

Mobile Application Using Interactive QR for Smart Recycling

Authors: Tristan Harris, Athen Jumper, Hasibur Rahman, Xavier Scott, Kentaro Watts, Hongchao Xue
Mentor: Dr Lindsay Wells



UNIVERSITY of
TASMANIA

Introduction

Recycling is an ever-important part of our waste management. Many people do not accurately recycle because of lack of information or motivation. This makes it harder to measure data around recycling. Due to the ubiquitous nature of mobile smart phones and QR code scanning technology, people have the ability to record and submit data without manually entering in information, therefore the project is based around utilising QR codes to share data.

The system uses QR codes on bins, real-time location tracking, and a reward model to encourage responsible waste disposal. Users scan QR codes to log recycling actions, which are validated based on proximity. Verified submissions are incentivised with tokens that can be redeemed for rewards.

Project Aims

The aim of this project was to utilise QR codes to streamline a digital submission system for recycling objects, to gamify the recycling process, and to develop a functionally powerful database and API to power the application front-end.

Ease of Use

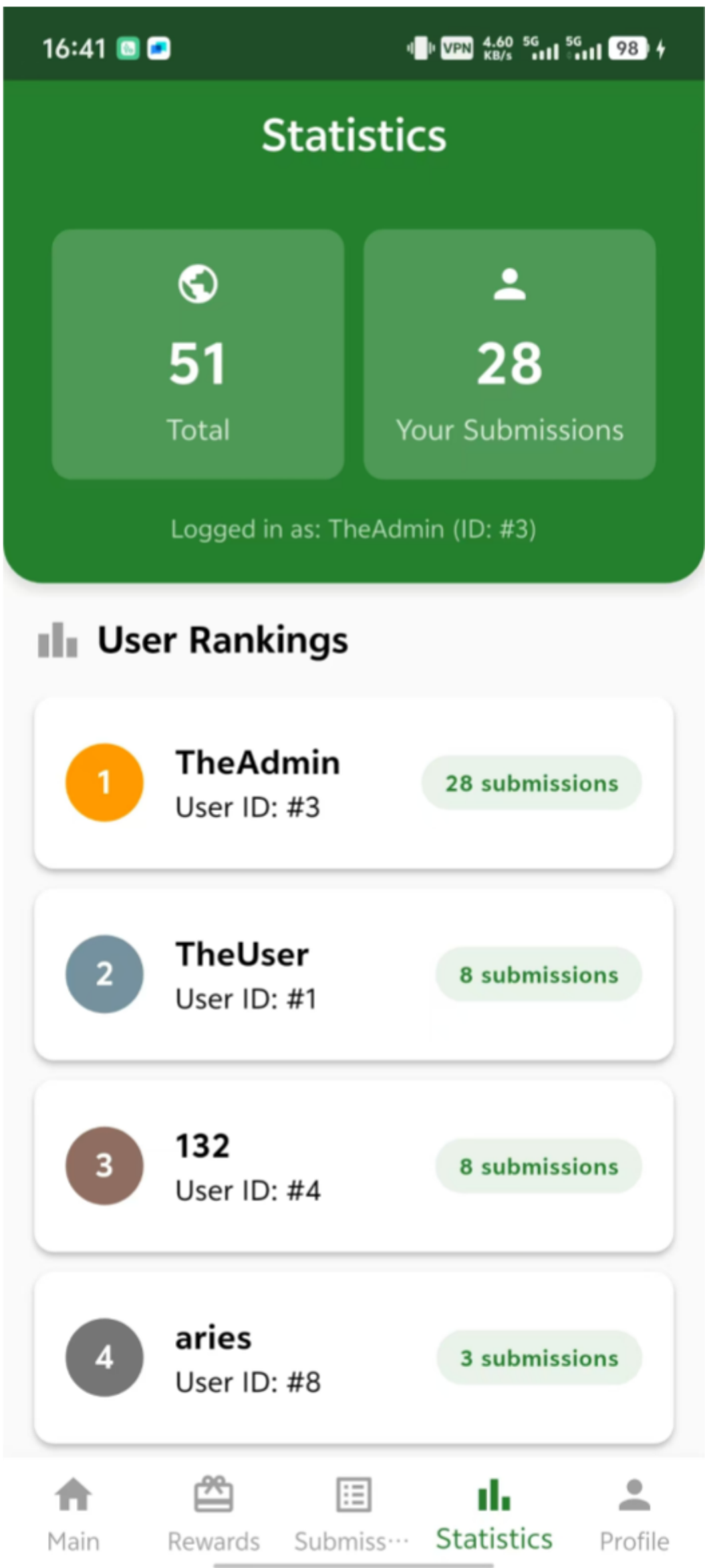
The application will automatically derive information for the user where possible. The application should be intuitive for all user roles and functions, with straightforward navigation and an affording user interface.

User Permissions

User roles will control access to application features such as submission data details.

Front-End Mobile Application

- Built with Flutter, enabling cross-platform support for both Android and web versions of the application.
- Implements QR code scanning with real-time location validation to ensure users are near the correct bin before submitting.
- User roles include User, Moderator, and Admin, each with specific feature access.
 - Users: scan bins, submit recycling data, claim rewards.
 - Moderators: review and approve submissions.
 - Admins: manage rewards, user data, and system content.
- JWT authentication secures user sessions and enforces role-based navigation logic.
- Application pages include:
 - Home Screen with access to core features.
 - QR Scanner that checks user distance before allowing form submission.
 - Submission Form to log recycled objects with optional details.
 - Rewards Page showing redeemable items based on accumulated points.
 - Admin Dashboards for user, bin, object, and reward management.
 - UI optimized for accessibility with simplified layouts, labelled buttons, and color contrast compliance.
- Backend integration handled via HTTP requests using token-based headers and JSON data handling.



The Statistics page with moderator/administrator view

System Backend

- Backend REST API implemented using the Python web framework Flask
- SQL database using MariaDB
- Hosted on Debian VPS provided by Nectar Cloud

REST API

- The Flask web application offers a REST API that is used by the Flutter frontend to indirectly query and modify the database.

Data Models

- The backend implements the logic of the system by defining various data models, such as User, Bin, Recyclable and Reward.
- Coordination between the backend and database is achieved using the Python ORM library SQLAlchemy.

Authentication

- The backend uses JWTs (JSON web tokens) to manage user authentication and to distinguish between users, moderators and administrators.

Distance Validation

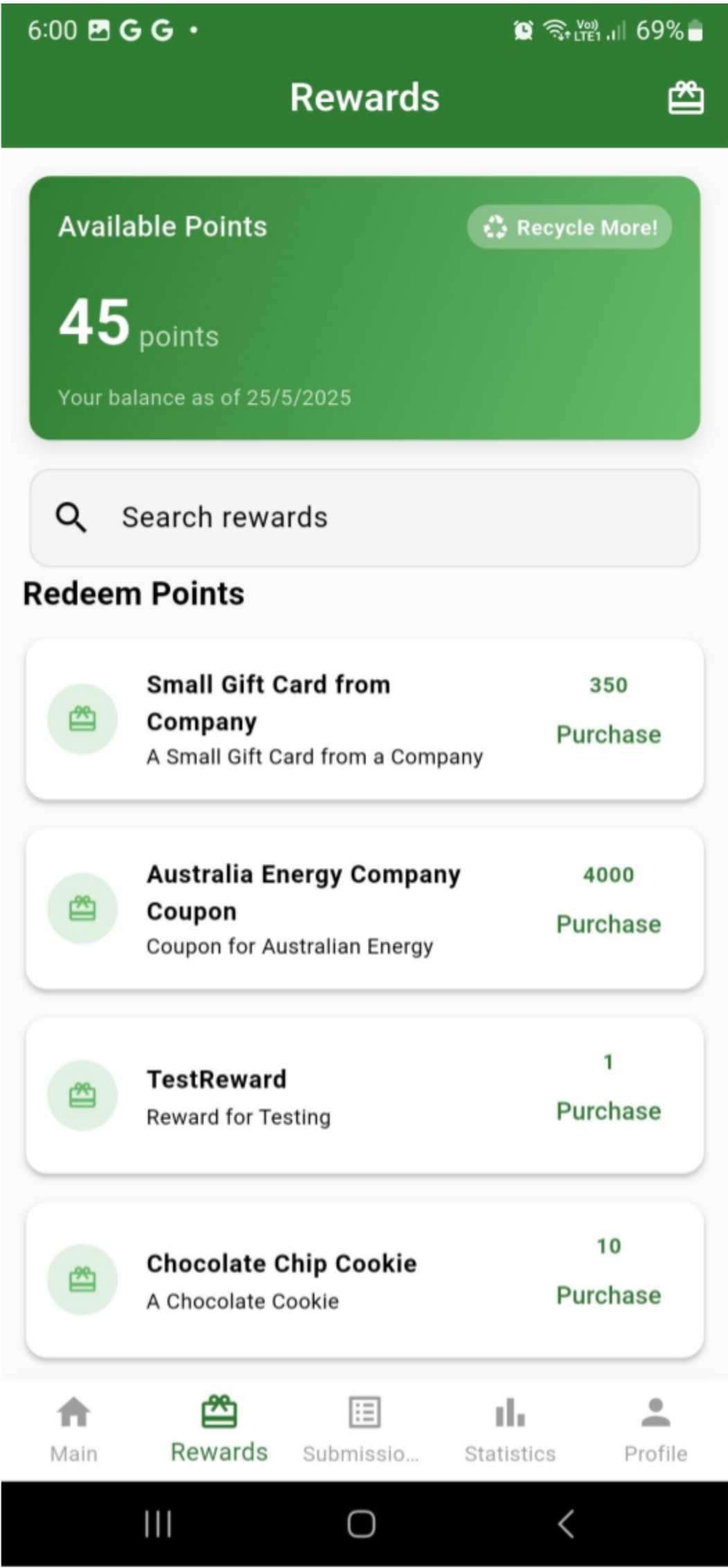
- The backend uses the Haversine formula to calculate the distance in metres between two coordinates, to ensure that the user is making a recycling submission near the specified bin.

Testing

- Integration and unit tests have been written for each of the endpoints presented by the backend, to minimise the potential for future bugs. This was achieved by integrating the Python testing framework pytest.



System Diagram



The rewards page, where users can redeem points from recycling. See if you can earn enough points for a chocolate chip cookie!