**EcoExchange**

# Lab 1: Problem Analysis

**Title: EcoExchange: A Sustainable Platform for Community Food Sharing and Waste Reduction**

1. **Description:**

A Community Food Exchange Platform solution developed that creates a digital marketplace for sharing surplus food items within local communities or between university students. The platform enables users to list, exchange, or donate excess food through a secure interface that includes features such as user verification, profile management, item listing, filtered searching, direct messaging, and transaction confirmation.

Food waste and food insecurity represent two significant challenges in our communities, with households discarding a substantial portion of purchased food while many individuals struggle to afford basic groceries. Traditional food redistribution systems often lack the flexibility and immediacy needed for perishable items. This platform bridges this gap by connecting those with excess food directly to those who need it, while simultaneously fostering community engagement in sustainability efforts. The solution particularly benefits students, residents, and temporary residents who may experience food surplus or scarcity at different times.

The platform implements a comprehensive security system for user verification and safe transactions. Users can create verified profiles, list available items, and browse offerings filtered by location and food type. The integrated chat system facilitates coordination between parties, while an administrative monitoring system ensures platform safety. The platform will provide real time notifications on exchanges and donations enabling users to respond accordingly. The solution includes algorithms to detect fraudulent activity and a reporting system for users to flag concerns.

1. **Problem Solutions:**

This application solution will be developed on top of a robust technology stack, featuring performance, scalability, and cross-platform ability. Its core programming language will be C#, as it is very object-oriented and will allow for developing a rich in features yet reliable application. SQL Server is intended to be used for the database, effectively storing all user information, food item details, and transaction records, and retrieving it when needed to maintain consistency and ensure security. The application framework shall be .NET Core: high-performance, cross-operational, and supporting a modular architecture that easily maintains upscaling. All put together, this technological stack forms a solid base for an efficient and agile solution that would meet the demands of its end-users, ranging from students to locals to tourists.

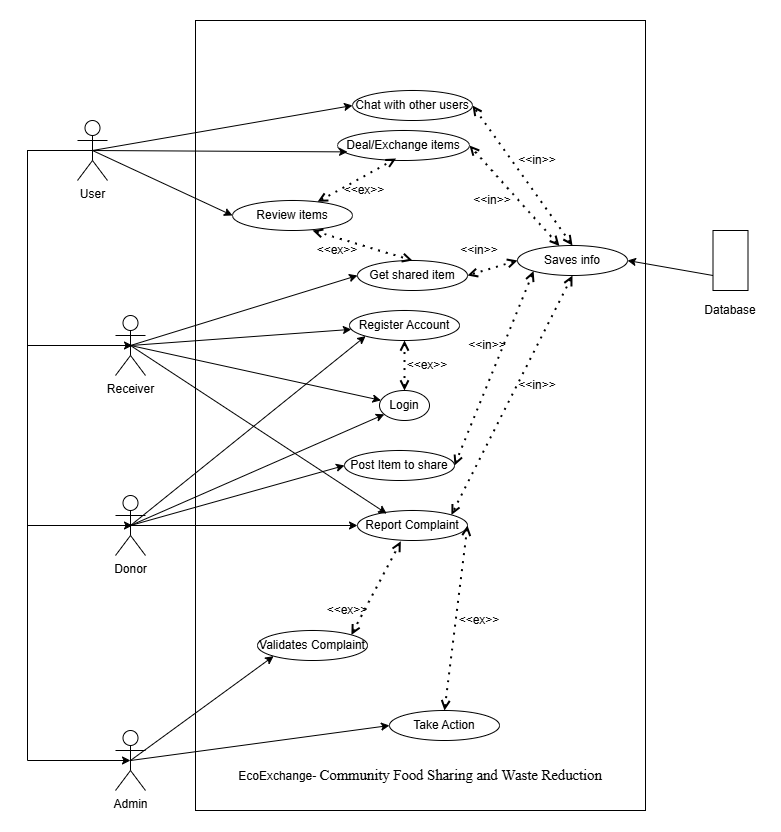
1. **Problem Statements:**
2. **Food Waste Reduction:** A significant portion of food purchased by households goes to waste due to overbuying or mismanagement of groceries. This platform solves this problem by providing a simple, accessible way for individuals to share or donate their surplus food before it expires, significantly reducing the volume of food waste in the community.
3. **Food Insecurity:** Many individuals and families face food insecurity, struggling to afford basic groceries, particularly in low-income communities. This platform provides a direct and dignified solution, allowing users in need to access free or low-cost surplus food from their neighbors, helping to alleviate hunger and reduce the stigma often associated with food assistance programs.
4. **Lack of Community Engagement in Sustainability:** Despite growing awareness of sustainability issues, there are limited local, community-driven solutions that encourage residents to actively engage in reducing waste. This app fosters a culture of sharing, collaboration, and responsible consumption by offering a platform where users can exchange or donate food, contributing to environmental sustainability and stronger community ties.
5. **Why is the idea important:**

By creating this platform, communities can not only reduce food waste but also build a culture of sharing and mutual support and it also encourages conscious consumption, helps those facing food insecurity, and create tighter bonds within the community.

1. **Objectives**:
2. **Food Waste Reduction:** A significant portion of food purchased by households goes to waste due to overbuying or mismanagement of groceries.
3. **Food Insecurity:** Many individuals and families face food insecurity, struggling to afford basic groceries, particularly in low-income communities.
4. **Lack of Community Engagement in Sustainability:** Despite growing awareness of sustainability issues, there are limited local, community-driven solutions that encourage residents to actively engage in reducing waste.
5. **Target Users:**

This application is designed to target three main groups of users such as students, residents, and tourists or casual users. Students, often managing groceries on tight budget, may lack access to high-quality ingredients, and by the help of this application, designed to enhance the sharing and utilization of resources can attract especially a member in smaller regional capacities. Residents can also use the tool to share surplus food or groceries directly with a neighbor, individual, or family in need which makes resources quickly accessible and available within one's own community. The app is also great for tourists and expats on short-term contracts who are planning a 1-3 month stay at best. The platform allows these users to donate any surplus food before departure, or access available resources while staying on site, reducing waste and encouraging sustainable practices.

# Lab 2: Defining Diagrams

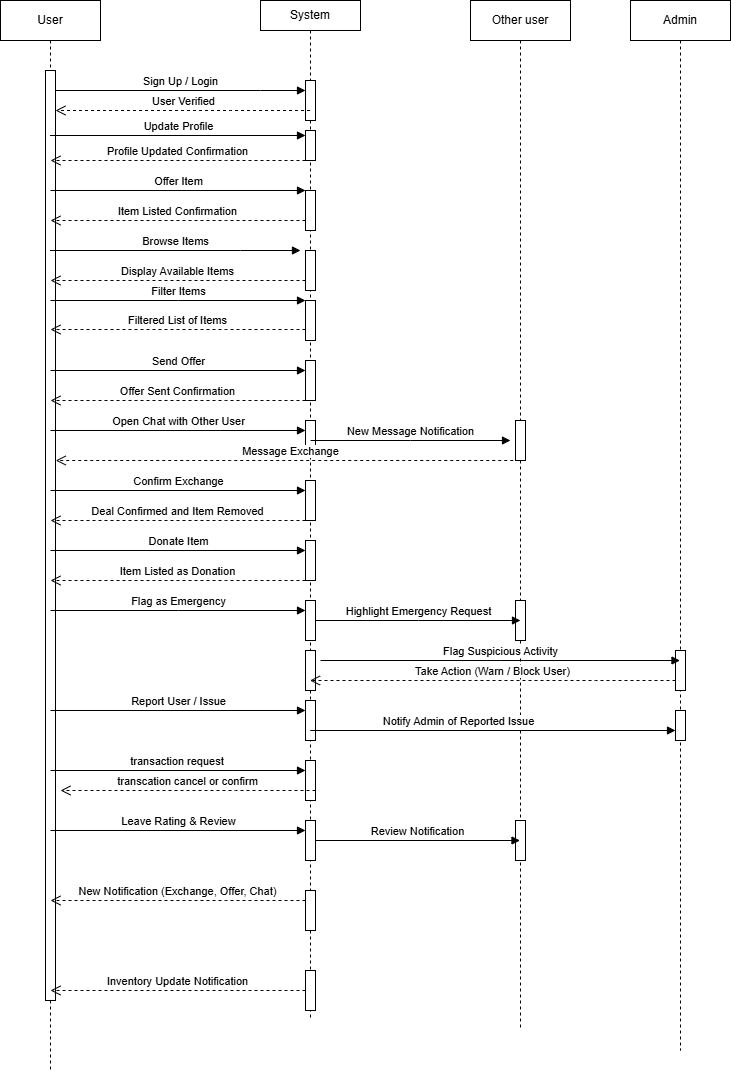


**Fig. 1: Use Case Diagram**

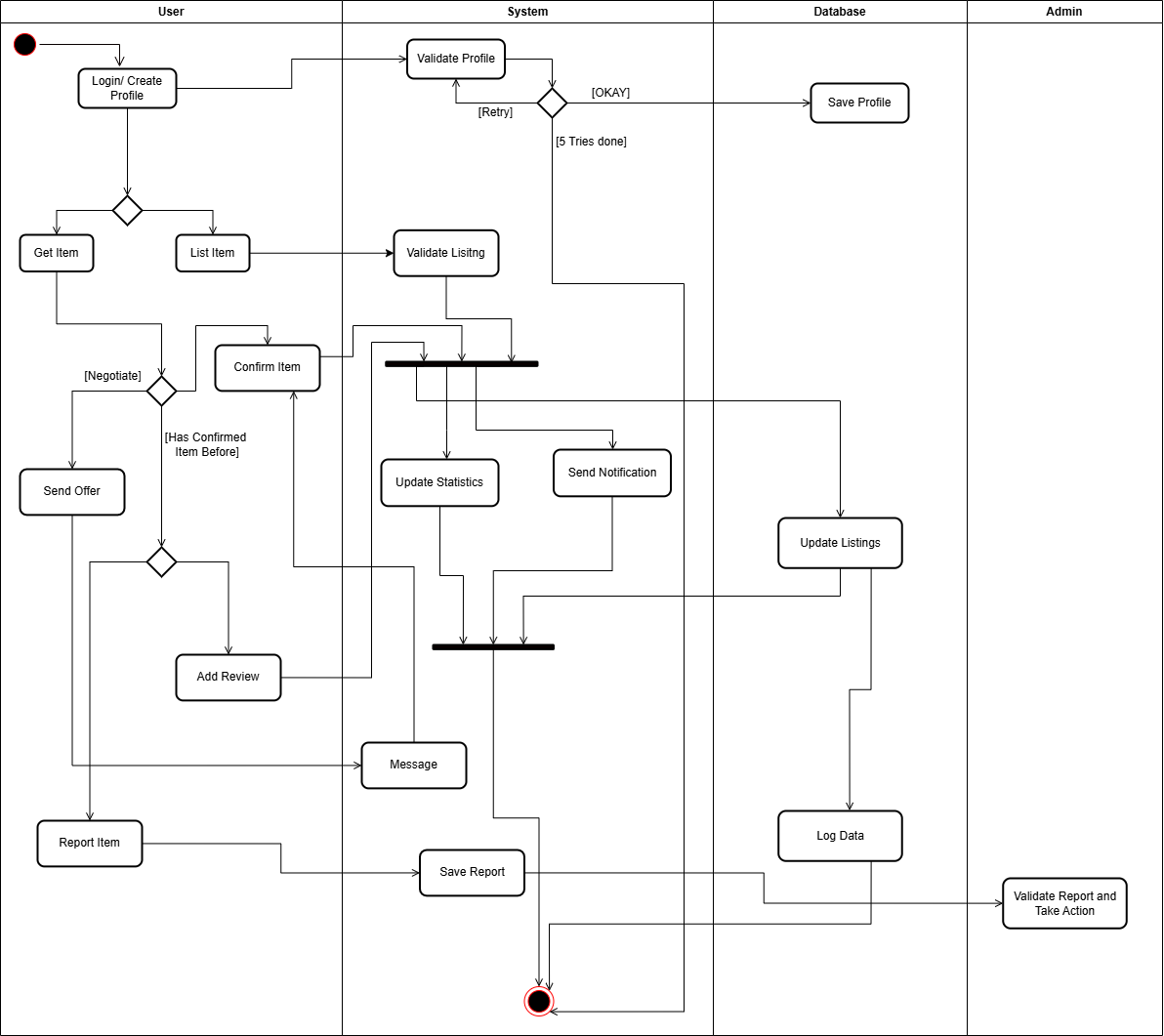
A diagram of a computer

Description automatically generated with medium confidence

**Fig. 2: Class Diagram**



**Fig. 3: Sequence Diagram**



**Fig. 4: Activity Diagram**

# Lab 3: Requirement Analysis:

1. **Basic Functionality:**

* **Profile Management**: Users will be able to modify profile pic/living area, other users can see this User’s profile statistics too.
* **Offer Item**: User can offer an Item they have in excess or surplus.
* **Browse Items**: User feed will be populated by items sorted by area.
* **Filter Feed**: Users will have options to search or filter by type of item/validity.
* **Send Offer**: Other Users can respond with an offer of their own.
* **Chat Option**: Users can directly message too to negotiate offers.
* **Confirm Deal**: Users can confirm the exchange. This will take the item off the marketplace and update statistics.
* **Donate**: Users can also offer Items at no cost.
* **Monitoring and Reporting**: Algorithms and Admins will be in place to monitor fraudulent activity and respond to the report.
* **Emergency Requests:** Allow users in urgent need of food (due to emergencies or temporary financial difficulties) to flag urgent requests, which could be highlighted on the feed.
* **Rating & Reviews:** Users can rate each other post-exchange and leave feedback, improving reliability and transparency on the platform**.**
* **Notifications tab**

1. **Functional Requirements**

**1. Profile Management**

1.1 The user will be able Login or Register with their credentials and important information.

1.2 The user can change their password by verifying their email.  
1.3 The system shall allow users to modify their profile picture and update their profile information.

1.4 The system shall display user profile statistics, which other users can view.

1.5 The profile will show reviews and ratings left by other users.

**Priority Level:**  High

**Precondition:** User is registered and logged in.

**2. Offer Item**

2.1 The system shall allow users to offer an item they have in excess or surplus.

2.2 Users will have the option to offer the item for nothing in return as a donation.  
2.3 The system shall enable users to enter item details, including name, description, and quantity.

2.4 User can set a priority level to indicate that item needs to be given away urgently.

**Priority Level:**  High

**Precondition:** User is logged in and has items to offer.

**3. User Feed**

3.1 Once a user is logged in, the system shall display a feed populated with items.

3.2 The feed will be customized based on the location.

3.3 Users can view items and their descriptions from the feed, which will include buttons such as send offer and message.

3.4 Users can sort the feed by categories the items are defined in.

3.5 There will be a search bar to search for specific items.

**Priority Level:**  High

**Precondition:** Items are available in the database.

**4. Send Offer**

4.1 The system shall allow users to send an offer in response to an item listed by another user.  
4.2 Users can select their own item to propose in exchange.

4.3 Users will add item description, quantity and other important information.

**Priority Level:**  High

**Precondition:** Users are logged in and browsing available items.

**5. Chat Option**

5.1 Once a user finds an item they want to barter, they can send a message to the other user.

5.2 The system will then allow them to message each other, bargain and agree on items to barter.

**Priority Level:**  Medium

**Precondition:** An active offer exists between two users.

**6. Confirm Deal**

6.1 Users shall be able to confirm an exchange, removing the item from the feed and updating profile statistics.

**Priority Level:**  High

**Precondition:** Both users agree on the terms of exchange.

**7. Monitoring and Reporting**

7.1 The system shall monitor user activity to detect potential fraud.

7.2 System also validates all offers sent and all items posted.

7.3 Users can report inappropriate behavior and fraudulent activity.  
7.4 Admins shall validate reports and suspicious activity to take corrective action.

**Priority Level:**  High

**Precondition:** System has active user engagement, and admins are available.

**8. Rating & Reviews**

8.1 The system shall allow users to rate and review each other after an exchange.  
8.2 Ratings and reviews shall be visible to all users to encourage transparency.

8.3 Ratings will be on a scale of 1-5 stars and users can also leave comments.

**Priority Level:**  Medium

**Precondition:** An exchange has been completed.

**9. Notifications Tab**

9.1 The system shall provide a notifications tab for updates on offers, alerts, confirmations, and messages.

9.2 Users can also see notification history to see recent events.

**Priority Level:**  Medium

**Precondition:** User is logged in.

1. **Non-Functional Requirements:**

**Usability:** The system should be easy to use, with clear options for offering, searching, and confirming items.

**Reliability:** The system must be reliable, with minimal downtime.

**Performance:** System response time for browsing and confirming items should be efficient.

**Security:** Ensure secure data handling for user information and activities.

**Scalability:** The system should handle increased demand, especially in emergency situations.

# Lab 4: System Design Specification

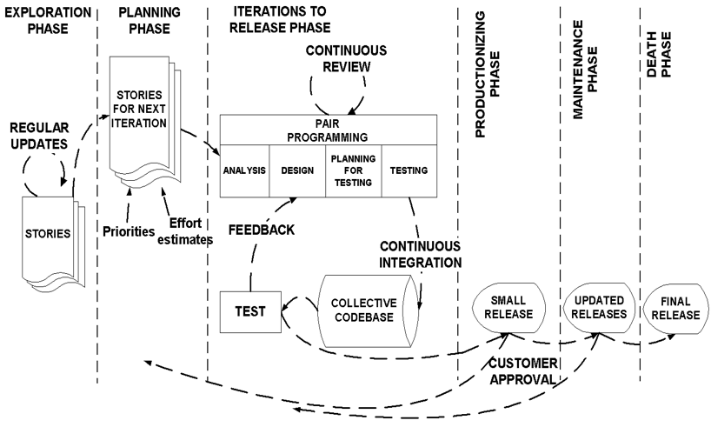
## Selection of Process Model: Extreme Programming (XP)

# Objective

The objective of this exercise is to select the Extreme Programming (XP) process model for the development of the EcoExchange platform, a community food sharing application. This document will present the reason for choosing XP, outline key project management roles, and describe their responsibilities.

# Overview of Extreme Programming (XP)

Extreme Programming (XP) is an Agile software development methodology that emphasizes customer satisfaction, flexibility, rapid iterations and feedback loop. It is particularly well-suited for web applications where requirements are expected to change frequently and where user involvement and maintenance is crucial.



## Reasons and Arguments for Selecting XP

* **Dynamic Nature of Requirements**:

The needs of users in terms of food sharing are highly variable, particularly in relation to food availability, demand, and urgency. These dynamic needs would require constant updates to the platform’s functionality and features. This is why an agile model is perfect for this application as agile models can have changing requirements. XP allows for quick iterations while maintaining a stable version of the application.

* **Community Feedback Involvement:**

The primary users of the **EcoExchange** platform (students, residents, and tourists) will have specific preferences for how food donations and exchanges work. Regular feedback from these diverse groups is critical to ensuring the platform meets their needs. XP emphasizes close, continuous interaction with customers to gather frequent feedback. XP's approach ensures **EcoExchange** can incorporate feedback swiftly, ensuring the platform stays user-friendly and aligned with real-world needs.

* **Iterative Development and Platform Evolution**:

**EcoExchange** is a living platform, and new features will be continually added to meet user demands. XP’s short development cycles (1-4 weeks) allow for frequent updates, which is essential for **EcoExchange** to adapt to changing user needs. Features like adding urgent donation tags or integrating real-time notifications on food availability can be implemented quickly, tested with users, and refined in the next iteration while maintaining a stable version of EcoExchange.

* **Security and Reliability for Sensitive Information**:

Since **EcoExchange** handles user profiles, location data, and potential transactions (in cases where users trade items for low-cost food), security is paramount. Ensuring that users can trust the platform is essential for adoption and long-term success. Features must be thoroughly tested for both security and usability. XP’s Test-Driven Development approach ensures that these critical features are robust from the start as expected while maintaining continuous integration so that the main functionalities are available.

* **Team Collaboration and Communication**:

**EcoExchange** relies heavily on user-to-user interaction, from **bartering food** to **chatting about exchange details**. This requires seamless communication between the platform’s front-end (user interface) and back-end (database management), with real-time updates on offers, transactions, and messages which can be complex for engineering teams to understand. It requires the team to be familiar with a particular tech stack to be able to roll out features and solve previous problems. XP encourages pair programming and collaborative development, making it easier for developers to integrate complex features.

* **Priority Features**:

EcoExchange has its core requirements graded on priority levels such as high, medium and low. XP advocates for dividing requirements into regular iterations so the application can have its high priority features first.

* **Long-Term Scalability**:

As **EcoExchange** grows, both in terms of user base and geographical coverage (across universities, neighborhoods, or even cities), scalability will be crucial. Additionally, as users continue to interact with the platform and submit new features or bug reports, it will be essential to manage the codebase efficiently. XP's focus on maintainable code, simplicity, and frequent releases ensures that **EcoExchange** can scale effectively. The **modular approach** used in XP also makes it easier to integrate new features without compromising the platform’s performance or usability.

## Roles and Responsibilities

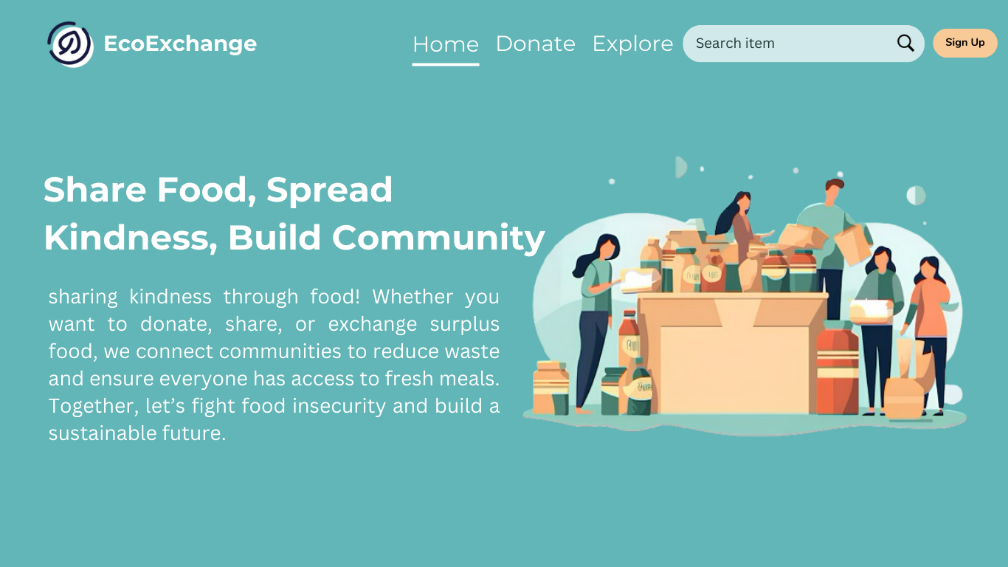
1. Requirement Analyzer
   * Decides when each requirement is satisfied.
   * Sets the implementation priority for requirements.
2. Project Manager (Big Boss)

* Oversees overall project direction and resource allocation.

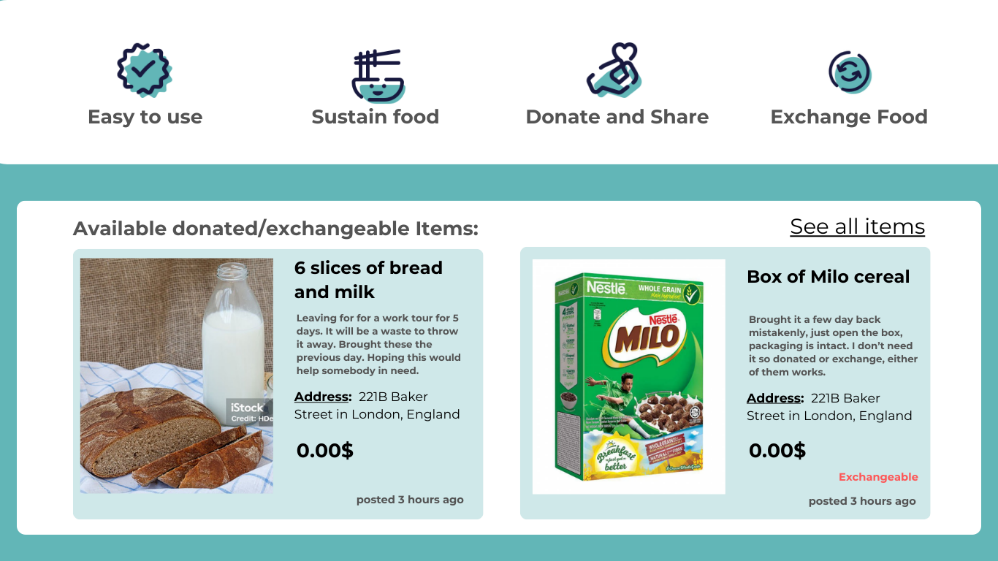
1. Programmer
   * Focuses on writing clean, maintainable code.
2. Tester
   * Assists the customer in writing functional tests.
   * Runs functional tests regularly to ensure quality.
   * Broadcasts test results and maintains testing tools
3. Tracker
   * Evaluates the accuracy of estimates to improve future planning.
   * Monitors the progress of each iteration and assesses whether goals are achievable within resource and time constraints.
4. Coach
   * Ensures that all team members understand XP principles and processes.
5. Consultant
   * Provides guidance on technical challenges and best practices.

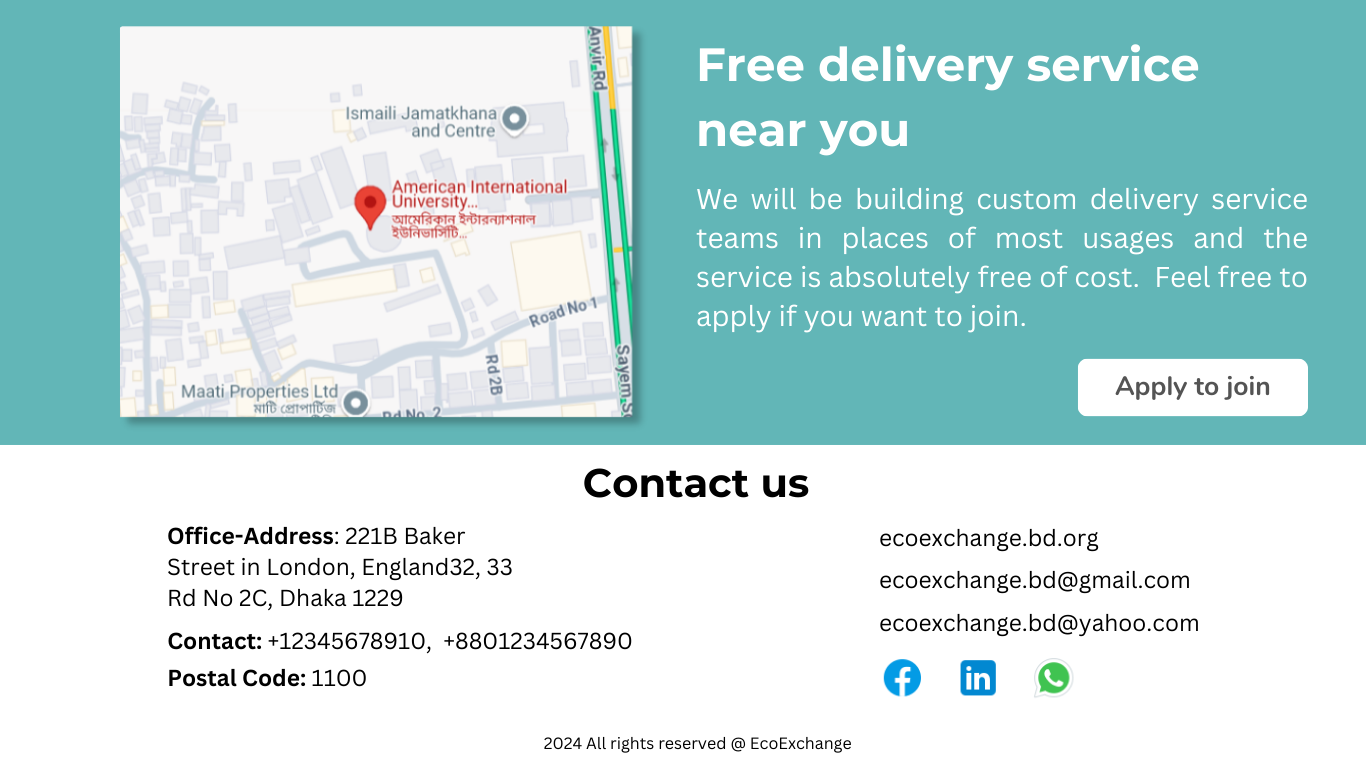
# Lab 5: UI/UX Design

1. **Home Page:**

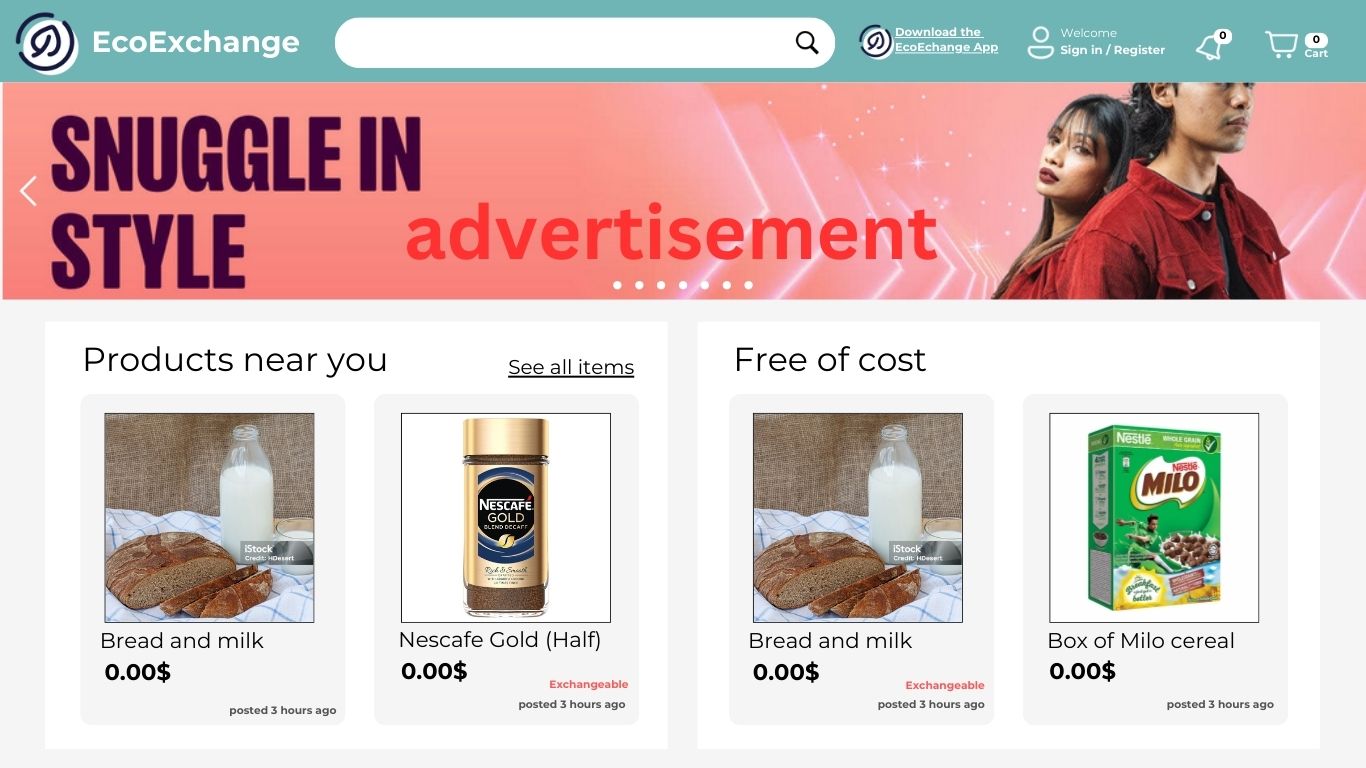


A screenshot of a computer

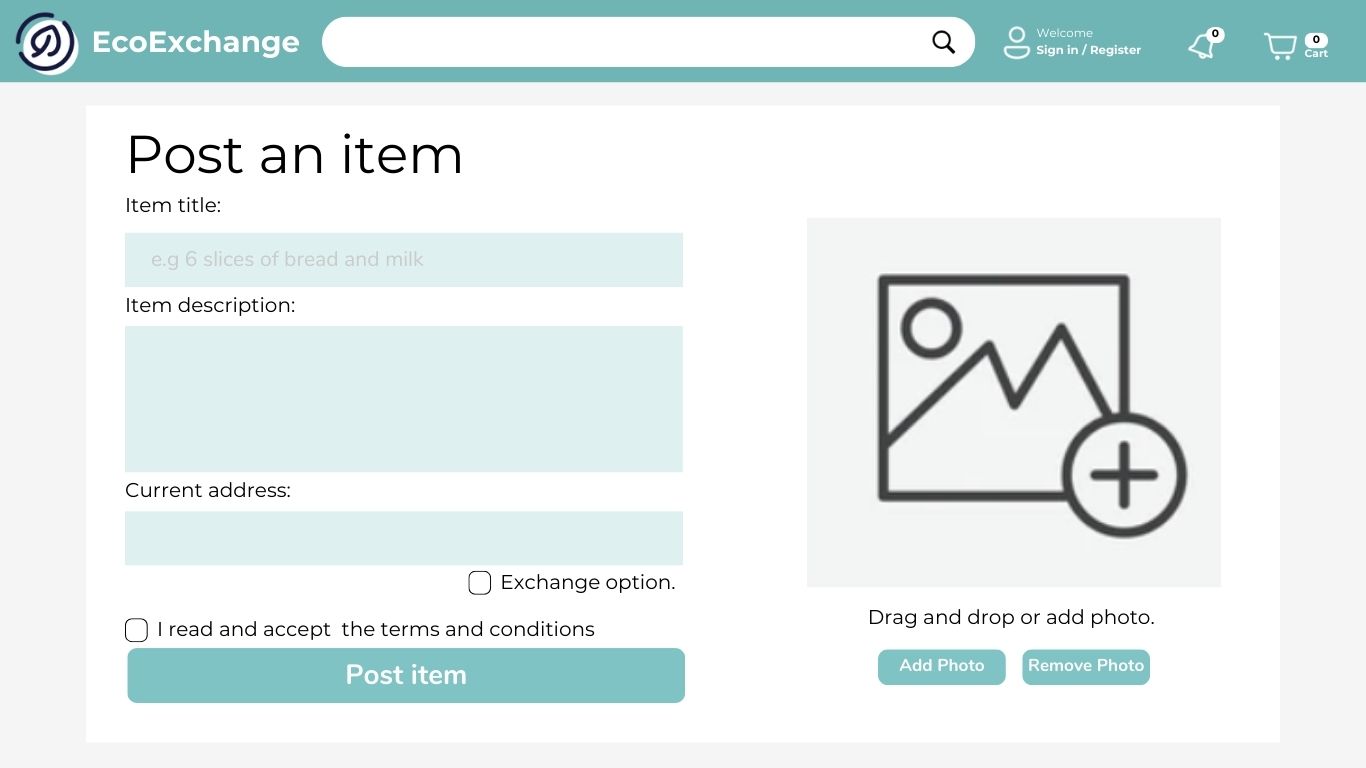
Description automatically generated



1. **Explore Page:**



1. **Post Item:**



1. **Get Item:**

