

Electron, Desktop Development at the Speed of Electricity

by: Aisling Fae

I admit, before doing the research for this piece I had a somewhat low opinion of Electron. My initial experiences using Electron apps were slow and clunky. I didn't realise that two apps I use on the daily, Slack and Discord, are both Electron apps, and they run pretty smoothly. If you, like me, had a low first impression of Electron, or you've never even heard of it, perhaps it is time to give it a second (or first, or third) look!

What is Electron?

Electron is a an open-source Framework for developing Desktop Applications using Web development technologies. It is basically a way for web apps that you would run on a website to run stand-alone on their own window, on their own environment. An Electron App consists of two main parts the Main "Browser" Process, and the various Renderer Processes. The code for the Main Process is written in Node.js. The code for the Rendering Processes is written in HTML and CSS, just like a Webpage. You can even use web development frameworks like Vue.js to write your rendering processes!

Why use Electron?

Electron's number one strength is it's turnaround speed. No other application development framework can go from 0 to fully functioning app as quick as Electron can. Recently we were able to turnaround an app for a client in 2 weeks, because it was built on top of an existing React Library. React, being a web development framework, integrates with Electron Seamlessly.

Another advantage of Electron is that it's super beginner friendly. The amount of knowledge of Video Rendering Libraries one would have to acquire to program a similar application in, say, C++ or Python and have a nice looking User Interface is daunting to say the least. With Electron, if you've ever written a webpage, you can write an Electron App.

And why would you want an Electron app instead of just a webpage? Electron makes it so that the app's environment is consistent. It handles the communication to the OS for you, so it is super simple to export installable packages for Linux, both deb and rpm, Mac OSX, and Windows. It's as simple as running `yarn dist`. It also cuts down on your support costs because you won't have to answer a thousand and one emails with "please use the latest version of Chrome".

The ease of development, and streamlined cross-platforming, does come at a cost, unfortunately.

The not so good about Electron

Electron apps used to be slow, slow enough to turn people off the technology. They've gotten unbelievably faster, but they're still slower and more resource intensive than other native apps. On most rigs that won't matter, but on small laptops or netbooks, they can be a considerable drain on Memory, Processor, and Battery usage.

The installable package that Electron generates tends to run on the large size for simple applications. You're looking at about a 60 MB minimum. Not a lot for a big software suite. But maybe too much for a simple single use applet. This is because with Electron you're kind of throwing everything plus the Kitchen sink along with your app to ensure it runs under any circumstances. Any Electron app comes with its own implementation of the Chromium engine, a sort of proto-web-browser. There will be a lot of things included with your app that your app itself might not use at all.

Trade-offs is the name of the game with modern software development. You're trading for development time, development costs, support costs, and cross-platform compatibility with slightly slower runtimes and slightly higher resource usage. Every developer is going to have to balance the needs and capabilities of their development team with the needs and resources of their clients.

Go forth and Electron

As mentioned, one of the most alluring features of Electron is its beginner friendliness. If you've never developed a GUI desktop app before, but you know HTML, CSS, and JavaScript, you can go on to electronforge.io right now and quickly find yourself `yarn make`ing your own app. If you don't have any web development experience, start by picking up the basics of HTML, CSS, and JavaScript, and also pick up Node.js.

If you, like me, have written tons of programs that run on the command line and always wanted to have a program with a window that users could click on, Electron gives you that chance. It delivers on the same promise that Microsoft Visual Basic delivered on back in the 90s, the promise of the Window, of the Graphical User Interface. Only this time it runs on any OS on any Machine. Go forth and get your Window, Electron makes it easy.