

# Frontend Development with React.js

## Project Documentation

### 1. Introduction

Project Title: FitFlex: Your Fitness Companion

Team ID: (NM2025TMID39050)

Team Leader:

Meena K(Mail ID: 24bsccs31@cttewc.edu.in)

Team Members:

Alexandra B(Mail ID: 24bsccs20@cttewc.edu.in)

Santhiya S (Mail ID: 24bsccs05@cttewc.edu.in)

Monisha T (Mail ID: 24bsccs35@cttewc.edu.in)

### 2. Project Overview

**Purpose:**FitFlex is designed to serve as an accessible fitness platform for individuals passionate about exercise and holistic well-being. It combines ease of use with powerful features to make fitness exploration enjoyable and effective.

**Objectives:**

- **User-Friendly Experience:** Create an intuitive interface for effortless navigation and personalized discovery of fitness routines.
- **Comprehensive Exercise Management:** Provide tools for organizing, saving, and managing workout plans with advanced search capabilities.
- **Technology Stack:** Utilize React.js and modern web technologies for an efficient, responsive, and interactive user experience.

**Features:**

- **Exercises from Fitness API:** Access a wide range of exercises from reliable fitness APIs across multiple categories.
- **Visual Exercise Exploration:** Browse workouts visually through curated image galleries.
- **Intuitive and User-Friendly Design:** Clean, modern interface for smooth navigation.
- **Advanced Search Feature:** Quickly find exercises or workout plans tailored to individual goals.

### 3. Architecture

**Component Structure:-** Exercise List Component- Exercise Detail Component- Search Component- Categories/Filters Component- Favorites/Save Routine Component- User Profile Component

**State Management:**Context API / Redux to manage exercises, user preferences, and saved routines.

**Routing:**React Router for navigation between exercise list, search results, and user profile pages.

## 4. Setup Instructions

Prerequisites: Node.js, npm/yarn, API key for fitness API.

Installation:

- Clone the repository.
- Navigate to the project folder.
- Run `npm install` to install dependencies.
- Configure API keys if required.
- Run `npm start` to launch the application.

## 5. Folder Structure

Client: Contains React components, pages, and assets.

Services/Utils: API service functions, helper utilities.

Context/Store: Global state management files.

## 6. Running the Application

Command: npm start

## 7. Component Documentation

Key Components: Exercise List, Exercise Detail, Search Bar, Categories, Favorites.

Reusable Components: Card, Button, Input, Modal.

## 8. State Management

Global State: Exercises, User Preferences, Saved Routines.

Local State: UI interactions (modals, search input).

## 9. User Interface

Pages:- Home Page (overview + categories).- Exercise List Page (search and filter results).- Exercise Detail Page (images + instructions).- Favorites Page (saved routines).- Profile Page (user preferences).

(Screenshots can be added later when the UI is ready.)

## 10. Styling

CSS Frameworks/Libraries: Tailwind CSS or Material UI.

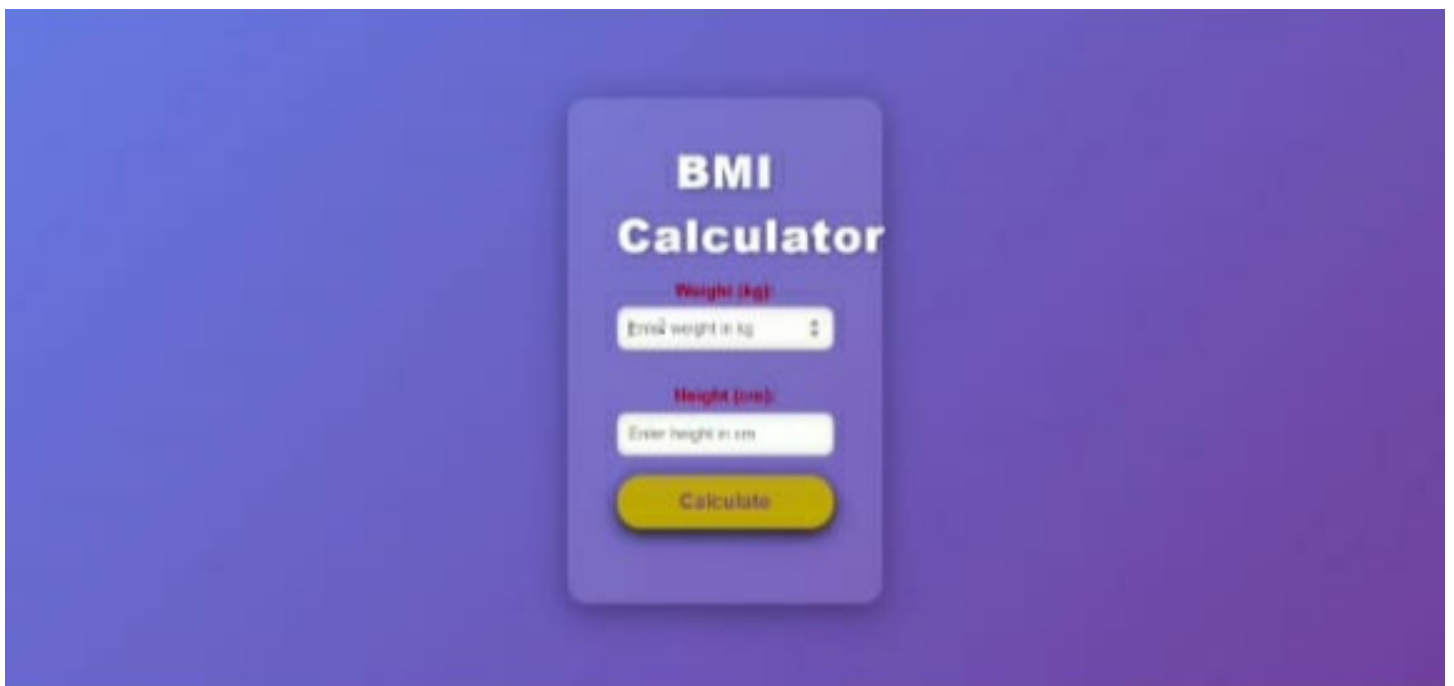
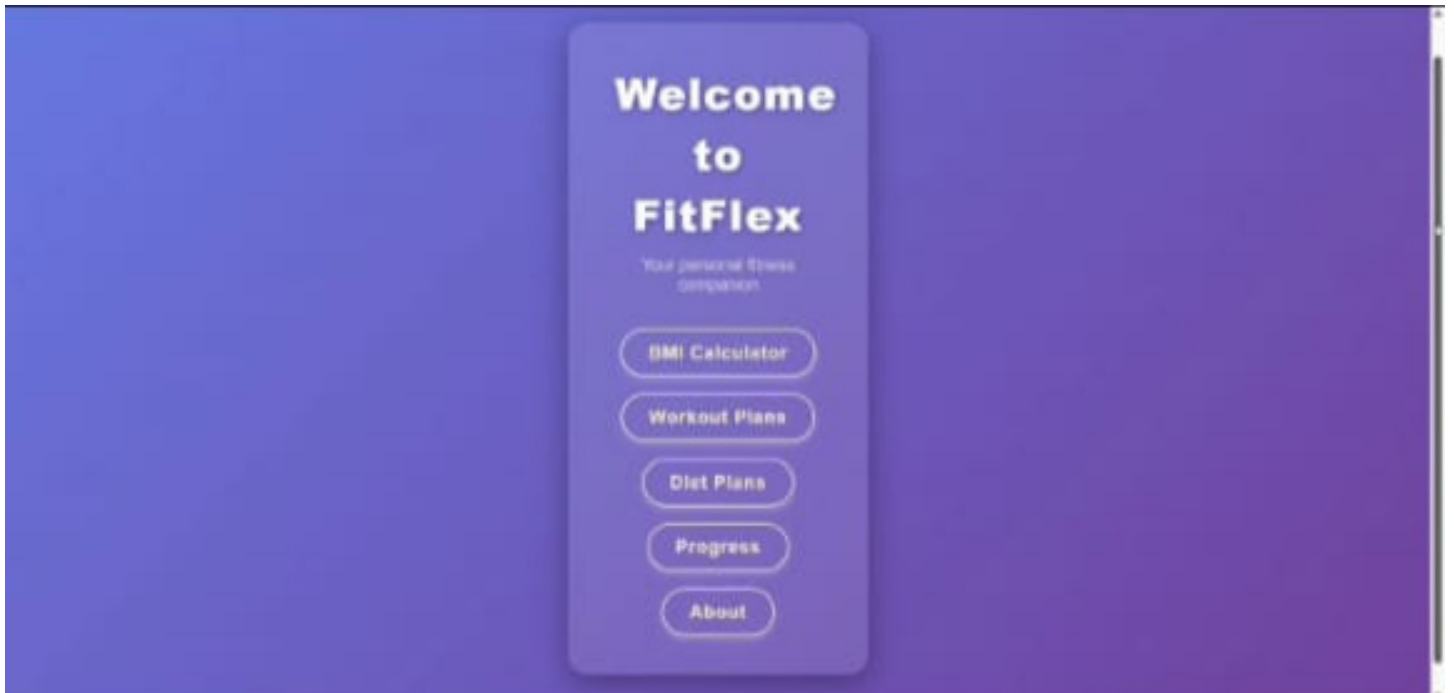
Design: Clean, modern, responsive design with mobile-first approach.

## 11. Testing

Testing Strategy:- Unit testing (exercise components).- Integration testing (search + exercise detail).- End-to-end testing (navigation flow).

Tools:Jest, React Testing Library.

## 12. Screenshots or Demo



## Workout Plans

🏃 Beginner  
Workout - 30 mins  
daily

🔥 Fat Burn HIIT -  
20 mins

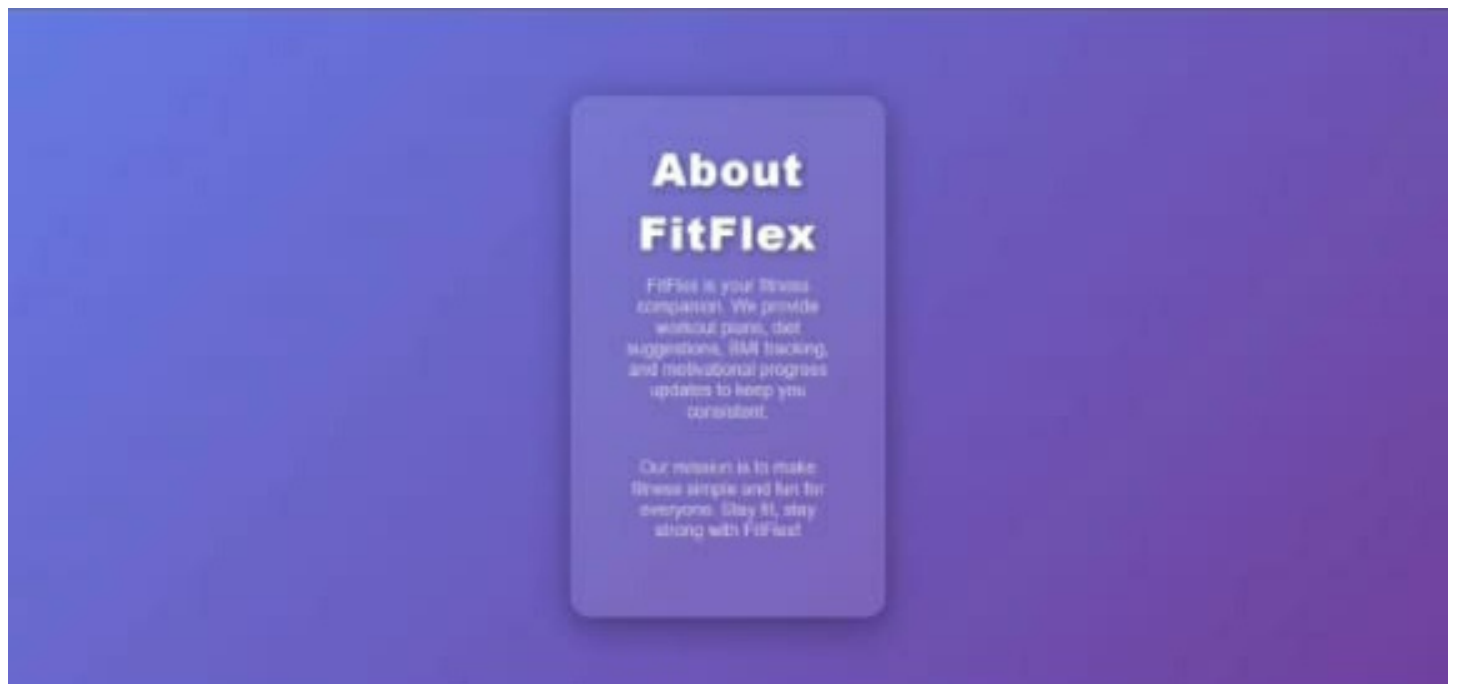
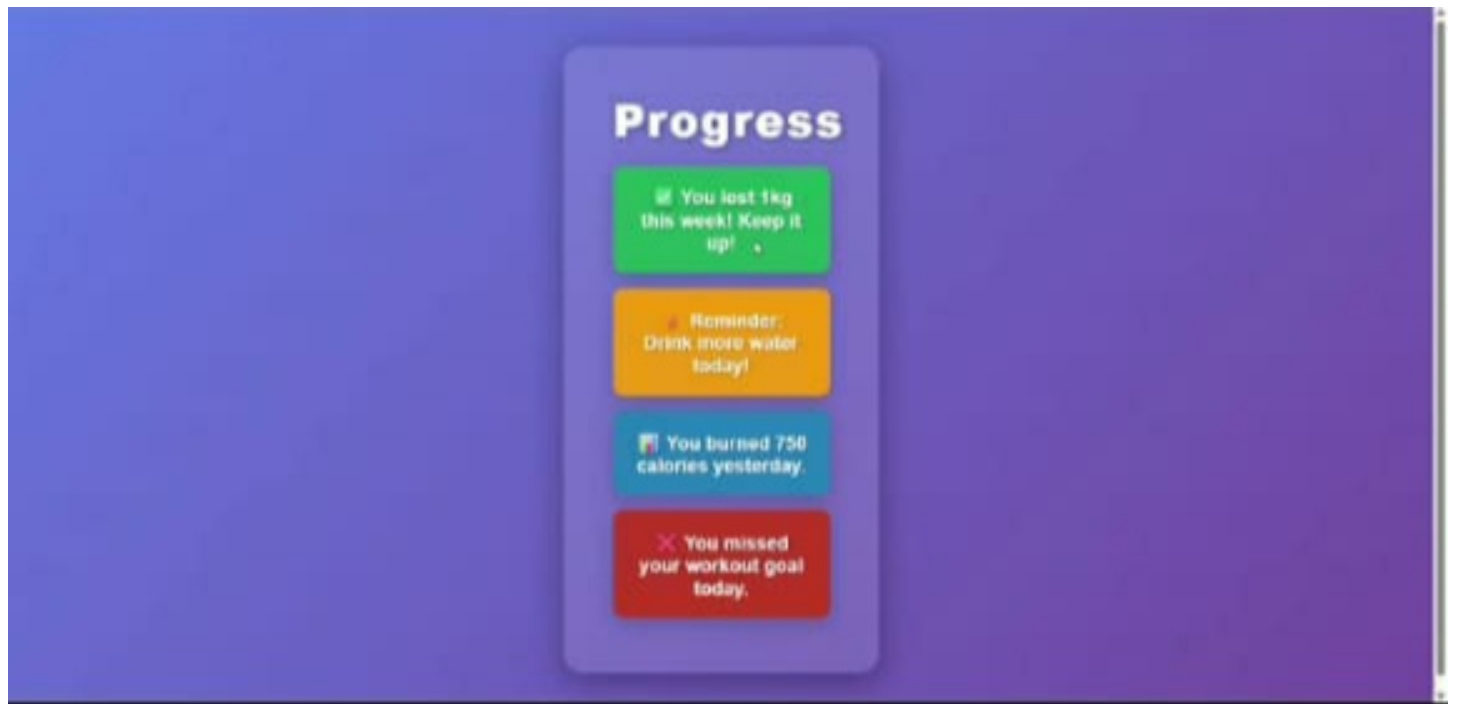
💪 Strength  
Training - 45 mins

## Diet Plans

⚖️ Weight Loss  
Diet - Low carbs,  
high protein

💪 Muscle Gain  
Diet - High  
protein, balanced  
carbs

🍎 Healthy  
Lifestyle Diet -  
Fruits, veggies,  
grains



### 13. Known Issues

Placeholder: Limited offline functionality.

Placeholder: Dependency on external API availability.

### 14. Future Enhancements

- Integration of meal plans and diet tracking.
- Personalized workout recommendations based on user profile.
- AI-based progress tracking.

- Social sharing of workout routines.