

Fit Flex: your personal Fitness companion

1. introduction

**Project Title: Fit Flex**

**Team Leader: Rekha S**

**Team Members: Yuva Sri M,**

**Akshaya G**

**Rekha Dharshini S**

2. project overview

**Purpose: Fit Flex is a web-based fitness platform designed to help users improve**

**their health and lifestyle. It offers guided workouts, premium equipment**

**recommendations, personalized plans, and expert support**.

**Features:**

**Wide exercise library categorized by body part and type (arms, chest, cardio,**

**abs/core, etc.)**

**Free, Premium, and Elite subscription plans with increasing benefits**

**Premium equipment showcases with filtering options (equipment vs bodyweight)**

**Progress tracking and analytics (in higher-tier plans)**

**Responsive design for seamless use across devices**

3. architecture

**Component Structure: Organized into multiple React pages — Home, About,**

**Exercises, Equipment, Premium. Uses reusable components such as Navbar,**

**Footer, Exercise Cards, and plan Cards.**

**State Management: Primarily managed using React hooks (use State, use Effect).**

**Scalable for Context API or Redux if global state management is required in the**

**future.**

**Routing: Implemented using React Router for smooth navigation between sections.**

4. setup instructions

**Prerequisites: Node.js (LTS version), npm or yarn.**

**Installation:**

**1. Clone the repository:**

**git clone https://github.com/rekhasaravanan2000-oss/fitflex.git**

**cd fit flex**



**2. Install dependencies:**

**npm install**

**3. Start the application:**

**npm start**

5. Folder structure

**Client: Contains src/ with components/, pages/, and assets/ (images, CSS).**

**Utilities: Functions for search filters, exercise handling, and subscription logic.**

6. running the application

This launches the app in **Frontend:**

**Run the following command inside the project root directory:**

**npm start**

**development mode (default: http://localhost:3000).**

7. component documentation

**Key Components:**

**Navbar: Handles site-wide navigation**

**Footer: Contains quick links, support info, and policies**

**Exercise Card: Displays individual workouts with category and filtering support**

**Plan Card: Renders subscription details for Free, Premium, and Elite tiers**

**Category Filter: Allows users to filter exercises by category/type**

**Reusable Components: Buttons, Cards, and Containers are shared across multiple**

**pages.**

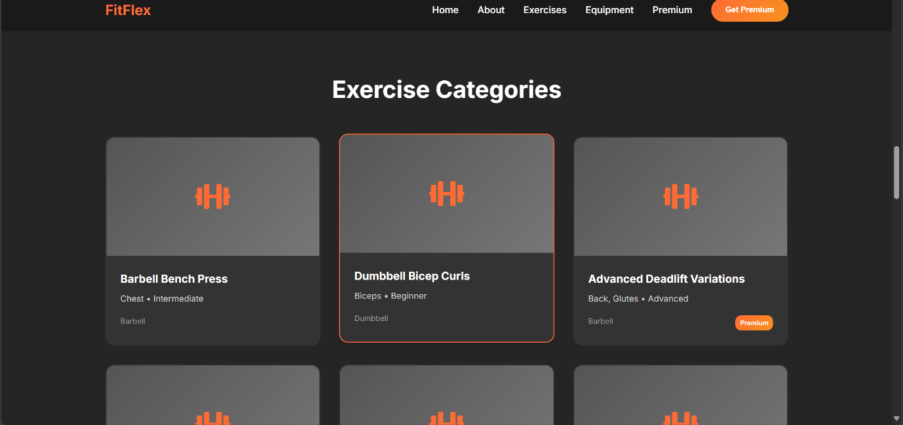
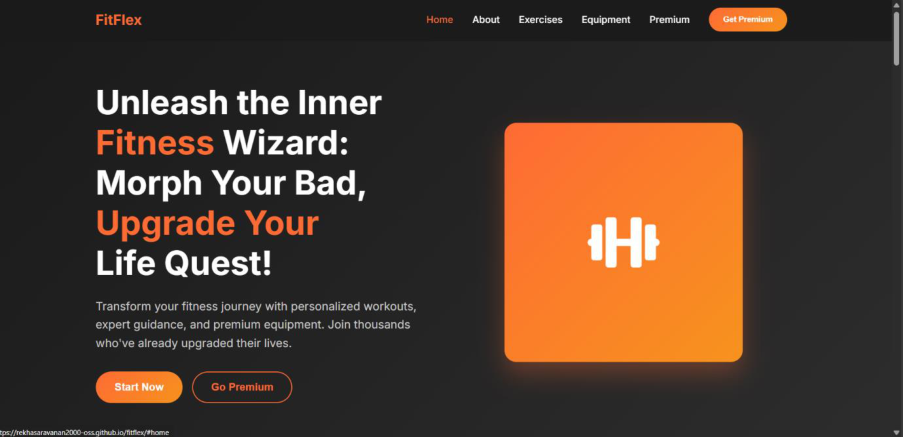
8. state management

**Global State: Currently not in use, but future integration possible with Context API or**

**Redux.**

**Local State: use State is used for category filtering, subscription plan selection, and**

**UI toggles.**

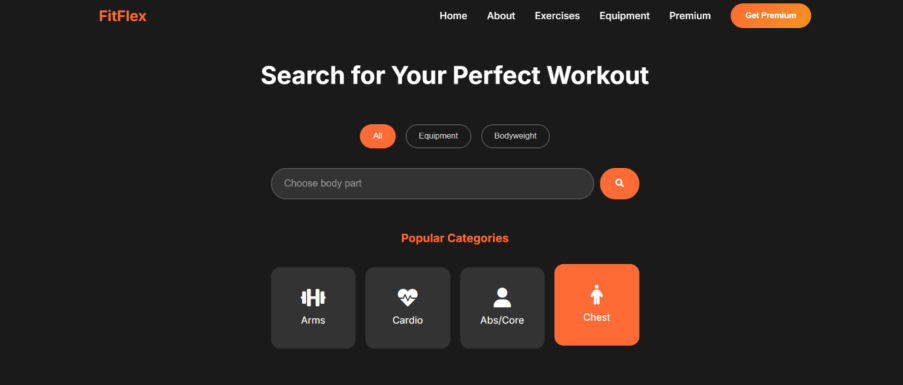
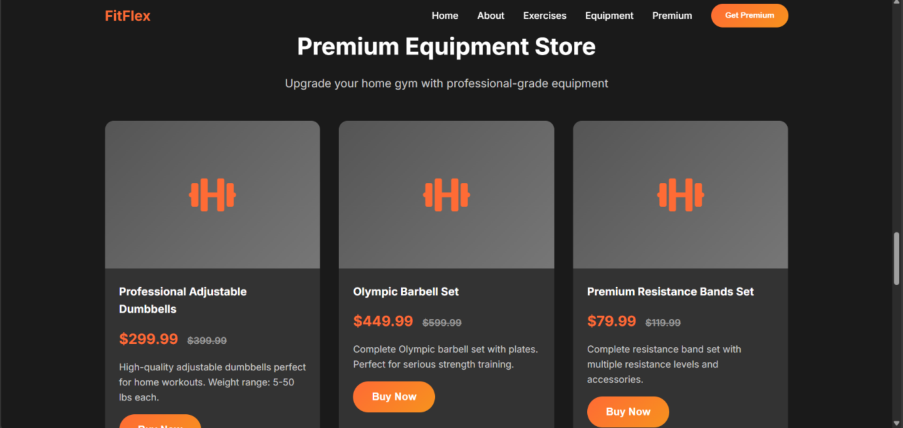


9. user interFace

**Homepage: Hero section with motivational tagline and CTA buttons (Start Now, Go**

**Premium)**.

**Exercises: Interactive library of workouts categorized by type.**



**Equipment: Premium store showcasing fitness equipment with filter options.**

**Plans: Subscription cards showing Free, Premium, and Elite tiers.**

**Support: Footer with links to FAQs, Privacy Policy, Terms of Service.**



10. styling

**CSS Frameworks/Libraries: Pure CSS with responsive design techniques. No heavy**   
**frameworks (e.g., Bootstrap, Tailwind) currently used.**

**Theming: Bright gradient backgrounds (yellow** →**orange), clean cards with**   
**shadows, and modern typography.**

11. testing

**Strategy:**

**Manual testing of navigation, rendering of components, and responsiveness across**

**devices.**

**Ensures plans, filters, and UI links work as expected.**

**Future improvements: Add unit tests and integration tests using Jest + React Testing**   
**Library.**

drive link:

**https://drive.google.com/file/d/17Q3MhnPpHXF1ewoIyK6hWfPolPcas0ro/view?usp=d**

**rivesdk**

**THANK YOU…!!!**