WEB TECHNOLOGIES

Set-up React Installation (npm)

Prof. Pavan A C
Department of Computer Science and Engineering

Acknowledgement:
Teaching Assistants(Harini B and Chandana MS)

Set-up React Installation (npm)

What is npm

NPM (Node Package Manager) is a package manager for NodeJS modules. It helps developers manage project dependencies, scripts, and third-party libraries. By installing NodeJS on your system, NPM is automatically installed and ready to use.

A package in NodeJS is a reusable module of code that adds functionality to your application. It can be anything from a small utility function to a full-featured library.

- Packages can be installed from the NPM registry.
- They are stored in the node_modules folder in your project.
- You can easily install, update, or remove packages with NPM commands.

Set-up React Installation (npm)

Installing NODE JS

Install NodeJS and NPM

https://www.geeksforgeeks.org/installation-guide/install-node-js-windows-macos-linux/

During installation, make sure to add Node.js to your PATH to ensure that both Node.js and npm are accessible globally in the command line.

Confirm installation in Terminal/Command Prompt:

```
node -v
npm -v
```

Set-up React Installation (npm)

Setting up React Project

1. **Create React App (Traditional, now legacy)**

```
npx create-react-app my-app

cd my-app

npm start
```

Then open http://localhost:3000/ to see your app.

1. **Vite (Modern, very fast)**

```
npm create vite@latest my-react-app -- --template react
```

Then open http://localhost:5173/ to see your app.

Replace my-react-app with your desired project name.

Set-up React Installation (npm)

Setting up React Project

Install Project Dependencies

- Move into your project directory:

```
cd my-react-app
```

- Install all dependencies:

```
npm install
```

To install specific dependencies

- Run

```
npm install <package-name>
```

Set-up React Installation (npm)

Project Structure

Project Structure Overview

A new React app (created with Create React App) has this folder structure:

```
my-app/
├── README.md
├── node_modules/
├── package.json
├── public/
│   └── index.html
├── src/
│   ├── App.js
│   ├── App.css
│   ├── index.js
│   └── ...
```

Not necessarily the same but similar to this structure.

Project Structure

node_modules/

- Contains all npm packages and dependencies your project uses
- Created by `npm install`
- You never edit files here directly!
- It can be safely deleted and restored using `npm install`.

package.json

- The heart of npm project setup
- Lists all dependencies the project needs (e.g. React, ReactDOM)
- Stores scripts (like `start`, `build`) and project metadata
- Allows easy sharing: cloning a project and running `npm install` sets up everything!

Setting up React Project

public/ Folder

- Contains static files served directly to the browser
- Most important: `index.html` – the root HTML page
- Other files: `favicon.ico` (tab icon), `manifest.json` (PWA config), etc.
- You rarely need to edit these, except for branding or adding static assets

src/ Folder

- Where almost ALL your code lives: components, logic, styles, tests
- Contains files like:
  - `App.js` (Main component, the “root” of your app)
  - `App.css` (related styles)
  - `index.js` (entry point that renders your app)
  - Other components, assets, tests, etc.

Set-up React Installation (npm)

Setting up React Project

App.js & index.js

App.js:

- Main React component.
- Build most of your app’s logic and UI here (or import from other components).

index.js:

- Entry point for React.
- Imports App and renders it into the DOM via `ReactDOM.render()`.

README.md & .gitignore

- README.md:
  - Project info, setup notes, usage instructions
- .gitignore:
  - Tells Git which files/folders to ignore (e.g., `node_modules/`, build files)

Setting up React Project

Commonly Added Folders (as your app grows)

- components/ — reusable UI parts
- pages/ — for “screens”/routes in multi-page apps
- assets/ — images, fonts, static assets
- hooks/ — custom React Hooks
- contexts/ — shared state with React Context API
- styles/ — CSS or styling files
- utils/ — helper functions

(*You can organize as needed—there’s no one “correct” way!*)

Core npm Commands with Create React App

- `npm start`

  Starts development server (auto-reloads on file changes)

- `npm run build`

  Bundles app for production (creates `/build` folder)

- `npm test`

  Runs tests (if present, uses Jest by default)

- `npm install <package>`

  Installs additional packages (e.g., `npm install axios`)

`PS : you'll majorly just be using npm start and npm install <package>`

Set-up React Installation (npm)

Links

Do have a look into these links and explore more as well

- https://www.freecodecamp.org/news/how-to-install-react-a-step-by-step-guide/
- https://create-react-app.dev/docs/getting-started/
- https://www.geeksforgeeks.org/node-js/node-js-npm-node-package-manager/
- https://kinsta.com/knowledgebase/install-react/
- https://www.w3schools.com/react/react_getstarted.asp
- https://www.pluralsight.com/resources/blog/guides/npm-start-for-react-tutorial-project

On folder Structure

https://blog.webdevsimplified.com/2022-07/react-folder-structure/

https://www.robinwieruch.de/react-folder-structure/

THANK YOU

Prof. Pavan A C

Department of Computer Science and Engineering