**FitFlex - Frontend Development with React.js**

**1. Introduction**

* **Project Title:** FitFlex

**Team Members:**

|  |  |
| --- | --- |
| **Team Members Name** | **Email id** |
| Guna Sundhari M **(Leader)** | Ca2201111033015@lngovernmentcollege.com |
| Dinesh P | Ca2201111033014@lngovernmentcollege.com |
| Harinath S | Ca2201111033016@lngovernmentcollege.com |
| Magizhnan S | Ca2201111033017@lngovernmentcollege.com |

**2. Project Overview**

* **Purpose:** FitFlex is a cutting-edge fitness application designed to enhance user engagement with exercise and wellness. It provides a seamless experience for discovering, customizing, and tracking workouts.
* **Features:**
  + Vast library of exercises tailored to different fitness levels.
  + Personalized training plans.
  + Interactive and user-friendly environment.
  + Smart recommendations and advanced filtering.
  + Real-time access to trending workouts.

**3. Architecture**

* **Component Structure:**
  + Breakdown of React components and their interactions.
* **State Management:**
  + Context API for managing global state.
* **Routing:**
  + React Router for navigation.

**4. Setup Instructions**

* **Prerequisites:**
  + Node.js, npm installed.
* **Installation:**
* npm install
* Axios

**5. Folder Structure**

* **Client:** Organized into components, pages, assets, etc.
* **Utilities:** Helper functions, custom hooks, and API services.

**6. Running the Application**

**npm start**

PS C:\web development\flexzone>npm start

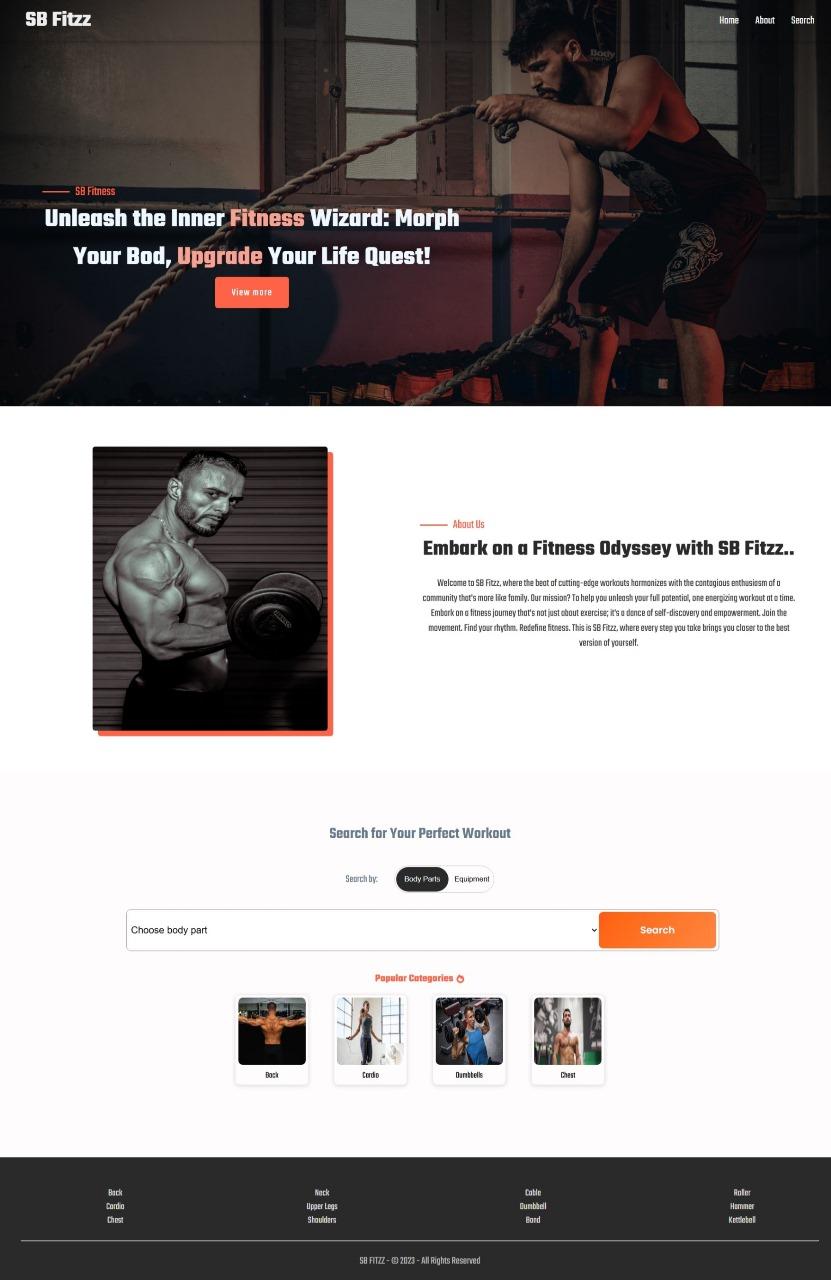
**7. Component Documentation**

* **Key Components:**
  + Exercise Library
  + Workout Planner

**8. State Management**

* **Global State:**
  + Managed using Rapid API.
* **Local State:**
  + Managed using React’s useState hook.

**9. User Interface**



**10. Styling**

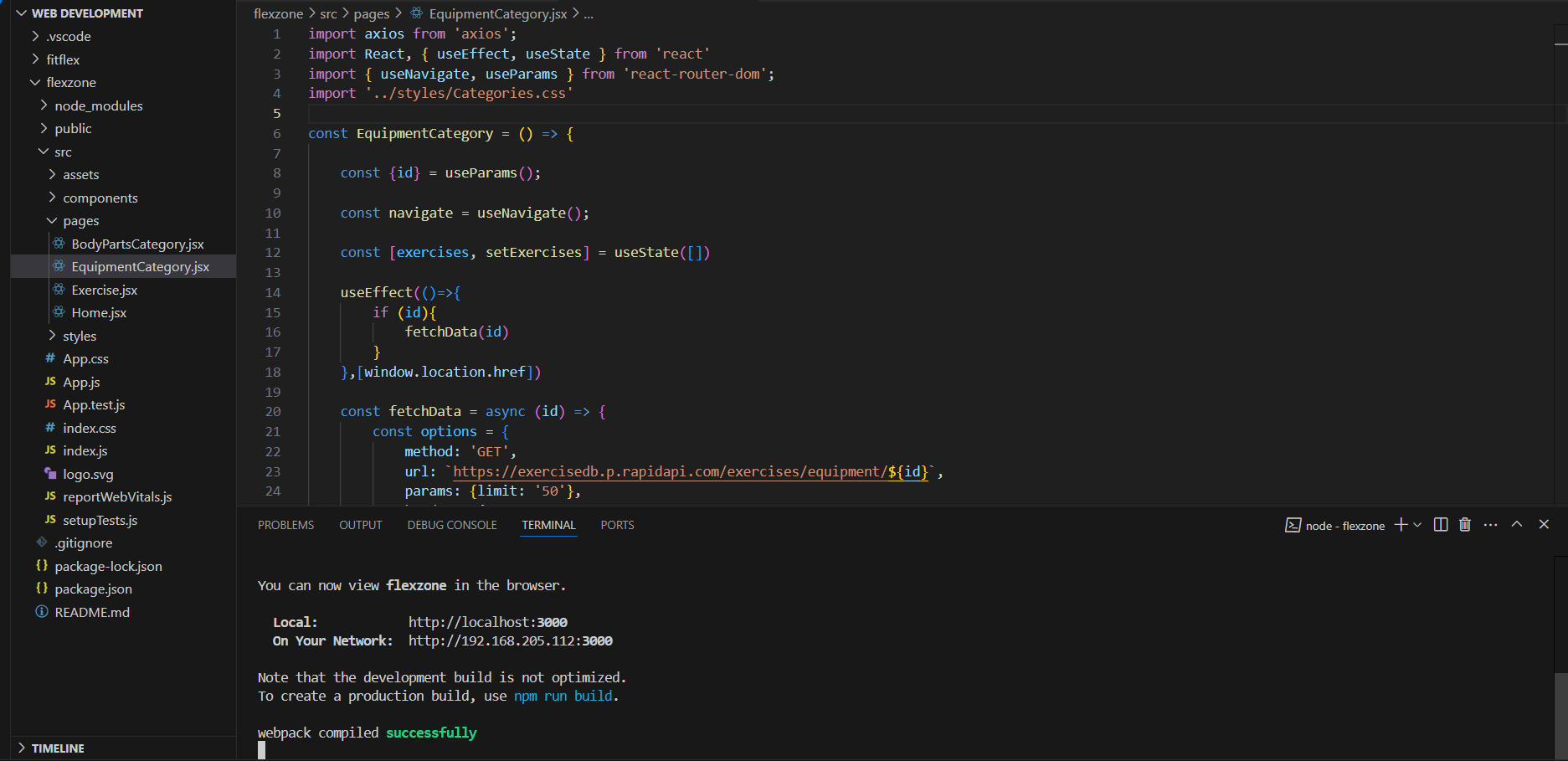
* **CSS Frameworks/Libraries:**
  + Styled-Components, Tailwind CSS.

**11. Testing**

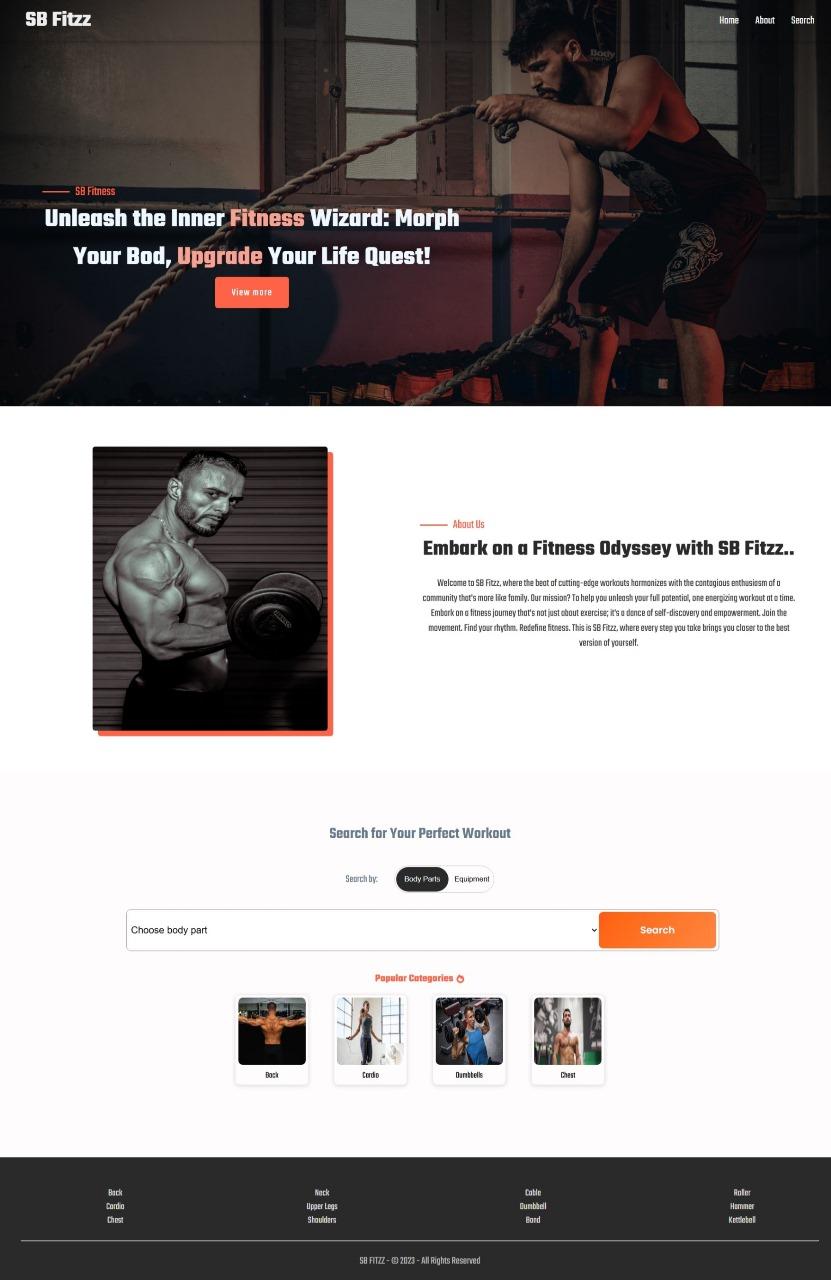
* **Testing Strategy:**
  + Unit tests using React Testing Library.
* **Code Coverage:**
  + Ensuring comprehensive component testing.

**12. Screenshots**

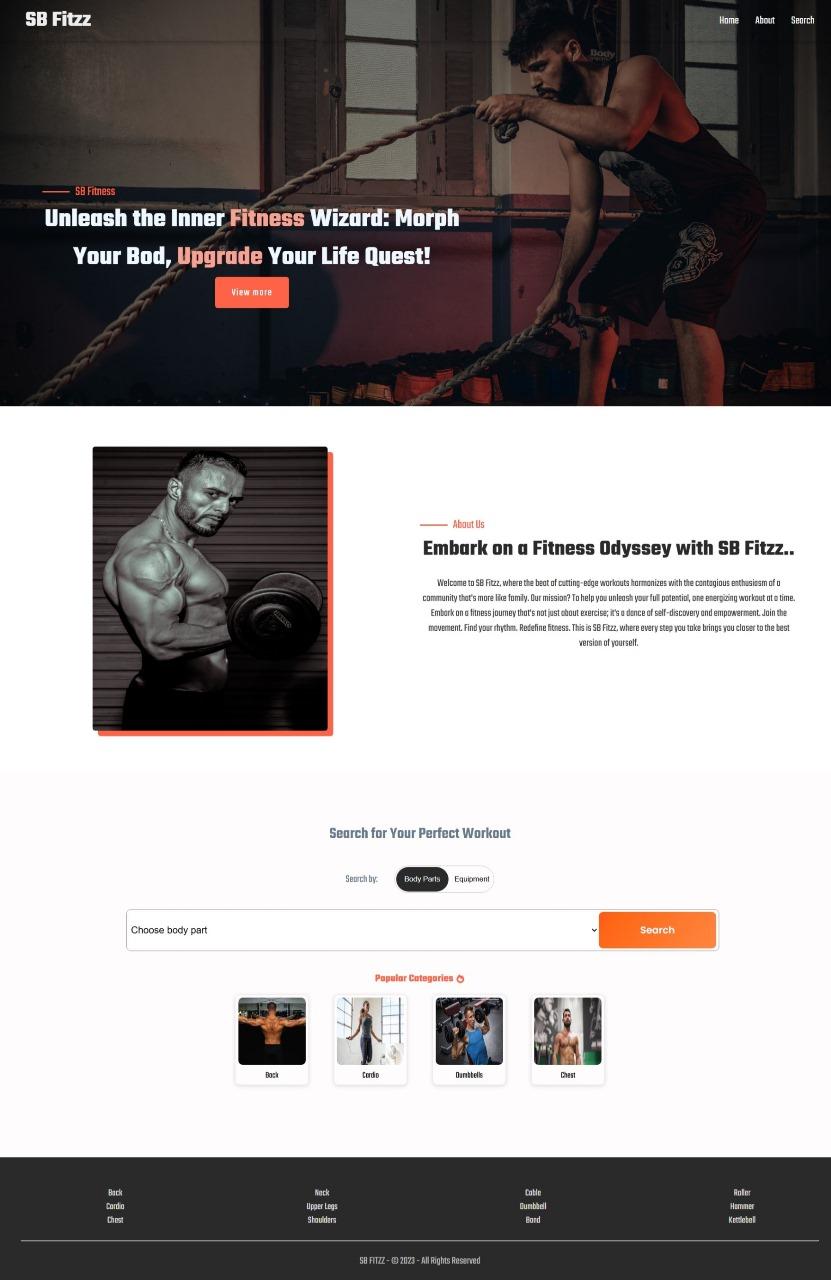
Coding:

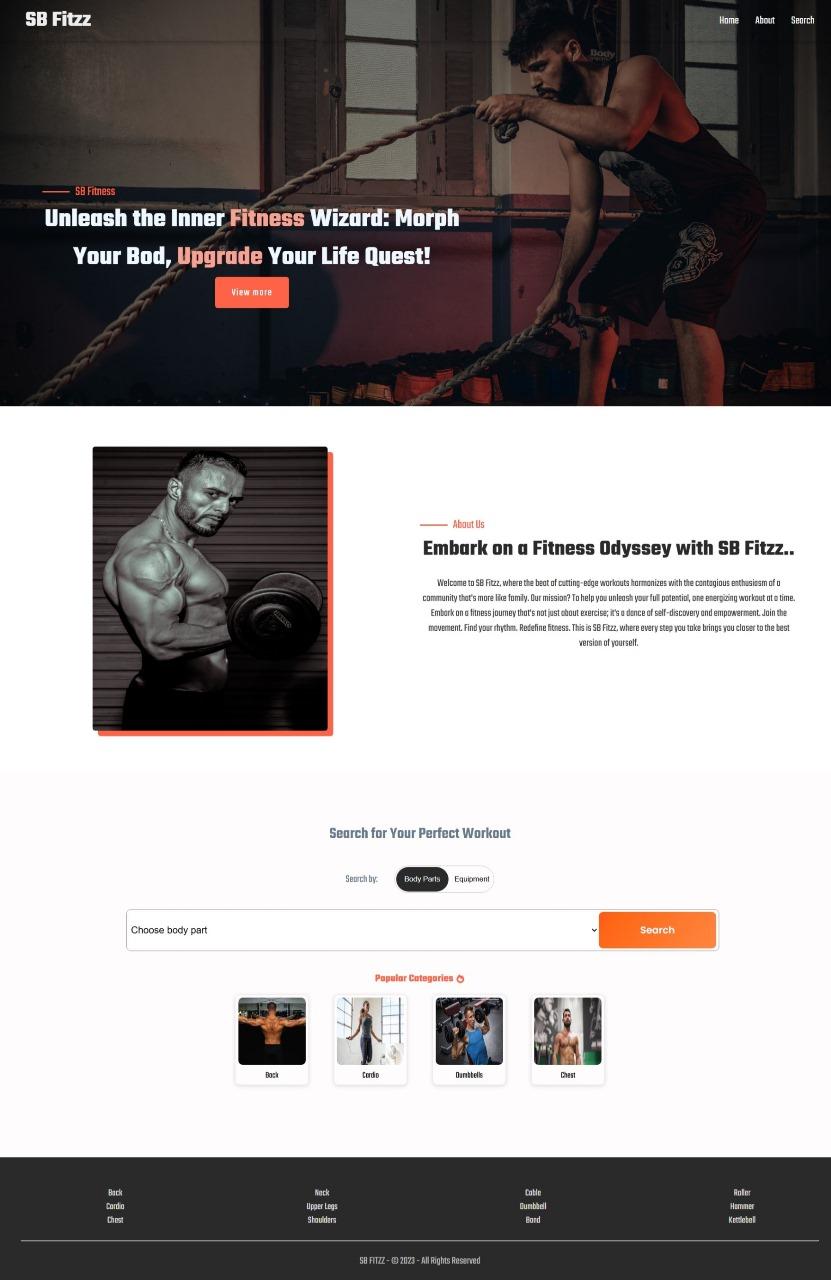


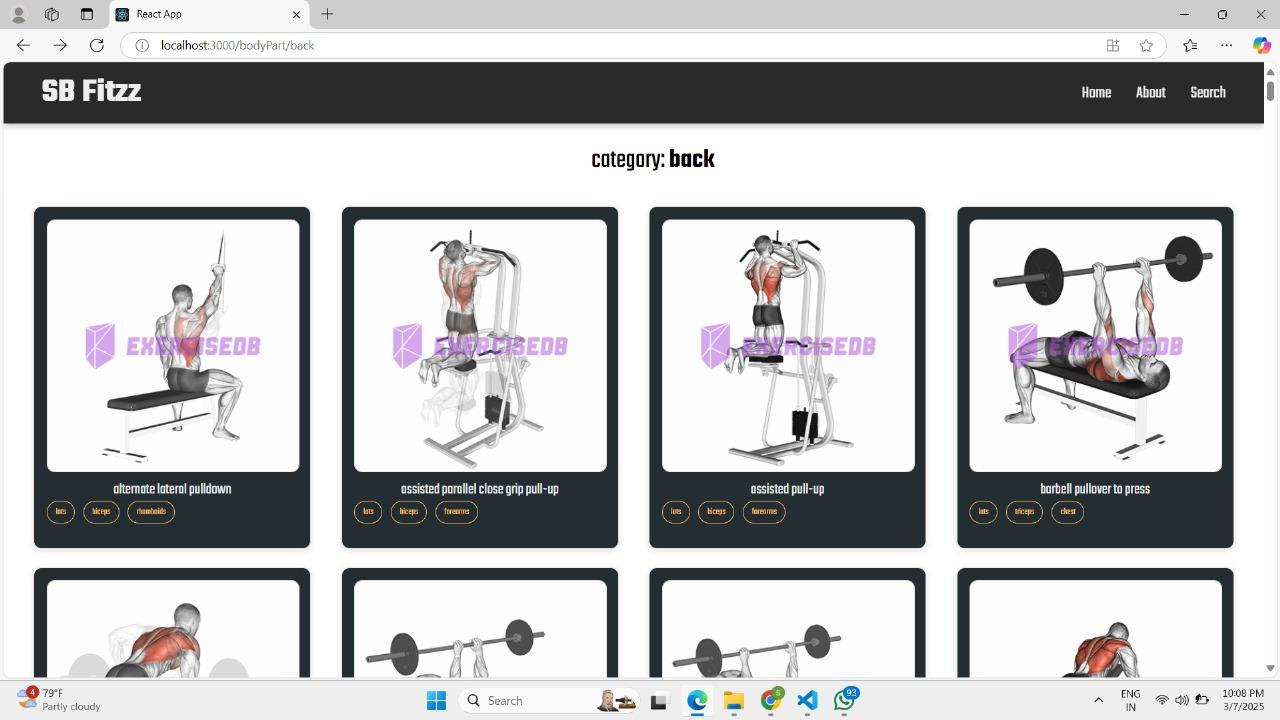
Home.jsx

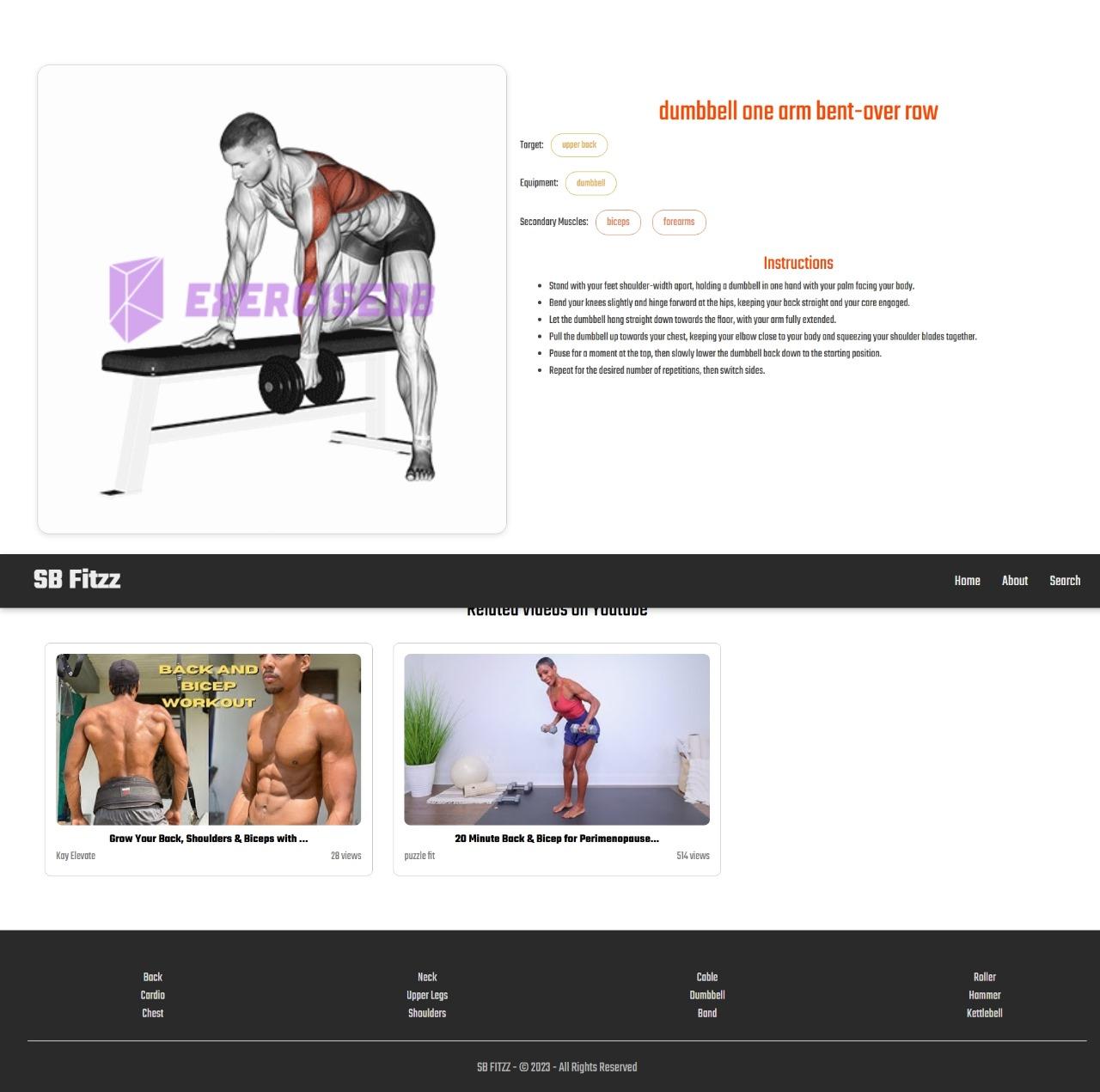
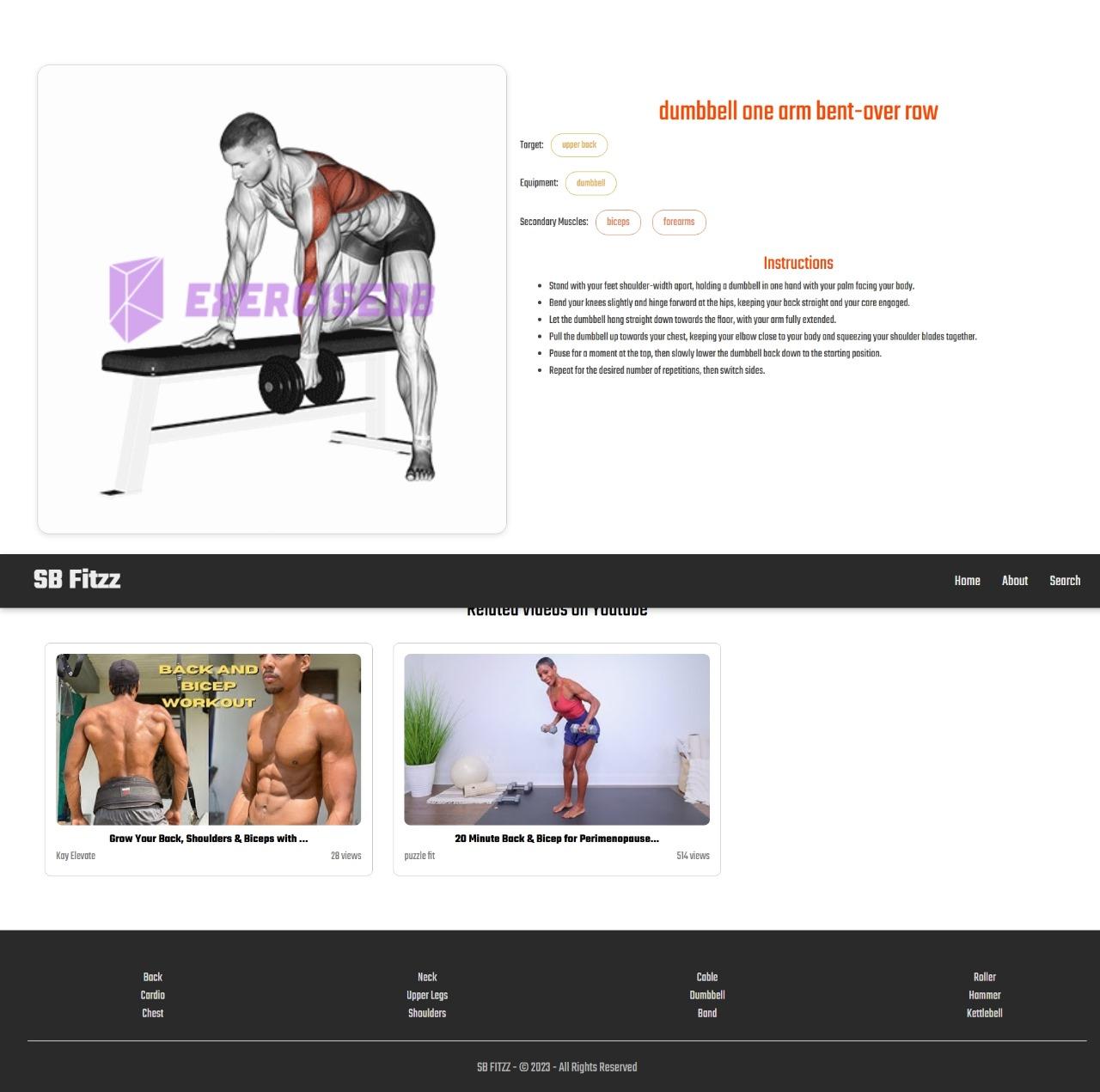


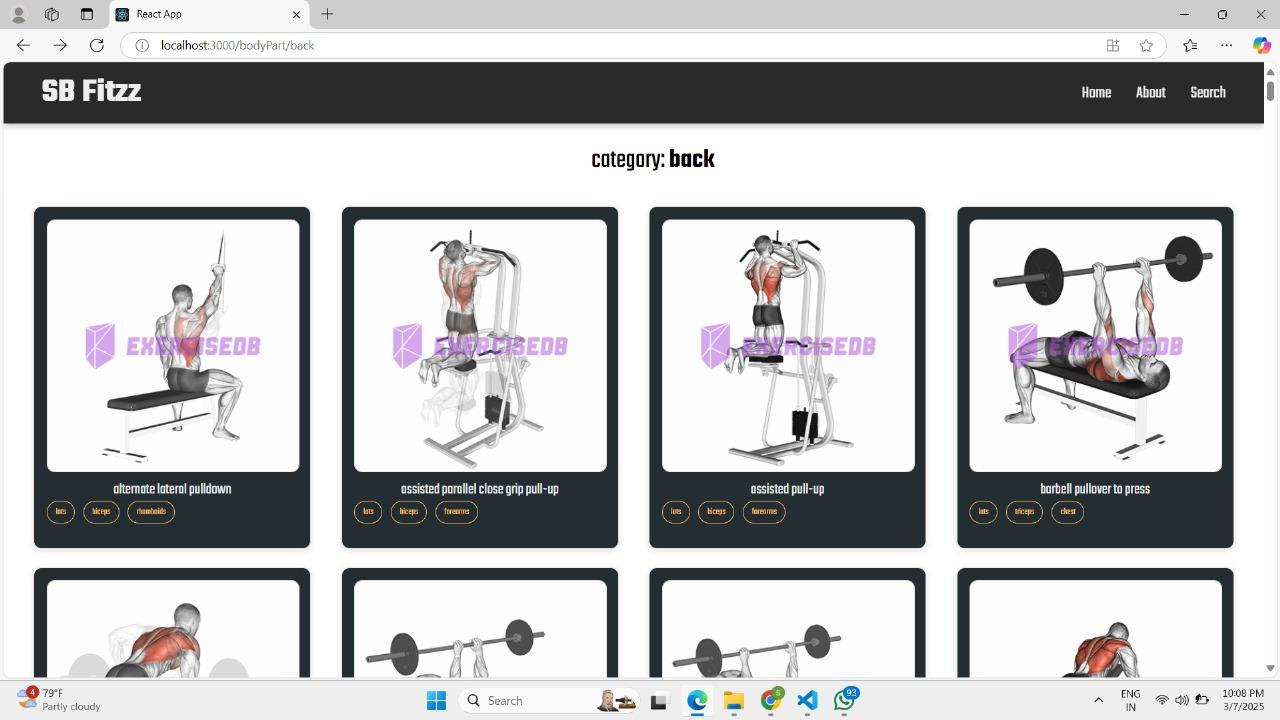
About.jsx

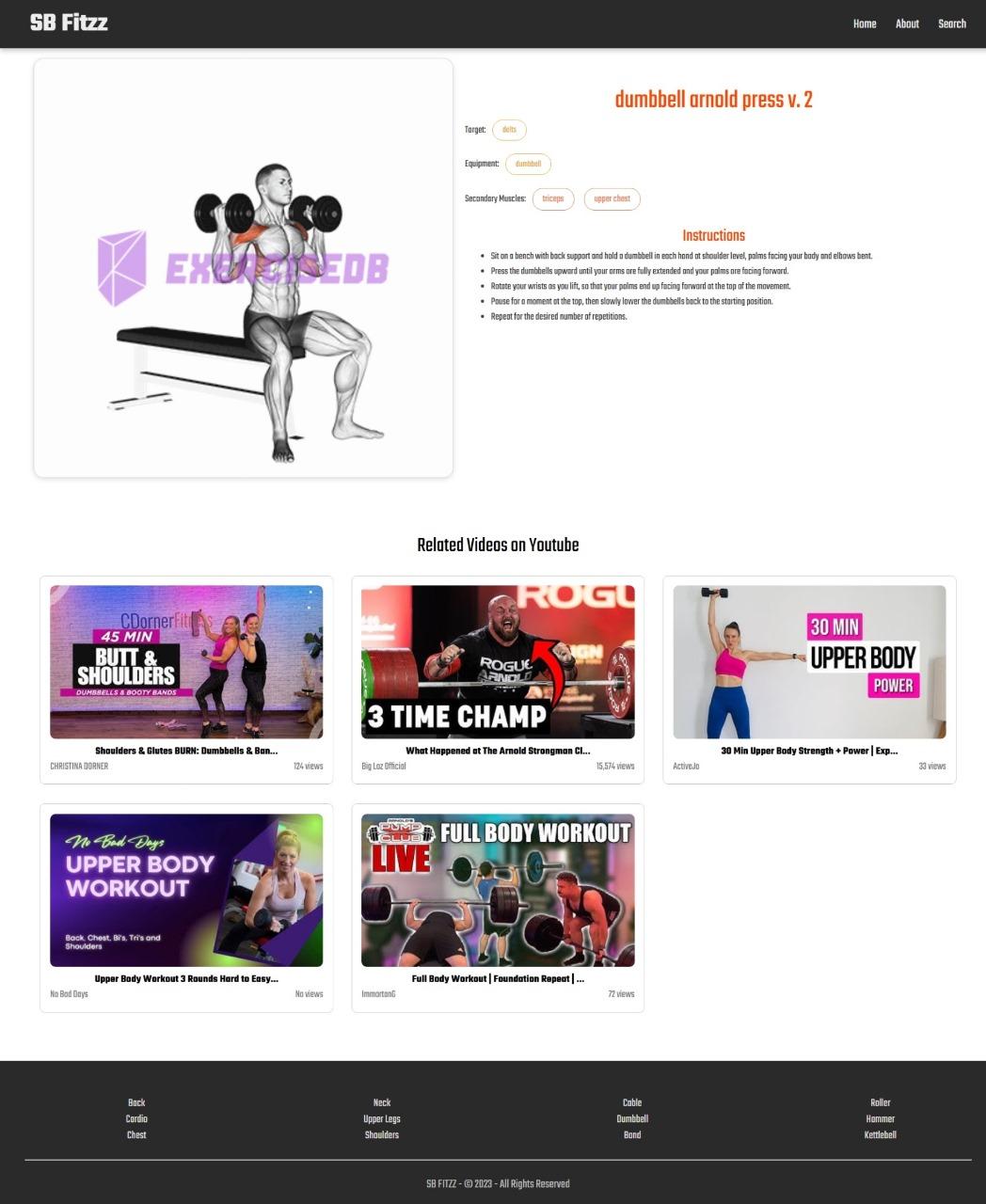


Homesearch.jsx

BodyPartsCategory.jsx

Excercise.jsx

EquipmentCategory.jsx

Exercise.jsx 

**13. Known Issues**

 **Bugs and Challenges:**

* Slow initial load time due to large exercise database.
* Performance issues in filtering and searching workouts.
* Inconsistent UI responsiveness on smaller screens.
* Occasional API request failures causing delays in loading workout data.
* Difficulty in managing state efficiently with complex workout tracking features.
* Authentication and session management inconsistencies.
* Limited error handling for failed requests.
* Lack of offline mode support.

**14. Future Enhancements**

* **Planned Features:**
  + AI-based workout recommendations.
  + Enhanced social sharing features.
  + Integration with wearable devices.