

Read through the site code and note the following:

- What directories do you see? How do you interpret their names?

Docs: The set of files that are manipulated by the installed apps

Node-modules: the node.js app resources

Public: The files which can be seen by the end user. These pull the app data from docs to display it on the live site

src: Actively called apps can place source files here

- Where is the Vue app defined? (Which file?)

The Vue App is installed in the node-modules directory but called from the src files as an active app

- What is listed in `package.json`?

Version data and Dependencies to keep the active apps running

Run `npm run build` and take a look at both the webpage that comes up and the output in the console.

Consider the following questions:

- what is in the `node_modules` directory?

The installed Node.js apps

- where can you find the directions for the scripts run with `npm run`, that is the `serve` and `build` scripts

The NPM online documentation

- what's the difference between the `dependencies` and `devDependencies` object in the `package.json` file?

Dependencies are required for the whole project while DevDependencies are only necessary for development builds

- Explore the `docs` directory. What do you see?

The altered files that were generated on the latest build

- Do you see the filenames of the static files? What seems odd about those filenames?

They have an app prefix that I assume is used to help with calling the files

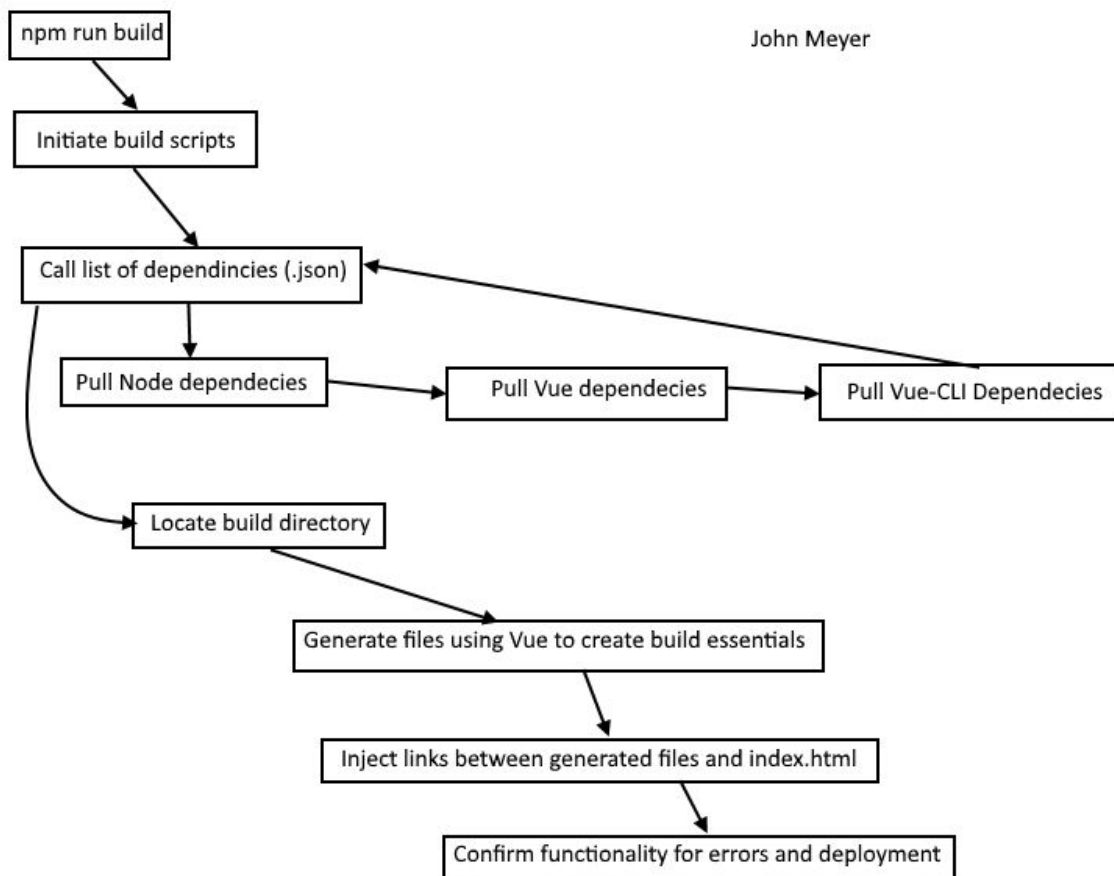
- Do you see the contents of your JS and CSS files? What has happened to those contents?

They appear to have been minified

- Three config files have been supplied for you: `vue.config.js`, `aliases.config.js` and `.babel.config.js`. What is the general purpose of each file.

They call the required directories to run each part of Vue

- Describe (in words and with a flowchart/diagram) what happens when the `npm run build` command is executed to the best of your ability.



- Make a diagram of the components of this system like the ones shown in the Practical JavaScript 2: Building Applications book *Types of Website*

Architectures. Do your best to document your interpretation of the architecture of this system.

John Meyer

