Katelyn Watkins

Kyw356

Journal

* For exploration 4, I decided to go to Glassdoor and search web development jobs. After looking for a bit, I came across a job posting by Lockerdome. The position was a junior web application developer. They are fairly new company, founded in 2008. LockerDome is a native advertising and publishing platform founded in 2008 in [St. Louis, Missouri](https://en.wikipedia.org/wiki/St._Louis,_Missouri), by CEO Gabe Lozano.[[1]](https://en.wikipedia.org/wiki/LockerDome#cite_note-LockerDome_was_founded_in_2008-1) Launched as a sports-themed [social network](https://en.wikipedia.org/wiki/Social_network), it expanded to other interests with the launch of LockerDome 3.0 in December 2013, and has since further broadened its audience by embedding its content [widgets](https://en.wikipedia.org/w/index.php?title=Software_wigdet&action=edit&redlink=1) across third-party media websites. Some of tools on their job posting are sass, HTML5, JavaScript, SockJS, Node.js, Redis, C++, and Git. And after looking at all the various technologies they work with, I got curious about what SockJS was. So, I decided to do this exploration and research on SockJS.
* What is SockJS:
  + SockJS is a JavaScript library (for browsers) that provides a WebSocket-like object.SockJS gives you a coherent, cross-browser, Javascript API which creates a low latency, full duplex, cross-domain communication channel between the browser and the web server, with WebSockets or without.
  + SockJS family:
    - [SockJS-client](https://github.com/sockjs/sockjs-client) JavaScript client library
    - [SockJS-node](https://github.com/sockjs/sockjs-node) Node.js server
    - [SockJS-erlang](https://github.com/sockjs/sockjs-erlang) Erlang server
    - [SockJS-tornado](https://github.com/MrJoes/sockjs-tornado) Python/Tornado server
    - [vert.x](https://github.com/eclipse/vert.x) Java/vert.x server
* Steps:
  + Make you have all npm and node dependencies.
  + Install SockJS with the command: npm install sockjs
  + Once installation is complete you should be to start the server with command: node server.js
  + I then started to do more research on how to create an application
  + I decided to go ahead and make a multiplexer
  + I found these two websites to help me get started:
    - <https://www.npmjs.com/package/websocket-multiplex>
    - <https://www.npmjs.com/package/sockjs-multiplexer>
  + I then used this and started merging it with HTML5 code and made 4 boxes with input field with each box
  + For the script, you need to start off with creating a new sockJS server and new multiplexer. To have multiple channels you need to add pipes with a channel on each pipe.
  + Type into the input a message and the submit button will send, and will say “sending” and then will received
  + You will have register all channels as well, and then express the server.
  + You must use a https or https for the sock\_URL, ex: I use <http://127.0.0.1:9999/multiplex>
* Troubles:
  + Some of the code on the example was outdated and was forbidden (403 error), so I had to figure out another way around it and find how to implement it with newer versions.
  + I couldn’t figure out sock\_URL, and realized it had to have http://
* Kinda new, so not a ton of documentation