Tampere University
Tampere University of Applied Sciences

COMP.SE.221 Sustainable Software Engineering	Group HeroEaters
Group Project: HeroEats App – Mobile app for rescuing food	Tuan Nguyen Anh Tran Sang Nguyen

HeroEats App – Mobile App for rescuing food

Table of Contents

1. Introduction

HeroEats is a mobile application designed to reduce food waste by connecting users with local restaurants, cafes, stores, etc offering surplus food at discounted prices. The goal is to reduce food waste, for example unsold food items at the end of date, or near-expiry items. The app promotes sustainable consumption by rescuing food that would otherwise be thrown away.

2. Design

The target users of the app are environmental-conscious individuals, budget-conscious students or workers, restaurants with daily leftover meals. Main focuses are reducing food waste, encouraging local food redistribution, and raising awareness on sustainable eating habits.

The design of HeroEats is focused on delivering a clean, friendly, and intuitive experience to users, reflecting the app's mission of sustainability and simplicity. The visual language emphasizes soft colors, rounded elements, and thoughtful spacing to create a modern yet approachable aesthetic.

The app should have easy navigation with minimal layout to reduce cognitive load. It has warmth, positivity vibe with earthy, eco-friendly color palette to inspire trust and reflect sustainability. The app is fully responsive and interactive with buttons, navigation bar, search, etc. Color palettes used in the app are #16423C, #6A9C89, #C4DAD2, #E9EFEC.

3. Layout and Features Overview

3.1 App Layout

Welcome Screen:

- Simple animation and logo to make the first impression calming and inviting. Auto-redirects based on login status.

Login/ Register:

- Minimal input fields with rounded corners and warm green buttons. Includes subtle hover/tap feedback.

Explore Screen:

- Search bar with filters
- Categories
- Grid of product cards showing images, prices, and names. Each card features a heart icon to favorite items.

My Orders Screen:

- Active orders and completed orders using switchable tab views
- Cards of items showing images, restaurants, prices, etc. with status of order

Favorite Screen:

- Displays saved items and restaurants in a scrollable list
- Allows removing favorites with a single tap

Cart Screen:

- Cards of items in the card waiting for checkout
- Checkout button navigating to payment gate
- Allows deleting items, adding more items

Profile Screen:

- Includes user avatar (with optional image picker).
- Credits for later discounts in the app.
- Profile information, order history, settings
- Log out button to log out the app

3.2 Main Features

Main features of HeroEats:

- Search products based on category and filters based on price, distance, rating
- Purchase orders with Paypal and Stripe
- Authentication, log in automatically, log out due to session and sign-up new account
- Check order status and history
- Add items and stores/restaurants to a favorite list
- Dark mode/ Light mode and change profile picture

Basic features:

- Local database integration
- Image handling
- Localization

Advanced features:

- Authentication
- Cloud database
- Camera

4. Functional Description

4.1 Architecture

HeroEats architecture includes backend, frontend and database.
The backend is structured for modularity, testability, and easy data management. Built using Java and the Spring Boot framework, it supports integration with multiple payment gateways, provides test utilities, and separates logic clearly across packages.
The frontend is developed using React Native+Expo with TypeScript and JavaScript, following a modular structure that promotes maintainability, reusability, and scalability. The architecture adheres to sustainable software development principles by encouraging clean separation of concerns and reusability of UI components. Key folders are api, app, assets, components, hooks, mocks, utils.

4.2 Structure Overview

The main actors are users, products, and stores. Main navigation flows can be:

- /welcome -> auto check login -> /login
- /explore -> browse food items -> search -> click to explore items -> add to cart -> make payments
- /explore -> browse food items -> add items to favorite -> /favorite -> modify favorite items
- /explore -> browse food items -> /my-orders -> browse active orders and order history

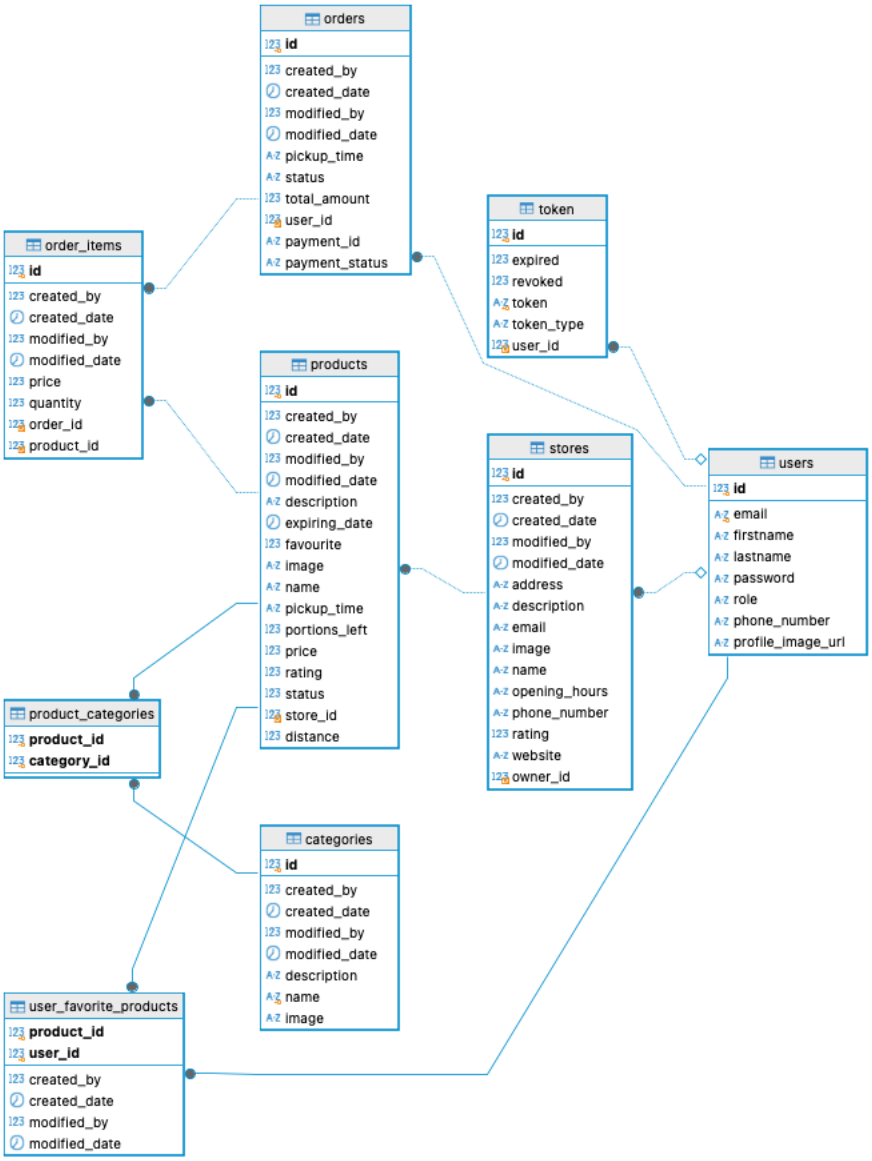

Figure 1. Class diagram of the app.

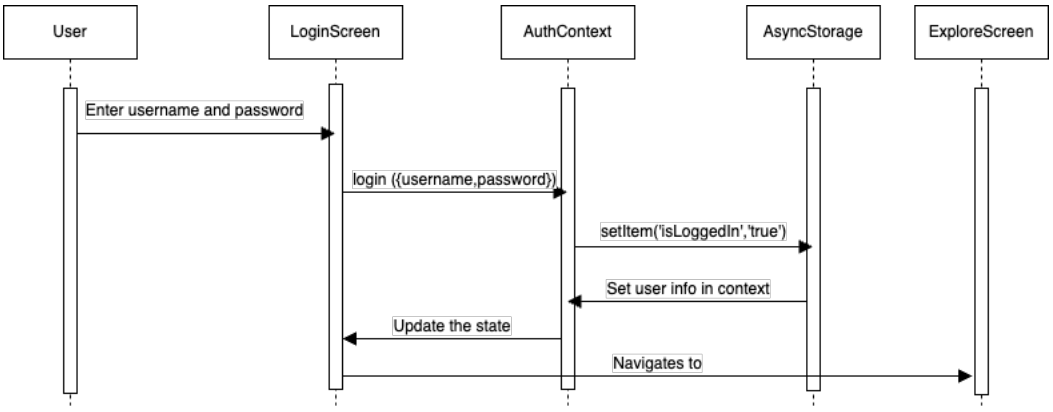

Figure 2. Sequence diagram when user logs in the app

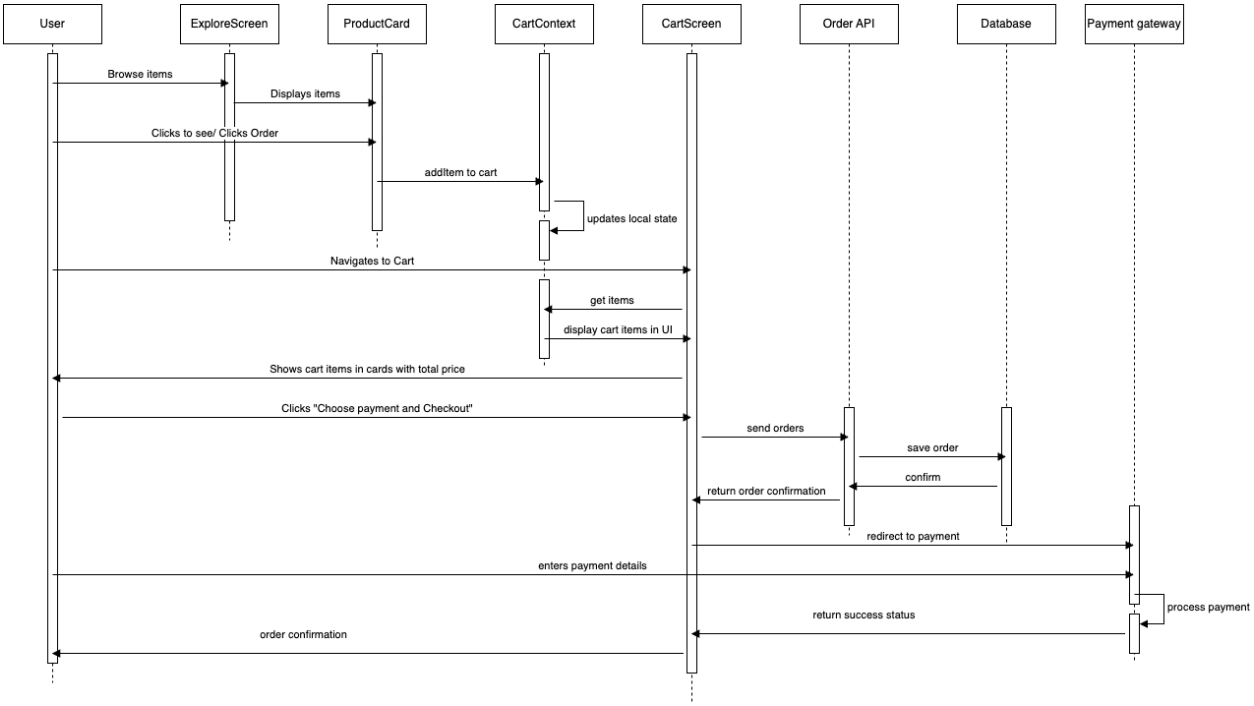

Figure 3. Flow of exploring items and making payments in the app

5. Technology and Implementation

5.1 Technology Stack

Backend: Java, SpringBoot,
Frontend: React Native (with Expo), Tailwind, Axios (APIs)
Storage: AsyncStorage
Data: Mysql Database

5.2 Work Division

Name	Role
Anh Tran	UI/UX, Figma sketch, frontend views Welcome, Login, Explore, Favorite, Profile, documentation
Tuan Nguyen	Backend + database, frontend views Orders, Cart, payment, order history
Sang Nguyen	Frontend APIs, data integration, loading indicator, video

5 User Guide

Home Flow

1. User opens app -> Welcome screen auto-checks login.
2. Logged in -> Redirected to Explore.
3. Not logged in -> Goes to login/register.

Core Features

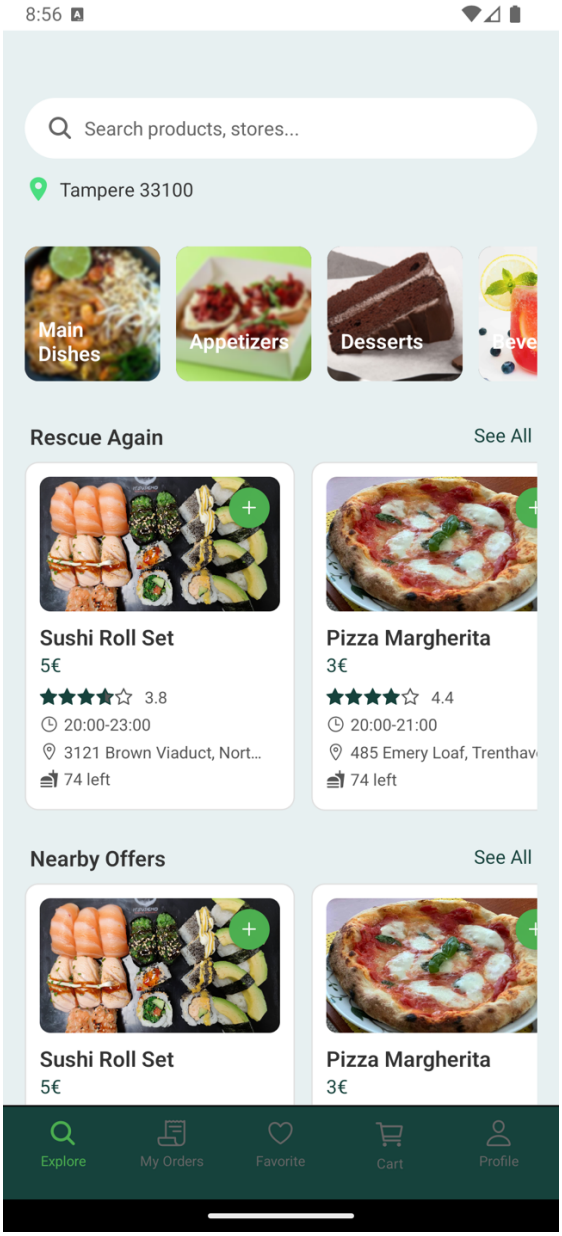

Figure 4. Explore: browse items, search with filters, click to see items details

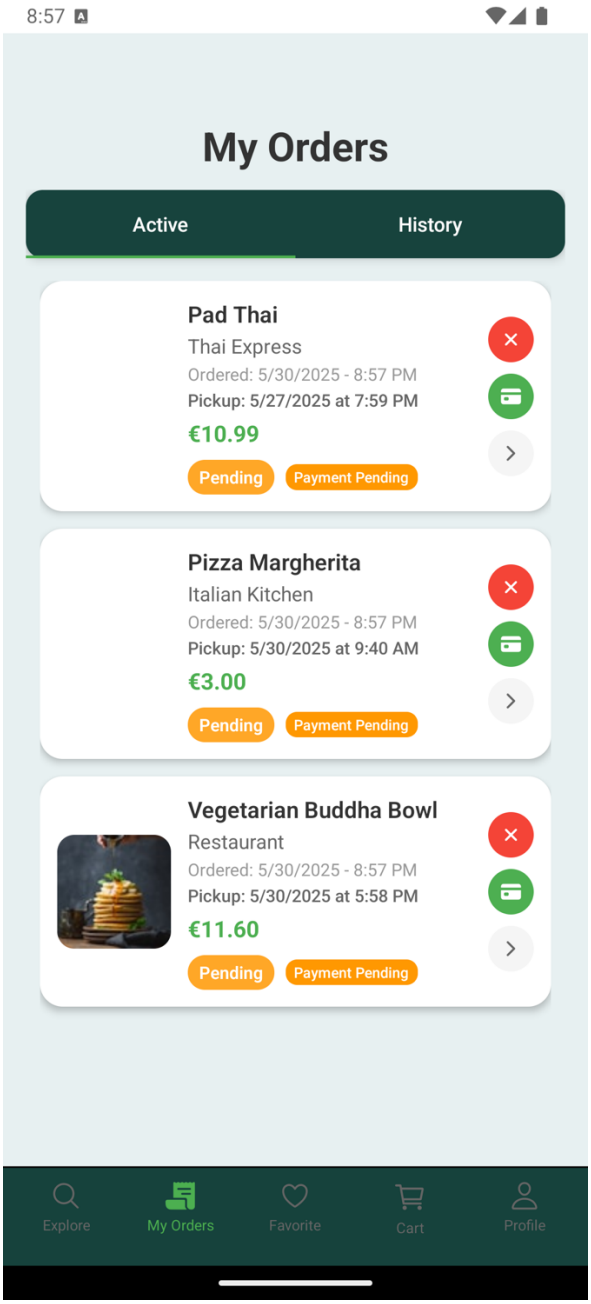

Figure 5. My Orders: see active orders/ order history

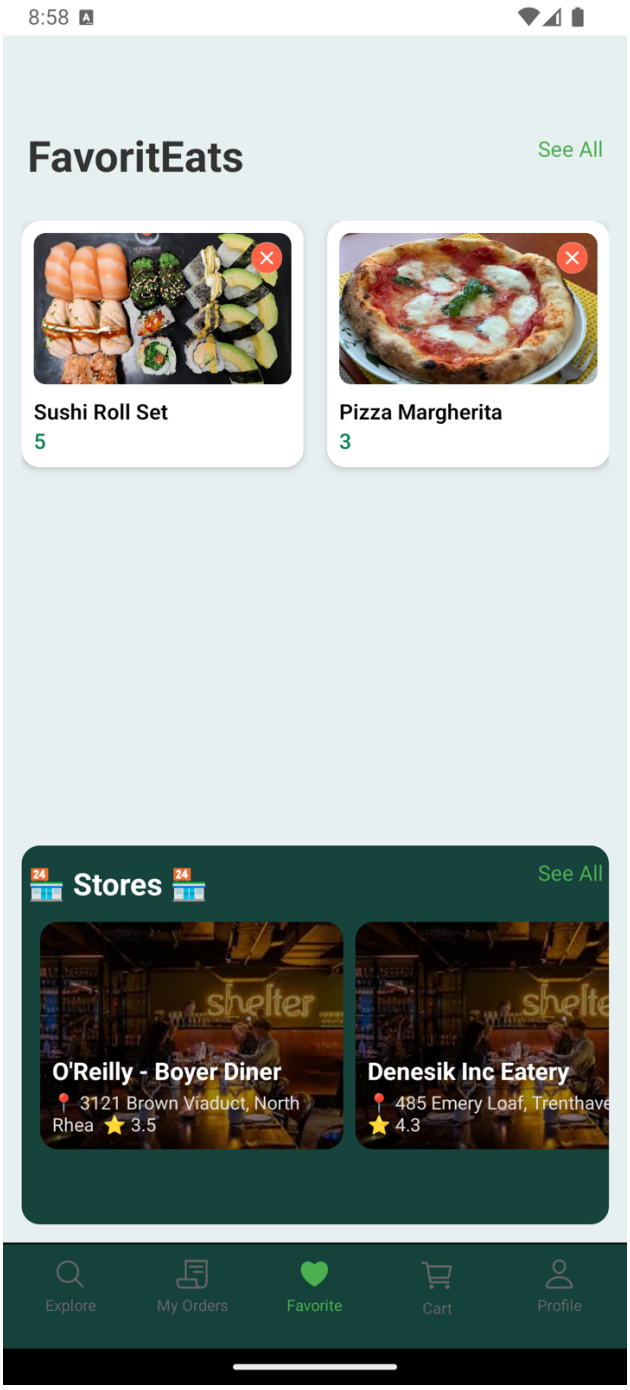

Figure 6. Favorite: see favorite offers and restaurants, click restaurant to see more info with ongoing items, search favorite based on categories and price/rating, add and remove favorite items

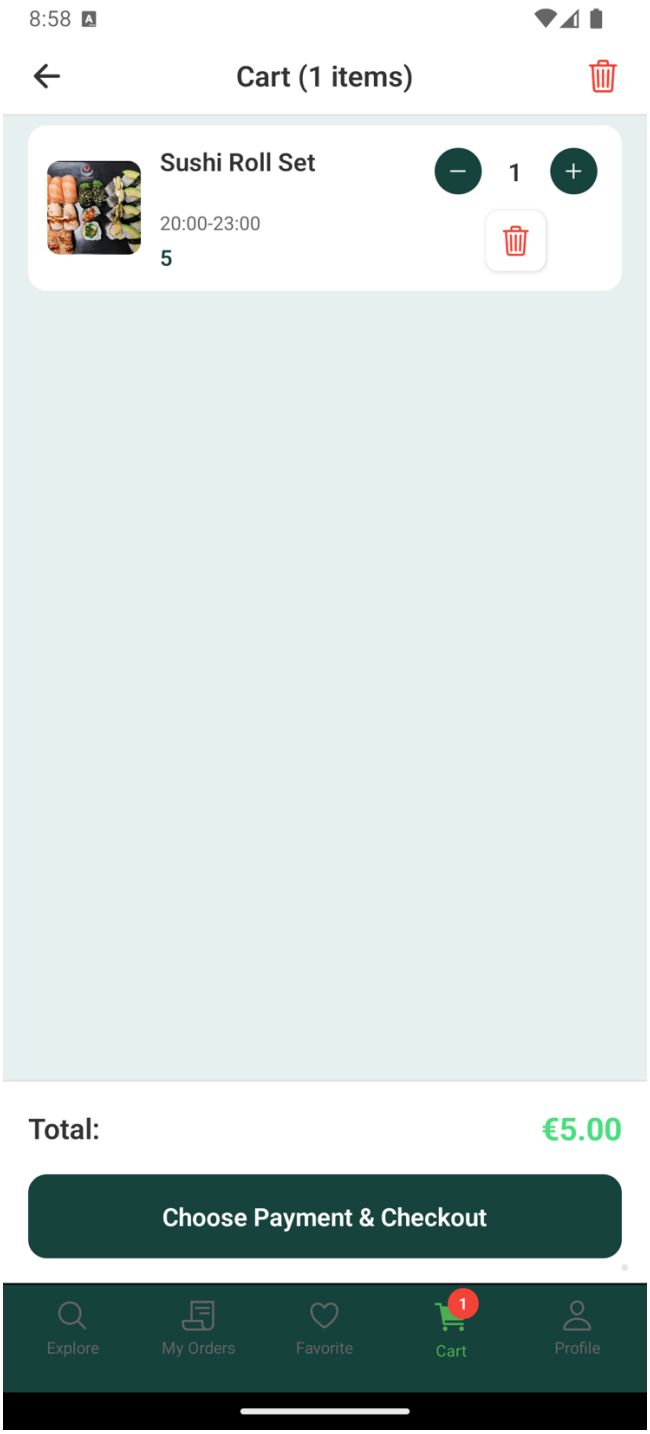

Figure 7. Cart: see offers added to cart, total, modify number of items, proceed to payment

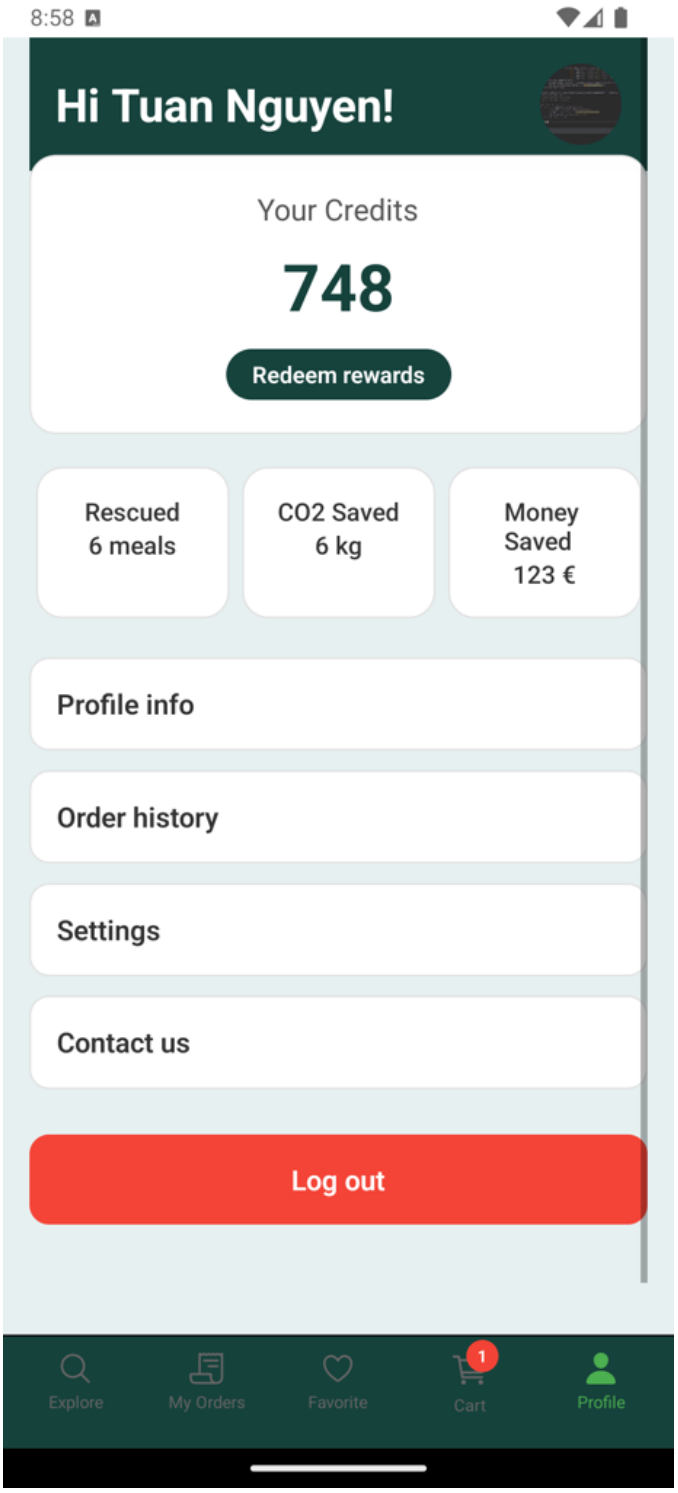

Figure 8. Profile: modify profile picture and information, see order history, settings, log out from the account to redirect to log in again

6 Installation and Testing

GitHub repo: https://github.com/trnvanh/sustainable_project.git
Requirements: Node.js, Expo CLI
Steps:

1. Clone repo from GitHub:
   git clone https://github.com/trnvanh/sustainable_project.git
2. Redirect to Front_end:
   cd sustainable_project/Front_end
3. Run: npm install
4. Launch app: npx expo start
5. Test the app on emulator or at localhost:8081/

Testing:

- Check login flow
- Navigate between tabs
- Search items
- Add items and proceed to checkout
- Save and remove favorites