



ELECTRON

Een introductie door [Kilian Valkhof](#)

Kilian Valkhof

User Experience Developer

Lid van Electron Maintainers team

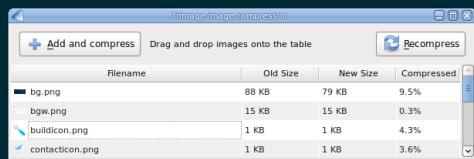


Partner bij **Fluxility**

Full service internetbureau uit Zoetermeer: apps, sites en maatwerk. 15 man, begonnen in 2005

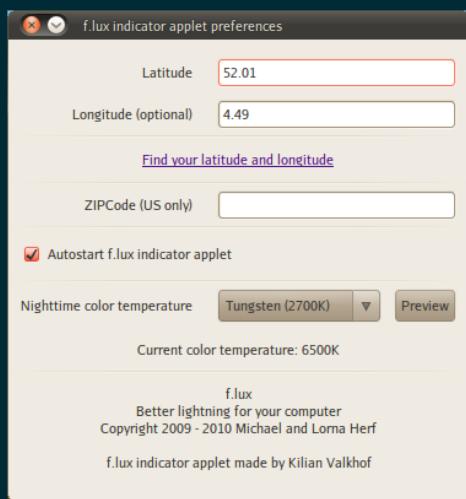
Trimage

(in Qt)



f.lux linux gui

(in GTK)



FromScratch

(in Electron!)

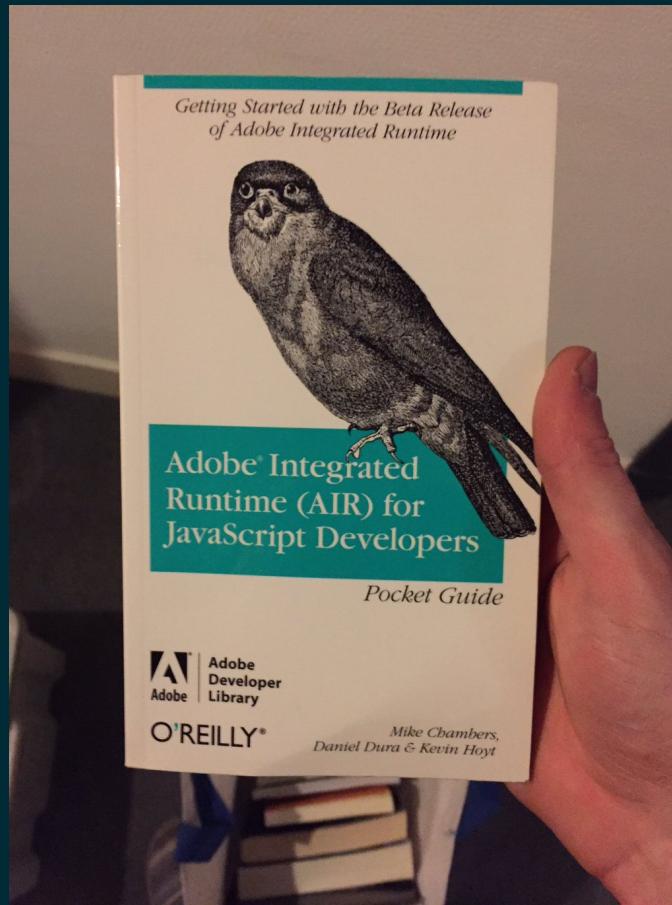


WAT IS



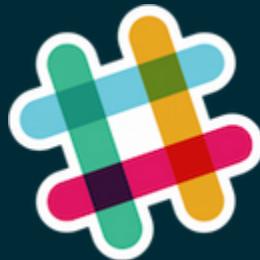
ELECTRON

Een *Library* om desktop applicaties te maken
met *Javascript, HTML en CSS*, op *Windows,*
Mac en Linux



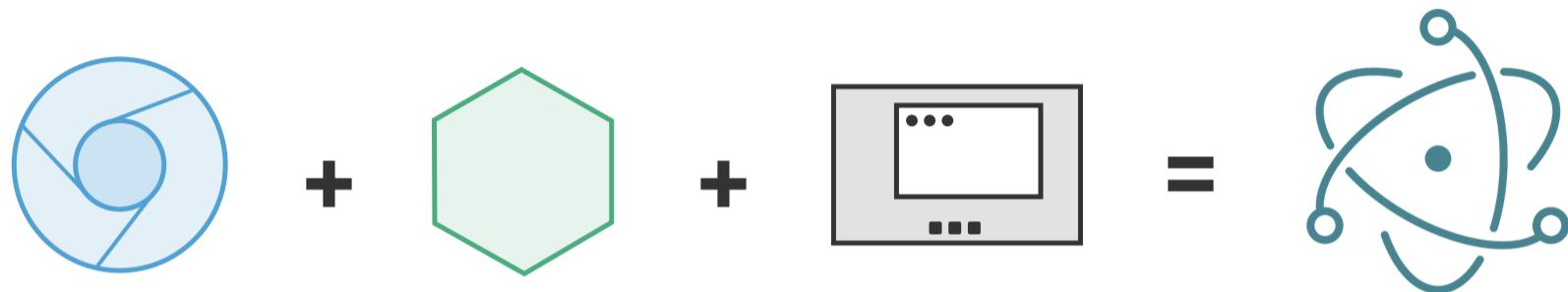
...maar dan modern

Wie gebruikt Electron?



en honderden meer: electron.atom.io/apps/

Hoe werkt Electron



Chromium
for making
web pages

Node.js
for filesystems
and networks

Native APIs
for three
systems

ELECTRON

Electron applicaties draaien twee processen:
Main & Renderer

Main

Node.js: App en Windowbeheer, integratie met OS en
Filesystem

Renderer

Chromium: één per window, DOM die je kent + wat extra's

IPC

Interprocess communication

IPC Main & IPC Renderer



**Pass messages
between the processes.**

Voordelen

Alles in één taal: Javascript

Renderen van een pagina, én native API's

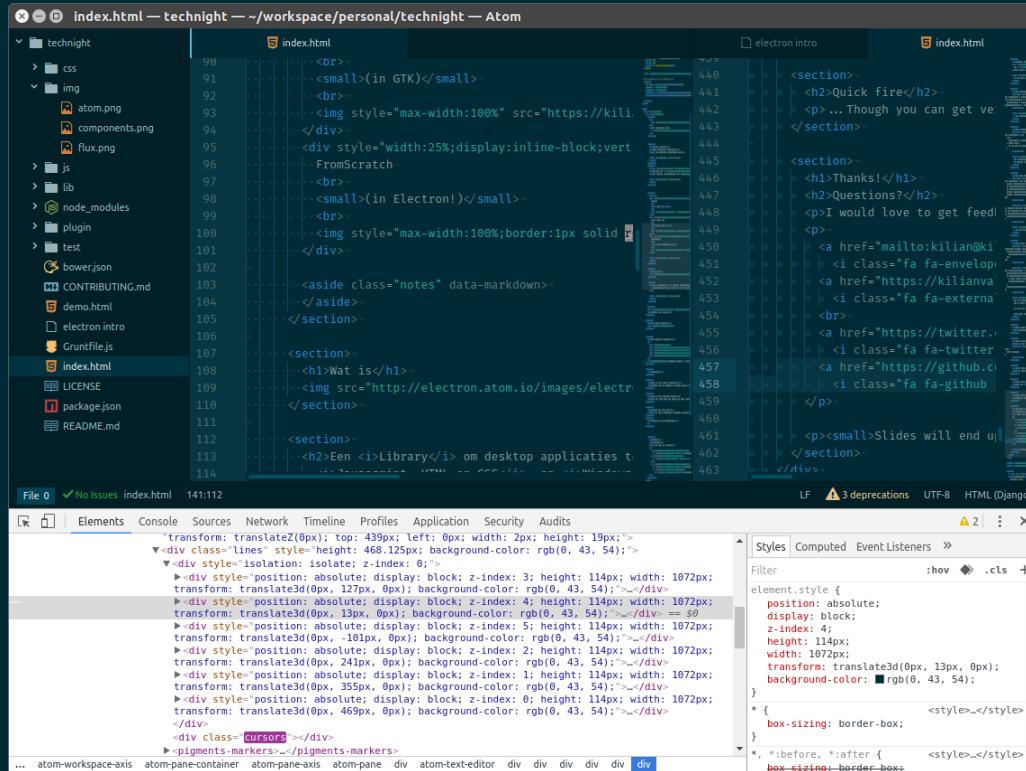
Alles in één browser: Chrome

Werkt het voor jou, dan werkt het voor iedereen

Toegang tot heel NPM

Meer dan 416K+ packages, inclusief leftpad!

Devtools in je applicaties!



Kies je eigen development flow

Ga los met: Webpack, PostCSS, ES6, Babel, Typescript, Elm of
what je wilt!

IN CODE

Package.json

Net als andere Node apps

Main.js

Main Process: app en window beheer

Index.html

Renderer process: Je UI

package.json

```
{  
  "name": "electron-voorbeeld",  
  "version": "1.0.0",  
  "main": "main.js",  
  "devDependencies": {  
    "electron": "^1.6.0"  
  }  
}
```

main.js

```
const electron = require('electron');
const app = electron.app;
const BrowserWindow = electron.BrowserWindow;

let mainWindow;

app.on('ready', () => {
  mainWindow = new BrowserWindow({ width: 800, height: 600 });
  mainWindow.loadURL('file://' + __dirname + '/index.html');
}) ;

app.on('window-all-closed', () => {
  app.quit();
}) ;
```

index.html

```
<!doctype html>
<html>
  <head>
    <meta charset="UTF-8">
    <title>Hallo Electron!</title>
  </head>
  <body>
    <h1>Hallo Electron!</h1>
    <p>
      We gebruiken Electron
      <script>document.write(process.versions.electron) </script>
    </p>
  </body>
</html>
```

package.json

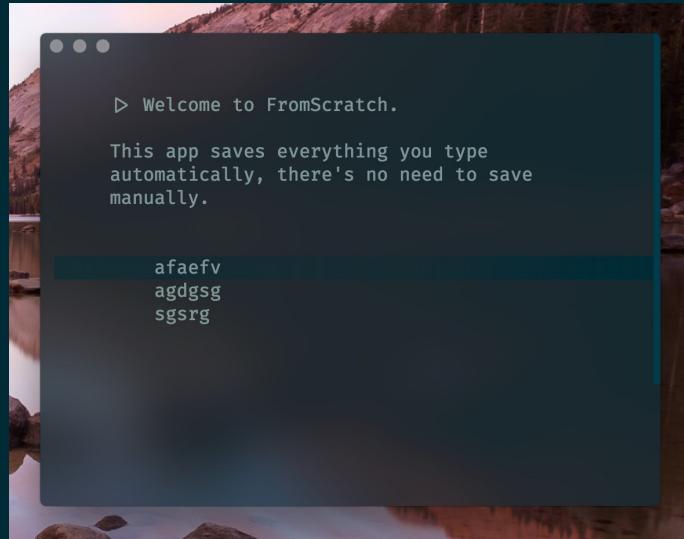
```
{  
  "name": "electron-voorbeeld",  
  "version": "1.0.0",  
  "main": "main.js",  
  "devDependencies": {  
    "electron": "^1.6.0"  
  },  
  "scripts": {  
    "start": "electron ."  
  }  
}
```

```
$ npm start
```



Ook mogelijk:

Achtergrondkleur, Frameless windows, transparency, hidden windows, macOS Vibrancy



Extra API's:

- Native menu's
- Native dialogs
- Native notifications
- autoUpdater

Niet alleen webpagina's

- System Tray applications
- Toegang tot mic + camera, clipboard en global shortcuts

UI in HTML

Doe wat je wilt! Custom UI, Bootstrap, Material UI

Specifiek voor OSX: Photonkit

The screenshot shows the Photonkit interface on a Mac OS X desktop. The window title is "Photon". The left sidebar contains a "Favorites" section with links to "connors", "Photon" (which is selected), "Downloads", "Documents", "Applications", "AirDrop", and "Desktop". Below that is a "Tags" section with color-coded circles for "Red", "Orange", "Green", and "Blue". At the bottom is a "Devices" section. The main content area is a table listing 15 SCSS files:

Name	Kind	Date Modified	Author
bars.scss	Document	Oct 13, 2015	connors
base.scss	Document	Oct 13, 2015	connors
button-groups.scss	Document	Oct 13, 2015	connors
buttons.scss	Document	Oct 13, 2015	connors
docs.scss	Document	Oct 13, 2015	connors
forms.scss	Document	Oct 13, 2015	connors
grid.scss	Document	Oct 13, 2015	connors
icons.scss	Document	Oct 13, 2015	connors
images.scss	Document	Oct 13, 2015	connors
lists.scss	Document	Oct 13, 2015	connors
mixins.scss	Document	Oct 13, 2015	connors
navs.scss	Document	Oct 13, 2015	connors
normalize.scss	Document	Oct 13, 2015	connors
photon.scss	Document	Oct 13, 2015	connors
tables.scss	Document	Oct 13, 2015	connors
tabs.scss	Document	Oct 13, 2015	connors

Packaging

Cross platform executables voor Mac, Windows en Linux.

electron-packager

Of met installers: electron-builder

Ook voor de Windows en Mac app stores!

Meer leren

electron/electron-api-demos

The screenshot shows a dark-themed website for 'ELECTRON API DEMOS'. On the left, a sidebar lists categories: 'WINDOWS' (selected), 'MENUS', 'NATIVE USER INTERFACE', 'COMMUNICATION', and 'SYSTEM'. Under 'WINDOWS', 'Create and manage windows' is selected, highlighted with a green bar. The main content area is titled 'Create and Manage Windows' and describes the `BrowserWindow` module. It includes a link to the full API documentation. Below this, three sub-sections are listed: 'Create a new window', 'Manage window state', and 'Create a frameless window', each with its supported platforms and process information. A large circular icon on the right features a stylized atom or electron model with orbits in blue, purple, and green.

ELECTRON API DEMOS

WINDOWS

Create and manage windows

Handling window [crashes and hangs](#)

MENUS

Customize [menus](#)

Register keyboard [shortcuts](#)

NATIVE USER INTERFACE

Open [external links](#) or system [file manager](#)

Use system [dialogs](#)

Put your app in the [tray](#)

COMMUNICATION

Communicate between the [two processes](#)

SYSTEM

Get app or user [system information](#)

Create and Manage Windows

The `BrowserWindow` module in Electron allows you to create a new browser window or manage an existing one.

Each browser window is a separate process, known as the renderer process. This process, like the main process that controls the life cycle of the app, has full access to the Node.js APIs.

Open the [full API documentation](#) in your browser.

Create a new window

SUPPORTS: WIN, OS X, LINUX | PROCESS: MAIN

Manage window state

SUPPORTS: WIN, OS X, LINUX | PROCESS: MAIN

Create a frameless window

SUPPORTS: WIN, OS X, LINUX | PROCESS: MAIN

DANK!

Vragen?

Ik ben hier te vinden:

 kilian@kilianvalkhof.com,  <https://kilianvalkhof.com>
 [@kilianvalkhof](https://twitter.com/kilianvalkhof),  [/kilian](https://github.com/kilian)