**Project Documentation**

**Fitness Tracker Web Application**

* **Project Overview:**

The Fitness Tracker is a web-based application designed to help users track and monitor their fitness activities, including workouts and meals. The project implements input forms for users to log their fitness data and offers statistics and progress charts to display fitness trends. The system uses a combination of front-end and back-end technologies, providing a smooth and interactive experience for users. This project also includes basic user authentication.

* **Project Team:**

1. Dhanashree Chandratre - Front-end Development
2. Jiya Sahu - Front-end Development
3. Apparao Tiruveedula - Backend Development
4. Jaya raju Kolli - Backend Development
5. Manikant Raj - Backend Development
6. Mayank Srivastava - Backend Development
7. Harshita Singh (**Team Lead**) - Backend Development

* **Project Technologies:**
* **Frontend:**
* HTML: For creating the structure of the web pages, including input forms for login, registration, and activity tracking.
* CSS: Styling and responsive design for better user experience.
* JavaScript: Handles form validation, user interactions, and dynamic chart generation.
* **Backend:**
* Node.js with Express.js: Server-side setup to handle requests and responses.
* MongoDB (via Mongoose): Data storage for user details, activity logs, and fitness progress.
* JWT Authentication: For user login and secure access to personal data.
* **Project Setup**:

1**. Installation:**

* Navigate to the backend folder and install dependencies:

`cd backend`

`npm install`

* Set up the environment variables by renaming `.env.example` to `.env` and filling in the necessary MongoDB URI and secret keys.
* Start the server:

`npm start`

2. **Frontend Setup:**

* Access the frontend files directly by opening `index.html` in a browser.
* Ensure the backend server is running to enable the communication between the frontend and backend (for user registration, activity tracking, etc.).
* **Project Features:**

1. **User Authentication:**

* Implemented via `userRoutes.js`. Allows users to register and log in.
* Secure user sessions using JSON Web Tokens (JWT).

2. **Activity Tracking:**

* Users can log workouts and meals, specifying details like exercise type, duration, and calories burned. This data is managed via `activityRoutes.js`.

3. **Data Storage:**

* Data is stored in MongoDB, and the connection is managed by `db.js`.

4. **Progress Charts:**

* Using JavaScript, the application dynamically generates charts to display fitness progress (e.g., total calories burned, time spent exercising).

5. **Error Handling**:

* Server-side error handling for invalid inputs and ensuring robust backend operation.
* **Project Resources/References:**

- [Express.js Documentation](https://expressjs.com/)

- [MongoDB Mongoose](https://mongoosejs.com/docs/)

- [JWT for Authentication](https://jwt.io/)

- [Chart.js for Data Visualization](https://www.chartjs.org/)

* **Project Risks:**
* **Security Risks:** Secure handling of JWT tokens is crucial for protecting user data.
* **Cross-Browser Compatibility**: Ensure that all functionality works across different browsers.
* **Data Loss**: Implement periodic backups of user data.
* **Additional Comments:**

Future enhancements could include adding more comprehensive user profiles and expanding the workout categories.