**Vue**

* Download Node JS from their website
* Use command: “npm install -g npm” to install npm (node package manager) replaceable by yarn
* “npm install -g @vue/cli” **#OR “**yarn global add @vue/cli”
* Create a new project using “vue create project-name”
* Navigate to the project using cd project-location
* Run the project using npm run serve
* Open the page .vue file using code editor and add template
* Dynamic information is denoted by {{ initial-message }}
* Run app in .vue file of the page to change value of information to Hello World

**Vue File-Structure**

There is the main app.vue file that has initial template and routes to other views, router-link is used to reference to other view files which nested inside app.js file. More focused on single page apps

Single page apps don’t send new requests to servers and are intercepted by vue or other frameworks that change the viewmodel based on the given script

**Vue Array-Handling**

Data from an array can be pushed to markup using v-for=”boxitem in box” :key=”item.id” attributes

<li v-for=”boxitem in box” :key=”item.id”>

<h1> {{ boxitem.boxName }} </h1>

<p> {{ boxitem.boxColor }} </p>

</li>

An array is added to data as

box: [

{boxName: ‘Box1’, boxColor: ‘red’}

{boxName: ‘Box2’, boxColor: ‘blue’}

{boxName: ‘Box3’, boxColor: ‘pink’}

]

**Files:**

**package-lock.json:** The goal of package-lock. json file is to keep track of the exact version of every package that is installed so that a product is 100% reproducible in the same way even if packages are updated by their maintainers. This solves a very specific problem that package. json left unsolved.

**package.json:** A package. json is a JSON file that exists at the root of a Javascript/Node project. It holds metadata relevant to the project and it is used for managing the project's dependencies, scripts, version and a whole lot more.

**Babel:** Babel is used to convert JSX syntax to javascript

**Jsconfig.json:** Used by editor to configure which folders of the project to include when running it

**manifest.json:** The web app manifest provides information about an application (such as name, author, icon, and description) in a JSON text file. The purpose of the manifest is to install web applications to the homescreen of a device, providing users with quicker access and a richer experience.

**robots.txt:** The robot exclusion protocol, better known as the robots. txt, is a convention to prevent web crawlers from accessing all or part of a website. It is a text file used for SEO, containing commands for the search engines' indexing robots that specify pages that can or cannot be indexed.

**Creating React Project:**

* Npx create-react-app project-name
* Cd project-location
* Npm start
* Edit index.js to include components and provide base structures
* Create a component with the initial message
* Import it in index as:

**ReactDOM.render(InitialMessage(message), document.getElementById("root"));**

* Give message =” Hello World” when calling the component in index