

Setting up a Node.js-based Front end app

© Prashanth Puranik, www.digdeeper.in

1. Install Node.js (version 14) from nodejs.org. Open the command prompt and check the installation went on fine (you should see the installed Node and npm versions).

```
node --version  
npm --version
```

2. In the command prompt, navigate to a folder where you would like to create the project folder. Now create the folder and switch to it (do not use spaces in the project name)

```
mkdir my-project  
cd my-project
```

NOTE: On Windows you need to use md

3. Within the project folder run the all of the following commands. The one below creates a package.json.

```
npm init -y
```

4. Create a public folder to house CSS, JS and HTML files.
5. Now we can start installing dependencies (they get installed in a folder called node_modules. Install live-server, cpx. Since these are development time dependencies we use the --save-dev flag (marks the dependencies in package.json)

```
npm i --save-dev cpx live-server
```

Install normalize and fontawesome

```
npm i normalize.css @fortawesome/fontawesome-free
```

6. Within a package.json script, configure live-server to start up and serve the files from, say, the public folder. Here we also make sure it opens up with the index.html page when server start up.

```
"start": "live-server --mount=/:public --open=/ --entry-file=index.html"
```

Within a package.json script, configure cpx to copy required files from node_modules to desired folder

```
"copy": "cpx \"node_modules/normalize.css/normalize.css\" public/external",
```

7. Run the copy script in order to copy desired files from node_modules to public/external folder

```
npm run copy
```

8. Run live-server in order to serve the web application

```
npm run start
```

NOTE: Any time you would like to install all packages again just run

```
npm install
```

© Prashanth Puranik, www.digdeeper.in