building and deploying a React application for an online portfolio to showcase our projects and skills :-

Here's a full code and process; then we'll turn this into a GitHub repo + PDF/README.

---

1. Project setup and structure

1. Create app:-

```
npx create-react-app reactnileshportfolio
cd reactnileshportfolio
```

2. Basic structure inside src/ (you can adjust names):-

```
Public/
        nilesh_kumar_ahirwar.pdf
src/
 components/
        Navbar.jsx
        About.jsx
        Projects.jsx
        Skills.jsx
        Resume.jsx
        Contact.jsx
        Layout.jsx
```

3. In src/index.js:-

```
import React from 'react';
import ReactDOM from 'react-dom/client';
import './index.css';
import App from './App';
import reportWebVitals from './reportWebVitals';

const root = ReactDOM.createRoot(document.getElementById('root'));
root.render(
 <React.StrictMode>
  <App />
 </React.StrictMode>
);
```

```
// If you want to start measuring performance in your app, pass a function
// to log results (for example: reportWebVitals(console.log))
// or send to an analytics endpoint. Learn more: https://bit.ly/CRA-vitals
reportWebVitals();
```

4. In src/App.jsx (basic layout + sections):-

```
import React, { Suspense } from "react";
import Navbar from "./components/Navbar";
import Layout from "./components/Layout";
import About from "./components/About";
const Projects = React.lazy(() => import("./components/Projects"));
const Skills = React.lazy(() => import("./components/Skills"));
const Resume = React.lazy(() => import("./components/Resume"));
const Contact = React.lazy(() => import("./components/Contact"));

function App() {
 return (
  <Layout>
   <Navbar />
   <main>
    <About />
    <Suspense fallback={<div>Loading sections...</div>}>
     <Projects />
     <Skills />
     <Resume />
     <Contact />
    </Suspense>
   </main>
  </Layout>
 );
}

export default App;
```

2. Core sections (About, Projects, Skills, Resume, Contact)

Create these files in src/components/.

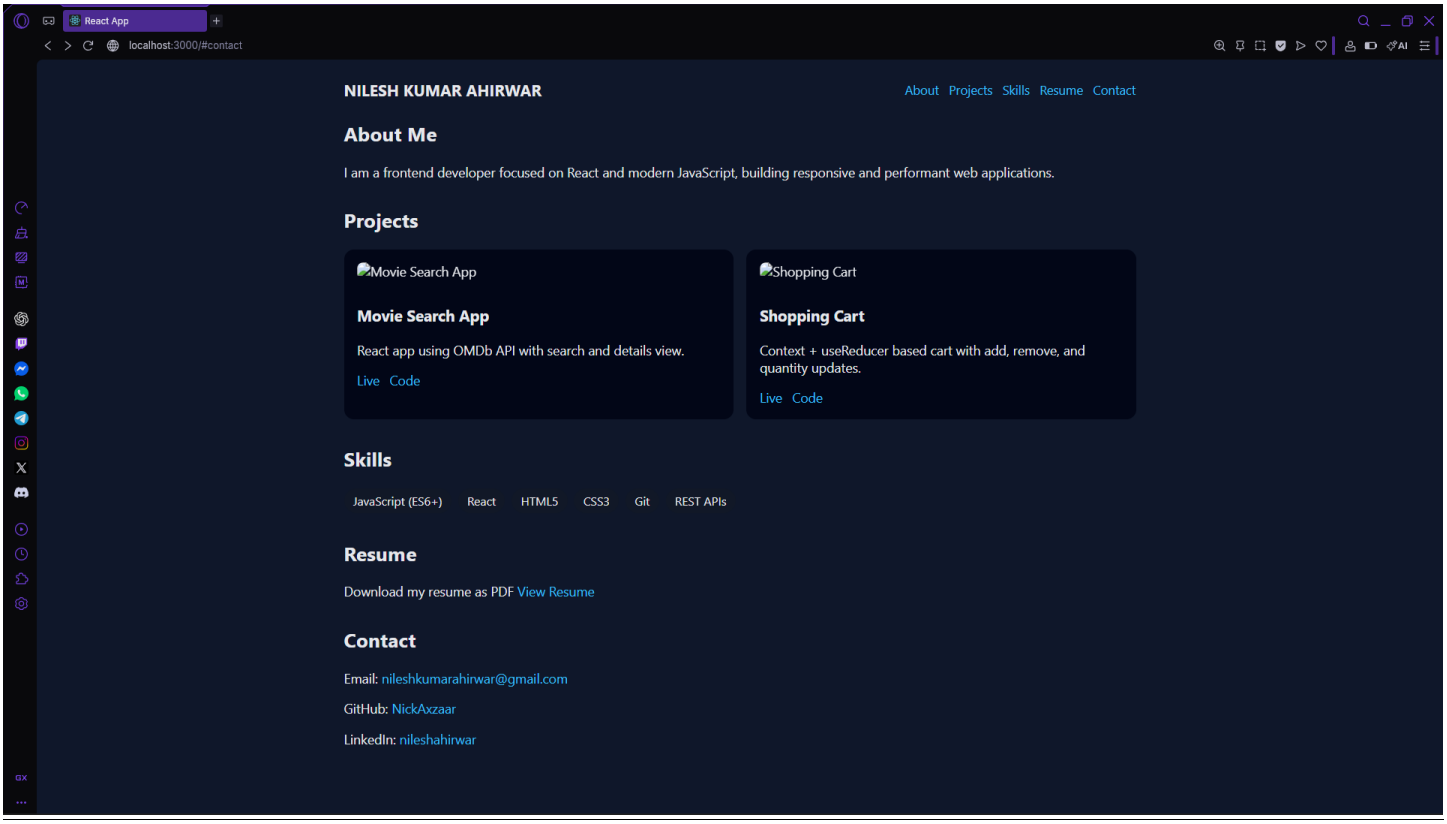
2.1 Layout (responsive container)

Layout.jsx:-

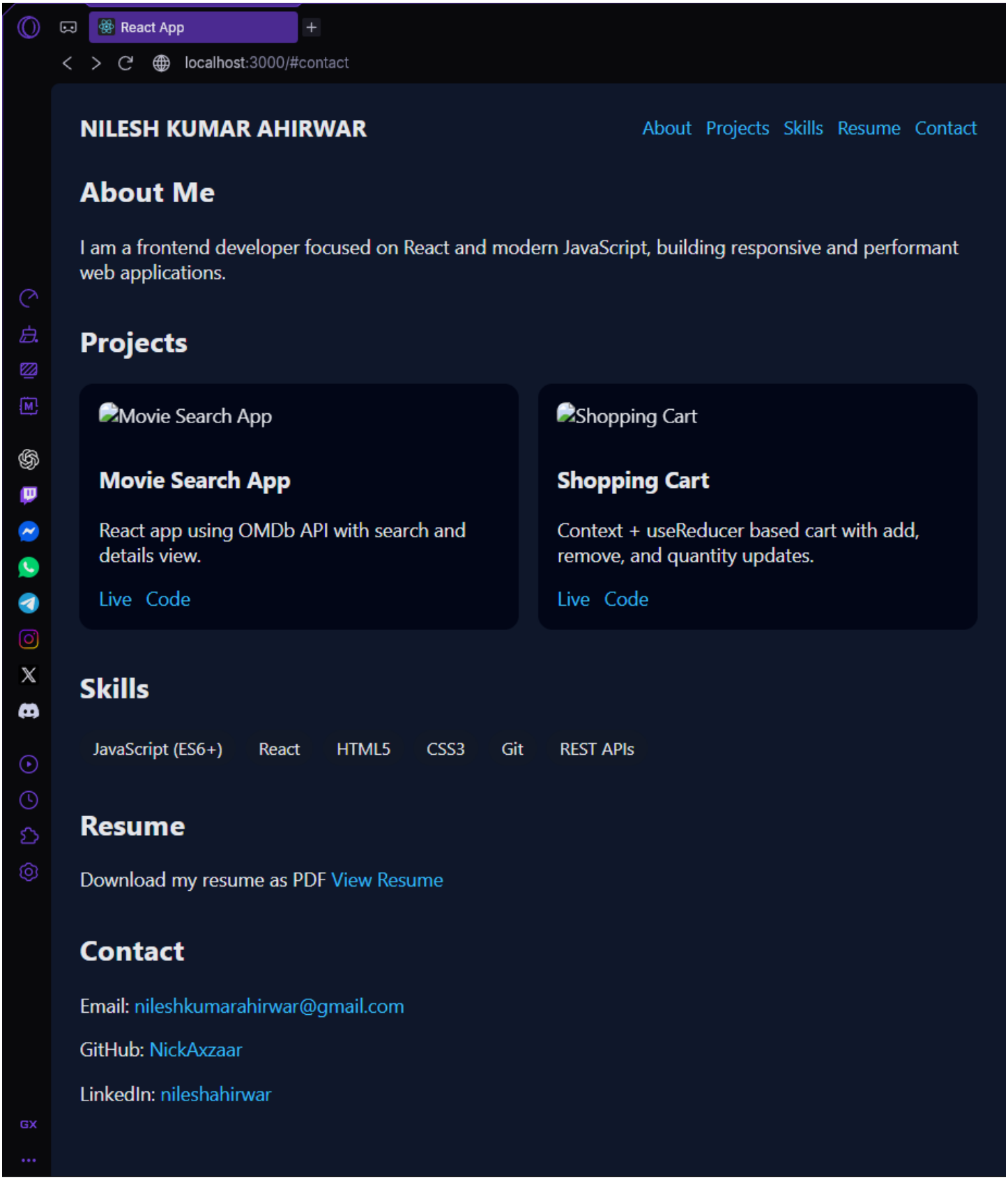
```
import React from "react";

function Layout({ children }) {
 return (
  <div
   style={{
    minHeight: "100vh",
    fontFamily: "system-ui, -apple-system, BlinkMacSystemFont, 'Segoe UI'",
    background: "#0f172a",
    color: "#e5e7eb",
   }}
  >
   <div style={{ maxWidth: "960px", margin: "0 auto", padding: "1.5rem" }}>
    {children}
   </div>
  </div>
 );
}

export default Layout;
```

Mobile Responsive Output ScreenShot

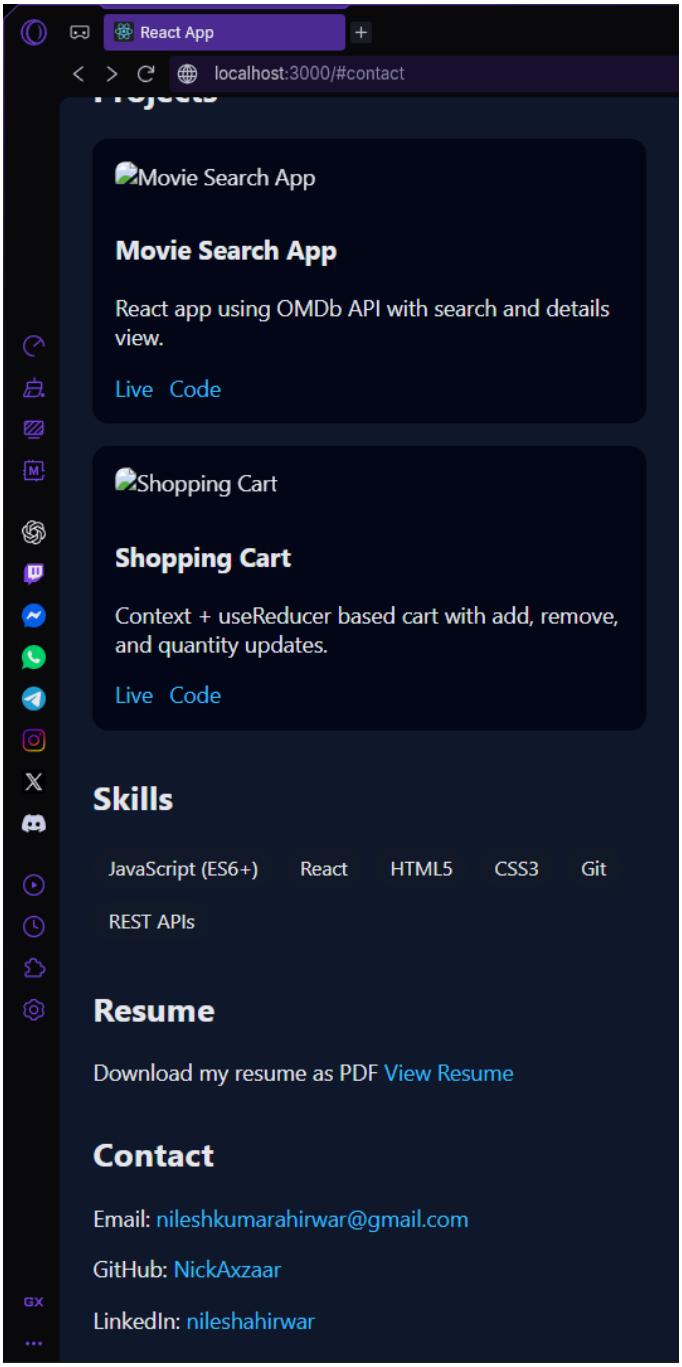
For Desktop, Laptop Landscape Size :-



For Tab or Pad Screen Size :-



For Mobile Phones Screen Size :-



2.2 Navbar with anchor links

Navbar.jsx:-

```
import React from "react";

function Navbar() {
 return (
  <header
   style={{
```

```
      display: "flex",
      justifyContent: "space-between",
      alignItems: "center",
      marginBottom: "1.5rem",
    }}
  >
    <div style={{ fontWeight: 700, fontSize: "1.2rem" }}>NILESH KUMAR AHIRWAR</div>
    <nav style={{ display: "flex", gap: "0.75rem", fontSize: "0.95rem" }}>
      <a href="#about">About</a>
      <a href="#projects">Projects</a>
      <a href="#skills">Skills</a>
      <a href="#resume">Resume</a>
      <a href="#contact">Contact</a>
    </nav>
  </header>
 );
}

export default Navbar;
```

2.3 About section

About.jsx:-

```
import React from "react";

function About() {
 return (
  <section id="about" style={{ marginBottom: "2rem" }}>
    <h2>About Me</h2>
    <p>
      I am a frontend developer focused on React and modern JavaScript,
      building responsive and performant web applications.
    </p>
  </section>
 );
}

export default About;
```

2.4 Projects section (with lazy-loaded images)

Projects.jsx:-

```
import React from "react";

const projects = [
 {
  id: 1,
  title: "Movie Search App",
  description: "React app using OMDb API with search and details view.",
  image: "/images/movie-app.png", // put sample images in public/images
  link: "https://nickaxzaar.github.io/Movies-Searching-App/",
  repo: "https://github.com/NickAxzaar/Movies-Searching-App",
 },
 {
  id: 2,
  title: "Shopping Cart",
  description:
   "Context + useReducer based cart with add, remove, and quantity updates.",
  image: "/images/cart-app.png",
  link: "https://nickaxzaar.github.io/Shopping-Cart-with-Context-and-Reducers/",
  repo: "https://github.com/NickAxzaar/Shopping-Cart-with-Context-and-Reducers",
 },
];

function Projects() {
 return (
  <section id="projects" style={{ marginBottom: "2rem" }}>
   <h2>Projects</h2>
   <div
    style={{
     display: "grid",
     gap: "1rem",
     gridTemplateColumns: "repeat(auto-fit, minmax(260px, 1fr))",
    }}
   >
    {projects.map((project) => (
     <article
      key={project.id}
      style={{
       borderRadius: "0.75rem",
```

```
          background: "#020617",
          padding: "1rem",
         }}
        >
         <img
          src={project.image}
          alt={project.title}
          loading="lazy"
          style={{
           width: "100%",
           borderRadius: "0.5rem",
           marginBottom: "0.75rem",
          }}
         />
         <h3>{project.title}</h3>
         <p>{project.description}</p>
         <div style={{ marginTop: "0.5rem", display: "flex", gap: "0.75rem" }}>
          {project.link && (
           <a href={project.link} target="_blank" rel="noreferrer">
            Live
           </a>
          )}
          {project.repo && (
           <a href={project.repo} target="_blank" rel="noreferrer">
            Code
           </a>
          )}
         </div>
        </article>
      ))}
     </div>
    </section>
  );
}

export default Projects;
```

2.5 Skills section

Skills.jsx:-

```
import React from "react";

const skills = ["JavaScript (ES6+)", "React", "HTML5", "CSS3", "Git", "REST APIs"];

function Skills() {
 return (
  <section id="skills" style={{ marginBottom: "2rem" }}>
   <h2>Skills</h2>
   <ul
    style={{
     display: "flex",
     flexWrap: "wrap",
     gap: "0.5rem",
     listStyle: "none",
     padding: 0,
    }}
   >
    {skills.map((skill) => (
     <li
      key={skill}
      style={{
       padding: "0.35rem 0.7rem",
       borderRadius: "999px",
       background: "#111827",
       fontSize: "0.9rem",
      }}
     >
      {skill}
     </li>
    ))}
   </ul>
  </section>
 );
}

export default Skills;
```

2.6 Resume section

Resume.jsx:-

```
import React from "react";

function Resume() {
 return (
  <section id="resume" style={{ marginBottom: "2rem" }}>
   <h2>Resume</h2>
   <p>
    Download my resume as PDF{" "}
    <a
     href="/nilesh_kumar_ahirwar.pdf"
     download="nilesh_kumar_ahirwar.pdf"
    >
     View Resume
    </a>
   </p>
  </section>
 );
}

export default Resume;
```

2.7 Contact section

Contact.jsx:-

```
import React from "react";

function Contact() {
 return (
  <section id="contact" style={{ marginBottom: "2rem" }}>
   <h2>Contact</h2>
   <p>Email: <a
href="mailto:nileshkumarahirwar991@gmail.com">nileshkumarahirwar@gmail.com</a></p>
   <p>GitHub: <a href="https://github.com/NickAxzaar/">NickAxzaar</a></p>
   <p>LinkedIn: <a href="https://linkedin.com/in/nileshahirwar">nileshahirwar</a></p>
  </section>
 );
```

```
}

export default Contact;
```

---

3. Responsive design basics

In src/index.css, add simple responsive styling:-

Index.css:-

```
body {
  margin: 0;
  font-family: -apple-system, BlinkMacSystemFont, 'Segoe UI', 'Roboto', 'Oxygen',
    'Ubuntu', 'Cantarell', 'Fira Sans', 'Droid Sans', 'Helvetica Neue',
    sans-serif;
  -webkit-font-smoothing: antialiased;
  -moz-osx-font-smoothing: grayscale;
}

code {
  font-family: source-code-pro, Menlo, Monaco, Consolas, 'Courier New',
    monospace;
}

a {
  color: #38bdf8;
  text-decoration: none;
}

a:hover {
  text-decoration: underline;
}

@media (max-width: 640px) {
  header nav {
    flex-wrap: wrap;
    justify-content: flex-end;
  }
}
```

The grid with repeat(auto-fit, minmax(...)) in Projects already makes cards responsive.

---

4. Performance: lazy loading and minification

1. **Component lazy loading**:- Already using React.lazy + Suspense for Projects, Skills, Resume, Contact. These are code-split into separate JS chunks in production.
2. **Image lazy loading**:- Using loading="lazy" on project images so below-the-fold images load only when needed.
3. **Minified JS/CSS**:- Create React App's production build automatically:
    - Minifies JavaScript and CSS
    - Hashes filenames for caching
    - Tree-shakes unused code

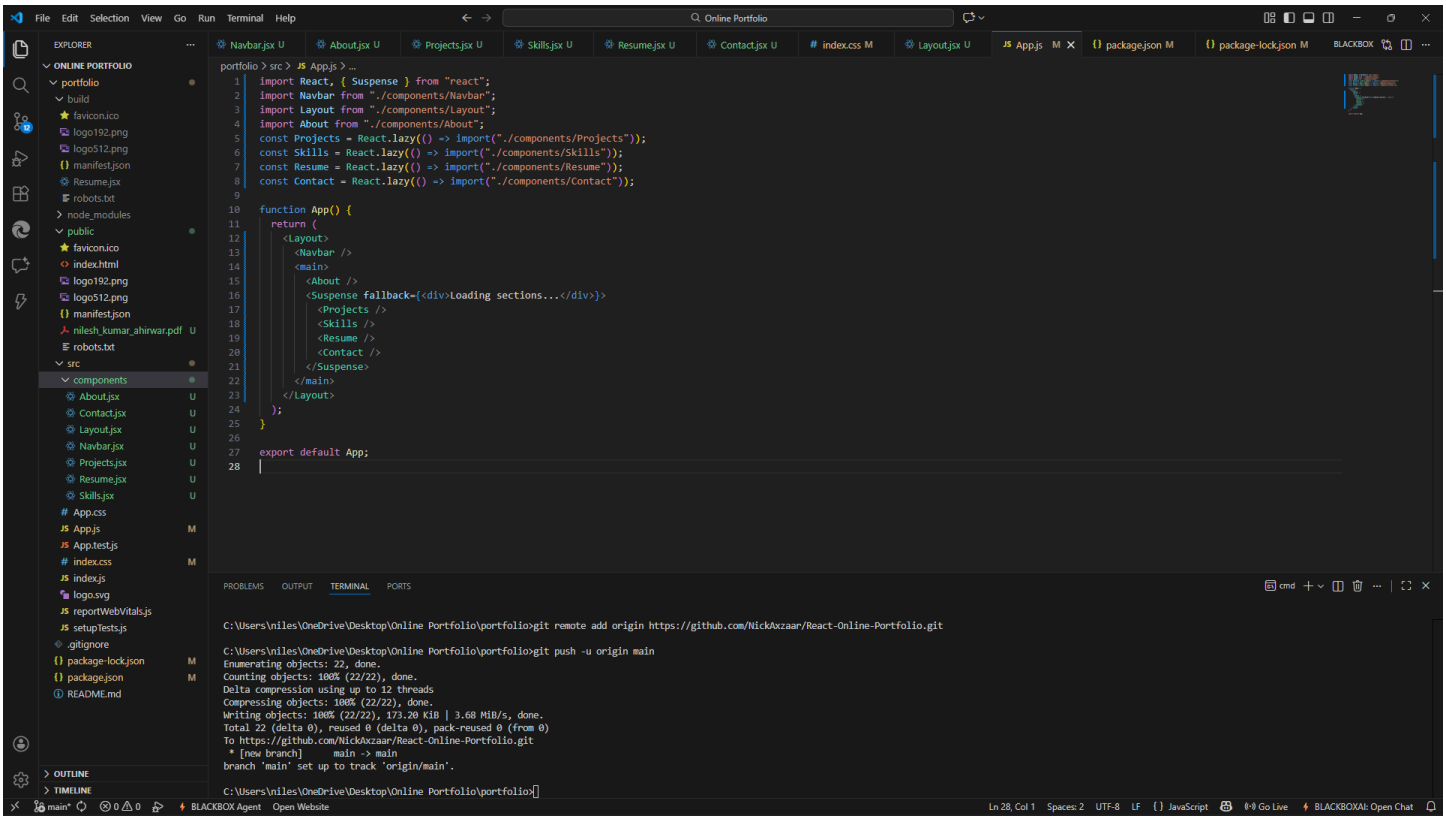
---

5. Production build and local verification

1. Install dependencies and build:-

```
npm install
npm run build
```

This creates a build/ folder with optimized assets:-

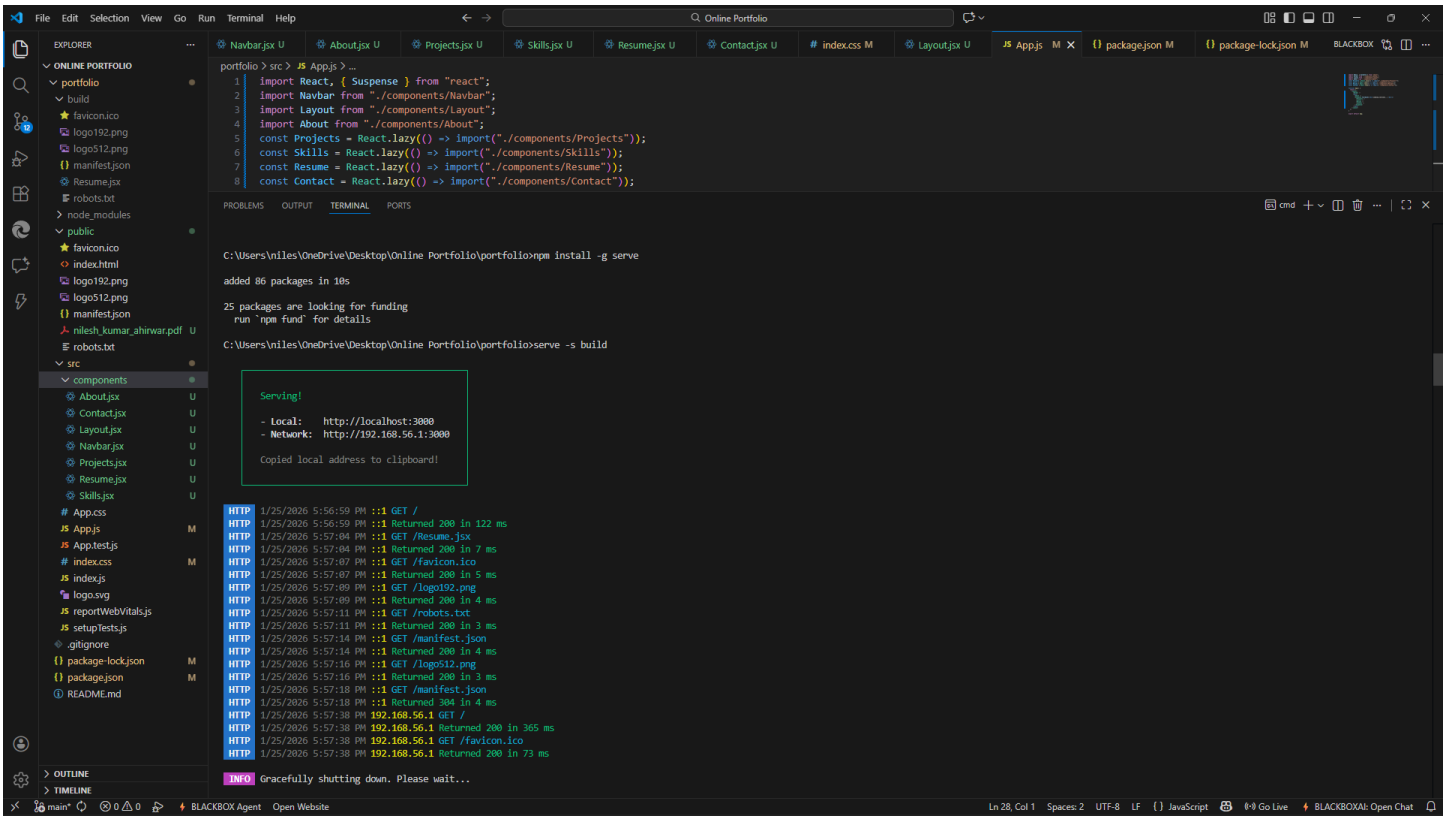


2. To test locally (one option):-

```
npm install -g serve
serve -s build
```

Then open the local URL (usually http://localhost:3000 or http://localhost:4173) and check:

- All sections render
- Navigation links scroll correctly
- Images load lazily (check Network tab)
- No console errors



6. Deployment

GitHub

1. Install dev dependency:-

```
npm install gh-pages --save-dev
```

2. In package.json add:-

```
 "homepage": "https://myusername.github.io/my-app",

"scripts": {
      "predeploy": "npm run build",
      "deploy": "gh-pages -d build",
}
```

Then:-

```
npm install --save gh-pages
```

3. Deploy:-

```
npm run deploy
```

our app will be live at the homepage URL.

---

7. Testing after deployment

1. **Accessibility and functionality**
    - Open the deployed URL.
    - Test navigation, links, resume download, contact links.
    - Check for errors in browser console.
2. **Cross-browser**
    - Test in Chrome, Firefox, Edge; if possible Safari.
    - Verify layout and fonts; ensure no critical JS errors.
3. **Responsiveness**

- Use DevTools → responsive mode.
- Test breakpoints: 320–480 px (mobile), 768 px (tablet), >1024 px (desktop).
- Ensure navbar and sections are readable and not overflowing.

---

8. GitHub repo, README, and screenshots

1. Initialize git and push:-

```
echo "# React-Online-Portfolio" >> README.md
git init
git add README.md
git commit -m "my react portfolio application"
git branch -M main
git remote add origin https://github.com/NickAxzaar/reactnileshportfolio.git
git push -u origin main
```

Created README.md in project root, include sections like:-

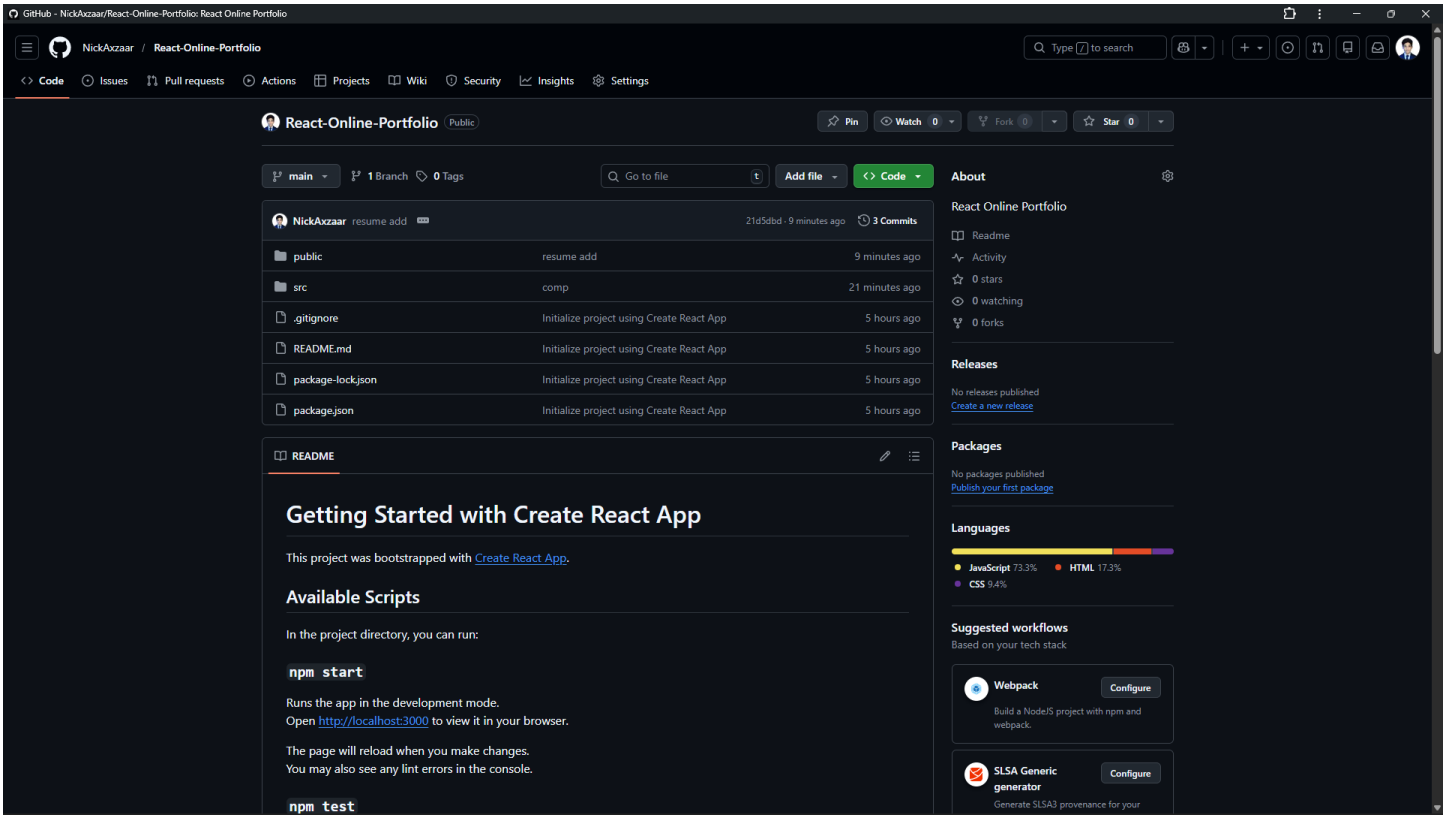
- Project title and description
- Tech stack (React, CSS)
- Features:-
    - About, Projects, Skills, Resume, Contact sections
    - Responsive layout
    - Lazy-loaded components and images
- Performance optimizations (lazy loading, production build)
- Production build commands (npm run build)
- Deployment steps (Netlify/Vercel/GitHub Pages) and final URL
- Challenges and solutions (for example: organizing sections, image optimization, deployment settings)
- Screenshots (/screenshots/home.png, /screenshots/projects.png) and deployed link

3. Screenshots:
    - Take 2–3 screenshots (desktop + mobile view).
    - Save in screenshots/ folder.
    - Reference in README using markdown:

![Home page](./screenshots/home-desktop.png)

![Mobile view](./screenshots/home-mobile.png)

4. PDF for assignment:-
    - I can export the README or a separate document (with same info) to PDF from VS Code, browser, or a Markdown-to-PDF tool.



https://github.com/NickAxzaar/reactnileshportfolio
