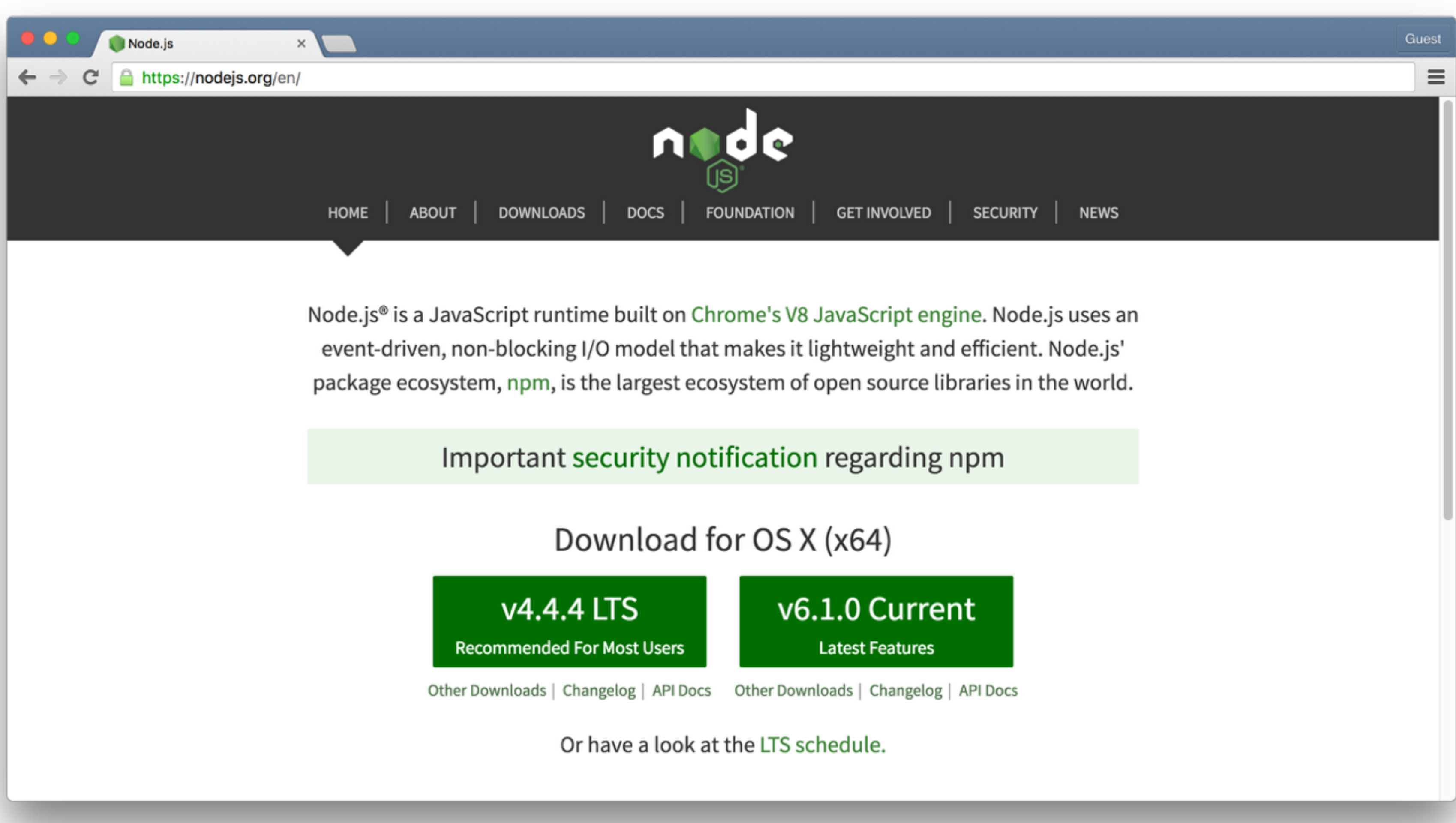


Try Out Electron



Jake Trent

@jaketrent jaketrent.com



HOME

ABOUT

DOWNLOADS

DOCS

FOUNDATION

GET INVOLVED

SECURITY

NEWS

Node.js® is a JavaScript runtime built on [Chrome's V8 JavaScript engine](#). Node.js uses an event-driven, non-blocking I/O model that makes it lightweight and efficient. Node.js' package ecosystem, [npm](#), is the largest ecosystem of open source libraries in the world.

Important [security notification](#) regarding npm

Download for OS X (x64)

v4.4.4 LTS

Recommended For Most Users

v6.1.0 Current

Latest Features

[Other Downloads](#) | [Changelog](#) | [API Docs](#)

[Other Downloads](#) | [Changelog](#) | [API Docs](#)

Or have a look at the [LTS schedule](#).

README.markdown

Node Version Manager

build passing

Installation

First you'll need to make sure your system has a c++ compiler. For OSX, XCode will work, for Ubuntu, the build-essential and libssl-dev packages work.

Note: `nvm` does not support Windows (see [#284](#)). Two alternatives exist, which are neither supported nor developed by us:

- [nvm-windows](#)
- [nodist](#)

Note: `nvm` does not support [Fish](#) either (see [#303](#)). Alternatives exist, which are neither supported nor developed by us:

- [bass](#) allows to use utilities written for Bash in fish shell
- [fast-nvm-fish](#) only works with version numbers (not aliases) but doesn't significantly slow your shell startup
- [fin](#) is a pure fish node version manager for fish shell
- [plugin-nvm](#) plugin for [Oh My Fish](#), which makes nvm and its completions available in fish shell

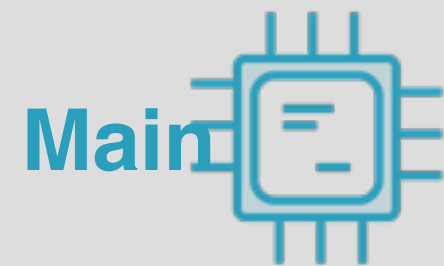
Note: We still have some problems with FreeBSD, because there is no pre-built binary from official for FreeBSD, and building from source may need [patches](#), see the issue ticket:

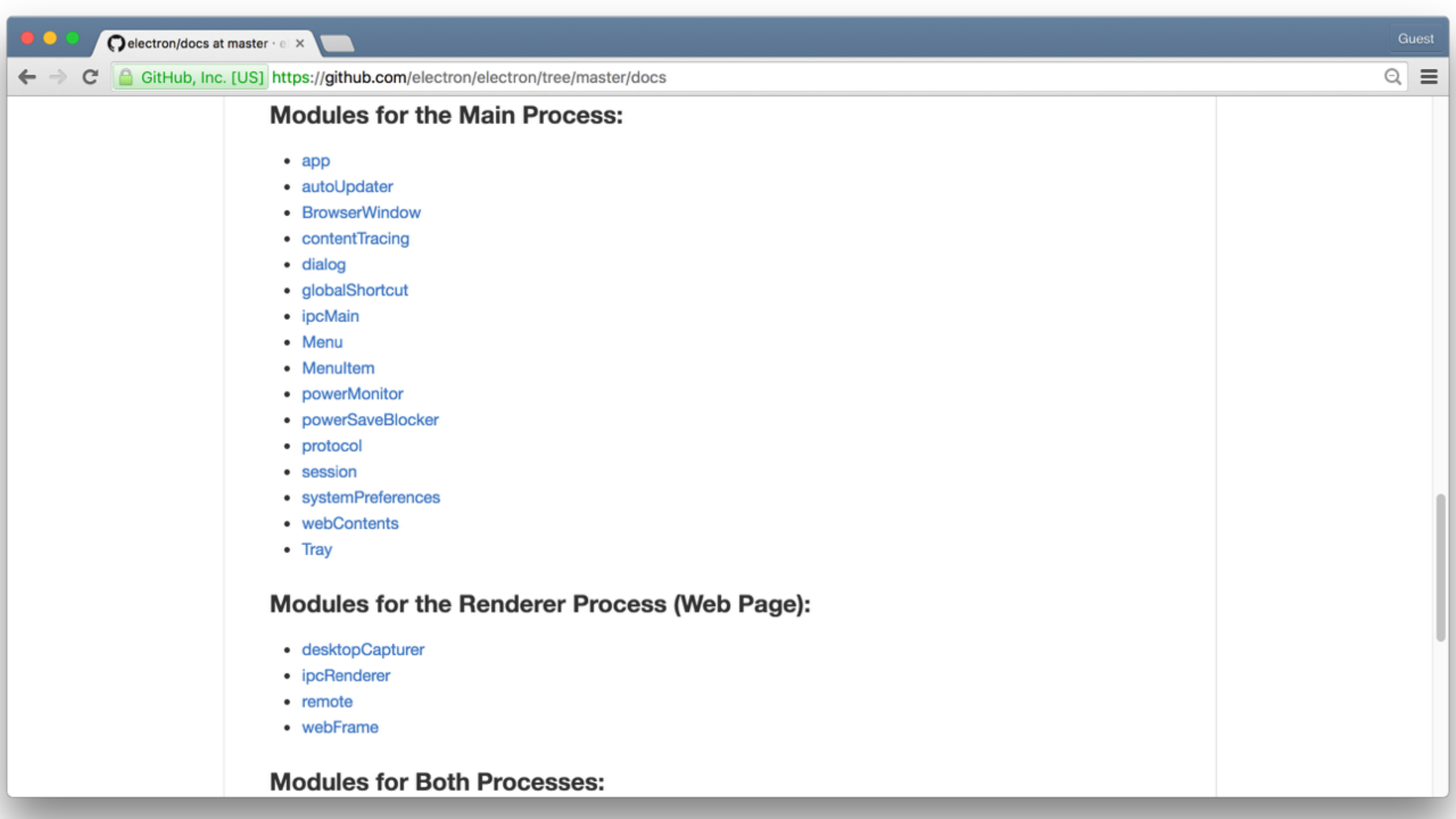


Tech



Process Types





Modules for the Main Process:

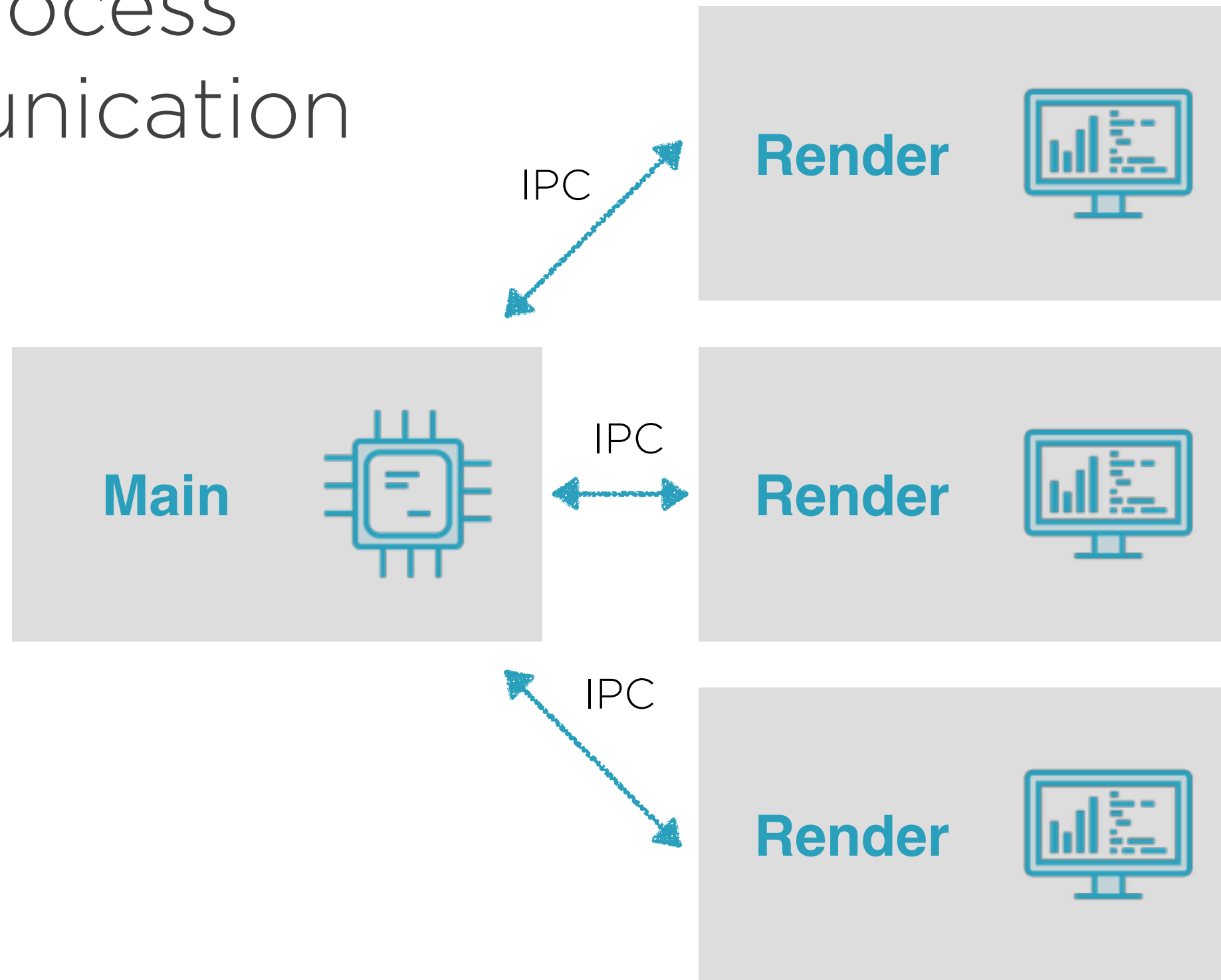
- [app](#)
- [autoUpdater](#)
- [BrowserWindow](#)
- [contentTracing](#)
- [dialog](#)
- [globalShortcut](#)
- [ipcMain](#)
- [Menu](#)
- [MenuItem](#)
- [powerMonitor](#)
- [powerSaveBlocker](#)
- [protocol](#)
- [session](#)
- [systemPreferences](#)
- [webContents](#)
- [Tray](#)

Modules for the Renderer Process (Web Page):

- [desktopCapturer](#)
- [ipcRenderer](#)
- [remote](#)
- [webFrame](#)

Modules for Both Processes:

Inter-process Communication



Example IPC

`ipc.on('start', fn)`



`renderer.webContents
.send('processing')`

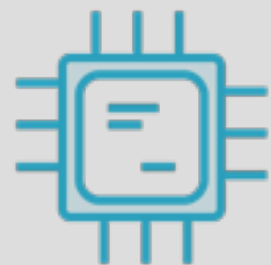


`ipc.send('start')`



`ipc.on('processing', fn)`

Main



Render



Demo

IPC!

Countdown timer app