**EPROJECT FITNESS TRACKER**

## ****1. Introduction****

The demand for skilled professionals in the IT industry has grown significantly. However, busy schedules and lack of practical exposure often hinder the learning process. To address this gap, Aptech’s **eProject** initiative offers a hands-on, laddered learning platform that allows students to apply theoretical concepts in a realistic and synchronous environment.

This documentation outlines the design and implementation of a **Fitness Tracker Application**, a real-world MERN-based (MongoDB, Express.js, React.js, Node.js) project. The app allows users to track fitness metrics including workouts, nutrition, and physical progress in an interactive and user-friendly manner.

## ****2. Objectives****

* Apply full-stack development knowledge in building a real-world application.
* Create an integrated and scalable fitness tracking platform.
* Enhance understanding of databases, APIs, UI/UX, and authentication.
* Provide a structured approach to modular application development.
* Gain experience in building RESTful APIs and single-page applications (SPA).
* Improve analytical, testing, and debugging skills.
* Create developer- and user-focused documentation and support tools.

## ****3. Problem Statement****

Fitness enthusiasts face challenges managing their workouts, diet, and overall health progress due to the lack of a centralized platform that is personalized, interactive, and comprehensive. Most apps in the market either lack specific features or are too generic.

### ****Need:****

An intuitive and feature-rich application that enables users to:

* Create and manage workout plans.
* Log and monitor food intake.
* Track progress with visual charts.
* Stay motivated with reminders and goal-setting features.

## ****4. Background and Market Insight****

The global fitness tech industry has witnessed exponential growth. With increased focus on health and wellness, digital tools that track fitness activities are highly sought after. Modern users expect:

* Personalized experiences
* Real-time progress visualization
* Seamless cross-device access

The **Fitness Tracker App** addresses these needs and provides a scalable foundation for future enhancements like social sharing, wearable integration, and AI-based coaching.

## ****5. System Architecture Overview****

### ****Architecture Style:****

* **Three-tier Architecture**
  1. **Presentation Layer** – React.js
  2. **Application Layer** – Node.js + Express.js
  3. **Data Layer** – MongoDB

### ****Technology Stack:****

| **Layer** | **Technology** |
| --- | --- |
| Frontend | MaterialUI |
| Backend | Node.js, Express.js |
| Database | MongoDB (Mongoose ODM) |
| Authentication | JSON Web Tokens (JWT), bcrypt |
| API Testing | Postman |
| Hosting | Vercel (Frontend), Render/Heroku (Backend), MongoDB Atlas |

## ****6. Functional Requirements****

### ****6.1 User Management****

* **Register/Login** using email and password.
* JWT-based authentication.
* Password hashing using bcrypt.
* Profile update with profile picture upload (via Cloudinary or local storage).

### ****6.2 Workout Management****

* Create/edit/delete workout routines.
* Add exercises with:
  + Name, category (e.g., cardio, strength)
  + Reps, sets, weights, notes
* Save templates for reusable workout plans.

### ****6.3 Nutrition Tracking****

* Log meals by type (breakfast, lunch, dinner, snacks).
* Input food items with calorie and macronutrient values.
* Track daily consumption against goals.

### ****6.4 Progress Tracking****

* Enter metrics like weight, BMI, body measurements.
* Record performance metrics like running time, lifting weights.
* Show progress through:
  + Bar/line charts (using Chart.js)
  + Weekly/monthly comparison

### ****6.5 User Dashboard****

* Overview of:
  + Today's workouts
  + Daily calorie intake
  + Latest progress logs
* Motivational stats and reminders

### ****6.6 Search and Filter****

* Filter entries by date, type, or keyword.
* Sort workouts by most recent, duration, or frequency.

### ****6.7 Reports and Export****

* Export progress reports in PDF or CSV format.
* Summary reports of weekly/monthly activity.

### ****6.8 Notifications and Alerts****

* Custom reminders for workouts and meals.
* Achievement badges when goals are reached.

### ****6.9 Preferences and Settings****

* Theme (dark/light mode)
* Units (metric/imperial)
* Time zone and notification preferences

## ****7. Non-Functional Requirements****

| **Category** | **Description** |
| --- | --- |
| **Performance** | Load time < 2 seconds; 100 concurrent users |
| **Security** | Data encryption, secure authentication |
| **Privacy** | GDPR-compliant data handling |
| **Reliability** | 99% uptime, regular backups |
| **Usability** | Responsive design, mobile-first UI |
| **Scalability** | Designed for horizontal scaling |
| **Accessibility** | Adheres to WCAG 2.1 AA standards |
| **Testing** | Unit tests (Jest), integration tests (Supertest) |
| **Monitoring** | Logs (Winston), Error tracking (Sentry) |

## ****8. REST API Structure (Sample)****

| **Method** | **Endpoint** | **Description** |
| --- | --- | --- |
| POST | /api/auth/register | Register a new user |
| POST | /api/auth/login | User login |
| GET | /api/workouts | Get user workouts |
| POST | /api/workouts | Add workout |
| PUT | /api/workouts/:id | Update workout |
| DELETE | /api/workouts/:id | Delete workout |
| GET | /api/nutrition | Get nutrition logs |
| POST | /api/nutrition | Add nutrition log |
| GET | /api/progress | Get progress data |
| POST | /api/progress | Add progress log |

## ****9. Testing Strategy****

* **Unit Tests** for functions and controllers using **Jest**
* **Integration Tests** for API endpoints using **Supertest**
* **UI Testing** with React Testing Library
* **Manual QA** for usability testing on multiple devices
* **Security Testing** using OWASP checklist

## ****10. Deployment & Hosting****

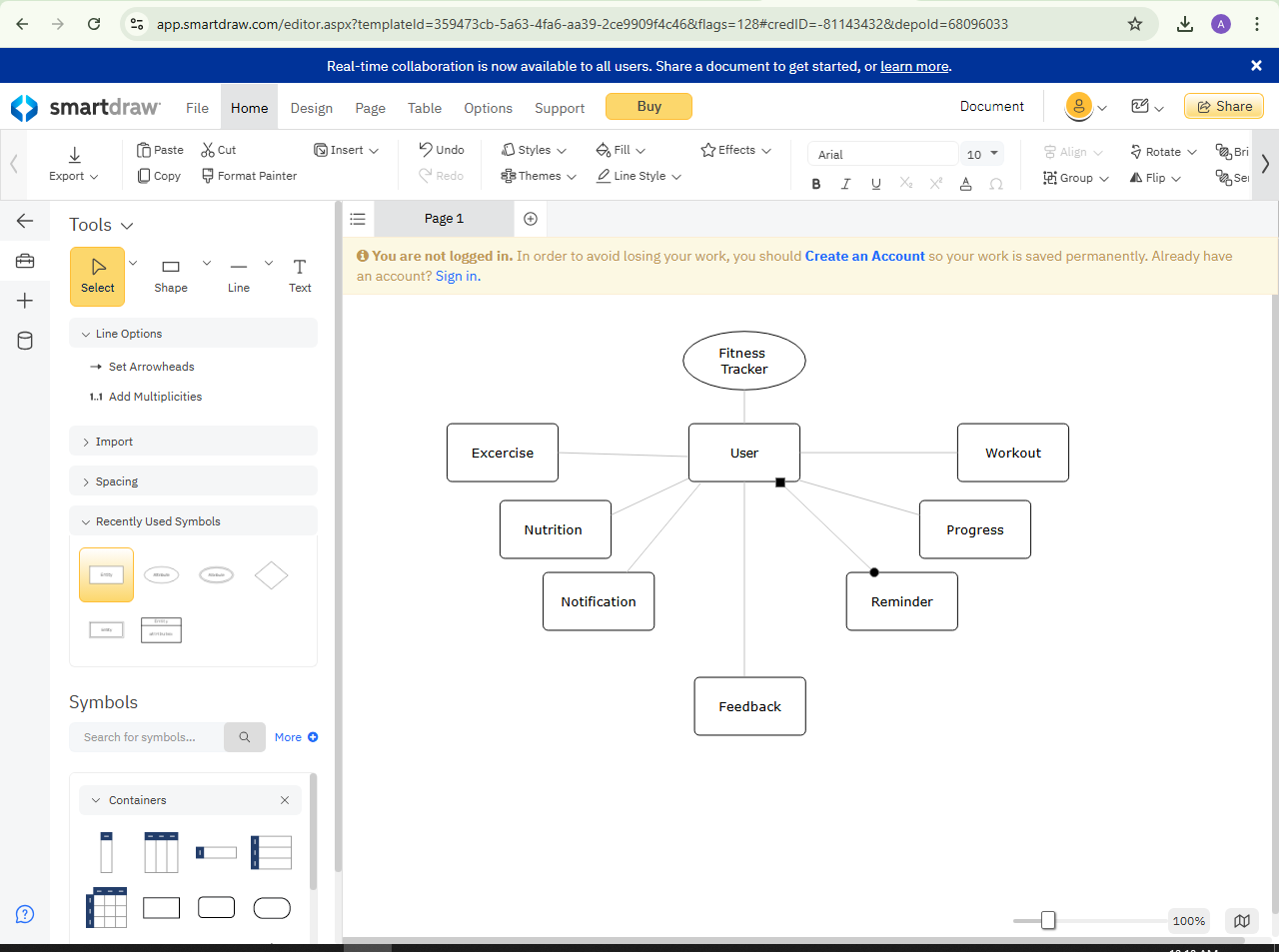
| **Component** | **Platform** |
| --- | --- |
| Frontend | MaterialUI |
| Backend API | Render / Heroku |
| Database | MongoDB Atlas |

## ****11. Future Scope****

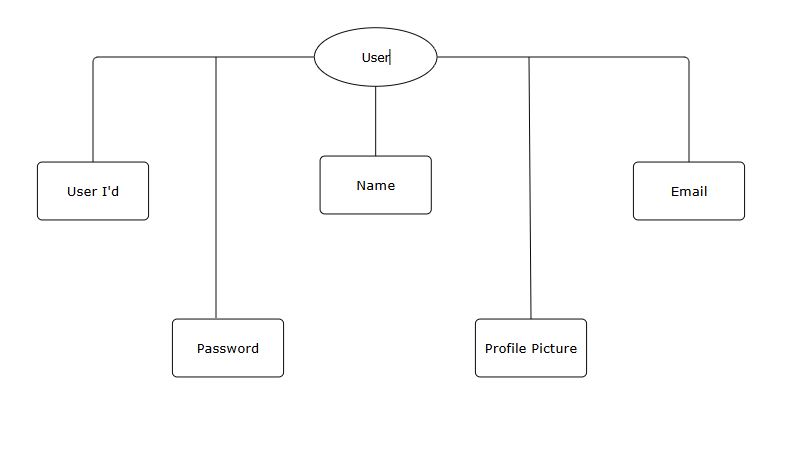
* Mobile app with React Native
* Integration with wearables (Apple Watch, Fitbit)
* AI recommendations for workouts and meals
* Social fitness challenges
* Voice-enabled logging
* Chatbot-based assistant

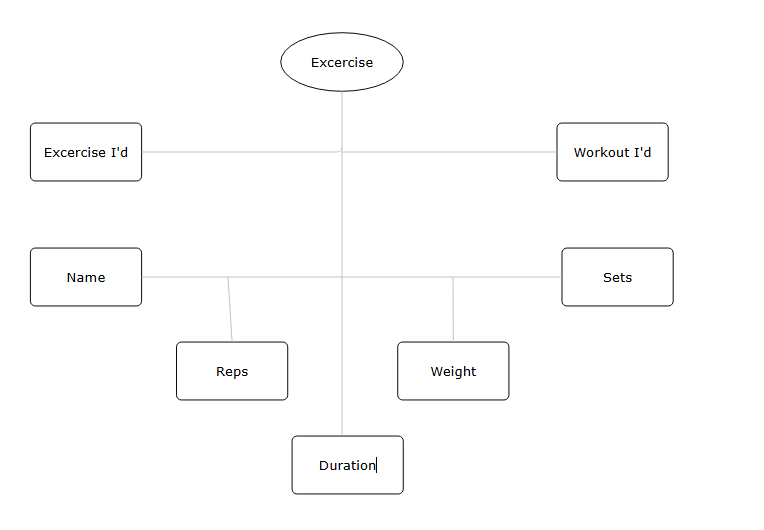
**ER Diagram**

**Fitness Tracker:**

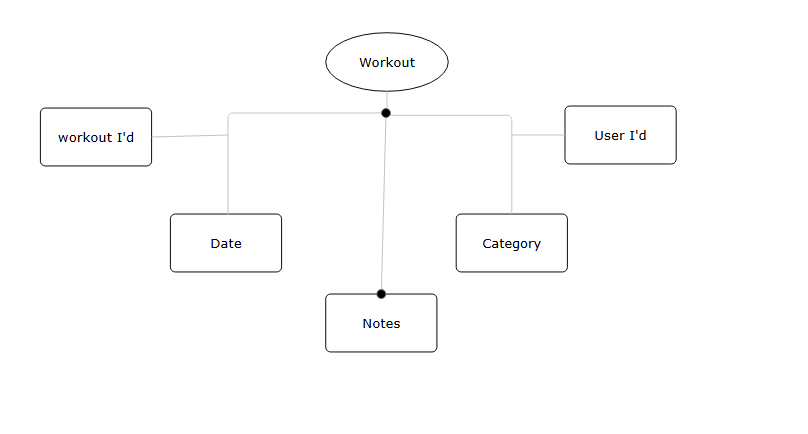
****

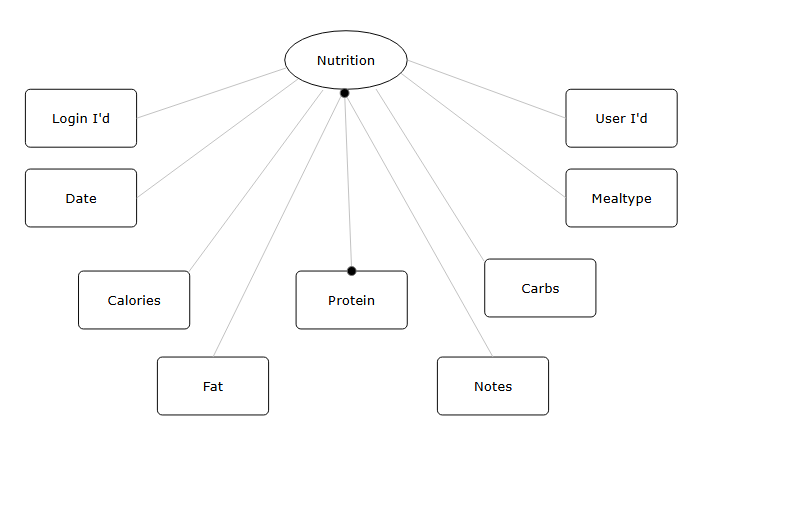
**User:**

****

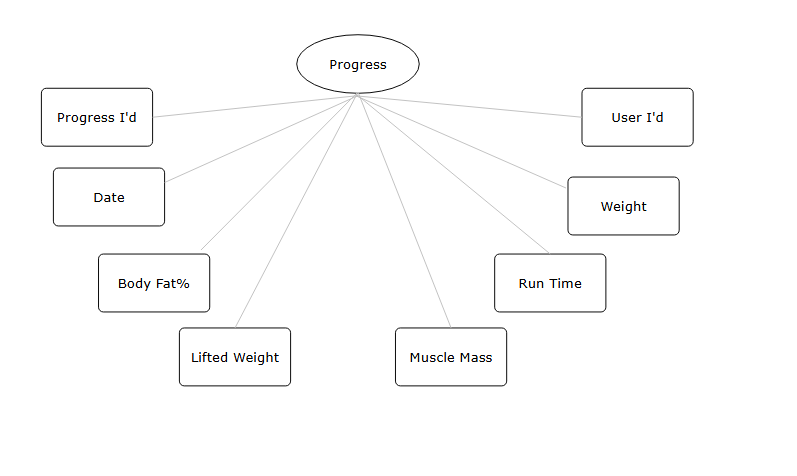
**Exercise:**

**Workout:**

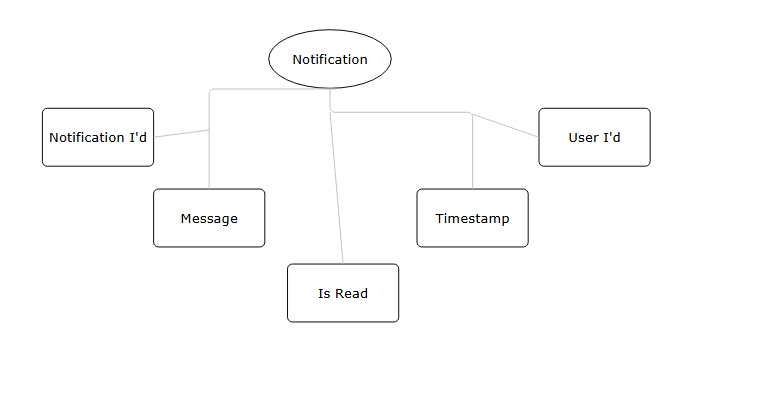
****

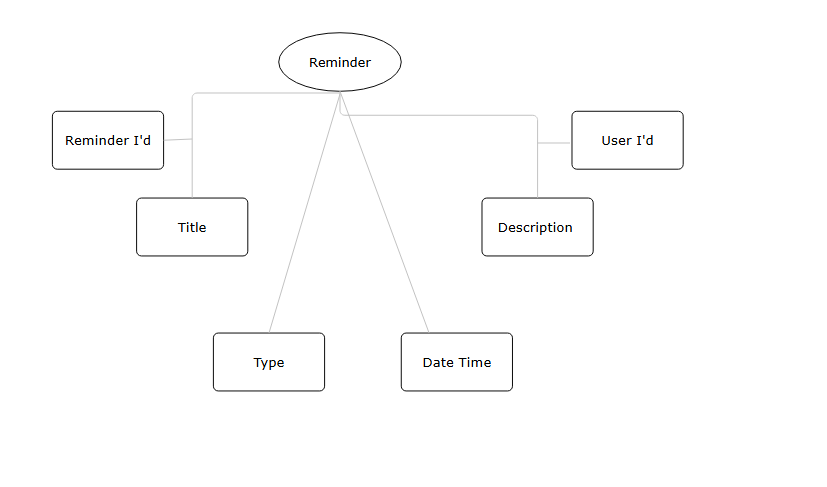
**Nutrition:**

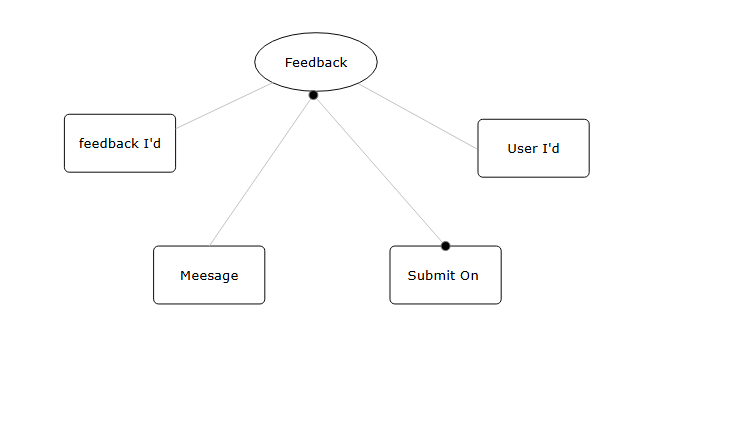
**Progress:**

****

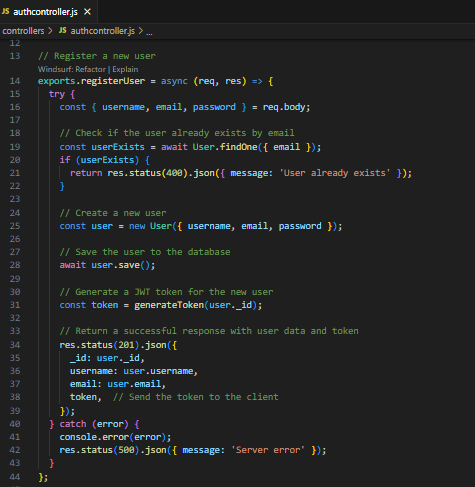
**Notification:**

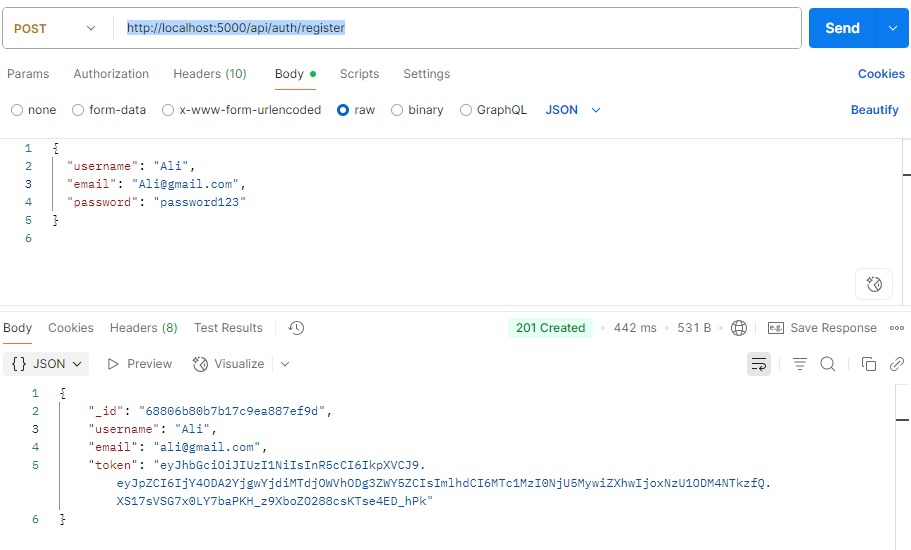
****

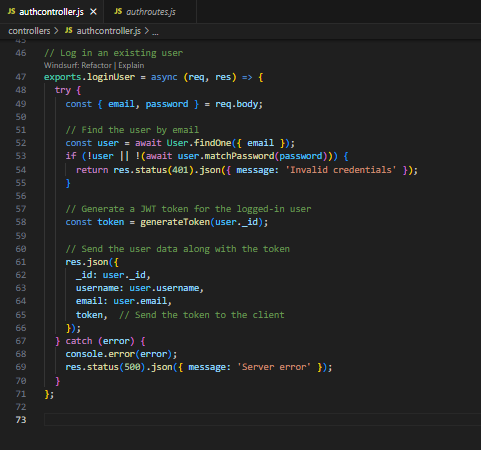
**Reminder:**

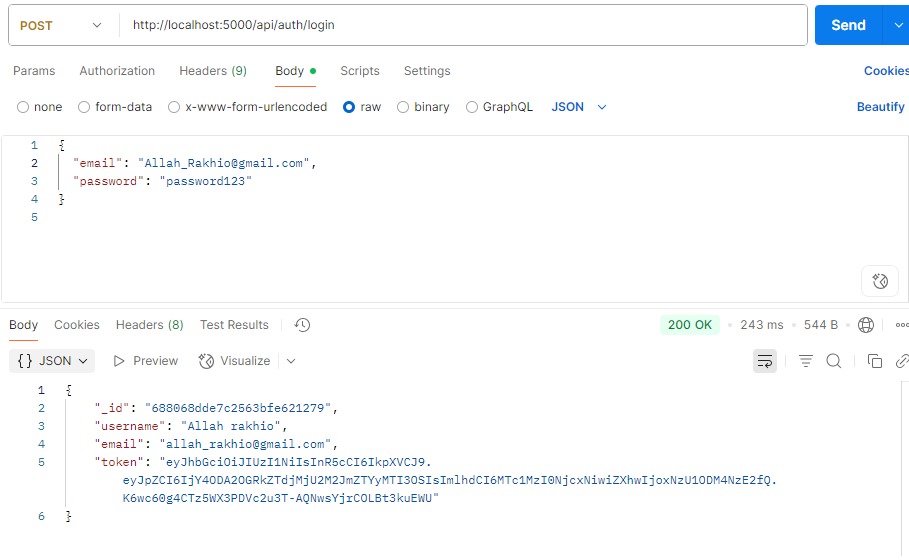
**Feedback:**

**Backend Code & Postman Screenshots:**

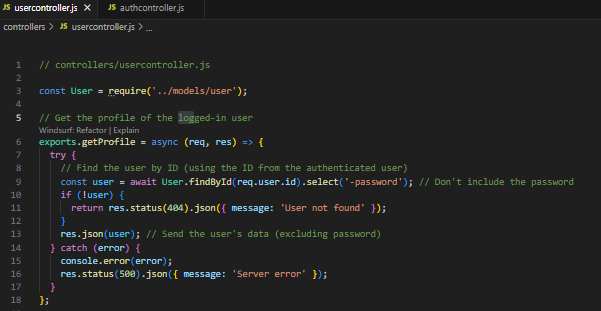
**User Register:**

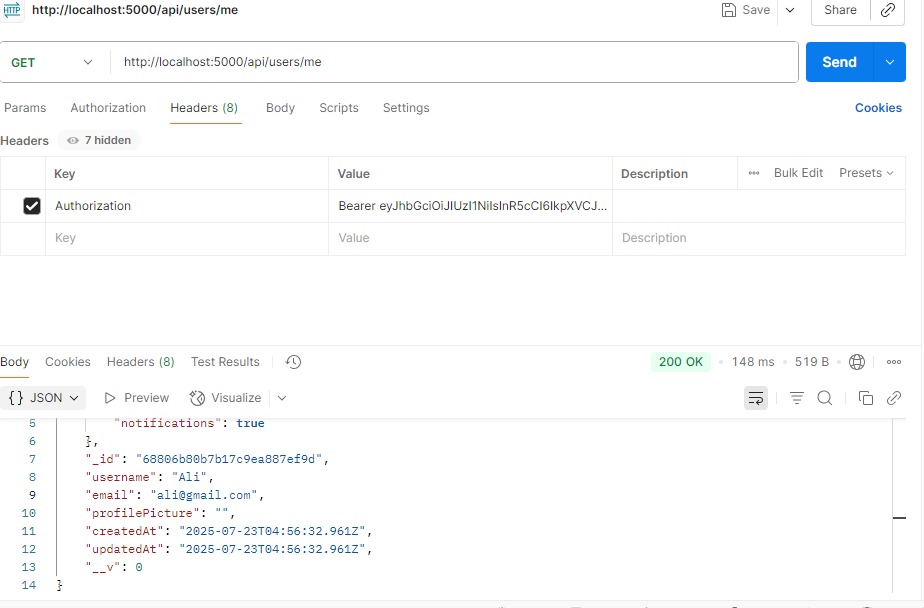


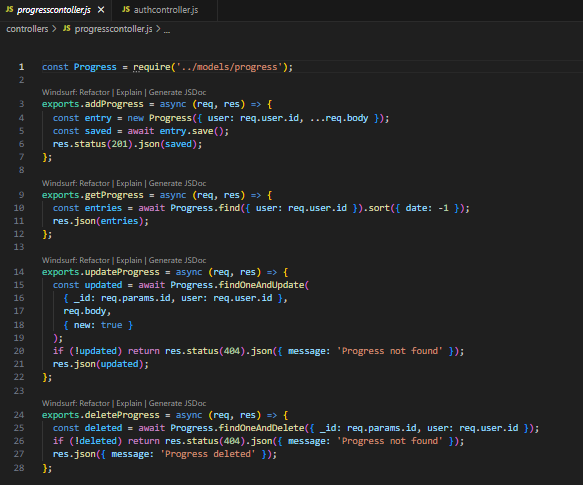
**Login:**

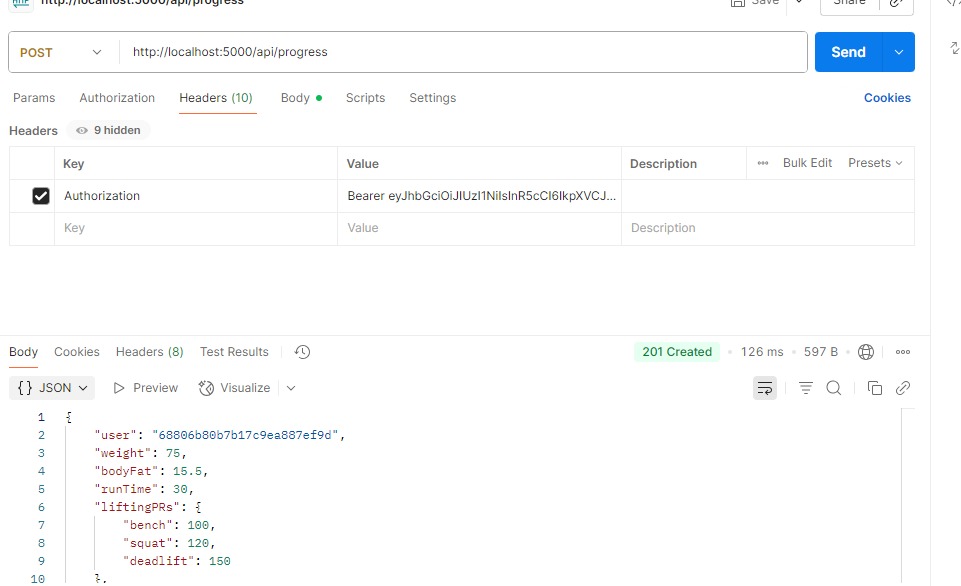


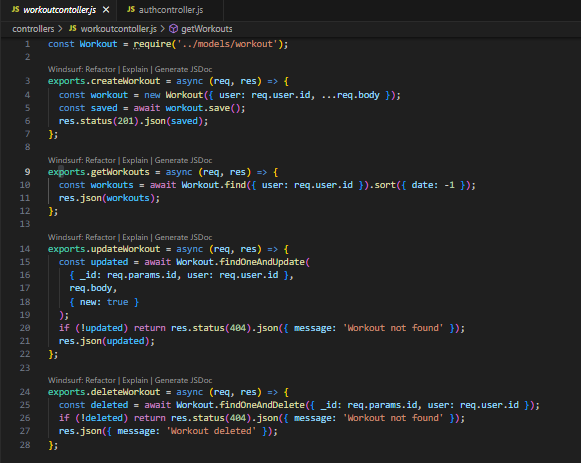
**Get User:**

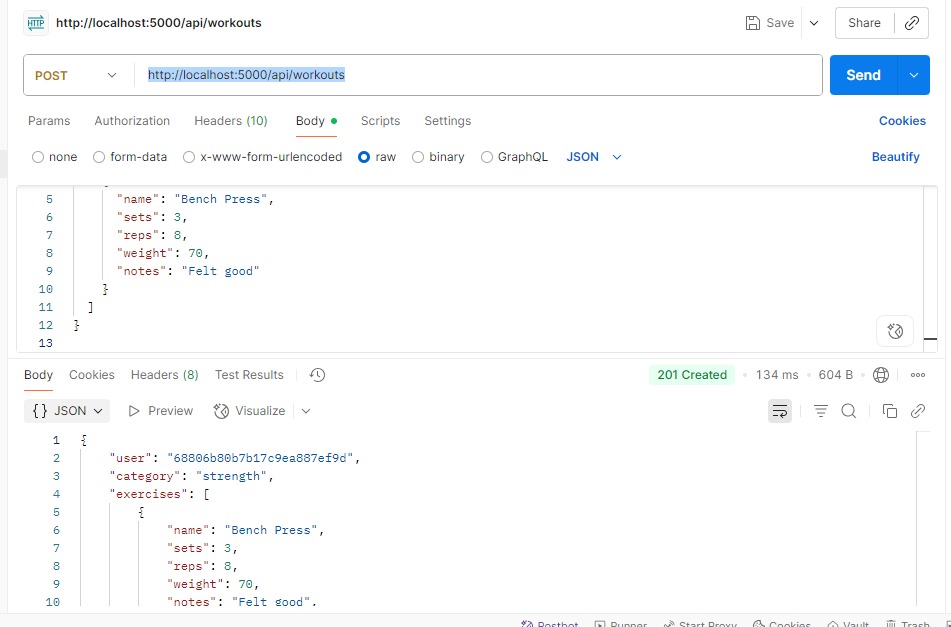


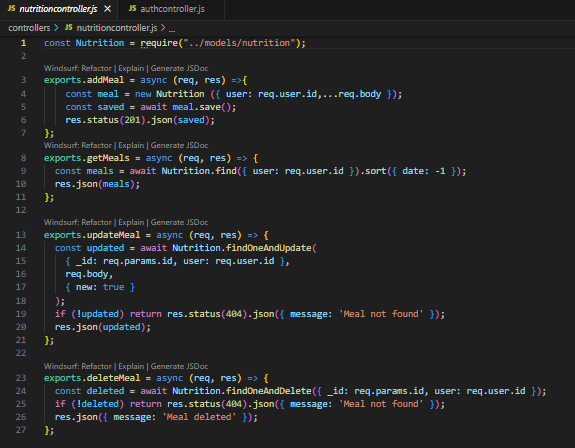


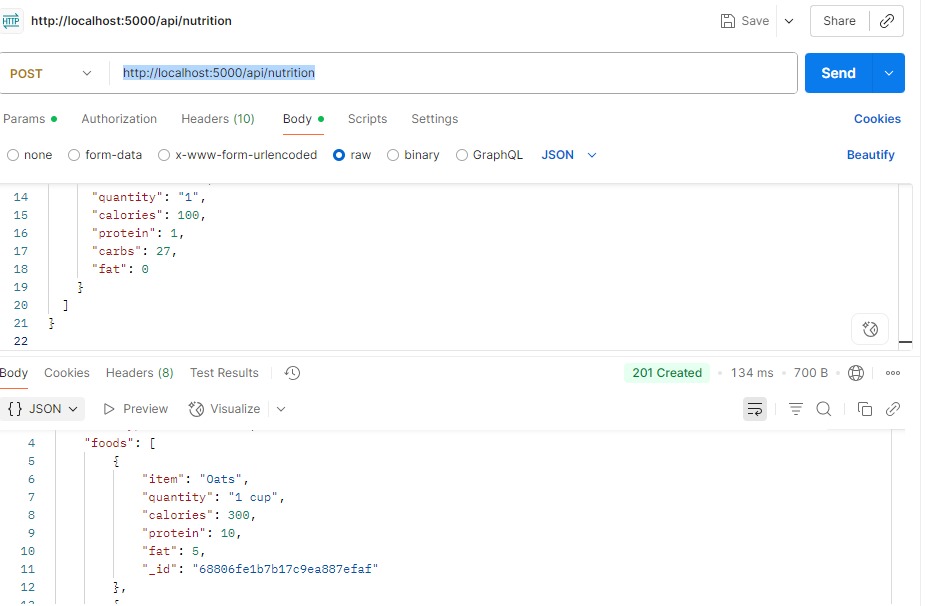
**Progress:**



**Workout:**



**Nutrition:**



**Fron-End:**