

Frontend Development with React.js

Project Documentation

Fitness Website Frontend Documentation

Introduction

- **Project Title:** SB Fitzz
- **Team Members:** The user prompt does not list team members.

Project Overview

- **Purpose:** The purpose of this project is to create a dynamic, user-friendly fitness website. It aims to empower users to find their perfect workout and "unleash the inner fitness wizard."
- **Features:** Key features include a homepage with a motivational banner, an "About Us" section detailing the project's mission, and a search page where users can find workouts by body part or equipment. The UI is sleek and modern, with a dark theme.

Architecture

- **Component Structure:** The application seems to be structured with distinct components for

different sections of the website. For example, there's likely a `Homepage` component, an `About` component, and a `Search` component. These components would be composed of smaller, reusable components like a `Navbar`, a `SearchForm`, and `WorkoutCard` components.

- **State Management:** Based on the simplicity of the features shown, it's likely that state management is handled using React's built-in `useState` hook for local component state, and potentially the Context API for managing global state like the search query or user authentication. There's no evidence of more complex state management libraries like Redux.
- **Routing:** The navigation bar with `Home`, `About`, and `Search` links, along with the changing URL in the screenshots, indicates that `react-router-dom` is being used to handle navigation and display different pages of the application.

Setup Instructions

- **Prerequisites:** You'll need Node.js and npm (Node Package Manager) installed on your system.
- **Installation:**
 1. Clone the repository from the source (not provided).

2. Navigate to the `client` directory: `cd client`
3. Install the dependencies: `npm install`

Folder Structure

- **Client:** The `client` folder is likely the root of the React application. It probably contains:
 - `src/components`: For reusable UI elements like `Navbar`, `SearchForm`, and `WorkoutCard`.
 - `src/pages`: For top-level page components like `HomePage`, `AboutPage`, and `SearchPage`.
 - `src/assets`: For images and other static assets.
 - `public`: The public-facing folder for the main `index.html`.
- **Utilities:** There may be a `src/utls` folder for helper functions, such as those that fetch data from an API or format workout information.

Running the Application

- Provide commands to start the frontend server locally.
 - **Frontend:** Navigate to the `client` directory and run: `npm start`
-

Component Documentation

- **Key Components:**

- **Navbar:** Renders the top navigation bar with links to **Home**, **About**, and **Search**.
- **HomePage:** Displays the main landing page with the hero banner and a call-to-action button. * **AboutPage:** Renders the "About Us" section, providing an overview of the project's mission. * **SearchPage:** Contains the search functionality, allowing users to filter workouts by body part or equipment. * **WorkoutCard:** A reusable component to display popular workout categories like "Back," "Cardio," and "Dumbbells."

- **Reusable Components:**

- The **Navbar** is likely a reusable component used across all pages.
- The **WorkoutCard** component, which displays the popular categories, is designed to be reusable for showcasing different workout types.

State Management

- **Global State:** If used, the global state would likely manage the current search query, which would be

accessed by both the search form and the results display.

- **Local State:** Local state within components would handle things like form input values in the `SearchForm` component and the toggling between search by "Body Parts" or "Equipment."

User Interface

- **Homepage:** A full-width banner featuring a person performing an intense workout, with a title that reads "Unleash the Inner Fitness Wizard."
- **About Page:** A clean layout with an image on the left and a description of the project on the right.
- **Search Page:** A simple interface with a dropdown menu to select a body part, a search button, and a section for popular categories displayed as clickable cards.

Styling

- **CSS Frameworks/Libraries:** Based on the clean, modern look and consistent design, it's possible the project uses a CSS framework like Tailwind CSS or Bootstrap, or a component library like Styled Components for scoped styling.

- **Theming:** The website uses a dark theme with a black background and high-contrast orange and white text, which gives it a slick, professional aesthetic.
-

Testing

- **Testing Strategy:** The provided images don't contain information about testing. However, a standard React project would use Jest and React Testing Library for unit and integration tests.
 - **Code Coverage:** There are no screenshots or information on code coverage.
-

Screenshots or Demo

- The uploaded images themselves serve as the screenshots for the homepage, about page, and search page.
-

Known Issues

- The provided images don't show any known bugs or issues.
-

Future Enhancements

- **Adding a search results page to display workouts based on the user's query.**
- **Implementing user authentication to allow users to save favorite workouts.**
- **Integrating a third-party API to populate the workout data dynamically.**
- **Adding animations and transitions to improve the user experience.**
- **Enhancing the UI with more components like a detailed workout view page.**