Task 4 – Elicitation Execution



**Project**

CSE6224 Software Requirement Engineering   
Term 2510

|  |  |
| --- | --- |
| **Tutorial Section** | TT2L |
| **Group** | Group E |
| **Project Title** | Campus Ride-Sharing Platform with  Parking System Integration |

|  |  |
| --- | --- |
| **Name** | **Student ID** |
| Kelven Yee Kai Wen | 1211111244 |
| Koh Xuan Lin | 1211109618 |
| Ow Ka Sheng | 1211108820 |
| Shazreen Binti Sheridan | 243UC247P3 |

## **1.0 Elicitation Technique Overview**

We employed three elicitation techniques to align with our project goals:

## **1. Questionnaire**

* Distributed to campus users to gather expectations and feature suggestions.
* Applied the **Kano Model** to categorize responses into dissatisfiers, satisfiers, and delighters.

## **2. Observation**

* Analyzed existing ride-sharing apps (**Kummute** and **BlaBlaCar**).

## **3. Prototyping**

* Developed early prototypes to visually and interactively demonstrate potential system features.
* Enabled stakeholders to better understand and validate system requirements.

## **2.0 Questionnaire**

## **2.1 Overview**

To gather stakeholder insights for the proposed Campus Ride sharing and Parking Integration platform, we conducted a structured questionnaire using Google Forms. The primary aim was to assess user expectations and perceptions regarding the system's proposed features, with the goal of classifying them using the **Kano Model**, a framework that helps prioritize features based on how they influence user satisfaction.

The questionnaire consisted of **21 questions** in total:

* **20 main questions** were designed to evaluate **10 key features** of the system.
  + For each feature, two types of questions were asked:
    - A **functional question** (how users feel when the feature is present)
    - A **dysfunctional question** (how users feel when the feature is absent)
  + This structure enables us to categorize features into **Delighters**, **Satisfiers**, or **Dissatisfiers**, based on users' emotional responses and expectations.
* **Question 21** was an open-ended item inviting respondents to share **suggestions or ideas for system improvement**, providing qualitative feedback that supports further refinement of our platform.

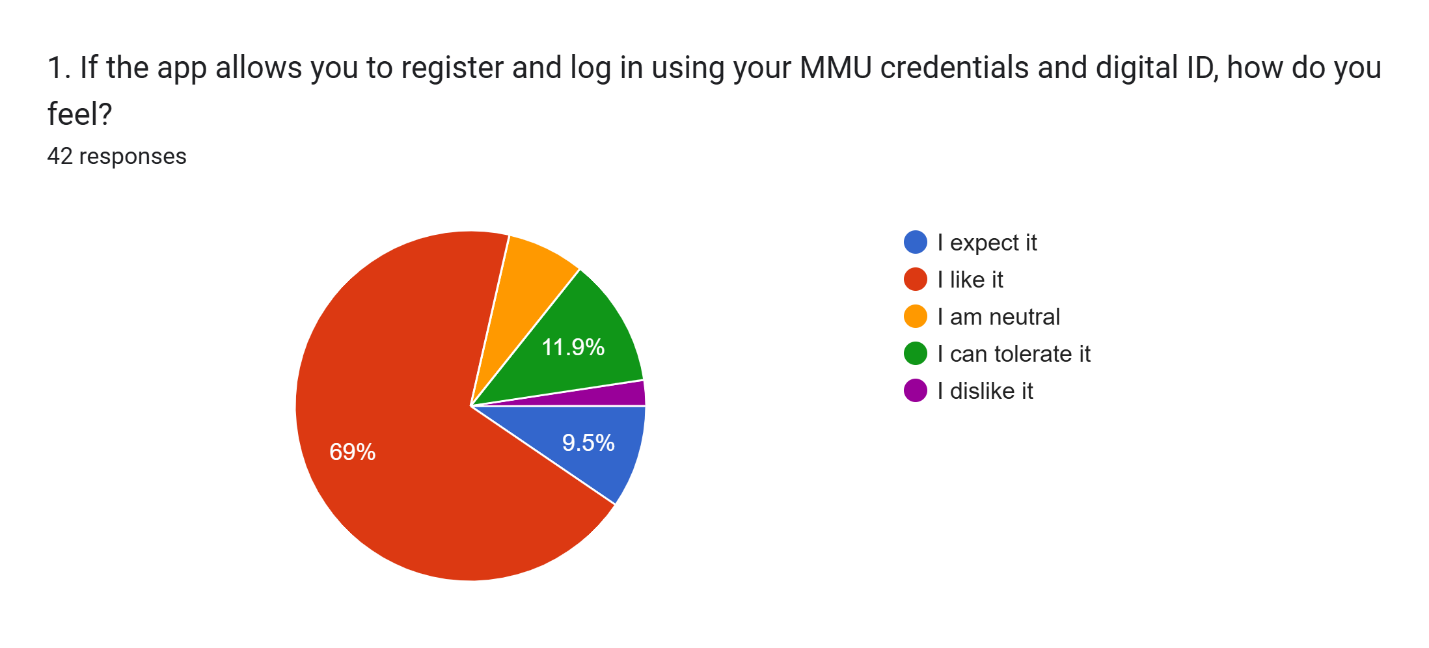
## **Result of Questionnaire**

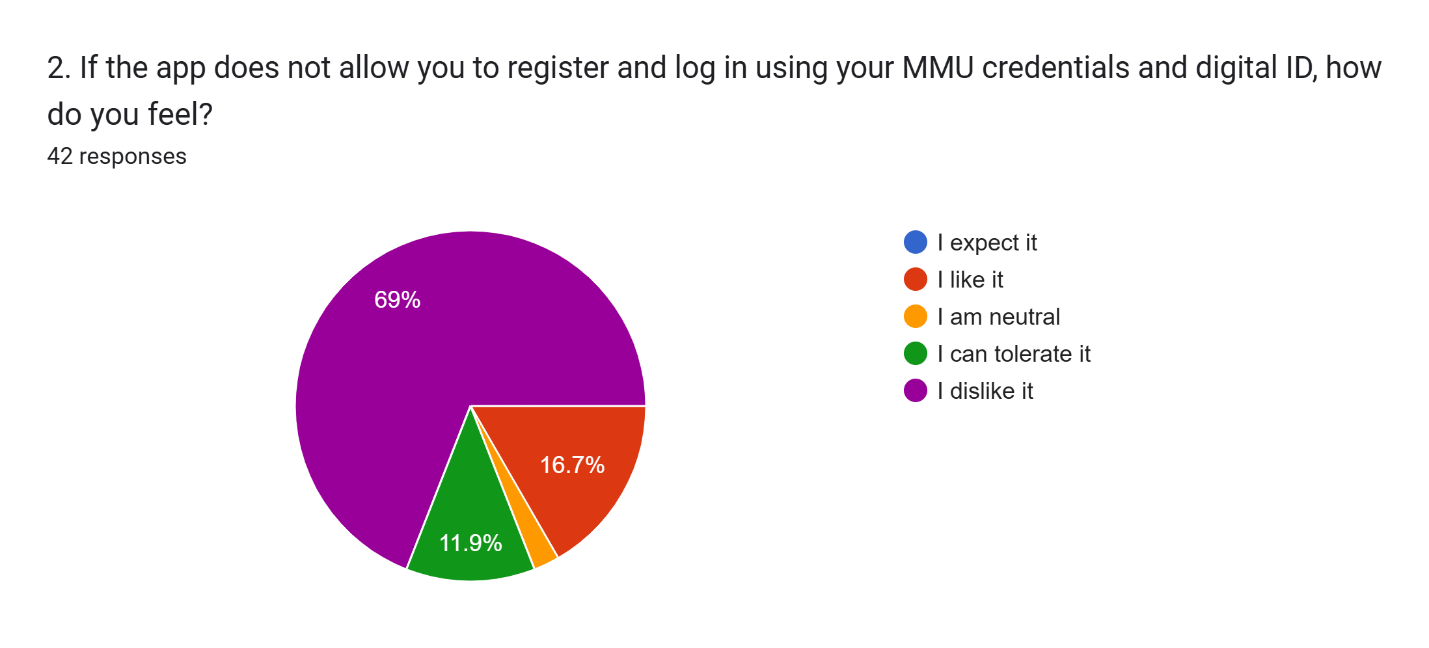
A total of 42 respondents have submitted the Google Form.A screenshot of a computer

AI-generated content may be incorrect.

*Figure 2.1 Google Form Respondent*

## **2.2.1 User Onboarding & Digital ID Verification**

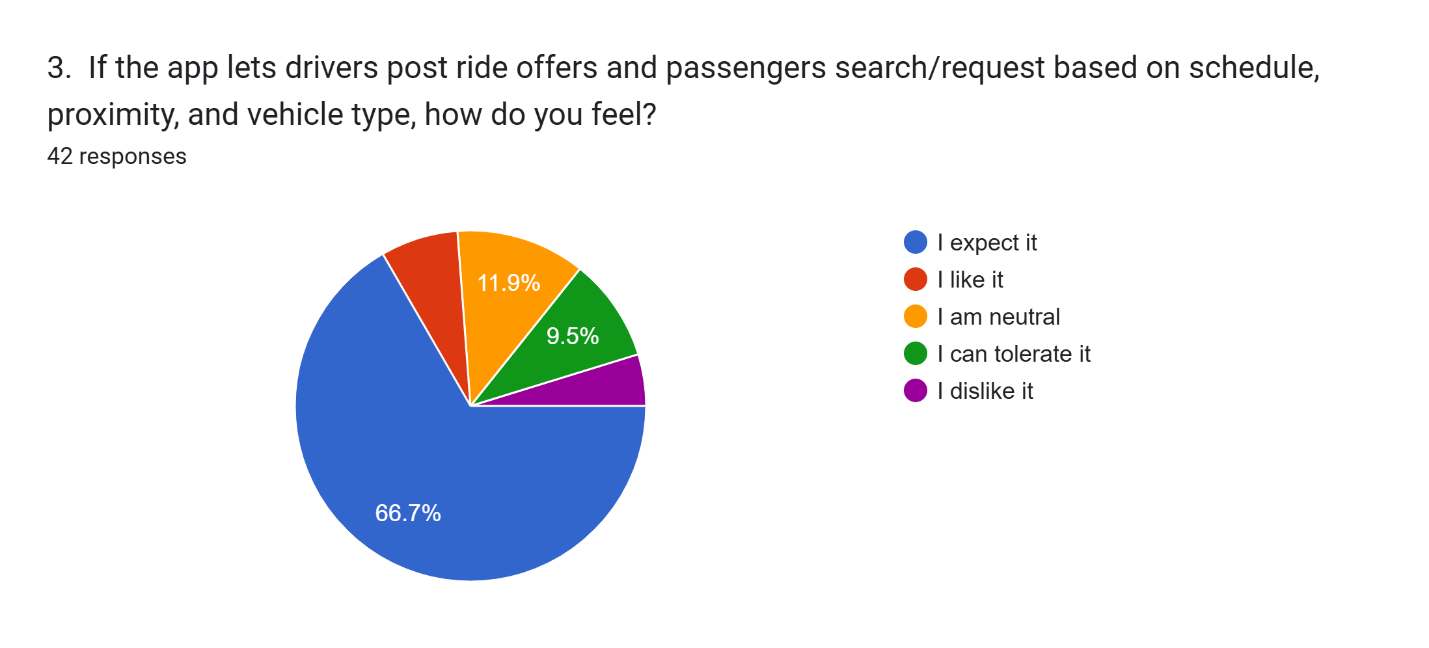


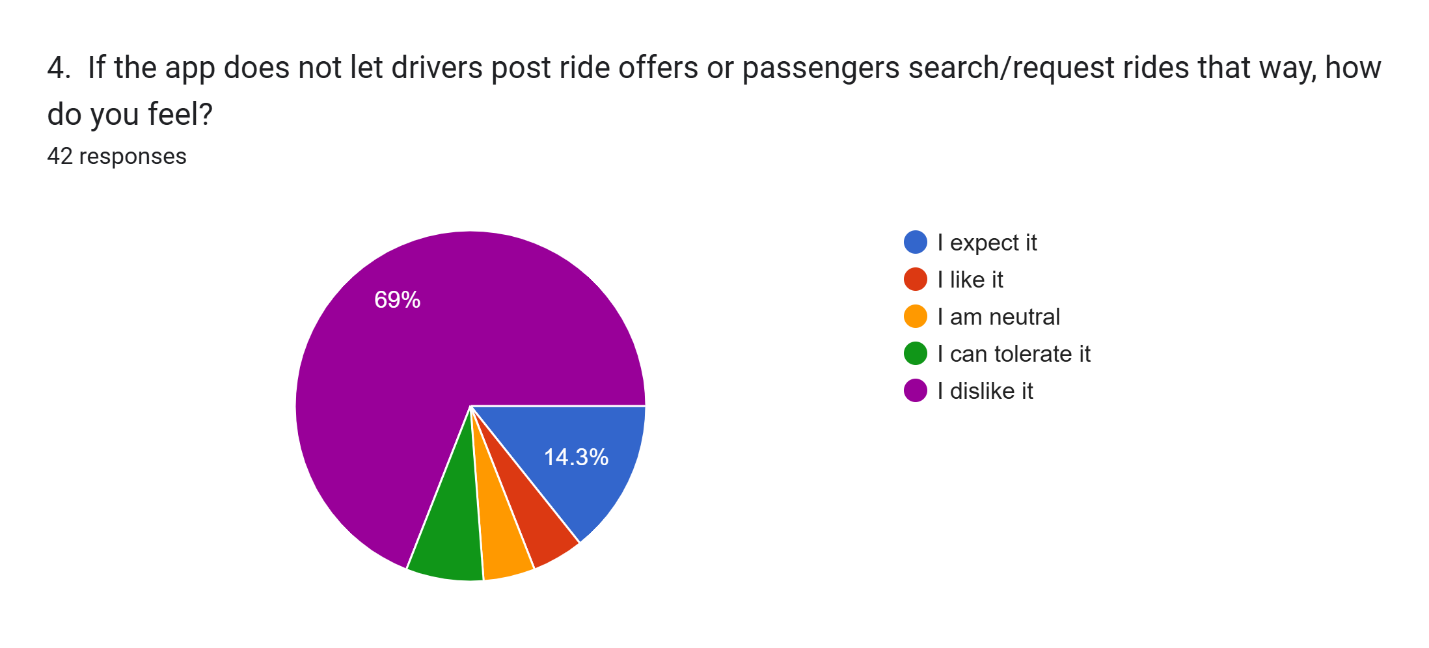


|  |  |  |
| --- | --- | --- |
| Question Type | Majority Response | Insight |
| Functional | I like it | Users highly value and expect the app to allow registration and login using MMU credentials and digital ID. |
| Dysfunctional | I dislike it | The absence of the ability to register and log in using MMU credentials and digital ID is strongly disliked by users. |

Kano Result: **Satisfiers**

## **2.2.2 Ride Offer & Request**

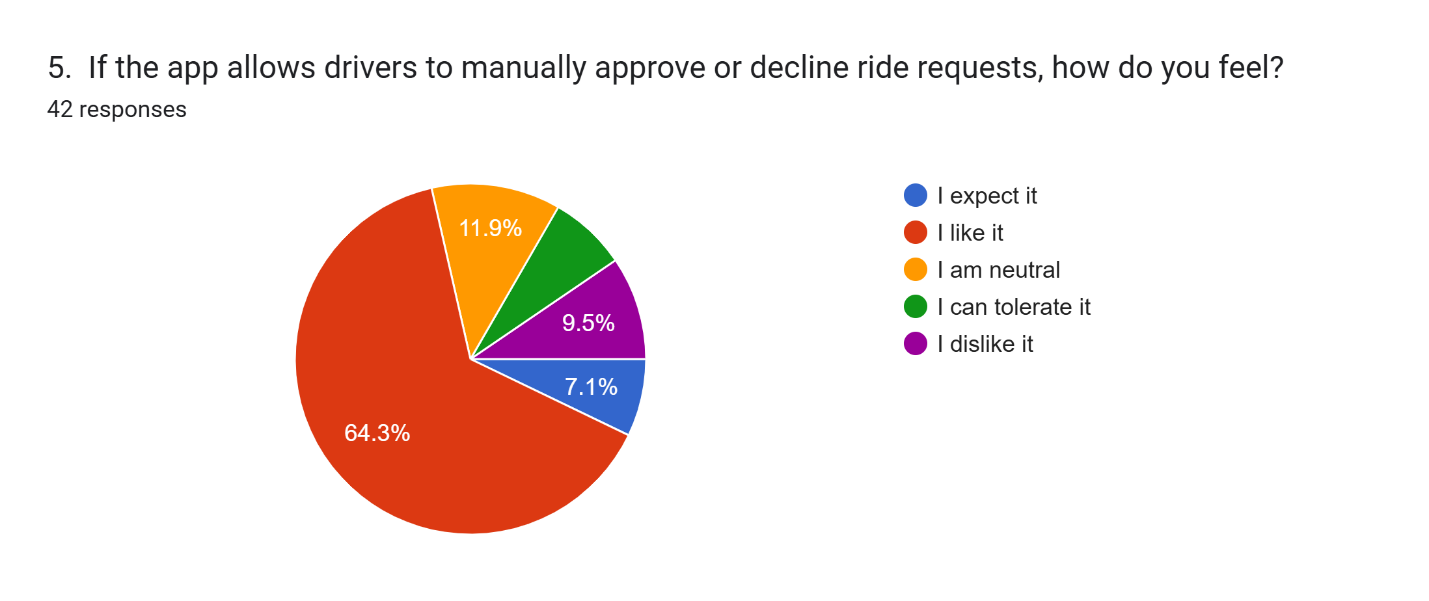


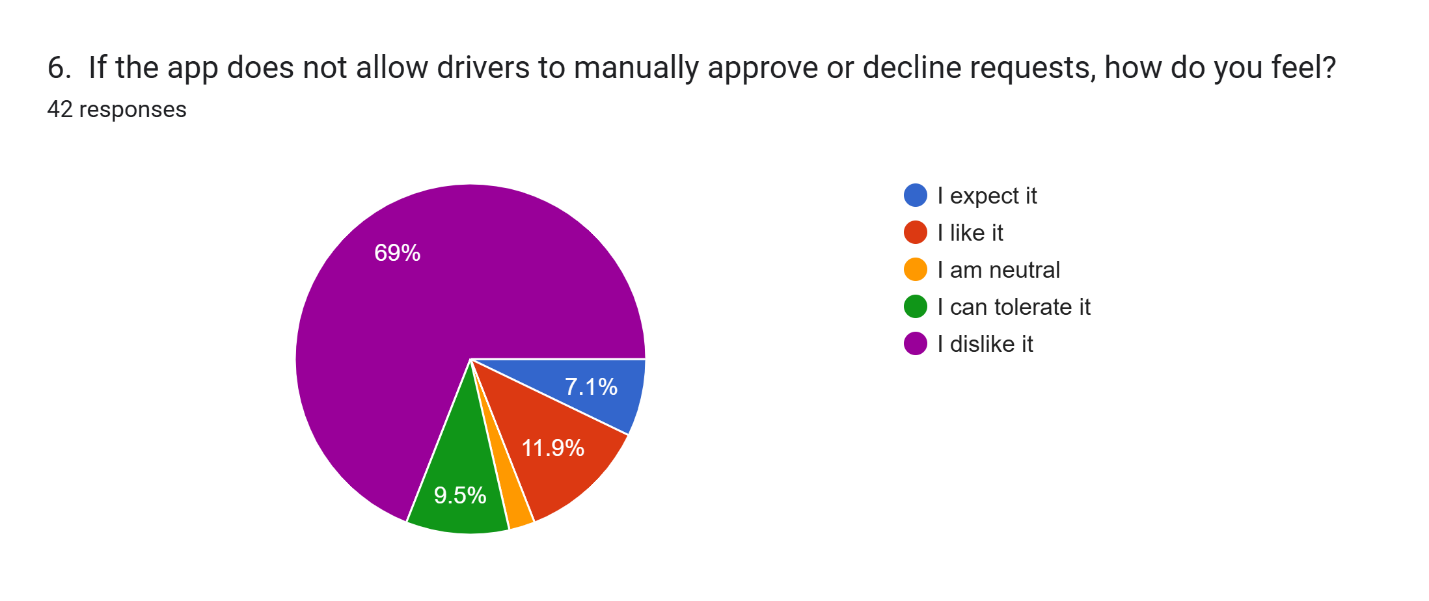


|  |  |  |
| --- | --- | --- |
| Question Type | Majority Response | Insight |
| Functional | I expect it | Users highly expect and value the app to allow drivers to post ride offers and passengers to search/request based on schedule, proximity, and vehicle type. |
| Dysfunctional | I dislike it | The absence of features allowing drivers to post ride offers and passengers to search/request rides based on schedule, proximity, and vehicle type is strongly disliked by users. |

Kano Result: **Dissatisfiers**

## **2.2.3 Manual Ride Matching**

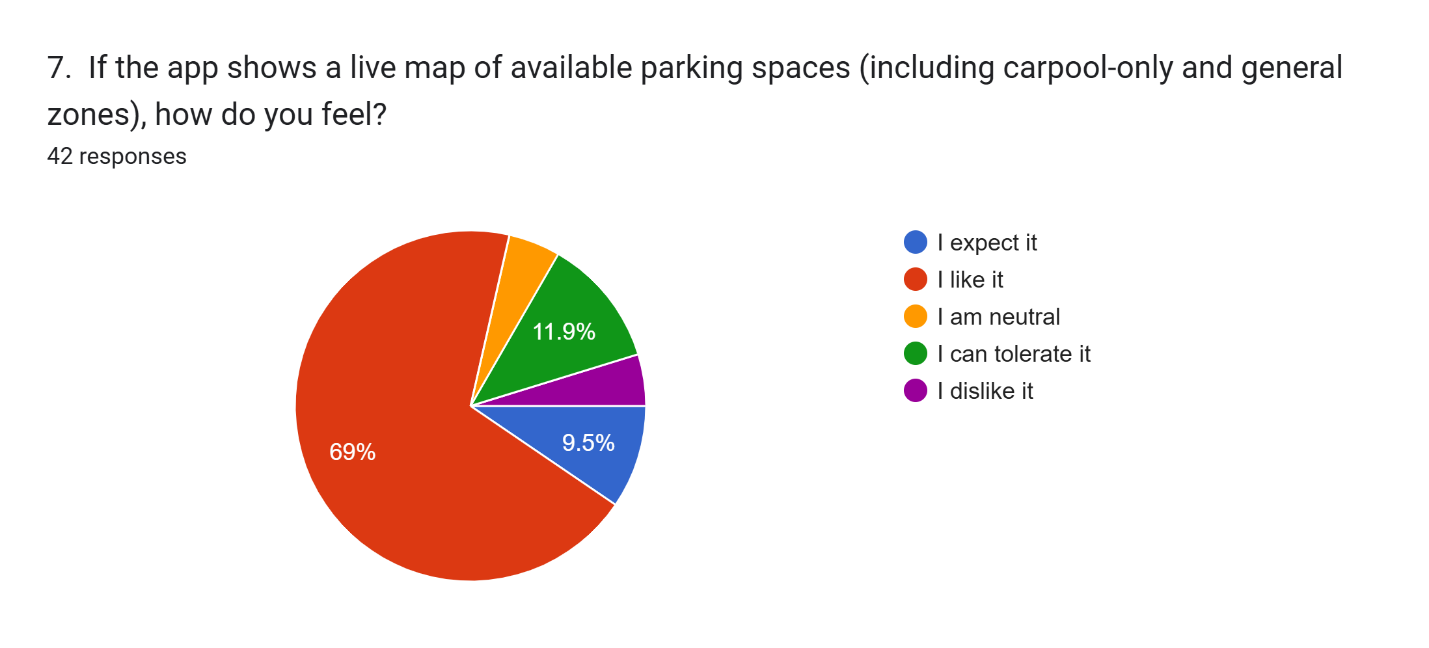


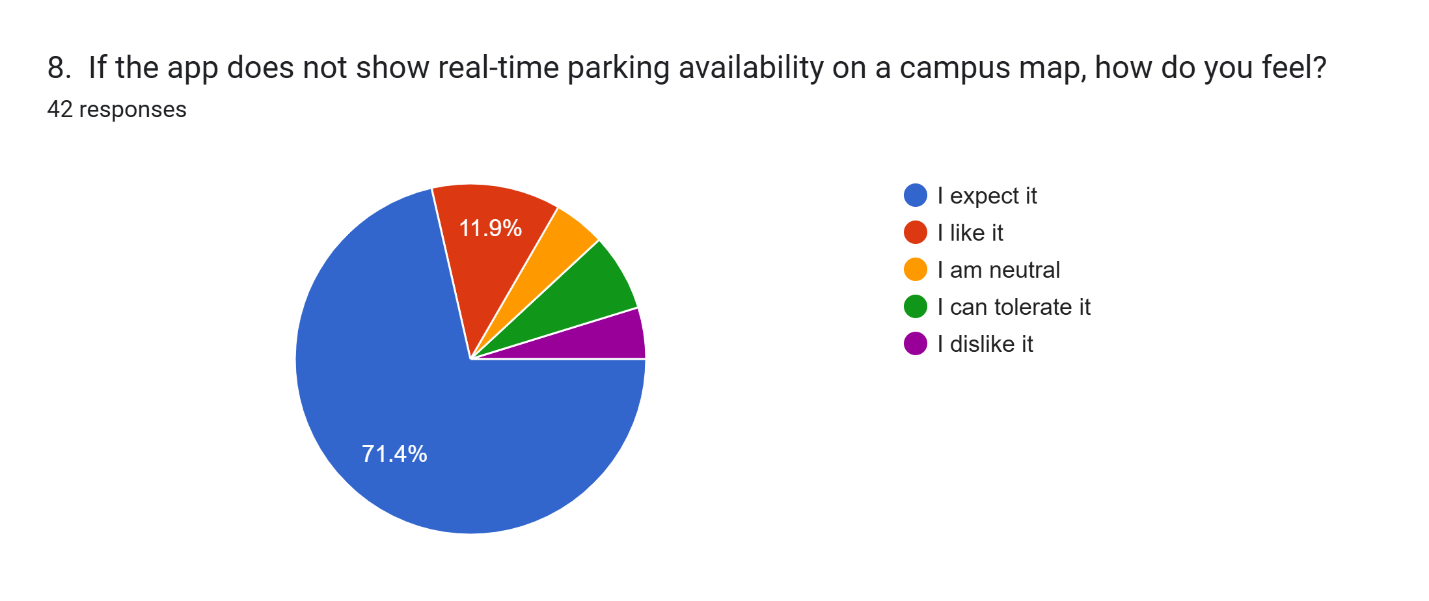


|  |  |  |
| --- | --- | --- |
| Question Type | Majority Response | Insight |
| Functional | I like it | Users strongly prefer and value the ability for drivers to manually approve or decline ride requests. |
| Dysfunctional | I dislike it | The absence of drivers being able to manually approve or decline ride requests is strongly disliked by users. |

Kano Result: **Satisfiers**

## **2.2.4 Real-Time Campus Parking Availability**

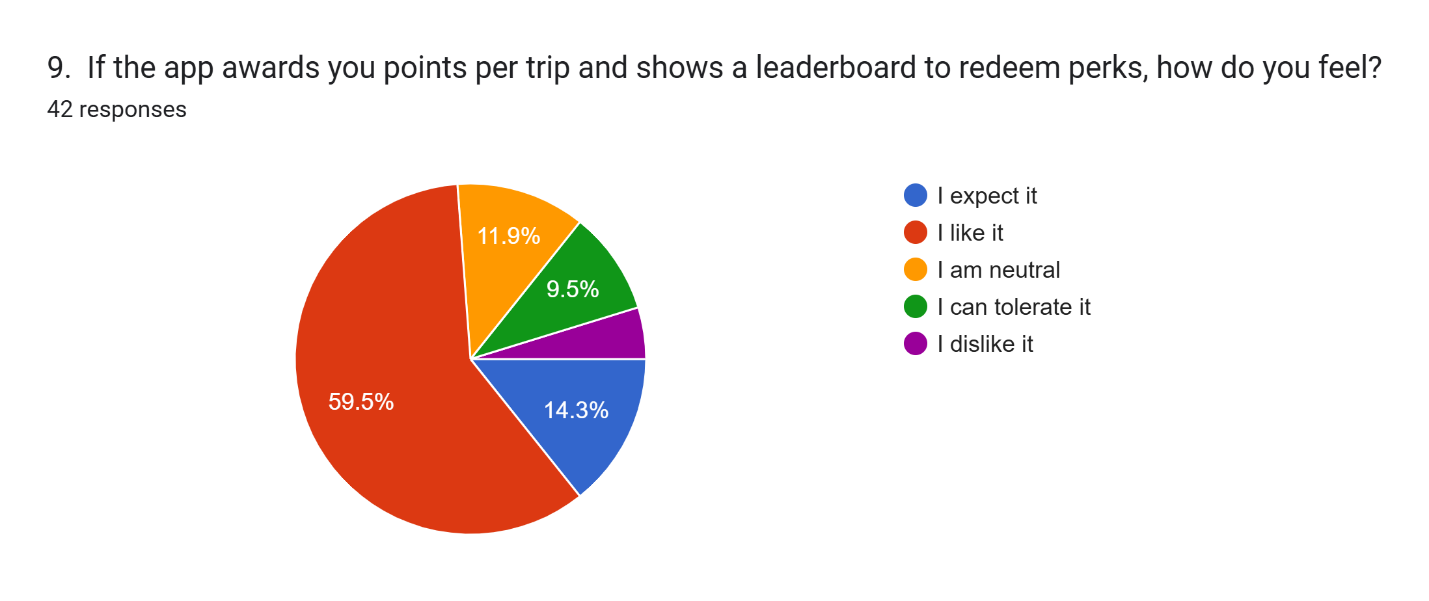


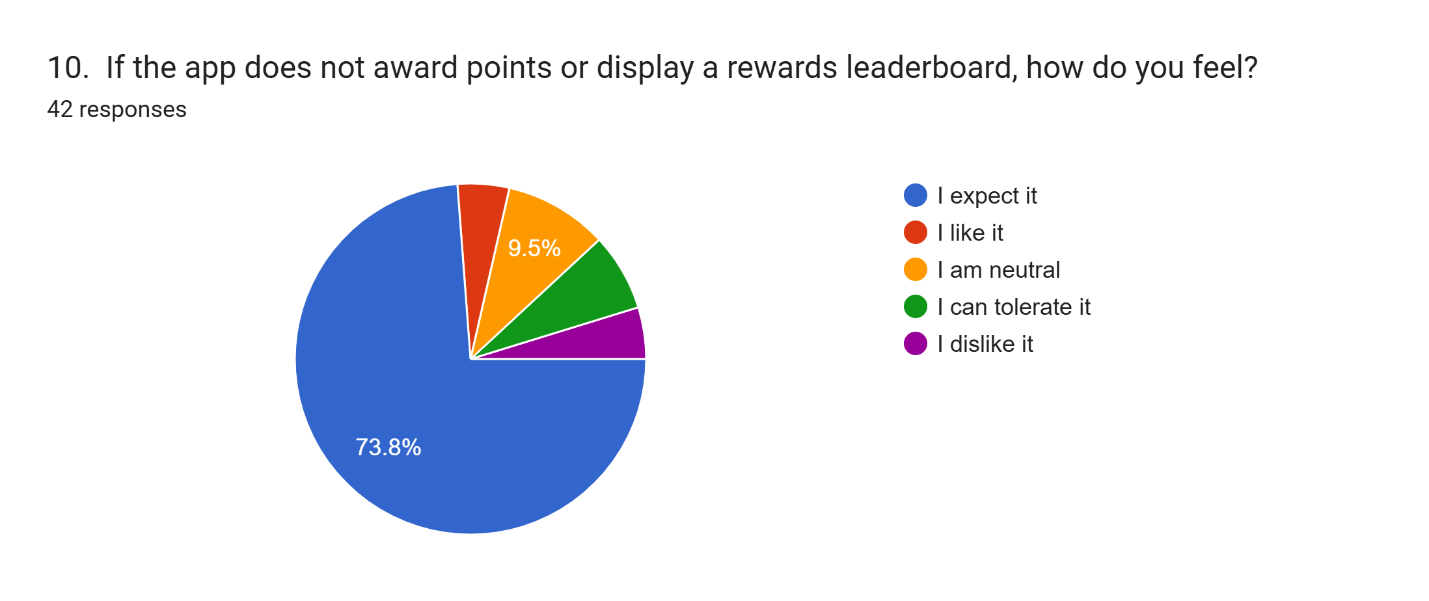


|  |  |  |
| --- | --- | --- |
| Question Type | Majority Response | Insight |
| Functional | I like it | Users highly prefer and value the app showing a live map of available parking spaces, including carpool-only and general zones. |
| Dysfunctional | I expect it | Users strongly expect the app to show real-time parking availability on a campus map. The absence of this feature would be a significant drawback. |

Kano Result: **Delighters**

## **2.2.5 Reward Points & Leaderboard System**

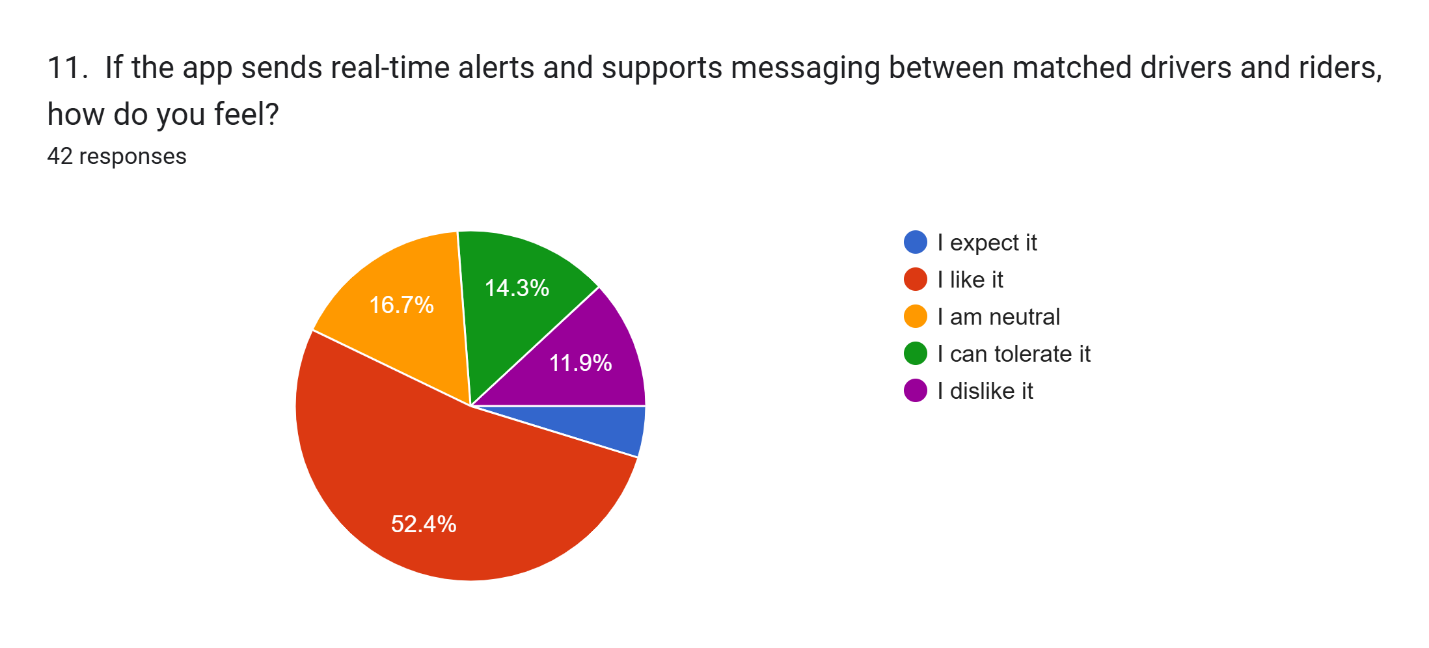


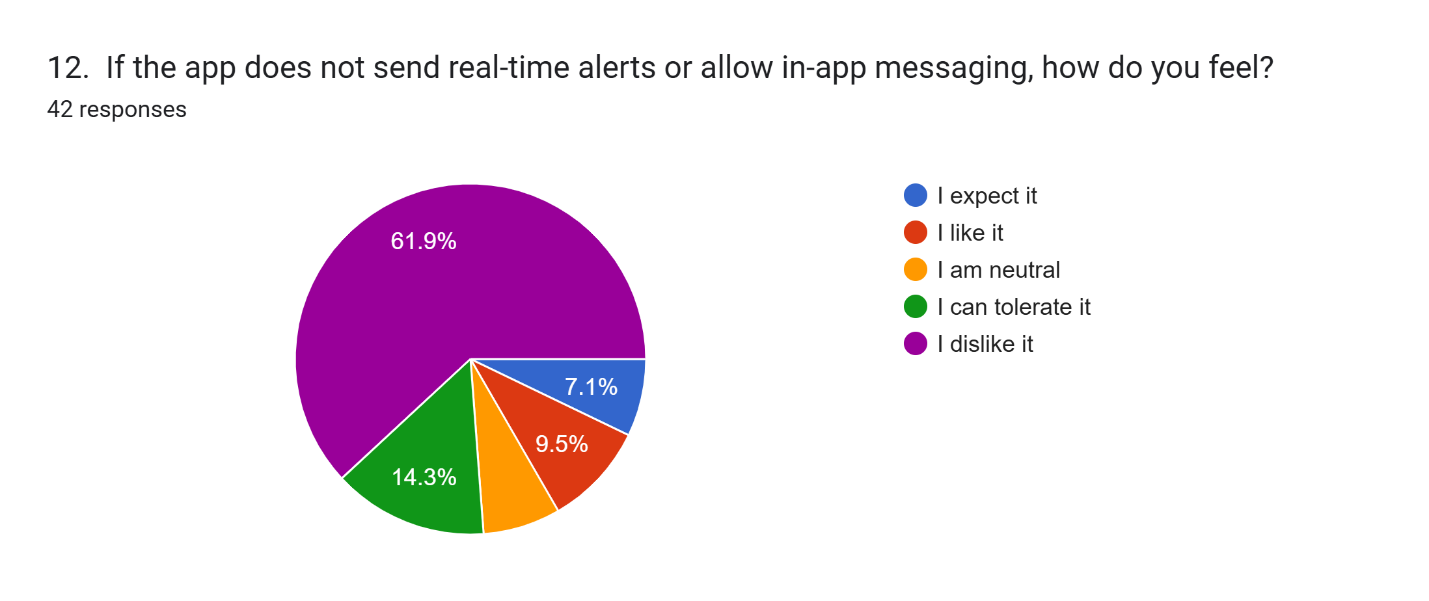


|  |  |  |
| --- | --- | --- |
| Question Type | Majority Response | Insight |
| Functional | I like it | Users highly appreciate and value the app awarding points per trip and displaying a leaderboard for redeeming perks. |
| Dysfunctional | I expect it | Users strongly expect the app to *not* award points or display a rewards leaderboard, suggesting a preference for simplicity or a focus on core functionality over gamification. |

Kano Result: **Delighters**

## **2.2.6 Push Notifications & In-App Messaging**

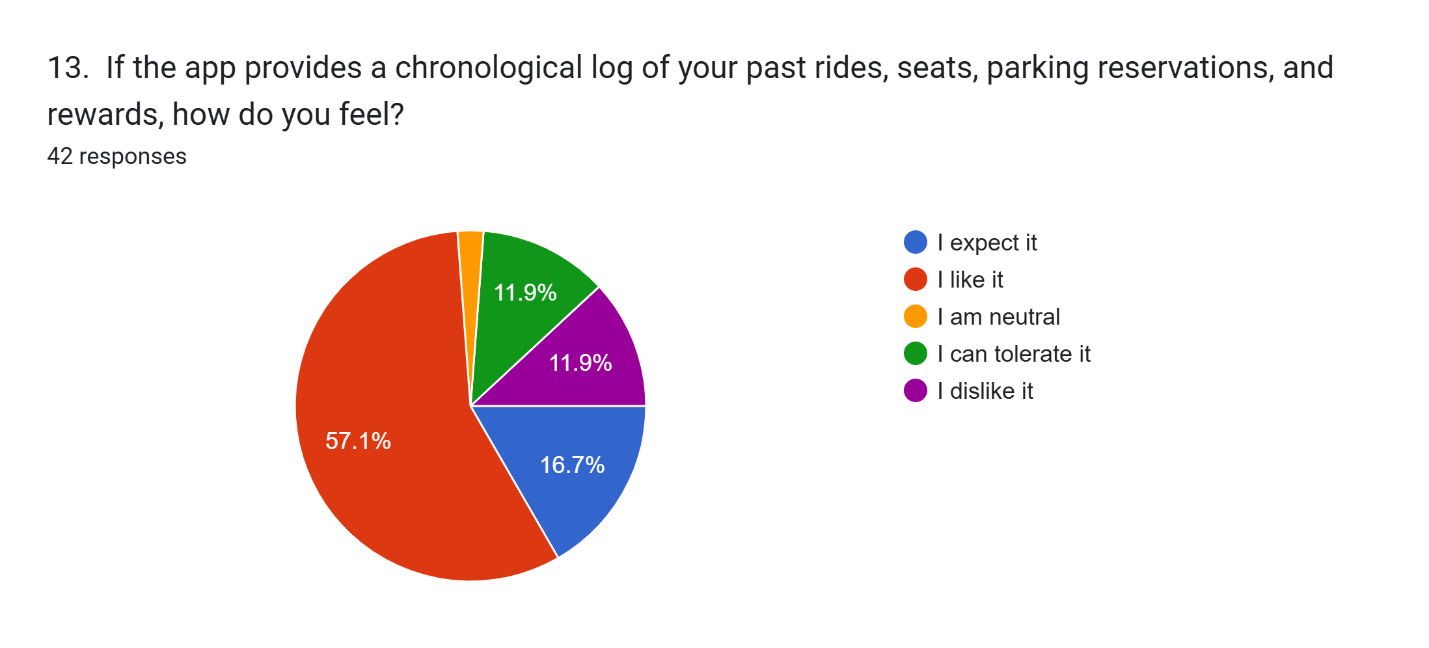


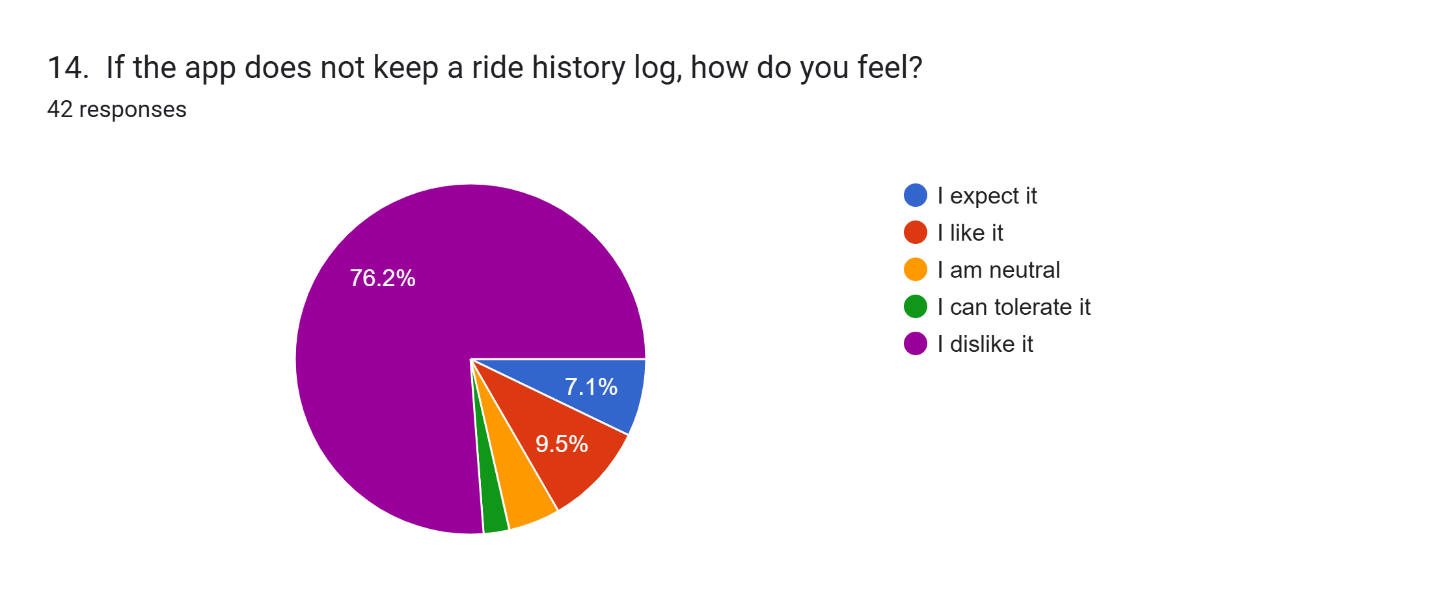


|  |  |  |
| --- | --- | --- |
| Question Type | Majority Response | Insight |
| Functional | I like it | Users highly appreciate and value the app sending real-time alerts and supporting messaging between matched drivers and riders. |
| Dysfunctional | I dislike it | The absence of real-time alerts and in-app messaging is strongly disliked by users. |

Kano Result: **Satisfiers**

## **2.2.7 Ride History Tracking**

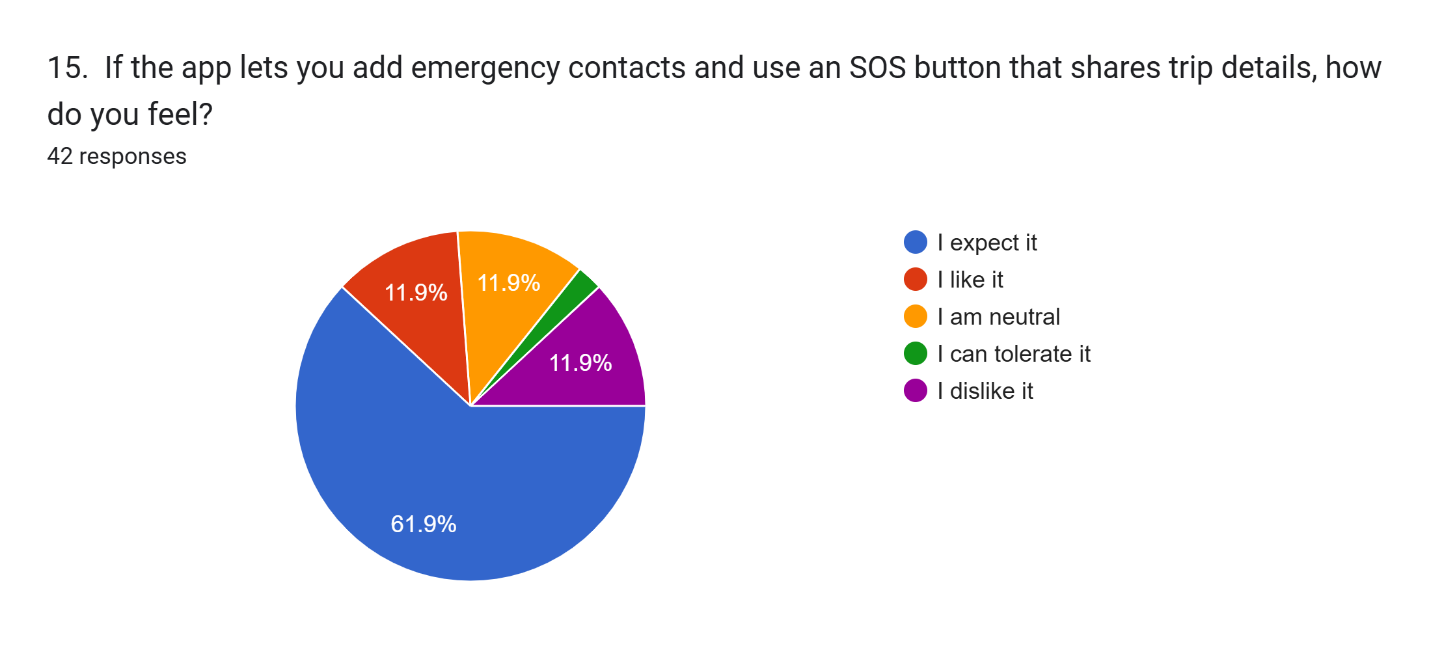


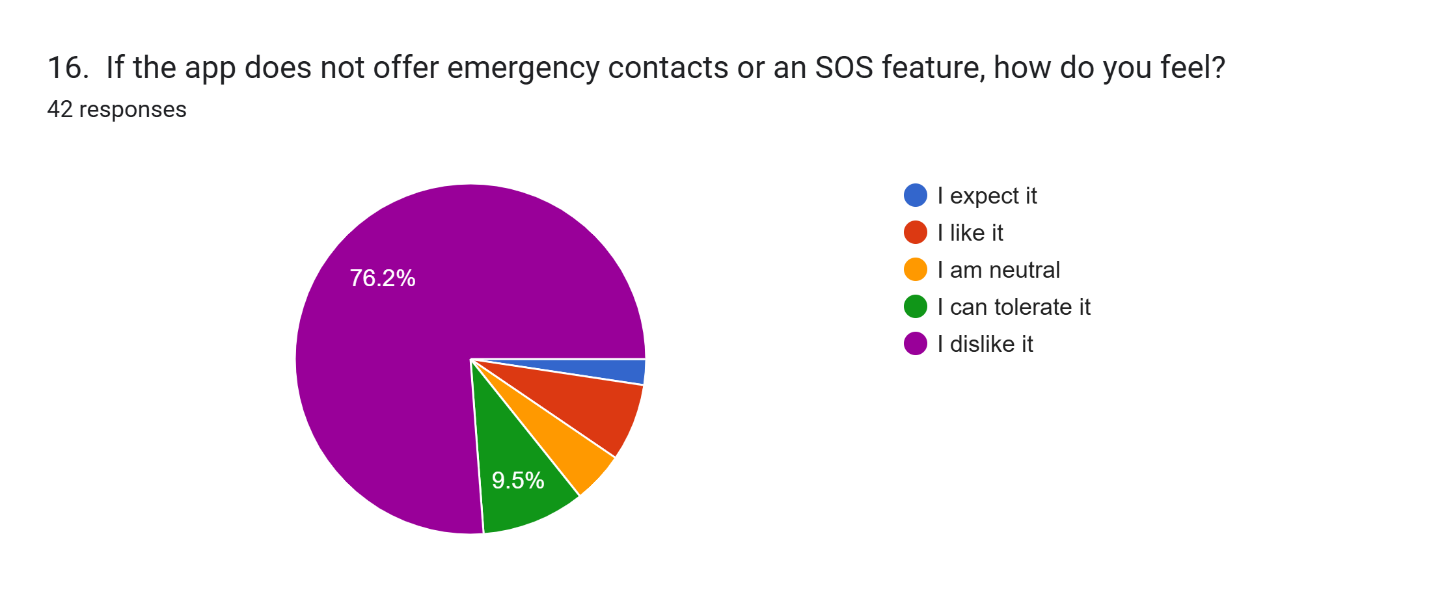


|  |  |  |
| --- | --- | --- |
| Question Type | Majority Response | Insight |
| Functional | I like it | Users highly appreciate and value the app providing a chronological log of past rides, seats, parking reservations, and rewards. |
| Dysfunctional | I dislike it | Absence of a detailed The absence of the app keeping a ride history log is strongly disliked by users. |

Kano Result: **Satisfiers**

## **2.2.8 Safety & Emergency Features**

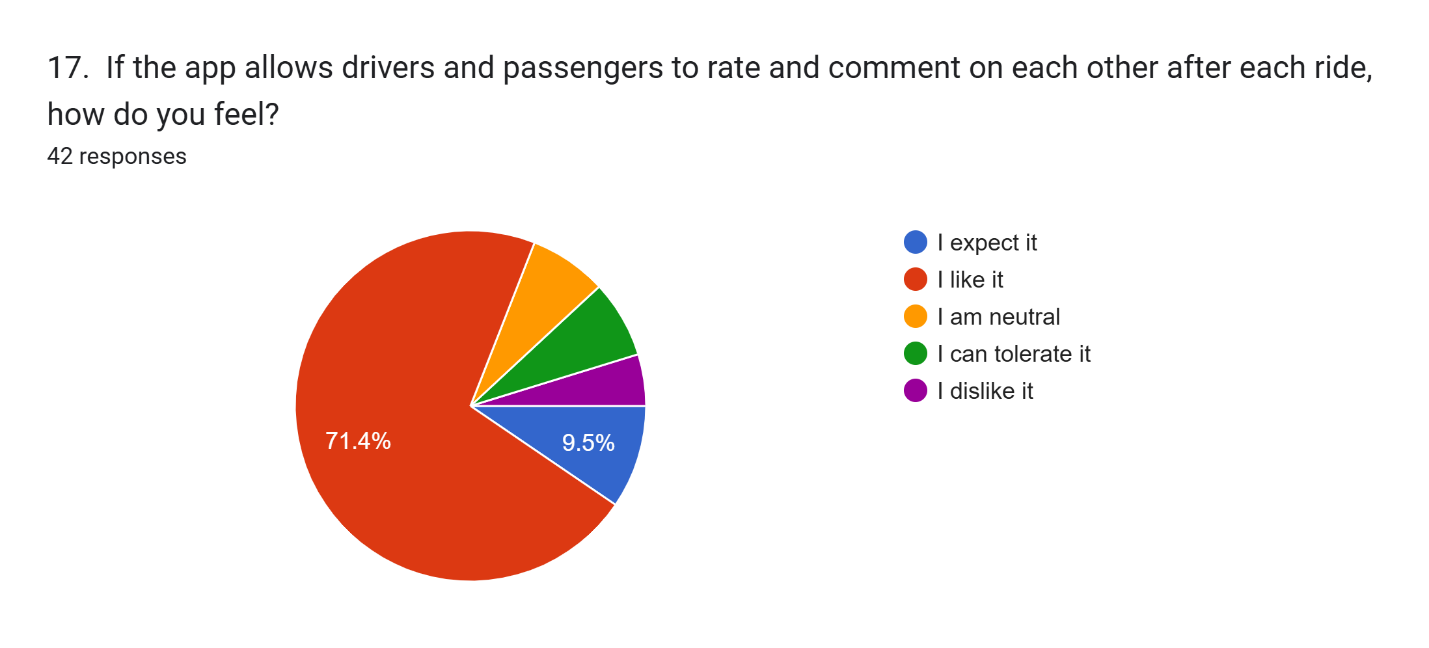


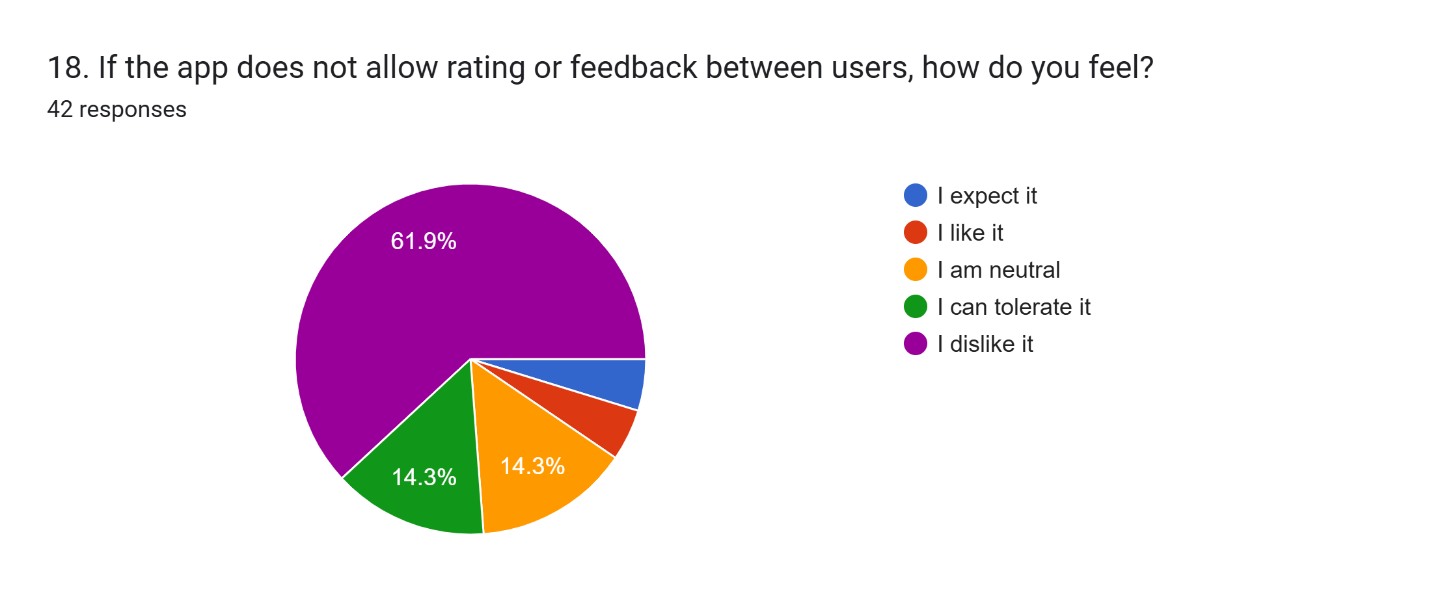


|  |  |  |
| --- | --- | --- |
| Question Type | Majority Response | Insight |
| Functional | I expect it | Users highly expect and value the app to allow adding emergency contacts and using an SOS button that shares trip details. |
| Dysfunctional | I dislike it | The absence of emergency contacts or an SOS feature is strongly disliked by users. |

Kano Result: **Dissatisfiers**

## **2.2.9 User Ratings & Feedback System**

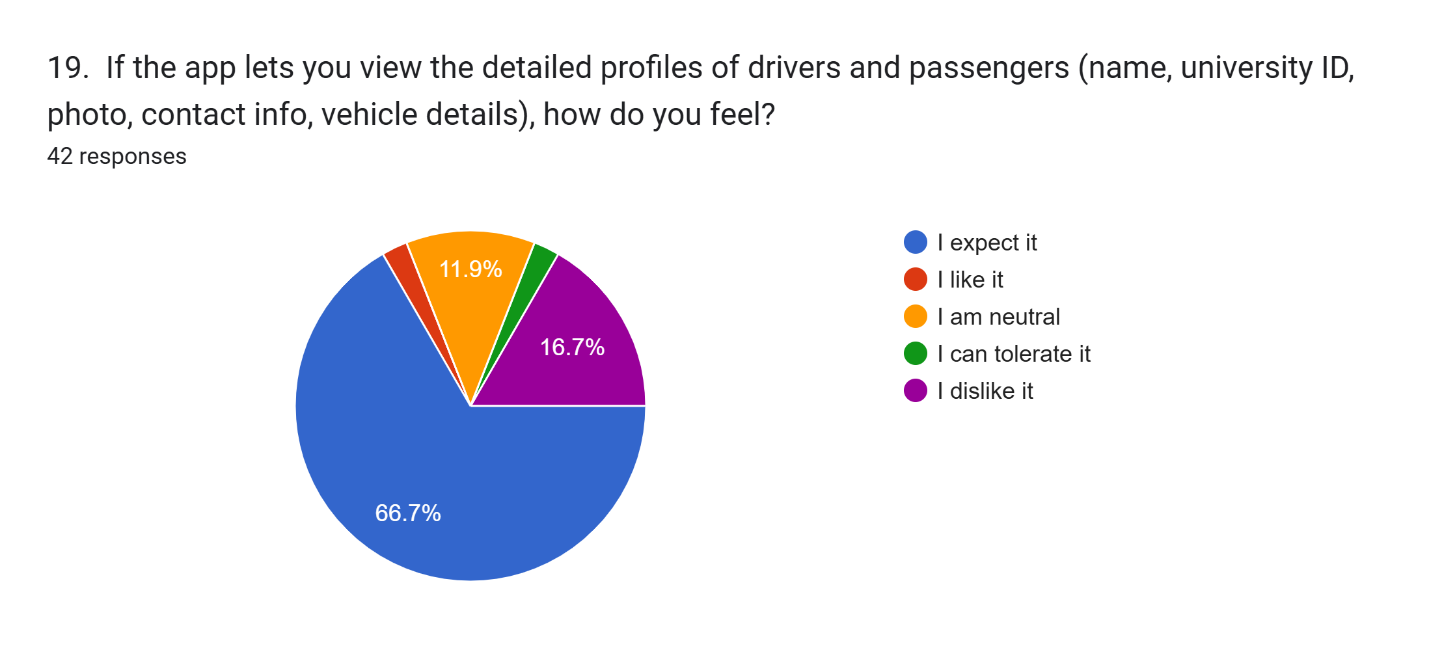


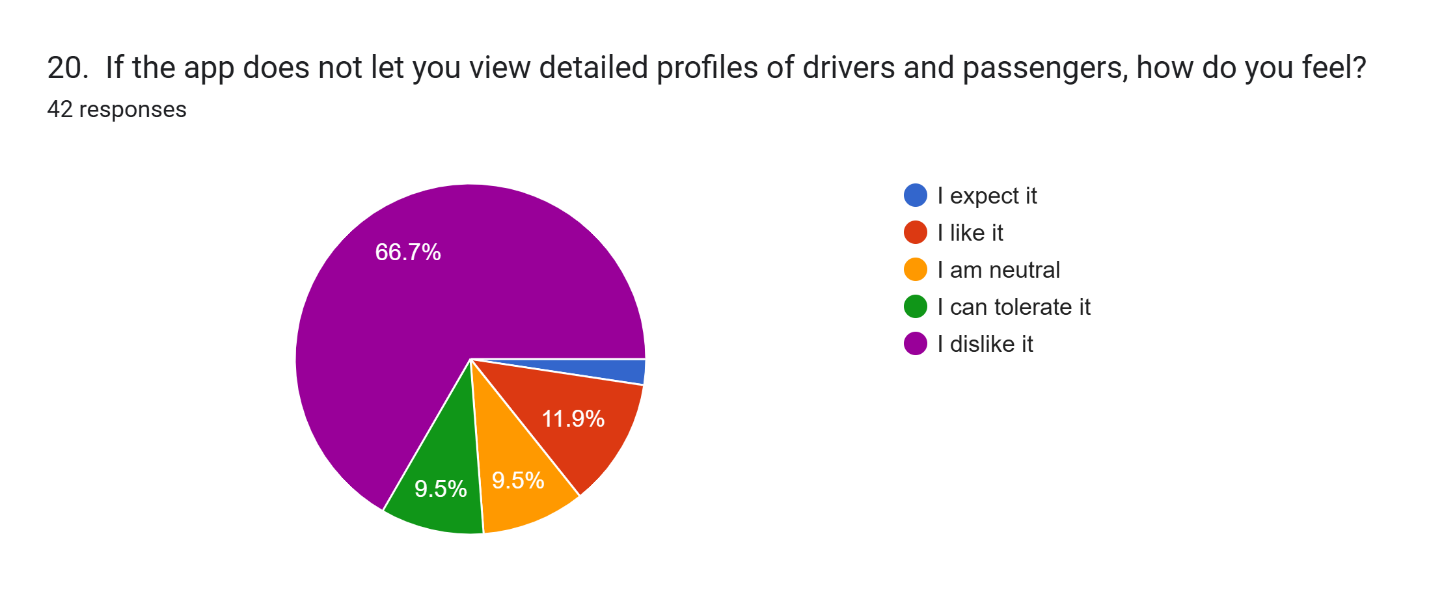


|  |  |  |
| --- | --- | --- |
| Question Type | Majority Response | Insight |
| Functional | I like it | Users highly value and appreciate the app allowing drivers and passengers to rate and comment on each other after each ride. |
| Dysfunctional | I dislike it | The absence of rating or feedback between users is strongly disliked by users. |

Kano Result: **Satisfiers**

## **2.2.10 Transparent User Profiles**





|  |  |  |
| --- | --- | --- |
| Question Type | Majority Response | Insight |
| Functional | I expect it | Users highly expect and value the app to allow viewing detailed profiles of drivers and passengers (name, university ID, photo, contact info, vehicle details). |
| Dysfunctional | I dislike it | The absence of the ability to view detailed profiles of drivers and passengers is strongly disliked by users. |

Kano Result: **Dissatisfiers**

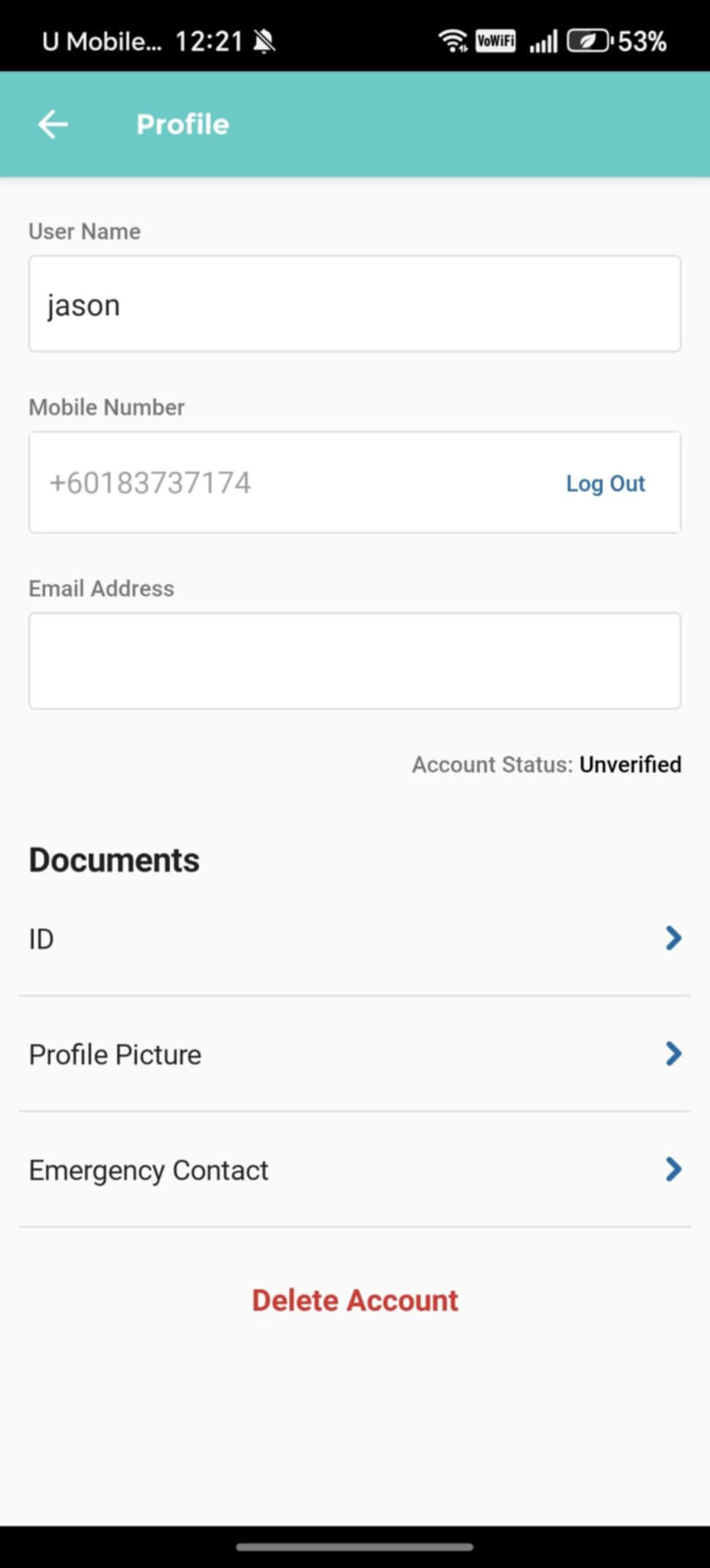
## **2.3 Source of Questionnaire**

<https://docs.google.com/forms/d/e/1FAIpQLScaYkZNJcKqdPVg1tEbTLIpDdZL9fAFlJL3WMWo-plqJN5LLA/viewform>

## **3.0 Observation**

## **3.1 Kummute - Kumpool**

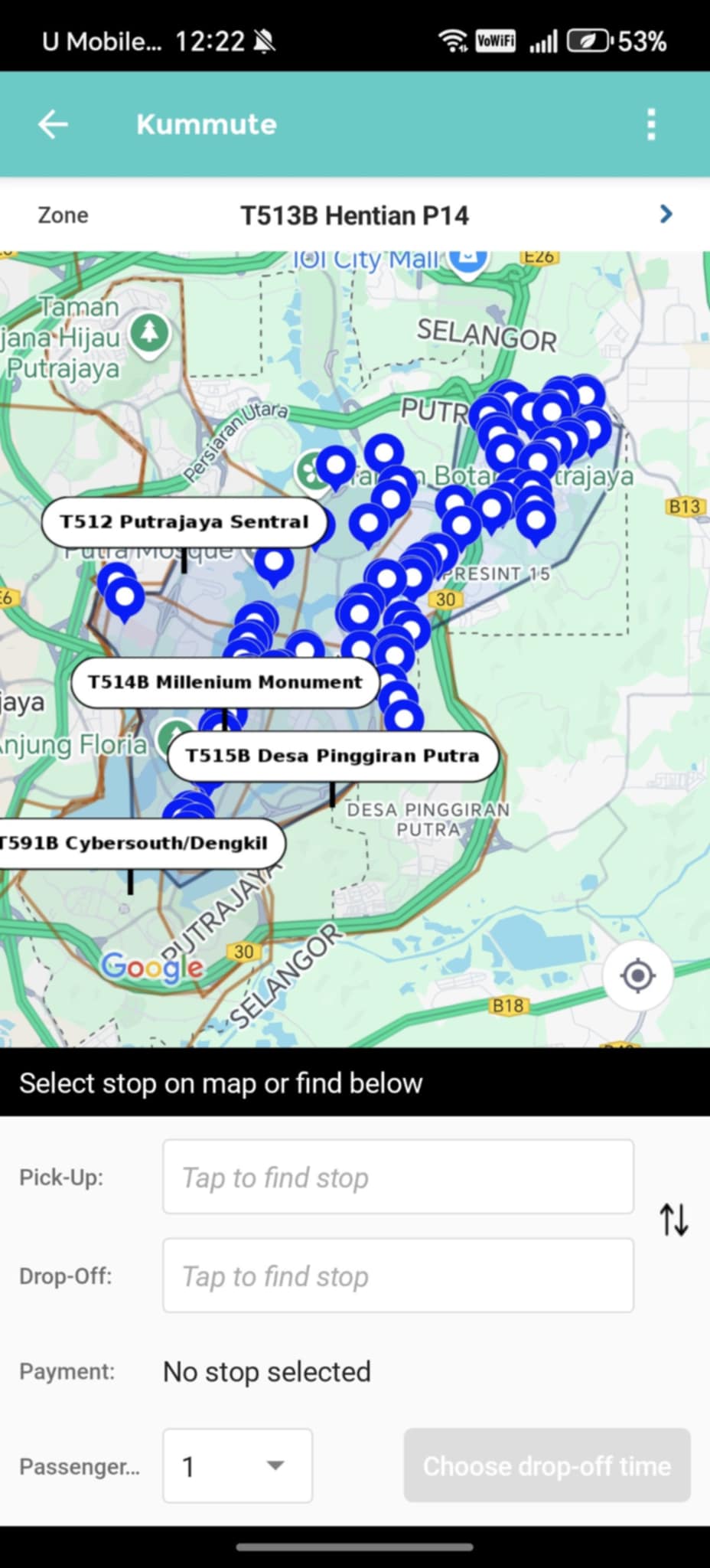
## **3.1.1 User Profile & Digital ID Verification**



*Figure 3.1.1 Kumpool User Profile Settings*

The observed Kumpool platform includes key profile features such as user information, emergency contact, and document uploads. This directly influenced our system’s design, where we implemented MMU Single Sign-On (SSO) for user verification and included similar profile elements like name, university ID, contact details, and vehicle information. These similarities show how the reference guided us in structuring secure and complete user onboarding.

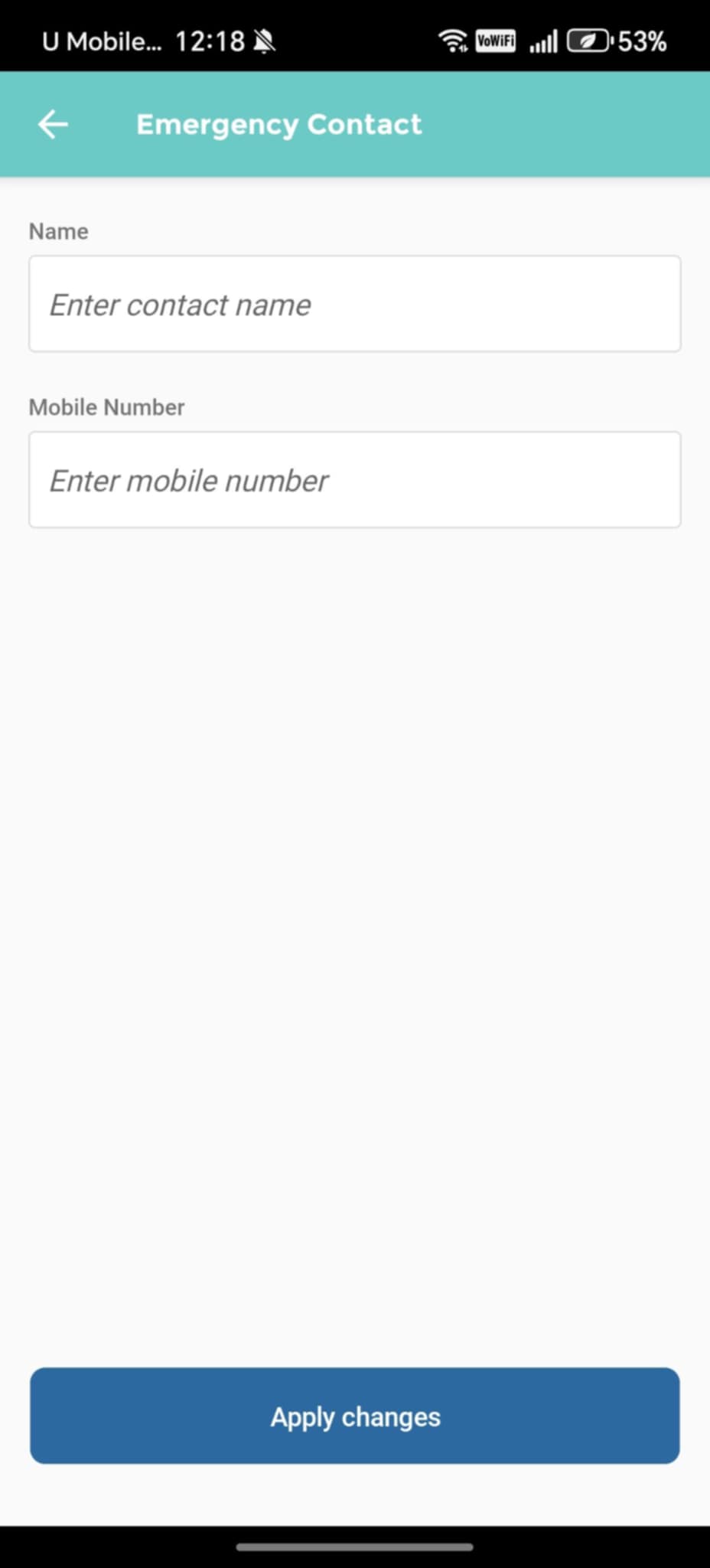
## **3.1.2 Ride Request & Searching**



*Figure 3.1.2 Kumpool Search Rides*

The system enables passengers to search for rides by filtering available options according to schedule, proximity, and preferred vehicle type, allowing them to quickly find the most suitable ride. Similar to Kumpool, riders can select their pickup and drop-off locations, choose a payment method, and specify the number of passengers.

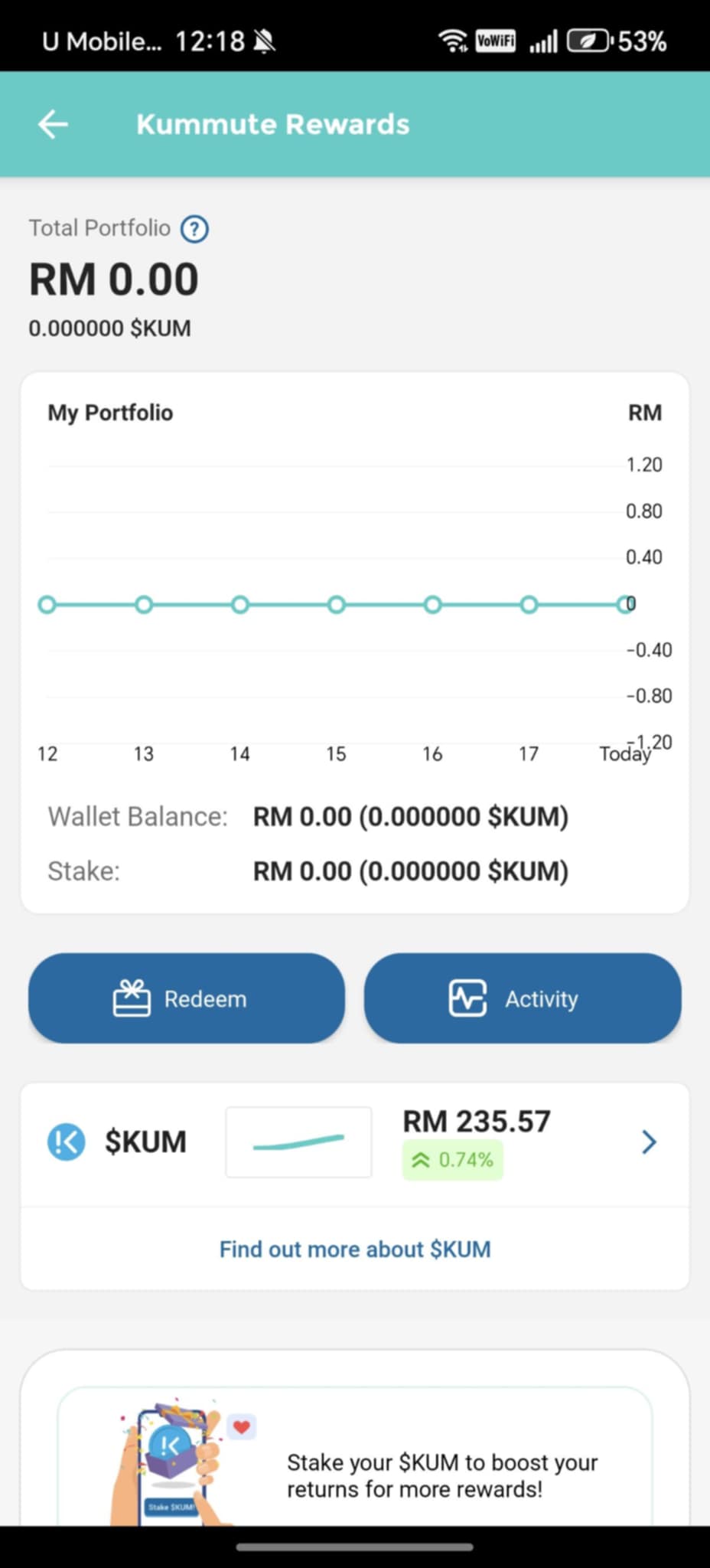
## **3.1.3 Safety & Emergency Features**



*Figure 3.1.3 Kumpool Emergency Contact*

The observed platform includes essential safety features such as emergency contact functionality, allowing users to input their contact details. This influenced our system design by highlighting the importance of having emergency contacts readily available. This observation led us to specify additional requirements for an alert system that automatically shares real-time ride information with designated contacts, ensuring enhanced user safety in critical situations.

## **3.1.4 Incentives & Reward System**

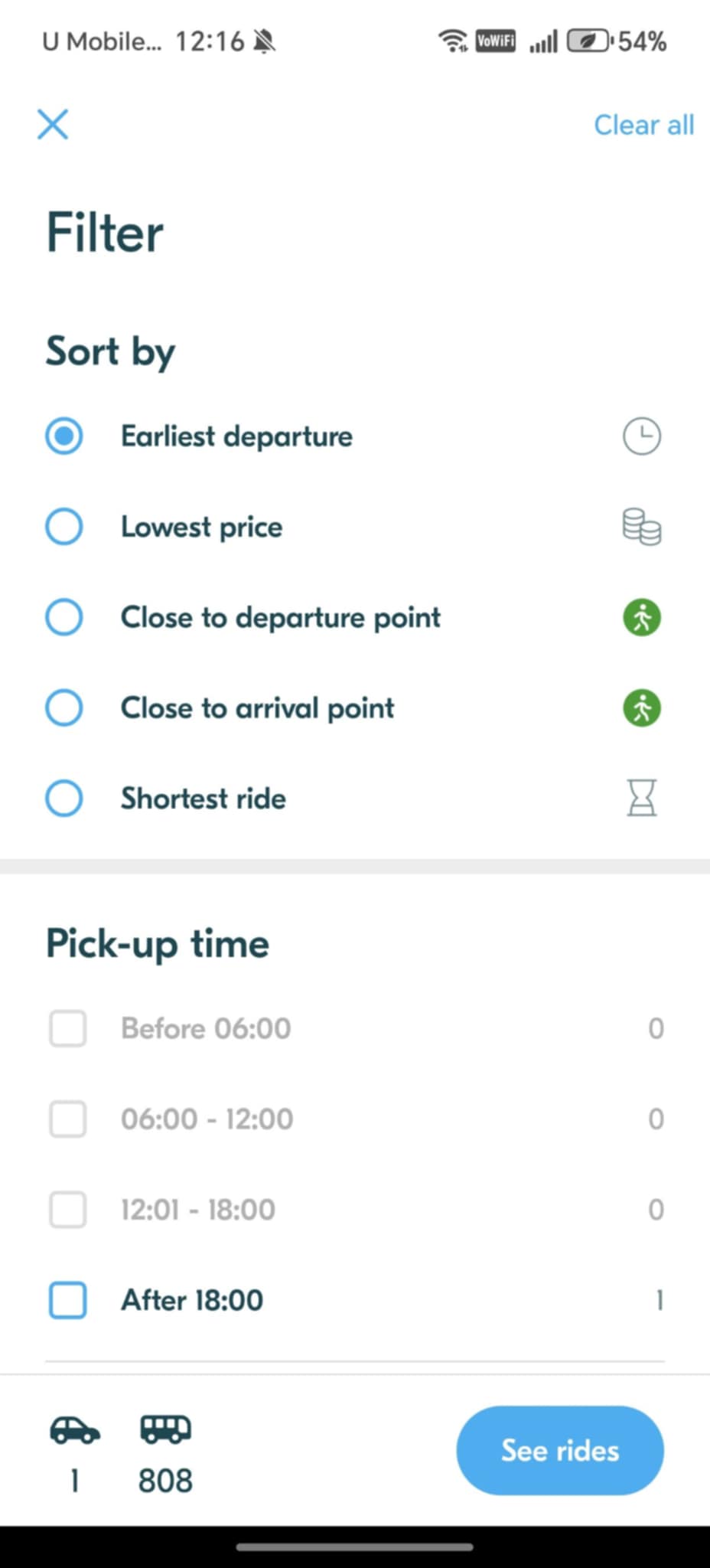


*Figure 3.1.4 Kumpool Reward System*

The observed Kummute Rewards interface showcases an effective incentive system that influenced our platform’s reward design. Inspired by this, we implemented a point-based structure where drivers and riders earn rewards, with university-specific redemption options. Visual elements like earnings tracking, leaderboards, and badges promote engagement and encourage ongoing participation in the carpooling program.

## **3.2 BlaBlaCar**

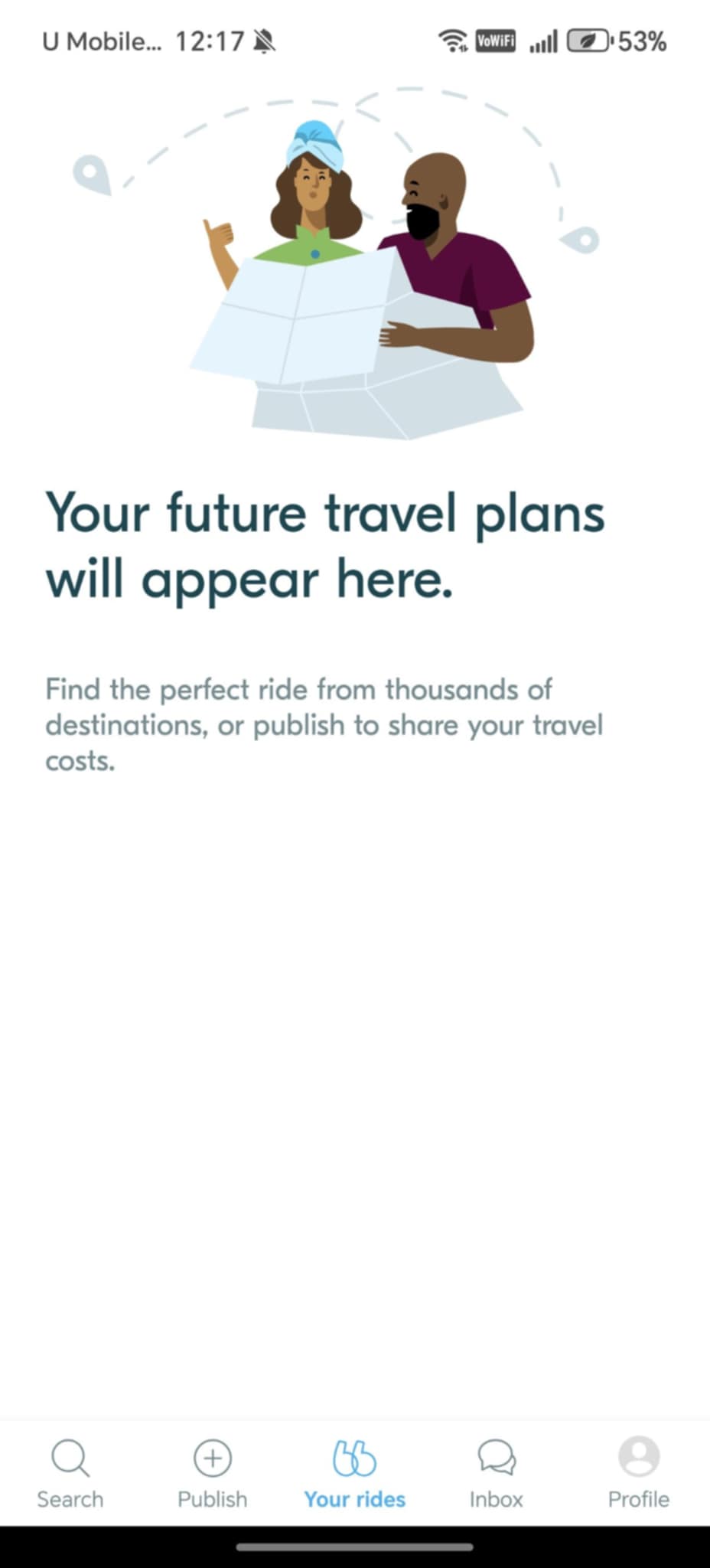
## **3.2.1 Rides Filter**



*Figure 3.2.1 BlablaCar Search Filter*

The observed BlaBlaCar filtering interface shows an effective ride search system that influenced our platform’s design. By studying its filtering options, such as time, proximity, and ride details. We developed similar features that allow users to find suitable rides quickly. Our system also gives drivers control over passenger approval, ensuring balanced and efficient ride-matching experience.

## **3.2.2 View Ride History**



*Figure 3.2.2 BlablaCar Ride History*

The observed BlaBlaCar travel plans screen influenced our implementation of the ride history feature. Inspired by its design, our system includes a clear chronological log where users can track past rides, seat usage, parking reservations, and rewards. This helps users monitor their activity while keeping the interface familiar and easy to navigate.

## **3.2.3 Chat History**



*Figure 3.2.3 BlablaCar Inbox Interface*

The observed BlaBlaCar inbox interface highlights a useful communication feature that shaped our system's design. Inspired by this, we implemented in-app messaging for drivers and riders to coordinate rides. This ensures smooth, real-time communication tailored for university carpooling.

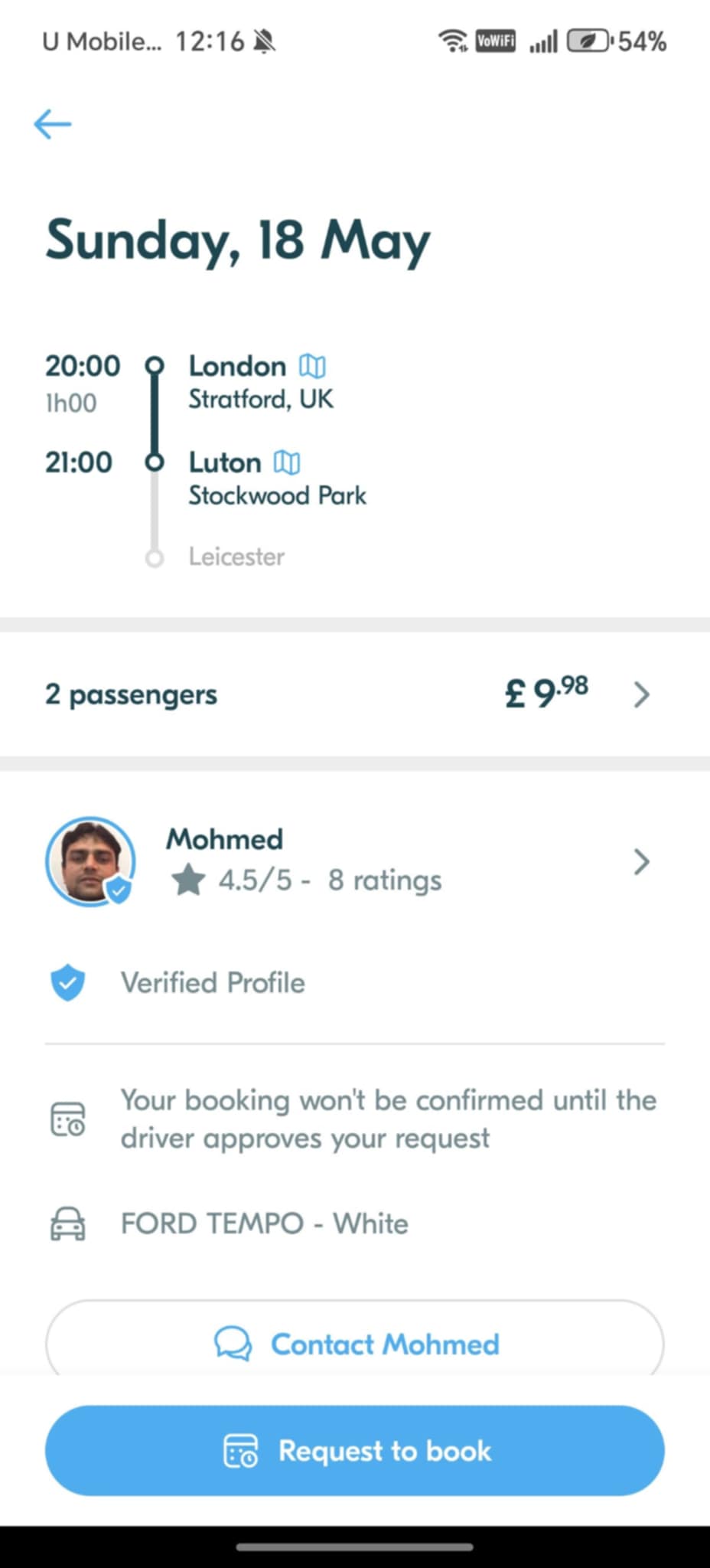
## **3.2.4 Rating & Feedback System**



*Figure 3.2.4 BlablaCar Rating & Feedback System*

The observed BlaBlaCar platform includes a mutual rating and feedback system that promotes trust and accountability. Our system adopts this approach, allowing drivers and passengers to rate and comment on each other after each ride. This transparent process builds user reputation, encourages positive behavior, and helps users make informed decisions, enhancing overall confidence in the platform.

## **3.2.5 Manual Matching System**



*Figure 3.2.5 BlablaCar Ride Matching Interface*

The observed BlaBlaCar platform uses a manual matching process, allowing drivers to review passenger profiles and approve or decline requests, enhancing safety and control. Our system adopts a similar approach, enabling drivers to make informed decisions based on passenger details, ensuring a secure and transparent ride-matching experience.

## **4.0 Prototyping**

## **4.1 User Onboarding & Digital ID Verification**

A screenshot of a sign in

AI-generated content may be incorrect.

The "MMU Rides" application enforces a mandatory **single sign-on (SSO)** system using MMU accounts for user access, eliminating the need for separate sign-up. This strictly limits usage to current MMU students, staff, and faculty, ensuring digital ID verification for all users.

## **4.2 Ride Offer & Request**

## **4.2.1 Ride Filter**

A screenshot of a phone

AI-generated content may be incorrect. A screenshot of a phone

AI-generated content may be incorrect.

The "MMU Rides" application features a **ride search and filter system** allowing passengers to find rides based on **schedule (date, time)**, **proximity (distance)**, and **preferred vehicle type (Sedan, SUV, Van, Hybrid, EV)**. Passengers can also specify the **number of seats needed** before requesting a ride.

## **4.2.2 Send Ride Request**

A screenshot of a phone

AI-generated content may be incorrect. A screenshot of a phone

AI-generated content may be incorrect.

The "MMU Rides" app allows passengers to **view ride details**, including driver information, origin, destination, schedule, vehicle, and available seats. Passengers can then **send ride requests** which are confirmed with a "Request Sent" message, pending driver approval.

## **4.2.3 Create Ride Offer**

A screenshot of a phone

AI-generated content may be incorrect.

Drivers can **create and publish ride offers** within the "MMU Rides" application. This involves specifying key details such as the **exact origin and destination points** of the ride, the **intended date and time** of departure, and the **number of available seats** they are offering for passengers.

## **4.3 Manual Matching System**

A screenshot of a phone

AI-generated content may be incorrect.

"MMU Rides" features a **manual matching system** where **drivers directly approve or decline incoming ride requests** through the application, giving them full control over who joins their carpool.

## **4.4 Real-Time Campus Parking Availability**

A screenshot of a map

AI-generated content may be incorrect.

"MMU Rides" provides **real-time campus parking availability** through an **occupancy map**, distinguishing between **"carpool-only" zones and general parking lots**. Users can view the number of available spaces in each designated area.

## **4.5 Push Notifications & In-App Messaging**

A screenshot of a chat

AI-generated content may be incorrect. Screens screenshot of a phone

AI-generated content may be incorrect.

"MMU Rides" facilitates communication through **push notifications** for ride confirmations, cancellations, parking spot reservations, and earned incentives. Additionally, it includes an **in-app chat feature** for direct communication between matched drivers and riders to coordinate pickup/drop-off details.

## **4.6 Transparent User Profile**

Screens screenshot of a phone

AI-generated content may be incorrect. A screenshot of a phone

AI-generated content may be incorrect.

The "MMU Rides" app promotes **transparency** by clearly **displaying user and vehicle details**. The User Profile View shows the user’s name, rating, and reviews, while the My Vehicles View lists each vehicle’s make, model, color, seats, and status, helping passengers verify rides and drivers manage their fleet.

## **4.7 View Ride History**

A screenshot of a phone

AI-generated content may be incorrect.

"MMU Rides" includes a **Ride History feature** that allows users to **view a chronological log of all past trips**, detailing whether they were a driver or passenger, along with **information on seats offered/used, parking spots reserved, and rewards earned**.

## **4.8 Rating & Feedback System**

A screenshot of a phone

AI-generated content may be incorrect. A screenshot of a chat

AI-generated content may be incorrect.

"MMU Rides" incorporates a **User Ratings & Feedback System** allowing both **drivers and passengers to rate and provide comments on each other after each completed ride**. This system helps maintain accountability and build trust within the community.

## **4.9 Incentives & Reward System**

A screenshot of a phone

AI-generated content may be incorrect. A screenshot of a phone

AI-generated content may be incorrect.

"MMU Rides" incorporates an **Incentives & Reward System**. Users **earn points per trip** (e.g., 10 for drivers, 5 for riders), which can be **redeemed for benefits like priority parking, bookstore vouchers, or cafeteria discounts**. A **leaderboard** further incentivizes participation with **badges and extra rewards** for top carpoolers.

## **4.10 Safety & Emergency Features**

A screenshot of a phone

AI-generated content may be incorrect. A screenshot of a phone

AI-generated content may be incorrect.

"MMU Rides" prioritizes user safety with features including an **SOS button for emergencies** that alerts pre-selected contacts and shares real-time ride details. Users can also **add and manage emergency contacts** directly within the app. Furthermore, the system supports **trip sharing**, allowing users to share their real-time location with family and friends, and offers an **auto-recording function** for audio during trips to enhance security. The app also provides **safety tips** to guide users on best practices for safe carpooling.

A screenshot of a phone

AI-generated content may be incorrect.

## **4.11 Admin Dashboard**

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

The "MMU Rides" system features a comprehensive **web-based Admin Dashboard and Reporting functionality**, allowing administrators to effectively **monitor and manage** the platform. The dashboard immediately displays key performance indicators (**KPIs**) such as **total rides, active carpools, parking occupancy, and reward redemptions**, alongside visual trends for **ride usage** and **parking utilization** with customizable date ranges. Complementing this, the reporting interface enables the **generation and export (CSV/PDF) of custom reports** on various aspects like ride usage and user growth, supported by a "Recent Activity" log for tracking real-time system events.

## **4.12 Prototype Source**

User Prototype

<https://jasonow718.github.io/Ride-Sharing-Platform-User-Prototype/>

Admin Prototype

<https://jasonow718.github.io/Ride-Sharing-Platform-Admin-Prototype/>

A screenshot of a computer

AI-generated content may be incorrect.

* Right-click anywhere on the webpage content displayed within the browser window.
* From the context menu that appears, select "Inspect" to open the developer tools panel.

A screenshot of a computer

AI-generated content may be incorrect.

* Within the developer tools, locate the device toolbar icon (usually a small mobile phone or tablet icon) at the top or top left of the panel and click it.
* Once the device toolbar is active, a dropdown menu for selecting device sizes should appear near the top of the webpage view. Choose "iPhone 12 Pro" from this dropdown to simulate the prototype on that specific mobile device size.

## **5.0 Summary**

|  |  |
| --- | --- |
| **Kano Category** | **Features** |
| **Dissatisfiers (Must-Haves)** | 1. Ride Offer & Request 2. Safety & Emergency Features 3. Transparent User Profiles |
| **Satisfiers (Expected Features)** | 1. User Onboarding & Digital ID 2. Manual Ride Matching 3. Push Notifications & In-App Messaging 4. Ride History Tracking 5. User Ratings & Feedback System |
| **Delighters (Unexpected Joys)** | 1. Real-Time Campus Parking Availability 2. Reward Points & Leaderboard System |