

# PROJECT DOCUMENTATION

# FITFLEX-Your Personal Fitness Companion

## 1. Introduction

- **Project Title:** FITFLEX-Your Personal Fitness Companion
- **Team ID:** NM2025TMID31192
- **Team Leader:** Ganapathi P – ganapathipandi60@gmail.com
- **Team Members:**
  - GOKULKANTH S – gokulkanth2007@gmail.com
  - GOPI K – kanagarajgobi2007@gmail.com
  - GUNA M – guna42961@gmail.com
  - GUNASEELAN R – gunaguna1023@gmail.com

## 2. Project Overview

- **Purpose:**  
FITFLEX is a fitness discovery platform built using **React.js**. It helps users explore workout routines categorized by **body parts** and **equipment**, with embedded YouTube tutorials and instructions.
- **Goals:**
  - Provide an intuitive and modern UI.
  - Fetch exercises dynamically from APIs.
  - Allow smooth navigation between pages and exercises.
- **Key Features:**
  - Search workouts by body parts or equipment.
  - View exercise details with instructions.
  - Watch related YouTube videos.
  - Responsive design for all devices.

## 3. Architecture

### Component Structure

- App.js — Root component, sets up routes.
- Navbar — Provides navigation and search.
- Hero — Landing section with background media.
- HomeSearch — Search workouts.
- Category Pages — Body parts & equipment filters.
- Exercise — Full detail view of a workout.
- Footer — App footer.

### State Management

- **Local State:** Managed using React useState and useEffect.
- **API Integration:** Axios used for fetching data from Fitness API & YouTube API.

### Routing

- Library: **react-router-dom**
- Routes:
  - o / → Home.jsx
  - o /category/bodyparts → BodyPartsCategory.jsx
  - o /category/equipment → EquipmentCategory.jsx
  - o /exercise/:id → Exercise.jsx

## 4. Setup Instructions

### Prerequisites

#### Node.js & npm

- Node.js is required to run React applications.
- npm (Node Package Manager) is used to install dependencies.
- [Download Node.js](#)

#### React.js

- React is the main JavaScript library used to build this project.
- If you don't have an existing React app, create one using:  
npx create-react-app my-app

- `cd my-app`
- `npm start`
- In SB Fitzz, the React app is already created, so you just need to install dependencies (`npm install`).

## Git

- Used for cloning and version control.
- [Download Git](#)

## Code Editor

- Recommended: **Visual Studio Code (VS Code)**
- [Download VS Code](#)

## Basic Knowledge

- HTML, CSS, JavaScript
- React concepts (components, props, hooks, state, routing)

## Installation

- **Get the code:**

- Download the code from the drive link given below:

[https://drive.google.com/drive/folders/14f9eBQ5W7VrLdPhP2W6PzOU\\_HCy8UMex?usp=sharing](https://drive.google.com/drive/folders/14f9eBQ5W7VrLdPhP2W6PzOU_HCy8UMex?usp=sharing)

### Install Dependencies:

- Navigate into the cloned repository directory and install libraries:

```
cd fitness-app-react  
npm install
```

- **Start the Development Server:**

- To start the development server, execute the following command:

```
npm start
```

### Access the App:

- Open your web browser and navigate to <http://localhost:3000>.

- You should see the application's homepage, indicating that the installation and setup were successful.

## Environment Variables

Create a .env file with:

```
REACT_APP_API_URL=<https://exercisedb.p.rapidapi.com/exercises/equipmentList>  
REACT_APP_YOUTUBE_API_KEY=<33cf3a7616msh4c3b1e3204f24e2p1294b3jsne16a7323d732>
```

## 5. Folder Structure

## › FITNESS APP

- › node\_modules
- › public
- › src
  - › assets
  - › components
  - › pages
  - › styles
- # App.css
- JS App.js
- JS App.test.js
- # index.css
- JS index.js
- img logo.svg
- JS reportWebVitals.js
- JS setupTests.js
- ◆ .gitignore
- { } package-lock.json
- { } package.json
- ⓘ README.md

```
▽ src
  > assets
  ▽ components
    ❁ About.jsx
    ❁ Footer.jsx
    ❁ Hero.jsx
    ❁ HomeSearch.jsx
    ❁ Navbar.jsx
  ▽ pages
    ❁ BodyPartsCategory.jsx
    ❁ EquipmentCategory.jsx
    ❁ Exercise.jsx
    ❁ Home.jsx
  ▽ styles
    # About.css
    # Categories.css
    # Exercise.css
    # Footer.css
    # Hero.css
    # Home.css
    # HomeSearch.css
    # Navbar.css
```

## 6. Running the Application

- **Start development server:**
- npm start
- **Build for production:**
- npm run build
- **Run tests:**
- npm test

## 7. Component Documentation

### Key Components

- **Navbar.jsx** – Top navigation with app links.
- **Hero.jsx** – Intro section with video background.
- **HomeSearch.jsx** – Allows searching workouts.
- **Footer.jsx** – Contains links and copyright.
- **About.jsx** – App description section.

### Pages

- **Home.jsx** – Displays Hero, About, and categories.
- **BodyPartsCategory.jsx** – Shows workouts filtered by body part.
- **EquipmentCategory.jsx** – Shows workouts filtered by equipment.
- **Exercise.jsx** – Displays instructions, exercise details, and YouTube videos.

## 8. State Management

- **Local State:**
  - Search queries stored in HomeSearch.
  - API data fetched and stored per-page.
- **Global State:** Not implemented – app uses component-level state.

## 9. User Interface

- **Pages include:**
  - Home (Hero + Search + About)

SB Fitzz

SB Fitness

**Unleash the Inner Fitness**

**Wizard: Morph Your Bod,  
Upgrade Your Life Quest!**

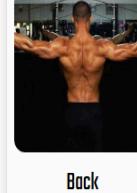
[View more](#)

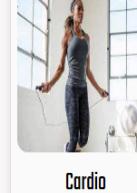
Search for Your Perfect Workout

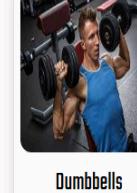
Search by: [Body Parts](#) [Equipment](#)

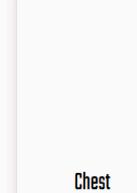
Choose body part [▼](#) [Search](#)

**Popular Categories** ⓘ

 Back

 Cardio

 Dumbbells

 Chest



### jack burpee

Target: [cardiovascular system](#)

Equipment: [body weight](#)

Secondary Muscles: [quadriceps](#) [hamstrings](#) [calves](#) [shoulders](#) [triceps](#) [core](#)

**Instructions**

- Start in a standing position with your feet shoulder-width apart.
- Lower your body into a squat position, placing your hands on the ground in front of you.
- Kick your feet back, landing in a push-up position.
- Perform a push-up, lowering your chest to the ground and then pushing back up.
- Jump your feet forward, landing in a squat position.
- Jump up explosively, reaching your arms overhead.
- Land softly and immediately lower back into the squat position to begin the next repetition.

Related Videos on YouTube

## 10. Styling

- **Frameworks Used:** Tailwind CSS / Bootstrap.
- **Custom CSS:** Stored in `src/styles/`.
- Each page/component has a dedicated CSS file for modularity.

## 11. Testing

- **Libraries Used:** Jest, React Testing Library.
- **Unit Tests:** Written in `App.test.js`.
- **Setup:** Configured with `setupTests.js`.

## 12. Screenshots / Demo

- **Demo Link:**

[https://drive.google.com/file/d/1mUEZRmCsz84WIFIsNe5cZjCSjzttT0\\_m/view  
?usp=sharing](https://drive.google.com/file/d/1mUEZRmCsz84WIFIsNe5cZjCSjzttT0_m/view?usp=sharing)

- Screenshot:



## 13. Known Issues

- API rate-limit may cause some exercises not to load.

- YouTube API sometimes fails to fetch related videos.

## 14. Future Enhancements

- Add login and user profile.
- Save favorite workouts.
- Add workout progress tracking.
- Implement offline caching with service workers.
- Enhance animations and transitions.