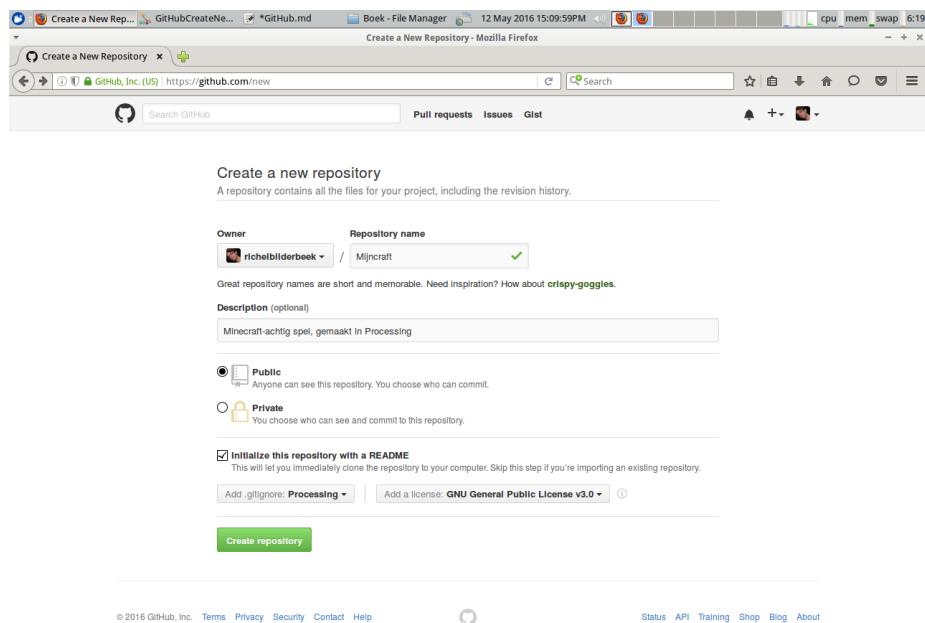


Figure 16: Maak een GitHub, eerste scherm

## Zet dingen op je eigen GitHub website

Op je eigen GitHub website kun je veel dingen doen.

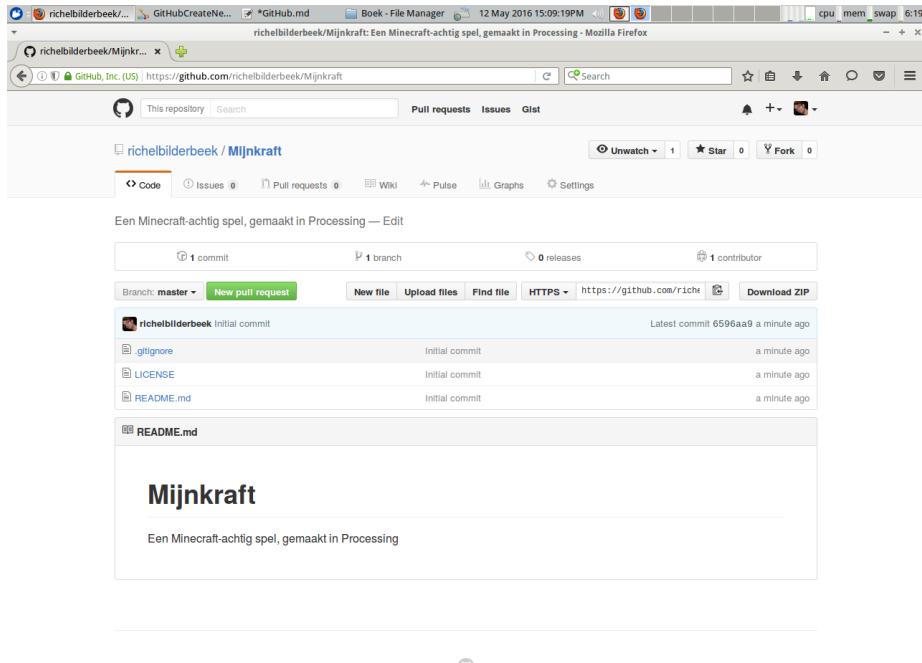


Figure 17: De Mijnkraft GitHub

- Klik op **Upload file** om een bestand op de website te zetten. Zet een mooi plaatje op de website.

Het bestand **README.md** bevat de welkomsttekst van je website. Dit gaan we aanpassen.

- Klik op het woord **README.md**
- Klik op het potloodje met de tekst **Edit this file**
- Zet hier iets leuks. Je kunt ook je plaatje laten zien, met bijvoorbeeld **! [Dit vind ik leuk] (plaatje.jpg)**
- Zet je code op GitHub

Als je nog geen GitHub account hebt, doe dan eerst deze les

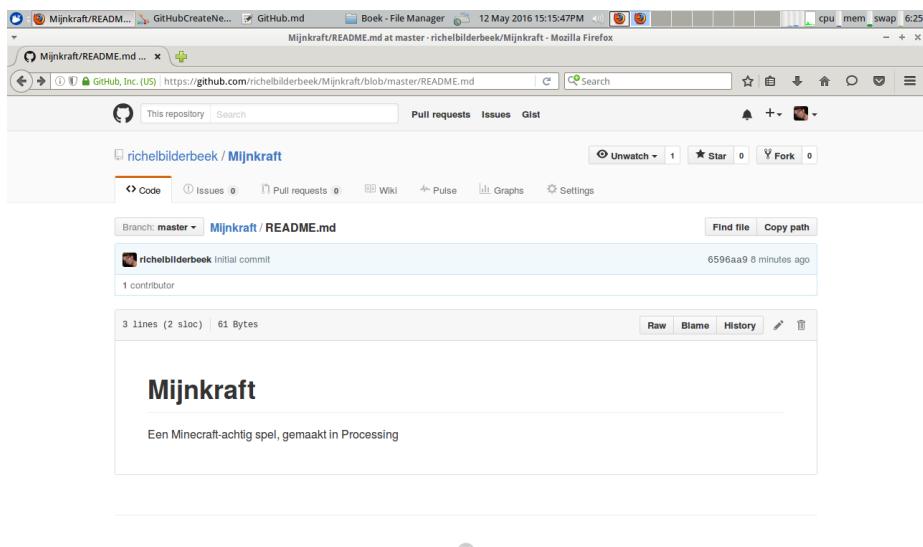


Figure 18: README.md wijzigen

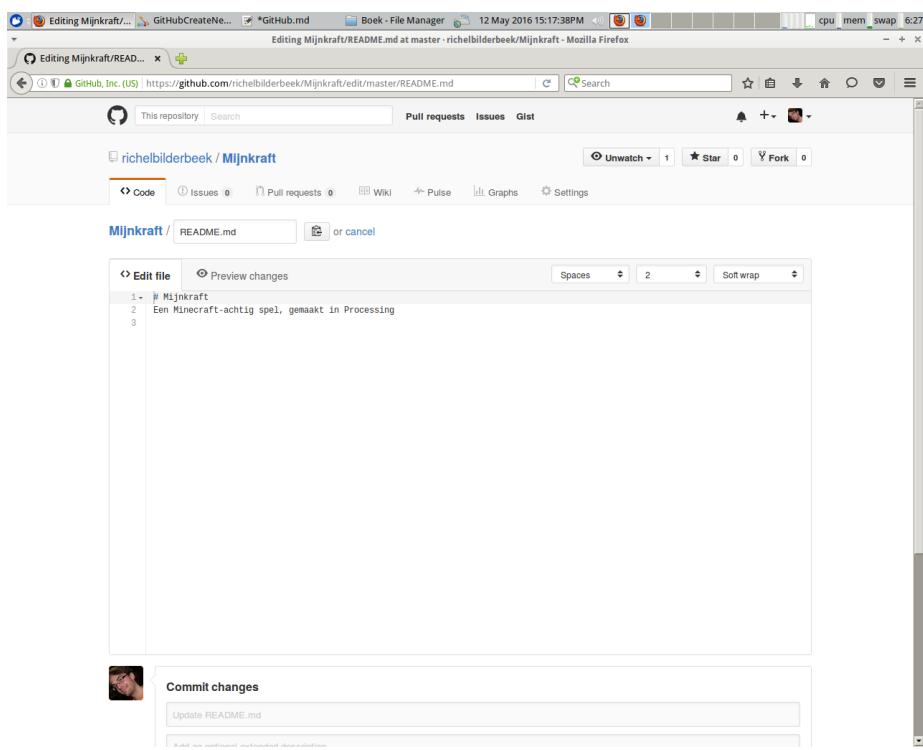


Figure 19: README.md wijzigen

## GitHub Pages

GitHub Pages is een service van GitHub waarmee elke gebruiker zijn eigen website kan maken. Deze les gaan we een GitHub Page met homepage maken.

Het verschil met een repository is dat die op [www.github.com/gebruiker](https://www.github.com/gebruiker) staat, terwijl een GitHub Page op [www.gebruiker.github.io](https://www.gebruiker.github.io) staat.

Een repository gebruik je om:

- \* Je code te bewaren
- \* Je code te delen met de wereld
- \* Samen aan code te werken

Een GitHub Page gebruik je om:

- \* Mensen jouw spellen te laten spelen
- \* De wereld te laten zien wat je kan

## GitHub Page aanmaken

- Open een webbrowser
- Ga naar [www.github.com](https://www.github.com)

Je komt nu op de GitHub homepage:

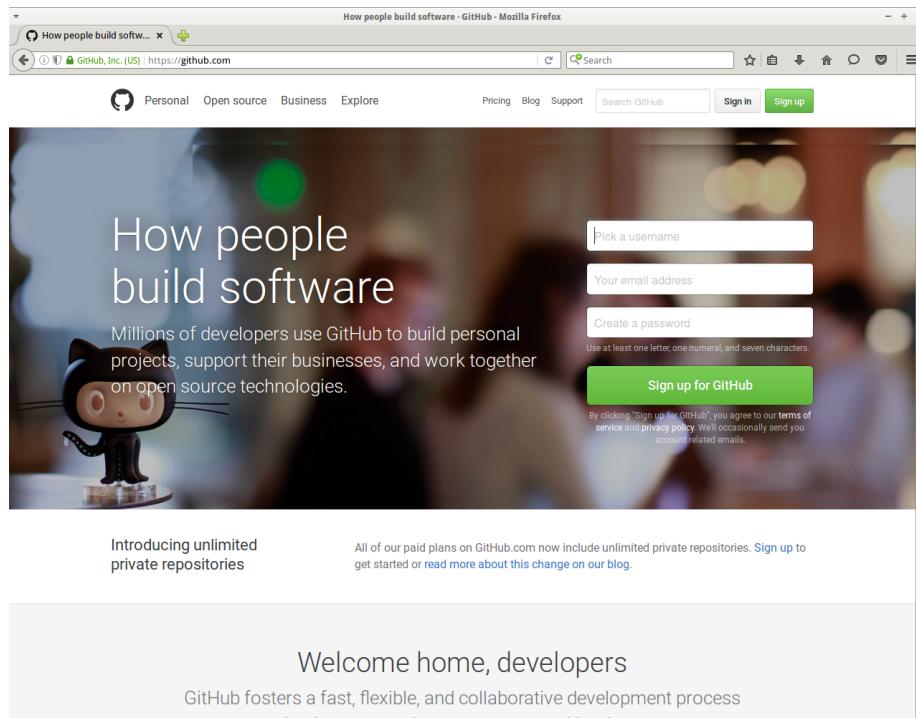


Figure 20: GitHub homepage

- Klik op **Sign in** (NL: Aanmelden)

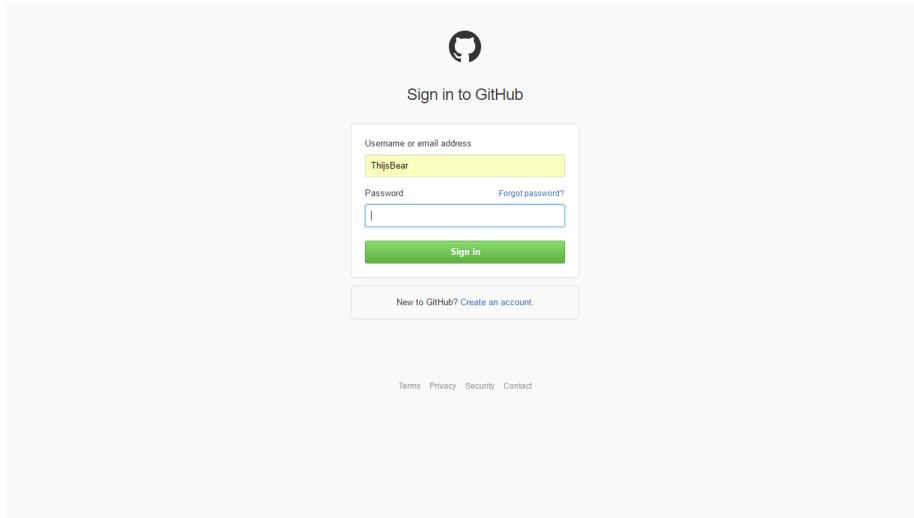


Figure 21: GitHub aanmelden

- Log in met je GitHub account (Heb je die niet? Doe dan eerst deze les!)

### GitHub Page repository maken

- Klik op het plusje bovenin
- Klik op **New repository**

Vul hier in:

- **Repository name:** de naam van je GitHub Page moet zijn `gebruiker.github.io`, dus bijvoorbeeld `thijsvb.github.io`
- **Description:** omschrijving van je GitHub Page, bijvoorbeeld `Mijn geweldige website!`
- Kies `Public`
- Vink aan: `Initialize this repository with a README`
- Kies bij `Add a license: GNU General Public License v3.0`

Je hebt nu een GitHub Page!

### HTML

Als je nu in een browser naar je GitHub Page gaat, is er nog niks. Je moet een homepage maken, en om dat te doen moet je HTML-bestanden kunnen schrijven.

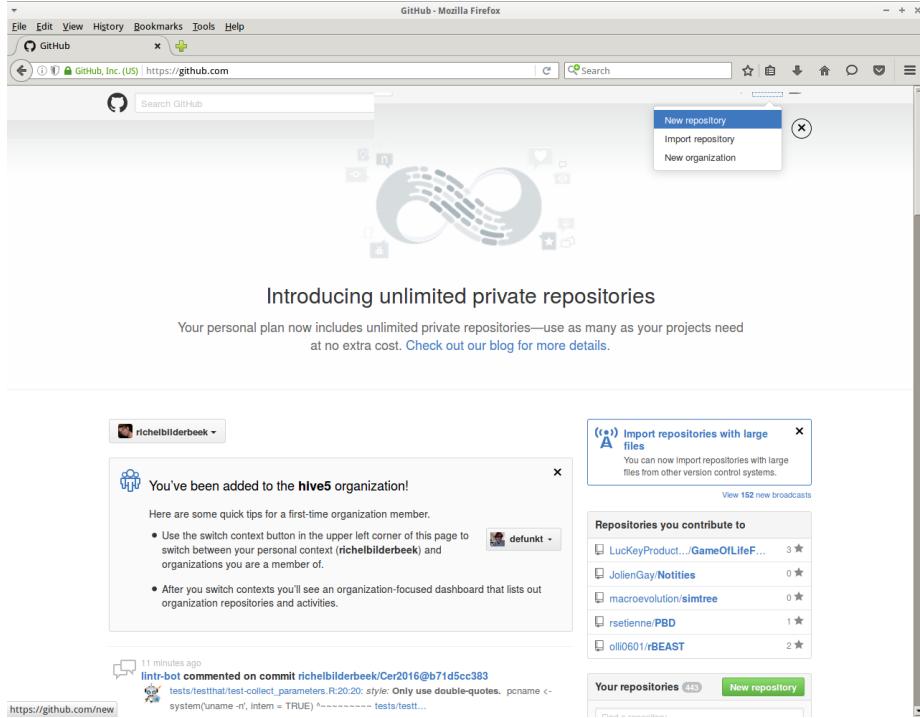


Figure 22: Maak een GitHub

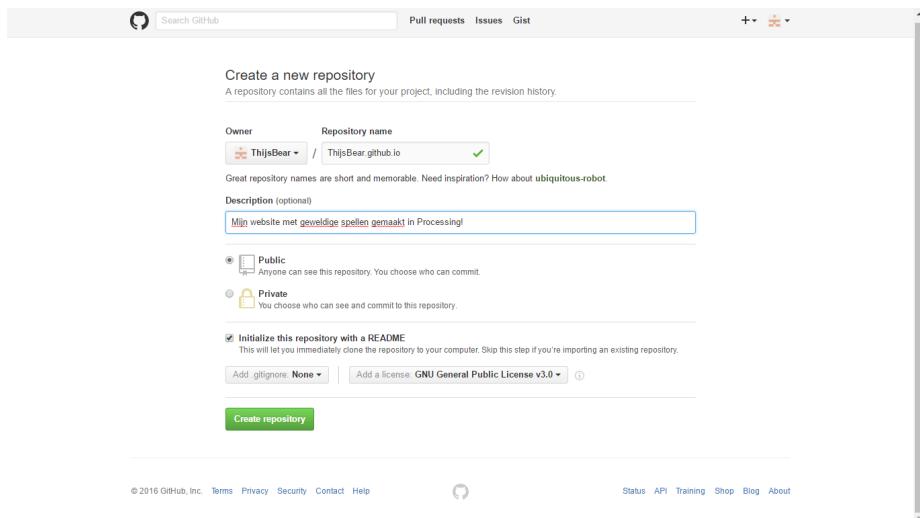
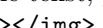


Figure 23: Maak GitHub Page repository

HTML staat voor HyperText Markup Language en is een opmaaktaal. Met een opmaaktaal kun je aangeven hoe tekst eruit moet komen te zien, een andere veel gebruikte opmaaktaal is Markdown (deze pagina is in Markdown geschreven).

Om tekstopmaak te doen in een programma zoals *Microsoft Word* heb je allerlei knoppen; dikgedrukte tekst, schuine tekst, plaatje, etc. In HTML heb je daar *tags* voor; **<b></b>**, *<i></i>*, .

Een stukje HTML met een tag gaat altijd op deze manier:

```
<tag>(n)iets</tag>
```

- **<tag>** opent de tag, in plaats van **tag** staat hier het soort tag wat je wil gebruiken (bijvoorbeeld **h1** of **p**)
- **(n)iets** kan een heleboel zijn, zoals teks, nog meer tags, of helemaal niks
- **</tag>** sluit de tag, hier moet in plaats van **tag** hetzelfde als in de openende tag staan

Een HTML-bestand begint altijd met **<!doctype html>**, dit vertelt een programma wat het bestand opent dat het in HTML geschreven is. De rest van het bestand staat in één **html** tag, die weer is opgedeeld in één **head** tag en één **body** tag.

In de **head** tag staat informatie *over* de webpagina, in de **body** tag staat wat er *op* de webpagina komt.

Een stukje HTML kan er bijvoorbeeld zo uit zien:

```
<body>
  <h1>Hello World</h1>
  <p>This is <b>my</b> website!</p>
</body>
```

## De homepage maken

Elke website is een map met bestanden, alleen staat de map niet op jouw computer. Een repository is ook een map, en de GitHub Page repository is ook jouw website. De homepage van elke website is het bestand **index.html**, die gaan we dus maken!

- Ga naar je GitHub Page repository
- Klik op **New file**
- Noem het nieuwe bestand **index.html**
- Begin het bestand met **<!doctype html>**
- Open op een nieuwe regel een **html** tag met **<html>**
- Open op een nieuwe regel een **head** tag met **<head>**
- Open op een nieuwe regel een **title** tag met **<title>**
- Typ hier de titel van je webpagina, bijvoorbeeld **Hello World**

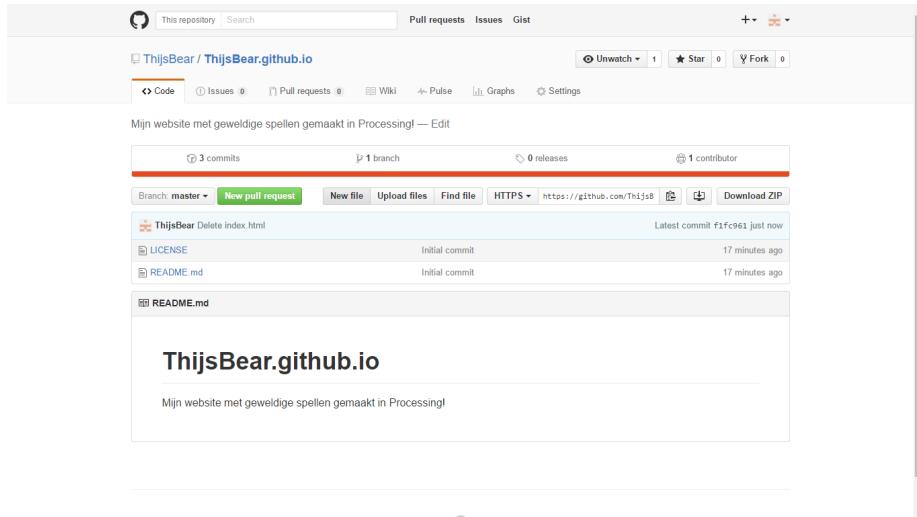


Figure 24: De GitHub Page repository

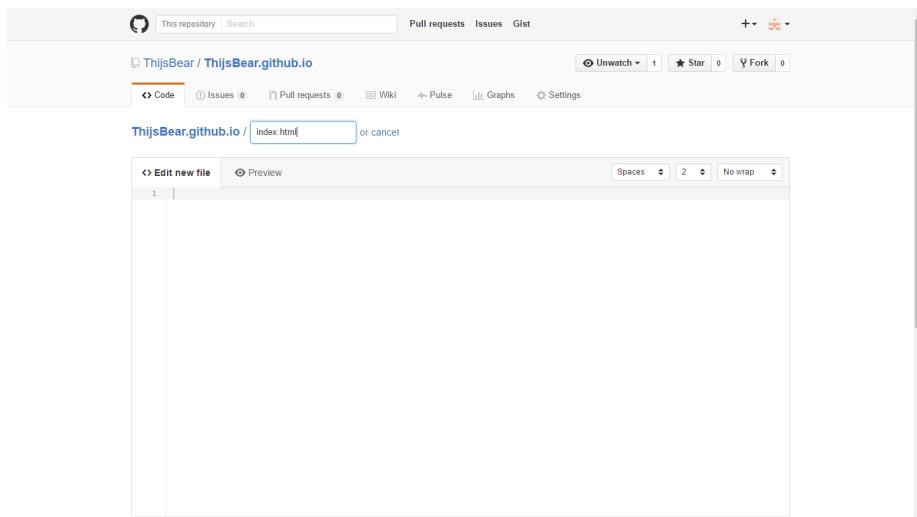
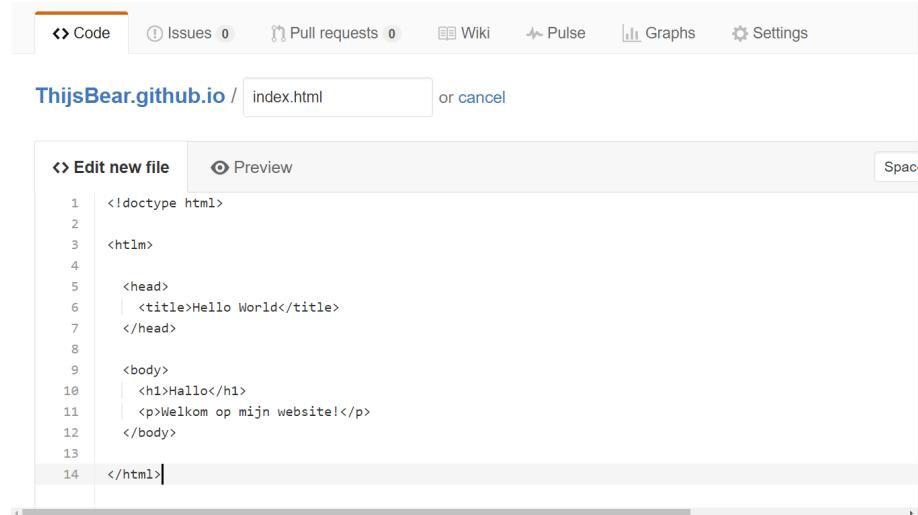


Figure 25: Maak index.html

- Sluit de **title** tag met `</title>`
- Sluit op een nieuwe regel de **head** tag met `</head>`
- Open op een nieuwe regel een **body** tag met `<body>`
- Open op een nieuwe regel een **h1** tag met `<h1>`
- Typ hier een kop voor je webpagina, bijvoorbeeld **Hallo**
- Sluit de **h1** tag met `</h1>`
- Open op een nieuwe regel een **p** tag met `<p>`
- Type hier een stukje tekst voor je webpagina, bijvoorbeeld **Welkom op mijn website**
- Sluit de **p** tag met `</p>`
- Sluit op een nieuwe regel de **body** tag met `</body>`
- Sluit op een nieuwe regel de **html** tag met `</html>`

Als het goed is ziet je bestand er nu zo uit:



The screenshot shows a GitHub repository interface. At the top, there are navigation links: Code, Issues 0, Pull requests 0, Wiki, Pulse, Graphs, and Settings. Below this, the repository name is shown as **ThijsBear.github.io**. A search bar contains the text "index.html". The main area displays the content of the index.html file:

```

<!DOCTYPE html>
<html>
<head>
| <title>Hello World</title>
</head>
<body>
| <h1>Hallo</h1>
| <p>Welkom op mijn website!</p>
</body>
</html>

```

Figure 26: index.html

- Klik onderaan op de groene **Commit new file** knop
- Ga in je browser naar je GitHub Page en bewonder je eigen website!
- Ga terug naar je GitHub Page repository
- Klik bij **index.html** op het padloodje om het bestand te bewerken
- Maak een paar veranderingen en klik op **Commit changes**
- Bewonder nu je eigen gepersonaliseerde website en bekijk ook eens die van je klasgenoten!

**Als je nog geen GitHub account hebt, doe dan eerst deze les**

**Als je nog geen GitHub Page hebt, doe dan eerst deze les**

## Processing.js

Processing.js is een JavaScript wat een Processing sketch op een webpagina kan runnen. Deze les gaan we een Processing sketch op onze GitHub Page zetten.

Met Processing.js kunnen mensen jou spel spelen, kunstwerk bewonderen of programma runnen op jouw GitHub Page. De code voor het spel, kunstwerk of programma staat op een GitHub repository.

### Processing.js downloaden

Processing.js is een JavaScript bestand. Om het te gebruiken moeten we het eerst downloaden. \* Ga in een webbrowser naar [www.processingjs.org](http://www.processingjs.org)

Je komt nu op de homepage

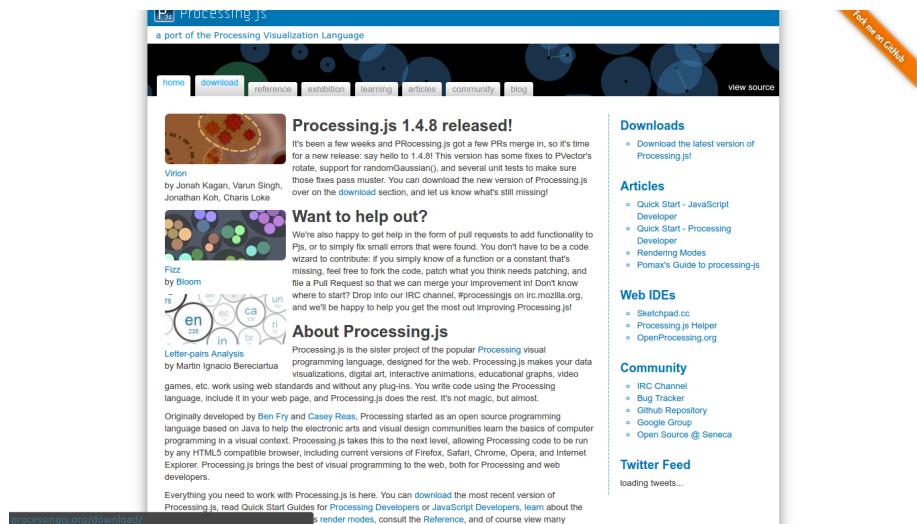


Figure 27: ProcessingJS homepage

- Klik bovenin op **download**

Je komt nu op de download pagina

- Klik met de rechter muisknop op **processing.js**
- Klik op **Link opslaan als...** (in het Engels: **Save link as...**)
- Kies een plek op je computer waar je het bestand zo terug kan vinden (Tip: Bureaublad of Desktop is vaak makkelijk te vinden!)
- Ga naar je GitHub Page repository
- Klik op **Upload file**

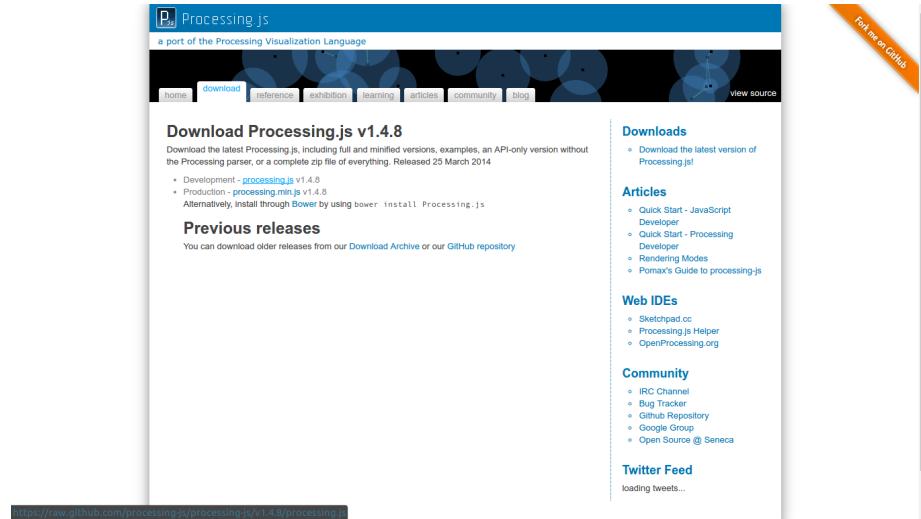


Figure 28: ProcessingJS download

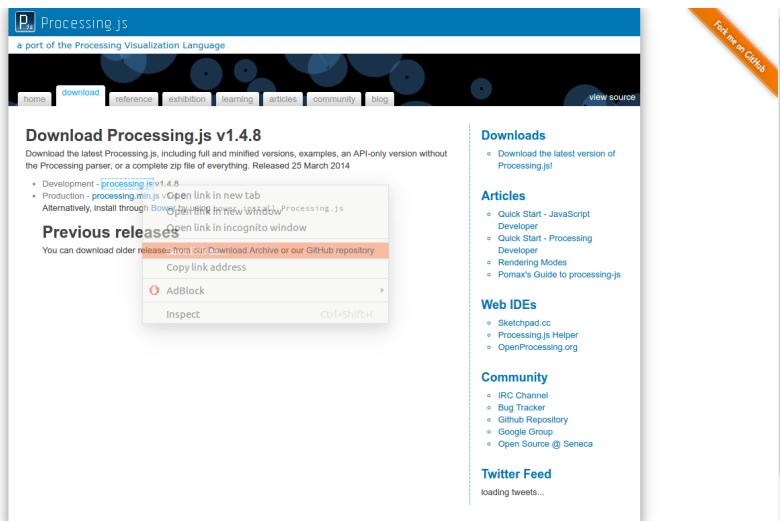


Figure 29: Link opslaan als

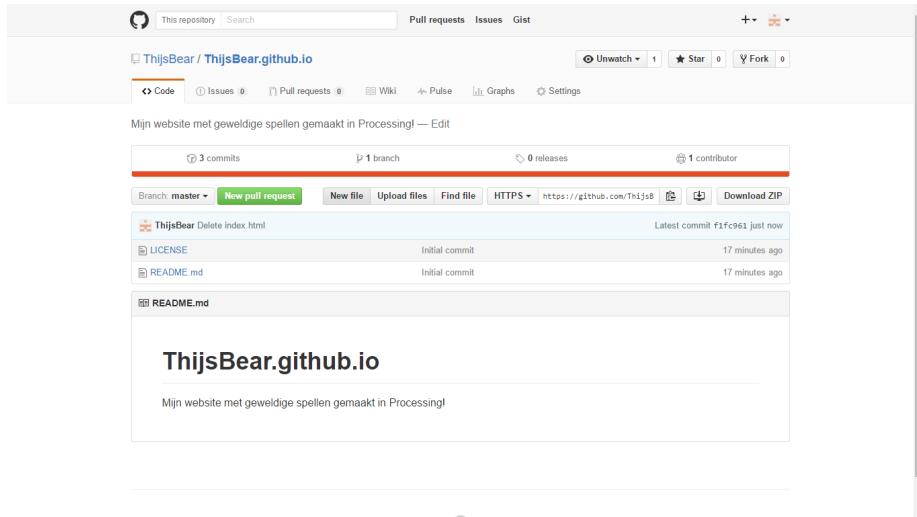


Figure 30: GitHub Page repository

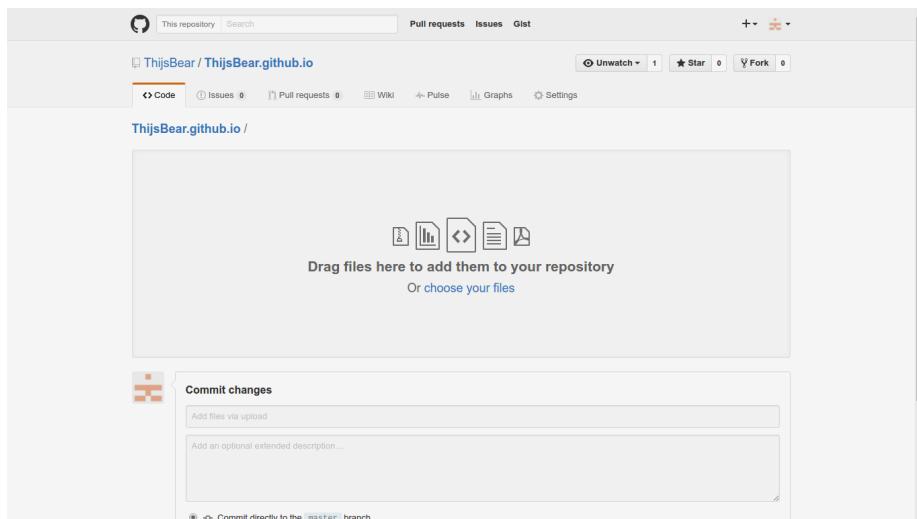


Figure 31: GitHub upload file

- Sleep `processing.js` dit scherm in of klik op `select your files` en dubbelklik op `processing.js`
- Klik onderaan op de groene knop `Commit changes`

## Processing sketch uploaden

- Zoek een Processing sketch op je computer om op je website te zetten (Tip: kies een simpele sketch)
- Als je in de sketch map kijkt staat daar een `.pde` bestand, deze hebben we straks nodig
- Ga naar je GitHub Page repository

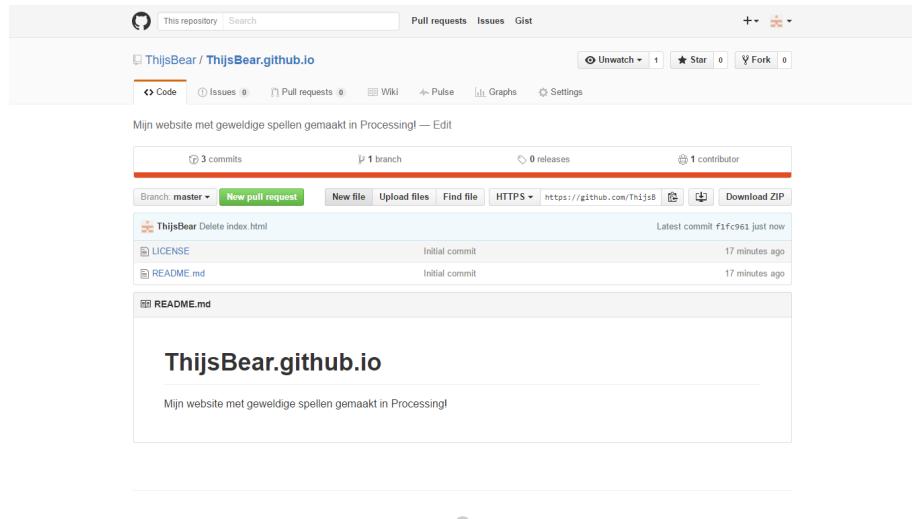


Figure 32: GitHub Page repository

- Klik op `Upload file`
- Sleep het `.pde` bestand dit scherm in of klik op `select your files` en dubbelklik op het `.pde` bestand
- Klik onderaan op de groene knop `Commit changes`

Nu ga je de sketch op je homepage zetten

- Ga naar je GitHub Page repository
- Klik op `index.html`
- Klik op het plusje rechts boven
- Open een nieuwe regel in head (dus tussen `<head>` en `</head>`)
- Typ hier deze regel over: `<script src="processing.js"></script>`
- Open een nieuwe regel in body (dus tussen `<body>` en `</body>`)
- Typ hier deze regel over: `<canvas data-processing-sources="voorbeeld.pde"></canvas>`

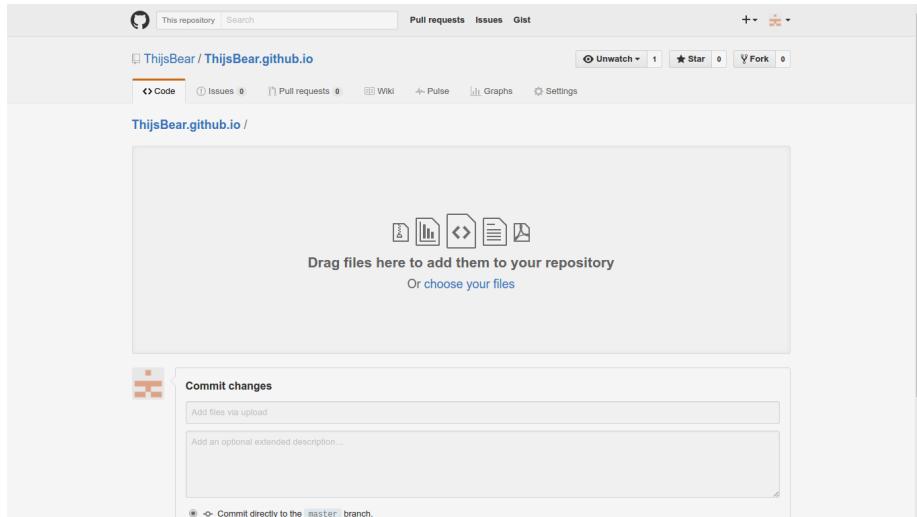


Figure 33: GitHub upload file

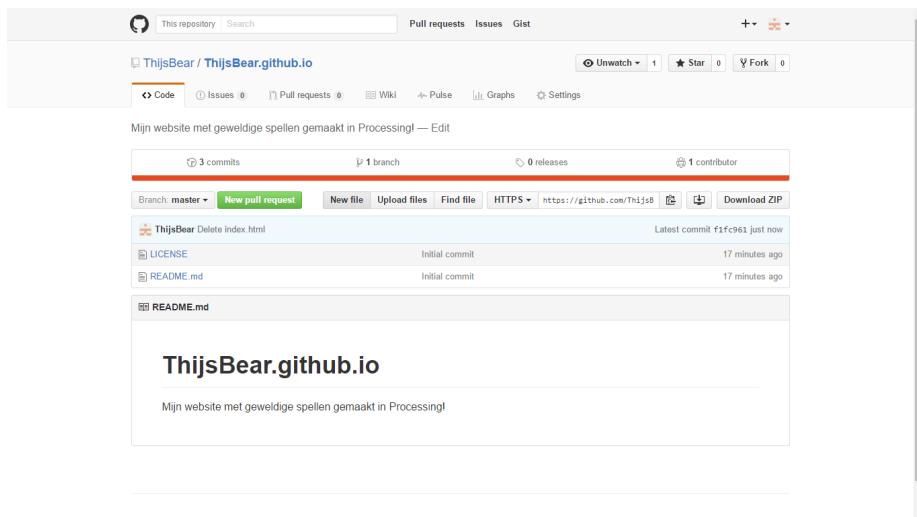


Figure 34: GitHub Page repository

- Vervang nu **voorbeeld** voor de naam van je .pde bestand

Het bestand moet er nu ongeveer zo uit zien:

```

1 <!doctype html>
2
3 <html>
4
5 <head>
6   <title>Hello World</title>
7   <script src="processing.js"></script>
8 </head>
9
10 <body>
11   <h1>Hallo</h1>
12   <p>Welkom op mijn website!</p>
13   <canvas data-processing-sources="voorbeeld.pde"></canvas>
14 </body>
15
16 </html>
17

```

Figure 35: index

- Klik onderaan op de groene knop **Commit changes**
- Ga naar je website en bewonder je werk!

## GitHub chat

In deze les gaan we de chatroom van de cursus in.

### Wat is een chatroom?

Er zijn meer soorten chatrooms. In deze les gebruiken we een IRC chatroom. IRC bestaat sinds 1986 en wordt nog veel gebruikt. Elke hackerspace heeft wel een eigen IRC chatroom (dit noemen we ook een ‘IRC kanaal’). Een IRC chatroom is net een echt gesprek met een groepje mensen: je weet niet wat er is gezegd voordat je met ze sprak, en je weet niet wat ze zeggen als je weer weggaat. Er zijn wel IRC chatrooms die wel de gesprekken bewaren.

### Gedragsregels

Een paar van <https://wiki.debian.org/IRC/Netiquette>:

- Wees behulpzaam
- Praat over het thema, in ons geval programmeren
- Probeer mee te doen met het gesprek, in plaats van nieuwe gesprekken te starten
- Stel jezelf voor als je mee doet in een gesprek en nieuw bent
- ‘Don’t feed the trolls’. Help de mensen die de sfeer verpesten niet hierbij: blijf altijd vriendelijk
- Heb plezier

## Website

Ga naar de GitHub van de cursus, <https://github.com/richelbilderbeek/Dojo>. Klik op irc: #dojoGroningen

### IRC inloggen

- Vul bij **Nickname** je bijnaam in
- Klik op **I am not a robot**

Some krijg je een vraag, zoals hier:

Beantwoord de vraag.

Klik dan op **Connect**

### IRC server login

Eerst komt er een scherm met veel tekst:

Wacht tot dit scherm voorbij is

### IRC chatroom

Je komt nu in de chatroom:

Rechts kun je een lijst zien van iedereen die in deze kamer aanwezig is.

Onderin kun je je tekst typen:

De tekst verschijnt bij iedereen meteen op hun scherm.

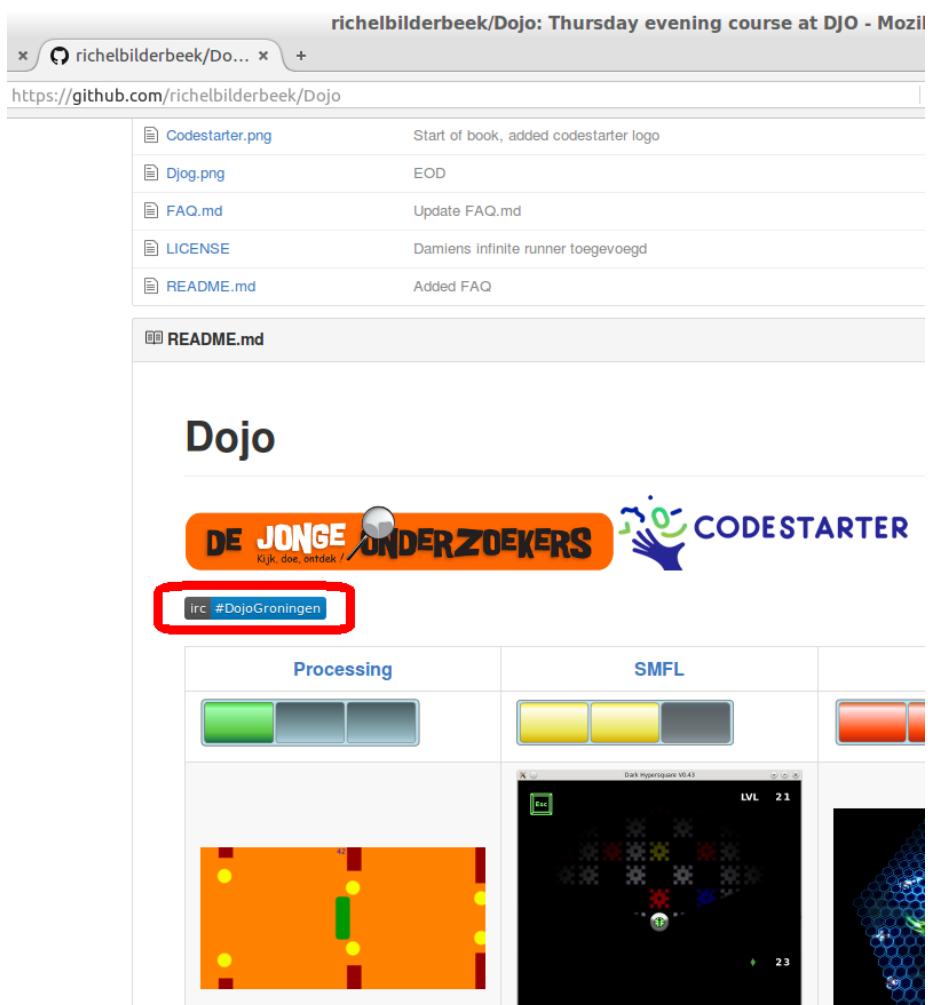


Figure 36: GitHub.png

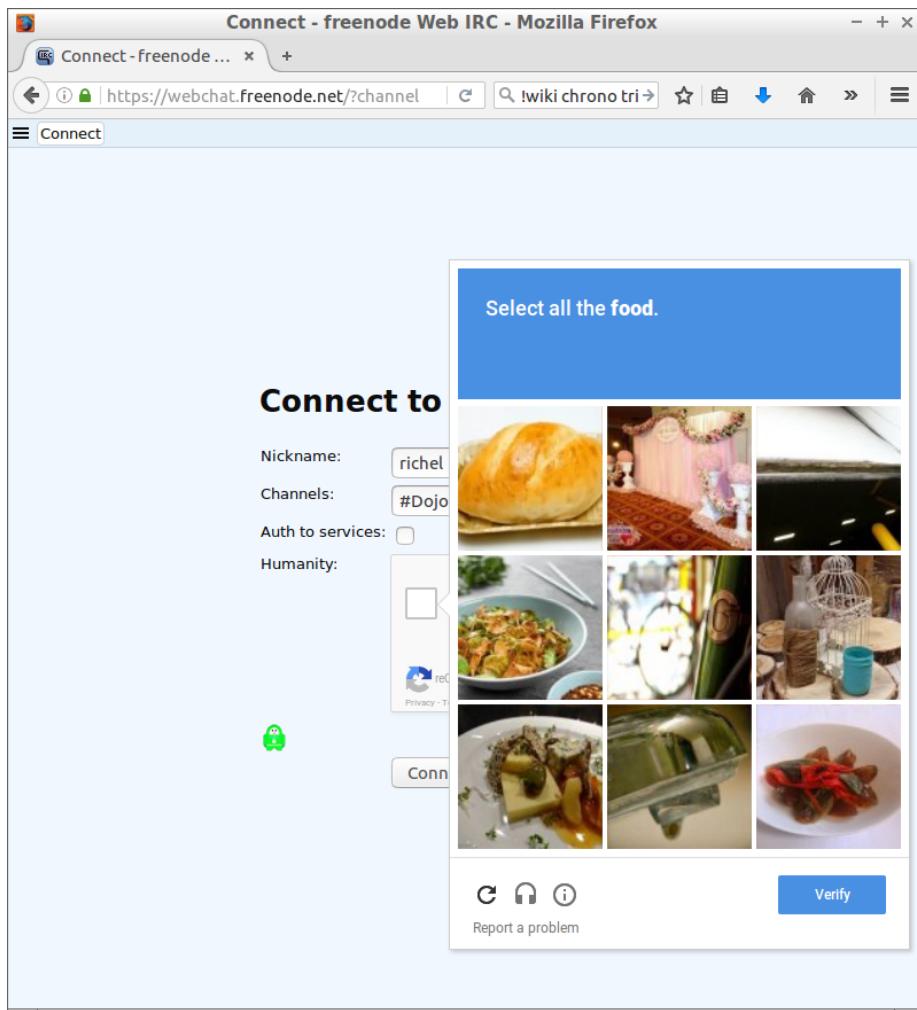


Figure 37: IrcInlog.png

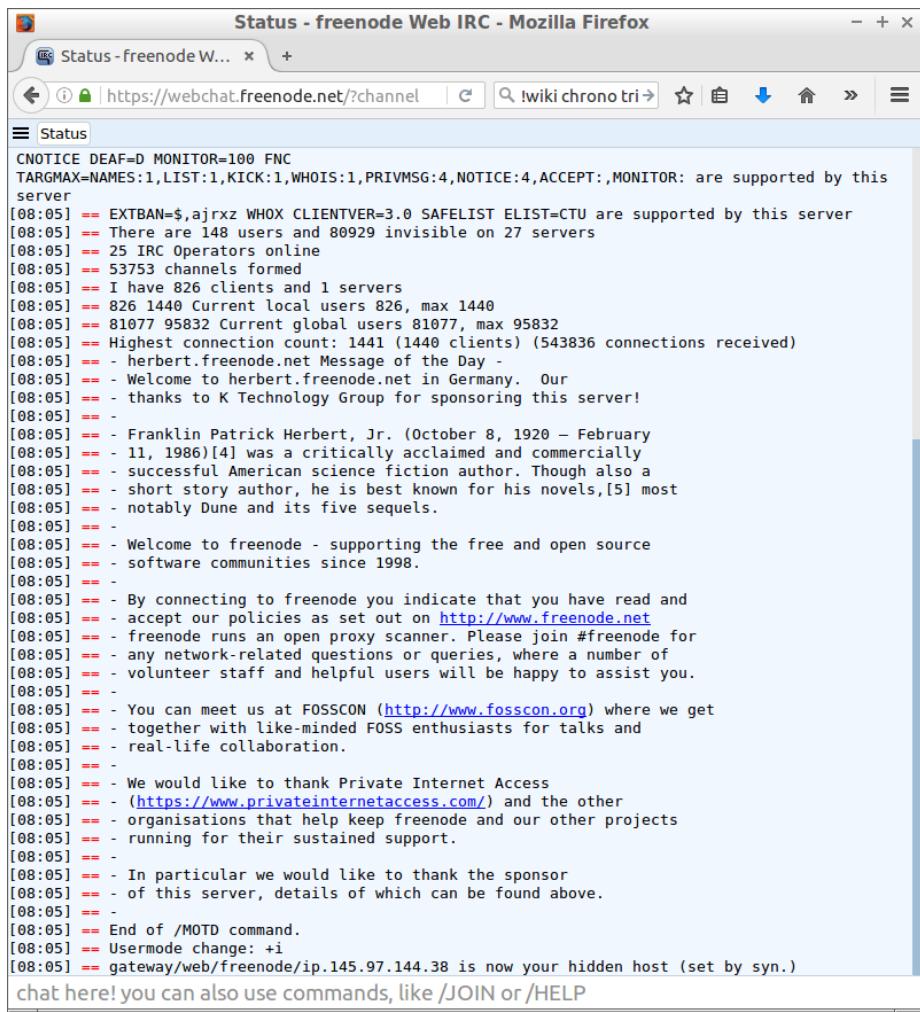


Figure 38: IrcServerMessages.png

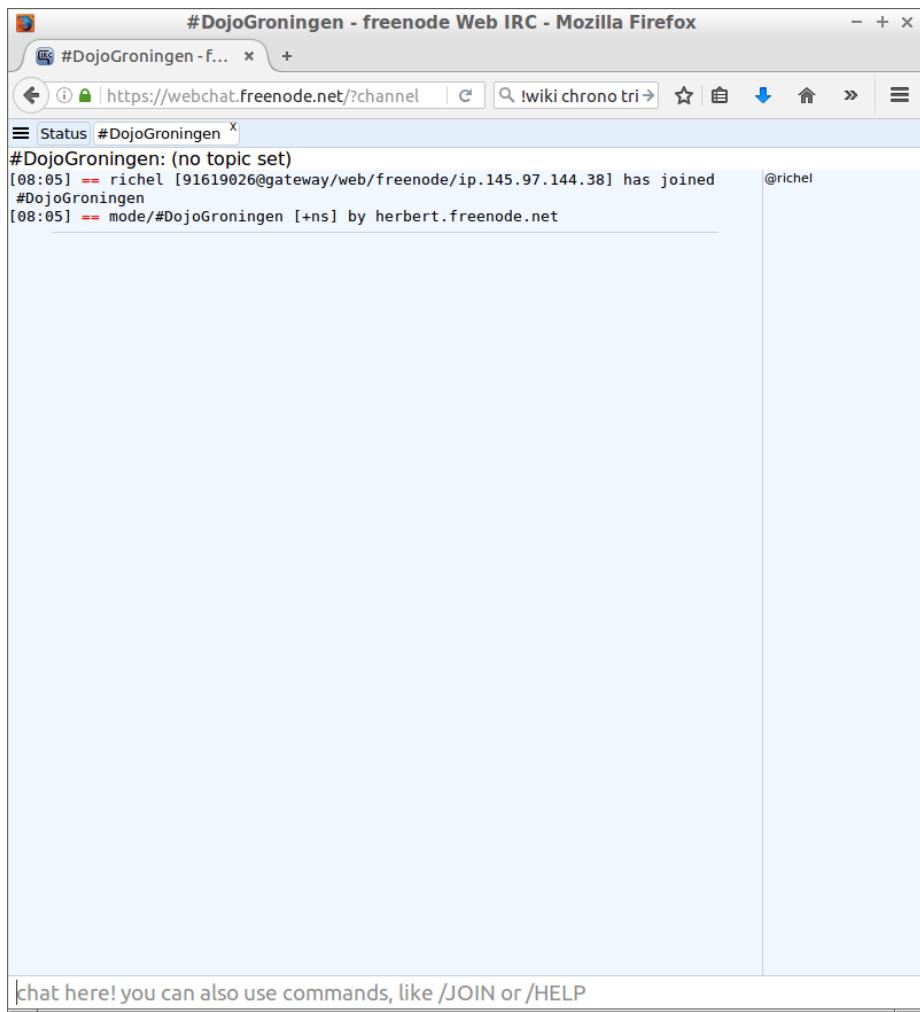


Figure 39: IrcChatroom1.png

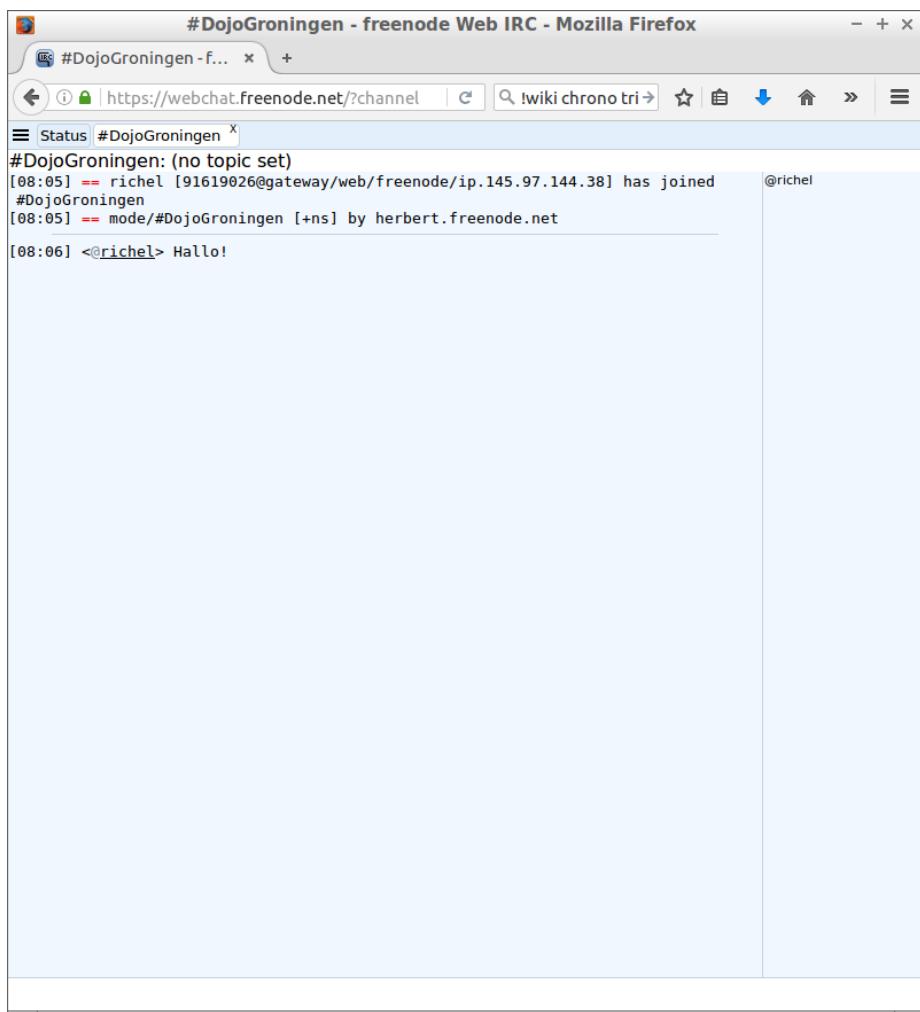


Figure 40: IrcChatroom2.png

## **git, GitHub samenwerken**

Samenwerken is lastig, zeker met programmeren: de ander (of jij) maakt misschien dingen stuk en/of kwijt.

In deze les gaan we leren samenwerken met git en GitHub. Dit doen we (nog) niet met code, maar met een Nederlandse tekst.

### **git, GitHub**

git houdt de geschiedenis van de tekst/code bij. Je kunt ook terug in de geschiedenis gaan. Als je per ongeluk een bestand hebt gewist, kun je deze weer terughalen.

### **Sprookje**

In deze les gaan we samen een sprookje maken. Hier staat de begintekst:

```
# Sprookje

## 1

Heel erg lang geleden ...

## 2

## 3

## 4

## 5

## 6

## 7

... en ze leefden nog lang en gelukkig.
```

Het sprookje is nog erg saai. Samen kunnen we er een leuk verhaal van maken.

### **De GitHub op je computer zetten**

Om het sprookje op je computer te zetten, start je (onder Windows) Git Bash, onder Linux een terminal.

Type dan het volgende:

```
git clone https://github.com/richelbilderbeek/Sprookje
```

Er is nu een map gemaakt, genaamd **Sprookje** met daarin het bestand **README.md**. **README.md** bevat de tekst van het sprookje en kun je openen in Kladblok/Wordpad/leafpad of een andere teksteditor.

## Samenwerken

Verander de tekst in **README.md** en save.

Doe dan in Git Bash:

```
git pull
```

```
git add --all :/
```

```
git commit -m "Verbetering"
```

```
git push
```

- **git pull**: update de GitHub
- **git add --all :/**: bekijk alles wat veranderd is
- **git commit -m "Verbetering"**: benoem de veranderingen **Verbetering**
- **git push**: publiceer je veranderingen

## Merge conflicts

Soms krijg je ‘Merge conflicts’. Dit gebeurt als **git** niet weet, waar welke veranderingen horen.

Stel, je hebt dit:

```
## 1
```

```
## 2
```

Nu maakt Jantje er dit van:

```
## 1
```

```
Er woonde een gevvaarlijke draak in de berg ten oosten van het dorpje.
```

```
## 2
```

En maakt Truus er tegelijkertijd dit van:

```
## 1
```

```
De prinses was op zoek naar de knapste ridder.
```

```
## 2
```

`git` kan dit niet samenvoegen, omdat de volgorde tussen de zinnen onduidelijk is.

Degene die het merge conflict krijgt, krijgt een `README.md` zoals dit:

```
## 1
<<<<<<<<<HEAD: 372546042986273465283736726872
Er woonde een gevaarlijke draak in de berg ten oosten van het dorpje.
=====
De prinses was op zoek naar de knapste ridder.
TAIL>>>>>>>>: 623483754623592736429736239874
```

## 2

De persoon zal dan zelf de zinnen moeten samenvoegen:

## 1

```
Er woonde een gevaarlijke draak in de berg ten oosten van het dorpje.
Ten westen van het dorpje was een kasteel met een prinses.
De prinses was op zoek naar de knapste ridder.
```

## 2

Dan weer:

```
git pull
git add --all :/
git commit -m "Verbetering"
git push
```

## Voorwoord

Dit is het Processing boek van de Dojo.

Processing is een programmeertaal.

Dit boek leert je die programmeertaal.

## Over dit boek

Dit boek heeft een CC-BY-NC-SA licentie.

(C) Dojo Groningen 2016

The screenshot shows the Processing 3.2.3 IDE interface with the title bar "Voorwoord | Processing 3.2.3". The menu bar includes File, Edit, Sketch, Debug, Tools, and Help. The toolbar has a play/pause button and a Java dropdown. The code editor window displays the following text:

```
Voorwoord
1 *****
2      PROCESSING
3 *****
4
5 Dit is het Processing boek
6 van de Dojo.
7
8 Processing is een programmeertaal.
9
10 Dit boek leert je die taal.
11
12 Over dit boek
13 -----
14
15 Dit boek heeft een CC-BY-NC-SA
16 licentie.
17
18 (C) Dojo Groningen 2016 CC-BY-NC-SA
19 */
20
```

At the bottom of the code editor, there are two logos: "DE JONGE ONDERZOEKERS" with the tagline "Kijk, doe, ontdek!" and "CODESTARTER". The status bar at the bottom shows "Done saving." and tabs for "Console" and "Errors".

Figure 41: Voorblad



Figure 42: Het logo van De Jonge Onderzoekers



Figure 43: Het logo van Codestarter



Figure 44: De licentie van dit boek