**Bootstrap - Custom Admin Theme**

**Environment setup:**

1. Pull in the node package manager into our folder, so we can pull in various packages and import those into our project: **$ npm init**
   1. Package name can left as default
   2. Description can be: Admin Theme
   3. Entry point: js/app.js
   4. Keywords: admin, theme
   5. Author: (your name)
   6. (else can be passed for now by kicking ‘enter’)
2. Create a ‘**.gitignore**’ file, and add into this file the ‘**node\_modules/**’ directory, this way git won’t track all of the scripts that we import into the node modules directory, we don’t want those as part of our repository.
3. Install webpack. Use webpack as the package manager which is going to automatically bundle up all of the JavaScript files into a single JavaScript file and compress it, and automatically take our CSS and sass and convert it to CSS and write that and compress: **$ npm install webpack --save-dev**
   1. Install webpack command-line tool: **$ npm install webpack-cli --save-dev**
   2. Install the development server: **$ npm install webpack-dev-server --save-dev**
   3. Install font awesome, which is the icon font that we’re gonna be using that to generate all the icons for the various setting and switches and toggles inside of our Admin Theme: **$ npm install @fortawesome/fontawesome-free --save-dev**
   4. Make webpack do its thing, which will take all the configuration settings you have inside of your webpack, run them all and then spit out the output stuff: **$ npx webpack --config webpack.config.js**
4. Install fibers: **$ npm install fibers --save-dev**
5. Install SASS: **$ npm install sass --save-dev**
6. In order to deal with live reloading CSS, pulling in CSS, the post CSS loader, which are things that basically required for bootstrap, in order to import and customize bootstrap: **$ npm install sass-loader node-sass css-loader autoprefixer postcss-loader --save-dev**
7. Install file loader: **$ npm install file-loader --save-dev**
8. For pulling out your CSS file and writing it to an individual file: **$ npm install mini-css-extract-plugin --save-dev**
9. Finally, we need bootstrap: **$ npm install bootstrap**
10. When we install bootstrap, it tells us that bootstrap requires a peer of jQuery and popper. In other words, jQuery and popper are both dependencies that are required for bootstrap to work, so we need to add those files: **$ npm install --save jquery popper.js**