

# Project Documentation: Calorie Tracker App (Web & Mobile)

## 1. Project Overview

### Idea and Vision

This app aims to be a user-friendly calorie tracker and exercise log, enabling users to track their daily food intake, exercise, and health progress. The app will be designed for ease of use by all age groups, from children to seniors. It will also be family-friendly, allowing family members (especially those abroad) to sync their data and monitor each other's health. Additionally, the app will integrate with wearable devices like smartwatches (Noise, Titan, Fastrack, etc.) to track steps, heart rate, and calories burnt.

Future enhancements will include AI-driven personalized routines and diet plans, support for low-cost health insurance premiums, and more advanced features such as weight loss/gain predictions based on the user's activity.

### Core Features

- User Authentication:**
    - Sign-up/login with personal details like height, weight, age.
  - Dashboard:**
    - Displays user progress on calories consumed and burnt.
    - Shows health-related data over time.
  - Calorie Tracker:**
    - Allows users to input food and exercise data (calories consumed and burnt).
    - Provides real-time feedback on daily calorie goals.
  - Exercise Log:**
    - Users can log exercises like gym or home workouts to calculate calories burnt.
  - Family Sync:**
    - Allows family members to sync their data for monitoring.
  - Smartwatch Integration:**
    - Integration with third-party devices (Noise, Titan, Fastrack) for real-time health data.
  - AI-Driven Recommendations:**
    - Based on user data, the app will suggest personalized routines for health improvement.
  - Health Insurance Integration (Future):**
    - Partner with insurance companies to provide discounts based on health data.
- 

## 2. Technical Architecture

### Frontend (Web)

- **Tech Stack:**
  - **React:** For building user interfaces.
  - **Next.js:** For server-side rendering and SEO optimization.
  - **Webpack** (with Module Federation for Micro Front-End architecture): For splitting the front-end into manageable micro-apps, making the app modular and scalable.
  - **Material-UI** or **Tailwind CSS:** For styling the UI components.
- **Key Components:**
  - **Authentication Module:** User login/signup.
  - **Dashboard Module:** Displays the user's health data.
  - **Tracking Module:** For food and exercise logs.
  - **Health Sync Module:** To synchronize health data from wearables and family members.

## Backend

- **Tech Stack:**
  - **Node.js + Express.js:** For building REST APIs.
  - **MongoDB:** For storing user data, exercise logs, and food logs.
  - **JWT Authentication:** For secure user login and access management.
  - **AWS** or **Heroku:** For deployment.
- **APIs:**
  - CRUD operations for managing user data.
  - Calorie and exercise data management.
  - API integrations for wearable devices.

## Mobile App (React Native)

- **Tech Stack:**
  - **React Native:** For building cross-platform mobile applications.
  - **Expo:** For easy mobile development and testing.
  - **React Navigation:** For navigating between app screens.
  - **Axios:** For API integration with the backend.
  - **Firebase:** For push notifications and real-time syncing.

---

## 3. Pros and Cons

### Pros

- **User-Friendly:** Easy for all age groups to use, from 6-year-olds to 60-year-olds.
- **Family-Oriented:** Provides the ability for family members to monitor each other's health.
- **Scalable:** Modular design using Micro Front-End architecture and backend APIs allows easy scaling of features.
- **Wearable Integration:** Syncs with popular smartwatches for real-time health tracking.
- **Personalized:** Future AI-based features for customized health routines and diet plans.
- **Future-Proof:** Can integrate with health insurance providers in the future to offer benefits.

## Cons

- **Learning Curve:** Requires knowledge of Micro Front-End architecture and React Native, which might take some time to master.
  - **Backend Complexity:** Handling real-time data from wearables and syncing between devices could increase backend complexity.
  - **Wearable Device Integration:** Different smartwatches have different APIs and protocols, making integration challenging.
- 

## 4. Tech Stack Overview

### Front End Tech Stack:

- **React:** For building interactive UIs.
- **Next.js:** For server-side rendering and improved performance.
- **Micro Front-End (Webpack Module Federation):** For a modular architecture that scales well.
- **Material-UI/Tailwind CSS:** For modern, responsive UI components.

### Back End Tech Stack:

- **Node.js + Express.js:** For server-side development.
- **MongoDB:** NoSQL database for flexibility and scalability.
- **JWT Authentication:** For secure, token-based authentication.

### Mobile App Tech Stack:

- **React Native:** For building cross-platform mobile applications.
  - **Firebase:** For push notifications and real-time syncing.
  - **Expo:** For an easy and fast mobile development process.
  - **Axios:** For making API calls to the backend.
- 

## 5. Future Scope

- **AI-Powered Recommendations:** Based on user data (calories consumed, exercise logs), AI can suggest personalized fitness routines and diet plans.
- **Health Insurance Integration:** Partner with insurance companies to offer discounts for healthy users, based on app data.
- **Expanded Device Support:** Integrate with more wearable devices and health trackers for better data aggregation.
- **Premium Subscriptions:** Offer users premium features such as advanced reports, personalized health insights, etc.

- **Social Features:** Allow users to share their progress with friends and family, adding a social aspect to health tracking.
- 

## 6. Timeline

### Phase 1: Learning & Setup (Month 1)

- **Week 1-2:** Learn Micro Front-End concepts and set up Webpack with Module Federation.
- **Week 3-4:** Set up basic Node.js + Express backend with MongoDB and JWT authentication.

### Phase 2: Initial Web App Development (Month 2–3)

- **Week 5-6:** Build the basic front-end with React + Next.js, focusing on user authentication and dashboard.
- **Week 7-8:** Break the front-end into Micro Front-End modules (authentication, dashboard, tracker).
- **Week 9-12:** Integrate APIs with the backend and deploy the initial version to Netlify/Vercel.

### Phase 3: Learn React Native (Month 4)

- **Week 13-14:** Learn React Native basics (components, navigation, API integration).
- **Week 15-16:** Build a mobile prototype with basic features like login and dashboard.

### Phase 4: Mobile App Development (Month 5–6)

- **Week 17-20:** Develop mobile app features: login, dashboard, calorie and exercise tracker.
  - **Week 21-24:** Test, optimize, and deploy the mobile app to the Play Store.
- 

## 7. Documentation & Other Considerations

### User Information Required:

1. **Personal Details:** Name, age, weight, height, gender.
2. **Health Goals:** Weight loss, weight gain, or maintenance.
3. **Diet Information:** Type of food consumed and number of servings.
4. **Exercise Data:** Type of exercise and duration.
5. **Health Sync Data:** Integration with wearables to track steps, heart rate, calories burnt, etc.

### Formulas for Calorie Tracking:

- **Basal Metabolic Rate (BMR):**
  - Mifflin-St Jeor Formula:

- For Men:  $BMR = 10 \times \text{weight} + 6.25 \times \text{height} - 5 \times \text{age} + 5$   
 $BMR = 10 \times \text{weight} + 6.25 \times \text{height} - 5 \times \text{age} + 5$
  - For Women:  $BMR = 10 \times \text{weight} + 6.25 \times \text{height} - 5 \times \text{age} - 161$   
 $BMR = 10 \times \text{weight} + 6.25 \times \text{height} - 5 \times \text{age} - 161$
  - **Calorie Deficit/Surplus:** Based on the user's goals (e.g., 500 calories deficit per day for weight loss).
- 

## 8. User Data Collection and Calculation Formulas Documentation

---

### 1. User Data Collection

To ensure a personalized and accurate tracking experience, the app will collect the following information:

#### 1.1. Basic User Information

- **Full Name:** For personalizing the user experience.
- **Email Address:** For account creation and communication (e.g., notifications, reminders).
- **Password:** For secure account access.
- **Gender:** Affects basal metabolic rate (BMR) calculations.
- **Age:** Required for calculating caloric needs and activity levels.
- **Height:** Required for calculating BMR and caloric needs.
- **Weight:** Required for calculating caloric needs and progress tracking.
- **Target Weight Goal:** (Optional) User's desired weight goal, which can be tracked.
- **Activity Level:** Sedentary, Light, Moderate, Active, Very Active. This will help estimate Total Daily Energy Expenditure (TDEE).

#### 1.2. Health-Related Information

- **Dietary Preferences:** Vegan, Vegetarian, Gluten-Free, etc.
- **Known Medical Conditions:** (e.g., diabetes, heart conditions) for any special diet or exercise recommendations.
- **Sleep Pattern:** (Optional) To help recommend healthier routines.

#### 1.3. Goals

- **Goal Type:** Lose Weight, Maintain Weight, Gain Weight. This helps in setting target caloric intake.
  - **Target Caloric Intake:** The number of calories a user wants to consume to meet their goal.
-

## 2. Formulas for Calculations

### 2.1. Basal Metabolic Rate (BMR) Calculation

BMR is the number of calories a body needs at rest to maintain vital functions such as breathing and digestion.

#### For Men:

$$\text{BMR} = 88.362 + (13.397 \times \text{weight in kg}) + (4.799 \times \text{height in cm}) - (5.677 \times \text{age in years})$$
$$\text{BMR} = 88.362 + (13.397 \times \text{weight in kg}) + (4.799 \times \text{height in cm}) - (5.677 \times \text{age in years})$$

#### For Women:

$$\text{BMR} = 447.593 + (9.247 \times \text{weight in kg}) + (3.098 \times \text{height in cm}) - (4.330 \times \text{age in years})$$
$$\text{BMR} = 447.593 + (9.247 \times \text{weight in kg}) + (3.098 \times \text{height in cm}) - (4.330 \times \text{age in years})$$

- This is the basic caloric need without any activity considered.

### 2.2. Total Daily Energy Expenditure (TDEE)

TDEE calculates how many calories a person needs to maintain their weight, factoring in their activity level.

#### Formula:

$$\text{TDEE} = \text{BMR} \times \text{Activity Factor}$$
$$\text{TDEE} = \text{BMR} \times \text{Activity Factor}$$

#### Activity Factors:

- Sedentary (little to no exercise): **1.2**
- Lightly Active (light exercise or sports 1-3 days/week): **1.375**
- Moderately Active (moderate exercise or sports 3-5 days/week): **1.55**
- Very Active (hard exercise/sports 6-7 days a week): **1.725**
- Extremely Active (very hard exercise or training twice a day): **1.9**

### 2.3. Caloric Deficit/Surplus for Weight Loss or Gain

- **To Lose Weight:** A typical caloric deficit of 500-1000 kcal/day is recommended for healthy weight loss (leading to 0.5–1 kg of weight loss per week).  $\text{Target Calories} = \text{TDEE} - 500$  (for weight loss)
- **To Gain Weight:** A caloric surplus of 500-1000 kcal/day is recommended for healthy weight gain.  $\text{Target Calories} = \text{TDEE} + 500$  (for weight gain)

The target caloric intake is adjusted depending on whether the user aims to lose or gain weight.

## 2.4. Macronutrient Breakdown

To ensure a balanced diet, the app will calculate the breakdown of macronutrients (proteins, fats, and carbohydrates) based on the user's target caloric intake.

### Macronutrient Ratios:

- Protein: **30%** of total calories
- Fat: **30%** of total calories
- Carbohydrates: **40%** of total calories

### Formula to calculate grams of each macronutrient:

- **Protein:**  $\text{Protein (grams)} = (\text{Target Calories} \times 0.30) / 4$   
 $\text{Protein (grams)} = \left( \text{Target Calories} \times 0.30 \right) / 4$
- **Fat:**  $\text{Fat (grams)} = (\text{Target Calories} \times 0.30) / 9$   
 $\text{Fat (grams)} = \left( \text{Target Calories} \times 0.30 \right) / 9$
- **Carbohydrates:**  $\text{Carbs (grams)} = (\text{Target Calories} \times 0.40) / 4$   
 $\text{Carbs (grams)} = \left( \text{Target Calories} \times 0.40 \right) / 4$

## 2.5. Exercise and Calories Burned

The user will enter their exercises and the app will calculate the calories burned based on the type of activity, intensity, and duration.

### Basic Formula to Calculate Calories Burned:

$\text{Calories Burned} = \text{MET} \times \text{weight in kg} \times \text{duration in hours}$   
 $\text{Calories Burned} = \text{MET} \times \text{weight in kg} \times \text{duration in hours}$

- **MET** (Metabolic Equivalent of Task) varies based on the exercise. For example:
  - Running (8 km/h) has an MET of **7**
  - Cycling (moderate intensity) has an MET of **6**
  - Walking (4 km/h) has an MET of **3.9**

## 2.6. Net Calorie Calculation

The **Net Calories** is the amount of calories the user can still consume for the day after accounting for both their intake and exercise:

$\text{Net Calories} = \text{Calories Consumed} - \text{Calories Burned}$   
 $\text{Net Calories} = \text{Calories Consumed} - \text{Calories Burned}$

### For Example:

- If the user consumes **2000 calories** and burns **300 calories** through exercise, their net calories would be:  $\text{Net Calories} = 2000 - 300 = 1700$   
 $\text{Net Calories} = 2000 - 300 = 1700$
-

### 3. Data Privacy and Consent

- **Consent for Data Collection:** The user should be informed that the app will collect health data, and they must consent to this before entering sensitive information.
  - **Access to Family Members' Data:** Users can choose to share their data with family members for monitoring purposes. This should be explicitly stated in the app's privacy policy.
- 

### 4. User Interface Guidelines

The app should ensure that all forms for entering data are simple, intuitive, and family-friendly. The UI should guide users in inputting:

- Personal data like height, weight, and age.
  - Food consumption through a searchable database of foods.
  - Exercise routines, with pre-defined exercises for easy selection.
  - Goal settings, allowing users to input their goals for weight loss, maintenance, or gain.
- 

### 5. Future Calculations and Integrations

- **AI-based Personalized Recommendations:** After a month of data collection, the app will start recommending customized exercise routines and diet plans based on user habits and preferences.
  - **Wearable Device Integration:** The app will sync with smartwatches to provide real-time heart rate, steps, and calories burned, allowing for more accurate net calorie calculations.
- 

### 6. Testing and Validation

- **Test Scenarios:** Ensure that the formulas work correctly for a wide range of data, including edge cases like extreme heights/weights and different activity levels.
- **User Feedback:** Collect feedback on the ease of use and accuracy of the app's calculations, adjusting as necessary.

## 9. App Workflow

The app will guide users through a seamless journey, collecting data, tracking progress, and offering insights. Here's the step-by-step workflow:

#### 9.1. User Signup

- **Page:** Signup/Login page.
- **Action:** Users enter their personal details (name, email, password).
- **Outcome:** A new user account is created or an existing user is logged in.



## 9.2. User Data Collection

- **Page:** User Profile Setup.
- **Action:** After logging in, users fill in their physical details (age, height, weight, gender, activity level).
- **Outcome:** The app calculates the Basal Metabolic Rate (BMR) and Total Daily Energy Expenditure (TDEE) to personalize the caloric needs.

## 9.3. Daily Tracking

- **Page:** Dashboard.
  - Users enter daily activities, meals, and exercise data.
  - **Food Tracking:** Users select foods from a pre-defined list or enter custom meals, specifying quantity.
  - **Exercise Tracking:** Users log exercise sessions, including activity type, duration, and intensity.
  - **Calories Burned:** The app calculates calories burned from exercises and updates the net caloric intake.
- **Outcome:** The app provides real-time updates on calories consumed, burned, and remaining for the day.

## 9.4. AI Recommendations

- **Page:** Suggestions.
- **Action:** Based on the user's past activity, preferences, and goals, the AI system suggests:
  - **Workout plans** (personalized exercises based on activity level, past workouts, etc.)
  - **Diet plans** (meal suggestions based on the caloric intake goal)
  - **Adjustments:** If users aren't meeting their target goals, the AI suggests necessary changes to diet or exercise.
- **Outcome:** Continuous improvement based on user data.

## 9.5. Goal Tracking and Progress

- **Page:** Progress Tracker.
  - Users can track their weight loss, muscle gain, or maintenance goals.
  - **Reports:** Daily, weekly, and monthly progress reports are generated.
  - **Graphs and Visuals:** Visual progress charts showing calorie intake, calories burned, weight changes, and exercise consistency.

## 9.6. Notifications & Alerts

- Users are notified about their daily progress, reminders to log food/exercises, and AI-generated suggestions for improvements.

## 9.7. Data Sharing (Optional)

- **Page:** Family Sync.
  - Users can share their data with family members or caregivers.

- Notifications are sent to family members if the user is nearing goal targets, has missed logging activities, etc.
- 

## 10. AI and Recommendation System

The AI system will drive personalized recommendations, making the app smarter over time. It will provide personalized workout routines, meal suggestions, and behavioral adjustments based on user data.

### 10.1. Data Collection

- **Inputs:**
  - User data (age, gender, weight, height, activity level)
  - Daily exercise logs (type, duration, intensity)
  - Meal logs (food type, calories consumed)
  - Goal information (weight loss, muscle gain, etc.)
  - Progress data (weight tracking, exercise consistency)

### 10.2. Data Preprocessing

- **Normalization:** Standardize values for weight, calories, exercise duration, etc., for easy comparison.
- **Cleaning:** Remove any incomplete or invalid entries.

### 10.3. AI Algorithms

The AI system uses a combination of supervised and unsupervised learning algorithms to make accurate predictions and suggestions:

- **Collaborative Filtering:** To recommend diets and exercises that other similar users have benefited from.
- **Reinforcement Learning:** The AI adapts over time, learning the user's preferences based on actions taken (diet, exercise).
- **Clustering:** Group users with similar goals and behaviors together, enabling the app to offer more tailored advice.
- **Linear Regression:** Used for predicting weight loss/gain and adjusting caloric intake recommendations.

### 10.4. Recommendation Flow

- **Diet Recommendations:**
  - Based on caloric intake goals, the AI system suggests meal plans. It factors in the user's dietary preferences (vegan, vegetarian, etc.).
  - As users log more meals, the system refines its recommendations to suit individual preferences.
- **Exercise Recommendations:**

- Based on the user's activity level, past workouts, and goals, the AI suggests personalized workout routines.
- It recommends workouts that can help the user achieve their targets (e.g., weight loss, muscle gain).
- **Behavioral Adjustments:**
  - If users are consistently under/over-consuming calories, the AI suggests changes (e.g., more protein, longer workouts).
  - If a user's progress is slower than expected, the AI can suggest revising their workout routine or caloric intake.

## 10.5. Real-Time Feedback

- AI continually monitors the user's daily logs (meals, exercise) and gives real-time suggestions to help them stay on track with their goals.
- 

# 11. Notifications and Alerts

To keep the user engaged and on track, the app will feature a robust system for notifications and alerts. These will be personalized based on the user's preferences and needs.

## 11.1. Types of Notifications

- **Daily Reminders:**
  - Reminder to log food intake (morning, afternoon, evening).
  - Reminder to log exercises after a workout.
- **Progress Alerts:**
  - Reminders when users are close to meeting their daily caloric intake goal.
  - Alerts when a user is behind schedule in achieving their weekly or monthly target.
  - Congratulatory notifications for meeting milestones.
- **AI-Based Suggestions:**
  - Push notifications when the AI recommends new exercises, meals, or behavioral changes.
- **Family/Health Alerts:**
  - Alerts to the family members when the user shares their progress or needs assistance (e.g., when missing exercise sessions).

## 11.2. Push and In-App Notifications

- **Push Notifications:**
  - Send notifications on mobile devices even when the app is not open (e.g., reminders to log meals, exercises).
- **In-App Notifications:**
  - Alerts inside the app for real-time tracking (e.g., progress bar updates).

## 11.3. Notification Preferences

- Users can control the frequency and type of notifications they want to receive:
    - **Frequency:** Daily, Weekly, or Immediate.
    - **Type:** Exercise, diet, goal tracking, or AI suggestions.
- 

## 12. Subscription/Monetization Model

The app will adopt a **freemium** model, with premium features available through subscription plans. The monetization strategy will focus on both subscription revenue and in-app purchases.

### 12.1. Free Version

- **Core Features:**
  - Basic profile setup and tracking (calorie consumption, exercise logs).
  - Basic AI-powered suggestions for diet and exercise.
  - Daily/weekly progress reports.
- **Limitations:**
  - Limited access to advanced features such as personalized workout plans, premium meal suggestions, and detailed progress analytics.

### 12.2. Premium Subscription Plans

The premium plans unlock advanced features:

- **Monthly Plan:** ₹200/month
- **Quarterly Plan:** ₹500/quarter
- **Yearly Plan:** ₹1,500/year (best value)

#### Premium Features:

- **Advanced AI Recommendations:** Personalized meal plans, workout routines, and behavior modification tips.
- **Custom Goals & Targets:** Set more detailed goals for weight, muscle gain, or fitness levels.
- **Data Sync:** Sync data with family members or caregivers.
- **Premium Content:** Access to special meal plans, expert health tips, and workouts by certified trainers.

### 12.3. In-App Purchases

- **Individual Meal/Workout Plans:** Users can buy one-off expert-created diet plans or exercise routines.
- **Health Insights:** Detailed reports on the user's progress and health stats for a fee.

### 12.4. Corporate Partnerships

- Collaborate with health insurance companies to offer discounted premiums for users who meet certain health criteria tracked via the app. This could serve as an additional revenue stream and incentivize users to stay active and healthy.

### **12.5. Ads**

- Display non-intrusive ads in the free version of the app. Premium users will have an ad-free experience.

## **13. Conclusion**

This project will create a user-friendly and scalable calorie tracker and exercise log app that will benefit users of all age groups, especially those looking to improve their health in a simple way. The integration with wearables, along with the use of AI-driven recommendations and future health insurance collaboration, gives the app significant long-term potential.

By following this roadmap, you can gradually progress from learning the necessary technologies to building and deploying a fully functional web and mobile app, with modular micro front-end architecture and cross-platform compatibility.