**E-Waste Management & Recycling System - SDLC Phase 1-3**

**Introduction**

The rapid increase in electronic waste (e-waste) poses significant environmental and health risks. Many individuals lack access to proper disposal methods, leading to improper dumping and pollution. The E-Waste Management & Recycling System aims to bridge this gap by providing an easy-to-use platform where users can request e-waste pickups, locate recycling facilities, access educational resources, and stay informed about recycling events. This document outlines the first three phases of the Software Development Life Cycle (SDLC): Planning & Analysis, Requirements Definition, and System Design.

**1. Planning & Analysis**

**1.1 Feasibility Study**

The E-Waste Management & Recycling System is designed to streamline e-waste disposal by allowing users to request pickups, find recycling centers, and access educational resources.

* **Technical Feasibility:** Uses a web and mobile-friendly platform with Google Maps integration.
* **Economic Feasibility:** Reduces costs for municipalities by improving recycling efficiency.
* **Operational Feasibility:** Encourages user participation through easy requests and notifications.

**1.2 Requirement Documents**

**Users:**

Residents, Recycling Centers, Admins.

**Core Features:**

* User registration and profile management.
* E-waste pickup request system.
* Recycling facility locator with Google Maps.
* Educational resources for e-waste disposal.
* Feedback and reporting system.
* Notifications for events and updates
* Mobile-friendly UI for better accessibility

**2. Requirements Definition**

**2.1 Software Requirements Specification (SRS)**

**Functional Requirements:**

* Allow users to register and manage profiles.
* Enable users to request e-waste pickups.
* Display nearby recycling centers on an interactive map.
* Provide educational content about safe disposal.
* Send notifications for events and updates.
* Allow users to give feedback.

**Non-Functional Requirements:**

* The system must be secure and scalable.
* It should work on mobile and desktop platforms.
* Data should be stored efficiently for faster processing.

**2.2 Use Case Documents**

**Use Case 1:** User Requests E-Waste Pickup

* User selects pickup location, types of e-waste, and submits a request.
* The request is assigned to the nearest recycling center.
* User receives a confirmation notification.

**Use Case 2:** Locating a Recycling Center

* User searches for nearby recycling centers.
* The system displays available centers on a map.
* User gets directions via Google Maps.

**2.3 Traceability Matrix**

**Functional Requirements Traceability**

* User Registration & Login is designed under the authentication system and implemented in auth.js, with test cases ensuring secure login and account management.
* E-Waste Pickup Request is handled by the pickup request system, implemented in pickup.js, and tested for proper request submission and tracking.
* Facility Locator is integrated with Google Maps API, designed to help users find recycling centers, implemented in locator.js, and tested for accuracy and responsiveness.
* Educational Resources are managed through a content management system, implemented in resources.js, and tested for accessibility and content correctness.
* Feedback & Reporting enables users to share their recycling experiences and report issues. It is implemented in feedback.js and tested for smooth submission and data collection.
* Notification System sends real-time updates about local recycling events and initiatives, implemented in notifications.js, and tested for timely alerts.
* Resource Sharing allows users to share tips and information on e-waste recycling through a community forum, implemented in sharing.js, and tested for engagement and content moderation.

**Non-Functional Requirements Traceability**

* Security is a priority, ensuring user data protection with Firebase Authentication and encrypted data storage. This is tested for compliance with security standards.
* Performance is optimized with fast response times using API optimization, caching, and load balancing techniques. Testing focuses on speed and efficiency.
* Usability is maintained through an intuitive and user-friendly interface built with React Native, tested for accessibility and smooth navigation.
* Scalability is achieved using a cloud-based database, allowing the system to handle increasing user loads effectively. Testing ensures seamless performance even with a growing user base.

**3. System Design**

**3.1 High-Level Design (HLD)**

**Architecture:**

**Frontend:** React.js for an interactive user interface.

**Backend:**  Node.js to handle requests.

**Database:** MongoDB for storing user data and recycling center details.

**APIs:** Leaflet for free /Google Maps API for geolocation services.

Razorpay and firebase cloud messageing

**Major Components:**

**User Module:** Registration, login, profile.

**Request Module:** E-waste pickup requests.

**Map Module:** Recycling facility locator.

**Notification Module:** Alerts and event updates.

**3.2 Low-Level Design (LLD)**

**1. User Authentication Module**

Technology: JWT-based authentication system.

**Flow:**

* User registers or logs in.
* Credentials are verified using a database.
* JWT token is issued for session management

**2. E-Waste Pickup Request Module**

Technology: REST API with database integration.

**Flow:**

* User submits a pickup request (e-waste type, location, time).
* The request is stored in the database.
* The system assigns the request to a recycler.
* Recycler receives a notification and confirms the pickup.

**3. Maps Integration Module**

Technology: Google Maps API.

**Flow:**

* User searches for a nearby recycling center.
* The system fetches locations from the database.
* The Google Maps API displays available centers.

**4. Notification System**

Technology: Firebase Cloud Messaging (FCM).

**Flow:**

* System generates an alert (pickup confirmation, recycling event).
* Notification is sent to the user’s device via FCM.
* User receives a push notification.

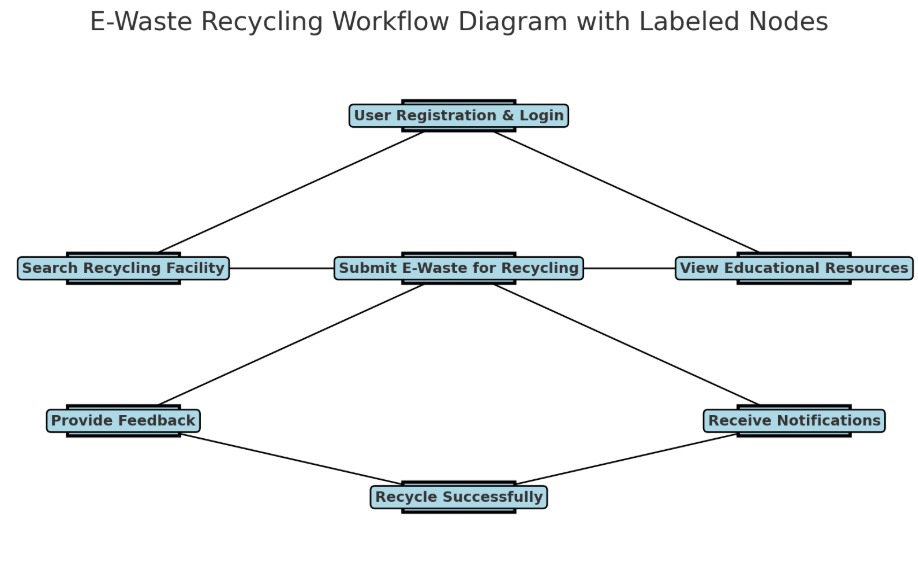
**5. Feedback & Reporting Module**

Technology: REST API with database storage.

**Flow:**

* User submits feedback after service completion.
* The system stores feedback in the database.
* Admin views reports for analysis.

**3.3 Architecture Diagrams**



**Conclusion**

The E-Waste Management & Recycling System aims to promote responsible e-waste disposal by making the process convenient and accessible. By providing pickup requests, a recycling center locator, educational resources, and real-time notifications, this platform empowers users to take action toward a cleaner environment. This document lays the foundation for the development process by defining system feasibility, requirements, and design structure, ensuring a well-planned and scalable implementation.