

Electron

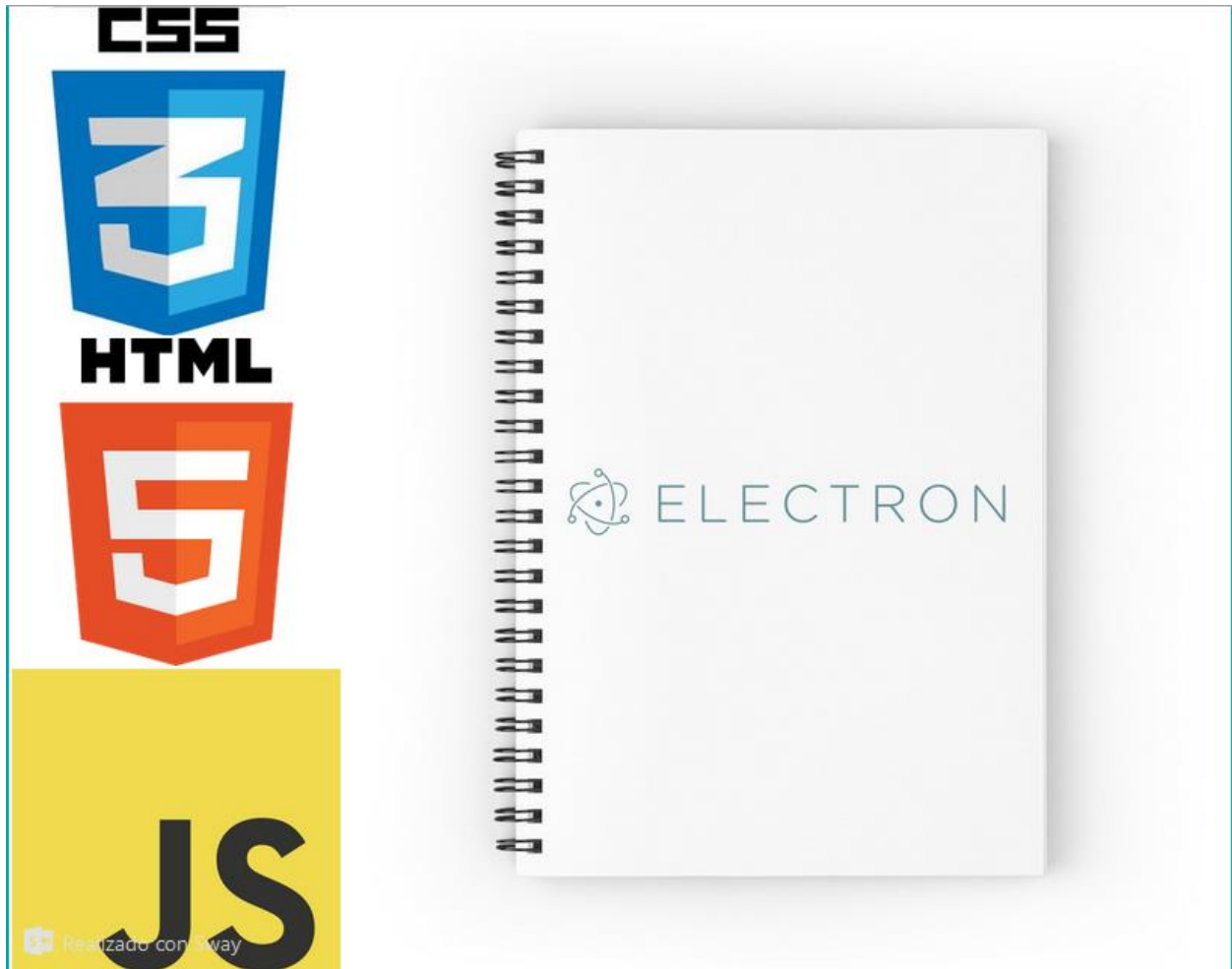




ELECTRON

Realizado con Sway

***Electron* es una herramienta que sirve para la construcción de aplicaciones de escritorio multiplataforma, soporta Html, Css y Javascript.**

Sway



Prerequisitos:	 Visual Studio <small>No es indispensable.</small>	 <small>Editor de Texto que funciona</small>
	 python <small>Lenguaje de Programación interpretado.</small>	 <small>software de control de versiones</small>
	 node JS <small>Entorno en tiempo de ejecución multiplataforma de código abierto para JavaScript</small>	 EL <small>Herramienta para desarrollo de aplicaciones de escritorio</small>

h Sway

Instalacion en Windows

<http://electron.atom.io/docs/v0.30.0/development/build-instructions-windows/>

- **Windows 7 o posterior**

Instalacion en Linux

<http://electron.atom.io/docs/v0.30.0/development/build-instructions-linux/>

- **Python 2.7 x**

Instalacion en Mac

<http://electron.atom.io/docs/v0.30.0/development/build-instructions-mac/>

- **OS X >= 10.8**

Mi primera Aplicacion

Se crea una carpeta con el nombre del proyecto

npm init

npm install electron-prebuilt --save-dev

con esto se crea el archivo package.js, donde se tiene la informacion del proyecto

Creamos el archivo main.js

Luego creamos el archivo index.html

<http://electron.atom.io/docs/latest/tutorial/quick-start/>

package.json - C:\Users\OSCAR\Desktop\hola_mundo\package.json - Atom

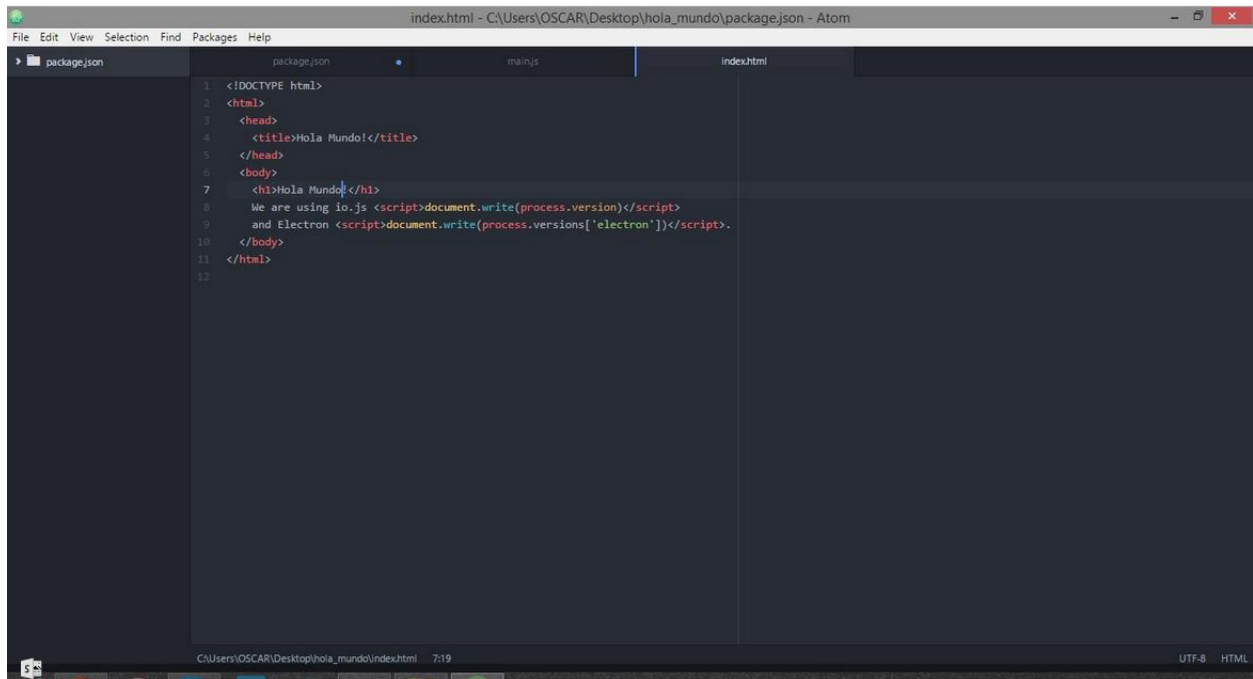
```
1 {
2   "name": "hola_mundo",
3   "version": "1.0.0",
4   "description": "",
5   "main": "main.js",
6   "scripts": {
7     "test": "echo \\Error: no test specified\\" && exit 1"
8   },
9   "author": "",
10  "license": "ISC"
11 }
12
```

516 UTF-8 JSON

main.js - C:\Users\OSCAR\Desktop\hola_mundo\package.json - Atom

```
1 var app = require('app'); // Module to control application life.
2 var BrowserWindow = require('browser-window'); // Module to create native browser window.
3
4 // Report crashes to our server.
5 require('crash-reporter').start();
6
7 // Keep a global reference of the window object, if you don't, the window will
8 // be closed automatically when the Javascript object is GCed.
9 var mainWindow = null;
10
11 // Quit when all windows are closed.
12 app.on('window-all-closed', function() {
13   // On OS X it is common for applications and their menu bar
14   // to stay active until the user quits explicitly with Cmd + Q
15   if (process.platform !== 'darwin') {
16     app.quit();
17   }
18 });
19
20 // This method will be called when Electron has finished
21 // initialization and is ready to create browser windows.
22 app.on('ready', function() {
23   // Create the browser window.
24   mainWindow = new BrowserWindow({width: 800, height: 600});
25
26   // and load the index.html of the app.
27   mainWindow.loadURL('file://' + __dirname + '/index.html');
28
29   // Open the devtools.
30   mainWindow.openDevTools();
31
32   // emitted when the window is closed.
33   mainWindow.on('closed', function() {
34     // Dereference the window object, usually you would store windows
35     // in an array if your app supports multi windows, this is the time
36     // when you should delete the corresponding element.
37     mainWindow = null;
38   });
39 });
```

394 UTF-8 JavaScript



```
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <title>Hola Mundo!</title>
5   </head>
6   <body>
7     <h1>Hola Mundo!</h1>
8     We are using io.js <script>document.write(process.version)</script>
9     and Electron <script>document.write(process.versions['electron'])</script>.
10  </body>
11 </html>
12
```

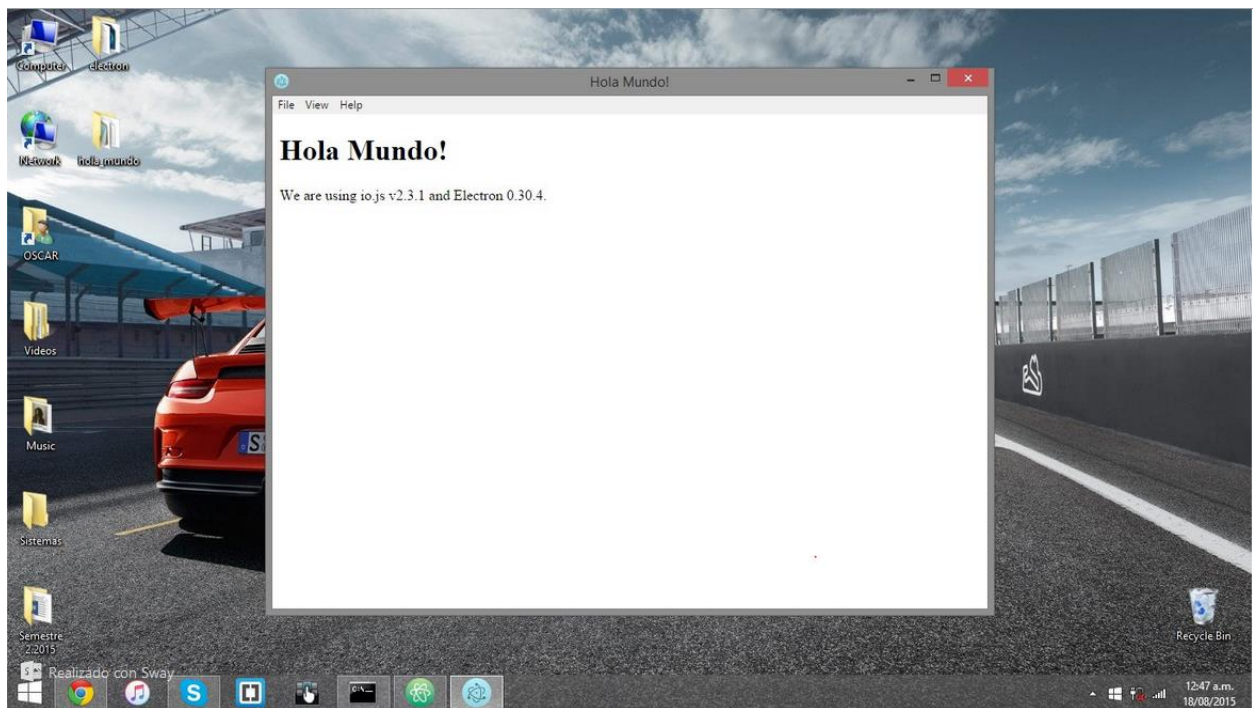
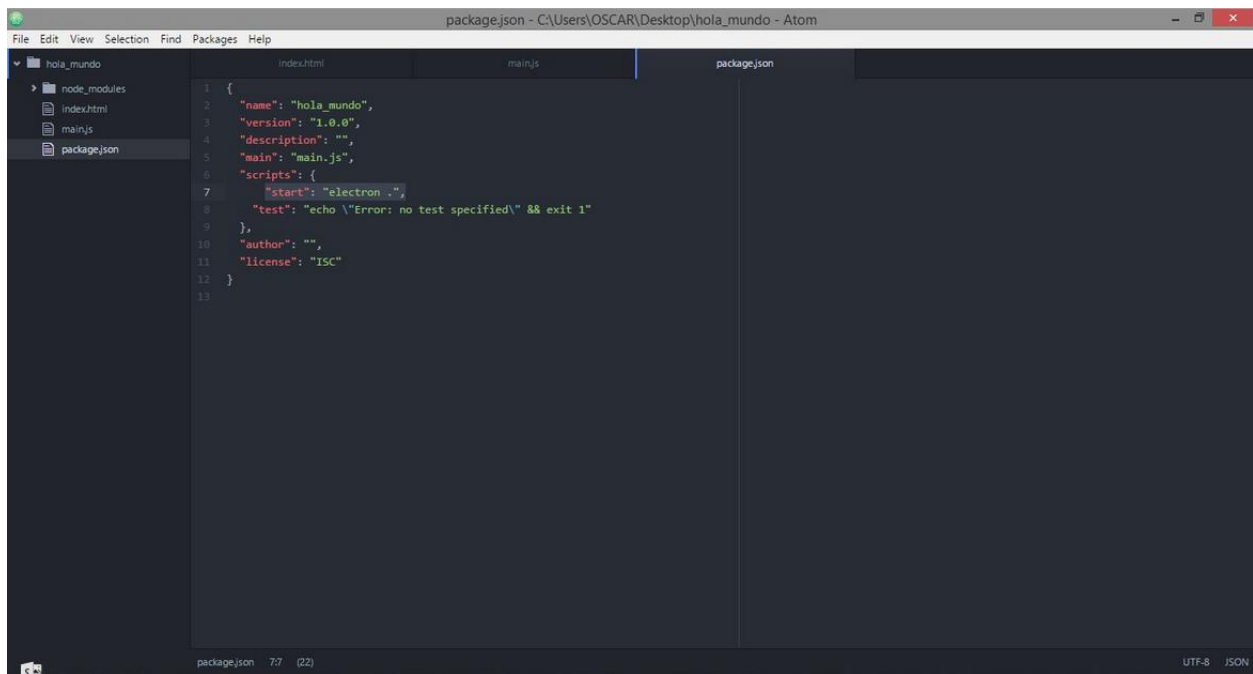
Ejecutar la aplicacion

Antes de ejecutar la aplicacion debemos añadir un script a package.js

"start": "electron .", esto en la parte de script

Ahora ejecutaremos el comando

npm start



Generar ejecutable

Pasos:

1. Instalar electron-packager # for use in npm scripts
\$>npm i electron-packager --save-dev

for use from cli \$>npm i electron-packager -g

1.- En el archivo package.json escribir el siguiente código dentro de scripts:

"build": "electron-packager . MyApp --platform=win32 --arch=x64 --version=0.27.2"

descripción del comando introducido

MyApp = Nombre de la APLICACION --platform =
Plataforma destino sobre la cual correrá la app ejm. all or
linux, win32, darwin (comma-delimited if multiple) --arch =
all, ia32, x64 --version = version de electron

2.- Ejecutar para generar la app de escritorio \$>npm run build

3.- Disfruta de tu app

mayor información en: <https://github.com/maxogden/electron-packager>

