**Software Requirements Specification  
for  
Eco Habits Tracker**

**Prepared by:** **Team 37**  
**Organization:** Mahindra University  
**Date Created:** Feb 25th

**Table of Contents**

1. **Introduction**  
   1.1 Purpose  
   1.2 Document Conventions  
   1.3 Intended Audience and Reading Suggestions  
   1.4 Product Scope  
   1.5 References
2. **Overall Description**  
   2.1 Product Perspective  
   2.2 Product Functions  
   2.3 Operating Environment  
   2.4 Design and Implementation Constraints  
   2.5 User Documentation  
   2.6 Assumptions and Dependencies  
   2.7 Functional Requirements Specification
3. **External Interface Requirements**  
   3.1 User Interfaces  
   3.2 Hardware Interfaces  
   3.3 Software Interfaces  
   3.4 Communication Interfaces
4. **System Features**  
   4.1 Use Case: Log a Habit
5. **Other Nonfunctional Requirements**  
   5.1 Performance Requirements  
   5.2 Safety Requirements  
   5.3 Security Requirements  
   5.4 Software Quality Attributes
6. **Other Requirements**

**1. Introduction**

**1.1 Purpose**

The purpose of this document is to outline the software requirements for **Eco Habits Tracker**, an application designed to encourage eco-friendly habits through tracking, gamification, and analytics. The app provides users with streak tracking, badges, daily eco tips, and leaderboards to motivate environmental awareness and responsibility.

**1.2 Document Conventions**

This SRS follows the IEEE Software Requirements Specification template. Requirements are uniquely identified (e.g., FR-1, FR-2) and prioritized accordingly. Formatting, terminology, and fonts are used consistently throughout.

**1.3 Intended Audience and Reading Suggestions**

This document is intended for:

* **Developers**: To understand functional, performance, and interface requirements.
* **Project Managers**: For planning and progress tracking.
* **Testers**: To create test cases based on specified requirements.
* **Stakeholders**: To validate that the app meets its intended purpose.

**1.4 Product Scope**

**Eco Habits Tracker** is a mobile application designed to help users track and develop sustainable habits. The app encourages daily participation through a gamified approach, including streaks, rewards, and performance insights. The main features include:

* **Logging Habits** (e.g., saving water, reducing plastic use, using public transport)
* **Tracking Progress** via streaks and eco-scores
* **Leaderboard & Achievements**
* **Personalized Eco Tips**

**1.5 References**

* IEEE SRS Template Guidelines
* Firebase Documentation
* Flutter Documentation

**2. Overall Description**

**2.1 Product Perspective**

Eco Habits Tracker is a mobile application designed to promote sustainable living by encouraging users to develop eco-friendly habits. Built using Flutter for cross-platform support and Firebase Firestore for real-time data storage and authentication, the app provides an interactive, gamified approach to tracking sustainability efforts.

The app operates independently and does not require external hardware beyond a smartphone. It leverages habit tracking, leaderboards, and insights to foster long-term engagement and behavioural change.

**2.2 Product Functions**

Key functionalities include:

* Habit Logging: Users can log daily eco-friendly actions from predefined or custom categories.
* Streak & Progress Tracking: Users receive feedback on their consistency through streaks and eco-scores.
* Achievements & Badges: Users unlock badges for completing milestones.
* Daily Eco Tips: AI-generated or expert-curated sustainability advice.
* Leaderboards: Compare progress with friends and the global community.

**2.3 Operating Environment**

* Platforms: Android & iOS
* Backend: Firebase Firestore (Cloud Database)
* Development Framework: Flutter (Dart-based)
* Hardware: Smartphone with an active internet connection

**2.4 Design and Implementation Constraints**

* Real-time Syncing: The leaderboard updates dynamically, requiring optimized network requests.
* Firebase Limitations: Firestore supports 1M daily reads on the free tier, so data must be efficiently managed.
* Battery & Performance Optimization: Ensures minimal background processing to conserve device resources.
* Cross-Platform UI: Uses Material Design (Android) and Cupertino (iOS) for a seamless user experience.

**2.5 User Documentation**

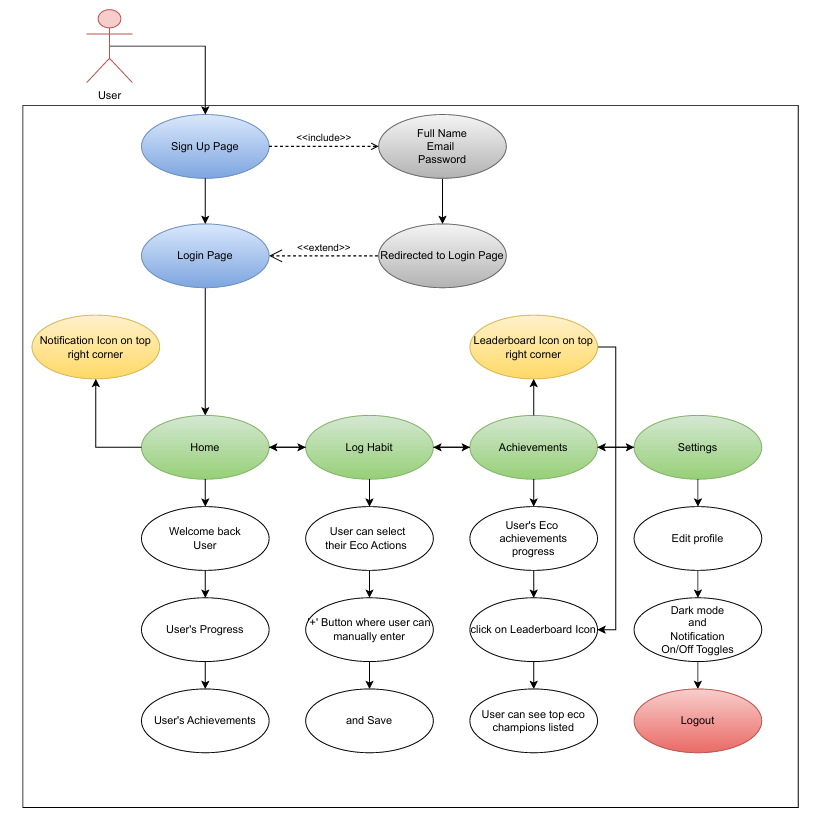
* User Manual: Step-by-step guide covering installation, setup, and feature usage.
* Online Help & FAQs: Integrated support system for resolving user queries.
* Push Notifications: Daily reminders for habit tracking.

**2.6 Assumptions and Dependencies**

* Users must have an internet connection for real-time updates.
* Firebase Authentication is used for secure login and profile management.
* AI-driven recommendations may rely on third-party APIs in future updates.

**2.7 Functional Requirements Specification**

* FR-1: Users shall be able to log eco-friendly habits daily.
* FR-2: Users shall be able to track progress using streaks and analytics.
* FR-3: The system shall award badges for habit completion milestones.
* FR-4: The leaderboard shall update in real-time to reflect user progress.
* FR-5: The app shall provide daily eco tips based on user activity.

**Use Case: UML(Example)**

**3. External Interface Requirements**

**3.1 User Interfaces**

The user interface of Eco Habits Tracker is designed to be intuitive, visually appealing, and user-friendly, ensuring seamless interaction with the app's features.

* **Home Screen**
  + Displays the user's current eco-streaks and eco-score.
  + Provides a quick-access button to log a habit.
  + Shows daily eco tips to educate users on sustainable practices.
  + Includes navigation links to the leaderboard, habit log, and settings.
* **Leaderboard Screen**
  + Displays user rankings based on eco-scores.
  + Provides a toggle to view rankings among friends vs. global users.
  + Highlights top users with badges and achievements.
  + Offers filtering options (e.g., weekly, monthly, all-time rankings).
* **Habit Log Screen**
  + Allows users to select and log eco-friendly habits from predefined categories (e.g., reducing plastic, saving water, using public transport).
  + Enables users to manually input details like duration, quantity, or type of habit.
  + Supports automated habit tracking via smartphone sensors where applicable.
  + Displays historical habit logs with a calendar view for better progress tracking.

**3.2 Hardware Interfaces**

The Eco Habits Tracker app interacts with various hardware components of a smartphone to enhance user experience:

* **Smartphone Sensors**
  + Accelerometer & GPS: Can be used to track walking or cycling habits (e.g., instead of using a car).
  + Camera: May allow users to scan QR codes or take photos as proof of eco-friendly actions.
  + NFC (Optional Future Feature): Could be used to verify sustainable purchases in stores.
* **Storage & Memory Considerations**
  + The app will store habit logs locally before syncing with Firebase.
  + Requires a minimum of 2GB RAM and a dual-core processor for smooth functionality.

**3.3 Software Interfaces**

The app integrates with multiple software frameworks and third-party services:

* **Firebase Firestore**
  + Serves as the real-time database to store user habits, eco-scores, and leaderboard data.
  + Supports offline data storage and automatic synchronization when reconnected.
  + Uses Firestore rules for access control and data integrity.
* **Flutter UI Framework**
  + Provides a cross-platform experience for both Android and iOS users.
  + Ensures smooth animations and responsiveness using Flutter’s widget-based UI.
  + Supports Material Design (Android) and Cupertino (iOS) UI elements for a native feel.
* **Third-Party Libraries & APIs**
  + May integrate with Google Maps API to promote eco-friendly travel routes.
  + Can use AI-based APIs to generate personalized sustainability tips.

**3.4 Communication Interfaces**

To ensure secure and reliable communication, the app follows best practices for data exchange:

* **HTTPS (Hypertext Transfer Protocol Secure)**
  + All data exchanged between the mobile app and Firebase backend will be encrypted using TLS (Transport Layer Security).
  + Prevents man-in-the-middle (MITM) attacks and data leaks.
* **Real-Time Data Syncing**
  + Firebase’s real-time sync updates leaderboards dynamically.
  + Ensures instant updates to eco-scores and progress tracking.
* **Push Notifications (FCM - Firebase Cloud Messaging)**
  + Sends daily habit reminders to encourage consistency.
  + Alerts users about challenges, new badges, or leaderboard updates.
* **Data Backup & Restore**
  + Users can backup their progress using Firebase and restore data when reinstalling the app.
  + Optional Google Drive or iCloud backup support in future update

**4. System Features**

**4.1 Use Case: Log a Habit**

Trigger

The user selects “Log Habit” from the home screen.

Basic Flow

1. The user selects a habit category from the provided list.
2. The system prompts the user to provide habit details (e.g., duration, quantity).
3. The system logs the habit and updates the user’s streak.
4. A confirmation message is displayed with updated eco-score.

Alternative Paths

* If offline, the habit is stored locally and synced later when the internet is available.

Postconditions

* The habit is recorded successfully, and the user’s progress is updated.
* The leaderboard is updated with the latest eco-score.

**5. Other Nonfunctional Requirements**

**5.1 Performance Requirements**

* The app shall maintain a response time of less than 2 seconds for habit logging.

**5.2 Safety Requirements**

* User data must be encrypted.

**5.3 Security Requirements**

* Authentication via Firebase Authentication.

**5.4 Software Quality Attributes**

* **Usability:** Intuitive and engaging UI.
* **Scalability:** Capable of handling increasing users.

**6. Other Requirements**

* **Accessibility:** Should support visually impaired users.
* **Internationalization:** Future support for multiple languages.

**Appendix A: Glossary**

* **Eco Score:** A metric that tracks a user's eco-friendly habits.
* **Streaks:** Consecutive days of habit completion.
* **Gamification:** The process of using game-like elements to increase engagement.
* **Leaderboard:** A ranking system displaying users' eco-friendly contributions.
* **Habit Logging:** The process of recording eco-friendly actions taken by users.
* **Sustainability Tips:** Daily eco-friendly advice to encourage greener living.

**Appendix B: Analysis Models**

* **Use Case Diagrams:** Detailed diagrams illustrating user interactions (e.g., logging a habit, tracking progress, viewing leaderboards) will be provided in supplementary documentation.
* **Sequence Diagrams:** Diagrams outlining the flow of data between the user interface, Firebase backend, and Flutter UI framework.
* **Data Flow Diagrams:** Diagrams that model how user habit data and settings are managed within the application.