Lab 1: Create a React Application

Step 1: Setting Up the Basic React App

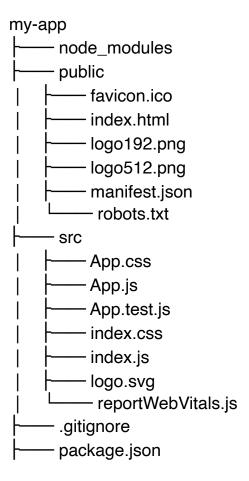
- 1. **Install Node.js and npm:** Make sure you have Node.js and npm installed. You can download them from <u>nodejs.org</u>.
- 2. Create a React App: Use the create-react-app command to set up a new React project. Open your terminal and run:

npx create-react-app my-app

3. cd my-app

Step 2: Exploring the File Structure

After the setup is complete, you'll have a directory structure that looks something like this:



README.md
yarn.lock / package-lock.json

Step 3: Understanding Each File and Directory

1. node modules

- Purpose: Contains all the npm packages and dependencies required by your React app.
- **Details:** This folder is automatically created when you run npm install and you typically don't need to interact with it directly.

2. public

 Purpose: Holds static assets that will be served directly without processing by Webpack.

Key Files:

- index.html: The main HTML file that gets served. It includes a <div
 id="root"></div> where your React app will be injected.
- favicon.ico: The favicon for your app.
- manifest.json: Used for configuring how your app behaves when installed on a device or served as a Progressive Web App (PWA).
- robots.txt: Instructs web crawlers how to interact with your site.

3. src

Purpose: Contains the React components and the main code for your application.

Key Files:

- App.js: The main React component, usually the root of your component tree.
- App.css: Styling specific to the App component.
- App.test.js: Contains tests for the App component using Jest.
- index.js: The entry point of your React application. This file renders the App component into the DOM.

- index.css: Global styles for your application.
- logo.svg: A React component that displays a logo, used in App.js.
- reportWebVitals.js: Used for measuring performance in your app.

4. Root Files

• **Purpose:** Configuration and meta-information about your project.

Key Files:

- .gitignore: Specifies files and directories that should be ignored by Git.
- package.json: Lists the dependencies, scripts, and metadata for your project.
- README .md: Contains basic information about your project.
- yarn.lock or package-lock.json: Ensures that the exact same versions of dependencies are installed every time.