**FitFlex: Your Personal Fitness**

**Companion(React Application)**

**1. Introduction:**

* **Project Title:** FitFlex: Your Personal Fitness Companion
* **Team Members:** [List team members and their roles]

|  |  |
| --- | --- |
| Team ID | SWTID1741147298 |

|  |  |  |
| --- | --- | --- |
| Team Leader | JANANI N | [jananiraj424@gmail.com](mailto:jananiraj424@gmail.com) |
| Team Member 1 | DIVYA J | [jdivya0223@gmail.com](mailto:jdivya0223@gmail.com) |
| Team Member 2 | AARTHI M | [aarthimurugan0257@gmail.com](mailto:aarthimurugan0257@gmail.com) |
| Team Member 3 | LOSHIKA M | [loshi5139@gmail.com](mailto:loshi5139@gmail.com) |

**2. Project Overview:**

* **Purpose:** FitFlex is designed to enhance the workout experience by providing an intuitive interface, personalized fitness plans, and dynamic search functionalities. It aims to create a seamless fitness journey by offering users an easy-to-navigate platform.
* **Features:**
  + Search for exercises via Fitness API with real-time updates
  + Visual exercise exploration with high-quality images and tutorial videos
  + User-friendly design and seamless navigation across different devices
  + Advanced search functionality with filters for muscle groups, difficulty level, and equipment type

**3. Architecture:**

* **Component Structure:**
  + **Navbar Component:** Displays the navigation menu.
  + **Hero Section:** Highlights trending workouts.
  + **Search Component:** Allows users to find workouts via keyword search.
  + **Category Component:** Displays workout categories.
  + **Exercise Component:** Shows exercise details, including images/videos.
* **State Management:**
  + Utilizes **React Context API** to manage global state.
  + **Local state** handled using React’s useState hook.
* **Routing:**
  + Uses **React Router** for navigation between pages.

**4. Setup Instructions:**

* **Prerequisites:**
  + Node.js & npm installed ([Download](https://nodejs.org/en/download/))
  + Code editor (VS Code, Sublime, or WebStorm)
* **Installation:**
  + **Clone repository:**
  + git clone [repository-link]

cd fitness-app-react

* + **Install dependencies:**

npm install

* + **Run the application:**

npm start

* + Open http://localhost:3000/ in your browser.

**5. Folder Structure:**

fitness-app-react/

│── src/

│ ├── components/ # Reusable UI components

│ ├── pages/ # Different screens/pages

│ ├── styles/ # CSS styles

│ ├── utils/ # Helper functions & API calls

│── public/ # Static assets

│── package.json # Project dependencies

│── README.md # Documentation

**6. Running the Application :**

* **Command to start frontend server:**
* npm start
* Runs on http://localhost:3000

**7. Component Documentation:**

* **Key Components:**
  + Navbar.js - Navigation menu.
  + Search.js - Handles user searches.
  + Category.js - Displays workout categories.
  + ExerciseDetail.js - Displays individual exercise details.
* **Reusable Components:**
  + Button.js - Custom button component.
  + Card.js - Displays exercise cards.

**8. State Management:**

* **Global State:** Managed via React Context API, ensuring shared state across components efficiently.
* **Local State:** Managed using useState and useEffect hooks for component-specific data handling and side effects.

**9. User Interface:**

* Includes responsive design with interactive elements, ensuring a seamless user experience across different devices.
* Features include workout galleries, real-time search, and exercise detail pages.

**10. Styling:**

* **CSS Frameworks/Libraries:**
  + Tailwind CSS for styling.
  + React Icons for icons.
* **Theming:**
  + Light/dark mode toggle available.
  + Custom themes implemented using CSS variables and Tailwind configurations.

**11. Testing :**

* **Testing Strategy:**
  + Unit tests using Jest to validate individual component functionality.
  + Component testing with React Testing Library to ensure UI elements render correctly.
* **Code Coverage:**
  + Ensuring all components function as expected by maintaining high test coverage.
  + Using Istanbul/NYC for tracking coverage reports and identifying untested areas.

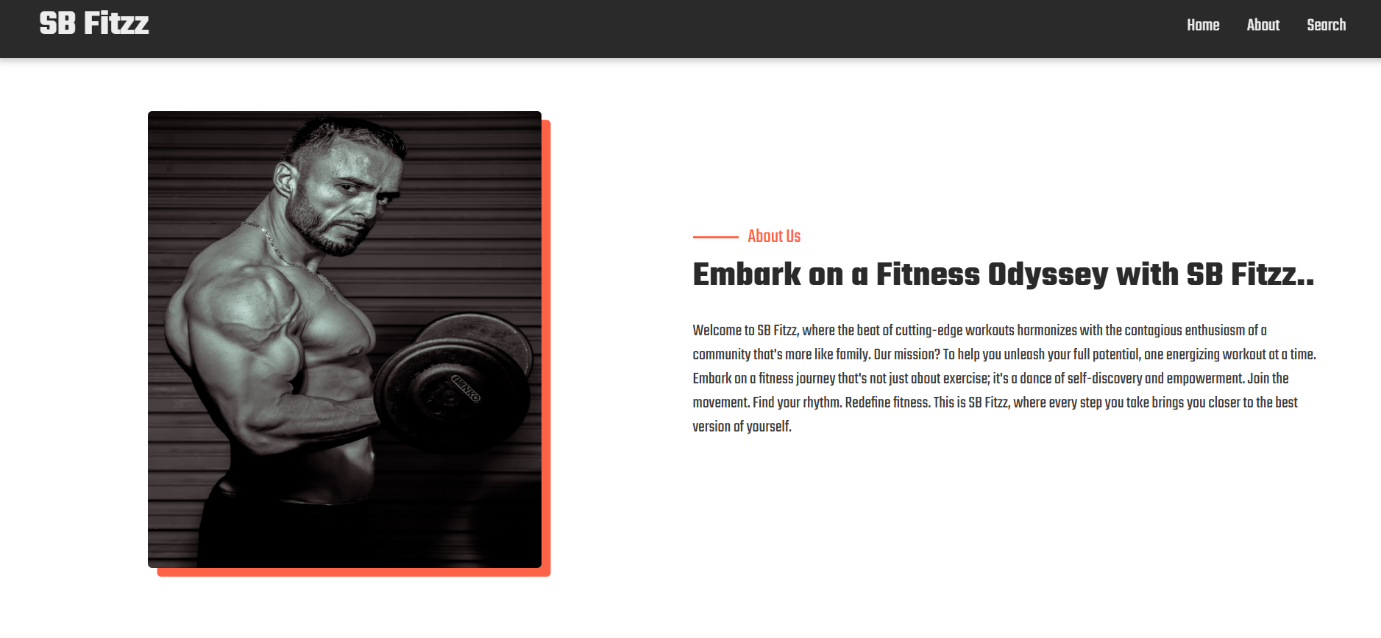
**12. Screenshots or Demo:**

* Screenshots of the homepage, search results, and exercise details.

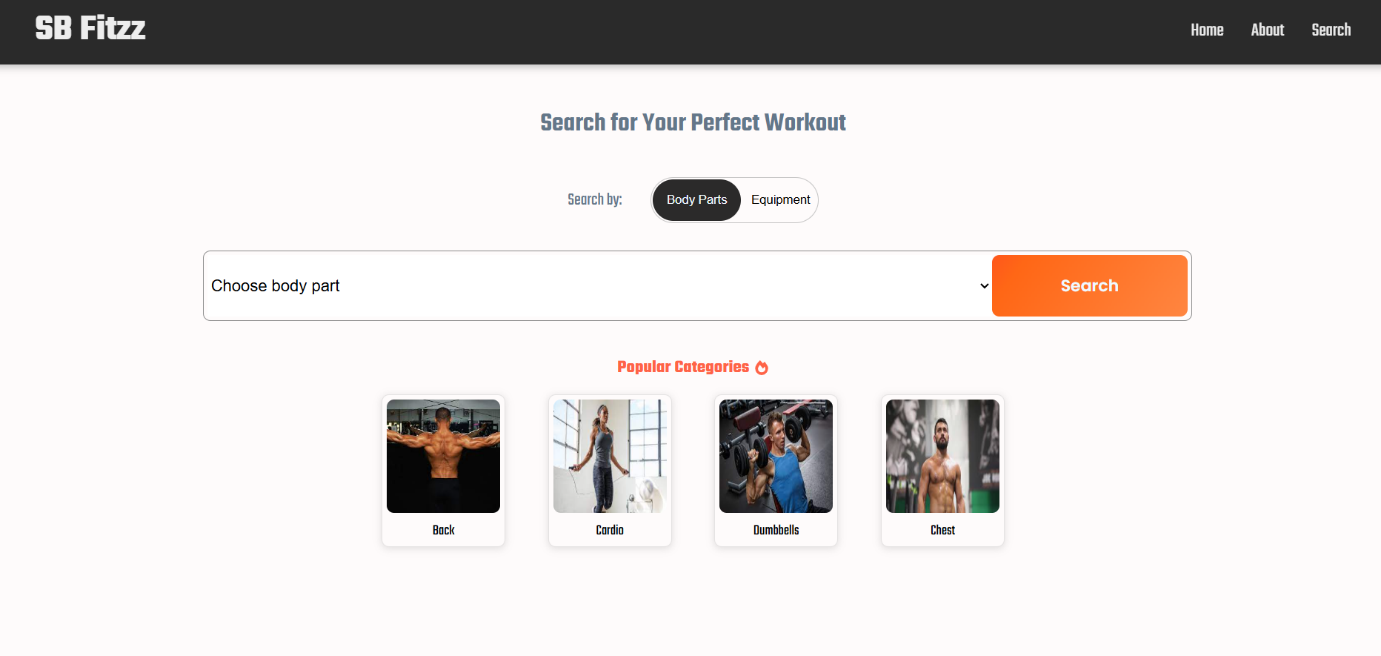
**Home page:**

****

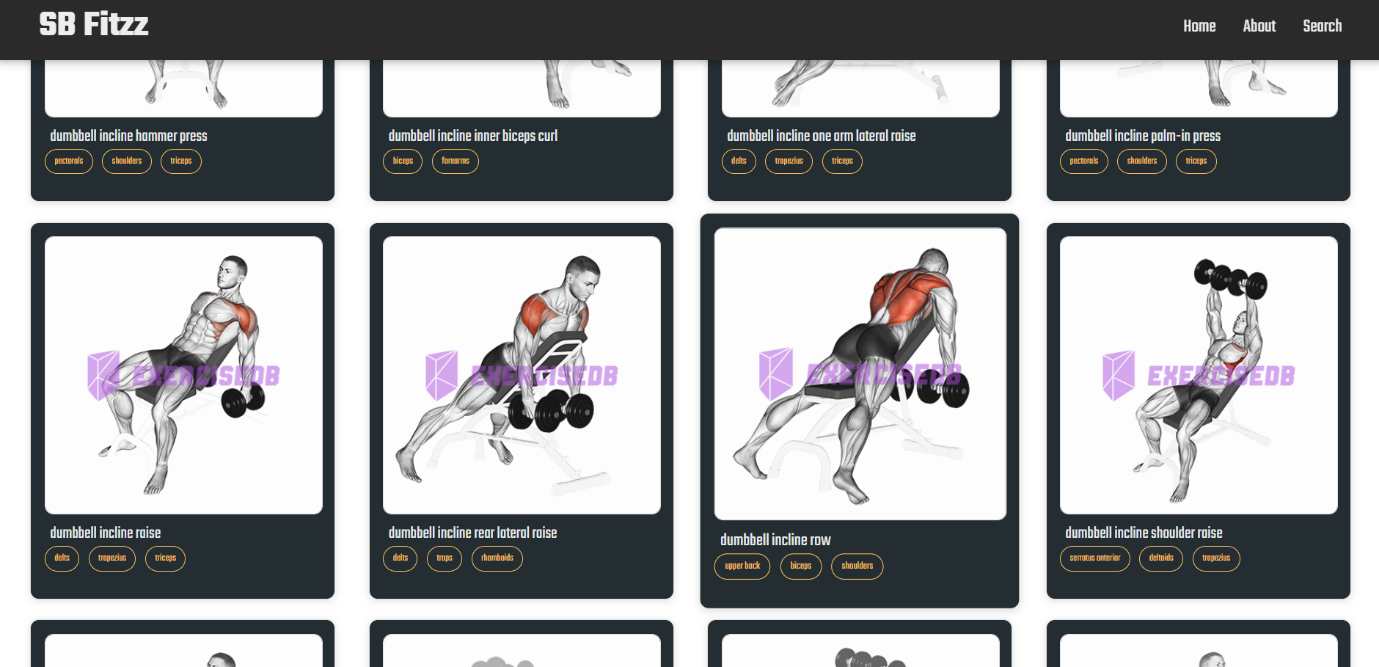
**About page:**



**Search page:**



**Search result page:**



**Demo Link:**

[**https://drive.google.com/file/d/1tDSD87HWDeCEobb2hl1xdIA7CDd\_06l9/view?usp=drivesdk**](https://drive.google.com/file/d/1tDSD87HWDeCEobb2hl1xdIA7CDd_06l9/view?usp=drivesdk)

**13. Known Issues:**

* API rate limits may affect search performance, potentially leading to temporary unavailability of certain exercises.

**14. Future Enhancements :**

* Implement user authentication to allow personalized user experiences and data storage.
* Add workout tracking features, enabling users to log their exercises, track progress, and set fitness goals.
* Expand language support to cater to a global audience.