Design Document

for

PedalPal

Version 1.0

Prepared by

Group # 4 **Group Name: Bit Brewers**

Raghav Manglik	**220854**	raghavkmanglik@gmail.com
Amogh Bhagwat	**220288**	amogh.2004b@gmail.com
Srishti Chandra	**221088**	chandra.srishti2403@gmail.com
Wadkar Srujan Nitin	**221212**	srujanwadkar@gmail.com
Anaswar K B	**220138**	anaswarkb013@gmail.com
Khushi Gupta	**220531**	khushi07g@gmail.com
Ananya Singh Baghel	**220136**	ananyabaghel2004@gmail.com
Pathe Nevish Ashok	**220757**	nevu.pathe1234@gmail.com
Debraj Karmakar	**220329**	debraj2003jsr@gmail.com
Kaneez Fatima	**220496**	kaneezfatimamehdi7@gmail.com

Course: **CS253**
Mentor TA: **Mr. Bharat**
Instructor: **Prof. Indranil Saha**
Date: **February 9, 2024**

Contents

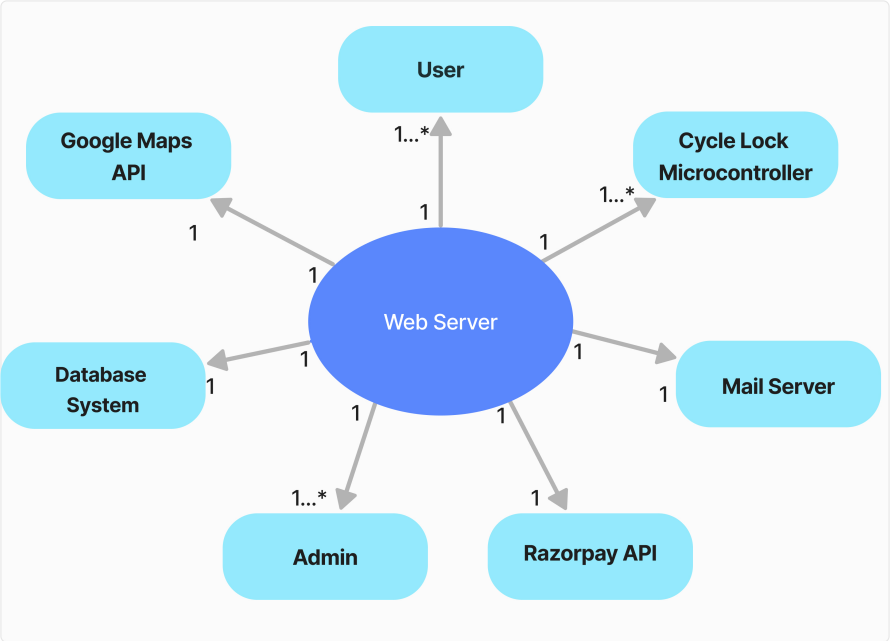
1. Revisions

Version	Primary Author(s)	Description of Version	Date Completed
v1.0	Raghav Manglik Amogh Bhagwat Srishti Chandra Wadkar Srujan Nitin Pathe Nevish Ashok Debraj Kamakar Khushi Gupta Ananya Baghel Anaswar K B Kaneez Fatima	First version of the Software Design Document	09/02/23

2. Context Design

2.1 Context Model

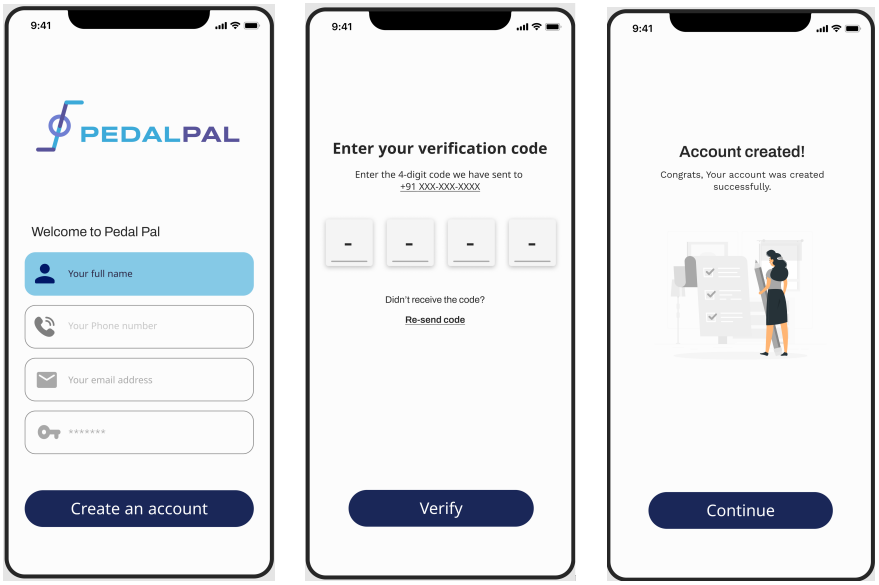
The context model of PedalPal is shown in the diagram. The context model shows the various entities that interact with the system, and the relationships between them. The entities are the user, the admin, the database, the payment gateway, the map, and the cycle lock. The user interacts with the system through the mobile app, while the admin interacts with the system through the web app. The database stores all the data of the system, while the payment gateway adds payment functionality. The Google Maps API adds the ability to display the map of IIT Kanpur campus along with the cyclehubs, and the microcontroller is used for the cycle lock functionality. The mail server is used to send password reset mails and reminder mails to the user.



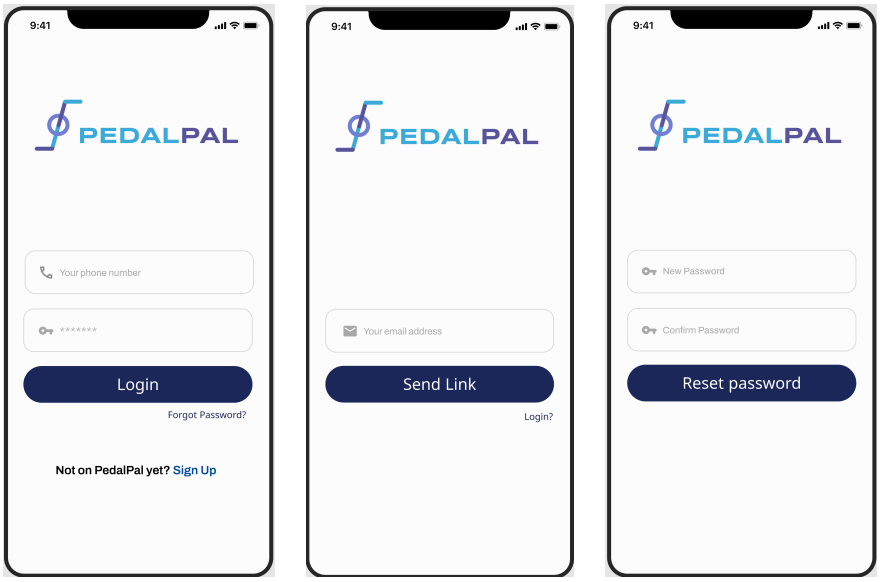
2.2 Human Interface Design

2.2.1 Login and Registration Pages

The registration user interfaces are shown here.

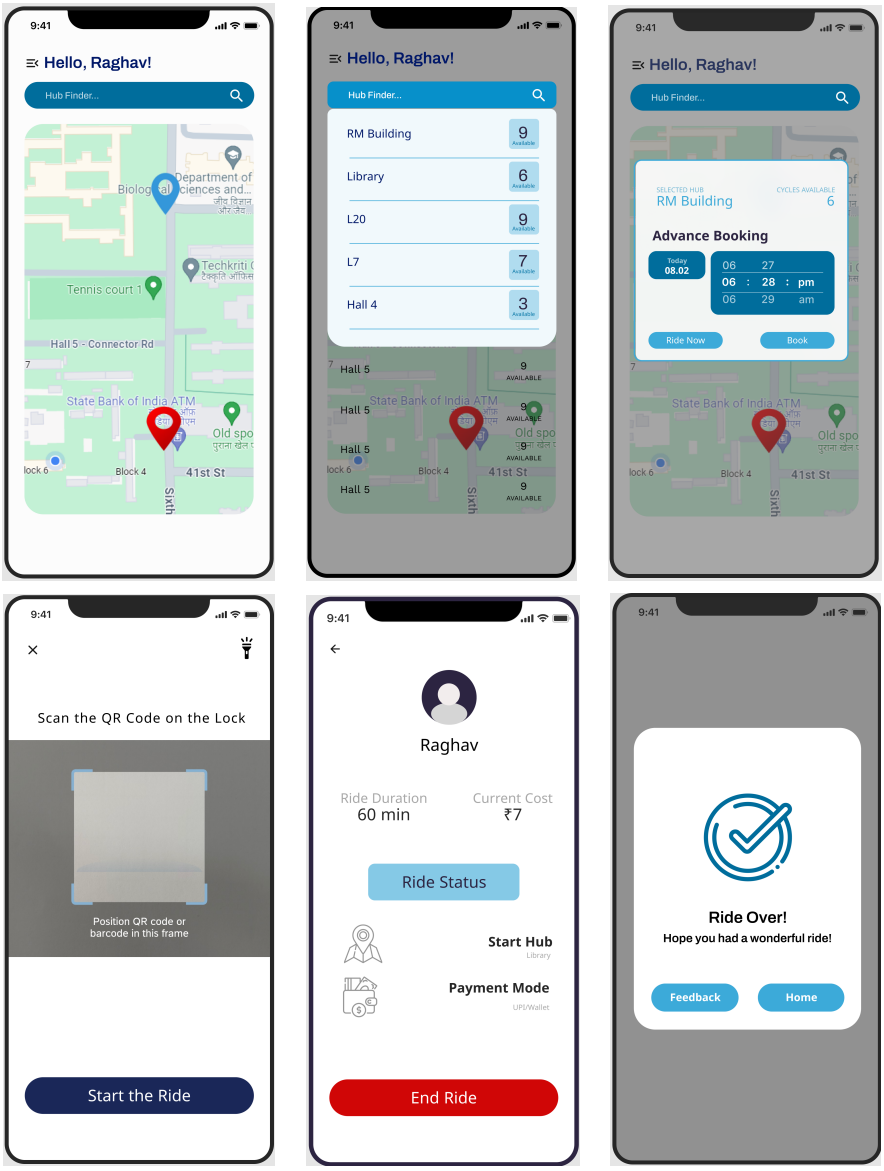


The login workflow is shown here. If the user has not created an account yet, they can click on Sign Up to create a new accout. If the user has forgotten their password, they can reset it using the ”Forgot Password” button.



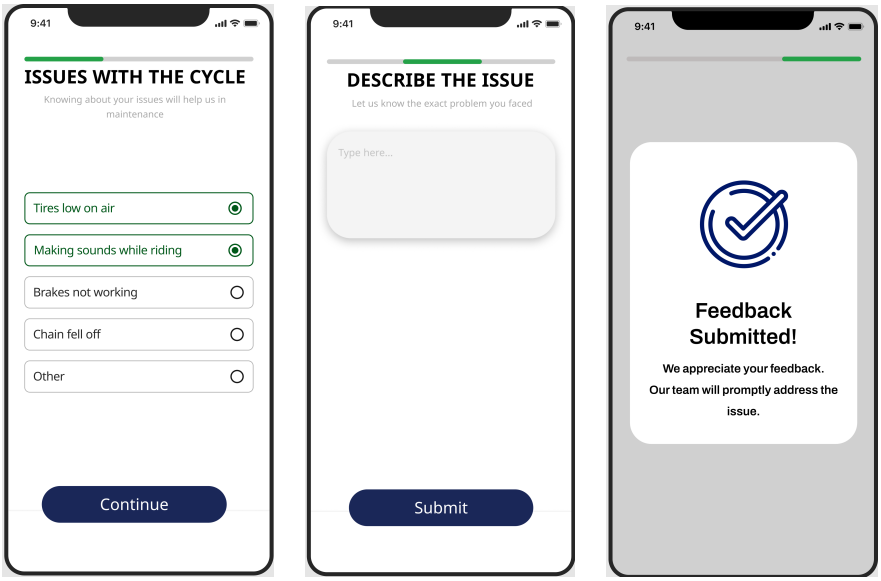
2.2.2 Booking a Ride

The user sees a map which shows the location of the user along with the hubs present. They can either choose the hub on their map, or search it from the search box. Once a hub is selected, the details of the hub are displayed, including the number of available cycles. The user can choose to start a ride instantly, or book a cycle for a later time (available only for subscribed users). If the user wishes to book for later, they are presented with a time picker. If the user wishes to ride instantly, they are presented with a QR code which they can scan to unlock the cycle. During the ride, statistics like ride time and current cost are shown to the user. Once the ride is over, the user can end the ride and the app will show the user the fare for the ride, and ask for feedback. The user can also report any issues with the cycle during the ride.



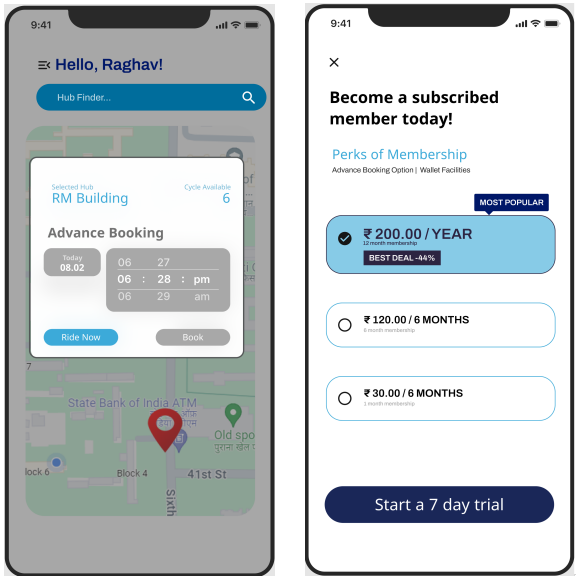
2.2.3 User Feedback

The user can select any issue they faced during the ride and provide a description. The user can also provide feedback on the ride.



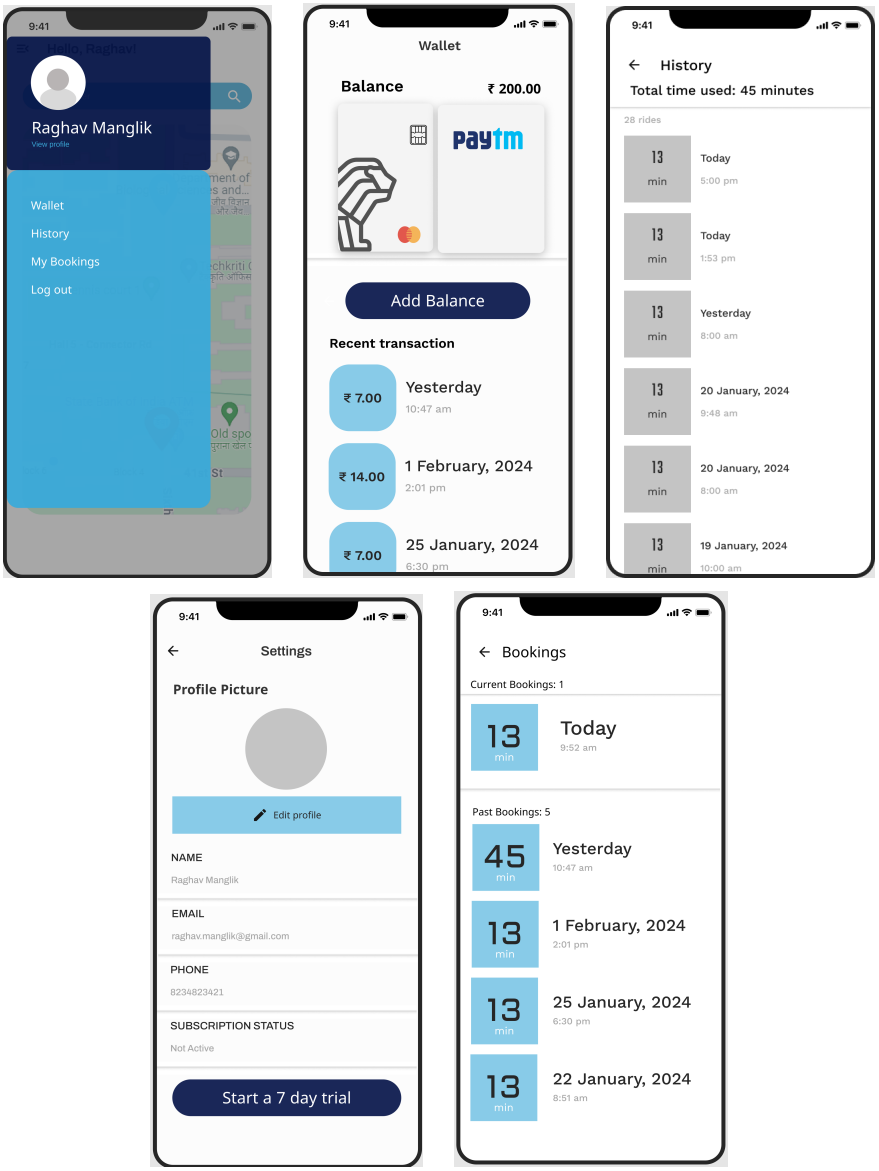
2.2.4 Subscription Model

The option of booking rides in advance is only available for subscribed users. This option is greyed out in case the user is not subscribed. If they tap on any greyed out option, an advertisement for the subscription model is shown.



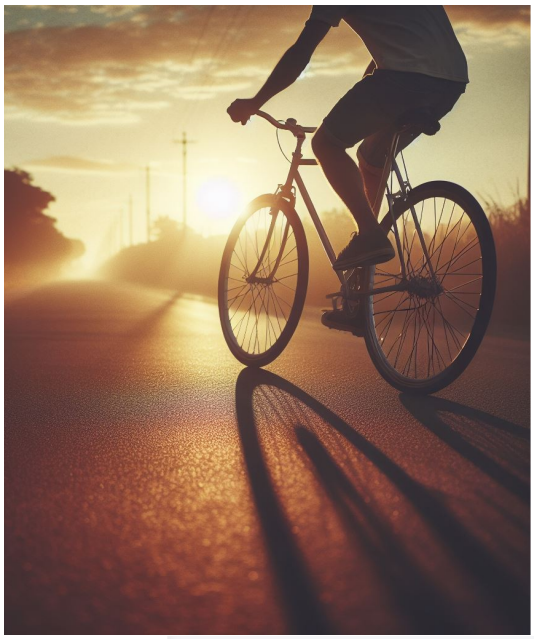
2.2.5 Settings and Analytics View

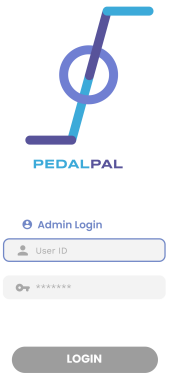
The navigation bar presents the user with options to view their wallet details, ride history, settings, bookings, and to log out.

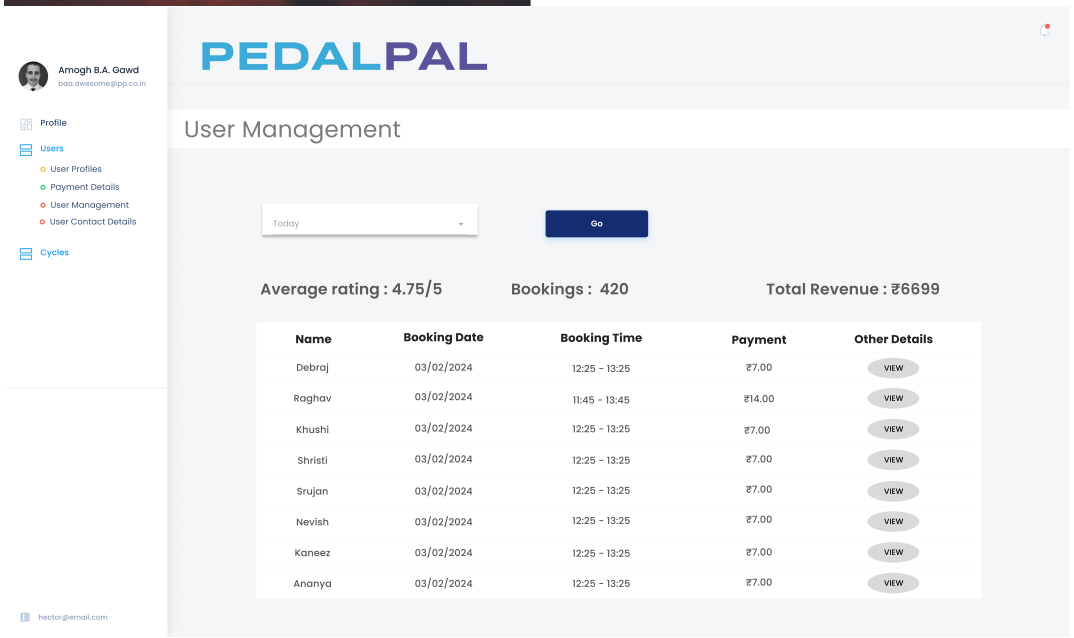


2.2.6 Admin Interface

The admin interface is used to manage the hubs and cycles. The admin can add new hubs, view the details of the hubs, and manage the cycles. The admin can also view the feedback and issues reported by the users.




PEDALPAL
Admin Login
User ID
****
LOGIN

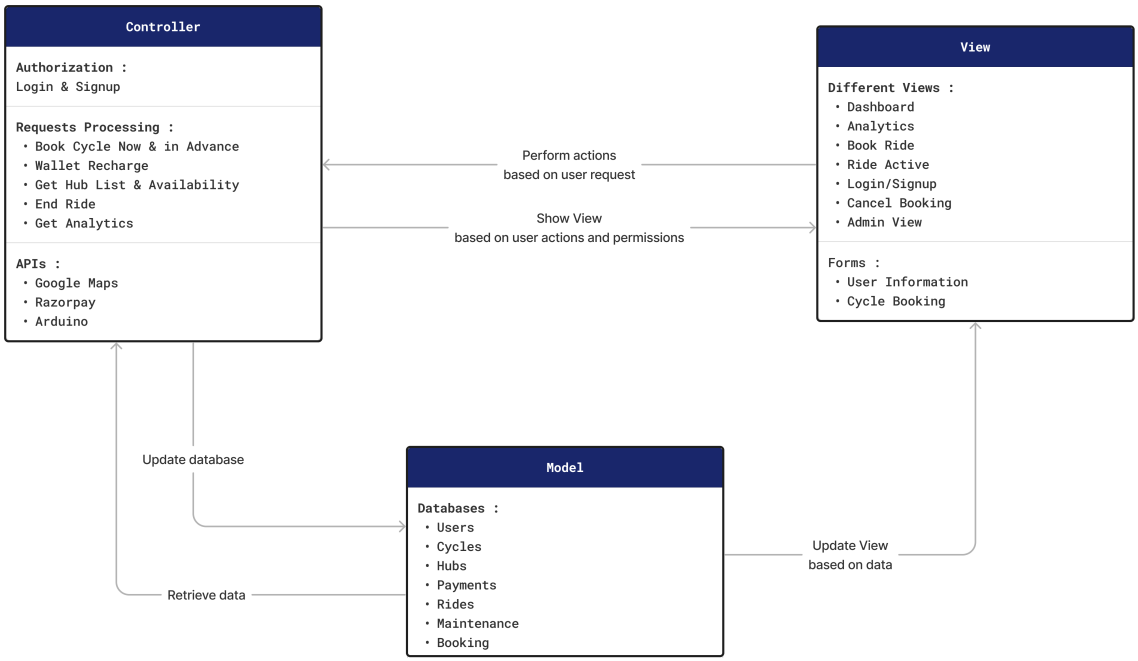

Amogh B.A. Gawd
baa.awesome@pp.co.in
Profile
Users
User Profiles
Payment Details
User Management
User Contact Details
Cycles
hector@email.com
PEDALPAL
User Management
Today
Go
Average rating : 4.75/5
Bookings : 420
Total Revenue : ₹6699
Name
Booking Date
Booking Time
Payment
Other Details
Debraj 03/02/2024 12:25 - 13:25 ₹7.00 VIEW
Raghav 03/02/2024 11:45 - 13:45 ₹14.00 VIEW
Khushi 03/02/2024 12:25 - 13:25 ₹7.00 VIEW
Shristi 03/02/2024 12:25 - 13:25 ₹7.00 VIEW
Srujan 03/02/2024 12:25 - 13:25 ₹7.00 VIEW
Nevish 03/02/2024 12:25 - 13:25 ₹7.00 VIEW
Kaneez 03/02/2024 12:25 - 13:25 ₹7.00 VIEW
Ananya 03/02/2024 12:25 - 13:25 ₹7.00 VIEW

3. Architecture Design

The overall architecture of our app follows the Model-View-Controller model. Given the complexity of the app, each functionality is sub-divided into smaller tasks, which are achieved by the Pipe-Filter architecture model.

The model-view-controller architecture is as follows:

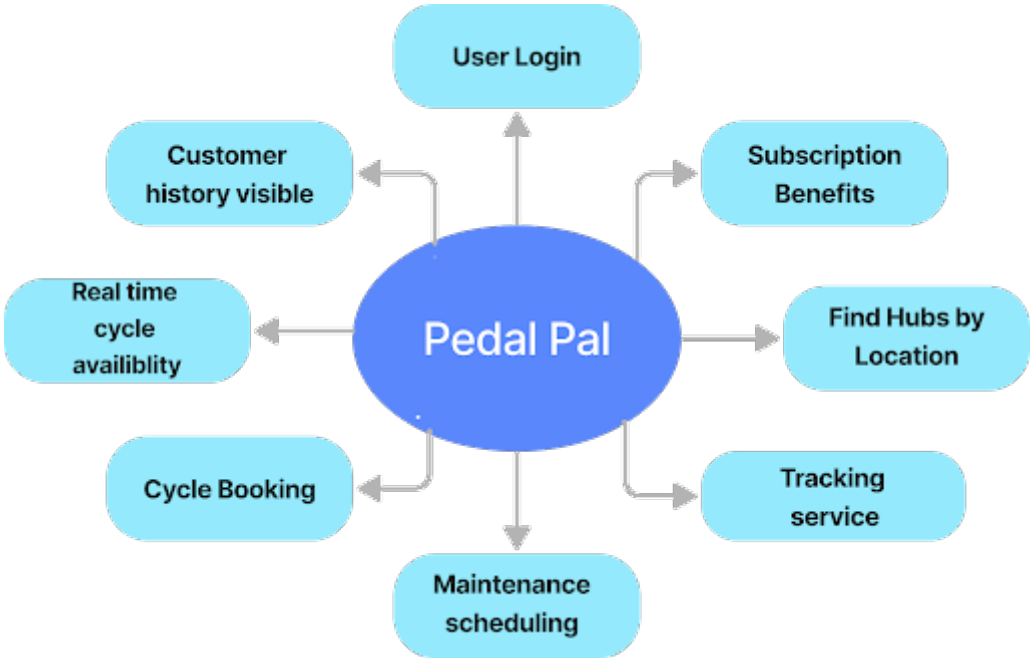
- **Model:** The model is the data layer of the app. It is the only layer that directly communicates with the database. It is responsible for the storage and retrieval of data from the database. We plan to use PostgreSQL as our database, while using Django as our backend framework to interact with the database.
- **View:** The view is the presentation layer of the app. It is responsible for the display of data to the user. It is also responsible for taking input from the user. The view for app users will be a mobile app, while the view for the admin will be a web app. The mobile app is planned to be built using Flutter, while the admin webpage is automatically generated by Django.
- **Controller:** The controller is the logic layer of the app. It is responsible for processing the data and deciding the flow of the app. The controller will be built using Django in the Python language. Google Maps API will be used for the map functionality, and RazorPay API will be used for the payment functionality. The hardware component of opening cycle lock will be handled by a microcontroller such as Arduino.

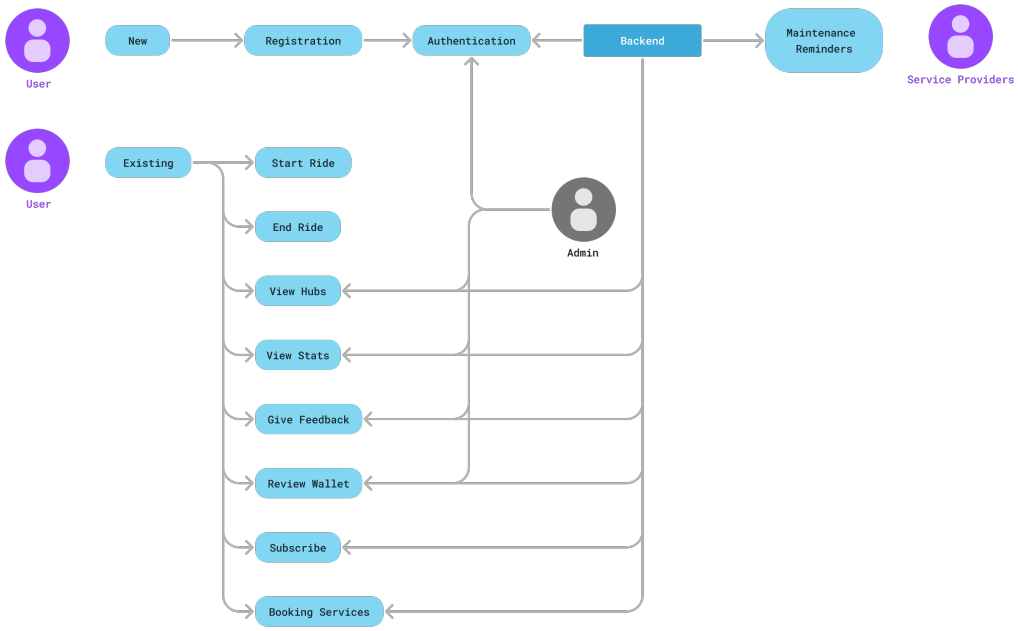


4. Object Oriented Design

4.1 Use Case Diagrams

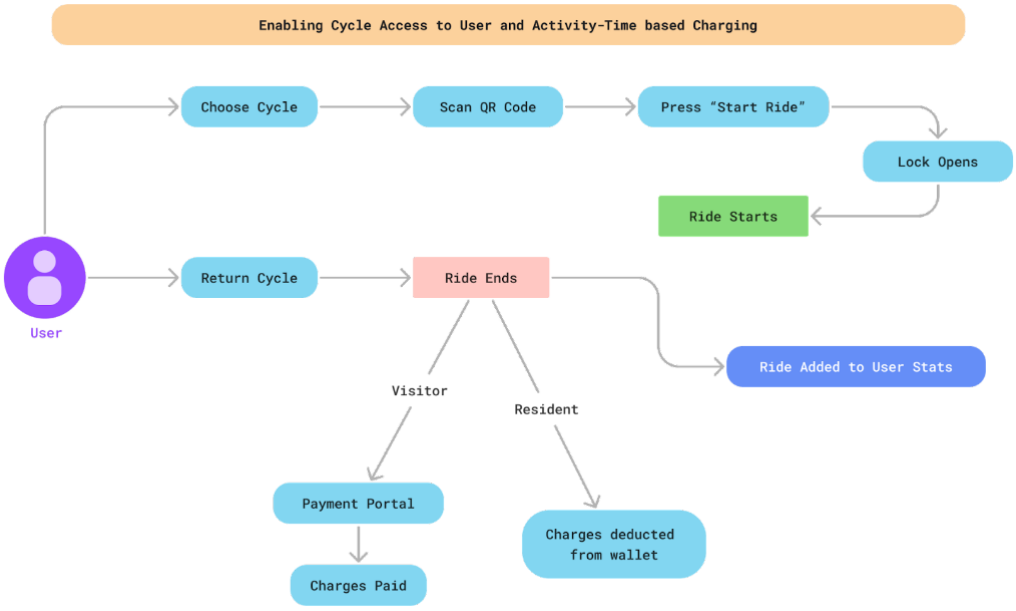
Overview of the use cases of PedalPal is as follows



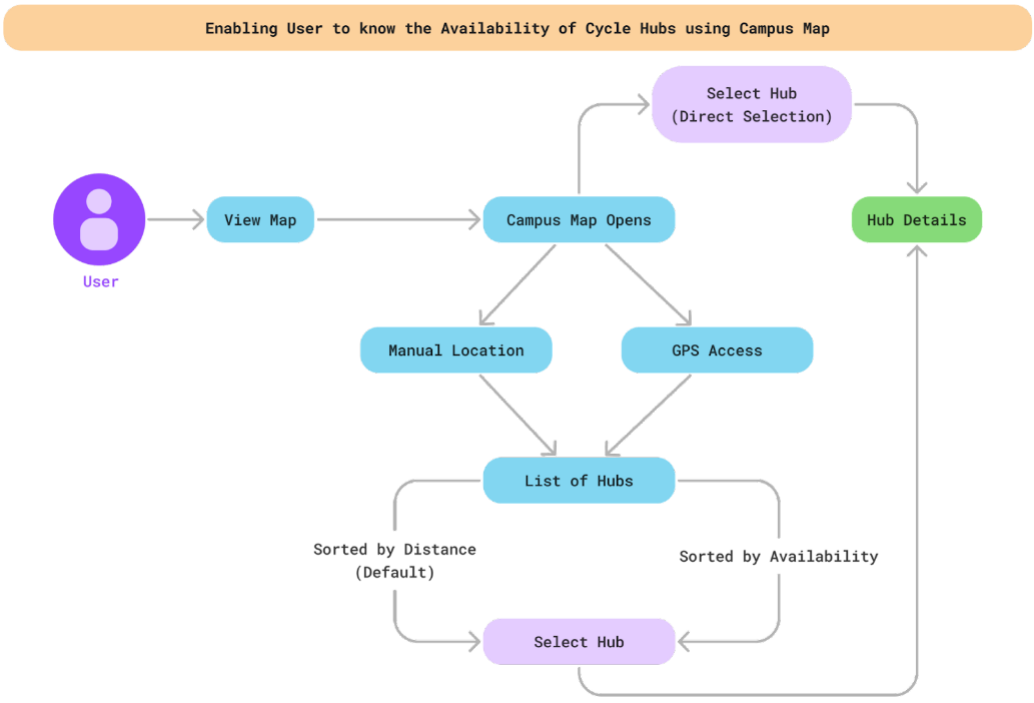


The various use cases are as follows

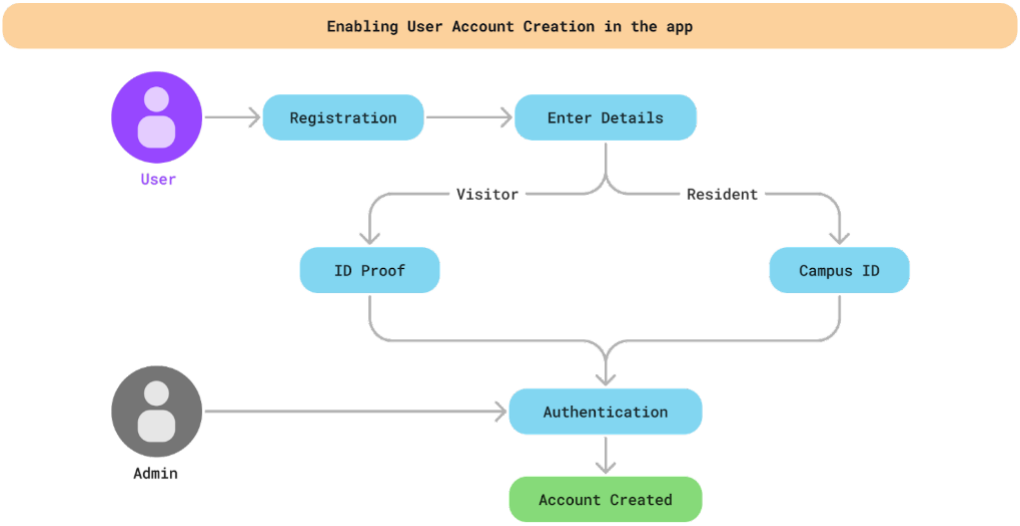
4.1.1 Use Case 1 (U1) - Ability to Start Ride and End Ride



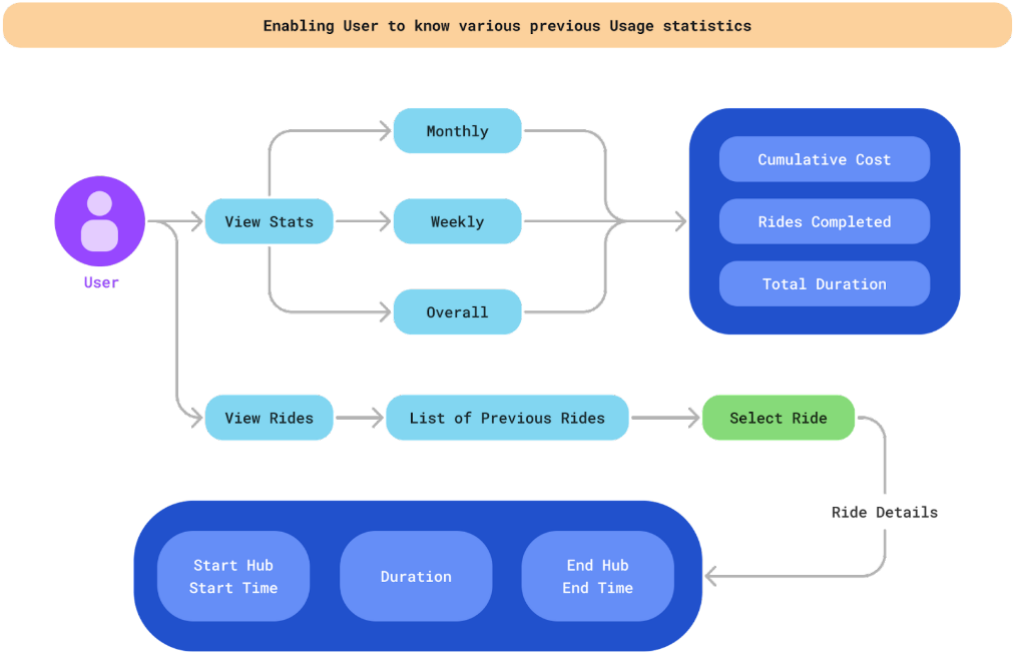
4.1.2 Use Case 2 (U2) - Ability to View Nearby Hubs on a Map



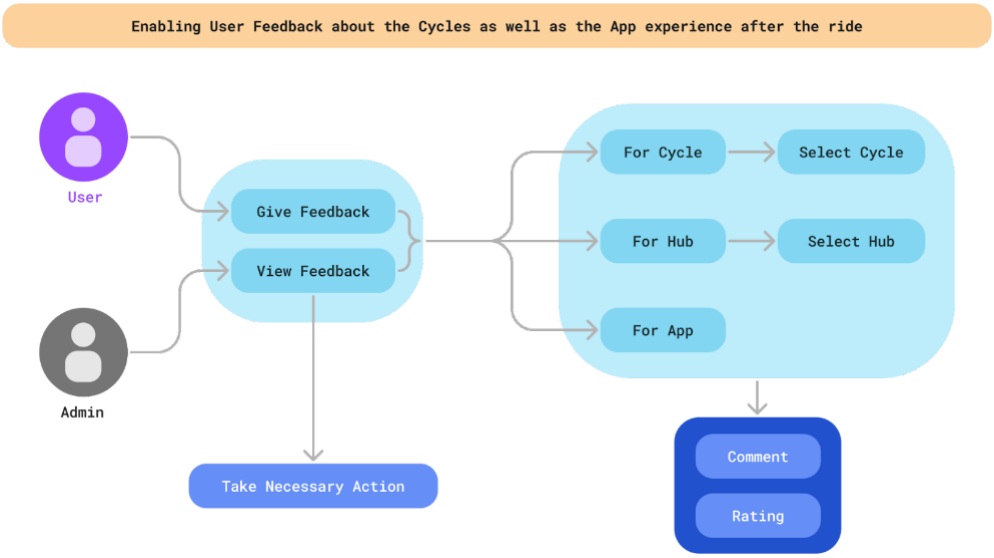
4.1.3 Use Case 3 (U3) - Ability to Create a new Account



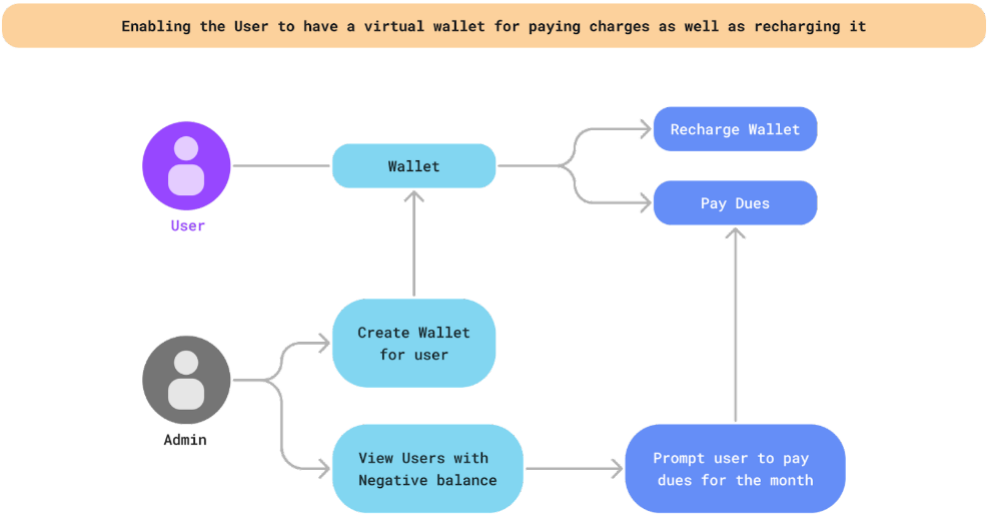
4.1.4 Use Case 4 (U4) - Ability to View Analytics



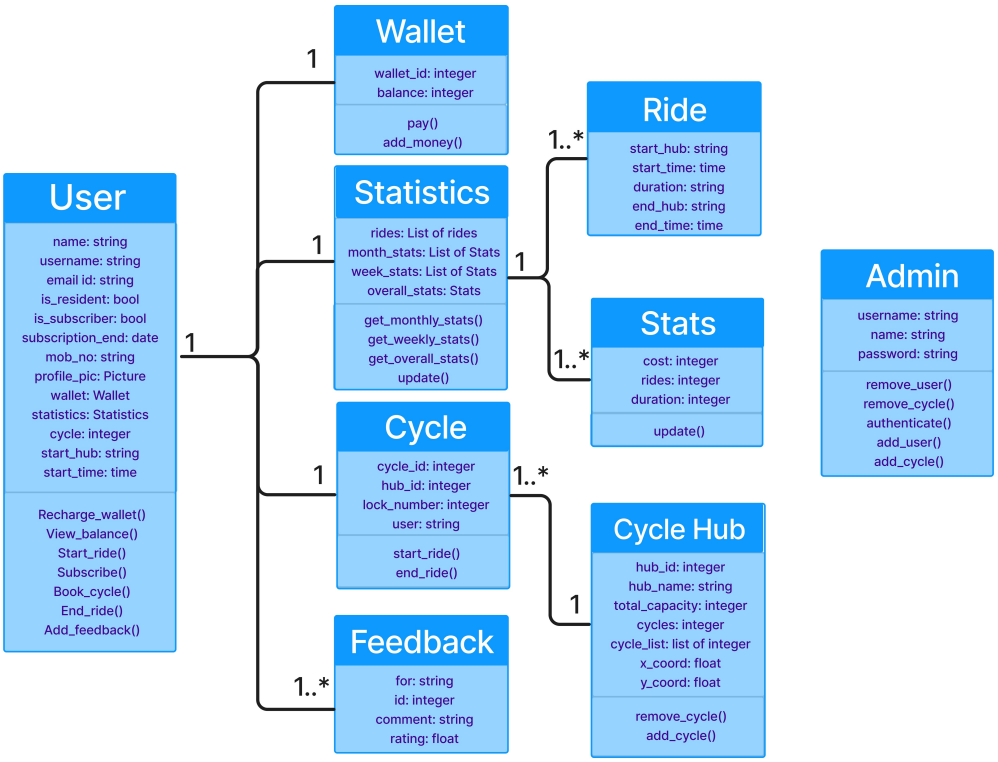
4.1.5 Use Case 5 (U5) - Feedback Mechanism



4.1.6 Use Case 6 (U6) - Ability to Use and Recharge Wallet

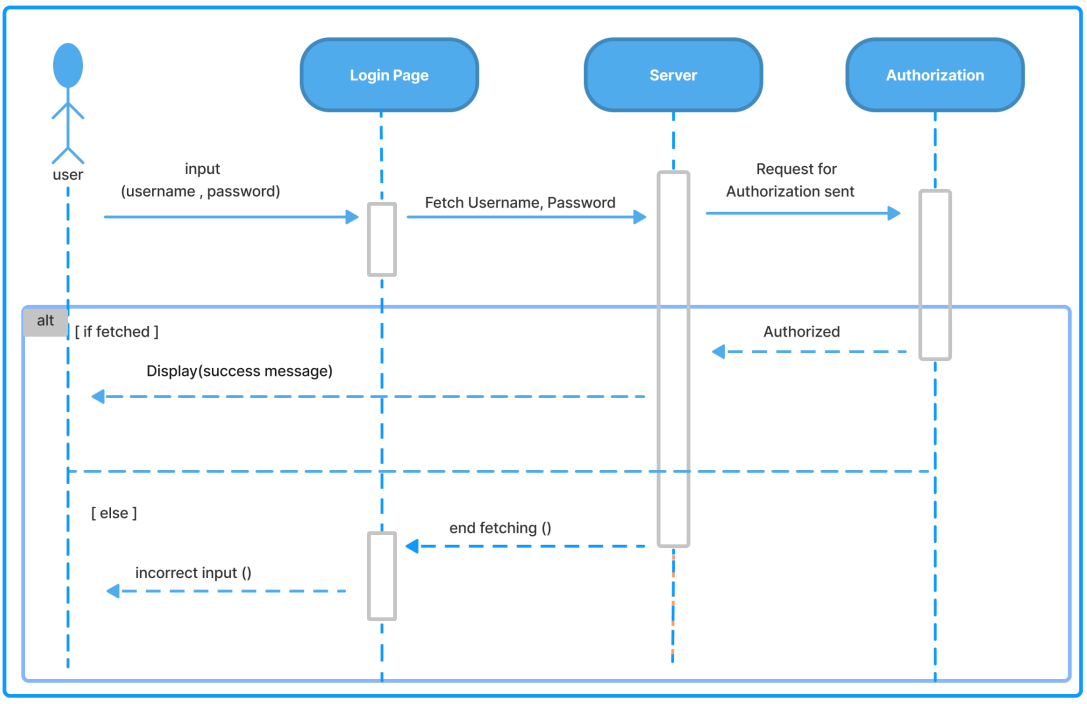


4.2 Class Diagrams

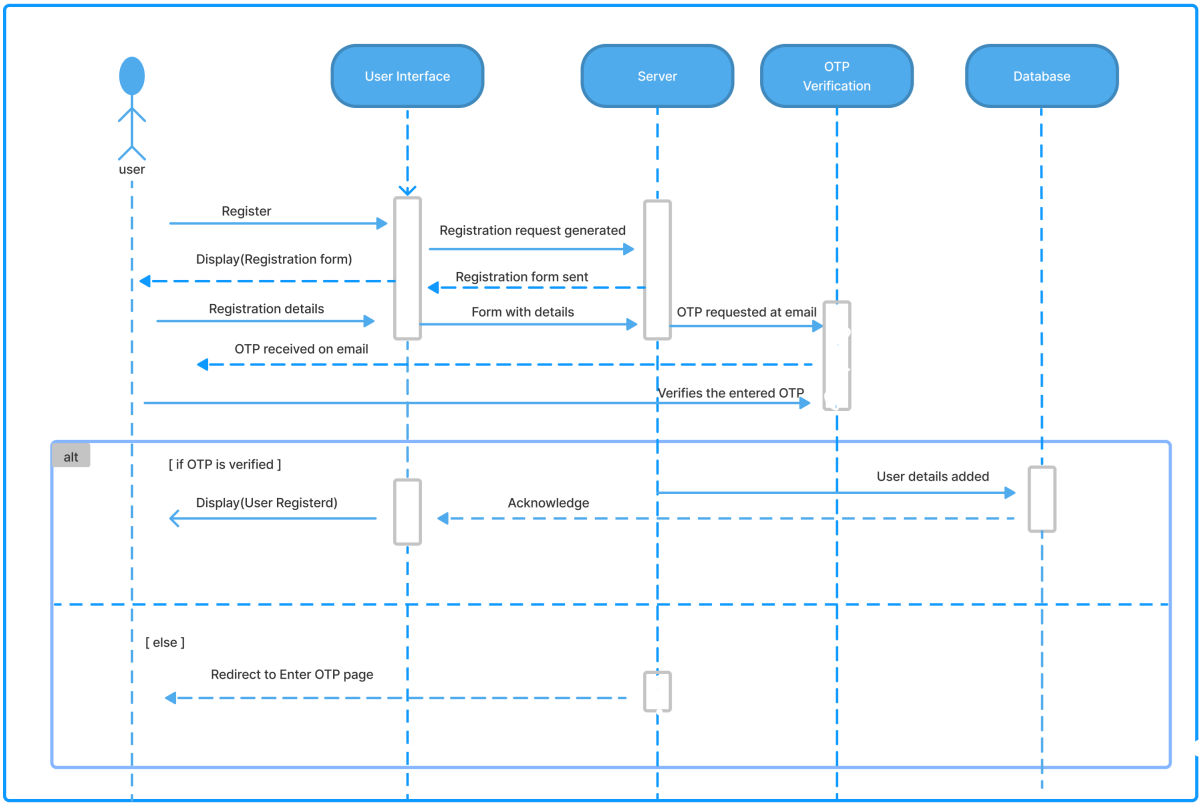


4.3 Sequence Diagrams

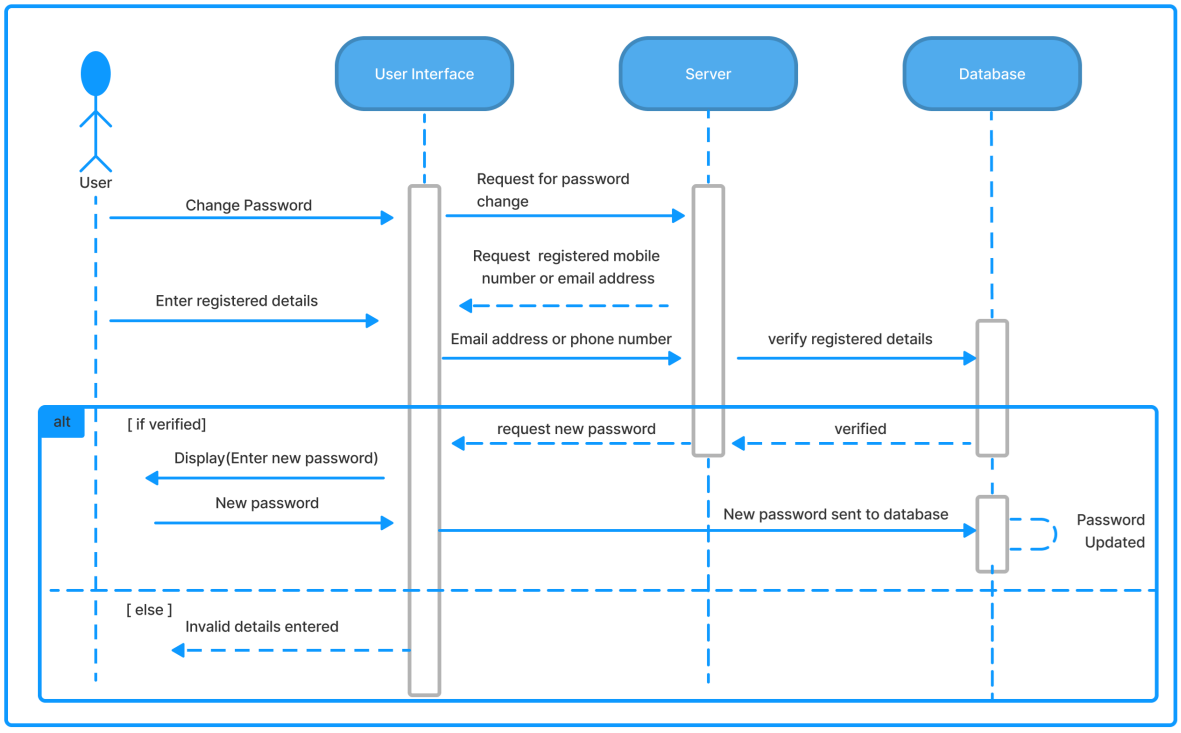
4.3.1 Login Sequence



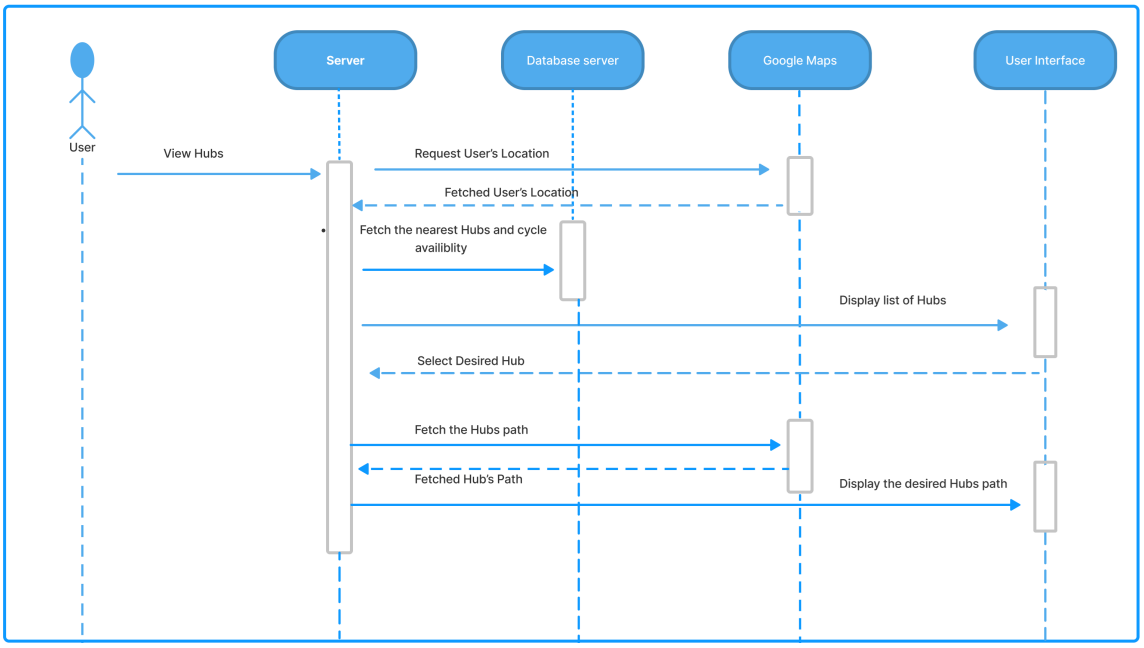
4.3.2 Registration Sequence



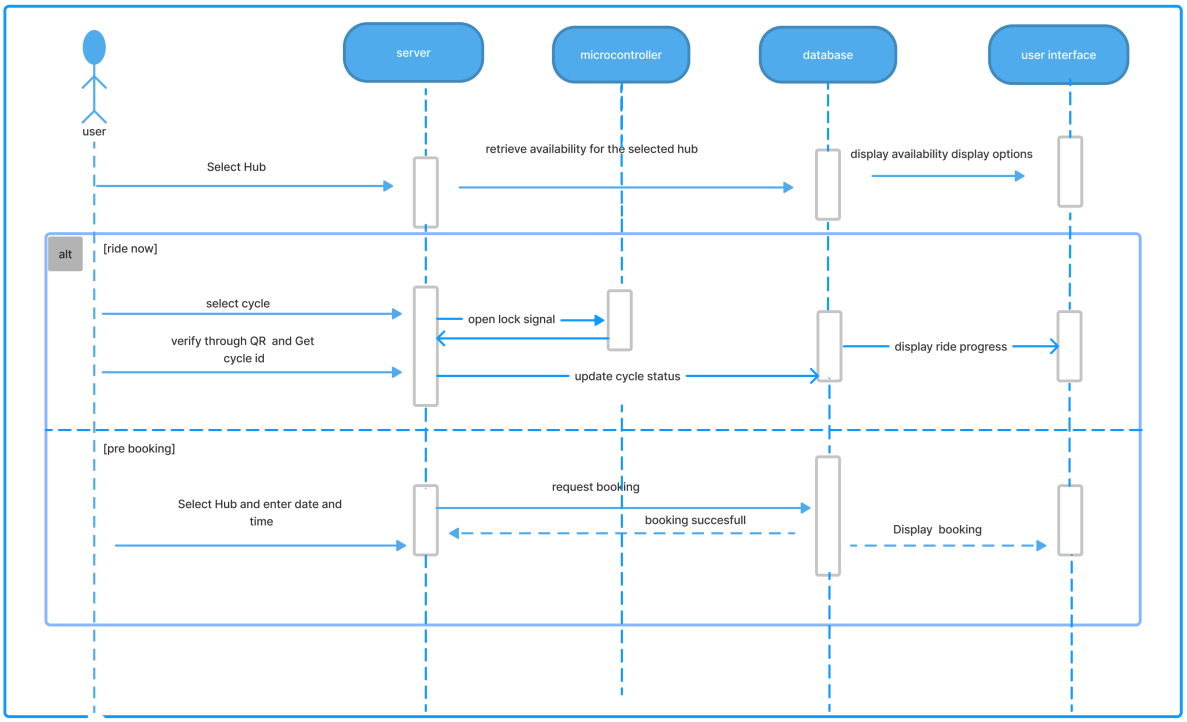
4.3.3 Reset Password Sequence



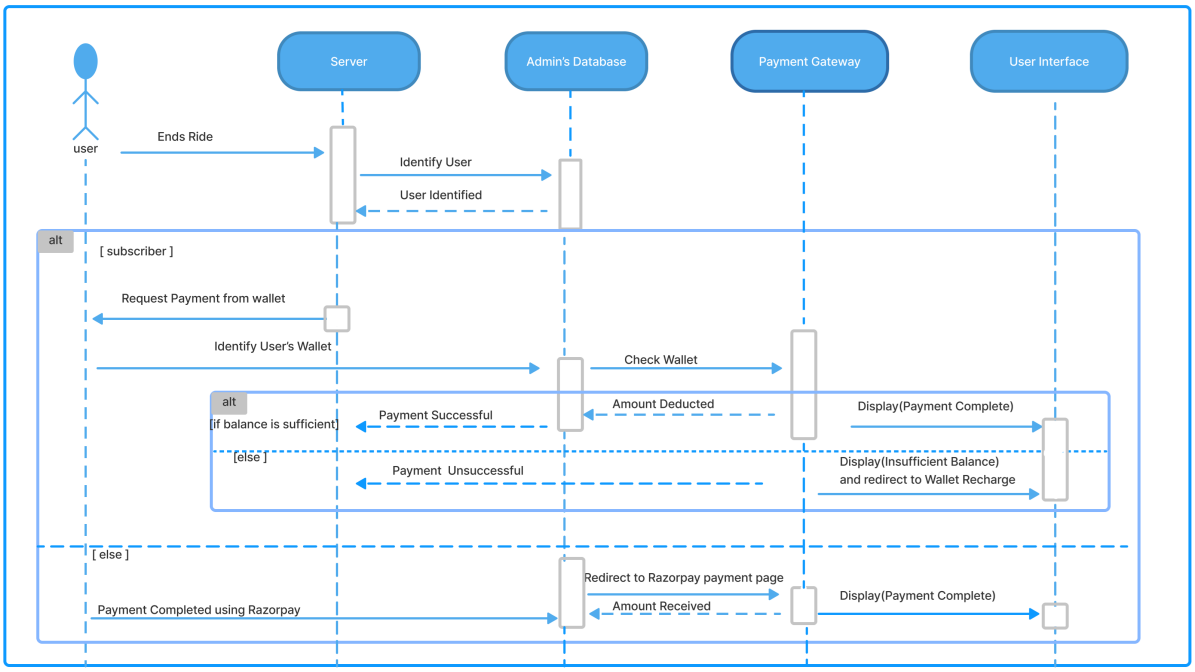
4.3.4 Fetch Cycle Hubs Sequence



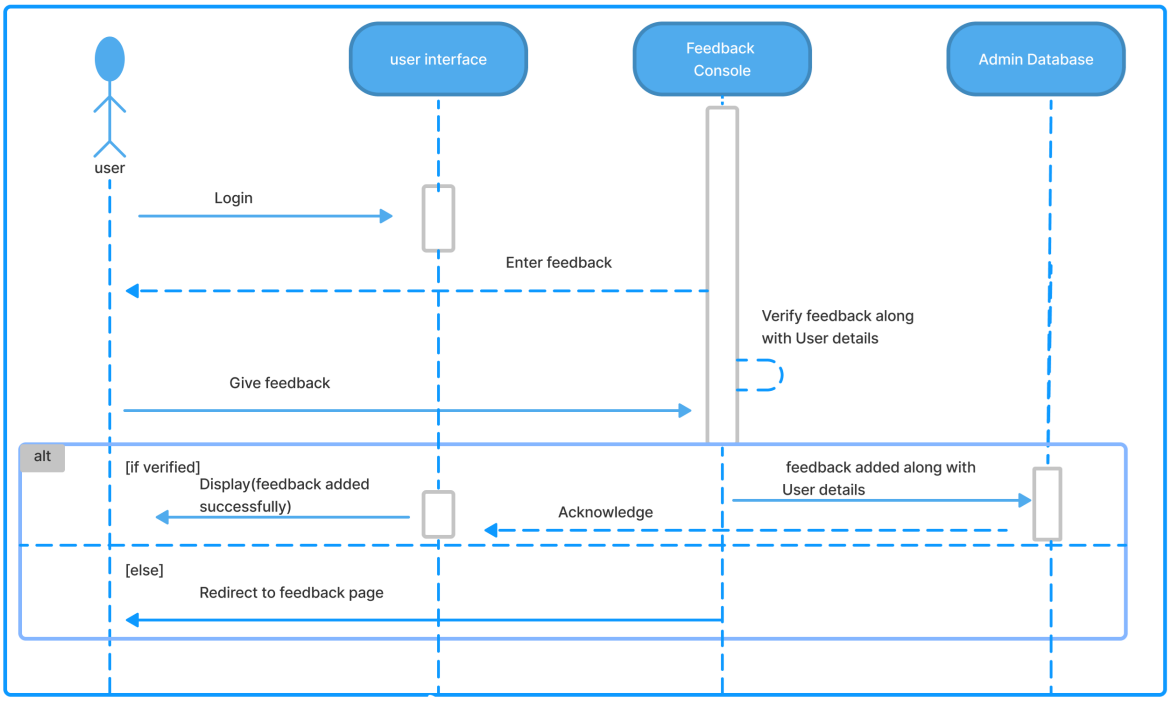
4.3.5 Book Ride Sequence



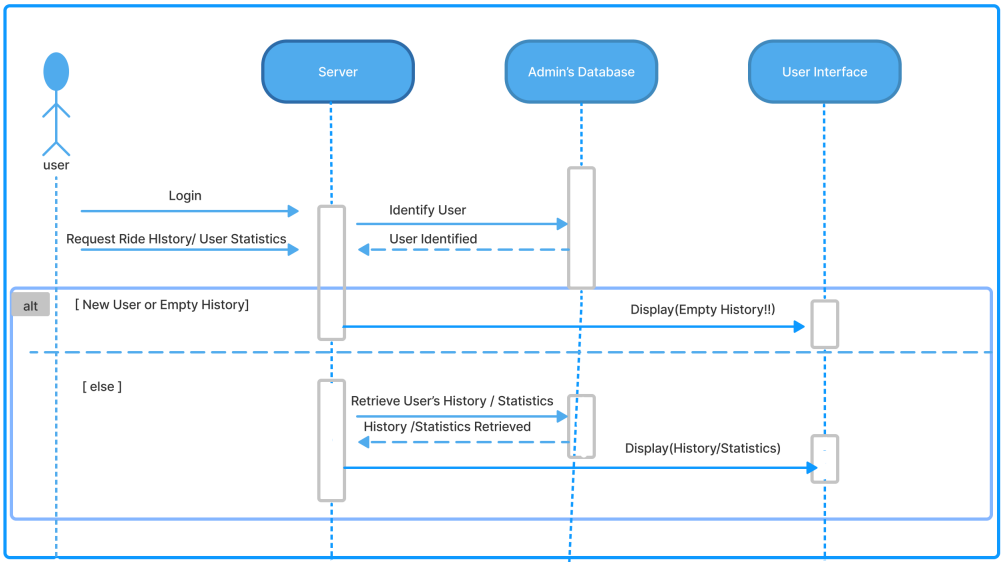
4.3.6 Payment Sequence



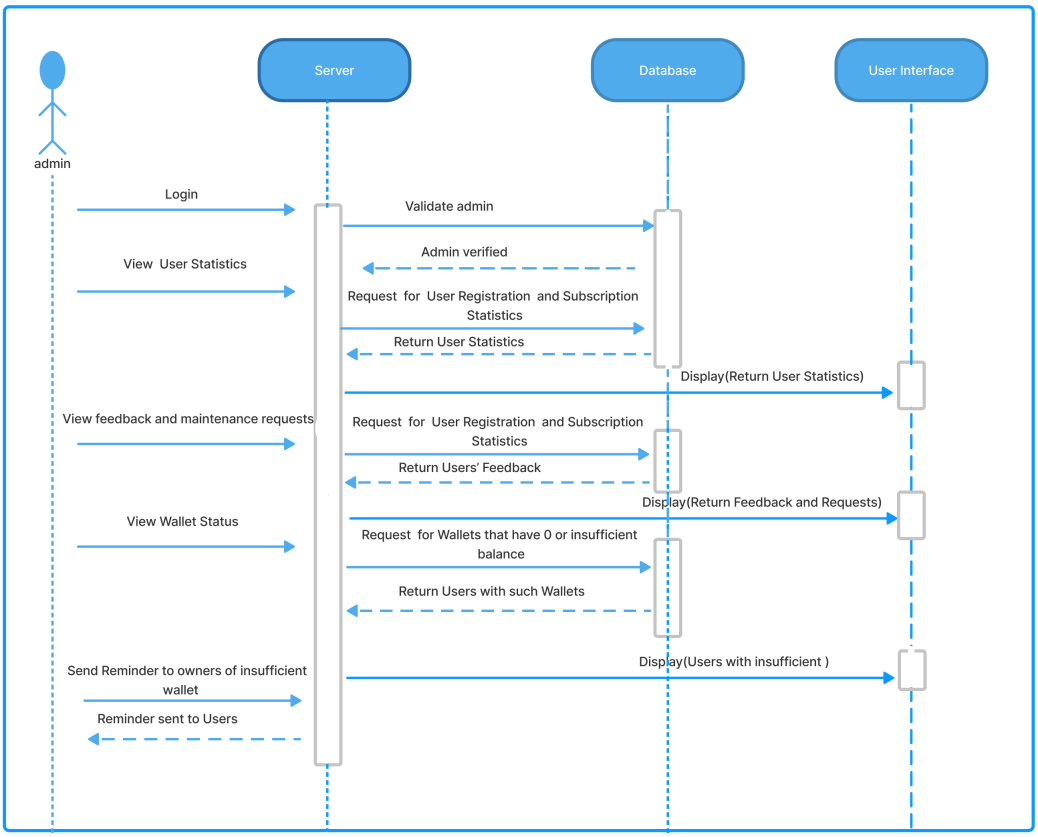
4.3.7 Feedback Sequence



4.3.8 View Analytics Sequence

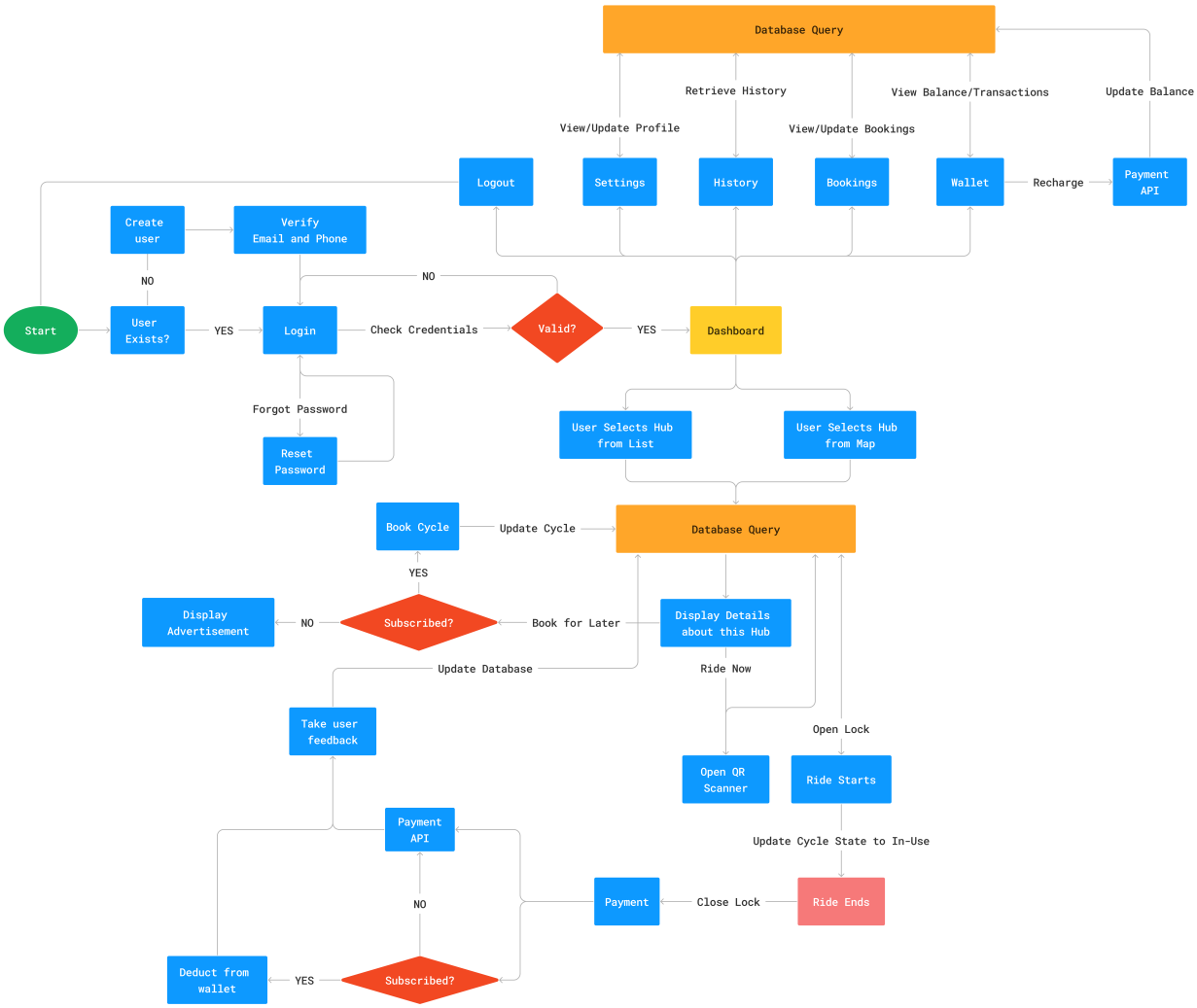


4.3.9 Admin Sequence



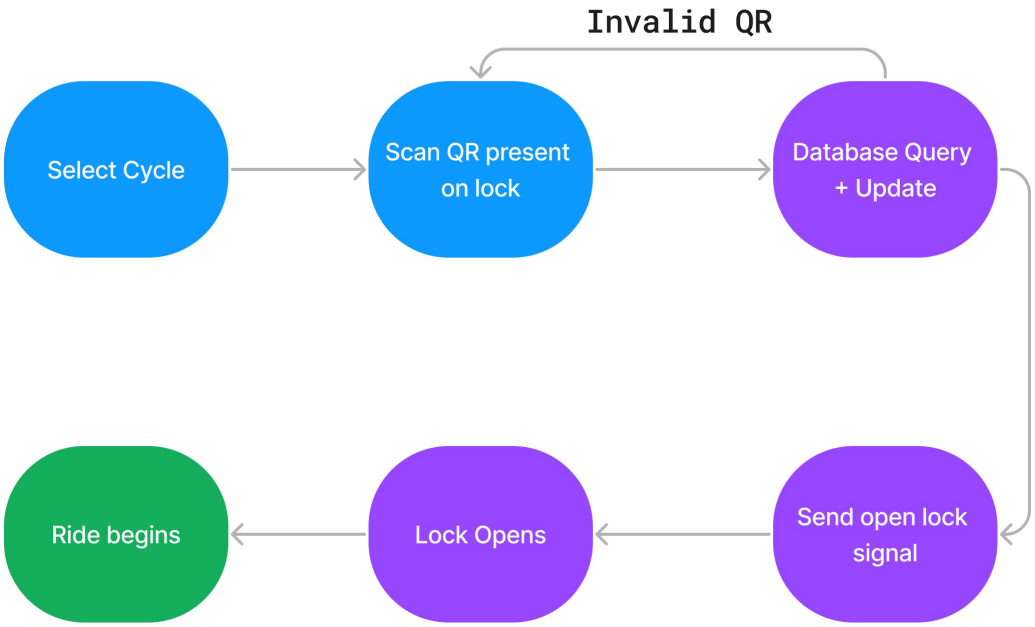
4.4 State Diagrams

4.4.1 Overall state diagram

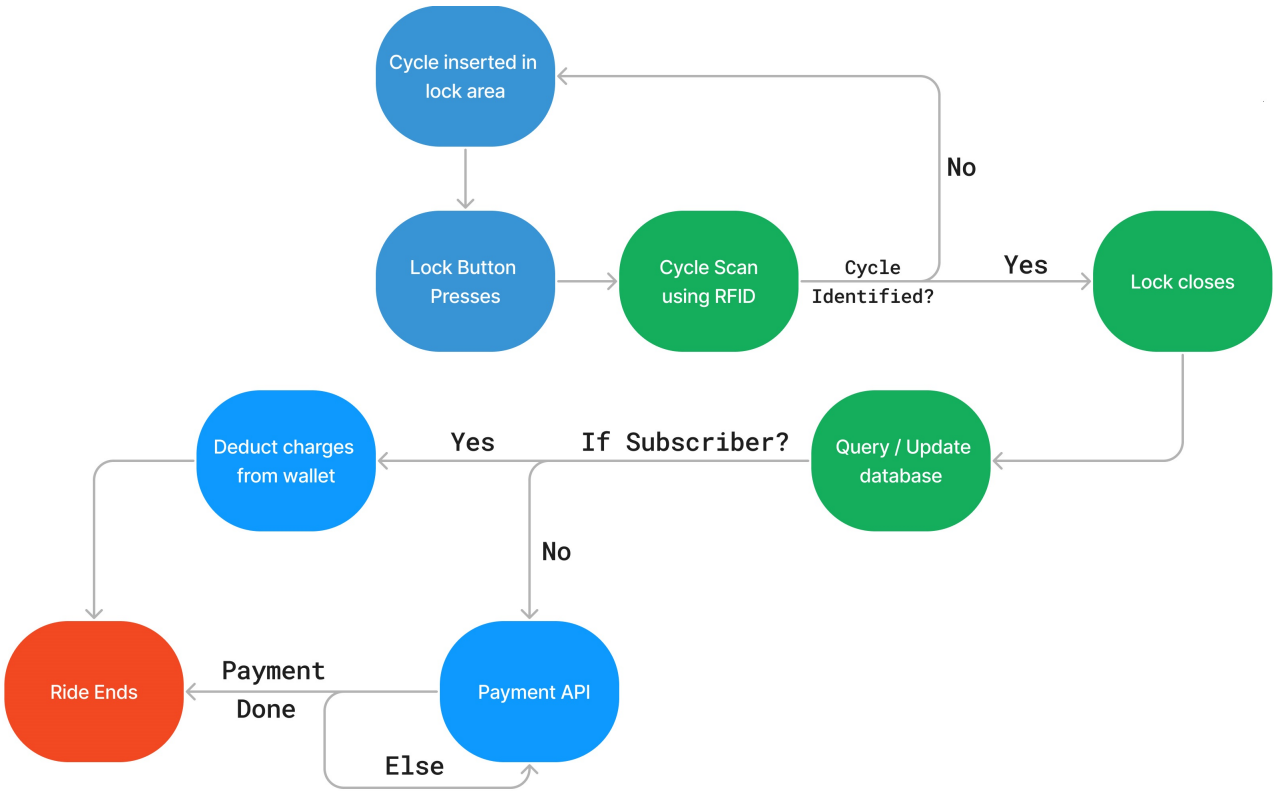


The image describes the high-level state diagram of the application from a user perspective. The customer can either be a subscribed customer or an unsubscribed customer. The customer can book a ride, view ride history, view wallet, view settings, view booking history, and give feedback. The orange boxes show a query to the database and the red diamonds show a decision point.

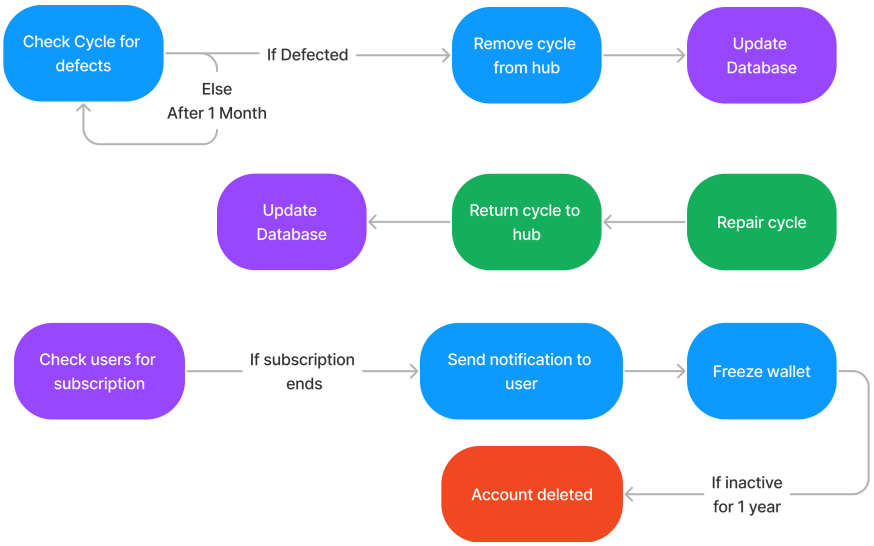
4.4.2 Ride Begin state diagram



4.4.3 Ride End state diagram



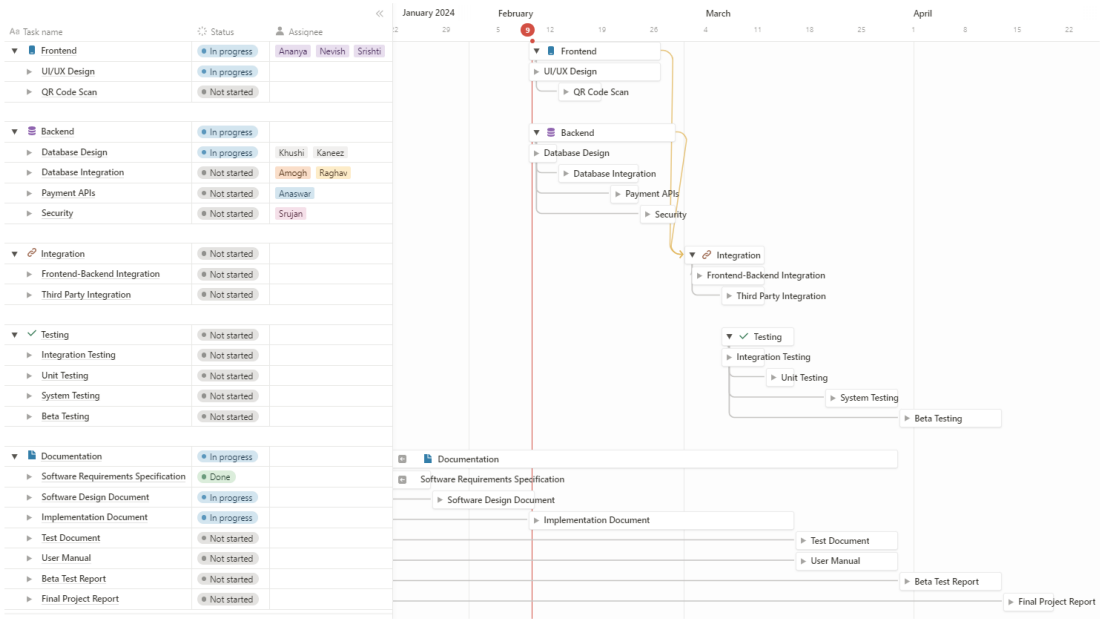
4.4.4 Miscellaneous State Diagrams



5. Project Plan

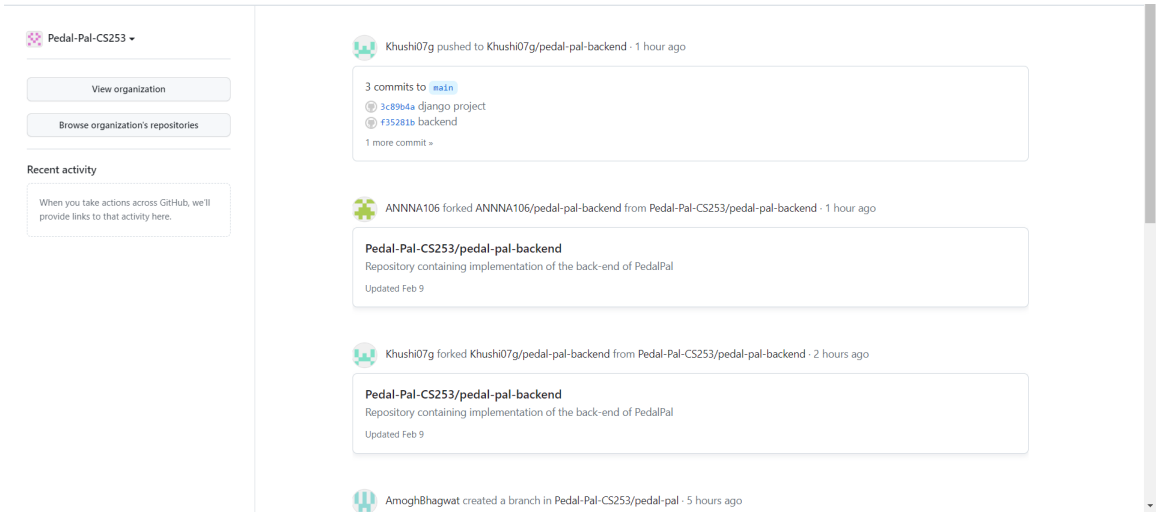
5.1 Project Planning

We are using Notion to track the progress and deadlines of the project. It also allows us to assign tasks to team members. The current Gantt chart looks like this:



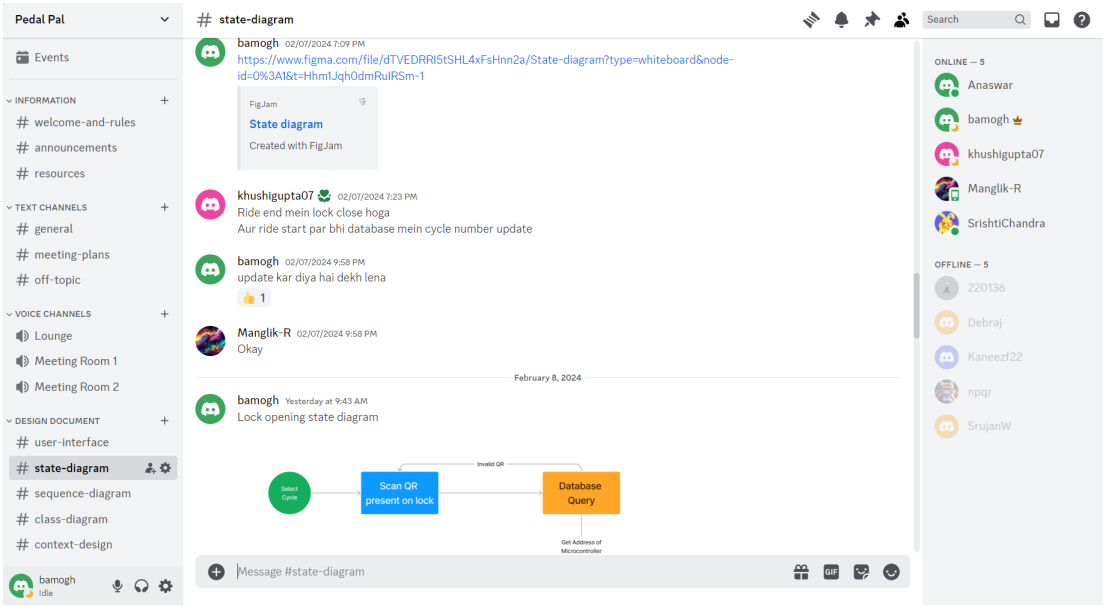
5.2 Code Collaboration

We are using GitHub to collaborate on the code. An organization is created for the purposes of this project and all team members are added to this organization. Separate repositories are maintained for the backend and frontend of our app.



5.3 Communication

We are using Discord for communication. It allows us to create separate channels for different topics, and keep data organised. It also allows us to have voice calls and video calls, which is useful for team meetings.



A. Group Log

S.No	Date	Timings	Venue	Description
1	31/01/2024	21:30 to 23:00	Discord	Studied deliverables for the design document. Divided work among teammates
2	03/02/2024	11:00 to 14:00	RM Building	Designed user interface Discussed project details like frameworks and other APIs
3	07/02/2024	21:30 to 22:00	Google Meet	Meet with TA to discuss details about the document.
4	08/02/2024	14:30 to 17:00	RM Building	Finalized all aspects of the design document Decided future plans