Group Project Final Report

Gethub: A Ride Booking System Project

Course:	Year / Section / Group Name / Number:
CMPE 103 – Object Oriented Programming	BSCPE 1-7 – Regularn't – Group 4

Team Members:

Name	Student ID	Role
Mark Christian Abucejo	2023-06660-MN-0	Lead Developer
Zybert Jio Sibolboro	2023-08274-MN-0	Full-stack Developer
Lorens Aron Mercado	2023-11266-MN-0	Back-end Developer
Renier Dela Cruz	2023-07658-MN-0	Back-end Developer
Kathlyn Estorco	2023-07480-MN-0	UI/UX Designer
Maeryl Venida	2023-08848-MN-0	UI/UX Designer
Luke Philip Lopez	2024-18529-MN-1	QA/Test Engineer

Instructor/Adviser: Engr. Godofredo T. Avena **Date Submitted:** July 05, 2025

Table of Contents

I. Introduction

The idea of calling for a ride is nothing new. It dates back to the 1600s, when horse-drawn carriages served as the world's first taxi. However, as time passed, people began to create motorized taxis in the 19th century, which started to replace horse-drawn carriages. The iconic yellow taxis became popular in New York City in the early 20th century, shaping the modern taxis that we know of today.

The evolution of mobile technology led to the rise of ride-hailing apps, revolutionizing how people book transportation. In 2009, Uber was founded in San Francisco, initially offering luxury car services before expanding to standard rides. Companies like Lyft, Grab, and DiDi emerged after Uber's creation, allowing commuters to choose from various apps that provide them with the most convenience. Today, ride-hailing apps operate globally, offering ease in daily commutes and disrupting traditional taxi industries.

Soon enough, ride-hailing apps started to penetrate Philippine transportation as it was spreading globally. Grab was the first to make an entry in the country's vehicle services in 2013, and was soon followed by Uber in 2014. Both services were eventually recognized by the Philippine government, through the DOTC, and received regulations for Transport Network Vehicle Service (TNVS) in the year 2015. Currently, there have been several vehicle services that were created locally to provide convenience to the masses.

The sole purpose of ride-hailing apps is to provide convenient and on-demand transportation by connecting users and drivers through an app. In today's fast-paced world, getting from one place to another safely and hassle-free is a top priority. That is why we developed **Gethub** — a ride-hailing app designed to provide faster, safer, and more accessible transportation for everyone. This project aims not only to address everyday commuting needs but also to serve as a practical application of all the concepts of Object Oriented Programming. Through **Gethub**, we intend to demonstrate how OOP concepts can be effectively used to build a simple yet functional ride-booking app.

II. Objectives

The general objective of this project is to demonstrate the application of object-oriented programming by developing a functional and straightforward ride-booking system app.

Specifically, the goal of the project is:

1. To apply object-oriented programming concepts, including classes, inheritance, and polymorphism.

2. To learn and implement file handling in Python for saving and retrieving application data.

3. To design and develop a graphical user interface using Tkinter.

4. To understand and apply fundamental software engineering principles, such as modularity and encapsulation, in a practical application.

III. Scope and Limitations

The Gethub Team focuses on the core functionalities of a ride-booking application, though it operates with several notable constraints that define its current operational boundaries.

Scope:

1. Booking a ride:

- Routing calculation from a pickup to a dropoff point that can be manually placed and removed by the user on the map

- Location name entry option with basic auto-suggest feature as an alternative for manual marker placement

- Vehicle selection options such as a car, a van, and a motorcycle

- Importing booking information from a saved booking file

• Clearing booking information

• Distance, estimated time of arrival, and cost calculation based on the generated route and the selected vehicle

• Confirming and cancelling a booking

2. Saving a ride:

• Database of confirmed and cancelled rides with booking information details displayed.

• Saving (downloading) the selected ride as a file that can be used on the next booking for a much easier booking.

3. User experience:

• Account creation, log in, and log out

• Account customization

• Account-specific ride database

• App configuration

• About us and contact us page for developer-user interaction

Limitations:

1. Limited Error Handling for External APIs:

• Limited auto-suggest search results

• Inefficient routing path calculation

• Slow calculations and importation of location information and paths

2. Real-time Driver Tracking: The system does not include real-time GPS tracking of drivers or vehicles.

3. Payment Integration: No actual payment gateway integration is implemented; fare calculation is for estimation purposes only.

4. Driver-Side Application: No interaction between rider and user. The system is purely user-facing; there is no separate application or interface for drivers to accept or manage rides.

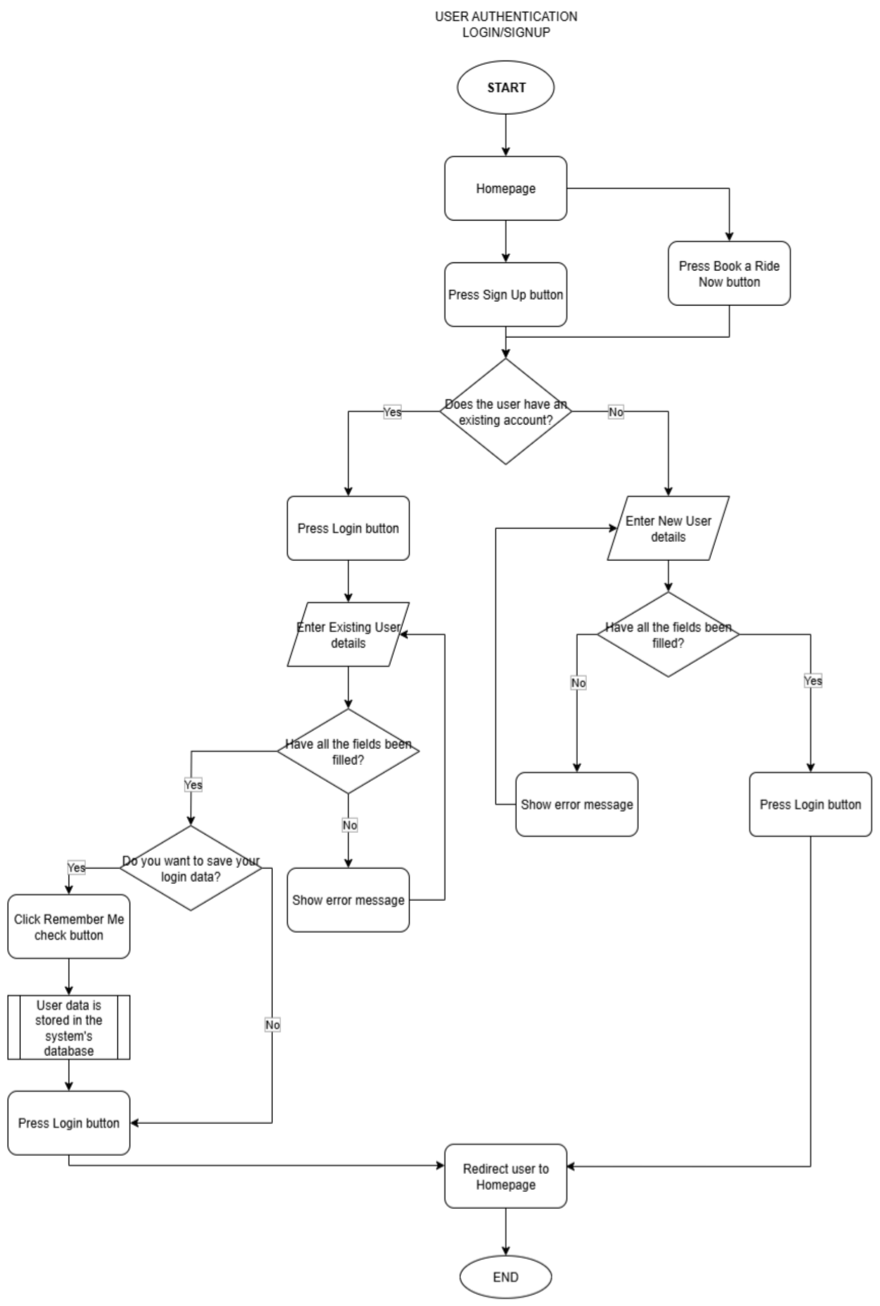
IV. Methodology

To maintain a smooth process in project creation, this project followed the **Software Development Life Cycle (SDLC)** using the **Agile** methodology as the development approach. The Agile framework is an iterative methodology — a flexible, trial-and-error approach to building better products and projects (Martins, 2025). This framework allowed the team to easily identify which features needed changes and improvements, and effectively prioritize tasks based on the most critical requirements and user needs. In addition, the project utilized a Component-Based Architecture, which breaks down the system into smaller, reusable, independent, and manageable components. With the use of this approach, the team was able to focus on coding, testing, and improving their respective parts individually and efficiently while maintaining the overall stability of the system.

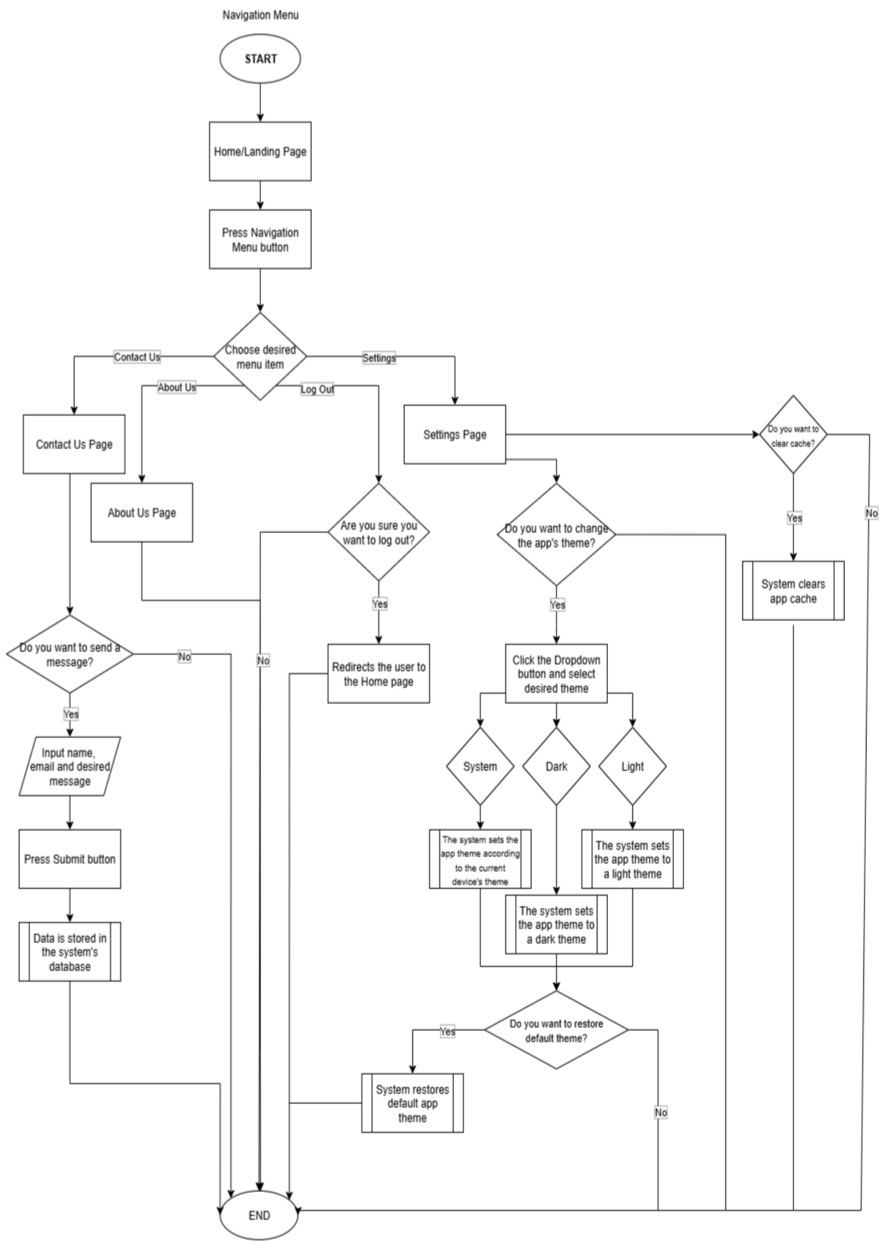
V. System Design

USER EXPERIENCE FLOWCHARTS

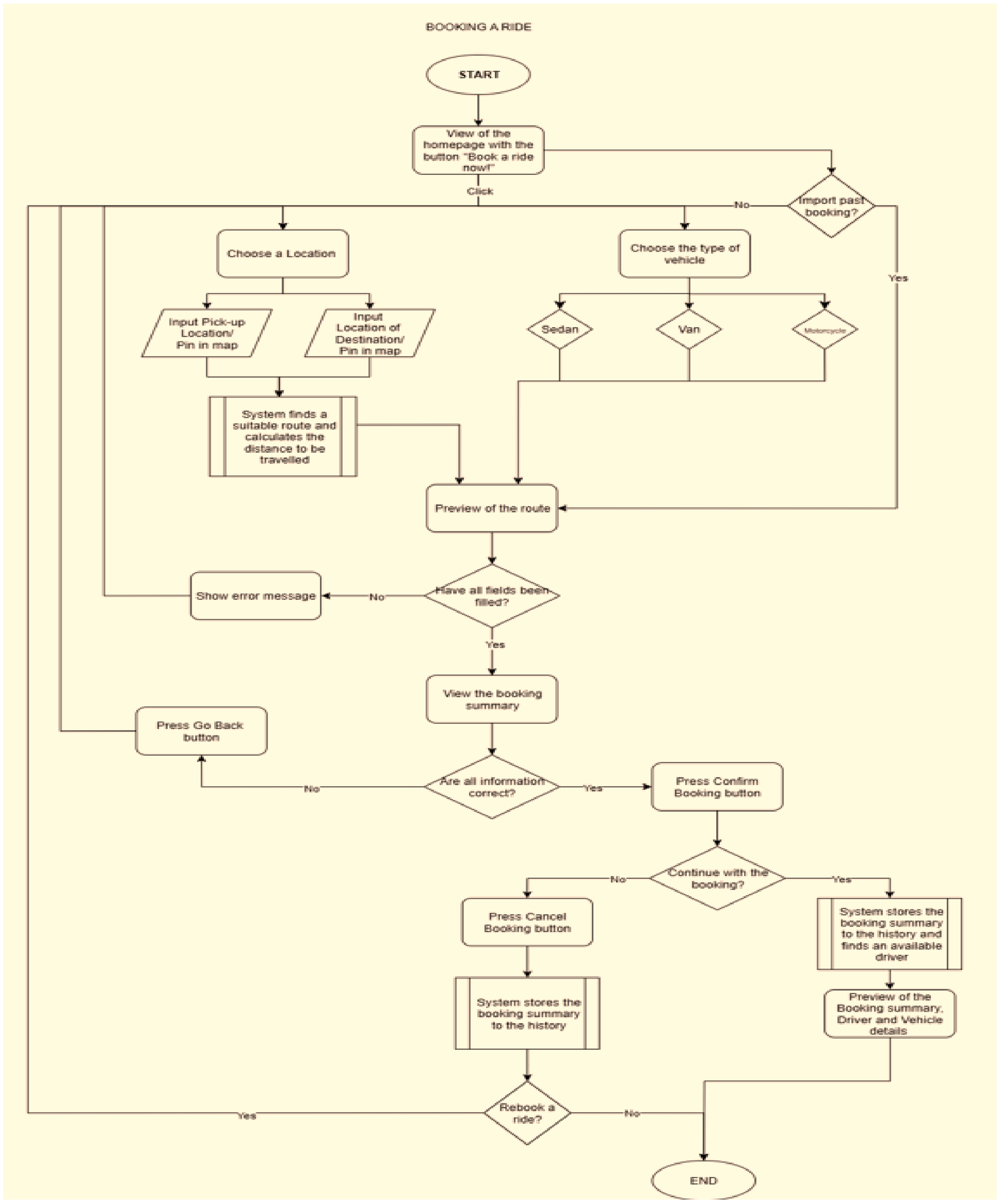
1. User Authentication: Log in/Sign up


USER AUTHENTICATION LOGIN/SIGNUP
START
Homepage
Press Sign Up button
Press Book a Ride Now button
Does the user have an existing account?
Yes
No
Press Login button
Enter Existing User details
Have all the fields been filled?
Yes
No
Do you want to save your login data?
Yes
No
Click Remember Me check button
User data is stored in the system's database
Press Login button
Show error message
Enter New User details
Have all the fields been filled?
No
Yes
Show error message
Press Login button
Redirect user to Homepage
END

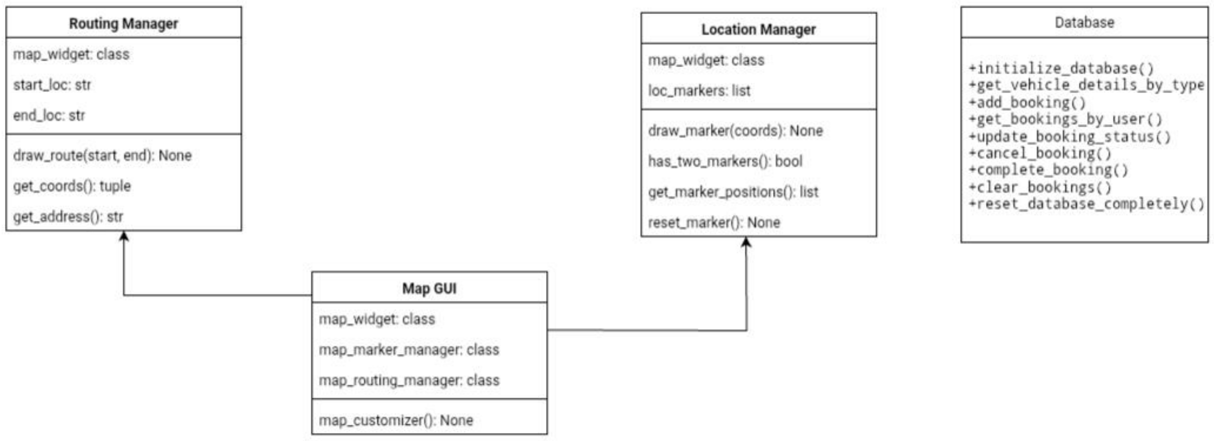
2. Menus Navigation

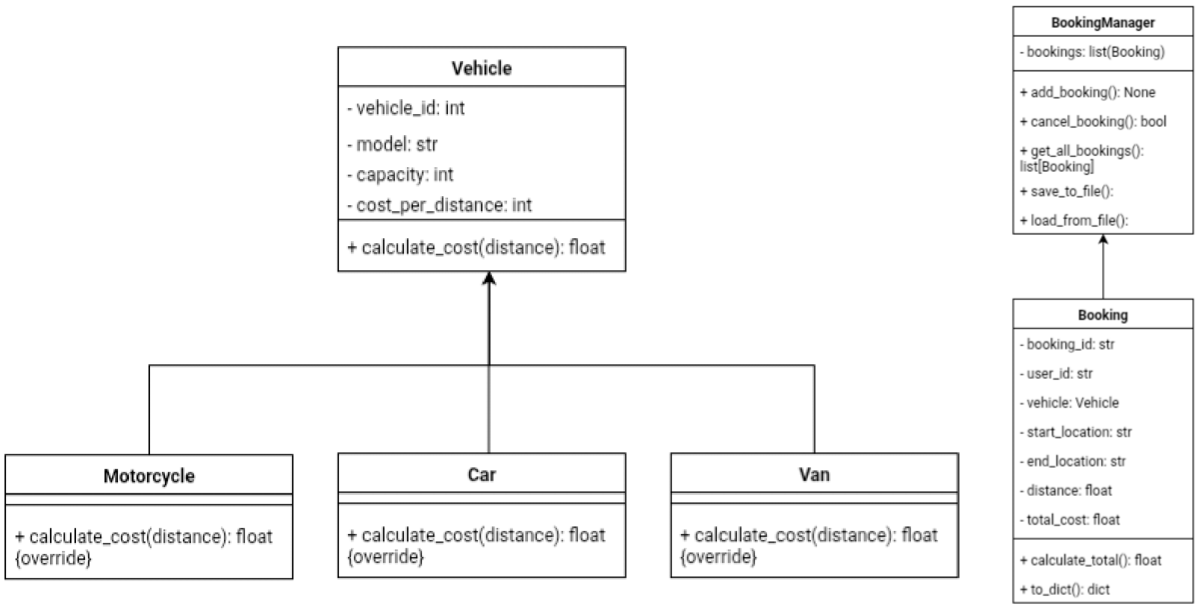

Navigation Menu
START
Home/Landing Page
Press Navigation Menu button
Choose desired menu item
Contact Us
About Us
Log Out
Settings
Contact Us Page
About Us Page
Are you sure you want to log out?
No
Yes
Redirects the user to the Home page
Settings Page
Do you want to clear cache?
Yes
No
System clears app cache
Do you want to change the app's theme?
Yes
Click the Dropdown button and select desired theme
System
Dark
Light
The system sets the app theme according to the current device's theme
The system sets the app theme to a dark theme
The system sets the app theme to a light theme
Do you want to restore default theme?
Yes
No
System restores default app theme
Do you want to send a message?
No
Yes
Input name, email and desired message
Press Submit button
Data is stored in the system's database
END

3. Ride Booking


BOOKING A RIDE
START
View of the homepage with the button "Book a ride now!"
Click
Import past booking?
No
Yes
Choose a Location
Choose the type of vehicle
Input Pick-up Location/ Pin in map
Input Location of Destination/ Pin in map
Sedan
Van
Motorcycle
System finds a suitable route and calculates the distance to be travelled
Preview of the route
Have all fields been filled?
No
Show error message
Yes
View the booking summary
Press Go Back button
Are all information correct?
No
Yes
Press Confirm Booking button
Continue with the booking?
No
Yes
Press Cancel Booking button
System stores the booking summary to the history and finds an available driver
System stores the booking summary to the history
Preview of the Booking summary, Driver and Vehicle details
Rebook a ride?
Yes
No
END

UNIFIED MODELLING LANGUAGE (UML) DIAGRAMS





VI. Technologies Used

The programming languages, frameworks, libraries, tools, and platforms used in the project are the following:

Programming Language: Python 3.X

Libraries/APIs: CustomTkinter – GUI

Pillow - Images

Tkintermapview – Map GUI

OpenStreetMap – Map Information

Open-Source Routing Machine – Route Calculation

Geopy – Geocoding

Geoapify – Location Addresses

SQLite – Database

Visual Studio Code - IDE

Google Docs – Documentation

Figma – Wireframes and Logos

Draw.io – Flowcharts

Git/Github – Version Control and Collaboration

Discord – Communication

Platforms:

VII. Implementation

Features Developed

1. Interactive Map Interface - Using TkinterMapView integrated into a CustomTkinter GUI, users can select pick-up and drop-off locations by clicking directly on the map. This enhances usability by visual location selection rather than manual address entry.

2. Route Display with OSRM - Once locations are selected, the system queries the OSRM API (using a Python wrapper around OSRM) to calculate and display the fastest route on the map. The route geometry is drawn interactively, showing the path between the two points.

3. Distance, Time, and Fare Calculation - Using OSRM's routing data, the system calculates the distance (in kilometers) and estimated travel time (in minutes). Based on these and the selected vehicle type, the system computes the fare in local currency (P) using predefined rate schemes.

4. Multiple Vehicle Types Selection - Users can choose from various vehicle categories (e.g., sedan, SUV, van), each with different fare rates per kilometer. This selection dynamically influences the fare calculation.

5. Ride Booking Management - Users can book a ride, which is then stored in the SQLite database. The system enables users to view existing bookings and cancel rides as needed, providing a straightforward booking management interface.

Implementation Details

1. Map Interaction: Tkintermapview handles map rendering and mouse click events to capture coordinates for pick-up and drop-off. These coordinates are passed to OSRM for routing.

2. Routing with OSRM: The Python OSRM wrapper sends requests to the OSRM server (local or public). It fetches route details including distance, duration, and step-by-step geometry. The route is decoded and drawn on the map widget.

3. Distance & Time Extraction: The OSRM response includes distance (in meters) and duration (in seconds). These are converted to kilometers and minutes, respectively, for display and fare calculation.

4. Fare Calculation Logic: Fare = Base fare + (Distance in km × Rate per km based on vehicle type). Rates are configurable and stored in the database or in code.

5. Database Management with SQLite: SQLite stores user bookings with fields such as booking ID, pick-up/drop-off coordinates, vehicle type, fare, and booking status. CRUD operations allow for managing bookings.

6. GUI Components: CustomTkinter forms and buttons facilitate vehicle selection, booking confirmation, viewing bookings, and cancellation. The interface updates dynamically with map and fare info.

VIII. Testing and Evaluation

Our app development process incorporates manual testing to ensure functionality, reliability, and a positive user experience.

Test Case Summary

Module	Test Case	Input/Scenario	Expected Result	Actual Result	Status
Authentication	Invalid password submission	User enters wrong password	Error message displayed	"Invalid credentials" shown	Pass
	Admin login verification	Admin username + correct password	Redirect to admin dashboard	Successful redirect	Pass
Booking	Vehicle selection → Fare calculation	Van selected for 10km distance	Calculated fare: ₱245.00	₱245.00 shown	Pass
	Empty location fields validation	No pickup/dropoff entered	Booking disabled	Button remains inactive	Pass
History	JSON export functionality	Click "Save" on completed booking	JSON file with booking data	File downloaded successfully	Pass
	Pagination (20+ bookings)	Scroll through history page	Smooth loading, no crashes	No rendering issues	Pass

IX. Results and Discussion

The final product, the Gethub Ride Booking System, successfully met its primary objectives, delivering a functional and user-friendly application for ride booking.

1. Interactive Map and Route Display: The integration of OpenStreetMap and OSRM allowed for seamless selection of pickup and drop-off locations, dynamic route display, and accurate calculation of distance and estimated time. This core functionality was a key objective and was fully realized.

2. Fare Calculation and Vehicle Selection: The system accurately calculates fares based on distance and selected vehicle type (Car, Van, Motorcycle), demonstrating the successful implementation of the pricing model and vehicle management.

3. Booking Management: Users can successfully book rides, view their booking history, and cancel active bookings, fulfilling the objective of comprehensive booking management.

4. User Authentication and Profile Management: The robust authentication system (signup/login) and profile management features (updating personal details, changing password, managing profile picture) were implemented as intended, providing a secure and personalized user experience.

5. User Interface and Experience: The CustomTkinter framework enabled the creation of a modern and intuitive graphical user interface, enhancing the overall user experience. Features like autosuggest for addresses and clear visual feedback on the map contributed significantly to usability.

Challenges Faced During Development

The development process, while successful for us, faced several challenges for the team.

1. API Integration Complexity: Integrating external APIs like Geoapify for autosuggest and OSRM for routing required careful handling of asynchronous requests, error management (e.g., connection timeouts), and parsing of complex JSON responses.

Ensuring smooth and reliable data flow between the application and these services was a significant hurdle.

2. Database Management and Schema Design: Designing an efficient and scalable SQLite database schema for users, vehicles, and bookings, along with implementing robust CRUD operations, demanded meticulous planning and execution. Ensuring data integrity and handling relationships between tables (e.g., user_id and vehicle_id in bookings) was crucial.

3. Real-time Updates and State Management: Managing the application's state across different components, especially concerning map markers, route lines, and booking information, proved challenging. The BookingInformationManager was developed to centralize this, but ensuring all UI elements reflected the correct state in real-time required careful synchronization.

4. User Experience Refinements: Iteratively refining the user interface and experience, particularly for features like password visibility toggling, error highlighting in forms, and the profile picture cropping functionality, required continuous testing and adjustments to achieve a polished feel.

5. Team Collaboration and Version Control: Coordinating development efforts among multiple team members using Git required consistent communication, adherence to branching strategies, and effective conflict resolution to maintain a stable codebase.

X. Conclusion

Throughout this whole project creation, we gained many valuable insights not only in applying the technical concepts of OOP but also in understanding the importance of effective teamwork and careful planning. Before the project creation, we were able to study the Python language, basic concepts of OOP, and the basics of using Tkinter in creating GUIs. Throughout the project creation, we gradually learned how to apply the knowledge we gained from our lectures on Object Oriented Programming (OOP) and Tkinter. Moreover, this project allowed us to fully grasp the concepts of OOP, different functions of GitHub, how to fix and handle conflicts, database handling, advanced coding, file organization, and the importance of understanding the documentation of the modules that are being imported in our program.

The entire process and outcome of the project creation left a significant impact on our experience as Computer Engineering students. It allowed us to experiment on our newly obtained skills, explore how we can implement the ideas that we have, and assess whether our current plans are doable or not. Additionally, it expanded our experience in coding and programming, designing, and software engineering principles.

Thanks to the continuous communication and teamwork of our team, we were able to successfully develop a simple yet functional ride-booking system app. The regular meetings and constant updates of changes made it easier for us to quickly address the challenges and make necessary adjustments along the way. This project allowed us to not only enhance our skills but also strengthen our ability to work collaboratively, manage time effectively, and solve problems as a team. Without the collaboration and proper planning, completing this project on time would not have been possible.

XI. Future Work

The Gethub team plans to implement additional features and improvements for future versions such as:

1. Add payment gateway (PayMaya API).
2. Multi-language support.
3. Vehicle and User Connection and Interactions
4. Implementation of AI on Gethub application

References

Bernstein, B. (2022, December 12). A brief history of ride sharing - Brett Bernstein - medium. *Medium*. https://medium.com/@bhbern/a-brief-history-of-ride-sharing-7d1eca9e4654

Bellis, M. (2025, May 12). *Hailing: History of the Taxi*. ThoughtCo. https://www.thoughtco.com/hailing-history-of-the-taxi-1992541

Clamosa, C. A. (n.d.). *TNC-and-TNVS*. Scribd. https://www.scribd.com/document/518152074/TNC-and-TNVS

Wikipedia contributors. (2025, June 17). *Ridesharing company*. Wikipedia. https://en.wikipedia.org/wiki/Ridesharing_company

Laoyan, S. (2025, February 20). What is Agile Methodology? (A Beginner's Guide) [2025] • Asana. *Asana*. https://asana.com/resources/agile-methodology

TomSchimansky. (n.d.). *GitHub - TomSchimansky/TkinterMapView: A python Tkinter widget to display tile based maps like OpenStreetMap or Google Satellite Images.* GitHub. https://github.com/TomSchimansky/TkinterMapView

Official documentation and tutorial | CustomTkinter. (n.d.). https://customtkinter.tomschimansky.com/

OSRM API Documentation. (n.d.). https://project-osrm.org/docs/v5.24.0/api/?language=Python#

GeoApify Tutorials. (n.d.). Geoapify. https://www.geoapify.com/tutorials/

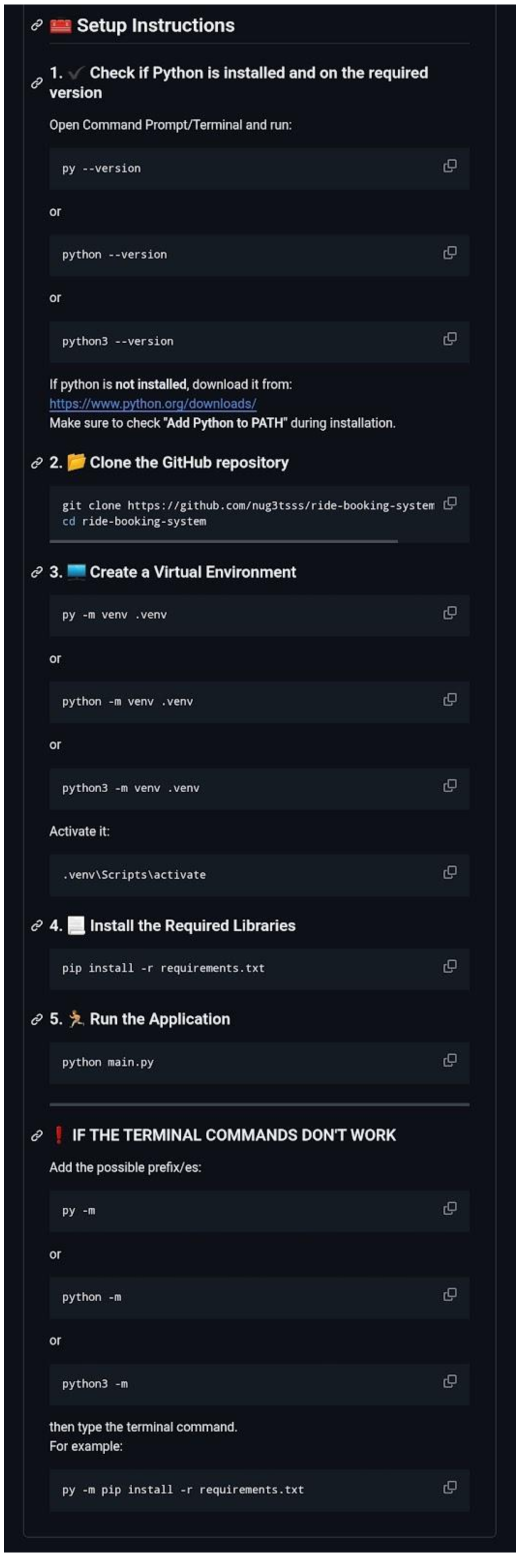
Welcome to GeoPy's documentation! — GeoPy 2.4.1 documentation. (n.d.). https://geopy.readthedocs.io/en/stable/

TkDocs Home. (n.d.). https://tkdocs.com/

Pillow. (n.d.). Pillow (PIL Fork). https://pillow.readthedocs.io/en/stable/

Appendices

Repository installation and setup



🧰 Setup Instructions

1. ✔️ Check if Python is installed and on the required version

Open Command Prompt/Terminal and run:

```
py --version
```

or

```
python --version
```

or

```
python3 --version
```

If python is **not installed**, download it from:
https://www.python.org/downloads/
Make sure to check **"Add Python to PATH"** during installation.

2. 📁 Clone the GitHub repository

```
git clone https://github.com/nug3tsss/ride-booking-system
cd ride-booking-system
```

3. 🖥️ Create a Virtual Environment

```
py -m venv .venv
```

or

```
python -m venv .venv
```

or

```
python3 -m venv .venv
```

Activate it:

```
.venv\Scripts\activate
```

4. 📃 Install the Required Libraries

```
pip install -r requirements.txt
```

5. 🏃 Run the Application

```
python main.py
```

❗ IF THE TERMINAL COMMANDS DON'T WORK

Add the possible prefix/es:

```
py -m
```

or

```
python -m
```

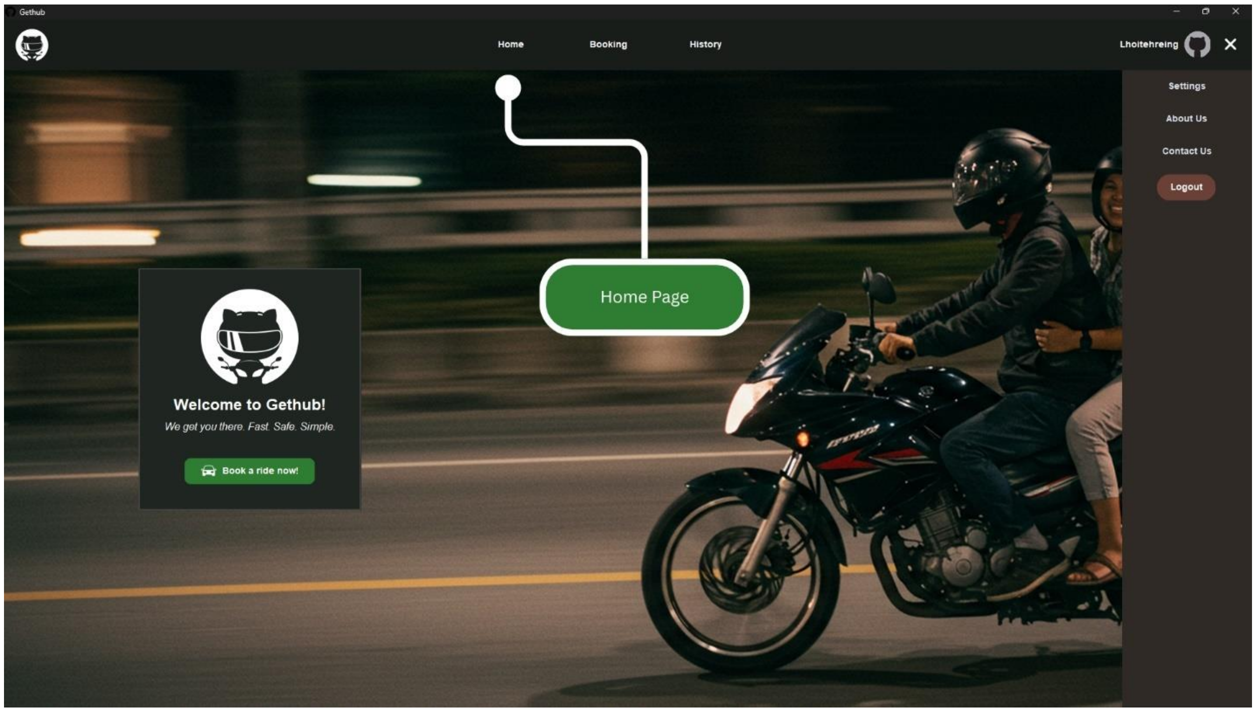
or

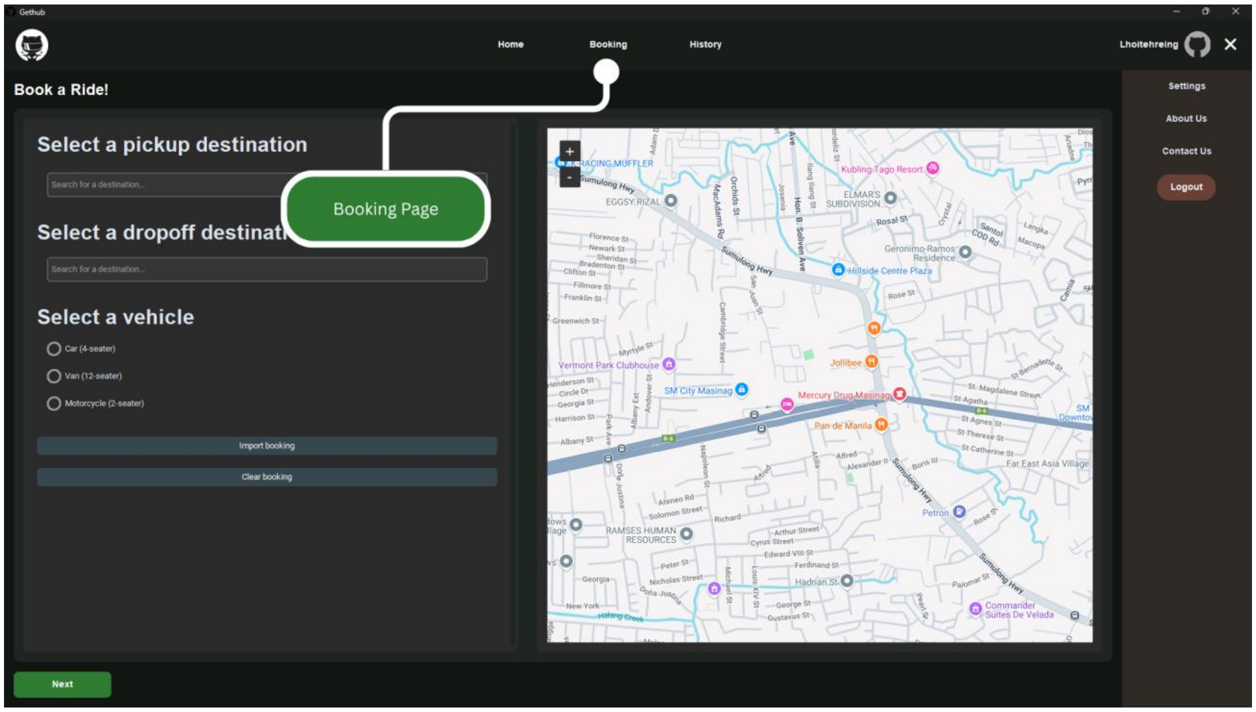
```
python3 -m
```

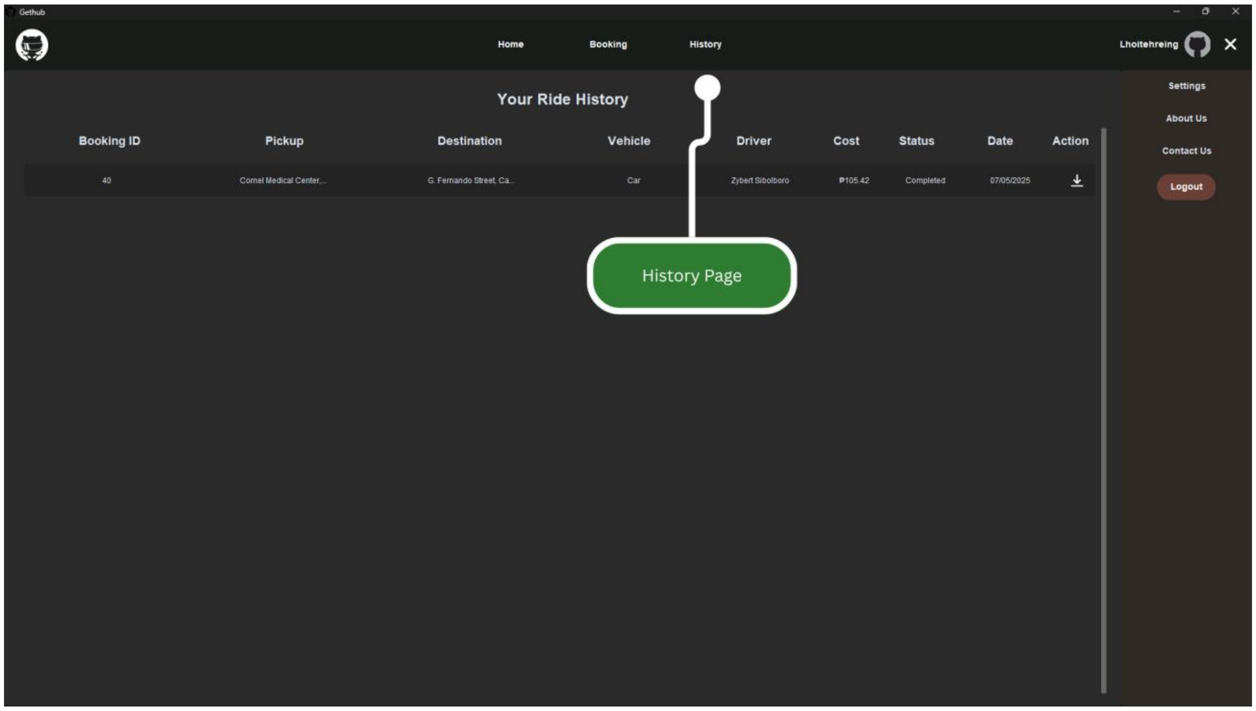
then type the terminal command.
For example:

```
py -m pip install -r requirements.txt
```

App menus guide

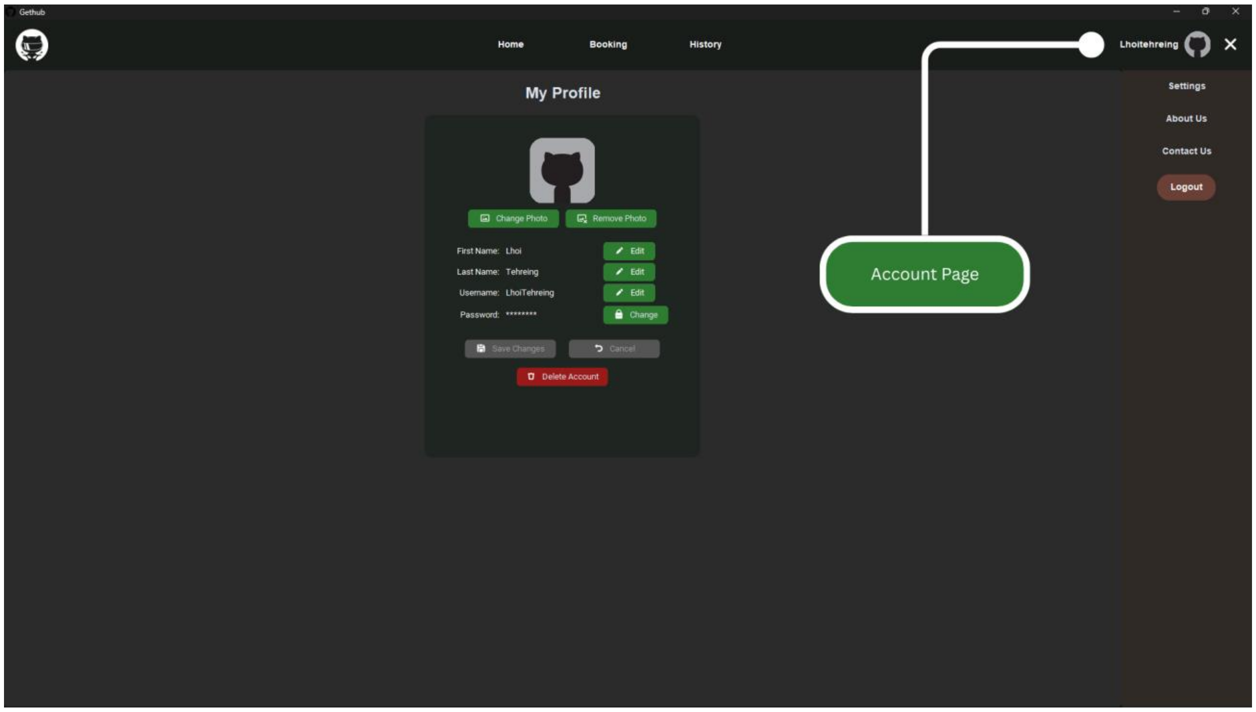






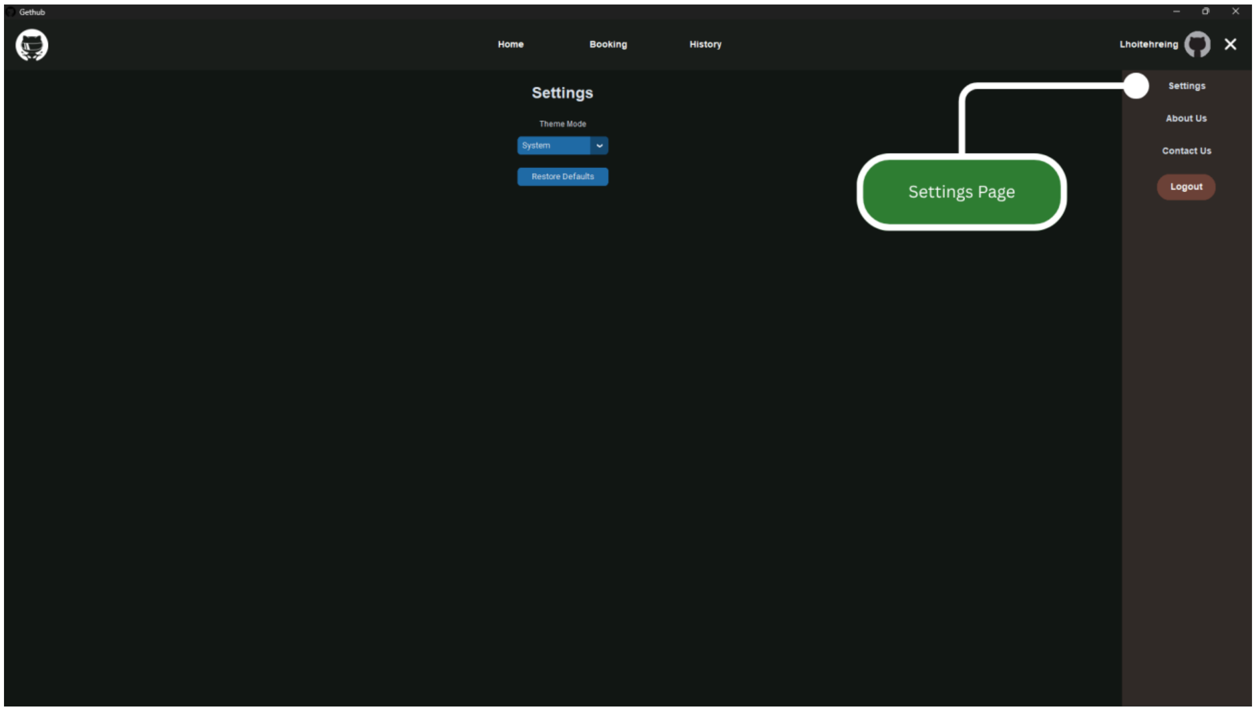
Your Ride History

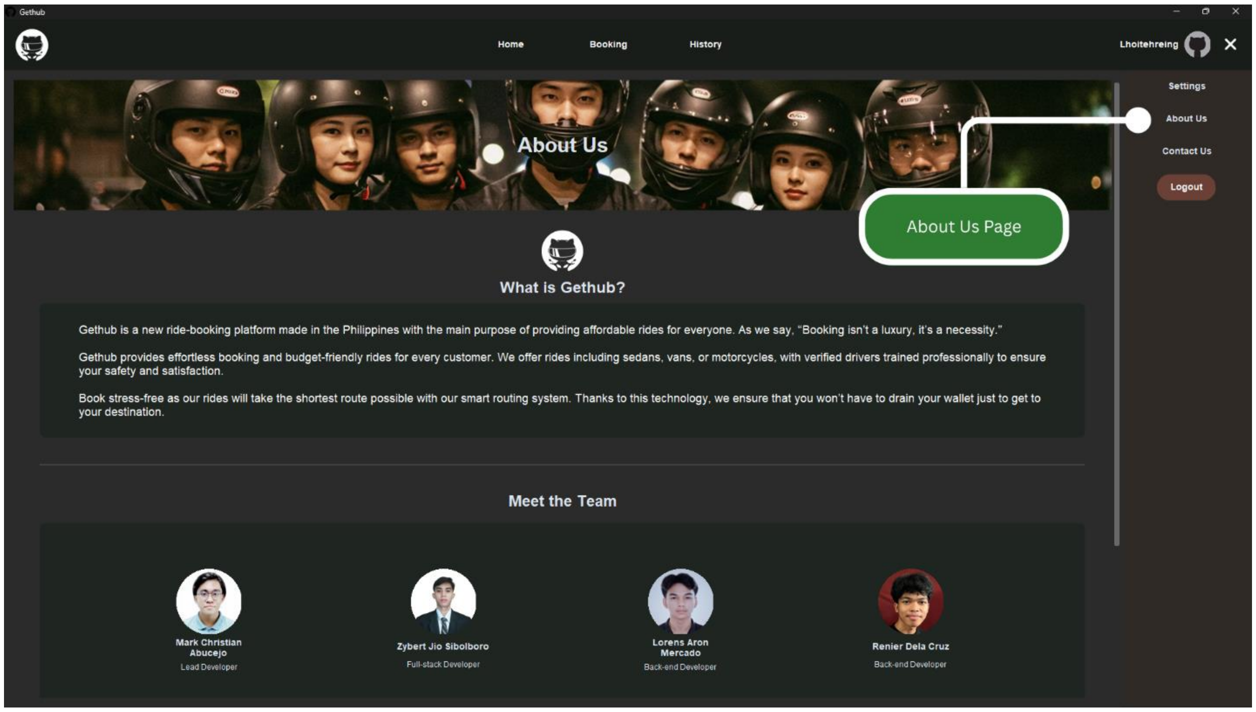
Booking ID	Pickup	Destination	Vehicle	Driver	Cost	Status	Date	Action
40	Cornel Medical Center,...	G. Fernando Street, Ca...	Car	Zybert Sibolboro	₱105.42	Completed	07/05/2025	⤓



My Profile

First Name: Lhoi
Last Name: Tehreing
Username: LhoiTehreing
Password: ********





What is Gethub?

Gethub is a new ride-booking platform made in the Philippines with the main purpose of providing affordable rides for everyone. As we say, "Booking isn't a luxury, it's a necessity."

Gethub provides effortless booking and budget-friendly rides for every customer. We offer rides including sedans, vans, or motorcycles, with verified drivers trained professionally to ensure your safety and satisfaction.

Book stress-free as our rides will take the shortest route possible with our smart routing system. Thanks to this technology, we ensure that you won't have to drain your wallet just to get to your destination.

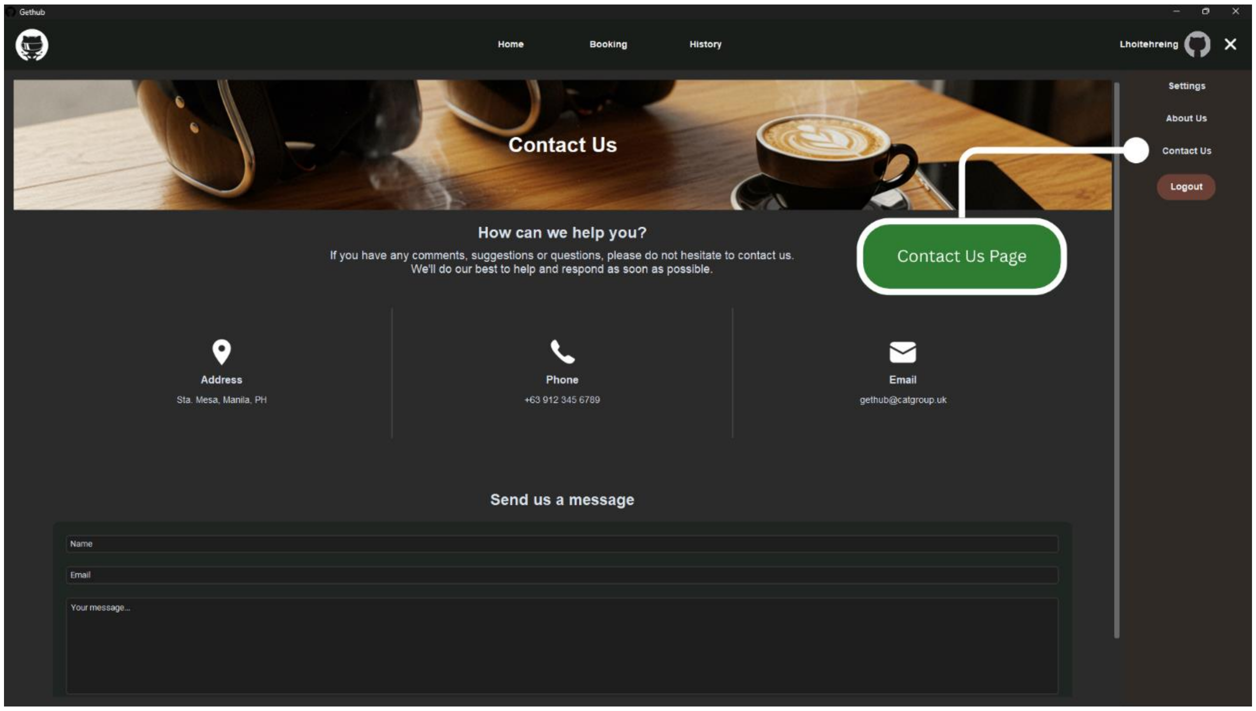
Meet the Team

Mark Christian Abucejo
Lead Developer

Zybert Jio Sibolboro
Full-stack Developer

Lorens Aron Mercado
Back-end Developer

Renier Dela Cruz
Back-end Developer



How can we help you?

If you have any comments, suggestions or questions, please do not hesitate to contact us.
We'll do our best to help and respond as soon as possible.

Address
Sta. Mesa, Manila, PH

Phone
+63 912 345 6789

Email
gethub@catgroup.uk

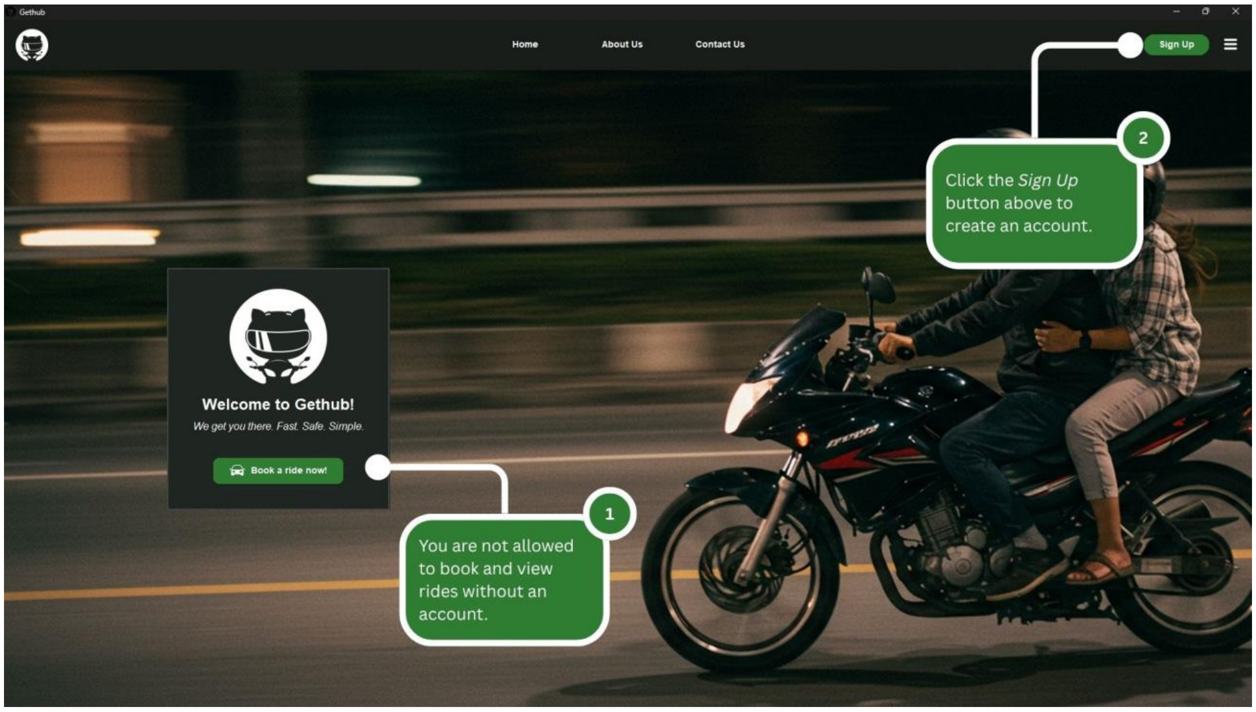
Send us a message

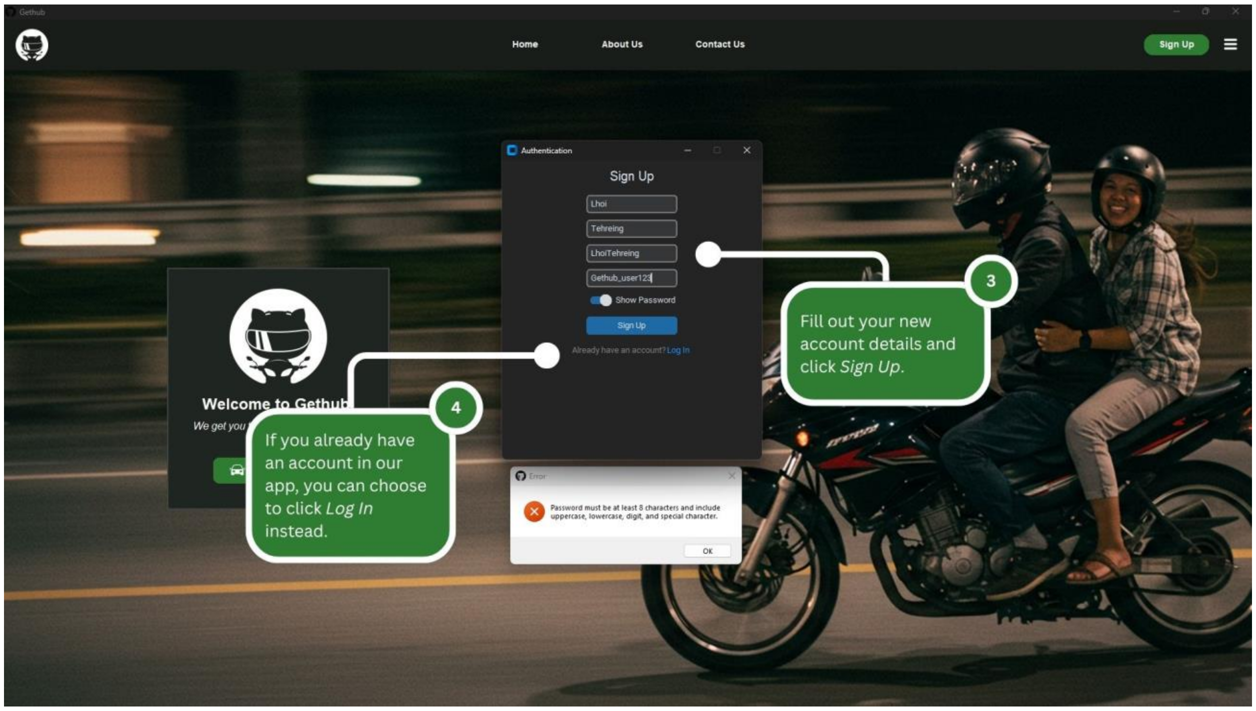
Name

Email

Your message...

New user sign-up guide


Gethub
Home
About Us
Contact Us
Sign Up
Welcome to Gethub!
We get you there. Fast. Safe. Simple.
Book a ride now!
1
You are not allowed to book and view rides without an account.
2
Click the *Sign Up* button above to create an account.



Sign Up

Lhoi

Tehreing

LhoiTehreing

Gethub_user123

Show Password

Sign Up

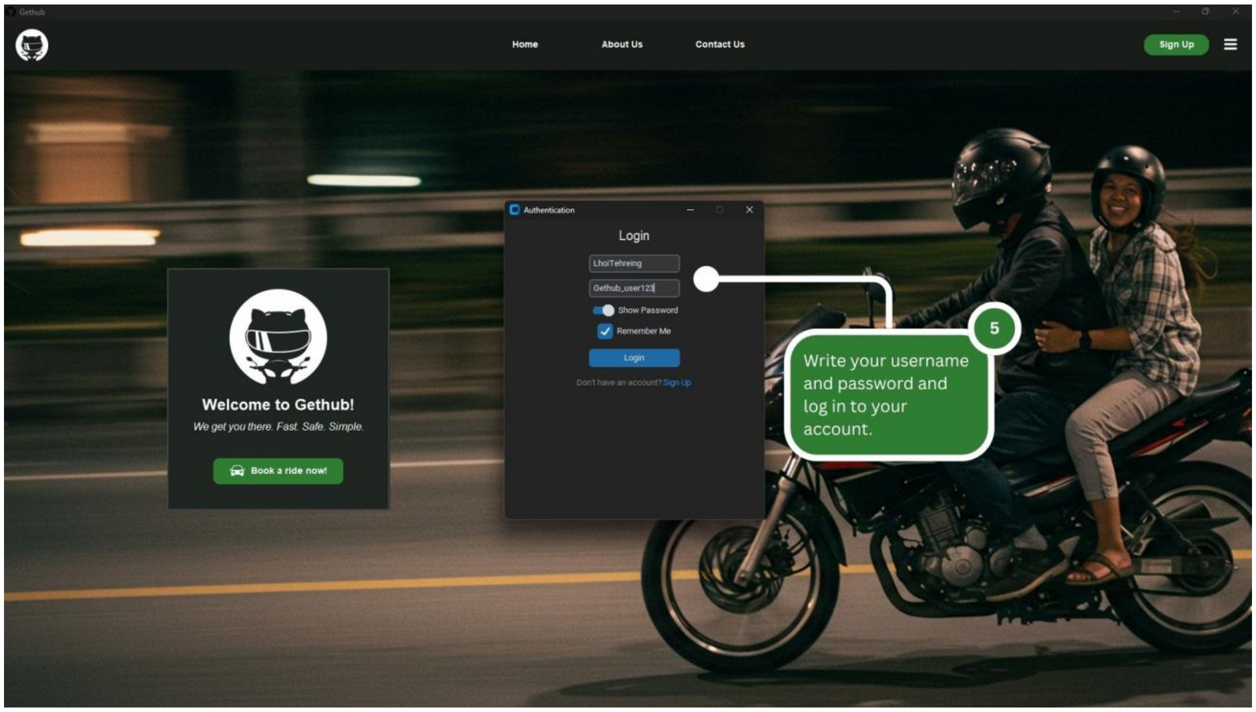
Already have an account? Log In

Error

Password must be at least 8 characters and include uppercase, lowercase, digit, and special character.

3 Fill out your new account details and click *Sign Up*.

4 If you already have an account in our app, you can choose to click *Log In* instead.



Welcome to Gethub!

We get you there. Fast. Safe. Simple.

Login

LhoiTehreing

Gethub_user123

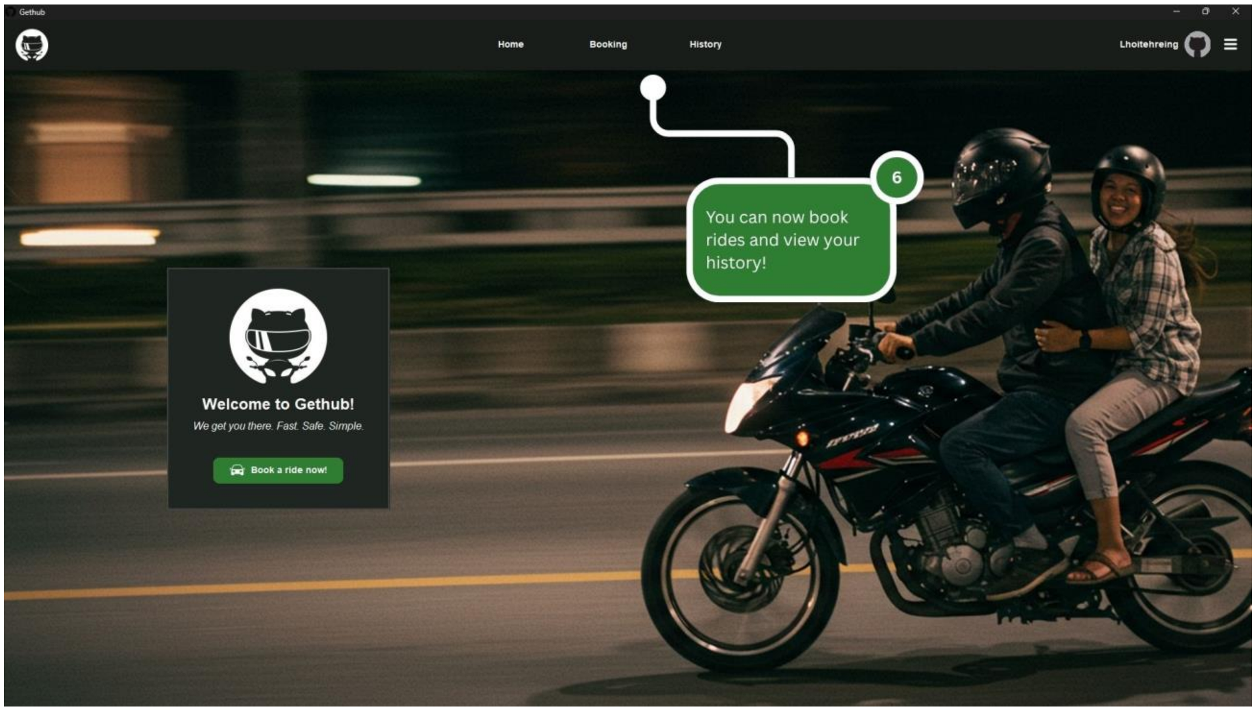
Show Password

Remember Me

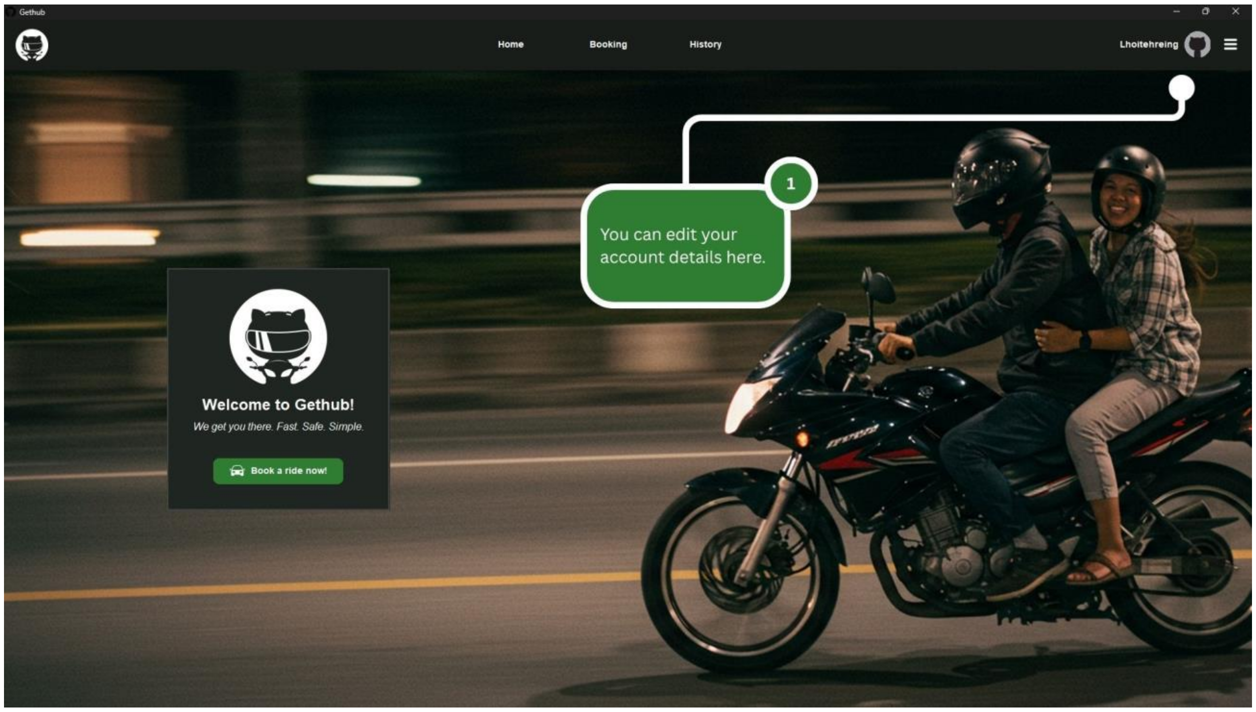
Login

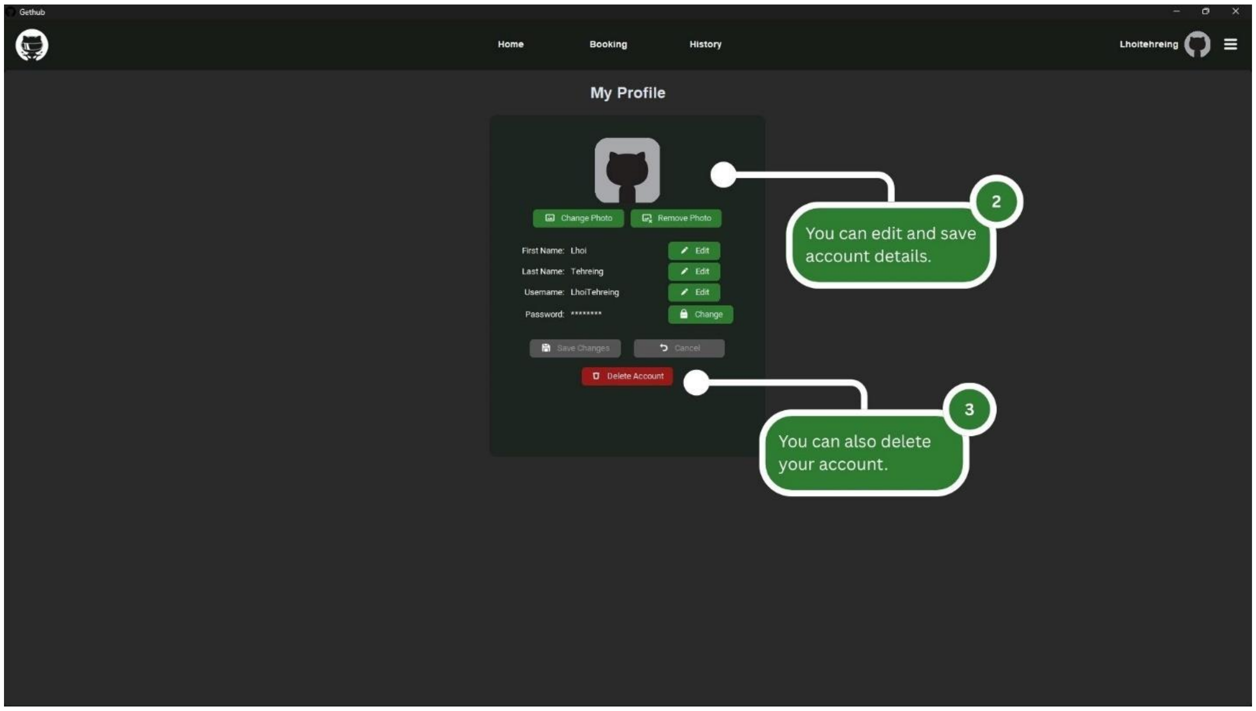
Don't have an account? Sign Up

5 Write your username and password and log in to your account.


Gethub
Home
Booking
History
Lhoitehreing
Welcome to Gethub!
We get you there. Fast. Safe. Simple.
Book a ride now!
6
You can now book rides and view your history!

Account customization guide


Gethub
Home
Booking
History
Lhoitehreing
Welcome to Gethub!
We get you there. Fast. Safe. Simple.
Book a ride now!
1
You can edit your account details here.

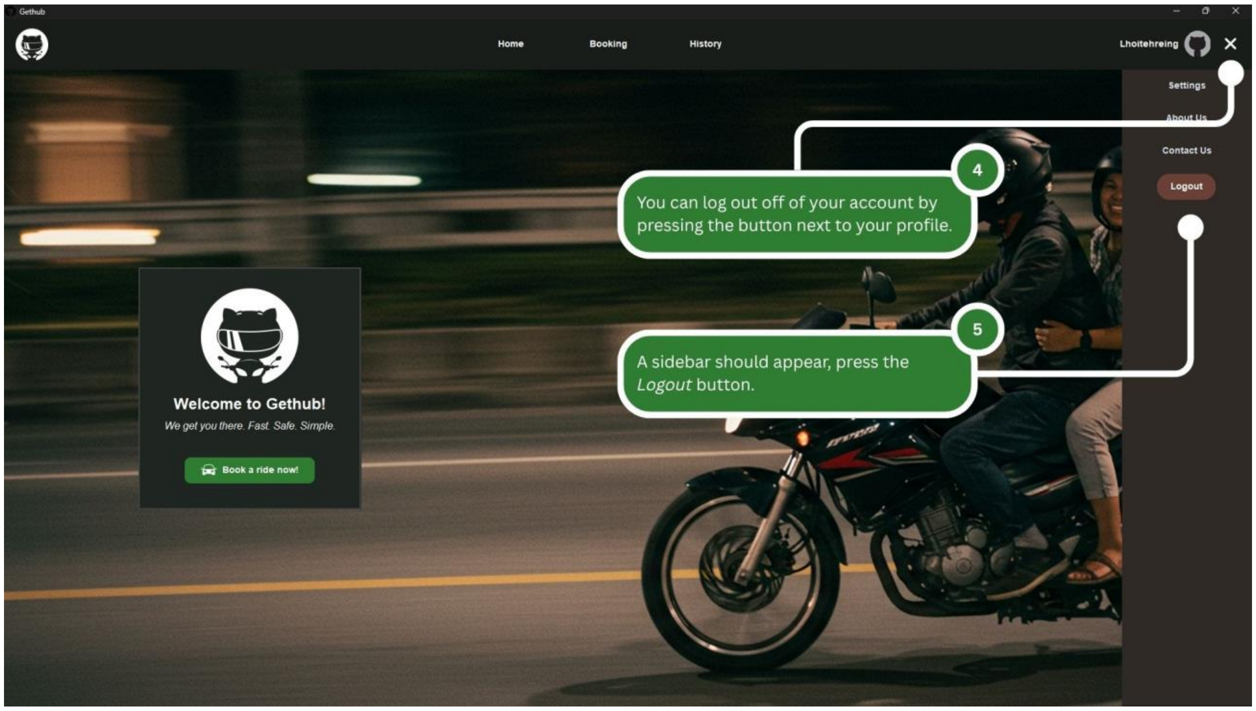


My Profile

First Name: Lhoi
Last Name: Tehreing
Username: LhoiTehreing
Password: ********

You can edit and save account details.

You can also delete your account.



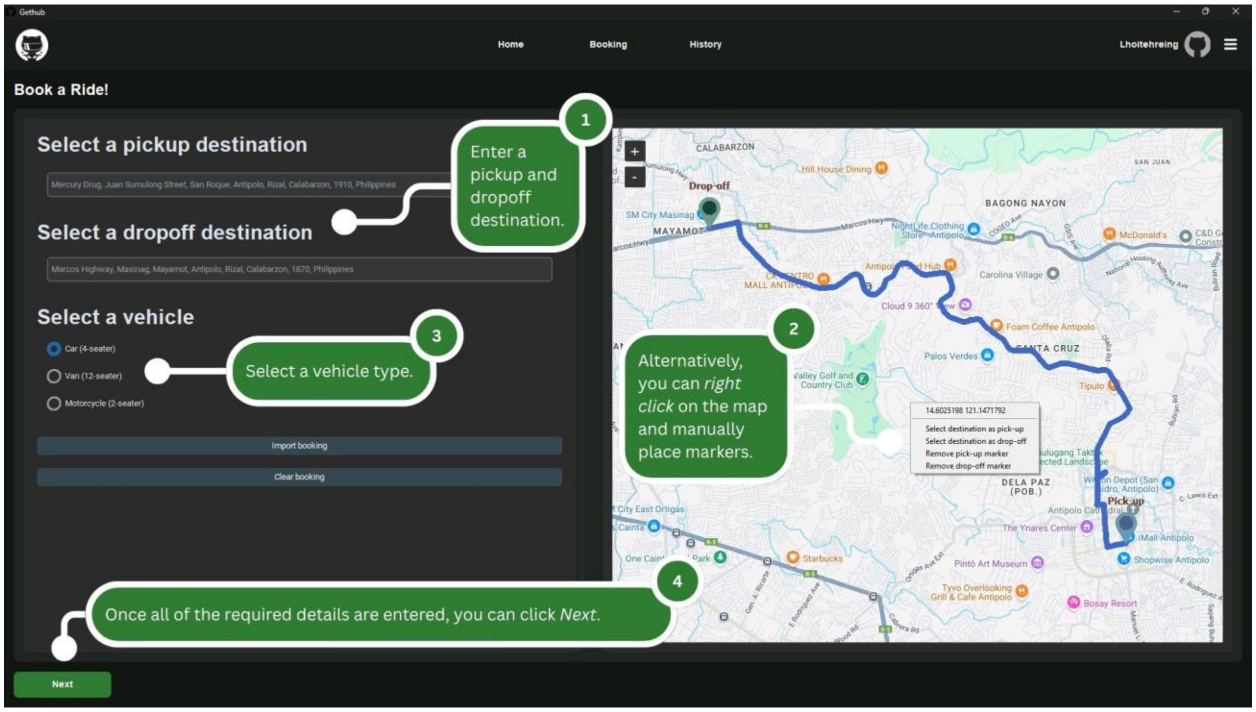
Welcome to Gethub!

We get you there. Fast. Safe. Simple.

You can log out off of your account by pressing the button next to your profile.

A sidebar should appear, press the *Logout* button.

Booking and confirming rides guide



Select a pickup destination

Mercury Drug, Juan Sumulong Street, San Roque, Antipolo, Rizal, Calabarzon, 1910, Philippines

Select a dropoff destination

Marcos Highway, Masinag, Mayamot, Antipolo, Rizal, Calabarzon, 1870, Philippines

Select a vehicle

[✓ Car (4-seater)]
Van (12-seater)
Motorcycle (2-seater)

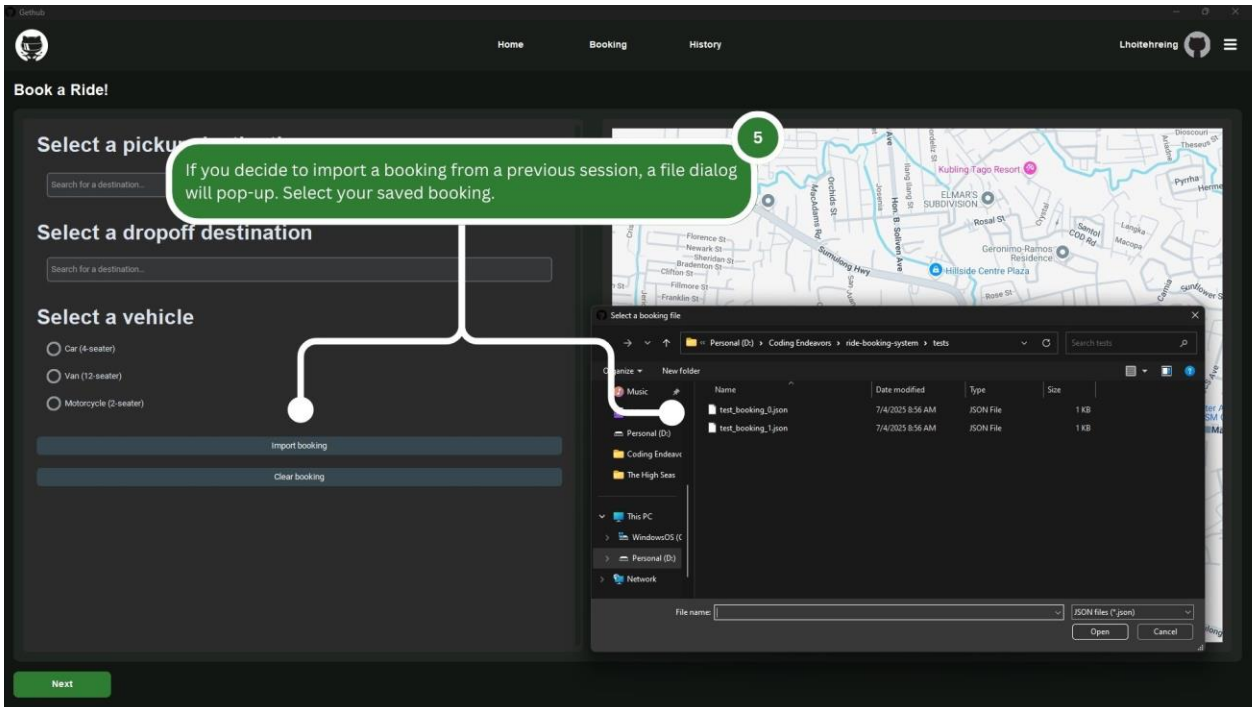
1 Enter a pickup and dropoff destination.

2 Alternatively, you can *right click* on the map and manually place markers.

14.6025198 121.1471792
Select destination as pick-up
Select destination as drop-off
Remove pick-up marker
Remove drop-off marker

3 Select a vehicle type.

4 Once all of the required details are entered, you can click *Next*.



Select a pickup destination

Search for a destination...

Select a dropoff destination

Search for a destination...

Select a vehicle

Car (4-seater)
Van (12-seater)
Motorcycle (2-seater)

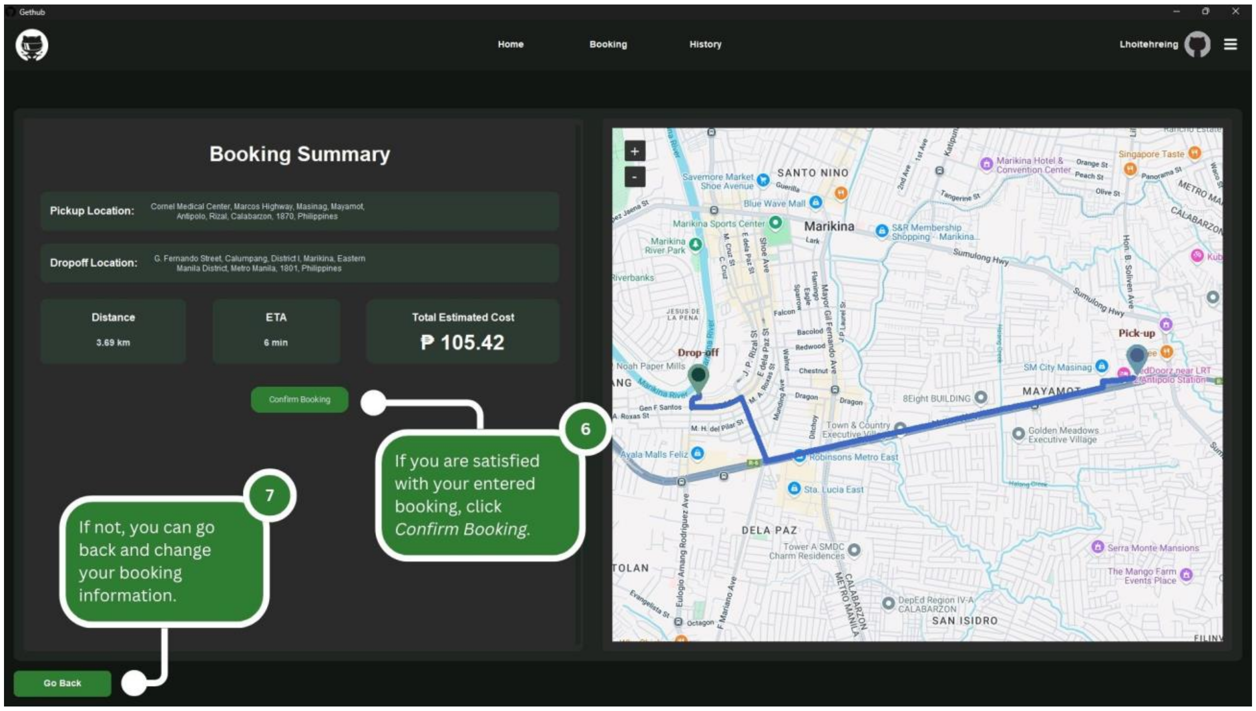
5 If you decide to import a booking from a previous session, a file dialog will pop-up. Select your saved booking.

Select a booking file

Personal (D:) › Coding Endeavors › ride-booking-system › tests

Name	Date modified	Type	Size
test_booking_0.json	7/4/2025 8:56 AM	JSON File	1 KB
test_booking_1.json	7/4/2025 8:56 AM	JSON File	1 KB

File name:
JSON files (*.json)
Open
Cancel



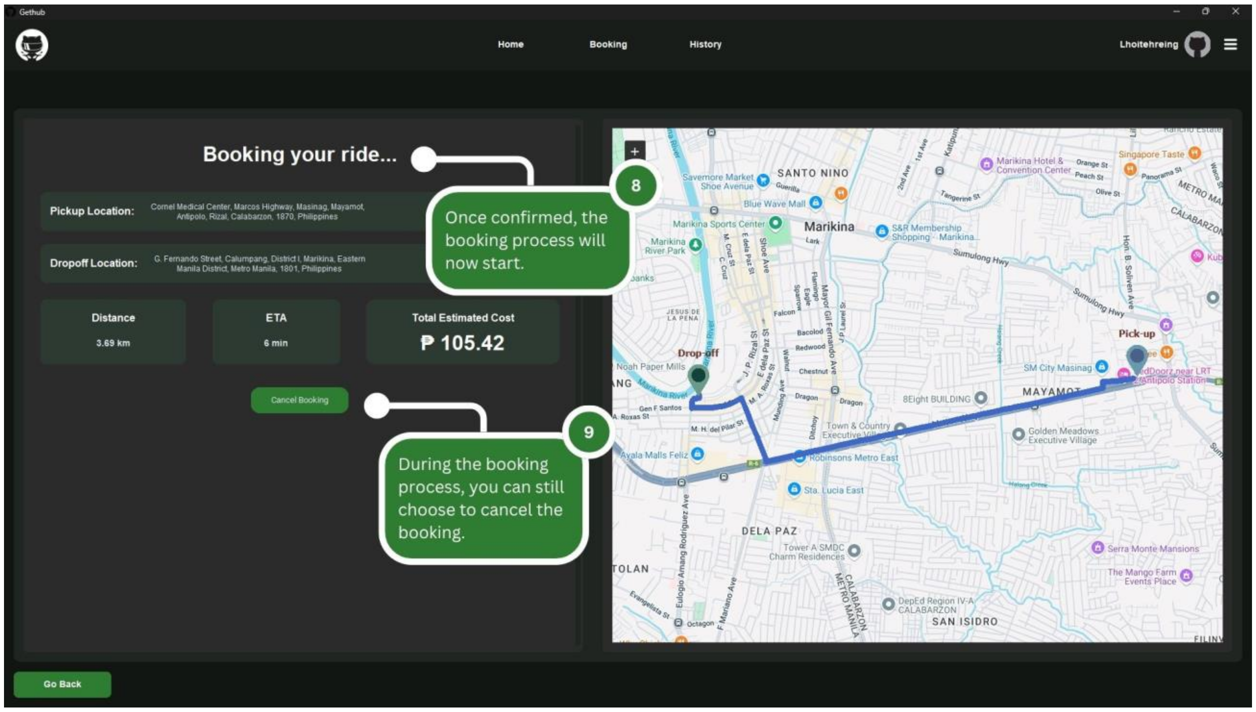
Booking Summary

Pickup Location:	Cornel Medical Center, Marcos Highway, Masinag, Mayamot, Antipolo, Rizal, Calabarzon, 1870, Philippines
Dropoff Location:	G. Fernando Street, Calumpang, District I, Marikina, Eastern Manila District, Metro Manila, 1801, Philippines

Distance	ETA	Total Estimated Cost
3.69 km	6 min	₱ 105.42

If you are satisfied with your entered booking, click *Confirm Booking.*

If not, you can go back and change your booking information.



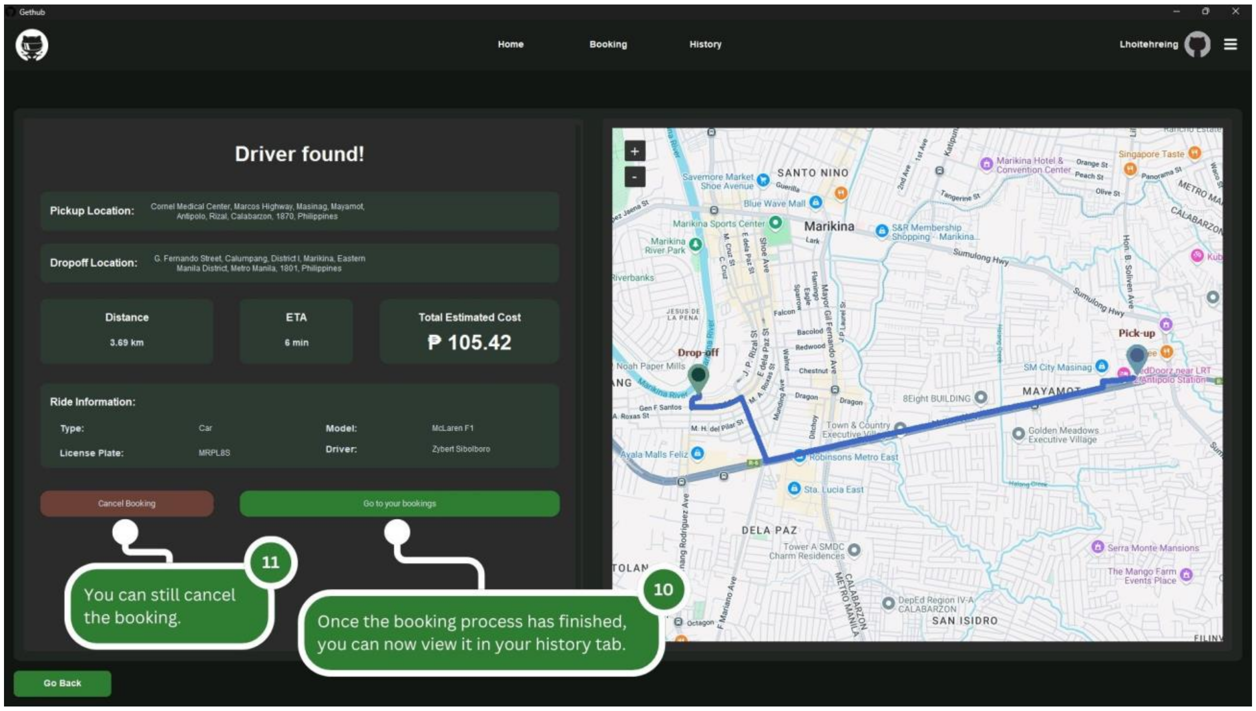
Booking your ride...

Pickup Location:	Cornel Medical Center, Marcos Highway, Masinag, Mayamot, Antipolo, Rizal, Calabarzon, 1870, Philippines
Dropoff Location:	G. Fernando Street, Calumpang, District I, Marikina, Eastern Manila District, Metro Manila, 1801, Philippines

Distance	ETA	Total Estimated Cost
3.69 km	6 min	₱ 105.42

Once confirmed, the booking process will now start.

During the booking process, you can still choose to cancel the booking.



Driver found!

Pickup Location: Cornel Medical Center, Marcos Highway, Masinag, Mayamot, Antipolo, Rizal, Calabarzon, 1870, Philippines

Dropoff Location: G. Fernando Street, Calumpang, District I, Marikina, Eastern Manila District, Metro Manila, 1801, Philippines

Distance	ETA	Total Estimated Cost
3.69 km	6 min	₱ 105.42

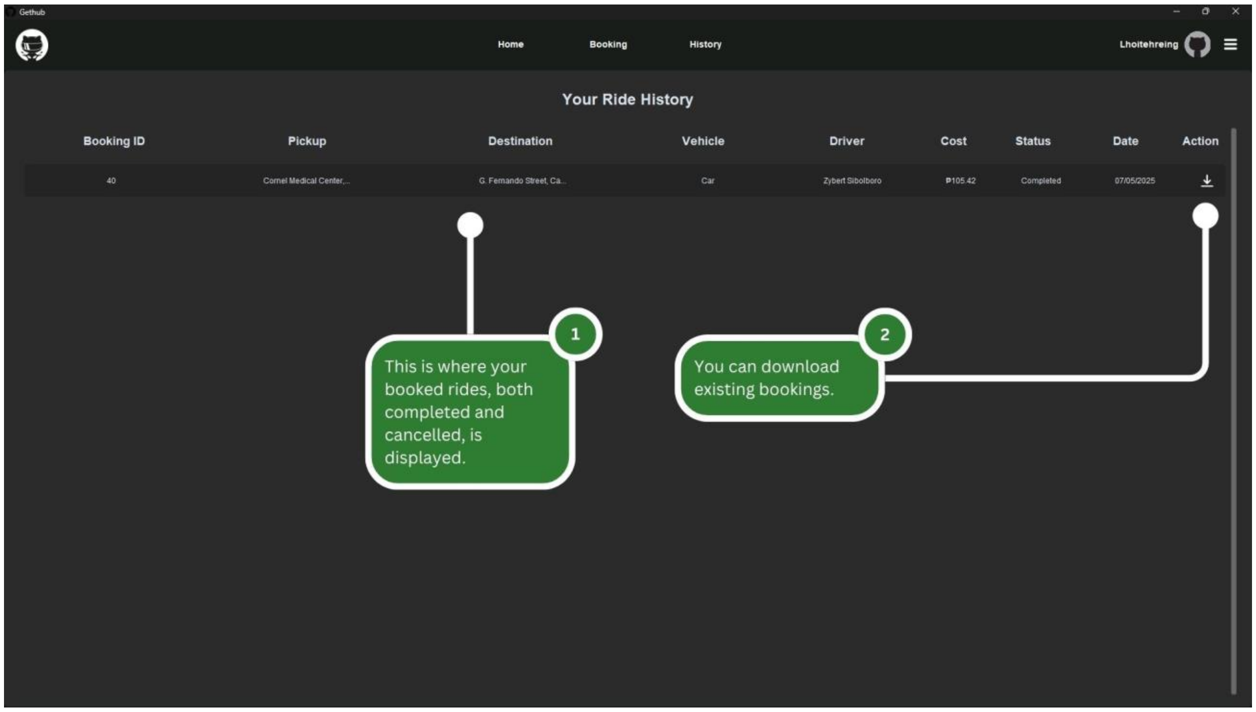
Ride Information:

Type:	Car	Model:	McLaren F1
License Plate:	MRPL8S	Driver:	Zybert Sibolboro

11 You can still cancel the booking.

10 Once the booking process has finished, you can now view it in your history tab.

Viewing and downloading booking history guide

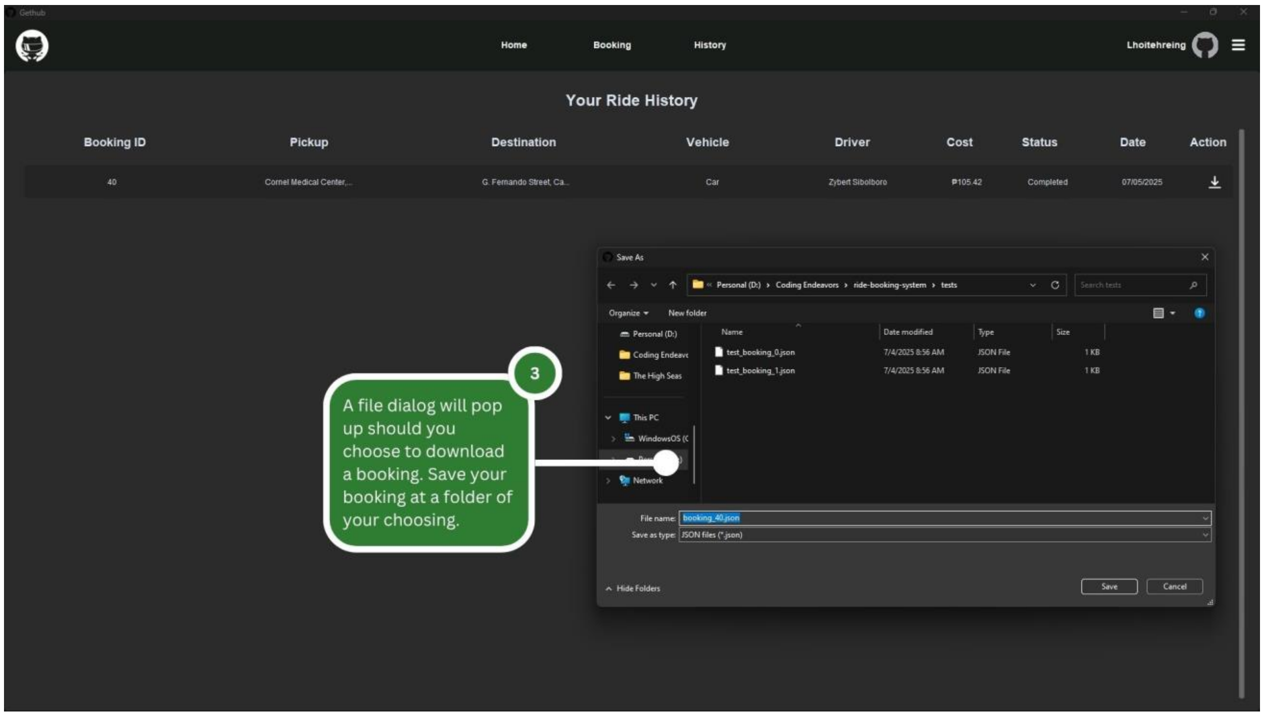


Your Ride History

Booking ID	Pickup	Destination	Vehicle	Driver	Cost	Status	Date	Action
40	Cornel Medical Center,...	G. Fernando Street, Ca...	Car	Zybert Sibolboro	₱105.42	Completed	07/05/2025	⤓

1 This is where your booked rides, both completed and cancelled, is displayed.

2 You can download existing bookings.



Your Ride History

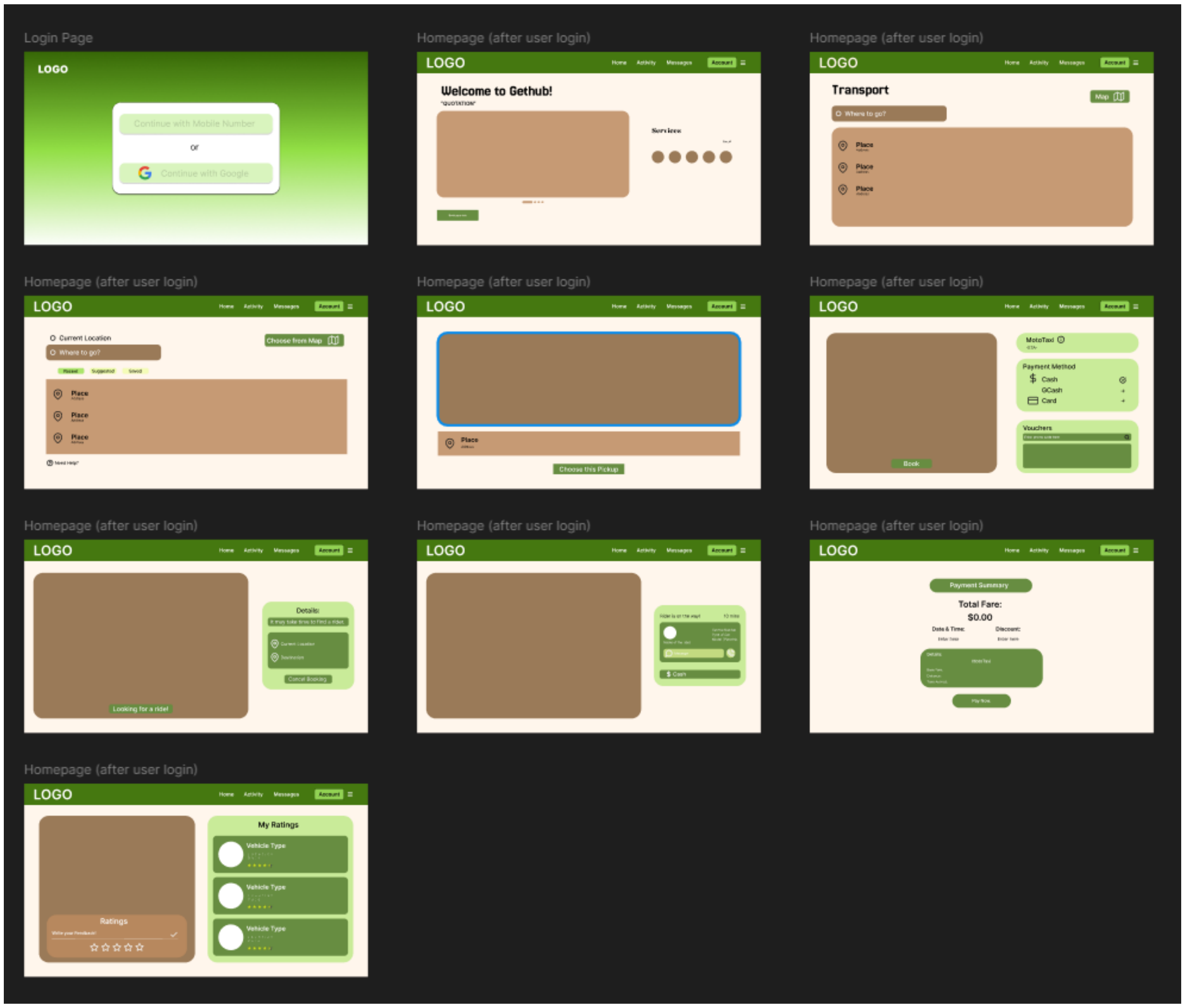
Booking ID	Pickup	Destination	Vehicle	Driver	Cost	Status	Date	Action
40	Cornel Medical Center,...	G. Fernando Street, Ca...	Car	Zybert Sibolboro	₱105.42	Completed	07/05/2025	⤓

Name	Date modified	Type	Size
test_booking_0.json	7/4/2025 8:56 AM	JSON File	1 KB
test_booking_1.json	7/4/2025 8:56 AM	JSON File	1 KB

File name: booking_40.json
Save as type: JSON files (*.json)

A file dialog will pop up should you choose to download a booking. Save your booking at a folder of your choosing.

Wireframes


Login Page
LOGO
Continue with Mobile Number
or
Continue with Google
Homepage (after user login)
LOGO
Home
Activity
Messages
Account
Welcome to Gethub!
Services
Transport
Where to go?
Place
Map
Current Location
Choose from Map
Need Help?
Choose this Pickup
MotoTaxi
Payment Method
Cash
GCash
Card
Vouchers
Book
Details:
Current Location
Destination
Cancel Booking
Looking for a ride!
Payment Summary
Total Fare:
$0.00
Date & Time:
Discount:
Pay Now
My Ratings
Vehicle Type
Ratings
Write your feedback