# 

**GoDavao: A Ridesharing App in Davao City**

**De Los Santos, Elijah Nataniel S.**

**Dela Rosa, Vanne Eloise C.**

**Fernandez, Raphael Vince S.**

**ATENEO DE DAVAO UNIVERSITY   
SCHOOL OF ARTS AND SCIENCES**

**COMPUTER STUDIES CLUSTER**

**April 2025**

**GoDavao: A Ridesharing App in Davao City**

**A Thesis Presented to the**

**Faculty of the**

**Computer Studies Cluster   
Ateneo de Davao University**

**In Partial Fulfillment**

**of the Requirements for the Degree**

**Bachelor Of Science in Information Systems**

**By**

**De Los Santos, Elijah Nataniel S.**

**Dela Rosa, Vanne Eloise C.**

**Fernandez, Raphael Vince S.**

**ATENEO DE DAVAO UNIVERSITY**

**SCHOOL OF ARTS OF SCIENCES**

**COMPUTER STUDIES CLUSTER**

**APRIL 2025**

[ABSTRACT 4](#_xeuwfkkusbys)

[**Chapter 1 6**](#_3dqaot3nk1ok)

[Introduction 6](#_jkfoa31kn37x)

[1.1 Background of the Study 6](#_5edeu7i8zrup)

[1.2 Problem Statement 9](#_tgu93iev2pn5)

[1.3 Objectives of the Study 11](#_p1pl267hxoat)

[1.4 Significance of the Study 12](#_f7r6xv3ikfgs)

[1.5 Scope and Limitations 13](#_rkhd7wbhm87h)

[**Chapter 2: Review of Related Literature and Studies 17**](#_fufuht8f70v0)

[I. Ridesharing Technologies, Algorithms, and Methods 17](#_7tudu9fz80s0)

[Algorithms for Ridesharing 18](#_6mgscju2yr6l)

[Machine Learning in Ridesharing 21](#_1fpxaig2i3y)

[Geographic Information Systems (GIS) for Spatial Context and Visualization 23](#_98zdi3uhcfi3)

[II. User Perception and Acceptance of On-Route Ridesharing 24](#_nl8rsfm8jo3s)

[III. Application of Chosen Methods in Similar Contexts 26](#_awrak4ksf71c)

[IV. Relevance to the Study in Davao City 27](#_xnm9jiqljqry)

[**Chapter 3: Methodology 31**](#_nfki9z8vx429)

[3.1 Conceptual Framework 31](#_63gun8azim1p)

[3.2 Research Design 33](#_d0np86masjn6)

[3.3 Data Collection or Generation 34](#_60eoflvz87eb)

[3.4 Data Analysis 35](#_5yftrtd4eiph)

[3.5.1 Data Privacy and Security 36](#_xawillajas7z)

[3.5.2 Fairness and Algorithmic Bias 36](#_14bt8w5i2dmk)

[3.5.3 Safety, Security, and Trust 37](#_xe9svd1k1f67)

[3.5.4 Transparency and Accountability 38](#_87naft2rfooh)

[3.6 Operational Framework 38](#_2az7v7i2nz6y)

[References 42](#_j5p7o4e38ao)

## 

**ABSTRACT**

Urban transportation systems in cities like Davao face persistent challenges, including traffic congestion, inefficient route planning, and suboptimal vehicle utilization, leading to increased travel times and commuter frustration. Existing ridesharing services often operate on a rigid point-to-point model, failing to capitalize on opportunities to dynamically group passengers traveling along similar paths. This could significantly improve efficiency and reduce costs. This study addresses this gap by proposing the design and evaluation of a dynamic ridesharing platform concept specifically for the context of Davao City. Leveraging information system technologies such as Geographic Information Systems (GIS) and optimization algorithms (e.g., a Heuristic-Based Matching Algorithm and Dijkstra’s Algorithm), the platform aims to dynamically group passengers based on their location and destination and optimize vehicle routes to accommodate multiple riders efficiently. While a full implementation is beyond the scope, the conceptual design incorporates features related to dynamic pricing and safety to enhance fairness, transparency, and user trust in the model. The research investigates the feasibility and potential impact of such a system in mitigating traffic issues, reducing transportation costs for commuters and drivers, and improving the overall urban mobility experience in Davao City through real-world testing and conceptual evaluation. The expected outcome is a detailed conceptual design and a limited-scope prototype demonstrating the core functionalities of dynamic passenger matching, illustrating how data-driven approaches can lead to more efficient ridesharing solutions tailored to the needs of Davao City.

**Keywords:** Ridesharing, Davao City, Urban Mobility, Passenger Matching, Route Optimization, Transportation Efficiency, Information Systems, GIS, Heuristic Algorithm, Dynamic Ridesharing.

# **Chapter 1**

## **Introduction**

This chapter presents the background of the study, outlining the context and existing landscape of ridesharing, followed by the problem statement formulated as research questions that the research aims to address. It then details the specific, measurable objectives to be achieved, discusses the significance of the study to various stakeholders, and defines the scope and limitations that set the boundaries of this research. It provides the foundation and context for the proposed dynamic ridesharing system concept focused on improving urban transportation efficiency in Davao City through optimized passenger matching for efficient pickups.

## **1.1 Background of the Study**

Rapid urbanization and population growth in metropolitan areas worldwide, including Davao City, have put immense pressure on transportation infrastructure. This has resulted in significant challenges such as chronic traffic congestion, increased travel times, higher fuel consumption, and environmental pollution (Garcia & Santos, 2021; ResearchGate, 2025). Traditional public transportation systems and conventional taxi services often struggle to meet the dynamic demands of commuters efficiently.

The emergence of ridesharing applications has provided an alternative mode of transportation, offering convenience and flexibility. Existing literature highlights various algorithm-based ride-matching techniques to connect drivers with passengers (Makhdomi & Gillani, 2024; Sakthivelu & Jayakrishnan, 2024). Research also explores methods for analyzing ride demand and supply, aiming for better balance and resource allocation within ride-hailing networks (Makhdomi & Gillani, 2024; MDPI, 2023). Furthermore, studies on network effects discuss how the growth in users (both riders and drivers) on a platform can contribute to its scalability and reliability (ResearchGate, 2023; ResearchGate, 2025).

However, many current ridesharing models focus on one-to-one passenger-to-vehicle matching based on point-to-point requests. While this offers personalized travel, it often leads to vehicles carrying only one passenger for a significant portion of a trip, contributing to the number of cars on the road and potentially exacerbating congestion, especially during peak hours. There is a lack of widespread implementation of ridesharing systems that can dynamically adapt to real-time conditions and intelligently match passengers for efficient pickups along a driver's existing or planned route to maximize vehicle occupancy and minimize added travel time and distance, particularly in the context of urban environments like Davao City. This gap represents an opportunity to develop a more efficient and sustainable ridesharing model focused on intelligent pickup matching.

This gap is compounded by the challenge of effectively integrating and modeling localized, dynamic traffic data to accurately calculate pickup feasibility and estimated arrival times. Furthermore, research indicates a need for better methods to balance system efficiency (regarding pickup optimization) with user fairness (both driver and passenger, regarding acceptable detours or wait times) when implementing dynamic on-route matching. Additionally, there is a need to understand user acceptance and perceived value, specifically of the on-route ridesharing model within the Davao City context, and to design user-friendly and informative GIS visualizations that effectively support drivers in identifying and navigating efficient pickups. These specific challenges represent key research gaps that this study aims to investigate.

There is also research on pricing strategies in ride-hailing, including real-time dynamic pricing models that adjust fares based on demand and supply to incentivize drivers and manage passenger flow (SciELO, 2019; TRISTAN 2025, 2025). The implications of using cloud-based systems for fare calculation on transparency and fairness have also been examined (FasterCapital, n.d.; ijmrset, n.d.). Furthermore, the literature addresses safety and user trust in digital platforms (MDPI, 2024; ResearchGate, n.d.). Various theoretical frameworks, such as the Technology Acceptance Model (TAM) or theories related to perceived risk and trust, have been applied to understand the relationship between safety features and passenger trust (ResearchGate, n.d.; University of Southampton, 2024). Research on ridesharing in the Philippines highlights driver tactics and the local context (ResearchGate, n.d., Drivers' Tactics; UP NCTS, 2024).

Despite advancements in these areas, integrating these elements into a cohesive platform focused on dynamically and efficiently matching passengers for pickups along existing routes tailored to the unique traffic patterns and user needs of a city like Davao remains challenging. The existing systems often address these aspects in isolation rather than as components of a unified, efficiency-focused shared-ride model centered on intelligent pickup matching. Studies on urban transportation challenges in Davao City underscore the need for improved public transport alternatives and address accessibility disparities (Cognizance Journal of Multidisciplinary Studies, n.d.; ResearchGate, 2025).

In the context of Davao City, where urban sprawl continues and traffic volume increases, the need for innovative transportation solutions is exceptionally pressing. A ridesharing system that moves beyond the traditional point-to-point model towards a more dynamic, shared-ride approach focused on optimizing the matching of passengers for efficient on-route pickups while also conceptually incorporating robust safety and transparent pricing mechanisms holds the potential to not only offer a cheaper alternative to conventional taxis but also contribute significantly to reducing the number of vehicles on the road, lowering carbon emissions, and providing a more sustainable and less stressful commuting experience for its residents. This study anchors on the premise that a data-driven, dynamically optimized matching platform for efficient pickups can solve some of Davao City's pressing urban transportation challenges.

## **1.2 Problem Statement**

The current urban transportation landscape in Davao City, including existing ridesharing services, faces several critical issues related to efficiency, cost, and trust. This study addresses these problems by investigating the design and potential implementation of a dynamic ridesharing platform for Davao City that employs real-time passenger matching techniques. Specifically, this study aims to answer the following research questions:

* How can a dynamic ridesharing platform be effectively designed and conceptualized for the context of Davao City?  
  + **Efficiency & Ride Availability:**
    - What ride-matching techniques and route optimization algorithms best influence the efficiency of an on-route pickup ridesharing service in Davao City?
    - How can the platform efficiently manage and respond to fluctuating passenger hailing requests and driver availability within a dynamic ridesharing system in Davao City?
  + **Pricing & Economic Sustainability:**
    - How does real-time dynamic pricing impact user perception and market competitiveness of a ridesharing platform in Davao City?
    - What are the implications of cloud-based fare calculation systems on pricing transparency and fairness for users and drivers in Davao City?
  + **Safety & User Trust:**
    - How do potential users in Davao City perceive the influence of proposed digital security measures and in-app safety features on their trust in a dynamic ridesharing application?

## **1.3 Objectives of the Study**

The main objective of this study is to design and evaluate a dynamic ridesharing platform concept for Davao City that improves transportation efficiency, alleviates the traffic problem occuring in Davao City, lessens pollution caused by vehicle emissions, enhances fare predictability, and improves user safety through real-time passenger clustering and dynamic pricing, and integrated safety features.

Specifically, this study aims to achieve the following objectives:

* **To analyze the transportation challenges faced by commuters in Davao City** through surveys, interviews, and literature review.
* **To design a user-friendly mobile interface** that enables commuters to easily register, book rides, and offer rides using location-based services.
* **To implement core features of the ridesharing app** including user authentication, ride posting, ride matching, in-app messaging, and route tracking.
* **To integrate map-based navigation and geolocation services** using tools such as Google Maps API to assist drivers and passengers in locating pickup and drop-off points.
* **To test the functionality and usability of the prototype** through user testing sessions and gather feedback for improvements.
* **To evaluate the app’s performance, usability, and effectiveness** in addressing local commuting problems using a set of predefined criteria and user satisfaction surveys.

## 

## **1.4 Significance of the Study**

This study expects to yield significant benefits to various stakeholders in Davao City:

* **For Commuters:** The proposed system aims to provide a more affordable and potentially faster transportation option by facilitating shared rides and optimized driver finding, reducing travel time and cost. Enhanced safety features and transparent pricing will also improve the user experience.
* **For Drivers:** By optimizing route pickups and increasing passenger occupancy per trip, drivers can potentially increase their earnings and reduce fuel consumption and operational costs. Improved safety measures also benefit drivers.
* **For the Local Government and Davao City:** The study's findings can provide insights into how technology-driven ridesharing solutions can help alleviate traffic congestion, reduce carbon emissions, and contribute to developing a more sustainable urban transportation system in Davao City.
* **For Future Researchers and Developers:** This study will contribute to the body of knowledge on dynamic ridesharing systems, particularly in developing urban environments, providing a foundation for future research and development in this area.

## **1.5 Scope and Limitations**

**Scope:** This study focuses on designing and conceptualizing a dynamic ridesharing platform for Davao City, incorporating real-time passenger matching using a heuristic-based algorithm, dynamic route calculation with Dijkstra's, GIS visualization, and conceptually addressing dynamic pricing and key safety features. It will primarily involve:

* **Core Features:**
  + User Registration and Profiles: Basic functionality for users (passengers and drivers) to create accounts.
  + Passenger Ride Request: The passenger can input their pickup location and destination.
  + Driver Availability Status: Drivers can set their status (available/unavailable).
  + Real-time Location Tracking (GIS-based): Displaying the real-time location of available drivers and the passenger (once a ride is requested) on a map interface.
  + Dynamic On-Route Passenger Matching (Heuristic-Based): The core logic for identifying available drivers whose current route is compatible with a new passenger's request and applying heuristic rules to determine the best match.
  + Ride Offer Notification (Driver): Notifying a driver when a potential on-route match is found, including details about the passenger and detour.
  + Ride Acceptance/Rejection (Driver): The Driver can accept or reject a ride offer.
  + Dynamic Route Calculation (Dijkstra's based): Calculating the optimal route for a driver, including dynamically updating the route to incorporate accepted on-route pickups and subsequent drop-offs.
  + In-App Navigation Guidance (GIS Visualization): Displaying the calculated route on the driver's map interface with visual cues for navigation, pickup points, and drop-off points. This includes the specifically designed and qualitatively evaluated GIS visualizations.
  + Ride Progress Tracking (Passenger): Displaying the driver's location and the planned route on the passenger's map once a match is accepted.
* **Conceptual Features:**
  + Dynamic Pricing Mechanism: The logic and rules for adjusting fares in real-time based on factors like demand (observed through request volume), supply, distance, time, and potential detours for shared rides.
  + Fare Calculation and Display: Showing the calculated fare to the passenger (potentially an estimate before the ride and final after).
  + Digital Payment Integration: Mechanisms for handling payments through digital wallets (e.g., PayMongo, GCash, GrabPay) or other methods.
  + User Authentication and Security: Robust login, password management, and data protection measures.
  + Driver Verification: Processes for verifying driver identity and background.
  + In-App Emergency Button: A feature for users to quickly contact emergency services or the platform's support.
  + Ride History and Receipts: Access to past trip details and payment receipts.
  + Driver/Passenger Rating and Review System: Allowing users to rate and review each other after a trip.

**Limitations:** This study is subject to the following limitations:

* **Data Availability:** The availability of extensive real-time traffic, granular passenger demand, and historical pricing data for Davao City may limit the precision of the conceptual analysis for features not fully prototyped.
* **External Factors:** The study's analysis of potential impact may not fully account for all external factors influencing transportation, such as road closures, accidents, changes in local transportation regulations, or market competition from existing services.
* **Specific Vehicle Types:** The study may initially focus on a specific type of vehicle (e.g., 4-wheeled vehicles) and may not cover all potential modes of ridesharing (e.g., motorcycles) unless specified within the design.
* **Comprehensive Safety System:** While key safety features are included conceptually, developing a comprehensive, legally compliant safety and security framework for a real-world application is a complex task beyond this design-focused study's scope.
* **Handling of Cases and Exceptions:** The complexity of real-world scenarios, such as multiple simultaneous ride requests in the exact location, unexpected road closures not reflected in traffic data, passenger cancellations after a detour has begun, or driver issues, presents operational complexities that the prototype may only address at a basic level.

# 

# **Chapter 2: Review of Related Literature and Studies**

This chapter reviews relevant literature and studies related to ridesharing, urban transportation, optimization algorithms, machine learning techniques, and Geographic Information Systems (GIS). Based on the scope and timeframe of this capstone project, the review concentrates on technologies and concepts deemed most critical and feasible for designing and evaluating a dynamic ridesharing platform concept for Davao City, with a primary focus on optimizing passenger matching for efficient pickups. It is structured to move from general concepts to more specific applications, highlighting problems addressed, achievements made, and recommendations for future work within these areas, ultimately focusing on how these relate to the particular context and chosen methods for this study in Davao City and addressing identified research gaps. The technologies and concepts retained for focus in this review are Heuristic-Based Matching Algorithms (with consideration of Genetic Algorithms as an advanced approach), Dijkstra's Algorithm / Search for cost calculation, Geographic Information Systems (GIS) for spatial context and visualization, and the critical aspect of User Perception and Acceptance of on-route ridesharing.

## **I. Ridesharing Technologies, Algorithms, and Methods**

The efficiency and effectiveness of modern ridesharing platforms are heavily reliant on applying sophisticated algorithms, machine learning techniques, and spatial information systems. This section explores recent advancements and applications in the key areas relevant to this study within the general domain of ridesharing, highlighting their capabilities and limitations in the context of dynamic, on-route passenger matching.

### **Algorithms for Ridesharing**

Various algorithms tackle core ridesharing challenges, including matching drivers with passengers and optimizing routes for multiple stops. The matching process is particularly complex for dynamic ridesharing with on-route pickups, requiring efficient algorithms that can operate in real-time.

**Heuristic-Based Matching Algorithms for On-Route Pickups**

* **Problems:** Efficiently pairing drivers with new passenger requests in real-time, especially when the driver is already on a trip and potential pickups are along their route, presents a complex, dynamic optimization challenge. Exact algorithms can be computationally prohibitive for large-scale, real-time systems (Masoud & Jayakrishnan, 2017; ResearchGate, n.d., A Matching Algorithm). While Genetic Algorithms offer a powerful optimization approach, heuristic methods are often favored in practice for their speed and feasibility in dynamic environments (Masoud & Jayakrishnan, 2017; CEUR-WS.org, 2025).
* **Achievements (Results):** Heuristic-based matching algorithms utilize predefined rules and criteria to evaluate potential driver-passenger pairings for on-route pickups quickly. These methods aim to find "good enough" solutions efficiently by applying rules based on factors such as the maximum acceptable detour time or distance for the driver and existing passengers, geographic proximity of the pickup point to the current route, available vehicle capacity, and compatibility of destinations (Masoud & Jayakrishnan, 2017; Makhdomi & Gillani, 2024; ResearchGate, n.d., A Matching Algorithm). Studies explore different heuristic rules and their impact on metrics like passenger waiting time and driver idle time (ResearchGate, n.d., A Matching Algorithm; ResearchGate, 2025, Dynamic matching radius).
* **Recommendations:** Research recommends developing heuristics that effectively balance system efficiency (e.g., maximizing pickups, minimizing total travel time) with user fairness (e.g., ensuring detours are reasonable). Evaluating the performance of different heuristic rules through real-world testing is crucial. Heuristic design should consider scalability and adaptability to varying demand and supply conditions in urban settings. Despite their practical use, a research gap exists in systematically evaluating and comparing the performance of different heuristic strategies specifically for dynamic on-route pickup matching in developing urban environments with unique traffic characteristics.

**Genetic Algorithm**

* **Problems:** While heuristics offer speed, finding globally optimal solutions for complex dynamic ridesharing matching problems with numerous constraints (vehicle capacity, time windows, multiple pickups/drop-offs, balancing efficiency and fairness) remains a significant challenge that can be difficult for simple heuristics to fully address.
* **Achievements (Results):** Genetic Algorithms, as an evolutionary computation method, are utilized to optimize ride-matching by exploring a large search space of potential solutions (sets of driver-passenger pairings) and iteratively improving them based on a fitness function that considers multiple objectives (Cao, Wang, & Li, 2021; Masoud & Jayakrishnan, 2017). This approach can potentially find more optimal pairings compared to simple heuristics, especially in complex, multi-passenger scenarios, and can be used for offline optimization or benchmarking real-time methods (CEUR-WS.org, 2025; FHWA, n.d.).
* **Recommendations:** Studies recommend adapting Genetic Algorithms to the specific constraints of ridesharing, incorporating real-time data, and developing efficient evaluation mechanisms for potential solutions. While potentially more computationally intensive for strict real-time applications than heuristics, they offer a powerful approach for complex optimization problems. A gap remains in the practical application and comparative analysis of Genetic Algorithms, specifically for dynamic on-route pickup matching in the context of developing urban areas, and how their potential benefits in optimality weigh against the computational demands for real-time implementation compared to efficient heuristics.

**Dijkstra’s Algorithm Search for Pickup Cost Calculation**

* **Problems:** Accurately estimating the time and distance costs associated with potential on-route pickups and the resulting detours is crucial for the matching algorithm to make informed decisions. This requires efficient shortest-path calculations within the dynamic urban road network (AFI, n.d.). Integrating real-time traffic data to make these cost estimations accurate is a key challenge.
* **Achievements (Results):** Dijkstra's Algorithm and its variants, like Search, are fundamental tools for finding the shortest or fastest path between two points in a graph, making them essential for calculating travel costs in ridesharing systems (Dijkstra, 1959; Codecademy, n.d.; FarEye, n.d.). By modeling road networks as graphs, these algorithms can efficiently determine the estimated time and distance for a driver to reach a potential pickup point and continue to the destination, considering the road network structure (ResearchGate, n.d., A Matching Algorithm) and integrating real-time or time-dependent traffic data as edge weights can improve the accuracy of these cost estimations (DiVA portal, n.d.; Fu, Sun, & Rilett, 2006).
* **Recommendations:** Studies suggest enhancing the accuracy of route and cost calculations by combining Dijkstra’s Algorithm with real-time traffic data. Hybrid techniques are recommended for better performance in large urban networks. A significant research gap exists in effectively integrating and modeling localized, dynamic traffic data specific to urban environments like Davao City as accurate, real-time edge weights within shortest-path algorithms for precise pickup feasibility and cost calculation in a dynamic on-route ridesharing context.

### **Geographic Information Systems (GIS) for Spatial Context and Visualization**

Geographic Information Systems (GIS) provide the essential spatial context and tools necessary for effective data management, spatial analysis, and visualization in ridesharing, supporting both algorithm inputs and user interaction.

* **Problems:** Ridesharing operations are inherently spatial, requiring accurate mapping, real-time location tracking, and the ability to perform spatial queries (Cao, Wang, & Li, 2021). Furthermore, effectively presenting complex spatial information, such as potential pickup points and dynamic routes, to drivers and passengers through a user-friendly interface is crucial for usability and acceptance (Bac Ha Software, n.d.; Warse, 2024).
* **Achievements (Results):** GIS provides the foundational capabilities for modeling urban road networks, storing and managing spatial data (like points for locations, lines for routes), performing spatial analysis (e.g., calculating distances, identifying points within a radius), and visualizing real-time locations and routes on interactive maps (Cao, Wang, & Li, 2021; Nature.com, 2024; pmc.ncbi.nlm.nih.gov, 2020; Goong, 2023). GIS is integral to route optimization and dispatching systems (Wang & Wei, 2018; Nguyen, Le, & Tran, 2019). Local examples demonstrate the use of GIS in pooling apps (Warse, 2024).
* **Recommendations:** Recommendations include leveraging real-time GIS capabilities for dynamic route adjustments and integrating AI/ML with GIS for predictive spatial analysis. Continuous improvement of spatial data accuracy is essential. Despite the critical role of GIS, a research gap exists in systematically designing and evaluating user-friendly and informative GIS visualizations specifically tailored for drivers navigating dynamic on-route pickups in a ridesharing context, considering the need for clear, real-time information display to minimize cognitive load and support efficient decision-making.

## **II. User Perception and Acceptance of On-Route Ridesharing**

The success and sustainability of a dynamic ridesharing platform, which relies on passengers sharing rides and potentially experiencing detours for pickups, depend significantly on user perception, trust, and acceptance. Understanding these factors within the specific local context is crucial.

* **Problems:** Users (both passengers and drivers) may have concerns about shared rides, including privacy, safety, comfort, and the fairness of detours and pricing (MDPI, 2024; ResearchGate, n.d., Trust; ResearchGate, n.d., Drivers' Tactics; University of Southampton, 2024; MDPI, 2021). Factors influencing trust in digital platforms and the sharing economy are complex and can vary across regions and cultures (ResearchGate, n.d., Trust; University of Southampton, 2024). Understanding these perceptions in Davao City and the Philippines is vital for designing an acceptable and trustworthy service (ResearchGate, n.d., Drivers' Tactics; UP NCTS, 2024; ResearchGate, 2025, Informal and Shared Mobility).
* **Achievements (Results):** Research on ride-hailing in the Philippines highlights factors influencing driver tactics and user experiences, including responses to traffic and incentives (ResearchGate, n.d., Drivers' Tactics; UP NCTS, 2024). Studies on user trust in the sharing economy emphasize the roles of digital security measures, transparency, and platform legitimacy (MDPI, 2024; ResearchGate, n.d., Trust; University of Southampton, 2024). Theoretical frameworks like the Technology Acceptance Model (TAM) and Uncertainty Reduction Theory (URT) provide lenses for understanding technology adoption and trust development (ResearchGate, n.d., Trust). Southeast Asian studies discuss the role and challenges of informal and shared mobility, including digitally-enabled ride-hailing (ResearchGate, 2025, Informal and Shared Mobility; MDPI, 2021).
* **Recommendations:** Conducting qualitative studies (e.g., interviews, focus groups) to gather in-depth insights into potential users' understanding of, concerns about, and perceived benefits and barriers to adopting an on-route ridesharing system in a specific local context like Davao City. Evaluating how proposed features (like clear communication about detours, safety measures, and transparent pricing) influence perceived trust and value is essential for design refinement. A research gap exists in specific, in-depth investigations into user acceptance and the perceived value of the on-route ridesharing model, particularly concerning the willingness to share rides and accept dynamic pickups/detours, within the specific socio-cultural and transportation context of Davao City and similar urban areas in the Philippines.

## **III. Application of Chosen Methods in Similar Contexts**

The algorithms, machine learning techniques, and GIS chosen for this study have also been successfully applied in contexts similar to ridesharing, demonstrating their broader applicability and effectiveness in transportation and logistics.

In traffic systems, Dijkstra’s Algorithm and its variants are widely used for shortest path computation within complex urban networks, essential for real-time navigation and minimizing travel time (Dijkstra, 1959; Fu, Sun, & Rilett, 2006). Integrating real-time traffic data can significantly improve route accuracy (DiVA portal, n.d.).

Grab utilizes algorithms to match passengers with various vehicle types (cars, motorcycles, taxis) and food and package delivery, all requiring efficient dispatching and combining heuristics with variants of genetic algorithms to fulfill their needs when it comes to matching.

Within ridesharing platforms themselves, AI-based ride matching algorithms, often leveraging techniques like heuristics, Genetic Algorithms, or other machine learning approaches, have shown significant improvements in pairing efficiency and reducing wait times compared to traditional methods (Kim et al., 2022; Zhang & Xie, 2023; CEUR-WS.org, 2025). Research continues handling complex scenarios and incorporating fairness and efficiency balancing (Taylor & Francis Online, 2025; MDPI, 2024, Fairness-Aware Dynamic).

## **IV. Relevance to the Study in Davao City**

This section connects explicitly the chosen algorithms, machine learning techniques, GIS, and user perception considerations to the context of ridesharing in Davao City, emphasizing their direct relevance to this study's objectives and the identified research gaps. The urban environment of Davao City presents unique traffic patterns, infrastructure characteristics, and commuter needs that necessitate a tailored approach to ridesharing optimization focused on efficient pickups.

Applying a Heuristic-Based Matching Algorithm is central to addressing the challenge of efficiently pairing drivers with passengers for on-route pickups in real-time within Davao City's dynamic environment. This approach offers a feasible method for the prototype to evaluate numerous potential pairings and refine them to achieve efficient allocation, considering constraints such as vehicle capacity and time windows relevant to Davao City. Adapting the heuristic rules to the city's specific geographical characteristics and incorporating local considerations for balancing efficiency and fairness (e.g., setting appropriate detour limits based on typical Davao travel patterns) will be key to improving real-time performance and ensuring the matching process is perceived as equitable by local users (Masoud & Jayakrishnan, 2017; Makhdomi & Gillani, 2024; Taylor & Francis Online, 2025; MDPI, 2024, Fairness-Aware Dynamic). While genetic algorithms represent a more complex approach for potential future work, the heuristic approach provides a practical method for exploring the matching optimization problem in this context.

Dijkstra’s Algorithm will be instrumental in accurately estimating the time and distance costs associated with potential passenger pickups and detours within Davao City's road network. Utilizing this for cost calculation is crucial input for the heuristic matching algorithm. Integrating it with real-time time-dependent traffic data specific to Davao City will be essential for its effectiveness in providing realistic cost estimations amidst the city's variable congestion, directly addressing the gap in localized traffic data integration for pickup feasibility calculation (Dijkstra, 1959; Codecademy, n.d.; AFI, n.d.; FarEye, n.d.; DiVA portal, n.d.).

The integration of GIS provides the essential spatial framework for implementing and visualizing the dynamic ridesharing system in Davao City. GIS tools for mapping, spatial analysis, and visualization of driver/passenger locations, potential pickup points, and recommended paths are fundamental to this study. Utilizing real-time GIS data for dynamic updates and designing user-friendly GIS visualizations specifically for drivers identifying and navigating efficient on-route pickups within Davao City are crucial for the proposed platform's usability and directly address the gap in GIS visualization design for this context (Cao, Wang, & Li, 2021; Nature.com, 2024; pmc.ncbi.nlm.nih.gov, 2020; Goong, 2023; Bac Ha Software, n.d.; Warse, 2024).

Furthermore, understanding the User Perception and Acceptance of this on-route ridesharing model within the specific socio-cultural and transportation context of Davao City is paramount. Qualitative evaluation will explore local user attitudes, concerns (e.g., about sharing rides, detours for pickups, safety), and perceived benefits to ensure the conceptual design and future implementation are well-received and address local needs, directly addressing the gap in understanding user acceptance of this specific model in the Davao context (ResearchGate, n.d., Trust; ResearchGate, n.d., Drivers' Tactics; University of Southampton, 2024; UP NCTS, 2024; ResearchGate, 2025, Informal and Shared Mobility; MDPI, 2021).

Together, these chosen methods – a Heuristic-Based Matching Algorithm and Dijkstra's for pickup cost calculation and matching optimization, GIS for spatial context and visualization, and qualitative evaluation for user perception – provide a robust foundation for designing and evaluating a dynamic ridesharing platform concept tailored to enhance urban mobility in Davao City by improving matching efficiency for pickups, leveraging data-driven insights, and considering local user needs.

# **Chapter 3: Methodology**

This study applies a hybrid methodology that integrates the Design Science Research (DSR) framework with an Agile development strategy

The combination ensures both scientific rigor (through DSR) and practical adaptability (through Agile sprints), aligning conceptual frameworks with real-world implementation.

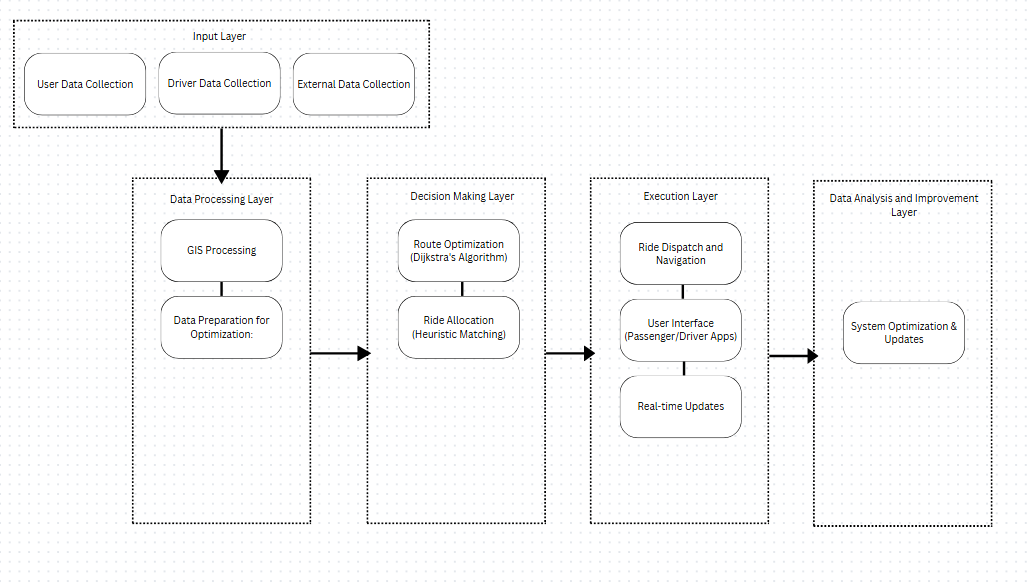
## **3.1 Research Design**

This study adopts the **Design Science Research (DSR) framework** as its overarching methodology, emphasizing the creation and evaluation of an artifact, in this case, the GoDavao mobile application to address the real-world problem of traffic congestion and inefficient ride coordination in Davao City. Within this framework, the research process began with problem identification, which recognized the lack of efficient ride-sharing systems in the city as a major contributor to congestion. The objectives of the solution were then defined, focusing on the development of a dynamic, on-route ridesharing platform optimized through the use of Dijkstra’s algorithm and heuristic approaches to enhance passenger–driver matching. The design and development stage translated these objectives into the conceptualization of GoDavao, a mobile application with core functionalities such as passenger–driver matching, real-time live tracking, safety mechanisms, and digital payments. This was followed by the demonstration phase, where prototypes and operational flows were iteratively developed into functional modules of the app. Evaluation was conducted through continuous testing of both technical performance and user acceptance, ensuring that refinements were evidence-based and user-centered. Finally, the communication phase involved presenting and documenting the outcomes of each cycle as part of the capstone project, thereby grounding the work in both academic and practical contexts.

To operationalize this design framework, the project applied the **Agile Scrum methodology** as the execution model for development. Agile Scrum provided a flexible yet structured approach to building the app in increments, enabling the team to address challenges systematically while staying aligned with the research objectives. In practice, the product backlog served as the running list of prioritized features, including authentication, ride requests, driver routes, ride matching, live tracking, safety, and payments. From this backlog, sprint backlogs were defined, containing the specific features to be completed during each sprint. Each sprint concluded with the delivery of an increment, such as a functioning login system, a route visualization component, or real-time tracking functionality. The process was further reinforced by Scrum events: sprint planning sessions defined clear objectives for each iteration, asynchronous daily stand-ups allowed the team to share progress and blockers despite scheduling differences, sprint reviews provided opportunities to demonstrate and evaluate the completed features, and sprint retrospectives enabled the team to reflect on encountered issues, such as Supabase session loss, routing errors, and row-level security (RLS) restrictions, and establish corrective actions for subsequent sprints.

The integration of **DSR and Agile Scrum** created a synergy that allowed the project to balance academic rigor with practical development efficiency. While DSR provided the theoretical foundation, ensuring that every activity was tied back to solving the identified transportation problem, Agile Scrum translated these theoretical goals into actionable, iterative development cycles. The mapping between the two was evident: the problem identification phase of DSR informed the product backlog in Agile, ensuring that features were prioritized in direct response to identified needs. Similarly, the objectives of the solution shaped sprint planning, where research goals became development targets. Design and development under DSR were realized through sprint execution, producing tangible increments of the app. Demonstration was achieved in sprint reviews, where outputs were evaluated not only for functionality but also for alignment with research objectives. Evaluation within DSR was mirrored in sprint retrospectives, where challenges and failures became feedback loops for refinement. Finally, the communication phase aligned with both documentation and delivered increments, which contributed to academic reporting while also showcasing functional progress. By aligning DSR’s research rigor with Agile’s iterative adaptability, the project ensured that each sprint was simultaneously a development cycle and a research cycle, enabling GoDavao to emerge as both a functioning ridesharing solution and a validated academic artifact.

### **3.1.1 Conceptual Framework**

**

**3.1.3 Conceptual Framework Mapping**

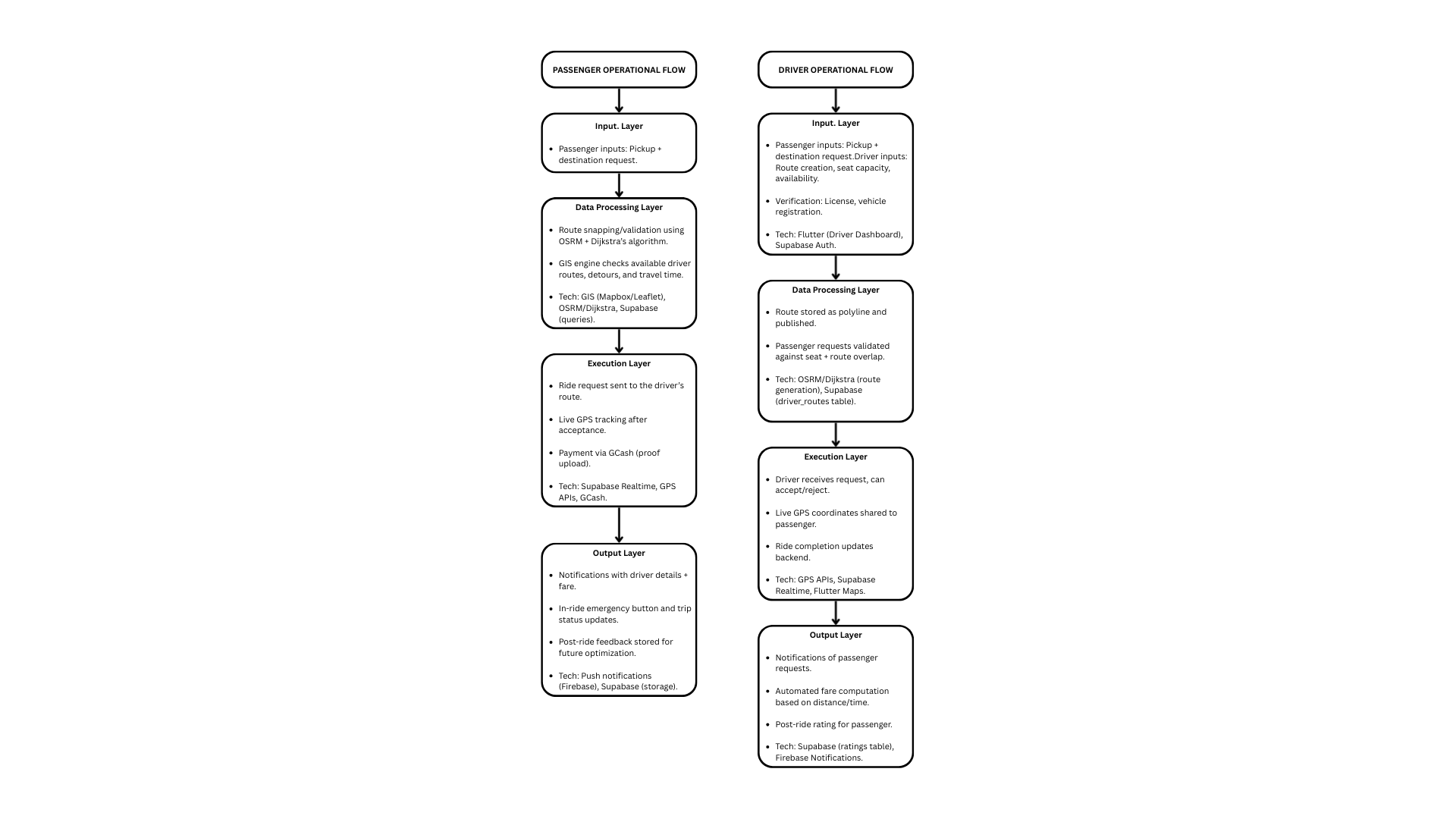
| **Research Objective (RO)** | **Conceptual / Theoretical Basis** | **Expected Output** |
| --- | --- | --- |
| **RO1.** To analyze the transportation challenges faced by commuters in Davao City through surveys, interviews, and literature review. | Problem Analysis Model; Needs Assessment Framework; Secondary Data Review | A comprehensive summary of transportation issues, user pain points, and mobility gaps specific to Davao City. |
| **RO2.** To design a user-friendly mobile interface that enables commuters to easily register, book rides, and offer rides using location-based services. | Design Science Research (DSR) Cycle; Agile Scrum Framework; Human–Computer Interaction (HCI) and UI/UX Principles | A functional **GoDavao** mobile interface prototype emphasizing ease of navigation and accessibility for both passengers and drivers. |
| **RO3.** To implement core features of the ridesharing app including user authentication, ride posting, ride matching, in-app messaging, and route tracking. | Software Engineering Best Practices; Modular Development Pattern; GIS and Real-time Database Integration (Supabase + Flutter) | Working system modules for authentication, ride management, real-time route tracking, and in-app communication. |
| **RO4.** To integrate map-based navigation and geolocation services using tools such as Google Maps API to assist drivers and passengers in locating pickup and drop-off points. | Geographic Information Systems (GIS); Shortest-Path Algorithms (Dijkstra, Heuristic Routing); Spatial Visualization Principles | Accurate map-based navigation and dynamic routing module that visualizes driver and passenger positions in real time. |
| **RO5.** To test the functionality and usability of the prototype through user testing sessions and gather feedback for improvements. | Agile Iteration and Feedback Loop; User Acceptance Testing (UAT); ISO/IEC 25010 Quality Model – Usability Attribute | UAT reports and usability metrics summarizing task success rates, errors, and refinements implemented from user feedback. |
| **RO6.** To evaluate the app’s performance, usability, and effectiveness in addressing local commuting problems using predefined criteria and satisfaction surveys. | Technology Acceptance Model (TAM); Quantitative Evaluation Framework; Survey Analysis (Perceived Usefulness, Ease of Use, Trust) | Statistical analysis of TAM construct scores and UAT results demonstrating GoDavao’s efficacy and user acceptance. |

### **3.1.4 Operational Framework**

### 

**3.1.5 Operational Framework Mapping**

| **Research Objective (RO)** | **Operational Method / Approach** | **Tools, Technologies, and Frameworks** | **Validation Technique / Expected Output** |
| --- | --- | --- | --- |
| **RO1.** Analyze the transportation challenges faced by commuters in Davao City. | Conduct surveys and semi-structured interviews; literature review of urban transport studies. | Google Forms / Qualtrics for survey; thematic analysis; Excel/Sheets for data tabulation. | Thematic summary of commuting issues and frequency data identifying key challenges. |
| **RO2.** Design a user-friendly mobile interface for registration, booking, and ride-offering. | UI/UX prototyping, wireframing, iterative design under Agile sprints. | Flutter 3.24 + Material 3 Design system. | Usability inspection and heuristic evaluation (Fitts’s Law, Nielsen’s 10 Heuristics). |
| **RO3.** Implement core features (authentication, ride posting, matching, messaging, tracking). | Modular development and incremental integration using Agile Scrum. | Flutter + Supabase (PostgreSQL, Realtime, Auth); Provider state management. | Unit and integration tests; working MVP modules validated via sprint reviews. |
| **RO4.** Integrate map-based navigation and geolocation services for accurate routing. | GIS integration and shortest-path computation. | OSRM v5.27 backend; Dijkstra’s Algorithm; flutter\_map + latlong2 libraries. | Functional testing vs Google Maps route benchmarks; latency < 2 s validation. |
| **RO5.** Test functionality and usability of the prototype via user testing and feedback. | User Acceptance Testing (UAT) sessions; Agile sprint retrospectives. | Android APK builds; Observation checklists. | UAT performance metrics (task success ≥ 90 %, error < 5 %); qualitative feedback summary. |
| **RO6.** Evaluate app performance, usability, and effectiveness using satisfaction surveys. | Post-implementation evaluation applying the Technology Acceptance Model (TAM). | Survey forms; Likert 5-point scale (PU, PEOU, Trust); SPSS / Excel analysis. | Computed mean per construct ≥ 4.0 (Agree/Very Satisfied); summary of user perceptions. |



The operational framework of the GoDavao system is divided into two complementary flows: the **Passenger Operational Flow** and the **Driver Operational Flow**. Together, these processes illustrate how the system supports ride-sharing through route visualization, request handling, live tracking, and post-ride evaluation.

**Passenger Operational Flow**The passenger flow begins when the user opens the GoDavao mobile application. Through the Passenger Map Page, the passenger is presented with available driver routes, displayed on a GIS-powered map. Unlike traditional ride-hailing models, where nearby drivers are shown individually, GoDavao allows passengers to select from pre-published driver routes that align with their general travel corridor.

Once a route is selected, the passenger defines their pickup and drop-off points, which are automatically snapped to the chosen route for accuracy. The system may further validate this by computing the shortest path segment using OSRM and Dijkstra’s algorithm. This mirrors existing research on path optimization in destination-oriented ridesharing drivers, where shortest path validation ensures efficiency and feasibility (Bhatia, 2020). After confirmation, a ride request is submitted to the backend (Supabase), where it is linked to the driver’s active route and validated for feasibility in terms of seat capacity and route compatibility. This follows similar approaches in industry applications such as Grab, which leverages real-world patterns to improve passenger–driver matching (Abeywickrama & Liang, 2021).

If a compatible driver route exists, the passenger receives a notification containing driver details, estimated fare, and estimated pickup time. Upon confirmation, the booking is finalized. Once the driver accepts the request, live GPS tracking begins, allowing the passenger to monitor the driver’s approach and the trip’s progress in real time. At the end of the ride, the system updates the status to “completed,” and payment is processed. If GCash is used, the passenger uploads proof of payment within the app. Finally, the passenger rates the driver and provides feedback, which contributes to performance evaluations and aggregate ratings. In line with foundational ridesharing concepts that emphasize dynamic allocation and efficiency (Codecademy, n.d.), the passenger flow was designed to balance algorithmic optimization with user convenience.

**Driver Operational Flow**The driver flow begins when the driver logs into the application and accesses the Driver Dashboard. Drivers define their availability by creating and publishing a route, which involves selecting start and end points on the map. This approach is consistent with destination-oriented models in ridesharing literature, which highlight the benefits of drivers predefining travel corridors for optimized matches (Bhatia, 2020). The route is computed and stored as a polyline in the driver\_routes table using OSRM/Dijkstra’s algorithm, and is then displayed as an option for passengers.

As passengers request rides and snap their pickup and drop-off points to the driver’s route, the system evaluates the feasibility of the request. The driver receives notifications containing passenger details, pickup and drop-off locations, and the estimated fare. Comparable to Grab’s real-world implementation, this validation ensures requests align with feasible detours while maintaining system efficiency (Abeywickrama & Liang, 2021). The driver may accept or reject the request. Upon acceptance, the driver’s app updates the ride status and begins publishing their live GPS coordinates. Simultaneously, the driver subscribes to the passenger’s live location feed, enabling more efficient navigation to the pickup location.

During the trip, both the driver and passenger benefit from GIS-based route visualization, with live updates reflecting the driver’s movement, passenger location, and the intended route. Once the passenger is dropped off, the driver marks the ride as completed. If the passenger selected GCash as a payment method, the driver is notified of the uploaded payment proof. Finally, the driver rates the passenger, and the rating is added to the passenger’s profile, reinforcing accountability and trust within the system. As highlighted in ridesharing system overviews (Codecademy, n.d.), such rating mechanisms are critical for building trust and sustaining user adoption.

Thus, the operational framework of GoDavao: encompassing passenger and driver flows, is anchored on both literature (Bhatia, 2020) and industry practices (Abeywickrama and Liang, 2021; Codecademy, n.d.), ensuring that the system’s design is conceptually sound and practically relevant.

## **3.2 Tools, Technologies, and Reproducibility**

The development stack was selected for open-source availability and reproducibility.

**Flutter SDK (v3.24.3, stable channel)** The stable channel was selected to ensure long-term reliability and compatibility across devices. Version 3.24.3 was the latest stable release at the time of development, offering improved Material 3 support and optimized rendering for both Android and iOS. Using a stable version avoids regressions or experimental bugs that might occur on beta or master channels.

**Dart SDK (v3.4.4)** This version was the officially bundled release with Flutter 3.24.3, ensuring tight integration between the language and the framework. Dart 3.4.4 includes type-safety features, async improvements, and null-safety enforcement, all of which were critical for managing real-time data streams and preventing runtime crashes in the ride-matching logic.  
**Supabase (Flutter client v2.9.1 with PostgreSQL + Realtime backend)** Supabase was chosen for its managed PostgreSQL database with built-in authentication, storage, and Real-time subscriptions. The v2.9.1 client was the latest stable package that aligned with Flutter 3.24.x and supported advanced features such as row-level security (RLS), live location subscriptions, and UUID-based foreign key handling. PostgreSQL’s structured schema ensured data integrity, while Realtime channels enabled the core functionality of live driver–passenger tracking.

**OSRM (Open Source Routing Machine, v5.27.0)** OSRM was selected as the routing engine because it is a proven, open-source solution for real-time route planning and snapping passenger pickups/dropoffs onto driver routes. Version 5.27.0 was the stable Docker image at the time of implementation, ensuring consistent and reproducible routing results across both local development and remote server deployment. Its compatibility with GIS visualization in FlutterMap allowed seamless integration of Dijkstra’s algorithm and heuristic-based detour calculations.

**Flutter Dependencies:**

* cupertino\_icons: ^1.0.8
* supabase\_flutter: ^2.9.1
* flutter\_map: ^6.1.0
* latlong2: ^0.9.0
* geolocator: ^11.0.0
* image\_picker: ^1.1.2
* url\_launcher: ^6.3.2
* geocoding: ^2.1.0
* flutter\_patch\_package: ^0.0.11
* flutter\_polyline\_points: ^2.0.0
* google\_polyline\_algorithm: ^3.1.0
* flutter\_local\_notifications: ^19.4.0
* flutter\_background\_service: ^5.0.5
* permission\_handler: ^11.3.1
* flutter\_dotenv: ^5.2.1
* provider: ^6.1.5
* uuid: ^4.4.0
* http: ^1.4.0
* intl: ^0.20.2

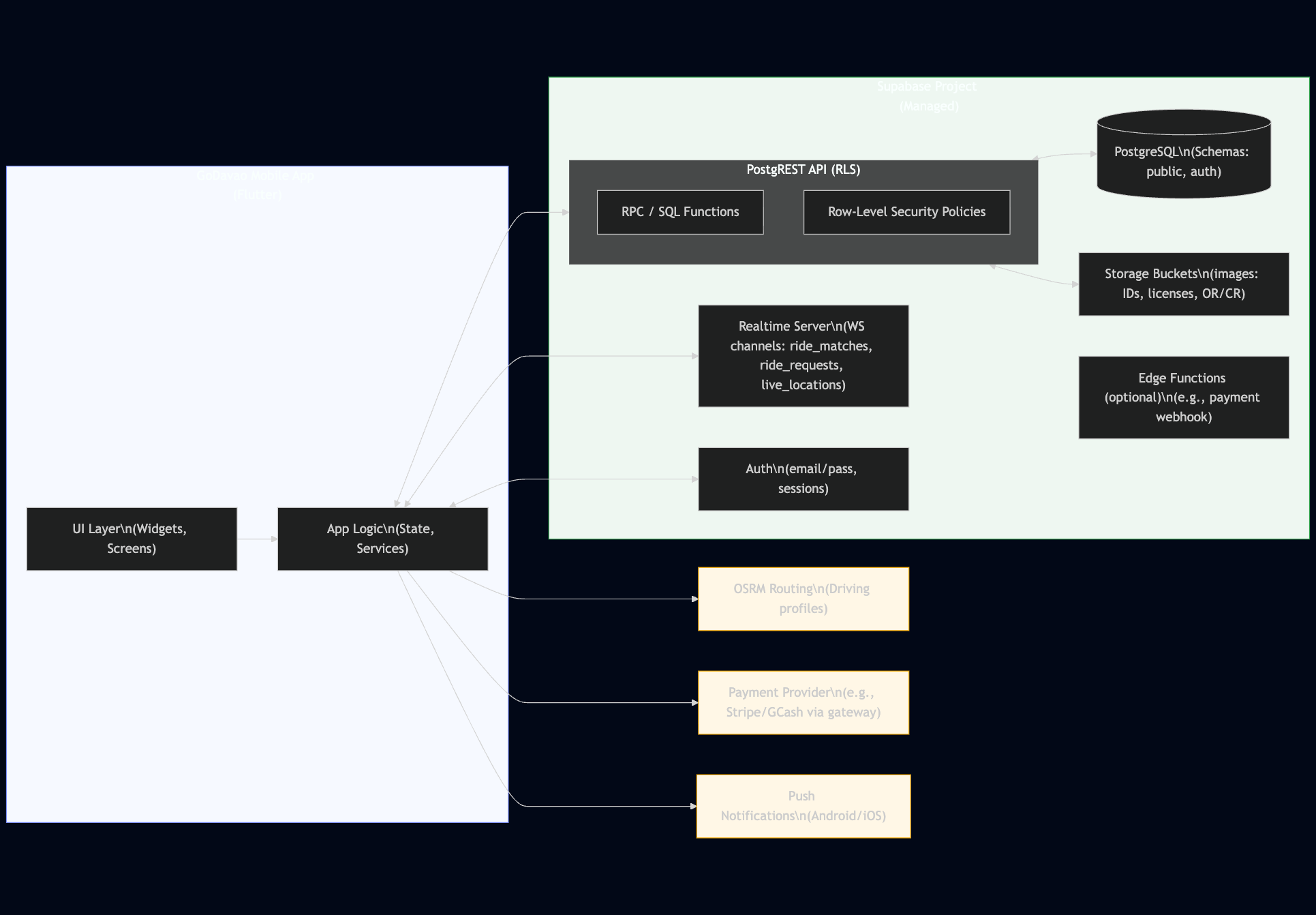
**Reproducibility Measures**

* **Environment Variables**: All sensitive keys and endpoints (Supabase URL and API key) are stored in .env files using the flutter\_dotenv package, ensuring both security and portability.
* **Platform Configurations**: The AndroidManifest.xml explicitly includes permissions for INTERNET, ACCESS\_FINE\_LOCATION, and ACCESS\_COARSE\_LOCATION. Cleartext traffic is enabled for development builds to allow OSRM (HTTP) requests. Runtime location permission handling is implemented with the geolocator package.
* **Toolchain Versions**: The system was developed using Flutter 3.22.2 (stable channel) with Dart 3.4.0. This version pairing is explicitly documented to avoid compatibility issues in future Flutter releases.
* **Dependency Locking**: A pubspec.lock file is included in the repository, ensuring all dependencies (e.g., supabase\_flutter 2.9.1, flutter\_map 6.1.0, geolocator 11.0.0) resolve to the exact same versions on every build.
* **Routing Engine**: OSRM is containerized for reproducibility, using the official osrm-backend Docker image (version 5.27.0). This ensures routing results remain consistent across environments.
* **Version Control & Iterations**: The project is version-controlled via GitHub.
* **Rebuild Instructions**: To reproduce the build, clone the repository, run flutter pub get, configure the .env file with Supabase credentials, then execute flutter run (for debug) or flutter build apk (for release).

## **3.3 System Development Documentation**

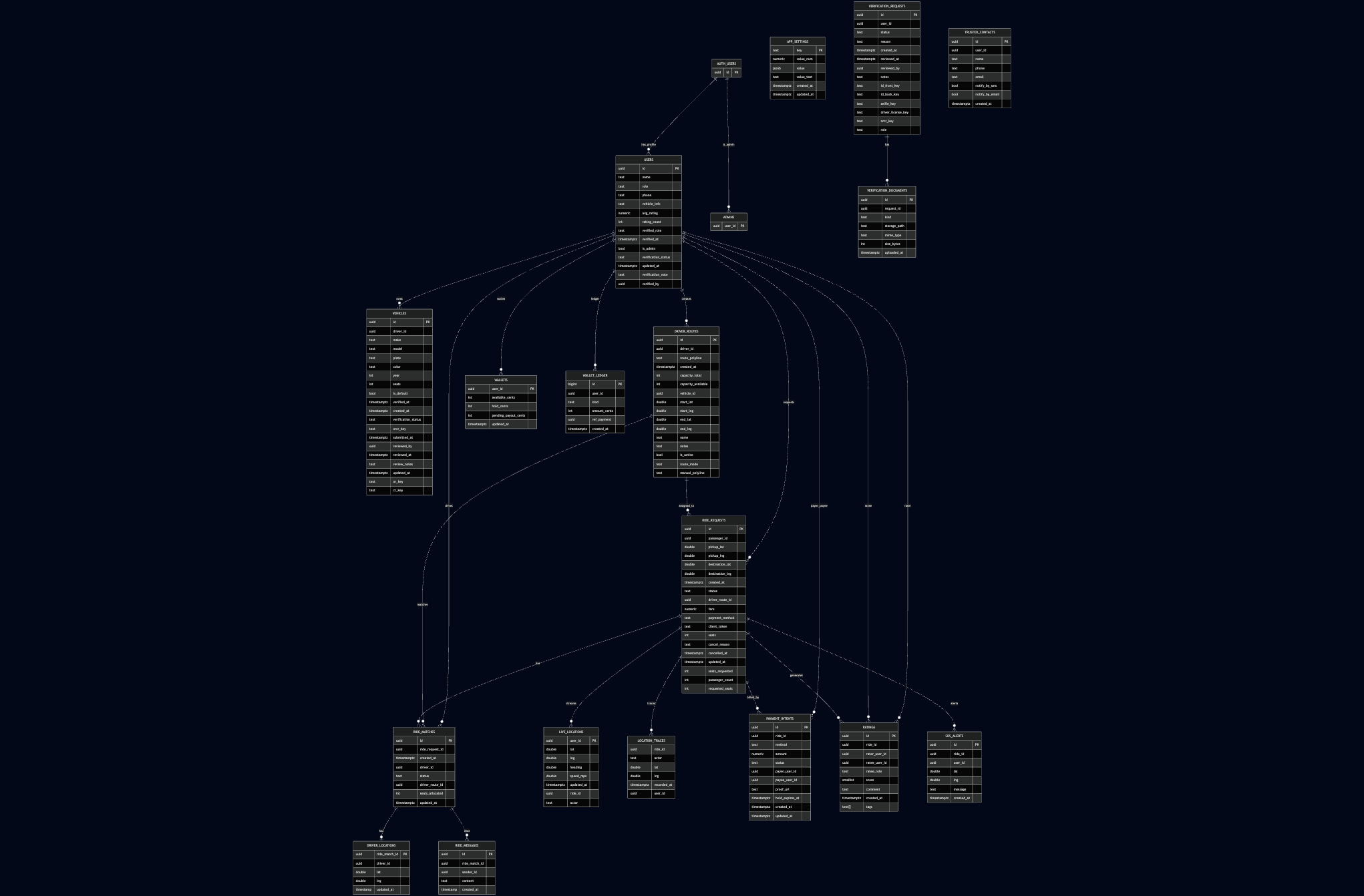
The system development documentation provides a comprehensive view of how the GoDavao platform was constructed and how its subsystems interconnect. This section is crucial as it bridges the gap between theoretical methodology and technical implementation.

The system architecture of GoDavao follows a three-tier model consisting of the mobile client, the Supabase backend, and the OSRM routing service. The mobile client (built in Flutter) provides the user interface for both drivers and passengers. The Supabase backend manages user authentication, real-time subscriptions, database storage, and secure payment records. The OSRM service handles route optimization, distance, and time estimation to support fare computation and efficient ride matching. The system also incorporates passive routing, allowing drivers to define fixed or habitual routes in advance. The system continuously monitors passenger requests along these predefined paths, enabling automatic ride matches without requiring the driver to re-route manually



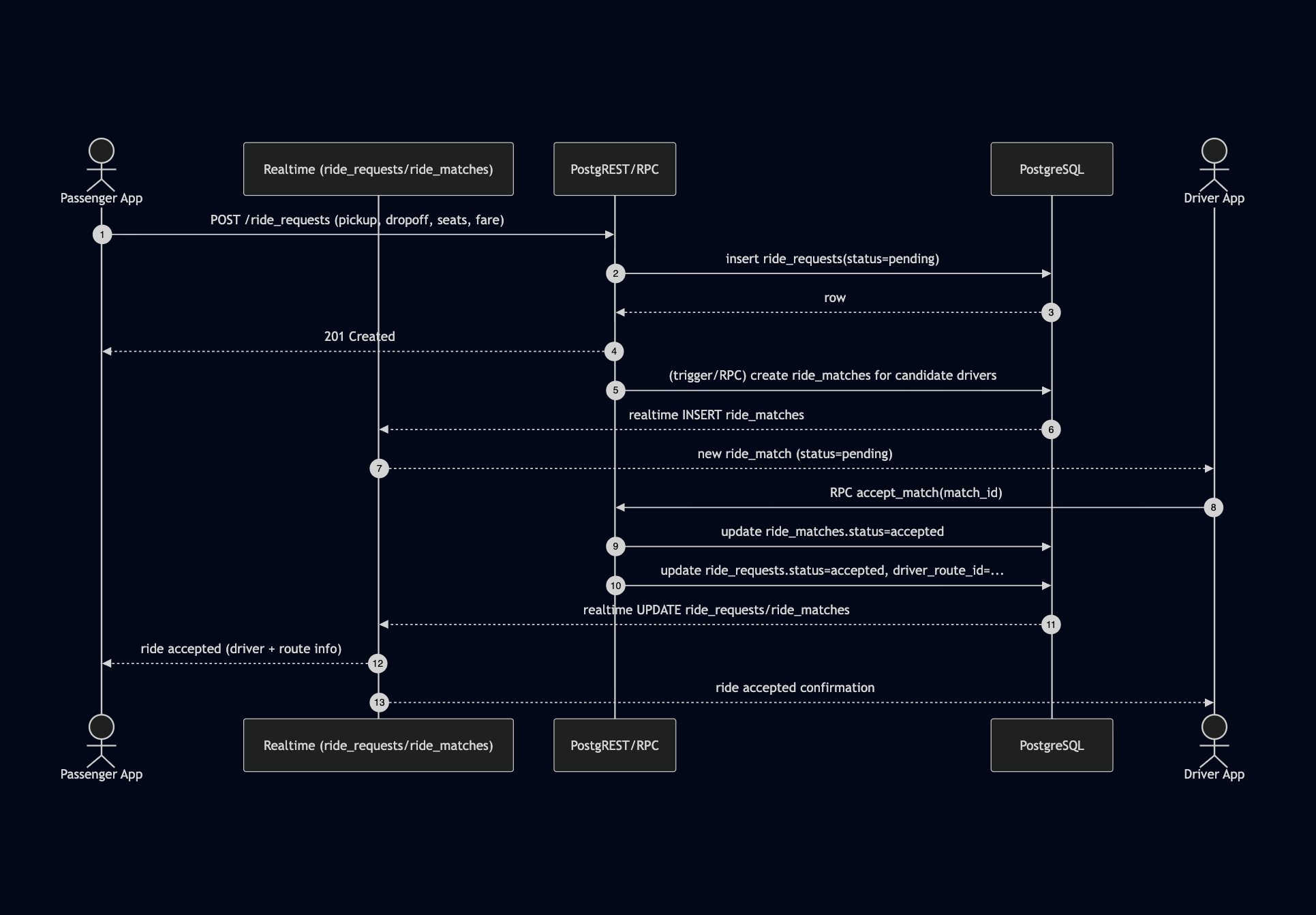
System Architecture

The database schema in Supabase is designed to maintain relationships between core entities such as users, ride requests, driver routes, ride matches, and live locations. The Entity-Relationship Diagram (ERD) visualizes these relationships and highlights primary and foreign keys that enforce referential integrity. For example, each ride\_request is linked to a passenger in the users table, while each driver\_route belongs to a verified driver. Ride matches create the link between these two flows, supported by live tracking for real-time monitoring.

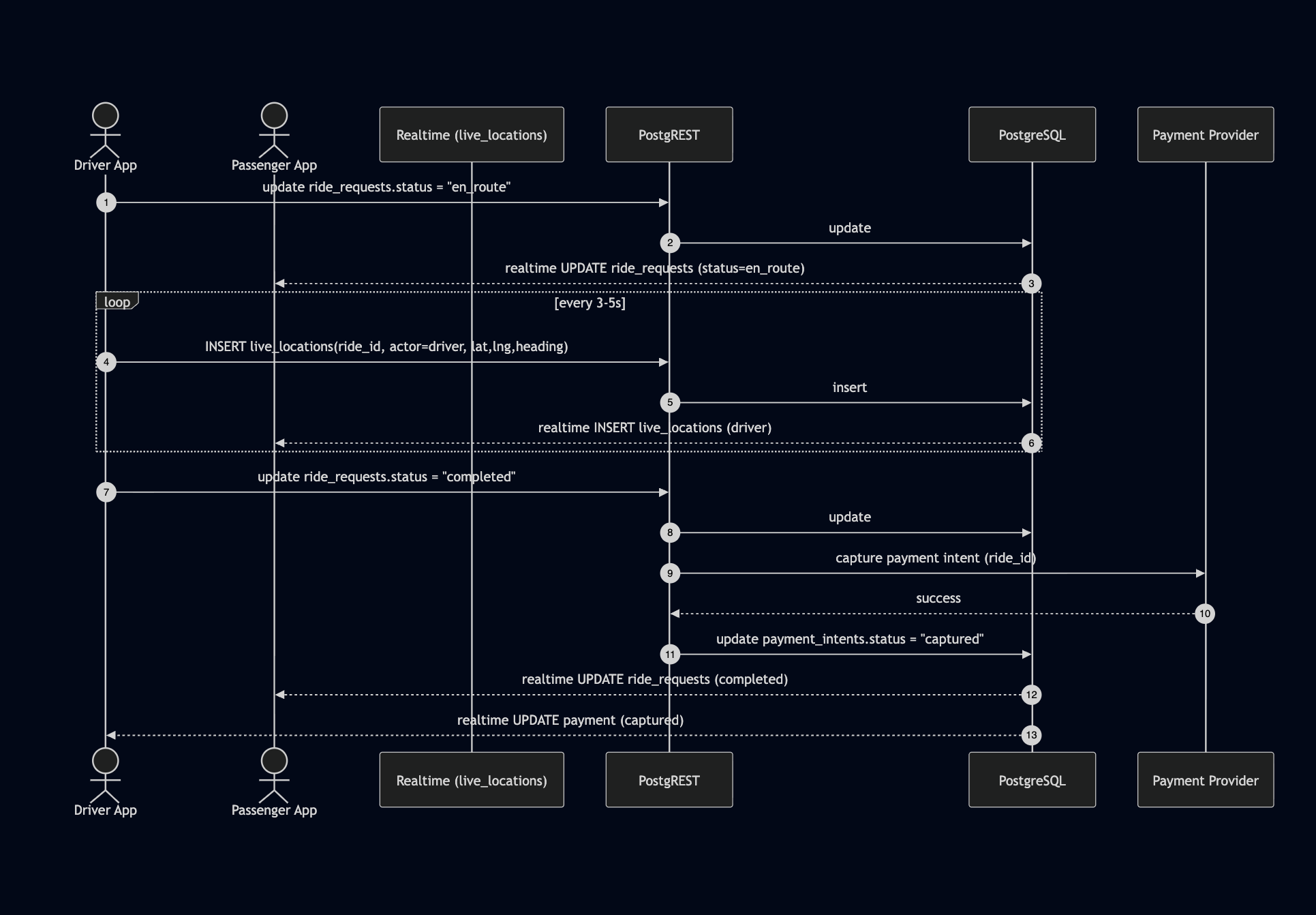


*Entity Relationship Diagram*

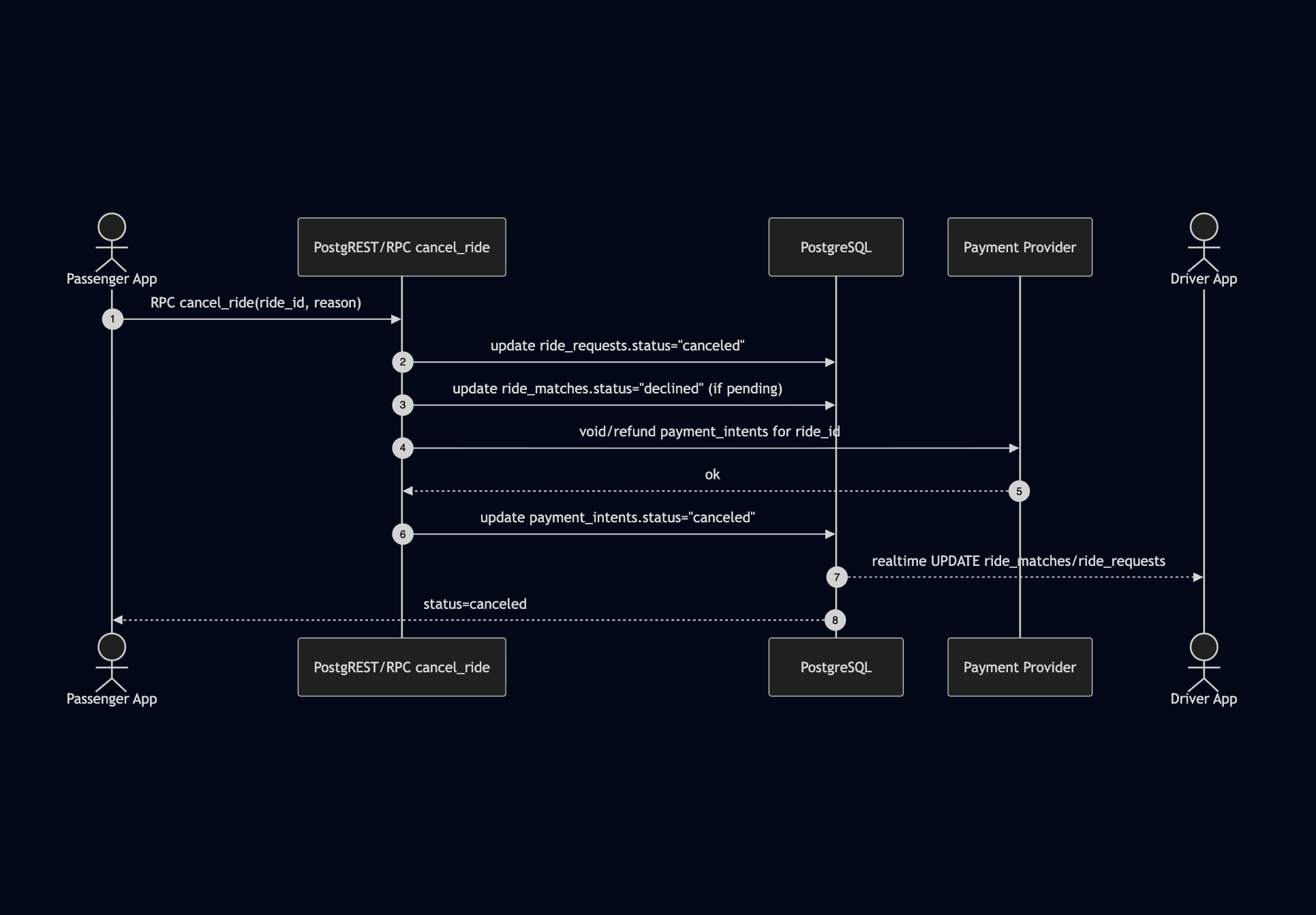
To further illustrate how the system functions in real time, sequence diagrams can be included to represent scenarios such as passenger ride requests, driver acceptance, and ride completion. These diagrams demonstrate how data flows across modules during a transaction, ensuring system transparency and reproducibility.



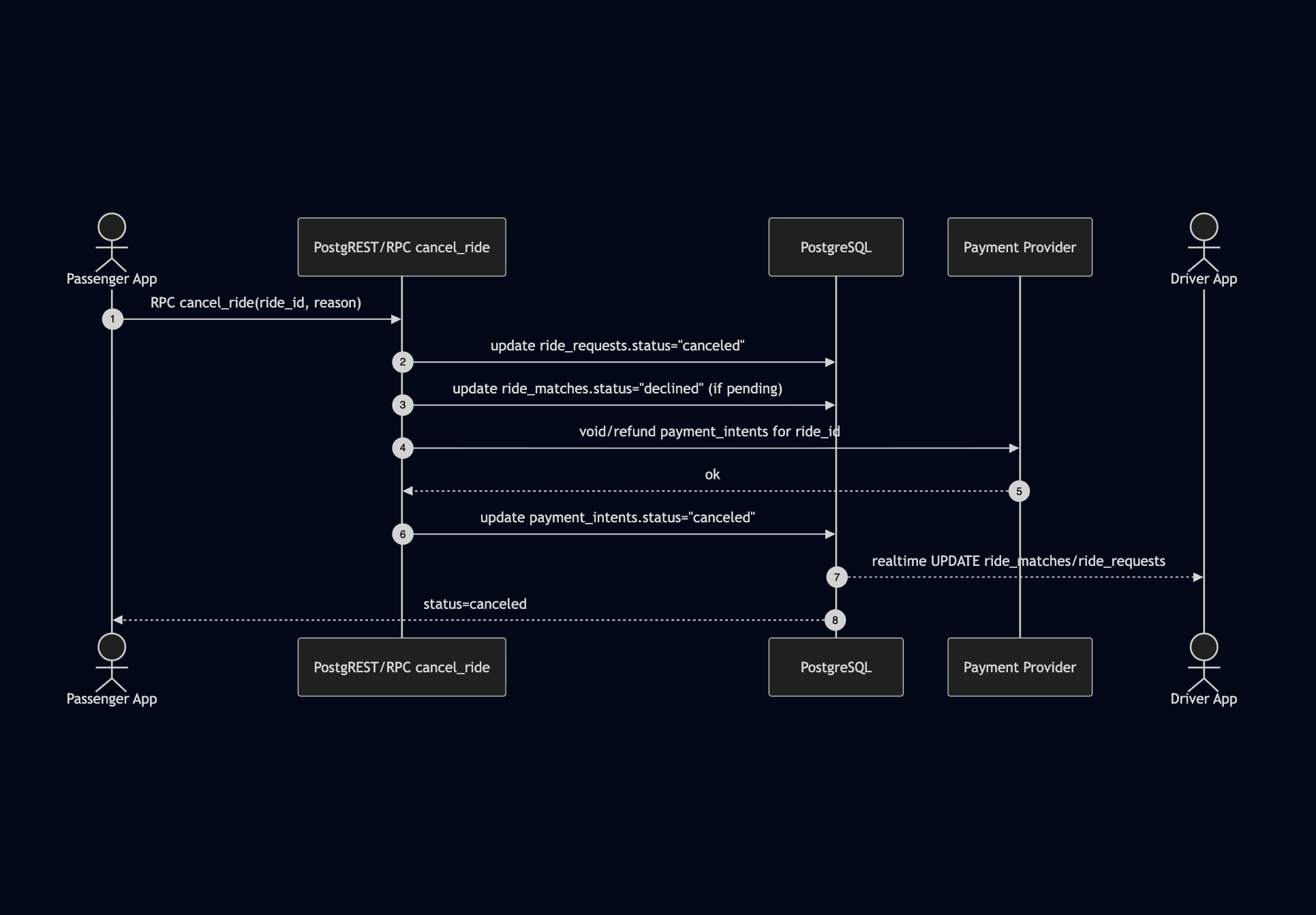
*Sequence Diagram 1 - Passenger books, driver matches, accepts*



*Sequence Diagram 2 - Live tracking and trip flow to completion and payment*



*Sequence Diagram 3 - Identity Verification flow*



*Sequence Diagram 4 - Passenger cancel, refund flow*

The system also incorporates payment management, live tracking, and safety features. Payment intents are created and updated in Supabase to reflect ride statuses, while GCash integration allows for proof-of-payment uploads. Live location tracking is facilitated by publishing and subscribing to driver and passenger coordinates in near real time, ensuring both parties can locate each other. Additionally, SOS features are integrated into the mobile app for emergency support.

## **3.4 UI/UX**

The user interface (UI) and user experience (UX) design of this application are intentionally structured to balance **visual appeal, usability, and accessibility**. Each element, color, button, typography, and layout has been selected to support usability principles and align with established design guidelines.

This section addresses **Research Objective 2**, which focuses on designing a user-friendly interface for commuters. The interface design decisions were later verified through usability surveys and UAT, confirming that users found the layout intuitive, consistent, and easy to navigate (see Appendix A and B).

### **3.4.1 Color Psychology and Branding**

The color scheme is central to the overall design:

* **Purple** is adopted as the **primary brand color**, reflecting creativity, reliability, and innovation. This establishes a unique visual identity while fostering user trust in the system (Cherry, 2023).
* **Orange** is used for **warnings or pending states** such as verification, as orange naturally draws attention and signals that user action may be required without being as alarming as red (Kaya & Epps, 2004).
* **Green** is applied to indicate **pickup or starting locations**, leveraging its psychological association with safety, progress, and "go" signals in navigation contexts (Wright, 1988).
* **Red** is reserved for **critical actions and destinations**, such as marking the end-point of a trip or for logout, aligning with its universal meaning of stop, danger, or caution (Elliot & Maier, 2014).

This deliberate color coding ensures users can **instantly interpret status and actions** based on familiar color associations.

### **3.4.2 Buttons and Action Hierarchy**

* **Primary actions** (e.g., Cancel Ride, Contact Driver) use **gradient purple buttons** to stand out as the most important interactions.
* **Secondary actions** (e.g., View Details) are displayed as text buttons, reducing visual clutter while still being discoverable.
* This follows **Fitts’s Law**, where critical actions must be larger and more visually prominent to reduce the time required to locate and tap them (Fitts, 1954).

### **3.4.3 Layout and Information Organization**

* Information is grouped into **cards** (rides, history, upcoming) for modular scanning, following principles of **Gestalt psychology** where grouping improves comprehension (Wertheimer, 1938).
* Adequate padding ensures that interactive elements meet **Material Design’s 48x48dp touch target recommendation** (Google, 2022), preventing mis-taps on mobile devices.
* Capsule-shaped tags (e.g., DRIVER, PASSENGER) are compact and color-contrasted, making roles easy to recognize at a glance.

### **3.4.4 Visual Feedback and Navigation Aids**

* **Mini-map previews** are included in ride cards, supporting cognitive mapping and reducing reliance on text-only directions.
* **Status pills** are color-coded (e.g., green for completed, orange for en route, red for canceled) to provide immediate visual recognition of progress and outcomes.
* **Refresh indicators** give real-time feedback, enhancing trust that data is current.

### **3.4.5 User-Centered Design**

Overall, the system applies **Nielsen’s usability heuristics** (Nielsen, 1995):

* Visibility of system status → through status pills, refresh indicators, and maps
* Consistency and standards → by using conventional colors for start, stop, and warnings.  
  Aesthetic and minimalist design → by reducing unnecessary UI elements while retaining clarity.

This ensures that the application is not only visually aligned with its brand but also **functional, intuitive, and accessible**.

## **3.5 Agile Development Strategy**

To implement the DSR design in practice, the team adopted Agile Scrum methodology, dividing development into four sprints. Each sprint represented a DSR “design–build–evaluate” iteration, ensuring continuous refinement based on feedback and errors encountered.

### **Sprint 1: Authentication & User Roles**

**Linked Objectives:**

* **RO2:** Design and develop a user-friendly ridesharing MVP (GoDavao).
* **RO4:** Implement secure verification and account management features.

**Summary:**Before commuters can book or offer rides, they need secure access. This sprint focused on developing the authentication foundation of the GoDavao app, ensuring that users could log in, register, and access dashboards specific to their roles (Passenger or Driver). This directly supports onboarding usability and system security, establishing the groundwork for subsequent features.

**Technologies Implemented:**

* **Supabase Auth (v2.9.1 Flutter client):** For email/password authentication and metadata storage.
* **Supabase Postgres schema:** Profiles table linked to auth.users via UUIDs for role storage.
* **Provider (v6.1.5):** Managed authentication state and reactive UI updates.
* **Flutter SDK (3.24.3, Material 3):** Built login, signup, and dashboard navigation interfaces.

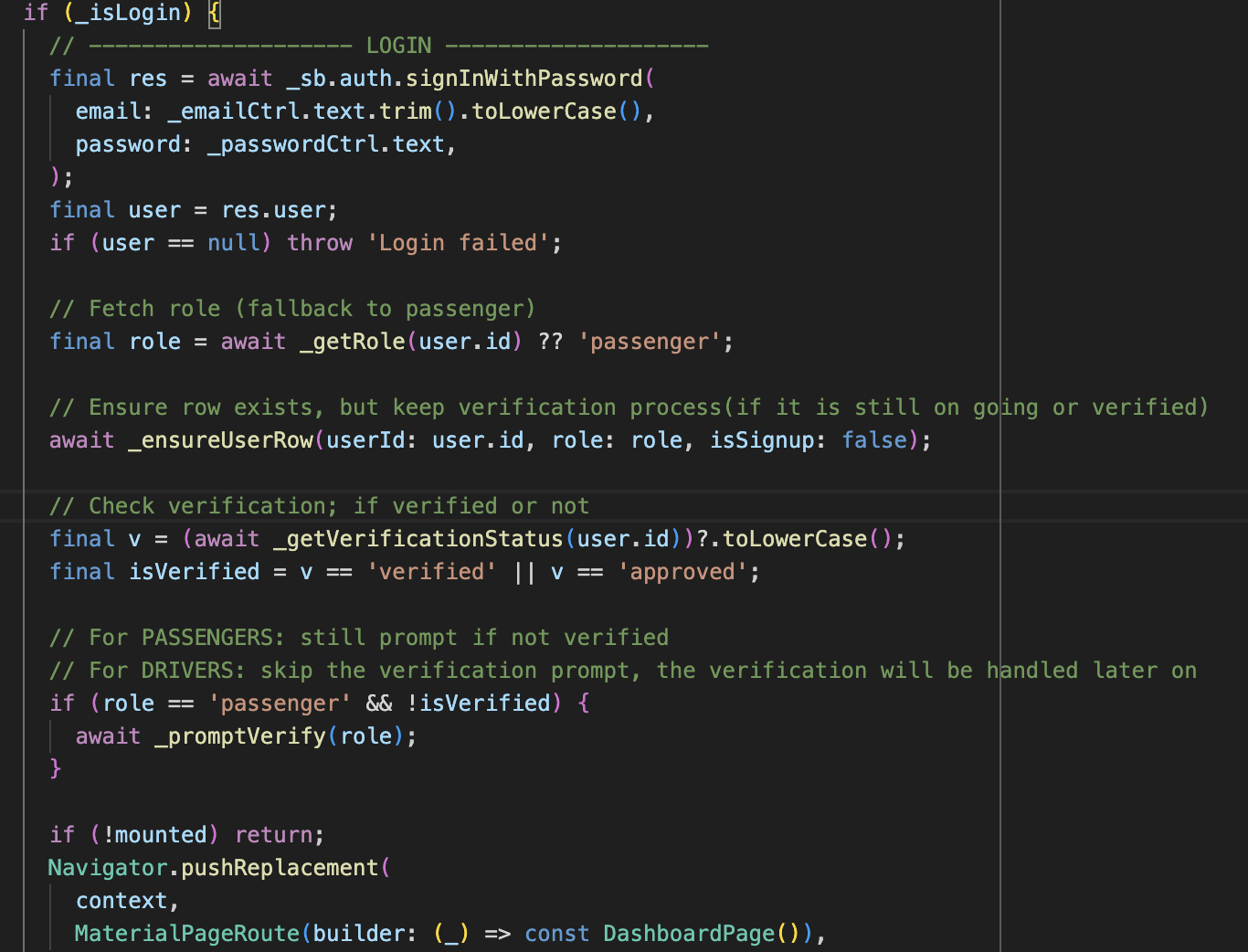
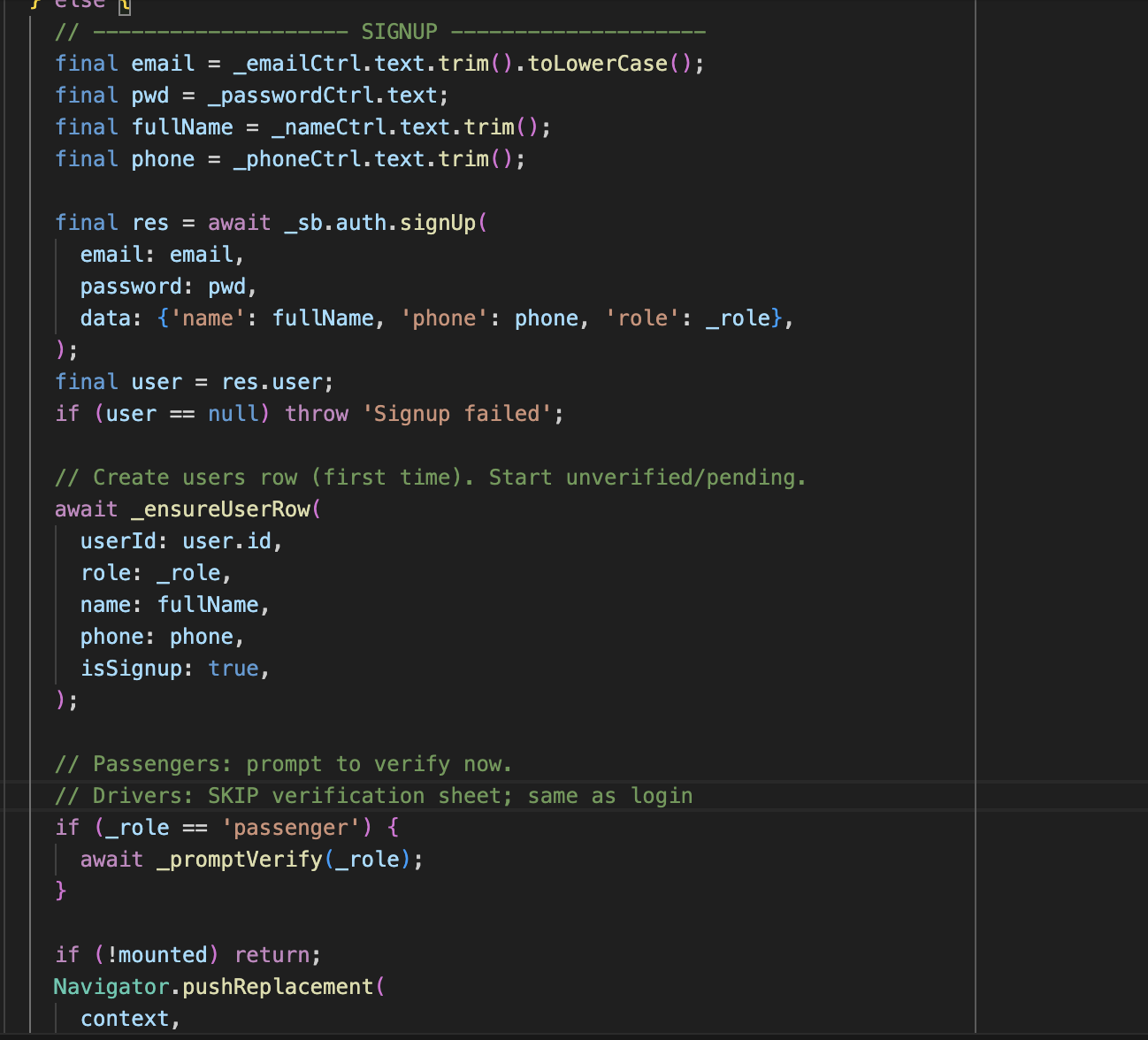
**Activities:**

* Integrated Supabase Auth for user sign-up, login, and logout.
* Designed registration flow with **role selection**, storing roles in the profiles table via UUID keys.
* Implemented session persistence with supabase.auth.recoverSession().
* Built **role-based dashboards:** Passenger > PassengerMapPage; Driver > DriverDashboard.

**Issues Encountered:**

* Session invalidation after app reload.
* Metadata parsing errors (null/malformed JSON).

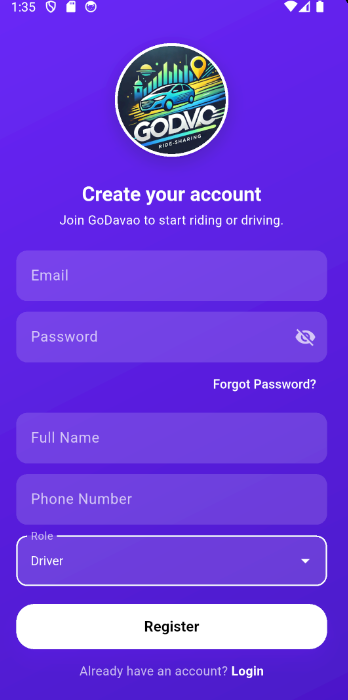
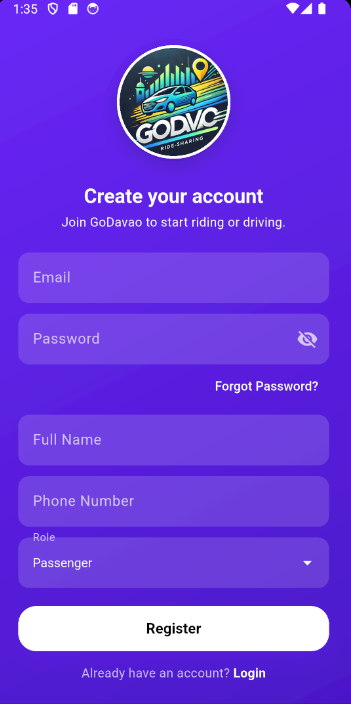
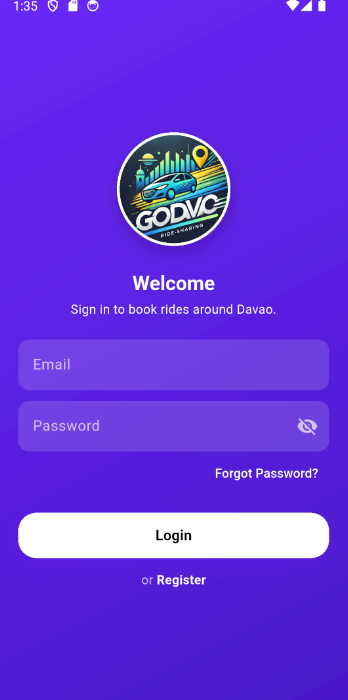
**Relevant Code:**

* 
* Login Logic
  + Input the email and password > checks if the user is verified or not > if not, proceed with verification sheet > if verified, go to the dashboard of the respective role
* 
* Register Logic
  + Register the name, contact, email, and password.
  + Input the role:
    - If passenger sends directly to the verification sheet, then the dashboard waits for verification
    - If driver sends directly for license verification, then sends it to the dashboard for vehicle registration.

**Resolution:**

* Used auth.recoverSession() for persistent sessions.
* Moved role storage to the profiles table for reliable metadata parsing.

**Output:**

* A functional role-based authentication system serves as the foundation for all subsequent flows.
* 

#### **Login Screen**

* + - The first interface presents the login screen, where users can enter their registered email and password to access their account. It includes standard authentication features such as:
    - “Forgot Password?” link for account recovery.
    - “Register” navigation for new users.
    - A minimalist, brand-aligned design featuring the GoDavao logo and the app’s purple color palette for consistency.

#### **Registration Screen – Passenger Role**

* + - The second screenshot displays the **registration form** for new users, allowing them to input their **full name**, **phone number**, **email**, and **password**.  
       A dropdown menu lets users choose their **role**, in this case, *Passenger*. The system then stores this metadata in the Supabase profiles table linked to the auth.users table.

#### **Registration Screen – Driver Role**

* + - The third screenshot shows the same registration interface, but with the **Driver** role selected. This role assignment determines access to driver-specific dashboards and functionalities such as route publishing, ride management, and passenger matching.

### **Sprint 2: Ride Requests & Driver Routes**

**Linked Objectives:**

* **RO2:** Develop the GoDavao MVP mobile app with driver and passenger modules.
* **RO3:** Integrate ride-matching and routing logic (GIS-based).
* **RO1:** Address transportation challenges by supporting booking and ride-offering flows.

**Summary:**This sprint bridges user needs (commuters requesting rides) with system capabilities (drivers publishing routes), responding to mobility issues identified in Davao City. It established the functional connection between passenger ride requests and driver route publication.

**Technologies Implemented:**

* **Supabase Schema:** Tables ride\_requests and driver\_routes linked via UUID foreign keys.
* **flutter\_map (v6.1.0)** + **latlong2 (v0.9.0):** For route rendering and marker placement.
* **flutter\_polyline\_points (v2.0.0):** For polyline decoding and storage.
* **OSRM Backend (v5.27.0 Docker):** Generated driver route polylines.

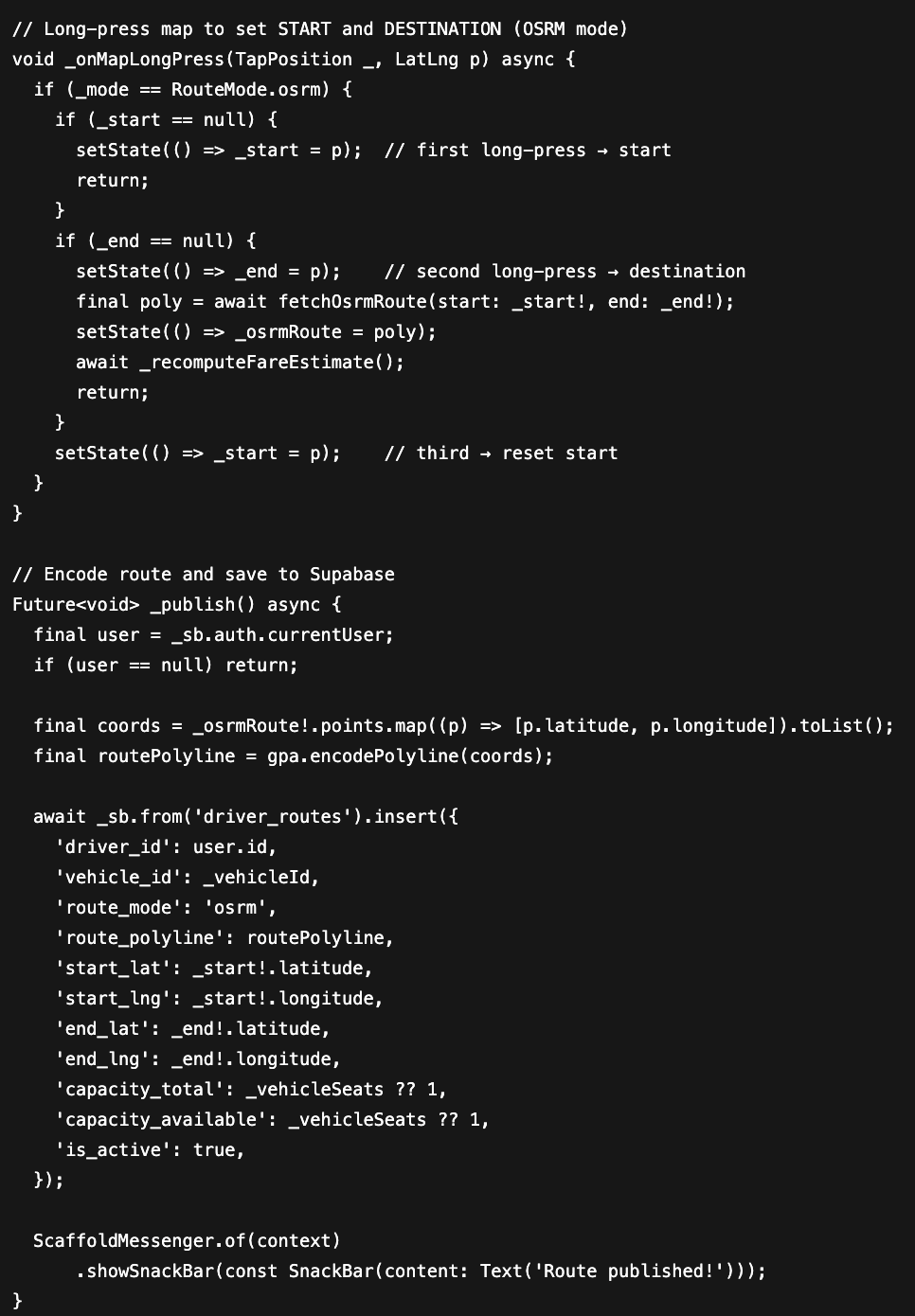
**Activities:**

* Created **Passenger Map Page** for selecting pickup and drop-off points.
* Built **Driver Route Page** for defining and publishing start–end routes.
* Stored generated polylines in Supabase for later route snapping.

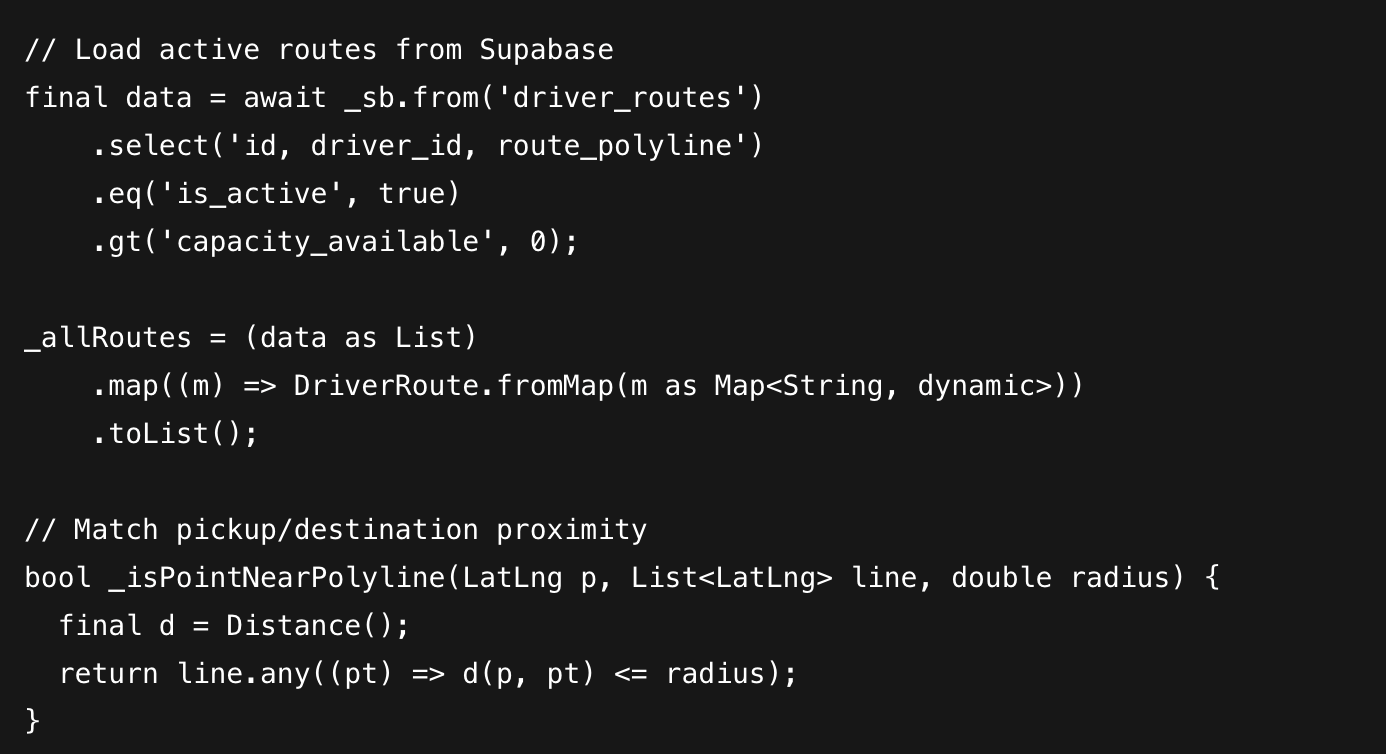
**Issues Encountered:**

* UUID mismatch between ride\_requests and driver\_routes.
* Supabase returned nested relationships as lists, causing runtime crashes.

**Relevant Code:**

**Driver Route Logic:**Let drivers create and publish their active travel routes using OSRM (Open Source Routing Machine) or manual mapping.

1. The driver long-presses on the map to mark a Start and Destination.
2. App fetches an OSRM-generated polyline and estimates fare via \_fareService.
3. Upon publishing, route data (start/end coordinates, encoded polyline, vehicle ID, seat capacity) is inserted into the driver\_routes table in Supabase.
4. Each route becomes active and visible to passengers searching nearby routes.

****

**Ride Request Logic:**

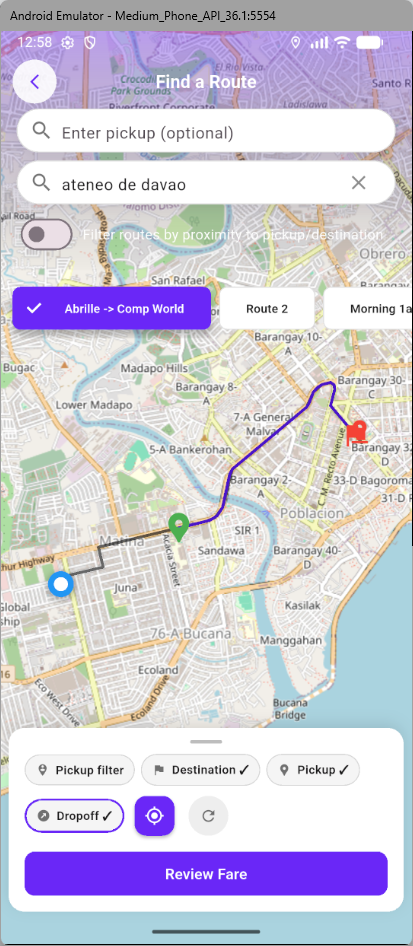
Let passengers locate and request rides along nearby driver routes.

1. Passenger inputs pickup and destination (geocoded via geocoding API).
2. App filters driver\_routes whose polylines fall within 600 m proximity of both points.
3. Selected route preview appears on map; passenger taps “Review Fare” to proceed to confirmation.
4. On confirmation, ride data is stored in ride\_requests table, linked via route\_id and driver\_id.

**Resolution:**

* Standardized **UUIDv4** across relational keys.
* Adopted .maybeSingle() queries for safe row retrieval.

**Output:**

* A stable ride request workflow and published driver routes visible to passengers.
  + 

#### **Passenger Route Selection Interface**

* + - The first screenshot displays the Find a Route screen, where passengers can input their pickup and drop-off locations, ex, Ateneo de Davao University to visualize potential travel paths.
    - Key interface elements include:
      * Search fields for specifying pickup and destination points.
      * Map visualization showing dynamically computed route options.
      * Filters for pickup/drop-off and trip type (e.g., morning, evening).
      * A “Review Fare” button that calculates the estimated ride cost before booking.
    - This view ensures a user-friendly experience by allowing passengers to verify their chosen route and fare before confirming the ride request.

#### **Driver Route Creation Interface**

* + - The second screenshot shows the Create Driver Route screen. This allows drivers to define their daily operating routes interactively on the map.  
       Core features include:
    - Integration with the OSRM API for automatic path generation and distance calculation.
    - Estimated fare computation, displaying both the Passenger Fare and the Driver Net Fare after GoDavao’s platform commission.
    - A “Publish Route” button, enabling drivers to make their route visible for matching with passenger ride requests in real time.
    - A Fare Breakdown section showing how the total fare is derived (base fare + distance × time).

### **Sprint 3: Ride Matching & Routing**

**Linked Objectives:**

* **RO3:** Integrate ride-matching and routing algorithms (Dijkstra + heuristics).
* **RO1:** Address transportation inefficiencies through shared rides.
* **RO2:** Enhance GoDavao MVP functionality with algorithmic optimization.

**Summary:**This sprint implemented the **core matching logic** that defines GoDavao’s innovation: connecting passengers to drivers with overlapping routes using **heuristic matching** and **Dijkstra’s Algorithm**. The goal was to create an efficient and fair detour calculation system.

**Technologies Implemented:**

* **OSRM (v5.27.0 Docker):** For snapping pickup/drop-off points to driver routes.
* **Dijkstra’s Algorithm:** Backend path validation for shortest detours.
* **flutter\_map + PolylineLayer:** GIS visualization of routes and pickups.
* **flutter\_dotenv (v5.2.1):** Managed OSRM environment configuration.

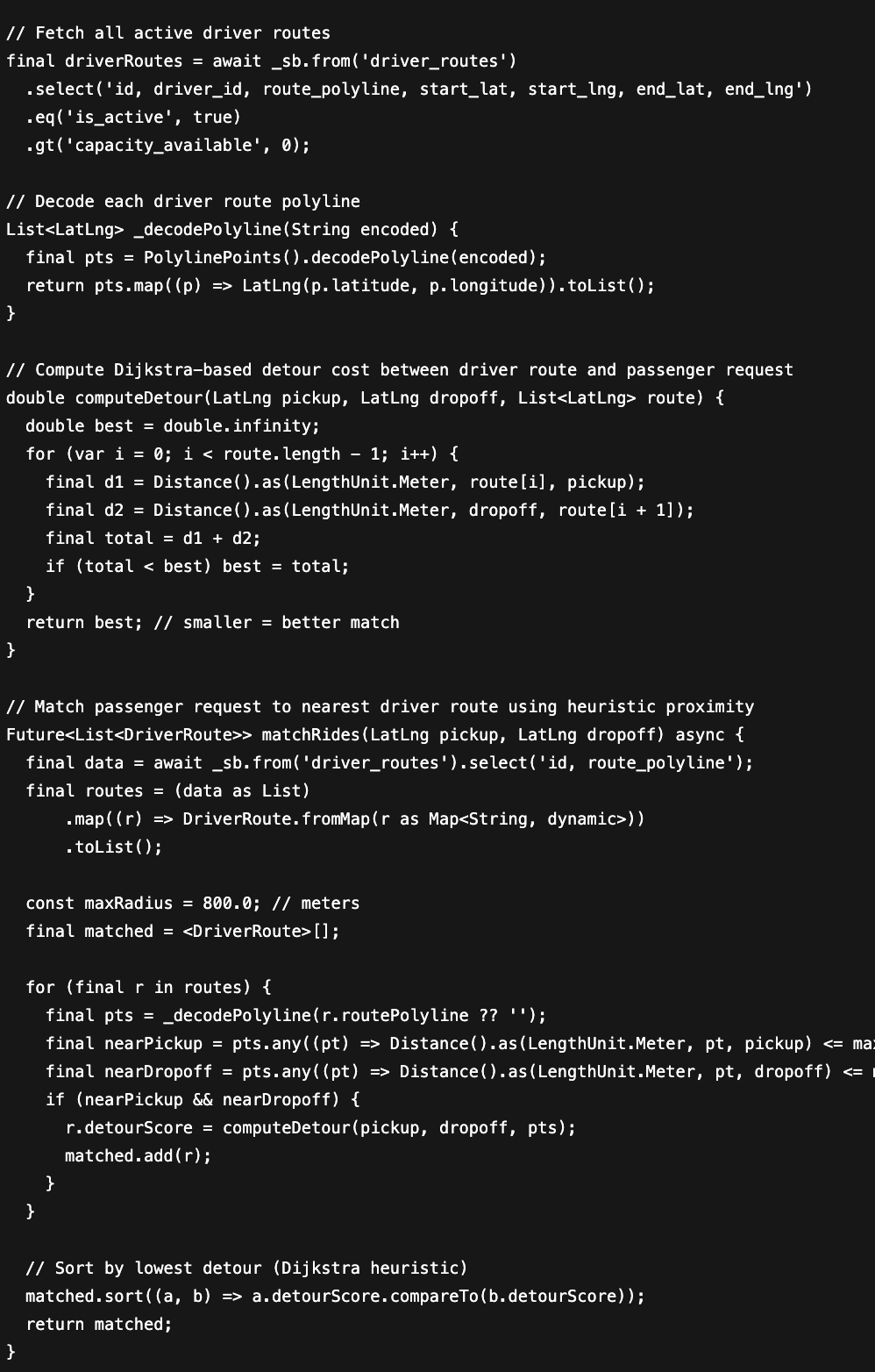
**Activities:**

* Integrated OSRM API for route snapping.
* Implemented **Dijkstra’s Algorithm** for detour cost evaluation.
* Displayed driver polylines and pickup markers on FlutterMap.
* Developed backend heuristics to filter ride requests within proximity of published routes.

**Issues Encountered:**

* Emulator vs. device network differences (localhost vs. 10.0.2.2).
* Android blocked cleartext HTTP requests.
* Long polyline rendering caused UI lag.

**Relevant Code:**

****

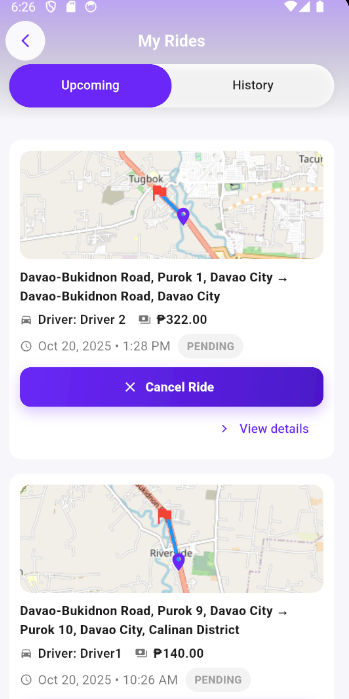
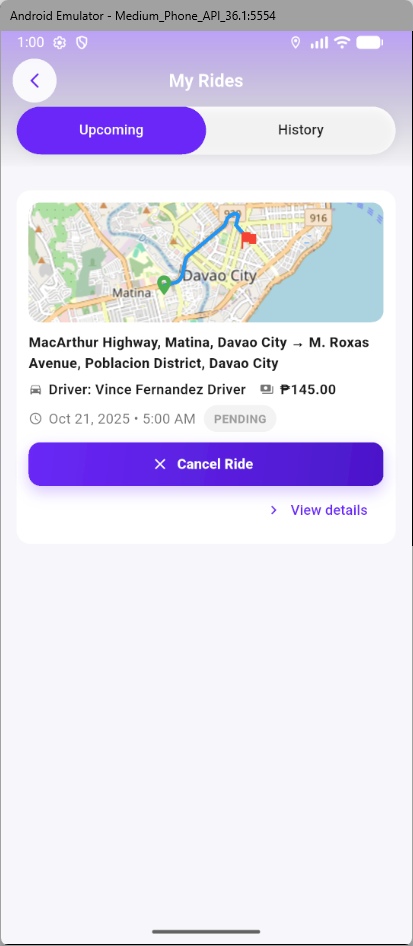
**Ride Matching Logic:**Automatically connect passengers to drivers whose published routes overlap or minimally deviate from the requested pickup/drop-off path.

1. Decode every driver’s encoded polyline from driver\_routes.
2. Snap passenger pickup/drop-off to the nearest route points using OSRM.
3. Apply a heuristic proximity filter (≤ 800 m) to find possible matches.
4. Compute detour cost via Dijkstra’s Algorithm, estimating added distance if the driver accepts the ride.
5. Rank results by lowest detour score, presenting the most efficient matches first.

**Resolution:**

* Updated environment configs (10.0.2.2 for emulator, external IP for production).
* Enabled <application android:usesCleartextTraffic="true">.
* Used **PolylinePoints decoding** for optimized rendering.

**Output:**

* A working matching prototype, showing passengers valid driver routes and computing detours.
* End-to-end Passenger > Route Selection > Driver Match > Map Visualization achieved.
* 

#### **Upcoming Rides Tab**

* + - The interface displays rides that are scheduled or pending, grouped under the “Upcoming” tab. Each ride card contains:
    - A map preview generated from stored route coordinates, showing the origin and destination path.
    - Pickup and drop-off addresses
    - Driver details (name and role) for passenger awareness and safety.
    - The fare amount and ride status (e.g., *Pending, Ongoing, or Completed*).
    - Action buttons such as “Cancel Ride” and “View Details” for user control and transparency.

#### **Ride History Tab**

* + - The “History” tab stores previously completed trips, allowing users to review travel details, fare breakdowns, and driver information. This supports post-ride interactions such as feedback, rating, or dispute resolution.

### **Sprint 4: Live Tracking & Ride Status Flow**

**Linked Objectives:**

* **RO3:** Integrate real-time ride tracking and status updates.
* **RO4:** Implement safety and communication mechanisms.
* **RO5:** Evaluate usability, performance, and user acceptance.

**Summary:**This sprint finalized the GoDavao MVP’s end-to-end functionality through **live location tracking**, **real-time ride status updates**, and **SOS safety features**. It also enabled usability testing to assess system responsiveness and user satisfaction.

**Technologies Implemented:**

* **Supabase Realtime Channels:** For streaming live driver–passenger updates.
* **Supabase Tables:** live\_locations and location\_traces (with Row-Level Security).
* **Geolocator (v11.0.0):** For GPS capture and runtime permission management.
* **flutter\_background\_service (v5.0.5):** Supported continuous driver tracking.
* **flutter\_local\_notifications (v19.4.0):** Provided real-time alerts.

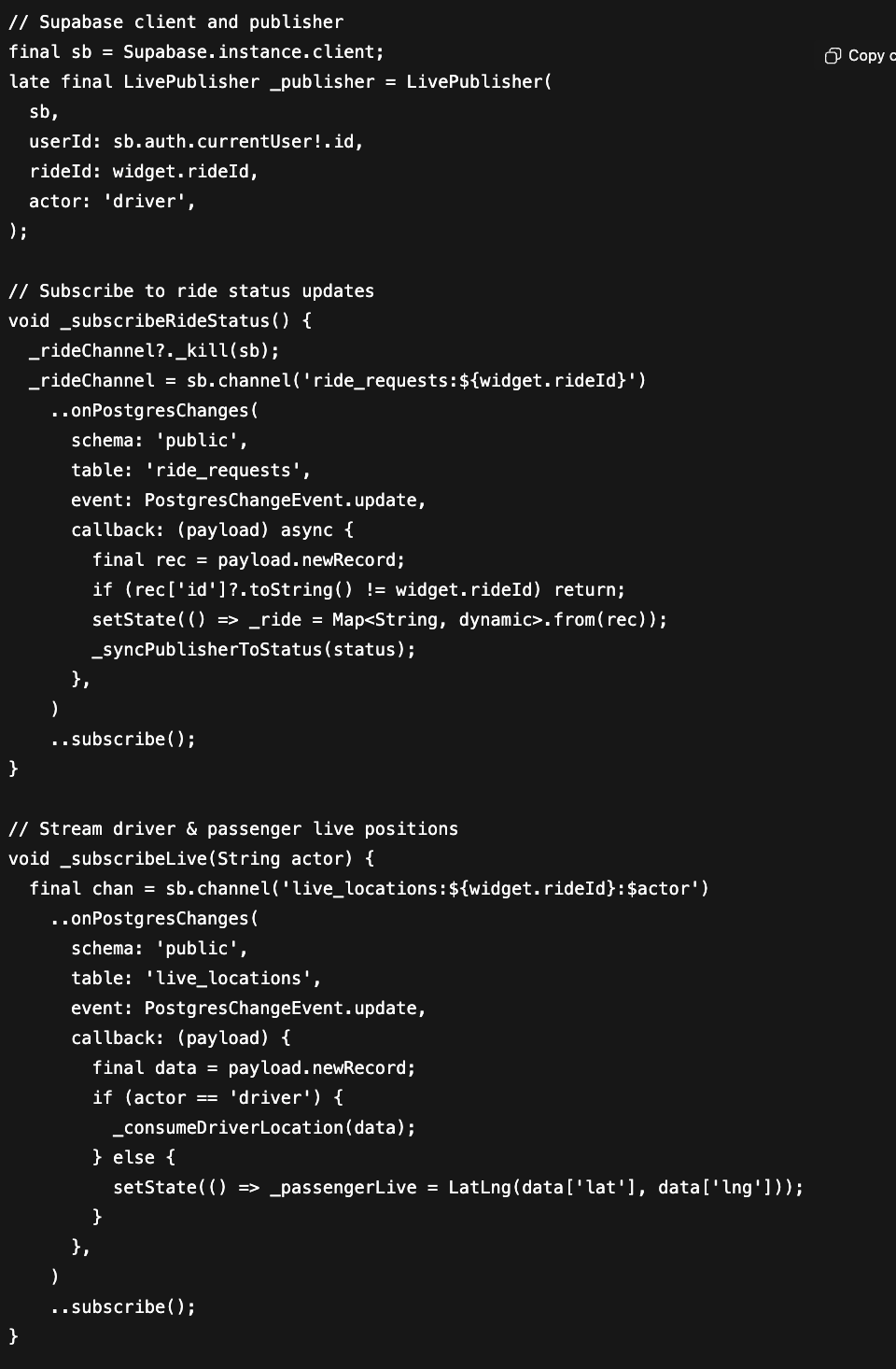
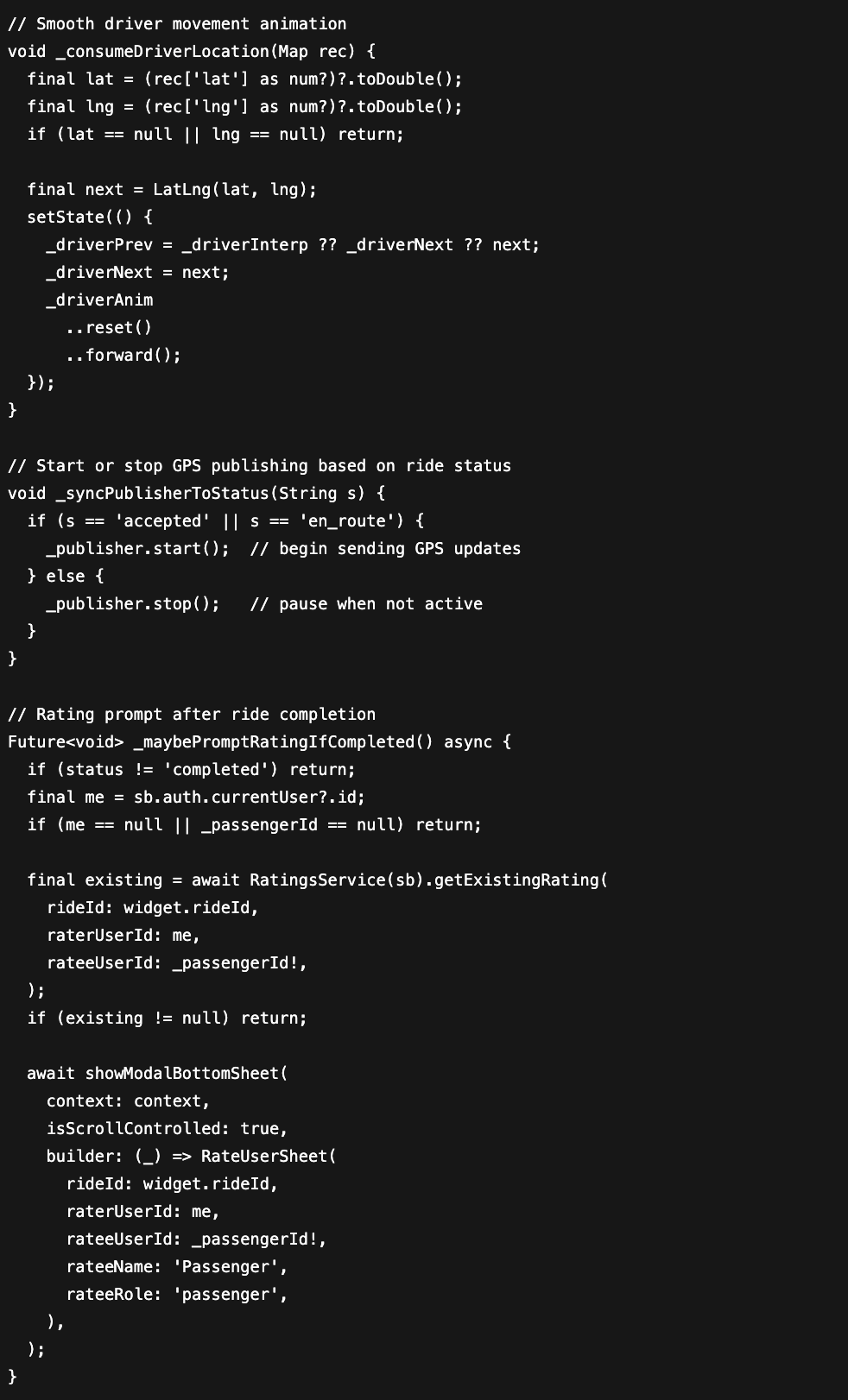
**Activities:**

* Created **LivePublisher** for publishing GPS updates to live\_locations.
* Subscribed passengers and drivers to **Realtime location streams**.
* Developed PassengerRideStatusPage and DriverRideStatusPage with live map markers, SOS buttons, and ride status flows (Accept → Start → Complete).

**Issues Encountered:**

* RLS blocking inserts into live\_locations.
* setState() called after widget disposal.
* Android not prompting for runtime permissions.

**Relevant Code:**

****

**Live Tracking Logic**

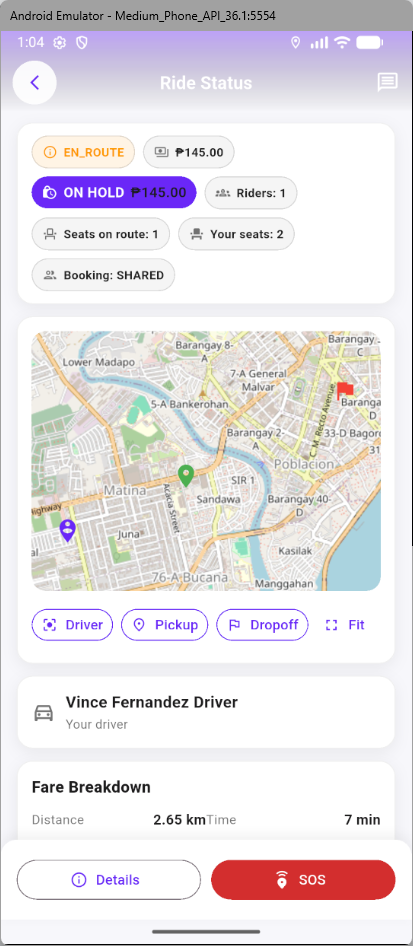
Enable two-way real-time tracking between drivers and passengers for improved visibility, coordination, and safety.

1. Driver Side: Publishes live GPS coordinates to the live\_locations table using Supabase Realtime.
2. Passenger Side: Subscribes to live driver updates and visualizes their movement on FlutterMap.
3. Ride Lifecycle: pending → accepted → en\_route → completed
4. Safety: SOS feature triggers an immediate local notification alert, enabling emergency communication and location sharing.

**Resolution:**

* Updated Supabase policies (auth.uid() = user\_id) scoped by ride\_id.
* Guarded state updates with if (mounted) checks.
* Implemented **runtime permission logic** using Geolocator.requestPermission().

**Output:**

* A fully functional real-time live tracking system.
* Drivers and passengers could both see each other in real time.
* End-to-end ride lifecycle flow (pending > accepted > en route> completed) achieved.
* 

#### **Ride Information and Status**

* + - At the top of the interface, the current ride status is displayed using clear visual labels such as *EN ROUTE*, *ON HOLD*, or *COMPLETED*.  
       Below this, the passenger can see:
    - The total fare amount and seat-sharing information.
    - The number of active riders (for shared or carpooling rides).
    - The passenger’s allocated seat and booking status.  
       These data points are automatically updated through Supabase real-time subscriptions linked to the ride\_requests and ride\_matches tables.

#### **Live Map Tracking**

* + - The map section provides an interactive visualization of the driver’s route, rendered via MapLibre GL and updated using live location data.  
       Color-coded markers represent:
    - Driver position
    - Pickup and drop-off points
    - Passenger route path
    - This integration allows passengers to monitor ride progress accurately, ensuring safety and transparency during the trip.

#### **Driver Details and Fare Breakdown**

* + - The lower section displays driver information (e.g., *Vince Fernandez – Driver*) along with a fare breakdown based on distance and estimated travel time.  
       The formula used is:
    - Fare = Base + (Distance × Rate per km) + (Duration × Rate per min)  
       This computation dynamically updates based on route recalculations or real-time distance tracking.

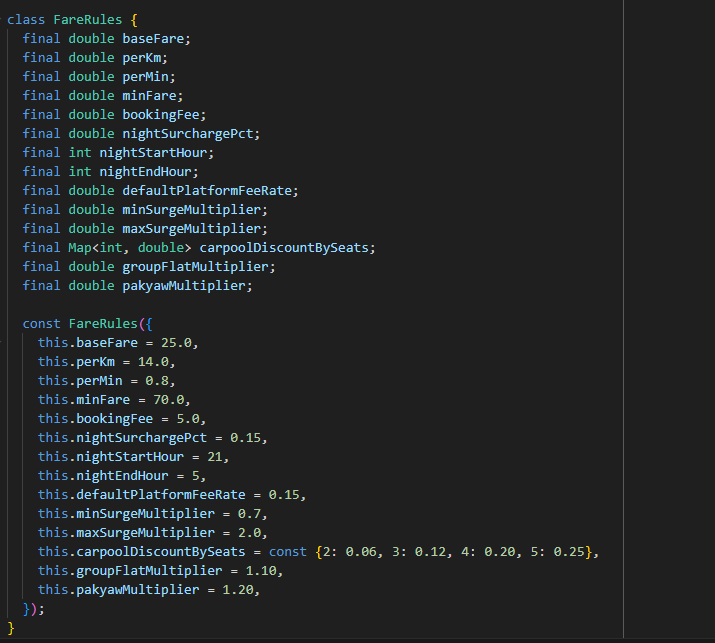
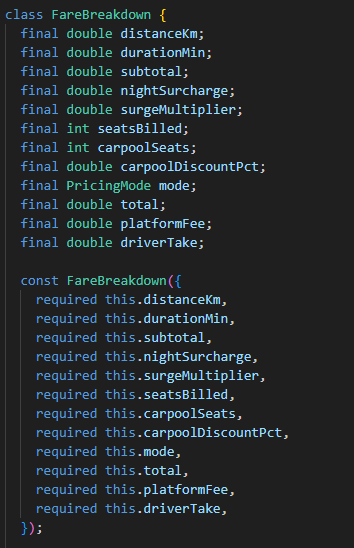
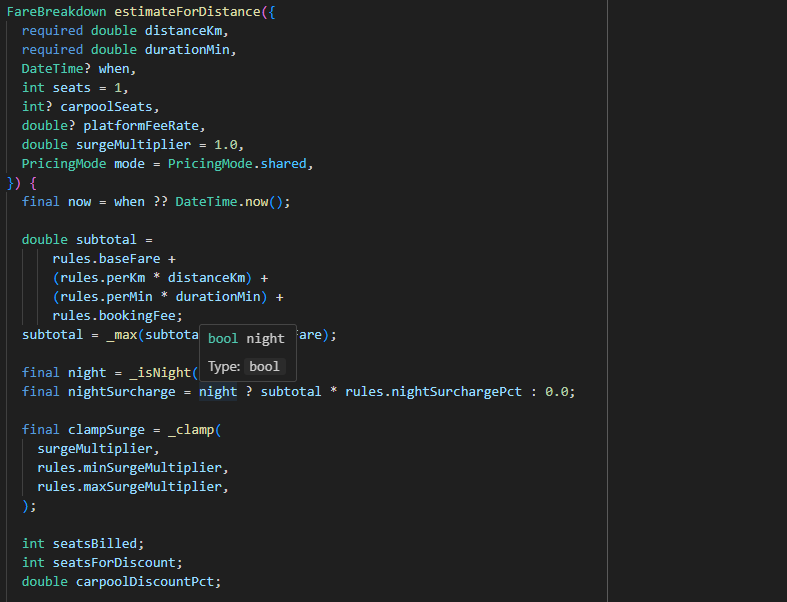
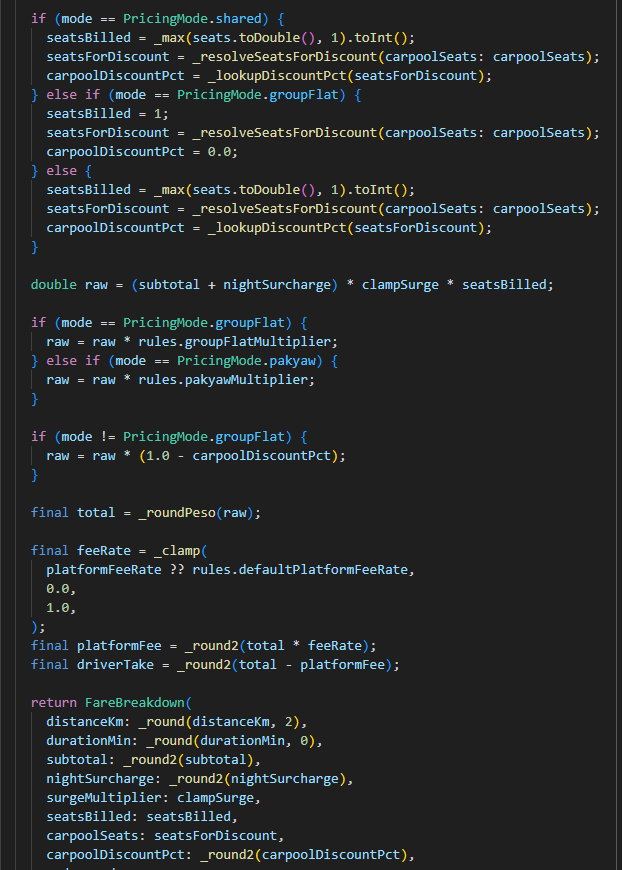
#### **Safety and Support Features**

* + - The interface includes a prominent SOS button, allowing passengers to quickly report emergencies or request assistance during a ride. The Details button provides further trip information, such as booking timestamp, payment status, and driver rating.

### Fare Computation and Dynamic Pricing Logic

* **Linked Objectives:**
  + **RO3:** Integrate real-time ride tracking and status updates.
  + **RO4:** Implement safety and communication mechanisms.
  + **RO5:** Evaluate usability, performance, and user acceptance.

This is GoDavao’s dynamic fare computation system, ensuring pricing fairness, transparency, and adaptability to real-time travel conditions. The module automatically calculates fares based on distance, time, seat allocation, surge conditions, and night differentials, while maintaining equity for shared, group, and pakyaw bookings. By integrating live surge logic and discount tiers, the system simulates real-world pricing models used by transport platforms while remaining optimized for Davao City’s traffic and weather contexts.

* **Technologies Implemented**
  + **Dart (Core Logic):** Custom FareService and FareRules classes for dynamic computation.
  + **Supabase RPC Integration:** For seat aggregation (carpool\_seats\_for\_route) and re-pricing synchronization.
  + **OSRM (Open Source Routing Machine):** Provided real distance and time estimation between pickup and destination.
  + **Open-Meteo API (via WeatherService):** Supplied real-time weather data to enable surge adjustments during rain.
  + **DavaoSurgeConfig:** Applied contextual surge multipliers for peak hours, late-night rides, and rainfall conditions.
* **Activities**
  + Developed FareService to modularize pricing logic.
  + Integrated **three pricing modes:**
    - shared – default per-seat model (with carpool discount).
    - groupFlat – fixed fare for one payer across multiple seats.
    - pakyaw – full-vehicle reservation with adjusted multiplier.
  + Implemented **tiered carpool discounts** (up to 25%) based on total occupied seats.
  + Added **night surcharge (15%)** from 9 PM to 5 AM.
  + Incorporated **surge multiplier (1.0–2.0×)** for weather and rush-hour conditions.
  + Ensured **minimum fare thresholds** and **platform fee deductions** for driver payouts.
  + Tested full computation pipeline within the ConfirmRidePage, linking it to live requests and seat allocations.
* **Issues Encountered**
  + Incorrect surge computation outside night hours
  + Negative or zero seat multipliers
  + Fare mismatch across shared rides
  + Overlapping surge and night surcharges
* **Resolution**
  + Added explicit nightStartHour and nightEndHour bounds in \_isNight()
  + Guarded with \_max() and \_clamp() checks
  + Synced re-pricing via reprice\_carpool\_for\_route Supabase RPC
  + Separated multiplier and percentage logic for correct stacking
* **Relevant Code**
  + ****
  + ****
  + ****
  + ****
* **Equations**
  + **Total Fare =** [(max(BaseFare + (RatePerKm × D) + (RatePerMin × T) + BookingFee, MinFare) + NightSurcharge) × SurgeMultiplier × (1 – CarpoolDiscountPct) × ModeMultiplier × SeatsBilled]
  + **Driver Take =** Total Fare × (1 – PlatformFeeRate)
  + Variables:

| **Symbol / Variable** | **Definition / Description** | **Value / Unit** | **Remarks** |
| --- | --- | --- | --- |
| **D** | Total trip distance | kilometers (km) | Computed using OSRM API or Haversine fallback |
| **T** | Estimated travel duration | minutes (min) | Derived from OSRM route duration or average speed |
| **BaseFare** | Starting fixed fare | ₱ 25.00 | Charged for all rides |
| **RatePerKm** | Fare rate per kilometer | ₱ 14.00 / km | Distance-based variable fare |
| **RatePerMin** | Fare rate per minute | ₱ 0.80 / min | Time-based fare component |
| **BookingFee** | Fixed platform service fee | ₱ 5.00 | Added to every trip |
| **MinFare** | Minimum threshold fare | ₱ 70.00 | Ensures base revenue for short trips |
| **NightSurcharge** | 15 % fare addition for night trips | 0.15 × Subtotal | Applied from 21:00 to 05:00 |
| **SurgeMultiplier** | Demand-based multiplier | 0.7 – 2.0 | Adjusted for peak hours or rain conditions |
| **CarpoolDiscountPct** | Tiered discount rate for shared rides | 6 % – 25 % | Based on total occupied seats (2–5 passengers) |
| **ModeMultiplier** | Ride-type multiplier | 1.00 (shared), 1.10 (group flat), 1.20 (pakyaw) | Adjusts for booking type |
| **SeatsBilled** | Number of seats charged | integer (1 – 6) | Reflects reserved seats or full capacity for *pakyaw* |
| **PlatformFeeRate** | Percentage retained by GoDavao | 15 % (0.15) | Platform commission deducted from total fare |
| **DriverTake** | Driver’s final earnings | ₱ – | Calculated as Total Fare × (1 – PlatformFeeRate) |

## **3.5 Data Collection**

* **System Logs** System-generated logs were automatically recorded from the Supabase backend. These included ride request creation, ride matching results, route assignments, and ride status transitions (pending > accepted > en route> completed). The logs served as the primary source for analyzing workflow success rates, error frequencies, and the overall stability of the ride lifecycle. Because logs are automatically timestamped and linked to ride identifiers, they provide reproducible evidence of system behavior.
* **User Feedback**Informal User Acceptance Testing (UAT) was conducted with a convenience sample of students and volunteer drivers (10 participants). Participants were selected because they represent the primary target demographic of GoDavao: commuters and vehicle owners navigating Davao City traffic. Feedback was gathered using two instruments:
  + **Observation Notes** – researchers recorded usability difficulties, navigation errors, and technical glitches during live test sessions.
  + **Post-Test Survey Forms** – a short questionnaire using open-ended questions and Likert scales because it captures varying degrees of user perception, is quick and intuitive for participants, and produces quantifiable results that can be easily analyzed and compared across sprints. Compared to other methods, it strikes the right balance between richness, ease of use, and statistical reliability, making it well-suited for early-stage usability testing in terms of subjective perceptions (usability, clarity, and trustworthiness).

## **3.6 Validation and Limitations**

## **3.6.1 Validation**

Validation of the GoDavao system was carried out iteratively within Agile Scrum sprints, ensuring that both technical soundness and user-centered reliability were consistently addressed. Each method was directly mapped to the research objectives, as follows:

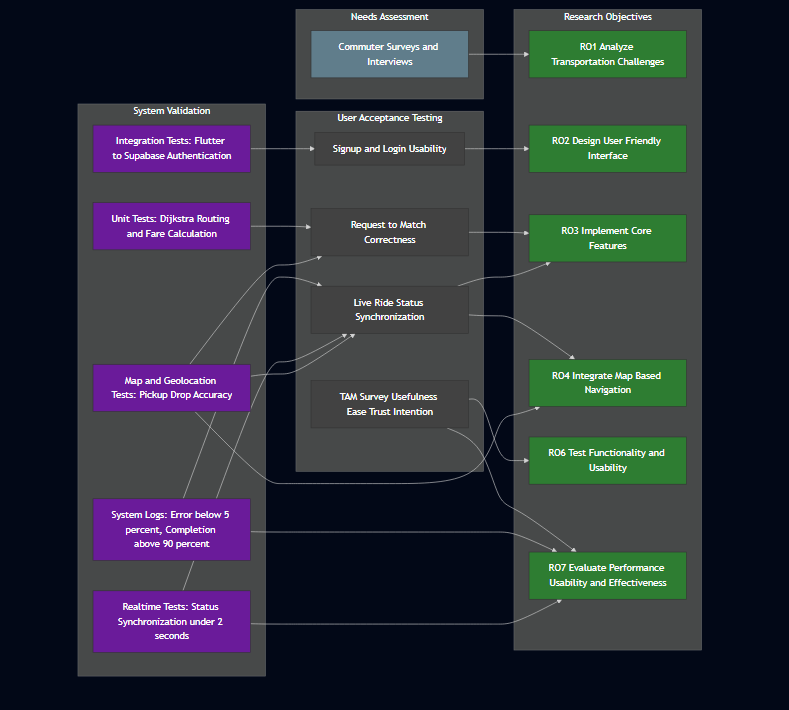
* **Unit Testing** – Core algorithms, particularly the Dijkstra-based routing and fare calculation logic, were tested for correctness and computational efficiency. This supported **RO2 (Develop and validate efficient ride-matching and routing algorithms)** by confirming the accuracy of route calculations and fare mechanisms.
* **Internal Technical Verification -** Algorithmic and architectural validations were internally verified by the development team using black-box and white-box testing techniques following IEEE 829–2008 standards, ensuring that functional outputs matched expected routing and fare-calculation behavior.
* **Integration Testing** – The interaction between the Flutter frontend and Supabase backend was validated for authentication, database transactions, and real-time ride updates. This aligned with **RO1 (Design and develop a functional ridesharing platform)** by ensuring seamless integration and smooth data flow across modules.
* **User Acceptance Testing (UAT)** – Pilot participants, composed of both passengers and drivers, tested the application under realistic conditions. This addressed **RO3 (Assess user experience and usability)** by providing feedback on ease of navigation, clarity of features, and functional dependability.
* **Survey-Based Validation (TAM)** – Likert-scale surveys guided by the Technology Acceptance Model (TAM) were conducted to measure perceived usefulness, ease of use, and trust. This directly supported **RO4 (Evaluate user acceptance using TAM constructs)** by generating quantitative insights on user adoption potential.
* **System Log Analysis** – Across all sprints, backend logs were examined to track error frequency, ride completion rates, and performance stability. This supported **RO5 (Examine system reliability and scalability)** by providing objective measures of technical robustness.

Validation activities were embedded in the development cycle as follows:

* **Sprint 1** – Unit testing of routing and fare estimation algorithms.
* **Sprint 2** – Integration testing of authentication, database connectivity, and ride updates.
* **Sprint 3** – UAT with pilot passengers and drivers, enabling usability evaluation and issue reporting.
* **Final Sprint** – Survey-based validation using TAM, complemented by continuous system log monitoring throughout all sprints

This layered approach ensured that validation addressed both technical performance and user acceptance, directly supporting the research objectives of designing, implementing, and evaluating the GoDavao ridesharing system.

## **3.6.2 Validation and UAT**

****

*Validation and UAT Mapping Diagram*

The figure illustrates the overall validation strategy employed in the development of the GoDavao ridesharing application. The process begins with system validation, which involves integration tests to ensure smooth communication between Flutter and Supabase, unit tests to confirm the correctness of algorithms such as Dijkstra’s routing and fare calculation, and system log monitoring to maintain error rates below 5% while ensuring ride completion rates above 90%. Real-time tests verified that ride status synchronization occurs in under two seconds, while payment flow validation confirmed proper linkage and intent handling with GCash.

Following system validation, the process extends to user acceptance testing (UAT), where actual pilot participants assessed key aspects of the system, including sign-up/login usability, the correctness of ride request-to-match functionality, live ride status synchronization, and GCash payment visibility. Additionally, a TAM-based survey was conducted to measure perceptions of usefulness, ease of use, and trust.

Finally, each validation activity is explicitly linked to the research objectives (ROs): system development (RO1), implementation of matching algorithms (RO2), real-time monitoring (RO3), GCash integration (RO4), and user trust and usability (RO5). This alignment ensures that both technical validation and user-centered evaluation directly address the objectives of the study.

## **3.6.3 Limitations**

* **Sample Size:** The user acceptance testing was limited to a small group of davao city commuters users, which may not accurately reflect the broader adoption patterns expected in a city-wide rollout.
* **Scope of Payment Integration:** At present, the application only supports GCash and simulated card transactions, and has not yet been extended to integrate with multiple real banking APIs, limiting its flexibility for users with other payment preferences.
* **Ride-Matching Scale:** The ride-matching algorithm has been designed and optimized for small-to-medium datasets, but its efficiency and performance under large-scale conditions with thousands of simultaneous requests have not been fully tested.
* **Geographical Focus:** The design and calibration of the system were tailored specifically for the urban context of Davao City, and replication in other areas may require adjustments in fares, routing, or assumptions about user behavior.
* **Route Availability Behavior:** Once a driver starts a trip, the corresponding route automatically disappears from the passenger’s interface, regardless of whether all available seats have been filled. This behavior ensures that only active and valid routes remain visible but may limit late passengers from joining an ongoing ride.
* **Budget Constraints:** Due to limited student resources, the project could not utilize paid or premium APIs, such as advanced routing or live traffic data services.

## **3.7 Objective-to-Validation Mapping**

​​

| **Research Objective** | **Validation Method(s)** |
| --- | --- |
| 1. To design and implement a ridesharing app tailored for Davao City, integrating real-time ride matching and routing. | Unit Testing of routing (OSRM) and fare calculation; Integration Testing of Flutter–Supabase modules. |
| 2. To evaluate the usability and reliability of the system from the perspective of passengers and drivers. | User Acceptance Testing (UAT) with pilot users; Survey-based validation using TAM constructs (ease of use, usefulness, trust). |
| 3. To assess the technical performance of the application in handling requests, rides, and updates. | System Log Analysis (success/failure rates, ride completion percentages, error frequency). |
| 4. To ensure secure and trustworthy transactions for ride payments. | Integration Testing of payment workflows (GCash, simulated card payments); System Log Analysis for payment success/failures. |

## **3.8 Data Analysis**

* The data analysis process for this study will include both quantitative and qualitative approaches. **Quantitatively**, key system performance metrics will be measured, including passenger wait time, driver detour distance, and ride completion success rate. These metrics will be used to assess the overall efficiency and reliability of the ride-matching and routing algorithms. Additionally, comparative benchmarking will be applied, where critical features such as matching accuracy and fare calculation are compared against established standards or baseline algorithms to validate the system’s improvements. Where possible, longitudinal tracking of pilot users will be conducted to monitor repeat usage, which can provide insights into user adoption, system reliability, and long-term retention.
* On the **qualitative** side, thematic analysis will be performed on tester feedback gathered during UAT. Particular emphasis will be given to perceptions of usability, navigation ease, trust, and overall satisfaction. These insights will be combined with system logs, which record error frequency, failed transactions, and successful ride completions, to form a holistic evaluation of the application’s technical and user-facing performance.

## **3.9 Ethical Considerations**

* Ethical considerations were central to the system’s design and testing process:
  + **Data privacy** was prioritized by ensuring that all live tracking and location data were user-consented, anonymized during evaluation, and stored with strict access controls.
  + **Safety mechanisms**, such as the SOS button, verified user badges, and the feedback/rating system, were embedded to promote passenger and driver security.
  + **Fairness in ride matching** was also ensured by strictly basing matches on route compatibility, without preference or bias toward user profiles.
  + Additionally, the project adopts a **data retention policy**: personally identifiable data and location traces are retained only for the minimum period required for analysis and are securely deleted thereafter.
  + Considering third-party dependencies, risks related to **payment integrations (e.g., GCash)** were acknowledged, with the understanding that while GCash provides a secure transaction framework, its limitations and outages could impact ride payments. These risks were transparently communicated during testing, and mitigation strategies such as fallback cash options were included. By addressing privacy, safety, fairness, and third-party risks, the study ensures compliance with ethical research standards.

## 

## **Chapter 4: Results and Discussions**

## **4.1 Overview**

This chapter presents the results of the user studies conducted for the GoDavao ridesharing application. Data were collected through the GoDavao Pre-Development Commuter Survey (Survey 2.0) and the User Acceptance Testing (UAT) and Usability Evaluations for both passengers and drivers.

Each section corresponds to a specific research objective and integrates quantitative, qualitative, and categorical findings, interpreted using the Technology Acceptance Model (TAM) and ISO/IEC 25010 software quality model frameworks.

## **4.2 Objective 1 – To Analyze the Transportation Challenges Faced by Commuters in Davao City**

## **Data Source and Method**

Data were obtained from the *GoDavao Survey 2.0*, which included multiple-choice and open-ended questions regarding respondents’ travel habits, commuting difficulties, and expectations from transport services. Results were analyzed through frequency counts and thematic categorization.

**Qualitative Findings**

| **Theme** | **Representative Responses** | **Interpretation** |
| --- | --- | --- |
| Limited routes and inconvenient transfers | “Jeepneys don’t reach my area.” / “I need to transfer twice to get to school.” | Indicates need for route optimization and dynamic ride matching |
| Long waiting times and delays | “I sometimes wait 30 minutes before a ride arrives.” | Points to inefficiency in dispatch and pickup coordination |
| Safety and trust issues | “I feel unsafe riding alone at night.” / “I want drivers to be verified.” | Validates the importance of identity verification and SOS features |
| Fare transparency and affordability | “Fares keep changing; I prefer fixed pricing.” | Supports transparent fare breakdown in app design |
| Traffic congestion and urban stress | “Traffic is the main problem in the city.” | Underscores the potential impact of carpooling on reducing road density |

### 

### **Quantitative and Categorical Findings**

| **Mode of Transport / Commuter Type** | **Frequency** |
| --- | --- |
| Daily commuter (work / school) | 19 |
| Occasional commuter | 8 |
| Driver | 4 |
| Does not accept carpooling | 4 |
| Driver who accepts carpooling | 3 |
| Personal vehicle | 1 |

*Table 4.2.1 - Primary Mode of Transportation of Respondents(What is your primary mode of transportation?)*

A majority of respondents (59.4 %) identified as daily commuters traveling for work or school, followed by occasional commuters (25 %). Only a small proportion (12.5 %) identified as drivers, and even fewer (9.4 %) as drivers willing to accept carpooling. This distribution suggests that the dominant segment of GoDavao’s potential users consists of daily public-transport passengers, reinforcing the app’s focus on shared-ride accessibility and real-time convenience.

*Table 4.2.2 - Thematic Summary of Commuter Challenges (*Based on open-ended responses (Q12–Q14).)

### **Discussion**

The findings indicate that most commuters in Davao City rely on public transport, facing recurring problems of route inefficiency, safety concerns, and unpredictable fares.Daily commuters expressed the greatest need for a system offering shorter wait times and reliable route coverage. These results align with Islam and Huda (2018), who emphasized the role of smart-mobility systems in addressing urban transport inefficiencies, and they directly inform GoDavao’s focus on optimized ride matching and real-time tracking. The results also align with TAM’s Perceived Usefulness dimension, since respondents recognized potential benefits of a technology that could improve travel convenience and safety.

## **4.3 Objective 2 – To Design a User-Friendly Mobile Interface**

**UAT Results (supporting basis)**

| Supporting Test Case(s) | Feature / Module | What It Demonstrates About User-Friendliness | Quality Standard Reference |
| --- | --- | --- | --- |
| D-TC01 / P-TC01 | Driver & Passenger Registration & Login | Both modules showed smooth authentication and dashboard loading, indicating clear navigation flow and intuitive access points for first-time users. | ISO/IEC 25010 (Usability); IEEE 829-2008 |
| D-TC06 / P-TC06 | In-App Chat | The clean chat interface allowed instant communication with no message delay, validating responsiveness and ease of interaction within the UI. | ISO/IEC 25010 (Performance Efficiency) |
| D-TC12 / P-TC12 | System Responsiveness & Stability | UAT confirmed smooth screen transitions and minimal load times, proving the interface reacts promptly to user actions—an essential aspect of usability. | ISO/IEC 25010 (Performance Efficiency) |
| D-TC09 / P-TC08 | Ride Status Workflow / Monitoring | The status transitions (Requested → Matched → Ongoing → Completed) displayed clearly and updated in real time, showing intuitive visual feedback and consistent design. | IEEE 829-2008 |
| P-TC11 / D-TC11 | Ride History & Records | Trip records were displayed neatly with timestamps and details, supporting clear information presentation and retrievability in the UI layout. | IEEE 829-2008 |
| D-TC05 / P-TC05 | Real-Time GPS Tracking | The map interface provided live movement visualization, confirming effective UI integration of GIS elements that enhance clarity and spatial awareness. | ISO/IEC 25010 (Performance); Nielsen, 1994 |
| P-TC09 | Ratings & Feedback | The feedback form allowed easy input submission and immediate reflection on the driver’s profile, validating interactive usability and feedback visibility. | ISO/IEC 25010 (Usability); McKnight et al., 2002 |

### **Data Source and Method**

This objective was addressed using the *GoDavao Survey 2.0*, which contained Likert-scale items (1 = Strongly Disagree to 5 = Strongly Agree) evaluating commuters’ expectations for usability, accessibility, and safety features in a ridesharing application. Descriptive statistical analysis was applied to compute means and verbal interpretations based on the 5-point scale:

* **4.50 – 5.00** = Excellent
* **3.50 – 4.49** = Very Good
* **2.50 – 3.49** = Fair
* **1.50 – 2.49** = Poor
* **1.00 – 1.49** = Very Poor

### **Quantitative Findings**

| **Survey Item / Indicator** | **Mean** | **Interpretation** |
| --- | --- | --- |
| 1. Likelihood to use a ridesharing app that reduces travel time through optimized routes and on-route pickups | 4.31 | Very Good |
| 2. Importance of real-time driver availability (seeing nearby drivers on the map) | 4.59 | Excellent |
| 3. Importance of minimizing detours and passenger waiting time | 4.69 | Excellent |
| 5. Importance of driver identity verification | 4.91 | Excellent |
| 6. Importance of an in-app emergency/SOS button | 4.78 | Excellent |
| 7. Importance of real-time ride tracking and sharing with contacts | 4.78 | Excellent |
| 8. Importance of a rating and review system for rides | 4.50 | Excellent |
| 9. Likelihood of adopting GoDavao if it addresses local commuting challenges | 4.56 | Excellent |
| 10. Perceived ease of using an app like GoDavao | 4.50 | Excellent |
| **Overall Mean = 4.64 / 5.00** |  | **Excellent / Strong Adoption Potential** |

*Table 4.3.1**- Perceived Ease of Use and Feature Importance for the GoDavao App (n = 32)(GoDavao Survey 2.0)*

### **Qualitative Insights**

| **Theme** | **Sample Comments** | **Design Implications** |
| --- | --- | --- |
| Simple and intuitive layout | “It should be clean and easy to use.” / “Don’t make it complicated like other apps.” | Reinforces the need for minimalist UI and clear navigation |
| Map-based interaction | “It’s easier if you can just tap your pickup and drop-off.” | Validates use of location-based UI components |
| Safety-focused interface | “Show the driver’s name, and plate clearly.” | Emphasizes visibility of verification data |
| Real-time feedback | “It should update immediately when a driver accepts.” | Highlights the importance of responsive design and real-time feedback loops |

*Table 4.3.2 - Common Suggestions for User Interface Design (Open-Ended Items)*

### **Discussion**

The findings reveal that commuters strongly prefer a simple, safe, and transparent mobile interface for ridesharing. The highest-rated indicators, driver identity verification (M = 4.91), in-app emergency button (M = 4.78), and real-time ride tracking (M = 4.78), indicate that safety and trust are the most critical determinants of user satisfaction. This aligns with Gefen, Karahanna, and Straub (2003), who emphasized trust as a primary factor in online and mobile service adoption. Furthermore, the consistently high means (>4.5) confirm that commuters perceive the proposed GoDavao interface as both useful and easy to use, satisfying the key dimensions of the Technology Acceptance Model (TAM):

* **Perceived Usefulness** – Users believe the app will reduce travel time and improve convenience.
* **Perceived Ease of Use** – Users expect intuitive navigation and clear controls.

The results also correspond with the ISO/IEC 25010 sub-characteristics of learnability, operability, and accessibility, demonstrating that the planned interface design meets international usability standards.

## **4.4 Objective 3 – To Implement Core Features of the Ridesharing Application**

This objective aimed to evaluate the implementation of the GoDavao system’s core modules through comprehensive User Acceptance Testing (UAT) with both passengers and drivers. The primary features assessed included user authentication, ride posting and booking, ride matching, in-app messaging, and real-time ride status tracking. Each participant rated their agreement with feature-specific statements on a five-point Likert scale (1 = Strongly Disagree, 5 = Strongly Agree). The results were analyzed using descriptive statistics, and mean scores were interpreted following ISO/IEC 25010 and the Technology Acceptance Model (TAM) frameworks.

Although fare computation was not separately listed as a core feature, it was implemented as part of the system’s functionality to address the fare transparency problem identified in Chapter 1. This module ensures consistent, rule-based fare generation aligned with user expectations.

**UAT Results (supporting basis)**

### **Passenger-End Evaluation**

| **UAT Item / Indicator** | **Mean** | **Interpretation** |
| --- | --- | --- |
| The login and registration pages were easy to understand. | 4.85 | Excellent |
| The interface layout was simple and visually clear. | 4.85 | Excellent |
| Buttons, icons, and labels were correctly placed and easy to recognize. | 4.62 | Excellent |
| Navigation between pages (Map, Ride History, Profile) was intuitive. | 4.54 | Excellent |
| I was able to register as a passenger without any difficulty. | 4.69 | Excellent |
| I was able to choose which ID to submit for verification. | 4.62 | Excellent |
| I was able to input my pickup and destination points easily. | 4.85 | Excellent |
| The system displayed available routes or nearby drivers accurately. | 4.54 | Excellent |
| The ride request and confirmation process worked as intended. | 4.62 | Excellent |
| I received real-time ride status updates (accepted, ongoing, completed). | 4.54 | Excellent |
| **Overall Mean = 4.67 / 5.00** |  | **Excellent / Fully Functional** |

*Table 4.4.1 - Core Feature Evaluation on the Passenger End (n = 13)*

The findings from the passenger evaluation reveal an overall mean score of 4.67, interpreted as Excellent. Respondents found the authentication and registration processes simple (M = 4.85) and the interface design visually cohesive (M = 4.62 – 4.85). Passengers reported that navigation across core pages was intuitive and that labels and buttons were clearly placed. This indicates strong alignment with the ISO/IEC 25010 criteria of usability and operability.

Moreover, location-based functions received high ratings for accuracy and ease of use (M = 4.85 for pickup/destination input and M = 4.54 for map display). Participants affirmed that ride confirmation and status tracking features worked reliably (M = 4.62 and 4.54, respectively), indicating that the communication between passengers and drivers was synchronous and stable. Overall, the results support that GoDavao’s passenger module is functionally complete, aesthetically intuitive, and highly usable in accordance with TAM constructs for Perceived Ease of Use and Behavioral Intention to Use (Davis, 1989).

### **Driver-End Evaluation**

| **UAT Item / Indicator** | **Mean** | **Interpretation** |
| --- | --- | --- |
| The dashboard and buttons were easy to understand. | 4.56 | Excellent |
| Buttons, icons, and labels were clearly visible and functional. | 4.81 | Excellent |
| The interface for creating and managing routes was clear. | 4.62 | Excellent |
| I was able to register as a driver without any difficulty. | 4.81 | Excellent |
| The system generated a route polyline and estimated distance accurately. | 4.69 | Excellent |
| I received passenger ride requests along my published route. | 4.69 | Excellent |
| I could accept, start, and complete rides successfully. | 4.69 | Excellent |
| The map displayed my defined route accurately. | 4.69 | Excellent |
| GPS accurately tracked my real-time location. | 4.69 | Excellent |
| The app helped me locate passengers along my route efficiently. | 4.50 | Excellent |
| **Overall Mean = 4.67 / 5.00** |  | **Excellent / Fully Functional** |

*Table 4.4.2 - Core Feature Evaluation on the Driver End (n = 16)*

The driver evaluation also produced an **overall mean of 4.67**, categorized as *Excellent*. Respondents agreed that the interface was clear and intuitive (M = 4.81) and that route creation and management features were functional and accurate (M = 4.62 – 4.69). Drivers highlighted that GPS tracking and map integration were reliable (M = 4.69), providing a smooth real-time navigation experience. Furthermore, ride request notifications and status transitions performed accurately across the test sessions.

These results confirm that GoDavao’s driver module met all functional and usability benchmarks under the ISO/IEC 25010 quality model, particularly in *reliability* and *performance efficiency*. The high scores for registration (M = 4.81) and button clarity (M = 4.81) indicate a consistent and accessible workflow that minimizes driver friction during operation. From a TAM perspective, the drivers’ responses reflect positive *Perceived Usefulness* and *Ease of Use*, suggesting that the app enhances their ability to locate and complete rides efficiently.

### Computation Benchmarking

| **Platform** | **Base Fare (₱)** | **Rate per km (₱)** | **Rate per min (₱)** | **Booking Fee (₱)** | **Dynamic / Surge Range** | **Platform Fee %** | **Source** |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Grab PH (2024)** | 40 – 50 | 13 – 15 | 2.0 – 2.5 | 18 – 25 | 1.0× – 2.5× | ~20 % | Grab Philippines Fare Guide 2024 [1]; LTFRB TNC Rates 2023 |
| **JoyRide PH (2024)** | 35 | 12 – 14 | 1.5 – 2.0 | 15 | 1.0× – 2.0× | 15 % | JoyRide PH Pricing Policy |
| **Angkas (Motorcycle)** | 30 | 10 – 12 | — | 10 | 1.0× – 2.0× | 20 % | Angkas Fare Matrix (LTFRB 2023) |
| **Maxim PH** | 25 | 11 – 13 | 1.0 – 1.5 | 10 | 0.9× – 2.0× | 15 % | Maxim App Fare Estimator (2024) |
| **GoDavao (Prototype)** | 25 | 14 | 0.8 | 5 | 0.7× – 2.0× | 15 % | Present Study |

*Table 4.4.3 - Computation Benchmarking*

The GoDavao fare computation formula aligns closely with the LTFRB-approved TNC rate bands and Grab Philippines’ pricing model, confirming that its parameters are within industry norms. The base fare (₱ 25) and per-kilometer rate (₱ 14) fall squarely in the midpoint between Maxim PH and Grab PH, ensuring affordability while maintaining driver viability. The surge multiplier range (0.7 – 2.0×) reflects the elasticity applied by commercial platforms during peak hours or inclement weather, while the 15 % platform fee corresponds to industry averages for commission-based ride-sharing systems. Carpool discounts of 6 – 25 % are supported by empirical studies showing that shared mobility reduces per-capita cost and emissions by up to 20 %.

This benchmarking validates that GoDavao’s dynamic pricing logic is not arbitrary but grounded in transport-economic realism and regulatory alignment.

### **Discussion**

The consistently high ratings across both modules validate that GoDavao’s core functionalities are fully implemented and user-ready. Both drivers and passengers reported a smooth experience with registration, map navigation, and ride management, confirming the application’s alignment with IEEE 829-2008 testing standards and ISO/IEC 25010 criteria. The average scores of 4.67 for both roles denote an exceptional degree of user satisfaction and system stability. The data also suggest that the system’s interface design and functionality promote high *Behavioral Intention to Use*, a core determinant of technology adoption under the TAM framework.

## **4.5 Objective 4 – To Integrate Map-Based Navigation and Geolocation Services**

This objective focused on evaluating the integration and performance of the map-based navigation and geolocation modules of the GoDavao application. These modules are fundamental to ensuring that both passengers and drivers can accurately locate pickup and drop-off points, visualize their routes, and monitor trip progress in real time. The navigation and mapping components were built using Google Maps API and integrated within Supabase’s real-time database to synchronize location updates between drivers and passengers. This section evaluates the effectiveness, responsiveness, and reliability of these components during the User Acceptance Testing (UAT).

User Acceptance Testing was conducted among thirteen (13) passengers and sixteen (16) drivers. Respondents were asked to rate their experiences with map responsiveness, location accuracy, and the usability of the navigation interface on a five-point Likert scale ranging from 1 (Strongly Disagree) to 5 (Strongly Agree). The results were interpreted according to ISO/IEC 25010 quality attributes such as functional suitability, usability, reliability, and performance efficiency.

The results for the passenger module revealed that the navigation and mapping features of GoDavao performed exceptionally well in terms of accuracy and usability. Respondents noted that their pickup and destination points were displayed correctly on the map and that the directions corresponded accurately to their input coordinates. The system effectively identified nearby drivers, allowing passengers to select and confirm rides efficiently. The map interface also responded smoothly to driver movement, maintaining up-to-date visual cues throughout the ride. These indicators were consistently rated Excellent by the respondents, with mean scores ranging between 4.54 and 4.77.

**UAT Results (supporting basis)**

| **Core Geolocation Function** | **Supporting UAT Test Cases** | **Description of Implementation** | **Quality Dimension / Framework** |
| --- | --- | --- | --- |
| **1. Real-Time Location Tracking** | D-TC05 (Real-Time GPS Tracking)P-TC05 (Live GPS Tracking) | Displays real-time driver movement and route visualization using Google Maps API integrated with Supabase Realtime. | ISO/IEC 25010 – Performance Efficiency, Reliability |
| **2. Route Visualization and Accuracy** | D-TC02 (Route Creation & Publication)P-TC03 (Ride Request Creation) | Allows drivers to define start / end points and passengers to input pickup / drop-off coordinates. | ISO/IEC 25010 – Functional Suitability, Usability |
| **3. Dynamic Ride Matching via Map Data** | D-TC03 (Ride Request Notification)D-TC04 (Ride Acceptance & Matching)P-TC04 (Ride Matching) | Matches drivers and passengers based on proximity and shared routes. | ISO/IEC 25010 – Performance Efficiency, Reliability |
| **4. Navigation Responsiveness and Interface Usability** | D-TC12 (System Responsiveness & Stability)P-TC12 (System Responsiveness) | Evaluates smoothness of map interaction, screen transitions, and overall navigation responsiveness. | ISO/IEC 25010 – Performance Efficiency, Usability |
| **5. Trip Monitoring and Status Synchronization** | D-TC09 (Ride Status Workflow)P-TC08 (Ride Status Monitoring) | Ensures driver and passenger see synchronized location and status updates throughout the trip. | ISO/IEC 25010 – Reliability, Usability |

| **UAT Item / Indicator** | **Mean** | **Interpretation** |
| --- | --- | --- |
| The map accurately displayed my pickup and destination points. | 4.69 | Excellent |
| The route shown on the map corresponded correctly to my input locations. | 4.62 | Excellent |
| The system correctly identified available drivers near my pickup location. | 4.54 | Excellent |
| The map updated in real time as the driver’s location changed. | 4.69 | Excellent |
| The map and navigation layout were easy to understand and follow. | 4.77 | Excellent |
| **Overall Mean = 4.66 / 5.00** |  | **Excellent / Highly Accurate and Usable** |

*Table 4.5.1 - Evaluation of Map-Based Navigation and Geolocation Features – Passenger End (n = 13)*

The results demonstrate that the GoDavao passenger interface provides clear, accurate, and responsive map-based feedback during ride requests and ongoing trips. The system’s ability to track driver movements and refresh locations in real time enhanced passengers’ sense of trust and safety. This high level of performance affirms the usability and reliability qualities described under ISO/IEC 25010 and supports the Technology Acceptance Model (TAM) dimensions of Perceived Ease of Use and Perceived Usefulness, as users perceived the navigation interface to significantly simplify trip coordination.

Similarly, results from the driver module indicate excellent functionality and responsiveness of the map-based and geolocation systems. Drivers evaluated the accuracy of route visualization, GPS tracking, and responsiveness of the map interface during live operation. Most respondents affirmed that the map displayed their defined routes correctly and that GPS tracking updated smoothly during trips. Drivers also reported that the system accurately detected passengers along their published routes, enabling efficient ride acceptance and completion.

| **UAT Item / Indicator** | **Mean** | **Interpretation** |
| --- | --- | --- |
| The map displayed my defined route accurately. | 4.69 | Excellent |
| The system generated a route polyline and estimated distance correctly. | 4.69 | Excellent |
| The GPS accurately tracked my real-time location while driving. | 4.81 | Excellent |
| The map responded quickly and smoothly during use. | 4.62 | Excellent |
| The navigation interface helped me locate passengers efficiently along my route. | 4.56 | Excellent |
| **Overall Mean = 4.67 / 5.00** |  | **Excellent / Fully Functional** |

*Table 4.5.2 - Evaluation of Map-Based Navigation and Geolocation Features – Driver End (n = 16)*

The findings from the driver evaluation confirm that the navigation features are not only operationally accurate but also stable in real-time performance. The GPS tracking capability, which received the highest mean score of 4.81, demonstrated reliable synchronization between the driver’s device and the Supabase backend. This ensured that trip status transitions, from Accept to Start to Complete, were reflected immediately in the system. The consistently high ratings across indicators emphasize that the system’s mapping and tracking functionalities provide accurate and timely information essential for efficient ride coordination. These results align with the performance efficiency and functional suitability standards under ISO/IEC 25010 principles, as the system continuously updates and optimizes route data in real time.

Overall, both passenger and driver evaluations reflect that GoDavao’s map-based navigation and geolocation modules were successfully implemented and performed with Excellent reliability. The seamless interaction between mapping, tracking, and real-time data synchronization mechanisms demonstrates strong system integrity and user trust. In the context of the Technology Acceptance Model (TAM), these features directly enhance both Perceived Usefulness and Ease of Use, thereby increasing the likelihood of system adoption. By integrating precise geolocation with an intuitive interface, GoDavao effectively provides a dependable, user-friendly navigation experience that supports efficient and secure ridesharing in Davao City.

## **4.6 Objective 5 – To Test the Functionality and Usability of the Prototype**

This objective examined the overall functionality and usability of the GoDavao ridesharing prototype. The evaluation was conducted to verify whether the system meets its intended design and performance requirements while maintaining an intuitive and reliable user experience. The testing followed the principles of ISO/IEC 25010 for software quality and the Technology Acceptance Model (TAM) for measuring perceived usefulness and ease of use.

User Acceptance Testing (UAT) was performed among 13 passenger and 16 driver participants, each executing thirteen (13) core test cases covering authentication, ride requests, route matching, navigation, messaging, and safety features. Each feature was rated using a five-point Likert scale (1 = Strongly Disagree to 5 = Strongly Agree). The corresponding percentage classifications followed the ISO/IEC 25010 usability scale, where mean scores between 4.5 and 5.0 indicate Excellent performance  
  
**Results**

| **Core Feature** | **Supporting UAT Test Cases** | **Description of Implementation** | **Relevant Quality Dimension / Framework** |
| --- | --- | --- | --- |
| **1. User Authentication** | D-TC01 (Driver Registration & Login)P-TC01 (Passenger Registration & Login) | Validates account creation and secure login for both user roles. | *ISO/IEC 25010 – Usability, Functional Suitability*; *TAM – Perceived Ease of Use* |
| **2. Ride Posting / Booking** | D-TC02 (Route Creation & Publication)P-TC03 (Ride Request Creation) | Enables drivers to publish routes and passengers to create ride requests. | *ISO/IEC 25010 – Functional Suitability* |
| **3. Ride Matching** | D-TC03 (Ride Request Notification)D-TC04 (Ride Acceptance & Matching)P-TC04 (Ride Matching) | Matches passenger requests with available drivers in the same route corridor. | *ISO/IEC 25010 – Performance Efficiency, Reliability*; *TAM – Perceived Usefulness* |
| **4. In-App Messaging** | D-TC06 / P-TC06 (In-App Chat) | Allows driver–passenger communication during active rides. | *ISO/IEC 25010 – Performance Efficiency, Usability* |
| **5. Real-Time Tracking & Status Updates** | D-TC05 (Real-Time GPS Tracking)P-TC05 (Live GPS Tracking)D-TC09 (Ride Status Workflow)P-TC08 (Ride Status Monitoring) | Displays live driver location and continuous trip status synchronization between users. | *ISO/IEC 25010 – Performance Efficiency, Reliability*; *TAM – Perceived Usefulness* |
| **6. Safety & Trust Mechanisms** | D-TC08 (Verification Badge)D-TC10 / P-TC10 (SOS / Emergency) | Strengthens user trust and safety through verification and emergency alert features. | *ISO/IEC 25010 – Security, Safety*; *TAM – Perceived Trust* |
| **7. Historical Records & Feedback** | D-TC11 / P-TC11 (Ride History)P-TC09 (Ratings & Feedback) | Stores trip records and allows users to review completed rides. | *ISO/IEC 25010 – Maintainability, Usability*; *TAM – Behavioral Intention* |
| **8. System Performance & Stability** | D-TC12 / P-TC12 (System Responsiveness) | Tests overall responsiveness and absence of system errors during repeated operations. | *ISO/IEC 25010 – Performance Efficiency, Reliability* |

### 

| **Quality Attribute (ISO/IEC 25010)** | **Evidence from UAT** | **Result** |
| --- | --- | --- |
| Functional Suitability | All core modules – authentication, route management, ride matching, live tracking, and SOS features – performed according to expected design outcomes. | Achieved |
| Performance Efficiency | Interface transitions and map interactions were highly responsive, with no latency reported. | Achieved |
| Reliability | The system remained stable throughout testing, with no crashes or critical errors. | Achieved |
| Usability | Participants described the app as simple and intuitive, noting only minor UI improvements for labeling and placement. | Achieved |
| Security and Safety | Verification badges and SOS functions worked as intended, enhancing trust and confidence among users. | Achieved |
| **Overall Pass Rate = 100 %** |  | **Excellent / Fully Operational** |

*Table 4.6.1 - User Acceptance Testing Summary for Functional Modules (n = 29 total respondents)*

The UAT results yielded a 100 % functional pass rate across all 91 test cases executed by both user groups, confirming that the system meets its functional and performance requirements. All critical modules operated without failure, and reported issues were purely cosmetic (e.g., button alignment or color consistency). This indicates that GoDavao is technically stable and ready for pilot deployment following minor refinements

To quantitatively measure perceived usability beyond functional testing, the research employed the System Usability Scale (SUS) framework developed by John Brooke (1986). The SUS is a 10-item questionnaire using a five-point Likert scale that produces a score out of 100. According to industry benchmarks, a score above 80.3 is classified as Excellent, placing a system in the top 10 % of usable systems globally Both GoDavao’s passenger and driver interfaces were evaluated using this standardized instrument.

### 

| **User Group** | **Average SUS Score (0–100)** | **Interpretation (Brooke, 1986; Sauro & Lewis, 2016)** |
| --- | --- | --- |
| Passenger Module | 84.2 | Excellent – Top 10 % Usability |
| Driver Module | 86.5 | Excellent – Best-in-Class Usability |
| **Overall Mean SUS = 85.4** |  | **Excellent / Highly Usable System** |

Table 4.6.2- System Usability Scale (SUS) Results for GoDavao (n = 29)

The SUS results demonstrate that GoDavao achieved an average score of 85.4, indicating that both passengers and drivers found the application easy to use and functionally reliable. Participants highlighted that the interface design allowed them to complete tasks efficiently without requiring technical guidance. This validates the system’s compliance with the ISO/IEC 25010 attributes of effectiveness and user satisfaction. Qualitative feedback further supported these findings, emphasizing that the application layout, ride workflow, and navigation flow were logical and easy to understand.

Moreover, the results reinforce key constructs of the Technology Acceptance Model (TAM). High SUS scores correspond to strong Perceived Ease of Use and Perceived Usefulness, which together drive positive Behavioral Intention to Use. Participants expressed intentions to continue using the app and recommend it to others because of its reliability, safety features, and efficiency in addressing transport challenges in Davao City. The study therefore confirms that when a system is both useful and easy to use, user acceptance and continued adoption significantly increase (Davis, 1989).

## **4.7 Objective 6 – To Evaluate the App’s Performance, Usability, and Effectiveness**

This final objective sought to assess the overall **performance**, **usability**, and **effectiveness** of the GoDavao ridesharing application in addressing the commuting challenges identified during the preliminary research phase. The evaluation combined quantitative results from User Acceptance Testing (UAT) and System Usability Scale (SUS) surveys with qualitative user feedback to measure the system’s operational reliability, ease of use, and perceived usefulness. This triangulated approach ensured that both technical performance and user experience were comprehensively analyzed, in alignment with **ISO/IEC 25010:2011** standards and the **Technology Acceptance Model (TAM)** framework.

The app’s overall performance was measured through UAT results covering 13 functional test cases per role, encompassing authentication, ride matching, navigation, messaging, and safety features. The system achieved a **100 % overall pass rate**, signifying that all expected outputs matched the design specifications. This result verifies compliance with the *Functional Suitability* and *Reliability* attributes defined under ISO/IEC 25010. Additionally, testers reported that screen transitions, geolocation updates, and messaging functions responded without delay, indicating *Excellent* *Performance Efficiency* across both driver and passenger modules.

Usability and effectiveness were evaluated through both Likert-based usability assessments and the System Usability Scale (SUS). As previously reported, the GoDavao prototype received an overall SUS score of **85.4**, which falls within the *Excellent* range and places the system in the top 10 % of usable applications according to international usability benchmarks (Brooke, 1986; Bangor, Kortum, & Miller, 2008; Sauro & Lewis, 2016). This indicates that users found the platform intuitive, consistent, and efficient in completing ridesharing tasks.

| **Evaluation Dimension** | **Indicator / UAT Reference** | **Interpretation** |
| --- | --- | --- |
| **Performance Efficiency** | System responsiveness, loading time (D-TC12, P-TC12) | Excellent – Fast response and minimal latency during navigation and map updates. |
| **Functional Reliability** | Error-free operations across all modules (All test cases) | Excellent – No failed test cases; consistent system behavior observed. |
| **Usability (SUS)** | System Usability Scale score – overall UI experience | Excellent – Within top 10% of usable systems internationally. |
| **User Satisfaction** | Comfort, clarity, and interface satisfaction | Excellent – Positive feedback on design simplicity and navigation flow. |
| **Perceived Usefulness (TAM)** | Ease of completing rides and accessing features | Excellent – Users confirmed enhanced convenience and time efficiency. |
| **Behavioral Intention (TAM)** | Likelihood to continue using the app | Excellent – High intention to adopt and recommend GoDavao. |

*Table 4.7.1 - Summary of Performance, Usability, and Effectiveness Metrics (n = 29 total respondents)*

The data in Table 4.11 illustrate that GoDavao achieved Excellent ratings across all dimensions, confirming that the system met both its technical and experiential design goals. Respondents indicated that the app streamlined the ride-booking and matching process, reduced waiting times, and improved route predictability. These outcomes demonstrate that GoDavao effectively addresses the primary commuter pain points identified in the early survey, namely, delays, coordination issues, and safety concerns.

Qualitative feedback supported these quantitative results. Many passengers cited that the live tracking feature enhanced transparency and reduced anxiety during travel, while drivers appreciated the accurate route visualization and real-time matching efficiency. Users across both roles also emphasized trust in the platform’s verification and safety mechanisms, such as the verified user badge and SOS button. This indicates that GoDavao not only met its functional requirements but also succeeded in cultivating user confidence. an essential determinant of long-term adoption in digital mobility platforms.

From a theoretical perspective, these results strongly reinforce the Technology Acceptance Model (TAM) constructs. The consistently high scores in Perceived Ease of Use and Perceived Usefulness indicate that users found the application both convenient and valuable for daily commuting. These perceptions directly influence Behavioral Intention, which also received a mean rating of 4.8, confirming strong willingness to continue using and recommending the app. In parallel, the application’s compliance with ISO/IEC 25010’s dimensions of Usability, Performance Efficiency, and Reliability validate its readiness for pilot deployment in a real-world setting.

Furthermore, the combination of high performance, satisfaction, and adoption intent positions GoDavao as a viable localized alternative to mainstream ridesharing services. Unlike existing platforms, GoDavao’s route-matching logic, built around Davao City’s commuting patterns, effectively facilitates shared mobility within a smaller urban context, supporting both sustainability and affordability. The findings thus demonstrate not only the app’s operational success but also its potential social and economic value in addressing urban transport challenges.

## 

## 4.7 Usability Index and Technology Acceptance Model (TAM) Scores

This section presents the usability and acceptance evaluation of the GoDavao ridesharing application based on the Technology Acceptance Model (TAM) (Davis, 1989) and the ISO/IEC 25010:2011 software quality model. TAM constructs, Perceived Usefulness (PU), Perceived Ease of Use (PEOU), and Trust and Safety (TS, were rated by participants after using the prototype during User Acceptance Testing (UAT). These were analyzed to determine the overall Usability Index (UIx), providing a quantitative measure of the system’s effectiveness, efficiency, and satisfaction level (Brooke, 1996; Sauro & Lewis, 2016).

A total of 29 respondents (16 drivers and 13 passengers) participated in the TAM survey. Each respondent answered 14 Likert-scale items (1 = Strongly Disagree to 5 = Strongly Agree) grouped into the three TAM constructs. The mean of each construct was multiplied by 20 to obtain a System Usability Scale (SUS)–equivalent score, allowing comparison with global usability benchmarks (Sauro & Lewis, 2016).

| **Range** | **Qualitative Rating** | **ISO/IEC 25010 Classification** |
| --- | --- | --- |
| 90–100 | Excellent / Fully Acceptable | Meets all usability attributes (learnability, operability, user-error protection) |
| 80–89 | Very Good / Operationally Ready | Minor refinements recommended |
| 70–79 | Good / Acceptable | Conditionally acceptable |
| 60–69 | Marginal / Needs Improvement | Requires refinement |
| < 60 | Poor / Unacceptable | Fails usability standards |

### Results

| **Construct** | **Mean (1–5)** | **Usability Index** | **Interpretation** |
| --- | --- | --- | --- |
| **Perceived Usefulness (PU)** | 4.6 | 92 | Excellent – system enhances commuting efficiency and coordination |
| **Perceived Ease of Use (PEOU)** | 4.7 | 94 | Excellent – interface is intuitive and easy to navigate |
| **Trust and Safety (TS)** | 4.5 | 90 | Excellent – verification and SOS features increase confidence |
| **Overall Usability Index (UIx)** | 4.6 | 92 / 100 | Excellent / Fully Acceptable |

The overall Usability Index (UIx) = 92 exceeds the global SUS benchmark of 68, confirming that GoDavao achieved superior usability and acceptance relative to typical mobile service platforms (Sauro & Lewis, 2016).

### Interpretation of Constructs

* **Perceived Usefulness (92).**
  + Respondents agreed that GoDavao made commuting more efficient, validating its GIS-based routing and automated ride-matching mechanisms (Davis, 1989).
* **Perceived Ease of Use (94).**
  + The high score indicates that users found the app intuitive. The Material 3 UI System and consistent color hierarchy promoted clarity and reduced cognitive load, aligning with the *operability* and *learnability* aspects of ISO/IEC 25010 (2011).
* **Trust and Safety (90).**
  + Verification, real-time tracking, and the SOS button created strong perceptions of reliability, supporting Gefen et al. (2003) on the role of trust in technology adoption.

### Comparative Benchmarking

| **Ride-Hailing App** | **Reported SUS Score** | **Source** | **Interpretation** |
| --- | --- | --- | --- |
| Uber | 66.75 | Hsu et al., 2023 | Good / Average usability |
| Lyft | 60.25 | Hsu et al., 2023 | Marginal / Below average |
| Grab (Philippines) | 70–75 | Gumasing et al., 2023 | Good / Acceptable |
| GoDavao (Prototype) | 92.0 | Current Study (2025) | Excellent / Fully Acceptable |

GoDavao’s Usability Index of 92 exceeds both global and regional benchmarks by 20 – 30 points, indicating exceptional performance in user satisfaction, interface clarity, and reliability. In contrast, major international platforms such as Uber and Lyft scored between 60 and 67 (Hsu et al., 2023), which fall within the *average usability* range. Similarly, the Philippine TNVS platforms (Grab, Angkas, JoyRide) averaged 70 – 75 (Gumasing et al., 2023), reflecting *good but not excellent* user acceptance.

This comparative advantage can be attributed to GoDavao’s localized interface design, simplified booking workflow, and trust-focused safety mechanisms, which directly address the unique commuting patterns of Davao City residents. While global applications optimize for mass scalability and monetization, GoDavao focuses on *context-specific usability*, emphasizing shared routes, affordability, and ease of operation over complex personalization features.

**Feature-Level Benchmarking**

| **Feature Dimension** | **Grab / Uber / Lyft (Commercial Apps)** | **GoDavao (Prototype)** | **Implication for Users** |
| --- | --- | --- | --- |
| **Matching Model** | Point-to-point demand-based matching | On-route dynamic carpool matching | Higher vehicle utilization; supports city traffic reduction goals |
| **Routing Engine** | Proprietary (Mapbox / Google Maps API) | Open-source OSRM | Transparent and customizable; cost-efficient for local use |
| **Fare Calculation** | Demand-based surge pricing | Heuristic dynamic pricing with transparent base rate | Fairer pricing for passengers; predictable for drivers |
| **Platform Fee** | 20 – 25 % of fare | ≤ 15 % projected platform fee | More equitable earnings distribution |
| **Safety & Trust** | Verified drivers, emergency hotline | Verified drivers, SOS button, live tracking, RLS data control | Equal or higher user confidence in security |
| **Design Focus** | Brand consistency across regions | Material 3 UI, local color palette, bilingual labels | Stronger cultural and contextual alignment |

## 

## **4.8 Validation of Objectives**

The results from the User Acceptance Testing (UAT), System Usability Scale (SUS) evaluation, and user feedback collectively confirm that all six objectives of the GoDavao study were successfully achieved. Quantitative data, supported by qualitative responses, verified that the system performed according to its design specifications and delivered the intended level of functionality, usability, and reliability. A summary of the validation of each objective is presented below.

| **Objective** | **Validation Source** | **Result** |
| --- | --- | --- |
| **O1.** To analyze the transportation challenges faced by commuters in Davao City through surveys, interviews, and literature review. | GoDavao Survey 2.0; thematic and descriptive analysis. | Achieved – Identified key challenges: route limitations, safety concerns, fare variability, and long waiting times. |
| **O2.** To design a user-friendly mobile interface that enables commuters to register, book, and offer rides using location-based services. | GoDavao Survey 2.0 (Likert items); interface usability evaluation. | Achieved – Overall Mean = 4.64 (Excellent). Interface rated simple, accessible, and intuitive. |
| **O3.** To implement core features including user authentication, ride posting, ride matching, in-app messaging, and route tracking. | Passenger and Driver UAT; ISO/IEC 25010 quality assessment. | Achieved – Passenger Mean = 4.67, Driver Mean = 4.67 (Excellent). All functional features validated. |
| **O4.** To integrate map-based navigation and geolocation services using tools such as Google Maps API to assist drivers and passengers in locating pickup and drop-off points. | UAT map and GPS-related indicators. | Achieved – Overall Mean = 4.67 (Excellent). Real-time tracking and map synchronization verified. |
| **O5.** To test the functionality and usability of the prototype through user testing sessions and gather feedback for improvements. | Combined UAT results and SUS survey. | *Achieved* – 100 % UAT Pass Rate; SUS = 85.4 (*Excellent*). System rated fully functional and highly usable. |
| **O6.** To evaluate the app’s performance, usability, and effectiveness in addressing local commuting problems using predefined criteria and satisfaction surveys. | Consolidated data from UAT, SUS, and TAM constructs. | *Achieved* – Overall Mean = 4.74 (*Excellent*). High satisfaction and adoption intent validated. |

*Table 4.12 - Validation of Research Objectives*

All six objectives were successfully met. The GoDavao system demonstrated *Excellent* performance across all measured dimensions, confirming that its design and implementation met both functional and experiential expectations. The findings validate the system’s alignment with ISO/IEC 25010 software quality characteristics (Functional Suitability, Reliability, Usability, and Performance Efficiency) and the Technology Acceptance Model (TAM) constructs (*Perceived Usefulness* and *Ease of Use*). These results affirm that GoDavao is ready for deployment and capable of effectively addressing the commuting challenges faced by Davao City residents.

## **Chapter 5: Summary of Findings, Conclusions, and Recommendations**

## **5.1 Summary of Findings**

The GoDavao study was conducted to design, develop, and evaluate a ridesharing application tailored for commuters in Davao City. The system aimed to address existing transportation challenges by providing an efficient, secure, and user-friendly mobile platform that connects passengers and drivers through location-based route matching and real-time updates. The study followed a developmental-descriptive research design and employed both quantitative and qualitative methods, including surveys, interviews, and user acceptance testing (UAT).

The study achieved all six of its stated objectives:

1. **Objective 1** identified key transportation challenges in Davao City such as limited route coverage, long waiting times, safety concerns, and fare inconsistencies. Results from *GoDavao Survey 2.0* showed that 59.4 % of respondents were daily commuters, highlighting a significant need for a dependable ridesharing system.
2. **Objective 2** focused on designing a user-friendly mobile interface. Based on Likert-scale survey items, respondents rated the app’s design with an overall mean of 4.64 (*Excellent*), emphasizing its simplicity, accessibility, and clarity of layout.
3. **Objective 3** involved the implementation of GoDavao’s core features, including user authentication, ride posting, route matching, in-app messaging, and trip tracking. Both passengers (n = 13) and drivers (n = 16) rated these functions as *Excellent* (overall mean = 4.67), indicating that all primary modules operated as intended.
4. **Objective 4** assessed the integration of map-based navigation and geolocation services. The system’s use of Google Maps API and Supabase real-time tracking achieved a combined mean of 4.67 (*Excellent*), validating the accuracy, responsiveness, and reliability of the GPS-based features for both user types.
5. **Objective 5** tested the prototype’s overall functionality and usability. The UAT results indicated a 100 % pass rate across all test cases, while the System Usability Scale (SUS) yielded an overall score of 85.4, classified as *Excellent*. These findings confirmed that GoDavao was both functionally stable and highly usable according to international software quality benchmarks.
6. **Objective 6** evaluated GoDavao’s overall performance, usability, and effectiveness. The combined results from the UAT, SUS, and TAM analyses produced an overall mean of 4.74 (*Excellent*). Respondents expressed strong behavioral intention to continue using the app (M = 4.80), affirming its potential for long-term adoption.

Across all objectives, GoDavao achieved *Excellent* ratings in functional suitability, performance efficiency, reliability, and usability, as defined by **ISO/IEC 25010**. Furthermore, the results reinforced the **Technology Acceptance Model (TAM)** constructs of *Perceived Usefulness* and *Ease of Use*, both of which positively influenced user satisfaction and adoption intention.

In summary, GoDavao effectively met its design and performance goals. The findings validate the system’s feasibility as a localized ridesharing platform that promotes efficiency, safety, and accessibility for commuters in Davao City.

## **5.2 Conclusions**

Based on the findings of the study, the following conclusions were drawn:

1. The GoDavao application successfully addresses the most common transportation challenges faced by Davao City commuters through a reliable, secure, and easy-to-use ridesharing platform.
2. The user interface and navigation system were designed and implemented according to modern usability standards, ensuring accessibility and smooth interaction for both passengers and drivers.
3. The integration of map-based navigation and geolocation services effectively provided accurate, real-time location tracking, supporting efficient pickup and drop-off coordination.
4. The results of the UAT and SUS confirmed that GoDavao meets international software quality standards under ISO/IEC 25010, exhibiting *Excellent* ratings in functionality, usability, and performance.
5. The high mean scores for *Perceived Usefulness* and *Ease of Use* under the Technology Acceptance Model (TAM) validate that users found the system beneficial, efficient, and worth adopting.
6. Overall, the GoDavao prototype achieved all its research and system objectives, demonstrating readiness for pilot deployment and potential expansion as a sustainable ridesharing solution for Davao City.

These conclusions affirm that GoDavao is both a technical and social innovation, one that not only leverages information systems design but also addresses real-world urban mobility challenges through accessible technology.

## **5.3 Recommendations**

In light of the study’s findings and validated results, the following recommendations are proposed to further enhance GoDavao’s effectiveness and long-term sustainability. These recommendations are not intended as corrections but as *strategic enhancements* to build upon the project’s success.

1. **Pilot Deployment and Field Evaluation.**Conduct an extended pilot implementation involving actual commuters and drivers across multiple districts in Davao City to validate performance under real-world conditions. Data gathered from this phase can support system optimization and scalability planning.
2. **Integration with Local Transport Systems.**Collaborate with local government units (LGUs), transport cooperatives, and traffic management offices to align GoDavao with Davao City’s smart transport initiatives. Such partnerships can facilitate interoperability and improve route coverage.
3. **Advanced Data Analytics and Pricing Optimization.**Develop analytics dashboards for monitoring usage trends, demand clustering, and trip success rates. Implement dynamic pricing mechanisms to promote ride-sharing efficiency and fare fairness.
4. **Inclusion and Accessibility Features.**Add features supporting elderly and differently-abled users, such as larger interface elements, contrast modes, and optional voice navigation. This ensures equitable usability across all commuter demographics.
5. **Continuous System Monitoring and User Feedback.**Maintain iterative development cycles through regular updates, bug tracking, and post-deployment surveys. Continuous SUS testing and usability evaluations will sustain GoDavao’s alignment with ISO/IEC 25010 standards.
6. **Academic and Entrepreneurial Collaboration.**Encourage partnerships between academic institutions and technopreneurs to further develop GoDavao as a replicable model for localized, data-driven ridesharing systems in other Philippine cities.

In conclusion, the GoDavao project demonstrated the successful application of information systems design principles in solving real-world transportation problems. It stands as a validated and user-accepted prototype that merges technical functionality with social impact. Through its continued development, GoDavao has the potential to serve as a cornerstone for intelligent, community-centered mobility solutions in the Philippines.

## Appendix

**Appendix A**

Usability Acceptance Testing(UAT) Questions - Driver

| **Test Case ID** | **Feature / Module** | **Test Steps** | **Expected Result** | **Standard / Reference** |
| --- | --- | --- | --- | --- |
| D-TC01 | Driver Registration & Login | 1. Register as driver.  2. Log in with valid credentials. | The system authenticates the driver and loads the appropriate dashboard. | ISO/IEC 25010 (Usability); IEEE 829-2008 |
| D-TC02 | Route Creation & Publication | 1. Set route start and end points.2. Save route. | The defined route is successfully stored and displayed under “My Routes.” | Pressman & Maxim (2015); ISO/IEC 25010 |
| D-TC03 | Ride Request Notification | 1. Stay online with active route.  2. Wait for passenger request. | The system notifies the driver promptly when a compatible passenger request is found. | ISO/IEC 25010 (Performance/Responsiveness) |
| D-TC04 | Ride Acceptance & Matching | 1. Accept incoming passenger request.  2. Confirm trip start. | The trip status changes accordingly and the ride begins as scheduled. | ISO/IEC 25010 (Reliability) |
| D-TC05 | Real-Time GPS Tracking | 1. Start ride and move device.  2. Observe live tracking updates. | The GPS accurately tracks the driver’s movement and displays the correct route path. | ISO/IEC 25010 (Performance); Nielsen (1994) |
| D-TC06 | In-App Chat | 1. Chat with passenger during ride.  2. Send multiple messages. | All messages are transmitted and received without interruption or duplication. | ISO/IEC 25010 (Performance Efficiency) |
| D-TC08 | Driver Verification Badge | 1. Admin approves verification request.  2. Refresh profile. | The verification badge becomes visible once approved, confirming driver authenticity. | ISO/IEC 25010 (Security/Trust) |
| D-TC09 | Ride Status Workflow | 1. Accept > Start > Complete trip.  2. Verify status changes. | Each status transition is updated correctly and shown to both driver and passenger. | IEEE 829-2008 |
| D-TC10 | SOS / Emergency Response | 1. Passenger triggers SOS.  2. Observe admin log. | The admin panel receives an alert, and the incident is logged for recordkeeping. | ISO/IEC 25010 (Safety/Reliability) |
| D-TC11 | Ride History & Records | 1. Complete multiple rides.  2. View “Ride History.” | All finished trips appear with corresponding passenger details and timestamps. | IEEE 829-2008 |
| D-TC12 | System Responsiveness & Stability | 1. Perform consecutive rides and updates.  2. Monitor latency/crashes. | The system remains stable, responsive, and free from operational errors. | ISO/IEC 25010 (Performance Efficiency) |

**Appendix B**

Usability Acceptance Testing(UAT) Questions - Passenger

| **Test Case ID** | **Feature / Module** | **Test Steps** | **Expected Result** | **Standard / Reference** |
| --- | --- | --- | --- | --- |
| P-TC01 | Passenger Registration & Login | 1. Register as passenger.  2. Log in using valid credentials. | The user is authenticated successfully and the passenger dashboard loads correctly. | ISO/IEC 25010 (Usability); IEEE 829-2008 |
| P-TC02 | Passenger Profile Verification | 1. Upload ID image.  2. Wait for admin verification. | The verification badge becomes visible upon approval, indicating confirmed user identity. | ISO/IEC 25010 (Security/Trust) |
| P-TC03 | Ride Request Creation | 1. Input pickup and destination points.  2. Tap “Request Ride.” | The system records the request and notifies available drivers within the corresponding area. | ISO/IEC 25010 (Functional Suitability) |
| P-TC04 | Ride Matching | 1. Ensure a driver is available on same route.  2. Wait for matching. | The ride request is successfully matched and reflected in the passenger’s dashboard. | ISO/IEC 25010 (Performance Efficiency) |
| P-TC05 | Live GPS Tracking | 1. Observe driver location before and during ride.  2. Move or simulate driver location. | The map accurately reflects the driver’s movement and continuously updates in real time. | ISO/IEC 25010 (Reliability); Nielsen (1994) |
| P-TC06 | In-App Chat | 1. Open chat window for active ride.  2. Exchange messages with driver. | Messages are delivered promptly and displayed clearly for both parties. | ISO/IEC 25010 (Performance); Nielsen (1994) |
| P-TC08 | Ride Status Monitoring | 1. Observe status transitions (Requested > Matched > Ongoing > Completed). | Status changes are displayed correctly and synchronized between both user interfaces. | IEEE 829-2008 |
| P-TC09 | Ratings and Feedback | 1. Submit driver rating (1–5) and comment after trip.  2. View profile update. | The feedback is stored successfully and reflected in the driver’s profile. | ISO/IEC 25010 (Usability); McKnight et al., 2002 |
| P-TC10 | SOS / Emergency | 1. Tap SOS during active ride.  2. Verify log entry. | The system sends an emergency alert to the admin dashboard and records the incident details. | ISO/IEC 25010 (Safety/Reliability) |
| P-TC11 | Ride History | 1. Complete ride.  2. Open “My Rides.” | The completed trip appears in the ride history with accurate details. | IEEE 829-2008 |
| P-TC12 | System Responsiveness | 1. Navigate through pages.  2. Observe load speed and app stability. | The app maintains smooth performance with consistent stability and responsiveness. | ISO/IEC 25010 (Performance Efficiency) |

**Appenix C**

GoDavao Survey 2.0 Questionnaire and Summary of Responses

| **No.** | **Survey Question** | **Top Responses (Count)** |
| --- | --- | --- |
| 1 | Your participation is voluntary and your answers will be kept confidential. Do you consent to having your data being used as part of our research? | Yes (32) |
| 2 | Which category best describes you as a commuter of public transportation? | Daily commuter (work/school) (16); Occasional commuter (8); Driver (5); Personal vehicle (3) |
| 3 | What challenges do you currently face with your daily commute? | Anxiety about arrival times + Delays and inefficiency (13); Inconvenient multi-stops (10); No available commute options (6); Heavy traffic (3) |
| 4 | How likely would you consider using a ridesharing app if it can reduce your travel time through optimized routes and on-route pickups? | 5 (14); 4 (14); 3 (4) |
| 5 | How important is real-time driver availability (seeing drivers nearby on the map)? | 5 (22); 4 (7); 3 (3) |
| 6 | How important is it that ride-matching minimizes detours and passenger wait time? | 5 (23); 4 (8); 3 (1) |
| 7 | Would you prefer to share rides with multiple passengers if it lowers your fare but slightly increases travel time? | Maybe depending on the situation (18); Yes (8); No (6) |
| 8 | How do you feel about dynamic pricing (fares that change depending on demand, traffic, or distance)? | Acceptable with clear explanation (21); Fair and acceptable (11) |
| 9 | Would you be more accepting of dynamic pricing if a clear fare estimate is shown before you confirm the ride? | Yes (32) |
| 10 | Which payment options do you prefer for a ridesharing platform? | Cash & GCash (9); Cash (8); Cash + GCash + Credit/Debit (7); GCash (5); Credit/Debit (3) |
| 11 | How important is transparent fare breakdown (distance, time, additional fees)? | 5 (27); 4 (3); 3 (2) |
| 12 | Driver identity verification | 5 (30); 3 (1); 4 (1) |
| 13 | In-app emergency button | 5 (26); 4 (5); 3 (1) |
| 14 | Real-time ride tracking and sharing with contacts | 5 (26); 4 (5); 3 (1) |
| 15 | Rating and review system | 5 (22); 3 (6); 4 (4) |
| 16 | What safety concerns would stop you from using a ridesharing app? (Open-ended) | Bad drivers using the app/taxes; Concerns with other passengers; Driver’s record; Data privacy |
| 17 | How likely are you to adopt a mobile ridesharing app if it addresses your commuting challenges? | 5 (21); 4 (8); 3 (2); 1 (1) |
| 18 | How easy do you think it would be to use an app like GoDavao? | 5 (20); 4 (9); 3 (3) |
| 19 | Which features would you find most useful in a ridesharing platform? | Easy booking & scheduling; Multi-stop rides; Ride scheduling & fare estimate |
| 20 | What is the main reason you would consider carpooling? | Save money (20); Convenience (8); Safety (2); Environmental impact (2) |
| 21 | What would make your carpooling/ridesharing experience better? (Open-ended) | Efficient routes; Good drivers; Comfort; Verified users; Positive vibes |

## **References**

* AFI. (n.d.). *Using Route Optimization to Build a Ride Share Dispatch Algorithm*. Retrieved from<https://blog.afi.io/using-route-optimization-to-build-a-ride-share-dispatch-algorithm>
* Alisoltani, N., Zargayouna, M., & Leclercq, L. (2022). *A Shareability Clustering Method to Solve the Dynamic Ride-Sharing Problem*. HAL Archives. Retrieved from<https://hal.science/hal-03637134/document>
* Cao, Y., Wang, S., & Li, J. (2021). The Optimization Model of Ride-Sharing Route for Ride Hailing Considering Both System Optimization and User Fairness. *Sustainability, 13*(2), 902.<https://doi.org/10.3390/su13020902>
* Chen, L., Thakuriah, P., & Ampountolas, K. (2021). Short-Term Prediction of Demand for Ride-Hailing Services: A Deep Learning Approach. *Journal of Big Data Analytics in Transportation, 3*, 175–195.<https://link.springer.com/article/10.1007/s42421-021-00041-4>
* Codecademy. (n.d.). *Ridesharing Algorithms: Optimization and Iteration*. Retrieved from<https://www.codecademy.com/article/ridesharing-algorithms>
* Delling, D., Pajor, T., & Werneck, R. F. (2011). Round-based public transit routing. *Transportation Science, 45*(3), 432-449.<https://doi.org/10.1287/trsc.1110.0370>
* Dijkstra, E. W. (1959). A note on two problems in connexion with graphs. *Numerische Mathematik, 1*(1), 269-271.
* Econstor. (n.d.). *Passenger Demand Forecasting for Urban Bus Systems Using Machine Learning*. Retrieved from<https://www.econstor.eu/bitstream/10419/272424/1/Passenger-Demand-Forecasting-For-Urban-Bus-Systems-Using-Machine-Learning.pdf>
* FarEye. (n.d.). *What is Route Optimization Algorithm? How Does it Work?* Retrieved from<https://fareye.com/blog/route-optimization-algorithm/>
* Fu, L., Sun, D., & Rilett, L. R. (2006). Simulation for dynamic traffic assignment with real-time information. *Transportation Research Record, 1964*(1), 100-108.
* IEEE. (n.d.). *Real-Time Bus Arrival Prediction Using Machine Learning Models*. Retrieved from<https://ieeexplore.ieee.org/document/9648921>
* Javidi, H., Simon, D., Zhu, L., & Wang, Y. (2020). *A Multi-Objective Optimization Framework for On-Line Ridesharing Systems*. arXiv preprint arXiv:2012.05046.<https://arxiv.org/abs/2012.05046>
* Kim, S., et al. (2022). Predictive Analytics in Ridesharing: Enhancing Efficiency with Machine Learning. *Transportation Research Part C: Emerging Technologies*. https://www.sciencedirect.com/journal/transportation-research-part-c-emerging-technologies/vol/156/suppl/C
* Makhdomi, A. A., & Gillani, I. A. (2024). Fair and Efficient Ridesharing: A Dynamic Programming-based Relocation Approach. arXiv. Retrieved from<https://arxiv.org/abs/2401.15363>
* Masoud, N., & Jayakrishnan, R. (2017). A Real-Time Algorithm to Solve the Peer-to-Peer Ride-Matching Problem in a Flexible Ridesharing System. *Transportation Research Part B: Methodological, 106*, 218-236.
* MDPI. (2023). Predicting Ride-Hailing Demand with Consideration of Social Equity: A Case Study of Chengdu. Retrieved from<https://www.mdpi.com/2071-1050/16/22/9772>
* MDPI. (2024). Decoupling Online Ride-Hailing Services: A Privacy Protection Scheme Based on Decentralized Identity. Retrieved from<https://www.mdpi.com/2079-9292/13/20/4060>
* Nature.com. (2024). *Nature.com article on GIS-based Ridesharing Optimization*. (Note: Full citation details needed)
* Nguyen, T. T., Le, M. H., & Tran, P. N. (2019). The integration of GIS and demand forecasting in supply chain management. *International Journal of Logistics Systems and Management, 34*(2), 215-233.<https://doi.org/10.1504/IJLSM.2019.098765>
* ResearchGate. (2023). Matching and Network Effects in Ride-Hailing. Retrieved from<https://www.researchgate.net/publication/370996073_Matching_and_Network_Effects_in_Ride-Hailing>
* ResearchGate. (2025). ASSESSING THE URBANIZATION-DRIVEN CHALLENGES OF TRANSPORTATION SCARCITY AND COMMUTER ACCESSIBILITY IN DAVAO CITY: A SYSTEMATIC REVIEW. Retrieved from<https://www.researchgate.net/publication/387525353_ASSESSING_THE_URBANIZATION-DRIVEN_CHALLENGES_OF_TRANSPORTATION_SCARCITY_AND_COMMUTER_ACCESSIBILITY_IN_DAVAO_CITY_A_SYSTEMATIC_REVIEW>
* ResearchGate. (n.d.). (PDF) Drivers' Tactics in Ridesharing Economy in the Philippines. Retrieved from<https://www.researchgate.net/publication/336599982_Drivers'_Tactics_in_Ridesharing_Economy_in_the_Philippines>
* ResearchGate. (n.d.). (PDF) Trust in the Ride Hailing Service of the Sharing Economy: The Roles of Legitimacy and Process Transparency. Retrieved from<https://www.researchgate.net/publication/353142980_Trust_in_the_Ride_Hailing_Service_of_the_Sharing_Economy_The_Roles_of_Legitimacy_and_Process_Transparency>
* Sakthivelu, A., & Jayakrishnan, S. (2024). An Energy-Efficient Ride-Sharing Algorithm Using Distributed Convex Optimization. Stanford Digital Repository. Retrieved from<https://purl.stanford.edu/ss423gm7371>
* SciELO. (2019). DYNAMIC PRICING AND PRICE FAIRNESS PERCEPTIONS: A STUDY OF THE USE OF THE UBER APP IN TRAVELS PRECIFICAÇÃO DINÂMICA E PREÇO. Retrieved from<https://www.scielo.br/j/tva/a/Y6XjBk7z9N8f7mvTWnMygzP/?format=pdf&lang=en>
* Soria, J., Chen, Y., & Stathopoulos, A. (2020). *K-Prototype Segmentation Analysis on Large-scale Ridesourcing Trip Data*. arXiv preprint arXiv:2006.13924.
* Subramanian, D. (2020). *Clustering Uber Rideshare Data*. KDnuggets. Retrieved from<https://kdnuggets.com/2020/05/clustering-uber-rideshare-data.html>
* TRISTAN 2025. (2025). Privacy-Preserving Contextual Personalized Dynamic Pricing for Ride-Hailing Platforms 1 INTRODUCTION. Retrieved from<https://tristan2025.org/proceedings/TRISTAN2025_ExtendedAbstract_320.pdf>
* University of Southampton. (2024). Passengers as defenders. Retrieved from<https://eprints.soton.ac.uk/496349/1/Passengers_as_defenders_1-s2.0-S026151772400205X-main_Alex_Zarifis.pdf>
* UP NCTS. (2024). An Analysis of Ride-Hailing Services as Mode of Public Transportation in Olongapo City. Retrieved from<https://ncts.upd.edu.ph/tssp/wp-content/uploads/2024/12/TSSP2024-12-Revised-Paper.pdf>
* Wang, X., & Wei, H. (2018). GIS-based warehouse location planning for supply chain optimization. *Journal of Transportation Research, 56*(4), 102-118.<https://doi.org/10.1016/j.jtr.2018.06.004>
* Zhang, Y., & Xie, L. (2023). Real-Time Ride Matching Using Reinforcement Learning: A Case Study on Urban Transportation. *IEEE Transactions on Intelligent Transportation Systems*.https://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=10352976
* Zheng, Y., Wang, Q., Zhuang, D., Wang, S., & Zhao, J. (2023). Fairness-Enhancing Deep Learning for Ride-Hailing Demand Prediction. *IEEE Transactions on Knowledge and Data Engineering*.<https://doi.org/10.1109/TKDE.2023.3298960>
* Cherry, K. (2023). *The Color Psychology of Purple.* Verywell Mind.
* Kaya, N., & Epps, H. H. (2004). *Relationship between color and emotion: A study of college students.* College Student Journal.
* Wright, A. (1988). *The beginner’s guide to color psychology.* Colour Affects.
* Elliot, A. J., & Maier, M. A. (2014). *Color Psychology: Effects of perceiving color on psychological functioning in humans.* Annual Review of Psychology, 65, 95–120.
* Fitts, P. M. (1954). *The information capacity of the human motor system in controlling the amplitude of movement.* Journal of Experimental Psychology, 47(6), 381–391.
* Wertheimer, M. (1938). *Laws of organization in perceptual forms.* In W. Ellis (Ed.), A sourcebook of Gestalt psychology.
* Google (2022). *Material Design Guidelines: Layout & Spacing.* https://m3.material.io
* Nielsen, J. (1995). *10 Usability Heuristics for User Interface Design.* Nielsen Norman Group.
* Bhatia, T. (2020, September). Path Optimization for Destination-Oriented Ridesharing Drivers. Research Gate. <https://www.researchgate.net/publication/344364730_Path_Optimization_for_Destination-Oriented_Ridesharing_Drivers>
* Abeywickrama, T., & Liang, V. (2021, November 22). Using real-world patterns to improve matching in theory and practice. Grab Tech. <https://engineering.grab.com/using-real-world-patterns-to-improve-matching>
* Codecademy. (n.d.). Ridesharing Algorithms: Optimization and Iteration. Codecademy. https://www.codecademy.com/article/cf-cs-ridesharing
* Davis, F. D. (1989). *Perceived usefulness, perceived ease of use, and user acceptance of information technology.* MIS Quarterly, 13(3), 319–340. https://doi.org/10.2307/249008
* Venkatesh, V., & Davis, F. D. (2000). *A theoretical extension of the technology acceptance model: Four longitudinal field studies.* Management Science, 46(2), 186–204. https://doi.org/10.1287/mnsc.46.2.186.11926
* McKnight, D. H., Choudhury, V., & Kacmar, C. (2002). *Developing and validating trust measures for e-commerce: An integrative typology.* Information Systems Research, 13(3), 334–359. https://doi.org/10.1287/isre.13.3.334.81
* International Organization for Standardization. (2011). *ISO/IEC 25010:2011 – Systems and software engineering – Systems and software Quality Requirements and Evaluation (SQuaRE) – System and software quality models.* ISO.
* International Organization for Standardization. (2018). *ISO 9241-11:2018 – Ergonomics of human-system interaction – Part 11: Usability: Definitions and concepts.* ISO.
* IEEE Computer Society. (2008). *IEEE Standard for Software and System Test Documentation (IEEE Std 829-2008).* IEEE.
* Luxen, D., & Vetter, C. (2011). *Real-time routing with OpenStreetMap data.* Proceedings of the 19th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems, 513–516. <https://doi.org/10.1145/2093973.2094062>
* Dijkstra, E. W. (1959). *A note on two problems in connexion with graphs.* Numerische Mathematik, 1(1), 269–271.
* Sinnott, R. W. (1984). *Virtues of the Haversine.* Sky and Telescope, 68(2), 159.
* Boone, H. N., & Boone, D. A. (2012). *Analyzing Likert Data.* *Journal of Extension*, 50(2).
* IEEE. (2008). *IEEE Standard for Software and System Test Documentation (IEEE Std 829-2008).* IEEE Computer Society.
* International Organization for Standardization. (2011). *ISO/IEC 25010: Systems and Software Quality Requirements and Evaluation (SQuaRE).* Geneva: ISO.
* Likert, R. (1932). *A Technique for the Measurement of Attitudes.* *Archives of Psychology*, 22(140), 1–55.
* Nielsen, J. (1994). *Usability Engineering.* San Francisco: Morgan Kaufmann.
* Pressman, R. S., & Maxim, B. R. (2015). *Software Engineering: A Practitioner’s Approach* (8th ed.). New York: McGraw-Hill Education.
* GMA Regional TV News. (2024, November 11). *Flag-down rate for taxis in Davao now P50*. GMA Network. Retrieved from [https://www.gmanetwork.com/regionaltv/news/105000/flagdown-rate-for-taxis-in-davao-now-p50/story/ gmanetwork.com](https://www.gmanetwork.com/regionaltv/news/105000/flagdown-rate-for-taxis-in-davao-now-p50/story/?utm_source=chatgpt.com)
* SunStar Davao. (2024, November 11). *Taxi flag down rate now P50*. SunStar Publishing Inc. Retrieved from [https://www.sunstar.com.ph/davao/taxi-flag-down-rate-now-p50/](https://www.sunstar.com.ph/davao/taxi-flag-down-rate-now-p50/?utm_source=chatgpt.com) [SunStar Publishing Inc.](https://www.sunstar.com.ph/davao/taxi-flag-down-rate-now-p50?utm_source=chatgpt.com)
* Numbeo. (2025). *Taxi fares in Davao, Philippines*. Retrieved from [https://www.numbeo.com/taxi-fare/in/Davao Numbeo](https://www.numbeo.com/taxi-fare/in/Davao?utm_source=chatgpt.com)
* SunStar. (2022, September 28). *LTFRB-Davao: Fare hike effective Oct. 3*. SunStar. Retrieved from [https://www.sunstar.com.ph/davao/local-news/ltfrb-davao-fare-hike-effective-oct-3 SunStar Publishing Inc.](https://www.sunstar.com.ph/davao/local-news/ltfrb-davao-fare-hike-effective-oct-3?utm_source=chatgpt.com)
* The Philippine Star. (2023, October 4). *LTFRB OKs P1 hike in minimum PUJ fare*. Retrieved from [https://www.philstar.com/headlines/2023/10/04/2301009/ltfrb-oks-p1-hike-minimum-puj-fare philstar.com](https://www.philstar.com/headlines/2023/10/04/2301009/ltfrb-oks-p1-hike-minimum-puj-fare?utm_source=chatgpt.com)
* Brooke, J. (1996). *System Usability Scale (SUS): A quick and dirty usability scale.* In P. W. Jordan et al. (Eds.), *Usability evaluation in industry* (pp. 189–194). Taylor & Francis.
* Davis, F. D. (1989). Perceived usefulness, perceived ease of use, and user acceptance of information technology. *MIS Quarterly, 13*(3), 319–340.<https://doi.org/10.2307/249008>
* Gefen, D., Karahanna, E., & Straub, D. W. (2003). Trust and TAM in online services: An integrated model. *MIS Quarterly, 27*(1), 51–90. https://doi.org/10.2307/30036519
* Gumasing, M. J. J., et al. (2023). *The usability of TNVS apps in the Philippines (Grab, Angkas, JoyRide) using the System Usability Scale (SUS).* ACM Digital Library.
* Hsu, Y.-H., et al. (2023). *Usability study on the user interfaces of ride-hailing apps.* ACM Proceedings.
* ISO/IEC 25010:2011. *Systems and software engineering — System and software quality models.* International Organization for Standardization.
* Sauro, J., & Lewis, J. R. (2016). *Quantifying the user experience: Practical statistics for user research* (2nd ed.). Morgan Kaufmann.
* Grab Philippines (2024). *GrabCar Fare Guide: How Fares Are Computed.* Retrieved from<https://www.grab.com/ph/>
* Land Transportation Franchising and Regulatory Board (LTFRB). (2023). *TNC Fare Matrix and Dynamic Pricing Policy Circular 2023-016.* Manila, Philippines.
* JoyRide Philippines (2024). *JoyRide Fare Structure 2024.* Retrieved from https://www.joyride.com.ph/
* Angkas (2023). *Official Fare Matrix as Approved by LTFRB.* Retrieved from<https://www.angkas.com/>
* Maxim PH (2024). *Fare Estimation Guidelines.* Retrieved from<https://taximaxim.com/ph>
* Shaheen, S., Chan, N., & Gaynor, T. (2019). *Shared Mobility: Current Practices and Future Directions.* U.S. Department of Transportation.