UTM
UNIVERSITI TEKNOLOGI MALAYSIA

SECD2523
DATABASE
SECTION 06
LECTURER: DR. LAYLA RASHEED ABDALLAH HASAN

BUSFLEX
BUS BOOKING SYSTEM

Phase 1- Project Proposal

Name	Matric No
Chuah Hui Wen	A23CS0219
Tan Xin Ying	A23CS0276
Anjum Siddiqua Tanveer Siddiqui	A23CS0289
Mathaba Hassan Mohamed Hassan	A23CS4044
Alwaeli Amr Khaled Abdo	A23CS4004

TABLE OF CONTENT

1.0 Introduction

With the rapid advancement of today's digital era, every business from all walks of life tries to find an easier way to ease its operations and improve customer experience. Among them, transportation services are very crucial to guarantee smooth people's movement, and companies like JMY Transport Malaysia have increasing demands for improved service delivery. The current system for JMY Transport Malaysia uses a manual bus booking system to manage customers' booking orders and order payments. This will be inefficient, with probable errors, as the number of orders goes up.

This will make quite a difference in terms of establishing a competitive advantage and meeting the growing demands of the clientele. It is now necessary to migrate towards an automated integrated facility setup. In enhancing the existing bus rental system at JMY Transport Malaysia, we look for an overall optimization of operations that give customers a better experience. This proposal describes the development that will optimize the booking process and, importantly, enhance the company's capacity to manage increased demand and support business growth through this digital solution.

2.0 Background Study

JMY Transport Malaysia is a company which provides a bus booking system for various university events such as student organization activities, athletic and academic competitions, conferences and trips. Currently, JMY Transport Malaysia is still using the manual bus booking system.

The existing manual bus rental system requires users to rent buses by contacting the system's staff through WhatsApp. Firstly, customers have to get the staff's contact through JMY Transport Malaysia facebook account. Then, customers should WhatsApp the staff and provide renter's personal details such as name, age, gender, occupation, IC number, and rental details such as date of booking and bus model. After getting the customers' details, the staff of JMY Transport Malaysia will record everything manually on papers. Next, the customer should pay a deposit through online banking and send a copy of receipt to the staff. Customers can also pay their tickets at the counter by cash payment. Lastly, the staff will verify all the details and process the booking.

With the current manual bus booking system used by JMY Transport Malaysia, it is very time consuming, This is because that manual bus booking system involves a lot of paperwork and Whatsapp messaging. This can waste the time of both the staff and the customers. Besides that, the current manual bus booking system will cause a higher error rate. Manual record of the staff will cause errors such as double bookings, incorrect pricing, and scheduling conflicts. These errors can lead to customer dissatisfaction and financial losses for JMY Transport Malaysia. Hence, our team decided to improve the manual bus booking system.

The proposed BusFlex online system will provide solutions to the operational inefficiencies of the existing bus rental system through digitization and smoothing of the process. The features incorporated into the design include automated bookings, real-time notifications, live chat support, and management of finance records.This will not only enable added customer experience in terms of convenient booking and access to bus models with their prices but also help to modernize JMY Transport Malaysia's way of carrying out operations: seamless workflow and higher customer satisfaction for long-term business growth.

3.0 Problem Statement

1. Hard to record the booking manually by using book or paper written

Staff of JMY Transport Malaysia are required to hand record on paper the customers' pick-up and drop-off locations, times, dates, and seat numbers. As a result, when customers wish to book a bus, staff must verify that the bus is still available by consulting the written record on paper. This will be very time-consuming and lead to unhappy customers. It is inconvenient for both staff and customers.

2. Staff forgot to send notification manually to customers

There is no application or online system for JMY Transport Malaysia's bus booking service that allows users to get pop-up message reminders. Because of this, staff of JMY Transport Malaysia must manually message each customer individually when their travel date approaches, informing them of their bus number via Whatsapp. Certain customers may not receive notifications about their bus numbers from JMY Transport Malaysia staff when there is an excessive volume of bus booking orders or when staff are overworked.

3. Miscommunication

If the bus departure time or location is changed suddenly, there will be a difficult situation. Customers might not be informed of any scheduling changes if they are not actively checking their messages or if they have given a wrong phone number. Customers may find it difficult to arrange their plans or may get dissatisfied because of the delay's of the bus departure time. It is not convenient for the staff members to always have their phones turned on in order to avoid missing any orders or questions.

4. Repeated questions asked from customers

Because customers can only obtain information through JMY Transport Malaysia's Facebook page, clients have restricted access to bus information. As a result, JMY Transport Malaysia staff are always required to respond to frequently asked inquiries. Customers frequently want to know things like the type of bus, how much it costs to get to certain well-known locations, whether seats are available, and what paperwork is needed in order to purchase a bus ticket.

5. Wrong location for pick up and drop off

Staff of JMY Transport Malaysia may enter orders incorrectly since they are required to manually document every order from customers. The bus driver could go to the incorrect place if the staff records the booking incorrectly, such as recording the incorrect pick-up or drop-off location. As a result, customers who are supposed to be picked up or left off at the specified places can end up somewhere else, which could cause problems like delays and missing buses. Customers will be upset and dissatisfied by these problems.

6. Calculating the profit margin wrongly

Online bus service booking system is not available from JMY Transport Malaysia. Thus, JMY Transport Malaysia does not have a system to save the details directly once the customers keyed in their details. So, during the end of the month, when the company wants to calculate the expenses and profit earned in the entire month, the employees and administrators need to manually key in every detail in an Excel worksheet or spreadsheet. There is too much information to key in, therefore this method takes a long time. Additionally, there is a chance that administrators and employees would key in incorrect data throughout the information entry process, which could result in inaccurate profit and expense calculations.

4.0 Proposed Solutions

Booking errors, data management problems, and delays in customer service are some of the operational inefficiencies caused by JMY Transport Malaysia's current offline manual bus booking system. We are trying to propose a new online bus booking system solution called BusFlex, which incorporates some of these technologies.

First, an automated booking and management system will be created so that customers can finish the entire booking process, from the first inquiry to the final confirmation. The BusFlex online bus booking system will handle payment processing, ticket issuance, and seat assignment automatically. Customers can easily browse various bus models and rental prices with the bus rental system. By simplifying the booking process and eliminating the need for manual data entry, BusFlex online bus rental system can help to improve customer satisfaction by operating more effectively. This can solve the problem of staff hard to record the booking manually by using a paper written method.

Next, an automatic notification system will be made available via BusFlex. Customers can receive updates and reminders immediately via email and SMS by using this notification system. Important details such as schedule modifications, updates on the status of reservations, and information about current promotions will all be included in these alerts. By providing customers with up to date information, maintaining continuous communication, BusFlex can help to prevent problems of staff forgetting to send notifications manually.

In addition, BusFlex will have a live chat option. Customers can receive support and response from the staff of JMY Transport Malaysia. The frequently asked questions can be minimized by integrating a live chat option and a dynamically updated FAQ section on the BusFlex website. Additionally, by doing this, staff can be freed up to concentrate on more important tasks and customer service duties. As a result, miscommunication problems can be solved.

BusFlex will also provide a real time update system which updates the details of each bus model availability and the booking fee of each bus model. As a result, the problem of repeated

questions for customers due to cannot find details of the bus model easily can be prevented. If the bus pick up location or time has any changes, this real time update system will notify the customers and drivers too. Hence, it can prevent problems like the bus and customers from going to the wrong location for pick up and drop off.

Furthermore, the BusFlex system will provide Financial Management and Reporting functions like automatically calculate the total and creating booking receipts for customers. Every model bus's rental cost will be entered into the BusFlex system. This could improve the accuracy of determining the total amount that the customers must pay and can help to streamline financial management. The automated tracking of the total cost of bus rental for each customer's order is possible in the BusFlex system by integrating a financial management module into BusFlex. This can help to prevent calculating the profit margin wrongly.

Additionally, BusFlex offers easy accessibility features and an user-friendly interface. The BusFlex platform will be made as user-friendly as possible so that the BusFlex system can be used by people of all skill levels, including those with limited tech skills. Customers can quickly check available bus models and rental prices using BusFlex. This could improve the service's quality, make sure that the system is usable by a larger audience, and raise customer happiness and satisfaction.

BusFlex is dedicated to providing customers with a secure system hosting experience. In order to access the services provided, customers must create an account and set a password. Additionally, during registration, BusFlex also prompts customers to set a security keyword, which can improve customers' account security. Security questions and password protection can also help JMY Transport Malaysia in safeguarding confidential rental information.

4.1 Technical Feasibility Study

The proposed bus booking system, BusFlex, is technically feasible with the tech we have today. We'll use familiar web development tools like HTML, CSS, JavaScript. The BusFlex system website will also be using SQL for the database management. This stack is well-documented and suitable for creating scalable web applications.

4.2 Operational Feasibility Study

An Information System (IS) support team is needed to operate the BusFlex system website because it has many features like a database system, sorting system. The system will handle bookings and ticket issues automatically, cutting down on mistakes and making our process faster. The support team must also provide support and solutions to the customers who have any problems.

4.3 Economic Feasibility Study

Cost-Benefit Analysis (CBA)

Assumptions	
Discount Rate	**10%**
Sensitivity Factor (costs)	**1.1**
Sensitivity Factor (benefits)	**0.9**
Annual change in production costs	**7%**
Annual change in benefits	**5%**

Estimated Costs	
Hardware	RM 10,000
Software	RM 10,000
Maintenance	RM 3500 annually
Marketing	RM 6000 annually
Staff Salary	RM 35,000 annually

Estimated Benefits	
Increase in Sales	RM 50,000
Savings	RM 25,000

Costs	Year 0	Year 1	Year 2	Year 3
Development Costs:				
Hardware	11 000			
Software	11 000			
Total	22 000			
Production Costs:				
Maintenance		3850	4120	4408
Marketing		6600	7062	7556
Staff Salary		38 500	41 195	44 079
Annual Production Cost		48 950	52 377	56 043
Present Value (PV)		44 500	43 287	42 106
Accumulated Costs		66 500	109 787	151 893

Benefits	Year 0	Year 1	Year 2	Year 3
Increase in Sales		45 000	47 250	49 613
Savings		22 500	23 625	24 806
Annual Total Benefits		67 500	70 875	74 419
Present Value (PV)		61 364	58 500	55 912
Accumulated Benefits		61 364	119 864	175 776
Gain or loss		(5136)	10 077	23 883
Profitable Index (PI)	1.09			

Assumptions:

1-Discount Rate: This is a software project with moderate risk, we'll use a 10% discount rate.

2-Sensitivity Factor: Adjusting for potential volatility in costs and benefits due to external factors such as market fluctuations or technological changes:

- Costs: Increased by 10% to account for possible overruns in development or operational costs.
- Benefits: Reduced by 10% to account for uncertainties in the projected increase in sales or operational savings.

3-Annual Change:

- Production Costs: Increase by 7% annually, reflecting typical inflation and rising operational costs.
- Benefits: Increase by 5% annually, a conservative estimate considering improved efficiencies and market growth.

Estimated Costs:

1-Development Costs:

- Hardware: GPS devices, servers, and computers might cost around RM 10,000.

- Software Development: Custom software development costs around RM 10,000.

2-Production Costs:

- Maintenance: Regular maintenance of the buses and software systems RM3500 annually.
- Marketing and promotional activities: around RM 6,000 annually
- Salaries for staff: salaries around RM 35,000 annually

Estimated Benefits:

Increase in sales from enhanced booking capabilities and market reach: RM 50,000 annually

Operational savings from automation (reduced manpower needs, error reduction): RM 25,000 annually

5.0 Objectives

1. **Automate Booking Recording and Seat Reservation to eliminate the need for manual entry.**

   Develop a digital application to automate the recording of customer booking, including pick-up/drop-off locations, dates, times, and seat numbers. This will eliminate manual booking record-keeping and improve customer satisfaction.

2. **Implement Automated Notification System to ensure timely delivery of notifications**

   Integrate an automated notification system that sends reminders and updates to customers via SMS or email. This ensures timely communication, reducing the possibility of missed notifications due to staff error.

3. **Enable Real-Time Communication of Schedule Changes to avoid miscommunication**

   Provide a feature to instantly notify customers of any changes to bus schedules, including departure times or locations. This will minimize miscommunication and ensure that customers receive updates promptly, regardless of staff availability.

4. **Create a Centralized FAQ and Information Portal to avoid repeated question**

   Include a customer self-service portal where users can access frequently asked questions and information about bus types, costs, seat availability, and required documentation. This will reduce repetitive inquiries and save staff time.

5. **Minimize Errors in Location**

   Incorporate validation mechanisms and geolocation features to accurately capture and confirm customer-provided pick-up and drop-off locations. This will reduce the possibility of incorrect data entries that cause customer dissatisfaction.

6. **Automate Profit Margin Calculation and Reporting to ensure the accuracy of financial reporting**

   Implement a financial module that tracks rental transactions in real time and automatically calculates monthly profit margins, expenses, and revenue. This will reduce manual data entry, save time, and ensure accurate financial reporting.

6.0 Scope of the project

1. User Sign-up & Security:

    - Users must provide their name, email, password, and phone number to sign up.
    - They will choose a security question and provide an answer to enhance account security.

2. User Categories:

    - Customers: Can book buses, view bus details, and make payments.
    - Staff: Manage bus inventory, customer profiles, orders, and income analysis.

3. Customer Features:

    - Select from different bus models (19-seater, 25-seater, 40-seater, 44-seater).
    - Check bus availability and rental fees.
    - Complete a booking with pick-up dates, times, addresses, and the number of passengers.
    - Proceed to payment and receive an invoice after confirming the booking.

4. Staff Features:

    - View and manage customer profiles (name, contact number).
    - Manage bus information (update availability to reserved when rented).
    - Approve customer bookings after payment.
    - Analyze company income through reports.

5. System Focus:

- Customer satisfaction: Simple booking process and secure payments.
- Reliability & safety: Secure accounts and bus availability management.
- Efficiency for staff: Easy inventory, order management, and income tracking.

6. Target Users:

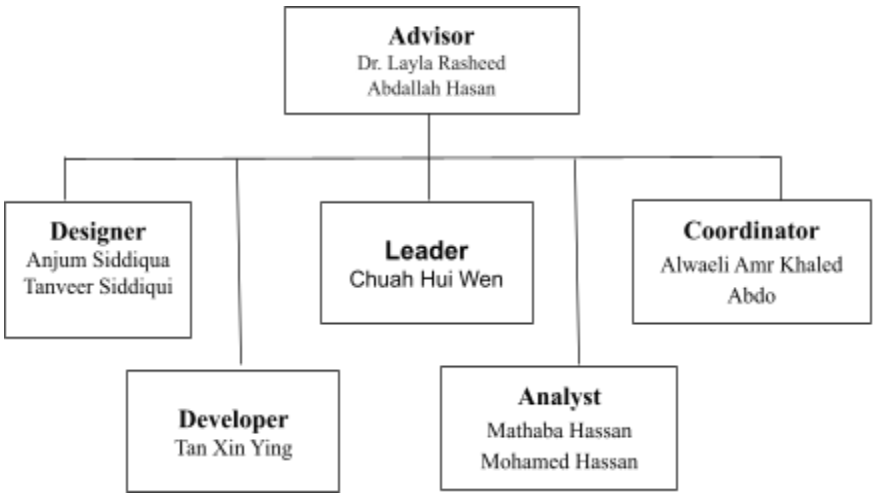
- The target users are university students who may want to rent buses for industrial visits and cultural tourism. Working professionals who might go on a company tour. Families that want to get together in one place for weddings or tours.

7. Compatible Platforms:

- The system will be compatible with **iOS** and **Android** platforms, making it accessible on smartphones and tablets.

7.0 Project Planning

7.1 Human Resource



Project Leader: Chuah Hui Wen

- Plans and schedules project tasks
- Coordinates team members and resources

Designer: Anjum Siddiqua Tanveer Siddiqui

- Ensures designs meet project requirements
- Designs user interfaces and experiences

Developer: Tan Xin Ying

- Writes and tests code
- Conducts testing and debugging

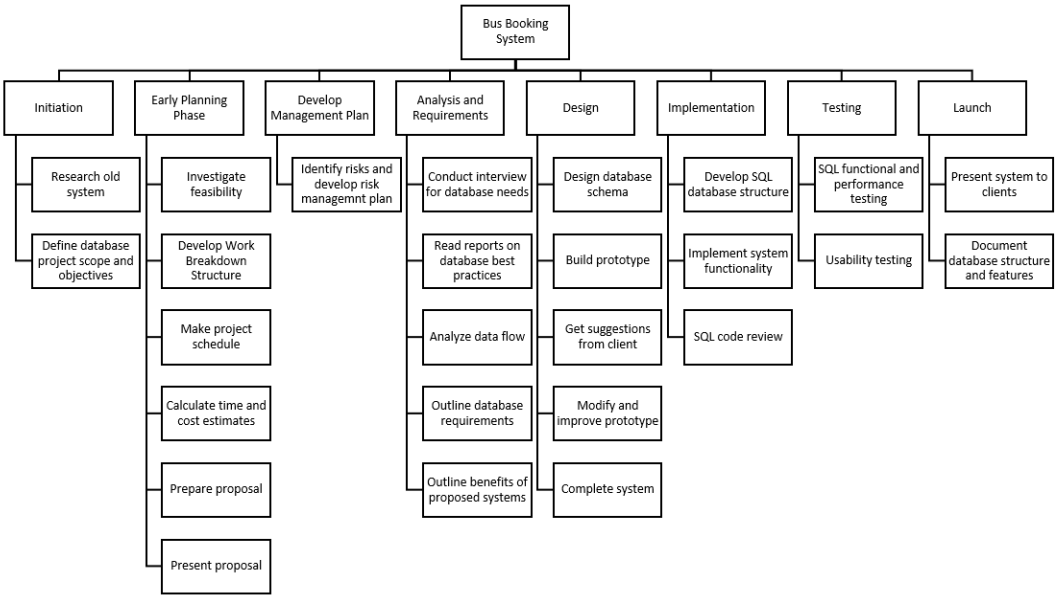
Analyst: Mathaba Hassan Mohamed Hassan

- Reviews plans and progress
- Provide advices to the team

Coordinator: Alwaeli Amr Khaled Abdo

- Compiling progress reports
- Gathering feedback

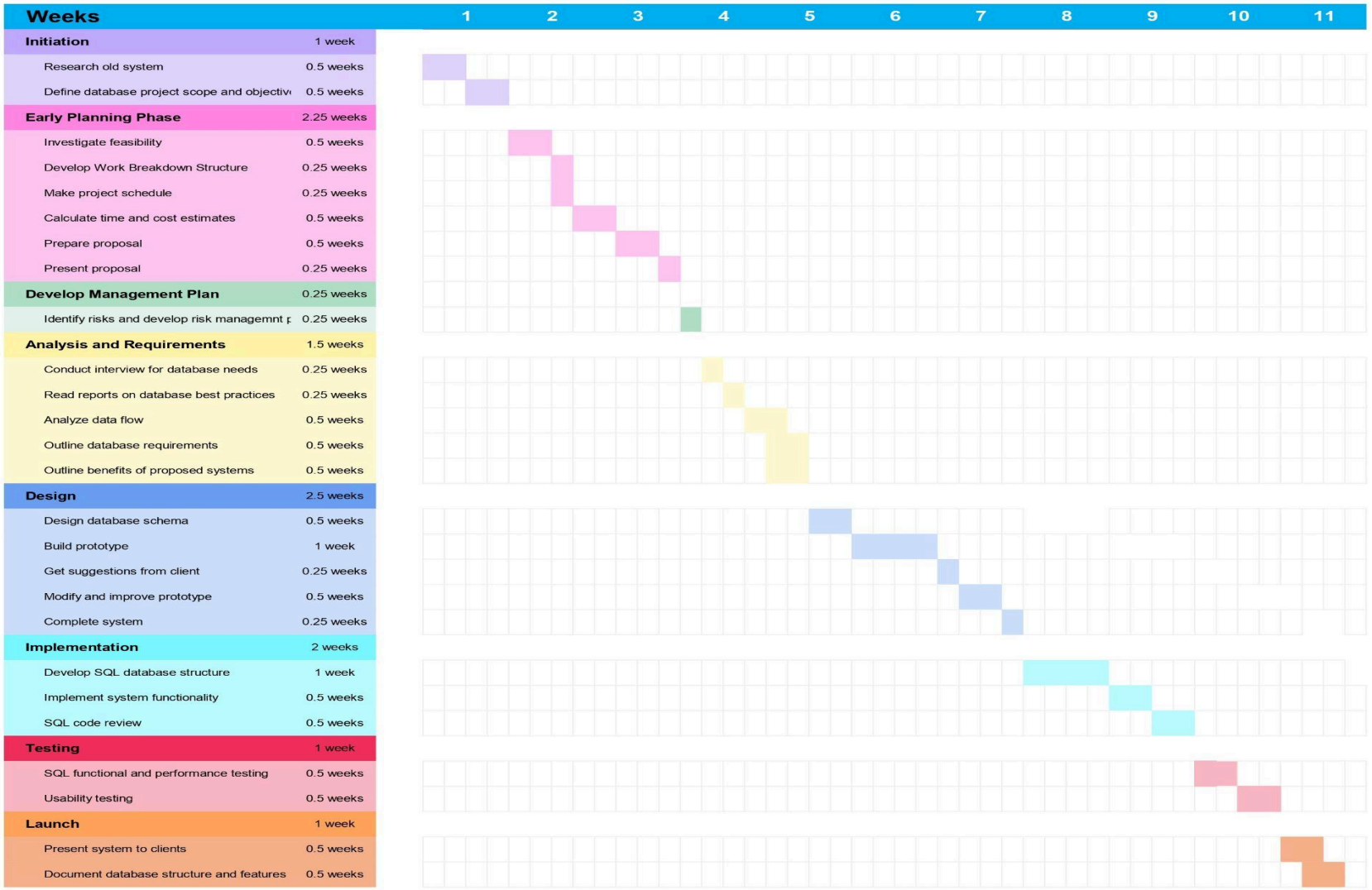
7.2 Work Breakdown Structure (WBS)



WBS stands for the Work Breakdown Structure, which is a project management deliverable that breaks down a project into small pieces of work or tasks that are relatively small. The aim of a WBS is organizing and defining the total scope of a project, which subsequently assists in managing, scheduling, and tracking the progress of every part of a project. WBS addresses deliverables or outcomes rather than just activities. Each level represents a more detailed breakdown of the project deliverables.

Level 1 of WBS shows the overall project which is the bus booking system. Level 2 shows the major deliverables of project phases. This project includes 8 phases which are initiation, planning, development management plan, analysis, design, implementation, testing and launch. Level 3 shows sub-deliverables or detailed components of each phase.

7.3 Gantt Chart



Gantt Chart in weeks for Bus Booking System

The Gantt chart for this project, which focuses on the bus booking system, is broken down into eight key phases: Initiation, Early Planning, Development Management Plan, Analysis, Design, Implementation, Testing, and Launch. Each phase is further divided into a series of specific tasks or milestones.

- **Initiation phase**: This phase consists of the research and setting the scope of the project, which will take 1 week. The deliverables here are the completion of research and defining the project scope and objectives.
- **Early Planning phase**: Spanning the duration of 2.25 weeks, this phase includes tasks such as developing the work breakdown structure (WBS), creating a project proposal, and outlining a timeline.
- **Development Management Plan phase**: Which we assigned only 0.25 weeks for, this phase focuses on creating a detailed management plan for guiding the project's execution and control.
- **Analysis and Requirements phase**: Taking up to 1.5 weeks, the tasks in this phase include gathering requirements, conducting risk analysis, and collecting data for analysis.
- **Design phase**: Which we assumed will take the longest, 2.5 weeks, this phase encompasses designing the project architecture, including creating prototypes and ensuring the database and code align with the project needs.
- **Implementation phase**: Which will take 2 weeks, this phase focuses on coding, system setup, and integrating all components into a functional system.
- **Testing phase**: Which will be done in 1 week, the testing phase ensures the functionality of the system through unit testing, system integration testing, and bug fixing.
- **Launch phase**: Which will be done in 1 week, this final phase includes the launch of the system, user training, and delivering documentation.

Overall, we estimate that the whole system will take approximately **11 weeks** to complete.

8.0 Requirement Analysis (based from AS-IS analysis)

8.1 Current business process (scenarios, workflow)

The scenarios and workflow of the current offline bus booking system for customers is as below:

1. Customers contact the admin of the bus booking system through Whatsapp or Facebook.
2. Customers will fill in a template for the bus booking information from the admin

   2.1. Customers will fill in the date, departure time, number of passengers, passenger nationality, departure destination and drop off destination.
3. Once received staff's confirmation, there will also be the price for payment.

   3.1 If a customer chooses to pay by online banking, the customer needs to email or send the invoice as a PDF to the admin.

   3.2 If the customer chooses to pay by e-wallet such as Touch and Go e-wallet, the customer needs to screenshot the details after the payment.
4. After payment is confirmed, receipt and bus ticket will be received from the staff.

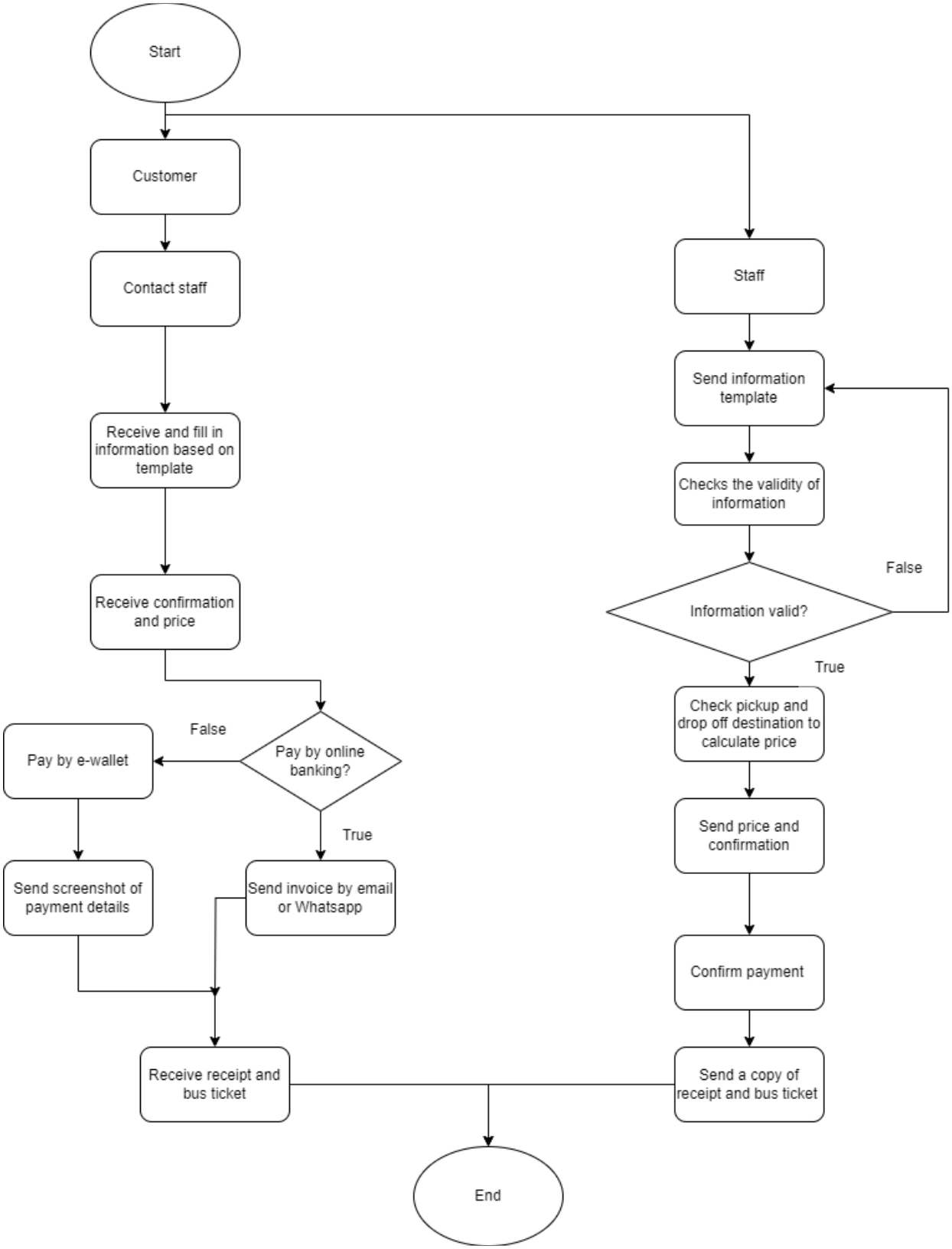
The scenarios and workflow of the current offline bus booking system for staff is as below:

1. Staff receives messages from the customer from Whatsapp or Facebook.
2. Staff sends the template of the information required for bus booking.
3. Staff checks the validity of the customers' information.

   3.1 If the customers' information is complete, check on the pickup and drop off destination to calculate the price.
4. After the price is calculated, send the price to the customer so that the customer can make payment.

   4.1 If a customer chooses to pay by online banking, the staff needs to receive an invoice of the payment by email or Whatsapp.

   4.2 If the customer chooses to pay by e-wallet such as Touch and Go e-wallet, the staff needs to receive the screenshot of the details of the payment.
5. After payment has been confirmed, the staff will send a copy of receipt and bus ticket to customers.



Current Offline Manual Bus Booking System Workflow

9.0 Transaction requirement (data entry, data update/delete, data queries)

9.1 Data Entry

1. **Enter Customer Information**
   Add new customer records with details like name, ID, age, gender, contact information, email, password, and security questions.

2. **Enter Bus Information**
   Add bus records including bus model, seating capacity, bus booking price, and availability status.

3. **Enter Booking Information**
   Record booking details such as booking date, pickup and drop-off locations, number of passengers, and total booking price.

9.2 Data Update/Delete

1. **Update/Delete Customer Information**
   Modify or remove existing customer records.

2. **Update/Delete Bus Information**
   Modify or remove bus details, including model, capacity, price, and availability.

3. **Update/Delete Booking Information**
   Modify or cancel bookings, including changes to pickup/drop-off locations and booking dates.

4. **Update/Delete Payment Status**
   Update payment statuses (e.g., pending, completed) and modify payment methods if necessary.

9.3 Data Queries

1. **Display Available Bus**

   Show a list of all buses available for booking, including their seating capacity and booking price.

2. **Display Booking Information**

   Retrieve and display details of bookings, including customer information, booking date, and status.

3. **Display Customer History**

   Show a customer's booking and payment history.

4. **Display Payment Details**

   Retrieve payment details, including total amount paid, payment method, and status.

5. **Generate Income Analysis Report**

   Display income, expenses, and calculated profit based on completed bookings.

10.0 Benefit and Summary of Proposed System

BusFlex is a bus booking system that can effectively streamline the bus booking process, enhance user experience, and provide convenience for both customers and staff. BusFlex's bus booking system is developed to replace the existing manual system which is used by JMY Transport Malaysia. The existing manual system used by JMY Transport Malaysia requires customers to message staff through WhatsApp to book a bus. Unlike the existing manual system used by JMY Transport Malaysia, BusFlex enables customers to book a bus online. It also allows the staff to easily manage the booking of customers online.

BusFlex can also enhance customer experience. The bus booking system provides a user-friendly interface that allows customers to view available bus models and booking fees easily. Then, customers can proceed to the booking page, select the desired bus model, and input the details. Customers can pay using different methods such as online banking, touch-n-go payment, or credit card. That helps customers to save their time when using the BusFlex system.

In addition, BusFlex improves the overall operational efficiency. This is because staff can efficiently manage bus inventory, track availability status, and update bus information in real-time. Staff can also easily view and manage booking orders, assign bus plate numbers and track order status. This can help staff to communicate with customers smoothly. BusFlex also provides automatic generation of invoices for each rental order. This can reduce the workload of the staff and prevent errors in billing processes.

Besides, BusFlex can ensure the safety of the customer and staff's personal information and bus booking order details. User registration with password protection and security questions enhances the security of customer accounts. Password protection and security questions can also help the staff protect sensitive rental data. As a result, it can gain trust among customers and staff who are using the BusFlex system.

Furthermore, BusFlex is more convenient compared to the manual bus booking system. During weekends or public holidays, the workers will not be working, so the manual bus rental system will not be available during these times. However, BusFlex allows customers to access

the booking system at any time. For example, customers can make bookings and inquiries outside of traditional business hours, they can also make bookings on weekends and public holidays. This enhances convenience and accessibility.

Overall, BusFlex is a good bus booking system. It solves the problem of manual systems that require the customers to message staff through WhatsApp to book a bus. It also solves the problem of the staff needing to record the order details manually. BusFlex's bus booking system offers a comprehensive solution for managing the entire bus booking process smoothly, from booking to payment. At the same time, BusFlex can ensure the safety and reliability of all users. BusFlex also provides user-friendly interfaces and with its everyday accessibility, it can also provide customer satisfaction. By automating invoicing, the system also helps to improve overall operational efficiency.

11.0 Summary

The analyses and subsequent proposal for the BusFlex system have provided an in-depth understanding of the current manual bus booking system that is currently being practiced by JMY Transport Malaysia. Through mapping workflows in customer inquiries, bookings, and payments, for example, key inefficiencies and complexities could be noted in present systems. We were able to highlight the shortfalls of the current system, such as manual entry being very time-consuming and there not being any updating in real time, through documentation. Business needs, such as those stated here, will help guide us toward an automated solution.

The automation of BusFlex in bookings, with real-time notifications of any changes and financial management, has accorded us the ability to indicate where the present manual system can be improved. This solution addresses miscommunication, errors in bookings, and dissatisfaction among clients to ensure that the workflow is rendered more efficient, easing the staff burden and enhancing the general user experience.

Through this process, we were able to learn a great deal on how best to manage JMY Transport Malaysia. Also, by documenting the AS-IS analysis and the proposed solutions in detail, we gained a very good foundation for designing a robust system to manage bus bookings efficiently and also to grow its business.

UTM
UNIVERSITI TEKNOLOGI MALAYSIA

SECJ2203
DATABASE
SECTION 06
LECTURER: DR. LAYLA RASHEED ABDALLAH HASAN

BUSFLEX
BUS BOOKING SYSTEM

Phase 2 - Database Conceptual Design

Name	Matric No
Chuah Hui Wen	A23CS0219
Tan Xin Ying	A23CS0276
Anjum Siddiqua Tanveer Siddiqui	A23CS0289
Mathaba Hassan Mohamed Hassan	A23CS4044
Alwaeli Amr Khaled Abdo	A23CS4004

Table of Contents

1.0 Introduction

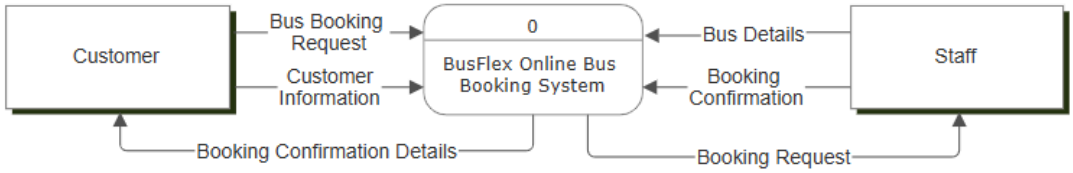
In today's busy world, having a good booking system is really important for businesses that want to keep up with customer needs. A lot of traditional ways of making reservations aren't very effective anymore. They often lead to double bookings, missed appointments, and trouble with keeping track of available times, which can make things difficult for both the business and the customers.

A reliable booking system makes it easier for customers to make reservations quickly and easily. While it also helps businesses stay organized. This kind of system keeps everything running smoothly by letting customers book with confidence and by helping staff manage appointments better. With a good booking system in place, daily operations get easier, scheduling issues are minimized, and customers end up happier. Overall, a well-designed booking system not only improves the reservation process but also helps businesses build strong, long-lasting relationships with their customers.This will result in improved profits the business makes.

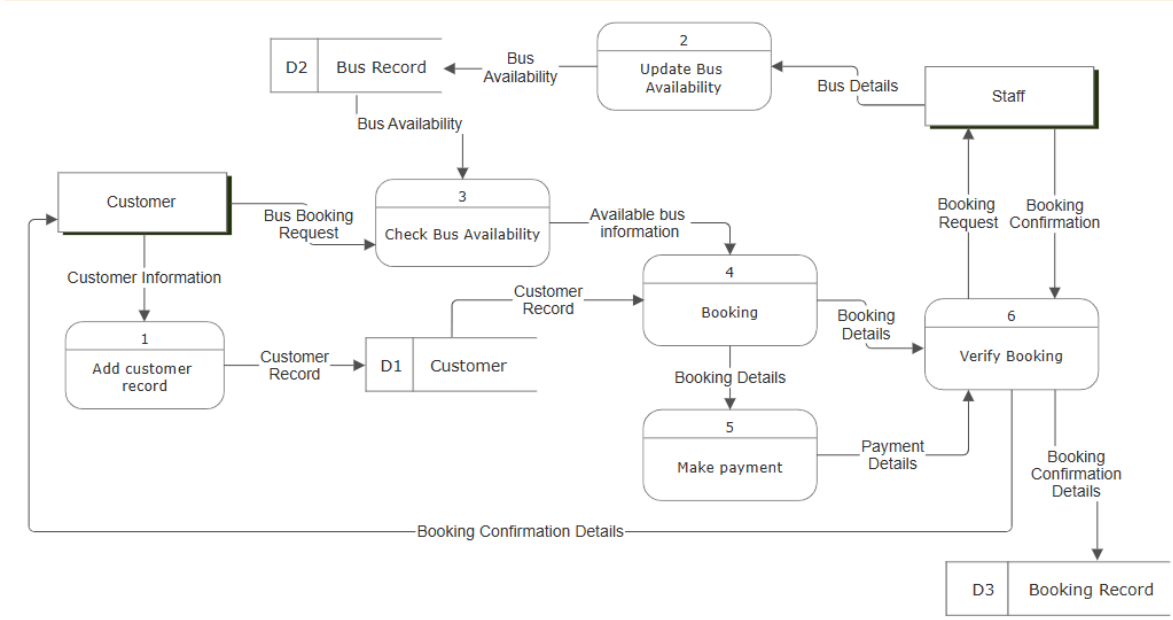
In this phase we will understand the process of the booking systems and its relationships deeply using ERD's. We will set the business rules required for this system. The data requirements will also be analyzed and the data type to use for each and every entity will be described in a detailed manner. This phase will set the basis for the data modeling and will give us a clear understanding of all the data related concepts.

2.0 Data Flow Diagram (DFD)

2.1 Context Diagram

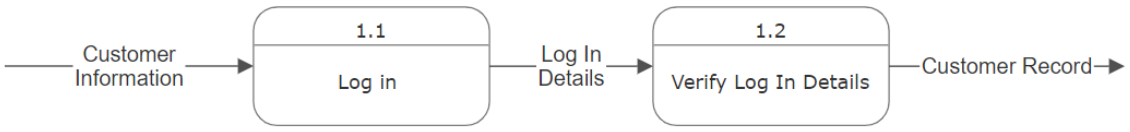


2.2 Parent Diagram (Level 0)

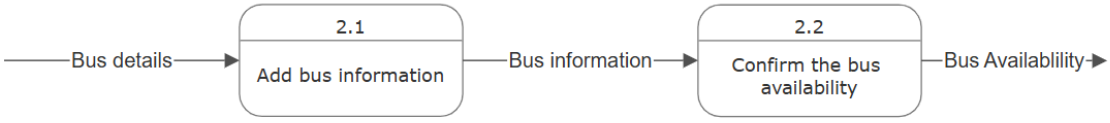


2.3 Child Diagram (Level 1)

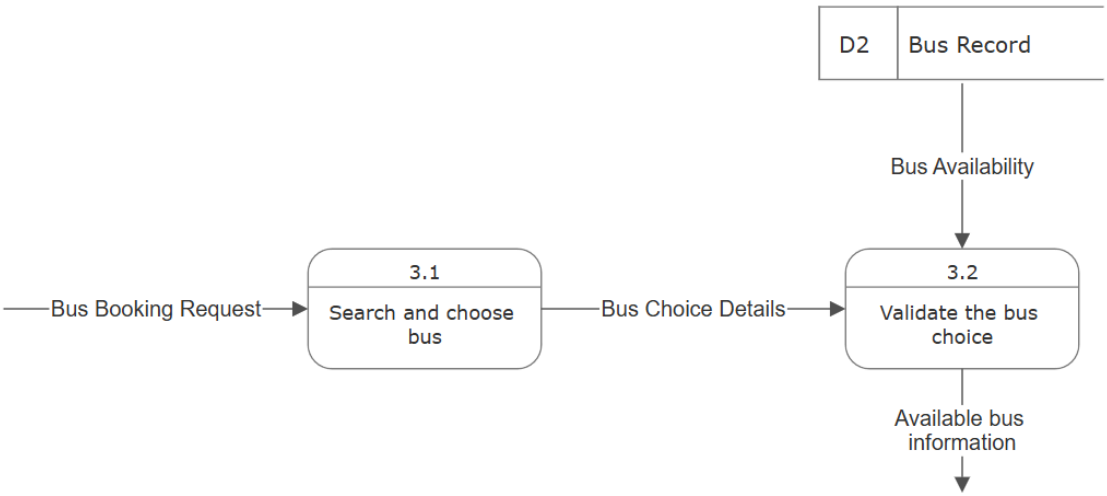
2.3.1 Process 1.0 <Add customer record >



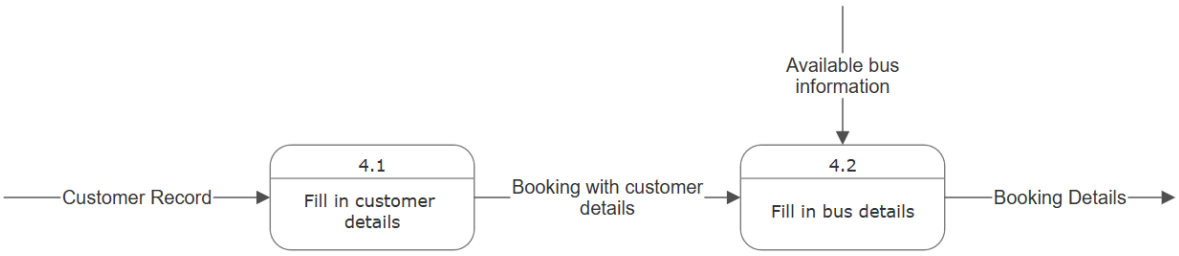
2.3.2 Process 2.0 <Update Bus Availability >



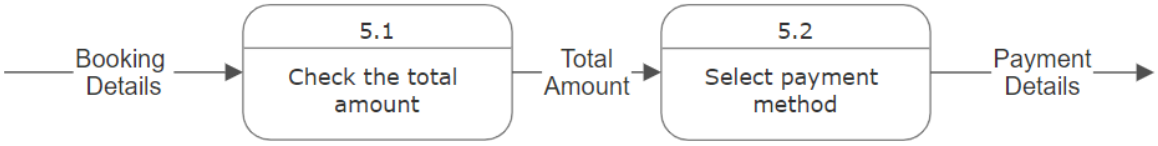
2.3.3 Process 3.0 <Check Bus Availability >



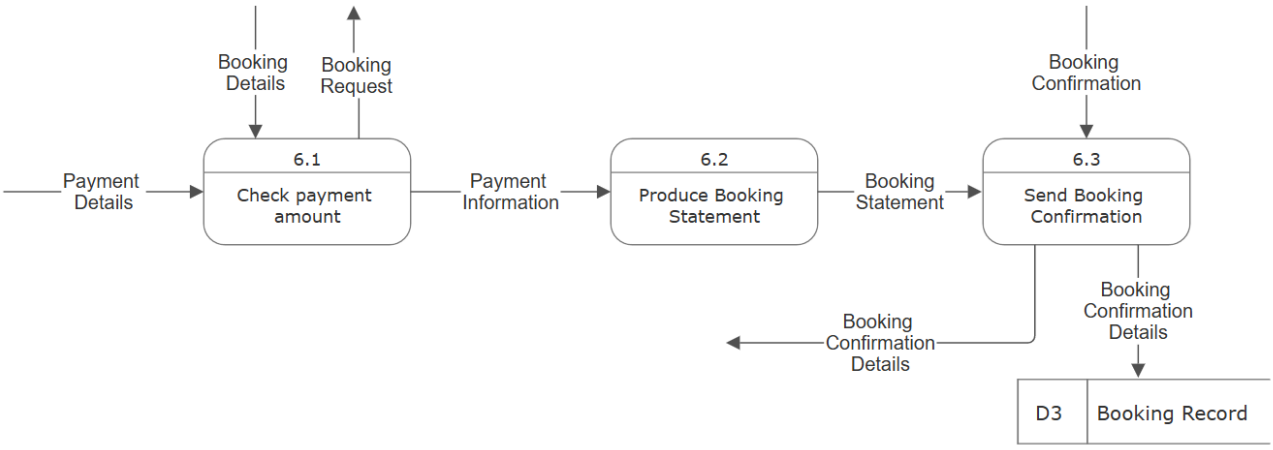
2.3.4 Process 4.0 <Booking>



2.3.5 Process 5.0 <Make payment >



2.3.6 Process 6.0 <Verify Booking >



3.0 Data & Transaction Requirement

3.1 Proposed business rule

Staff:

- View and manage the bus availability like add or delete bus for booking.
- Approve or reject bookings.
- Change the booking price.
- Can send any message regarding the booking to the customer.
- Able review and update payment records.
- Issue refunds when booking gets canceled.
- Provide customer support and handle any customer queries when needed.

Customer:

- Can login to the account.
- Enter date, pickup and drop off location, no. of passengers to book available buses online.
- Cancel booking up to 5 days prior to the booking.
- Able to view the seat capacity of the bus before booking.
- Securely pay using e-wallets, online banking or cards.
- View booking history and payment status.

3.2 Proposed data & transactional

3.2.1 Proposed Data Requirement

Staff

The data stored in the Staff entity includes StaffID, StaffName, Position, ContactNo and StaffPassword. The StaffID is defined as the primary key. The StaffName is a composite attribute. The StaffName consists of Sfirstname and Slastname. A staff can manage many bookings of a customer.

Customer

The data stored in the Customer entity includes CustomerID, CustName, Password, Email, SecurityQues, SecurityAns, Age, Gender, Address and CustPhone. The CustomerID is defined as the primary key. The CustName and Address are the composite attributes. The CustName consists of Cfirstname and Clastname, whereas the Address consists of Street, City, and Postcode. A customer can make 0 or many bookings.

Booking

The data stored in the Booking includes BookingID, CustomerID, BusID, BookingDate, PickUpLocation, DropOffLocation, NumOfPassengers, TotalPrice, Status. The BookingID is the primary key. The CustomerID and BusID are Foreign keys which are related to the Customer entity and Bus entity. A Booking is made by a Customer. A booking made by a Customer is managed by a staff. A booking has a payment. A booking triggers 0 or many notifications and a bus may have none or many bookings.

Notification

The Notification entity has NotiID, BookingID, StaffID, Message. The NotiID is the Primary Key. The BookingID and StaffID are the Foreign keys with reference from Booking entity and Staff entity. Notification can be triggered by a booking.

Bus

The Bus entity contains BusID, Model, SeatCapacity, BookingPrice, Availability. The BusID is the primary key. A bus has zero or more bookings.

Payment

The data stored in Payment are PaymentID, BookingID, AnalysisID, PaymentMethod, PaymentAmount, PaymentDate, PaymentStatus. The PaymentID is the Primary key. The BookingID and AnalysisID are foreign keys which refer to the Booking entity and IncomeAnalysis entity. Booking has 0 or 1 payment. 1 Income Analysis has 1 or more payment.

IncomeAnalysis

The data stored in IncomeAnalysis are AnalysisID, TotalIncome, TotalExpenses, Profit. AnalysisID is the Primary key. The Profit is a derived attribute calculated from the TotalIncome-TotalExpenses. An income analysis has analysis of 1 or many payments.

3.2.2 Proposed Transactional Requirement

Data Entry

1. **Enter Customer Information**

   Add new customer records with details like name, ID, age, gender, contact information, email, password, and security questions.

2. **Enter Bus Information**

   Add bus records including bus model, seating capacity, bus booking price, and availability status.

3. **Enter Booking Information**

   Record booking details such as booking date, pickup and drop-off locations, number of passengers, and total booking price.

Data Update/Delete

1. **Update/Delete Customer Information**

   Modify or remove existing customer records.

2. **Update/Delete Bus Information**

   Modify or remove bus details, including model, capacity, price, and availability.

3. **Update/Delete Booking Information**

   Modify or cancel bookings, including changes to pickup/drop-off locations and booking dates.

4. **Update/Delete Payment Status**

   Update payment statuses (e.g., pending, completed) and modify payment methods if necessary.

Data Queries

1. **Display Available Bus**

   Show a list of all buses available for booking, including their seating capacity and booking price.

2. **Display Booking Information**

   Retrieve and display details of bookings, including customer information, booking date, and status.

3. **Display Customer History**

   Show a customer's booking and payment history.

4. **Display Payment Details**

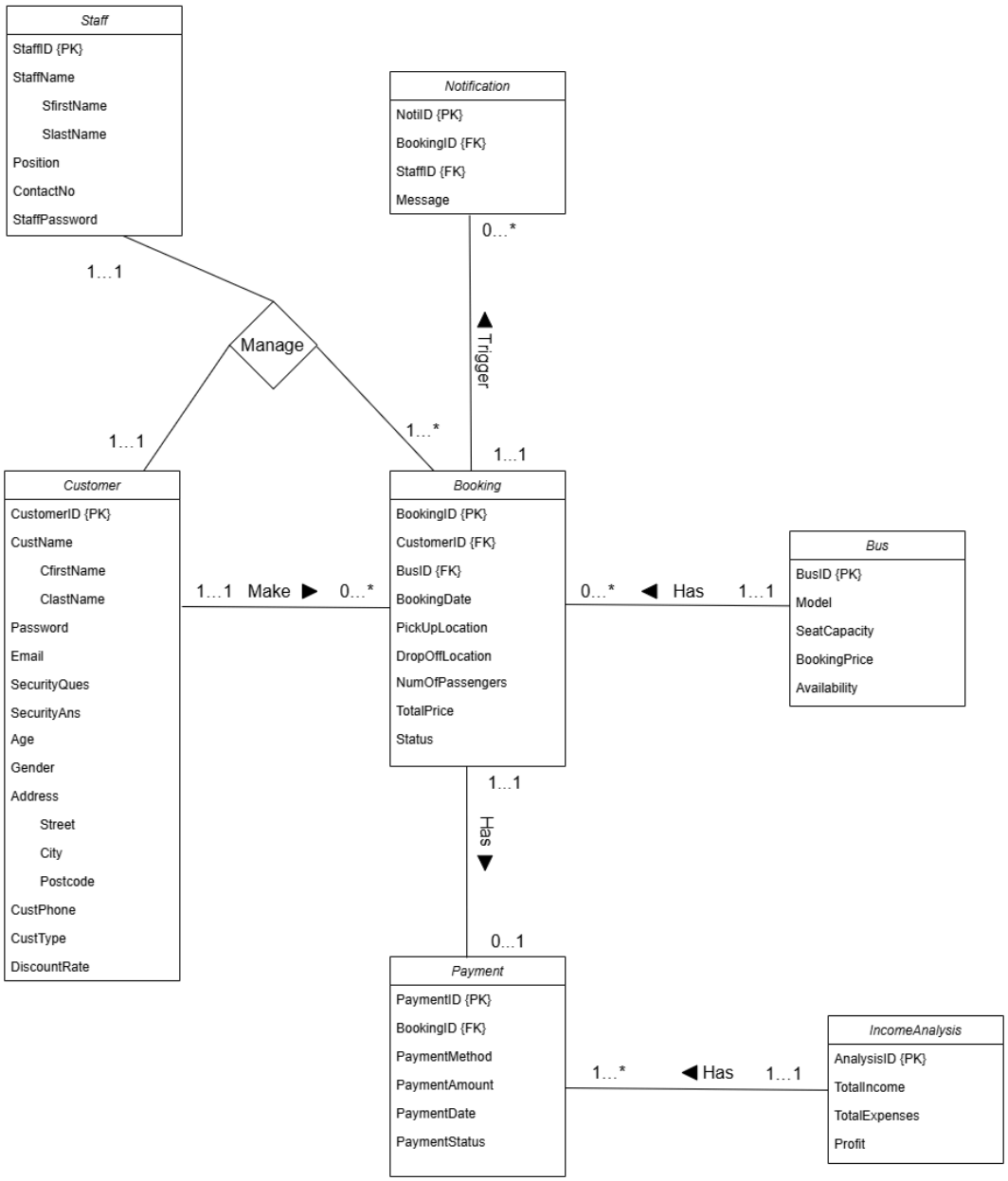
   Retrieve payment details, including total amount paid, payment method, and status.

5. **Generate Income Analysis Report**

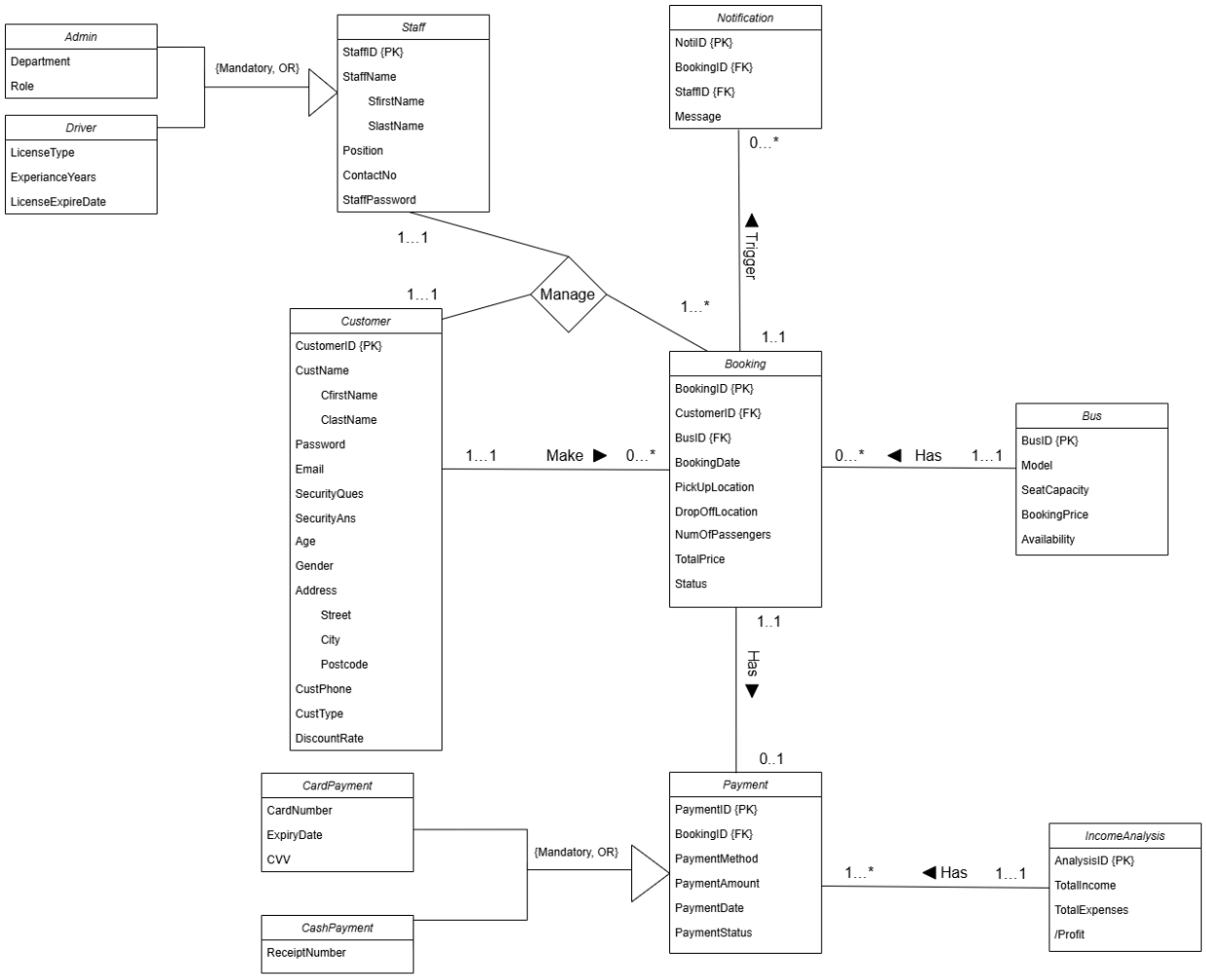
   Display income, expenses, and calculated profit based on completed bookings.

4.0 Database Conceptual Design

4.1 Conceptual ERD



4.2 Enhanced ERD (EERD)



5.0 Data Dictionary

5.1 Description of Entity

Entity	Description	Occurrence
Staff	Staff's information	Staff manage booking by updating availability of vehicles. Staff approve or reject the booking made by customers.
Customer	Customer's information	Customers make bookings and payment.
Notification	Notification's information	Notifications are generated for each booking, including details like the booking is approved or rejected.
Booking	Booking's information	Bookings made by customers for bus rentals, including pickup and drop-off locations, rental date, passenger count and generate payment.
Bus	Bus's information	Buses for rentals with details like model, seat capacity, rental price and availability updated by staff.
Payment	Payment's information	Payment generated after the customers made booking and payment being made by the customers.
IncomeAnalysis	IncomeAnalysis's information	Analysis of income and expenses associated with bookings and payments.

5.2 Description of Relationship

Entity	Multiplicity	Relationship	Multiplicity	Entity
Customer	1…1	Make	0…*	Booking
Booking	1…1	Trigger	0…*	Notification
	1…1	Has	0…1	Payment
Bus	1…1	Has	0…*	Booking
IncomeAnalysis	1…1	Has	1…*	Payment

5.3 Description Attributes

Entity	Attribute	Description	Data Type	Null	Multi valued
Staff	StaffID	Uniquely identifies a Staff (PK)	VARCHAR(8)	No	No
	StaffName	Name of the Staff	VARCHAR(60)	No	No
	SfirstName	First name of Staff	VARCHAR(30)	No	No
	SlastName	Last name of Staff	VARCHAR(30)	No	No
	Position	Position of the Staff	VARCHAR(20)	No	No
	ContactNo	Contact number of the Staff	VARCHAR(20)	No	Yes
	StaffPassword	Password for Staff account	VARCHAR(15)	No	No
Customer	CustomerID	Uniquely identifies a Customer (PK)	VARCHAR(8)	No	No
	CustName	Name of the Customer	VARCHAR(60)	No	No
	CfirstName	First name of	VARCHAR(30)	No	No

		Customer			
	ClastName	Last name of Customer	VARCHAR(30)	No	No
	Password	Secure password for Customer account	VARCHAR(15)	No	No
	Email	Email of the Customer	VARCHAR(30)	No	No
	SecurityQues	Security question for Customer account	VARCHAR(50)	No	No
	SecurityAns	Answer for the security question	VARCHAR(10)	No	No
	Age	Age of the Customer	INT(3)	No	No
	Gender	Gender of the Customer	CHAR(1)	No	No
	Address	Address of the Customer	VARCHAR(100)	No	No
	Street	Street Address of Customer	VARCHAR(50)	No	No
	City	City Address of Customer	VARCHAR(40)	No	No
	Postcode	Postcode Address of Customer	VARCHAR(10)	No	No
	CustPhone	Phone Number of the Customer	VARCHAR(20)	No	Yes
	CustType	Customer Type	VARCHAR(20)	No	No
	DiscountRate	DiscountRate given to customer	FLOAT(2,2)	No	No
Notification	NotiID	Uniquely identifies a notification (PK)	VARCHAR(8)	No	No
	BookingID	Foreign key of Booking which	VARCHAR(8)	No	No

		uniquely identifies a booking (FK)			
	Message	Gives a message to notify the customer	VARCHAR(100)	No	No
Booking	BookingID	Uniquely identifies a booking (PK)	VARCHAR(8)	No	No
	CustomerID	Foreign key of Customer which uniquely identifies a customer (FK)	VARCHAR(8)	No	No
	BusID	Foreign key of Bus which uniquely identifies a bus (FK)	VARCHAR(8)	No	No
	BookingDate	Date the bus is booked	DATE	No	No
	PickUpLocation	Pickup location for the bus	VARCHAR(50)	No	No
	DropOffLocation	Drop off location for the bus	VARCHAR(50)	No	No
	NumOfPassesngers	Number of passengers for the bus	INT(3)	No	No
	TotalPrice	Total price for the booking	DECIMAL(8,2)	No	No
	Status	Status of the booking for example: approved, rejected	VARCHAR(10)	No	No
Bus	BusID	Uniquely identifies a Bus (PK)	VARCHAR(8)	No	No
	Model	Model of the Bus	VARCHAR(10)	No	No
	SeatCapacity	The seating capacity of a bus	INT(2)	No	No
	BookingPrice	The price of a bus	DECIMAL(8,2)	No	No

		for booking			
	Availability	The status of availability of a bus to rent	VARCHAR(3)	No	No
Payment	PaymentID	Uniquely identifies a payment (PK)	VARCHAR(8)	No	No
	BookingID	Foreign key of Booking which uniquely identifies a booking (FK)	VARCHAR(8)	No	No
	PaymentMethod	Method of payment, for example: cash, e-wallet	VARCHAR(10)	No	No
	PaymentAmount	Total amount paid	DECIMAL(8,2)	No	No
	PaymentDate	Date of payment	DATE	No	No
	PaymentStatus	Status of the payment, for example: successful, pending, failed	VARCHAR(10)	No	No
IncomeAnalysis	AnalysisID	Uniquely identifies an income analysis (PK)	VARCHAR(8)	No	No
	TotalIncome	Total income generated	DECIMAL(8,2)	No	No
	TotalExpenses	Total expenses incurred	DECIMAL(8,2)	No	No
	Profit	Calculated profit	DECIMAL(8,2)	No	No

6.0 Summary

In phase 2, our group members gained a deeper understanding of the foundation for bus booking systems and how they operate by completing the conceptual database design. We refined our business rules, outlined the data and transaction requirements, and developed data flow diagrams, including the context, parent (level 0), and child (level 1) diagrams. This approach allowed us to map out critical processes such as booking a bus, checking availability, and managing booking schedules.

The data flow diagrams have given us a clear view of the complex interactions between entities and processes in the system architecture. We also created both a conceptual and enhanced entity-relationship diagram (ERD) along with a data dictionary, which were essential for defining the database structure and ensuring consistent terminology.

Our data and transaction requirements emphasize the need for a robust system capable of handling issues such as customer cancellations, and scheduling conflicts. By completing the ERDs, we refined the relationships between entities, creating a comprehensive blueprint for the database. The data dictionary served as a reference guide, ensuring clarity in system terminology.

In summary, this phase represents significant progress in transitioning our project from a conceptual model to a structured database design, laying a solid foundation for subsequent development phases and supporting the success of our bus booking system.

UTM
UNIVERSITI TEKNOLOGI MALAYSIA

SECD2523
DATABASE
SECTION 06
LECTURER: DR. LAYLA RASHEED ABDALLAH HASAN

BUSFLEX
BUS BOOKING SYSTEM

Phase 3 - Database Logical Design

Name	Matric No
Chuah Hui Wen	A23CS0219
Tan Xin Ying	A23CS0276
Anjum Siddiqua Tanveer Siddiqui	A23CS0289
Mathaba Hassan Mohamed Hassan	A23CS4044
Alwaeli Amr Khaled Abdo	A23CS4004

Table of Contents

1.0 Introduction

In today's busy world, having a good booking system is really important for businesses that want to keep up with customer needs. A lot of traditional ways of making reservations aren't very effective anymore. They often lead to double bookings, missed appointments, and trouble with keeping track of available times, which can make things difficult for both the business and the customers.

A reliable booking system makes it easier for customers to make reservations quickly and easily. While it also helps businesses stay organized. This kind of system keeps everything running smoothly by letting customers book with confidence and by helping staff manage appointments better. With a good booking system in place, daily operations get easier, scheduling issues are minimized, and customers end up happier. Overall, a well-designed booking system not only improves the reservation process but also helps businesses build strong, long-lasting relationships with their customers.This will result in improved profits the business makes.

In this phase, we will thoroughly analyze the booking system's processes and their relationships through Entity-Relationship Diagrams (ERDs). We will define the essential business rules for the system and analyze data requirements, specifying detailed data types for each entity. This phase will also cover the Database Logical Design, where entities and relationships will be translated into Relational Database Schemas following normalization principles to eliminate redundancy and ensure data integrity. Additionally, we will develop SQL Statements for both Data Definition Language (DDL) to create and modify database structures and Data Manipulation Language (DML) to manage and query data. Finally, we will design the system Interface to ensure seamless interaction between users and the database, establishing a strong foundation for data modeling and comprehensive understanding of data-related concepts.

2.0 Overview of project

In the phase of database logical design, we will transform the conceptual Entity Relationship Diagrams (ERD) created in Phase 2 into logical ERDs. In this step, modifications will be made to address the complexities of a bus booking system, setting the stage for a more structured and efficient database. Each entity in the logical ERD will be converted into a relational schema, with attributes defined and primary keys created for each table. This is particularly critical for a bus booking system, where entities such as passenger, booking, bus, driver, and payment must be clearly defined and interconnected.

Normalization will also take place during this phase to eliminate redundancies and dependencies within the relational schema. For a bus booking system, this ensures that passenger information is not duplicated, bookings are properly associated with buses and drivers, and data integrity is maintained. Simultaneously, the data dictionary will be updated as needed to reflect changes brought about by the normalization process.

Finally, we will propose SQL statements to validate the logical ERDs against the system's transaction requirements using interface design. This step is essential to confirm that the database structure supports key operations such as ticket booking, rescheduling, and payment processing. In the context of a bus booking system, ensuring responsiveness and reliability is crucial, as it directly impacts the user experience and the system's overall efficiency.

3.0 Database conceptual design

3.1 Updated business rule

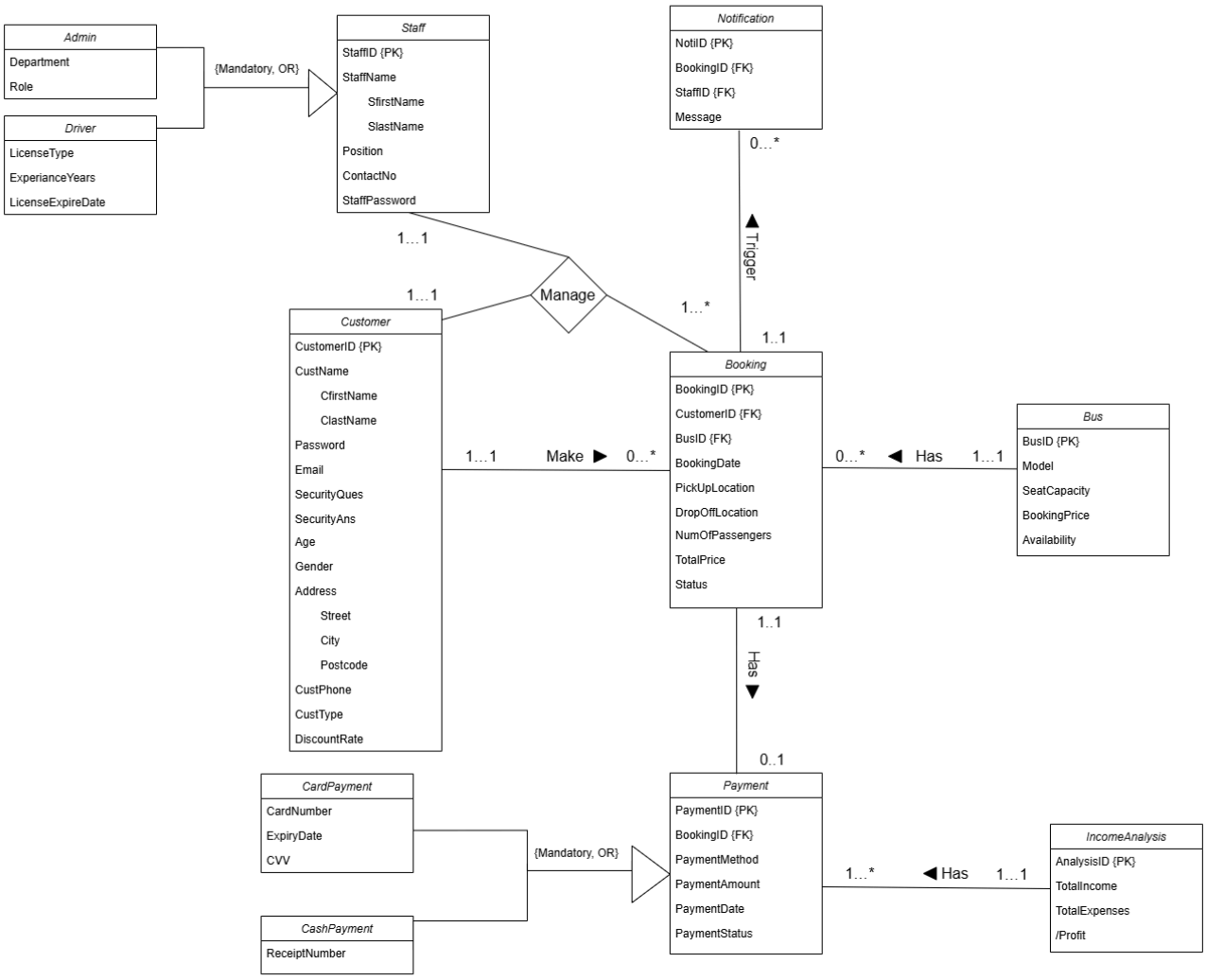
Staff:

- View and manage the bus availability like add or delete bus for booking.
- Approve or reject bookings.
- Change the booking price.
- Can send any message regarding the booking to the customer.
- Able review and update payment records.
- Issue refunds when booking gets canceled.
- Provide customer support and handle any customer queries when needed.

Customer:

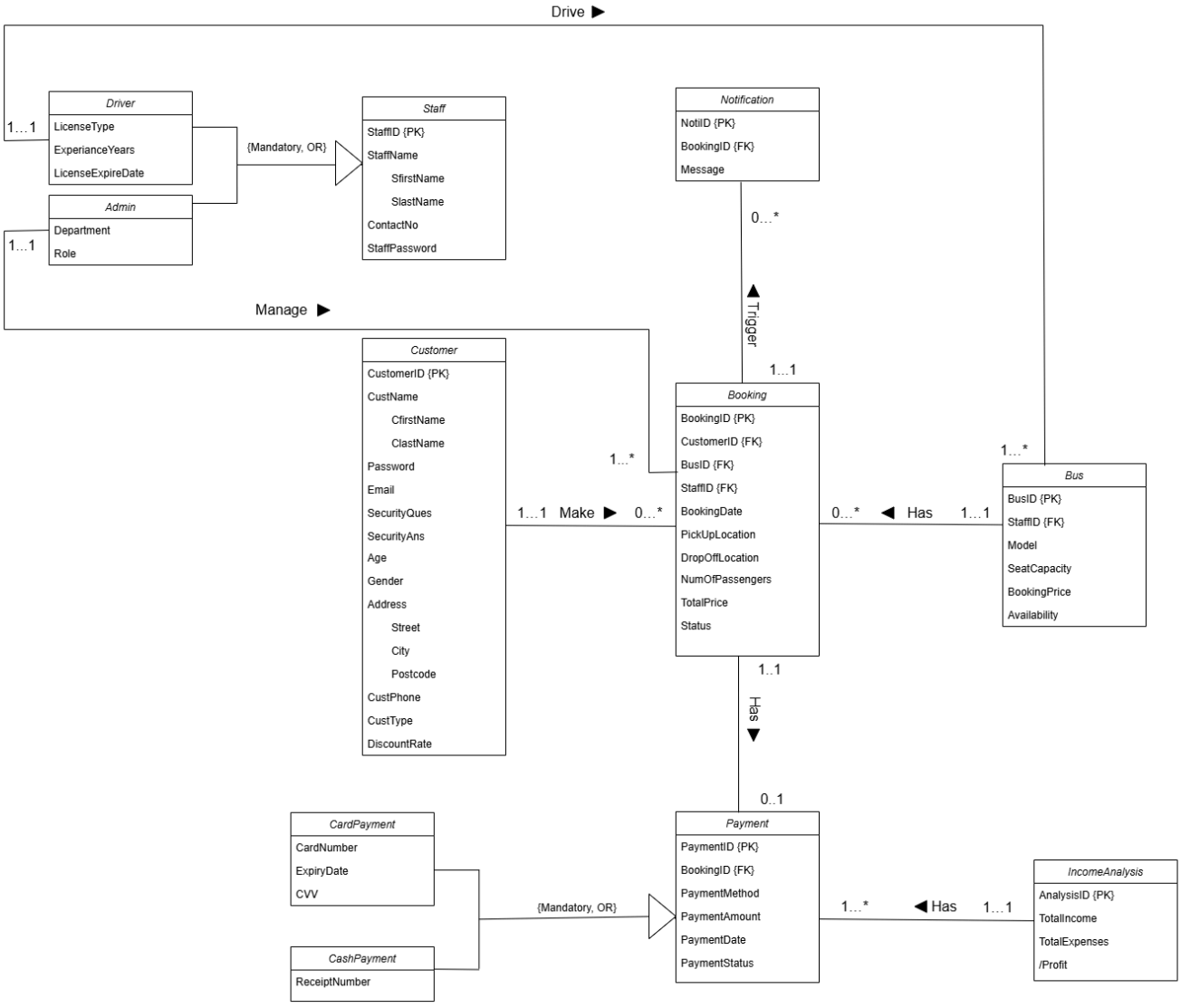
- Can login to the account.
- Enter date, pickup and drop off location, no. of passengers to book available buses online.
- Cancel booking up to 5 days prior to the booking.
- Able to view the seat capacity of the bus before booking.
- Securely pay using e-wallets, online banking or cards.
- View booking history and payment status.

3.2 Conceptual ERD



4.0 DB logical design

4.1 Logical ERD



4.2 Updated Data Dictionary

Description of Entity

Entity	Description	Occurrence
Staff	Superclass for Admin and Driver	Staff can be associated with either Admin or Driver, but not more than one at a time.
Admin	Hold admin's information	Admin manage bookings by approving or rejecting bookings of the customer.
Driver	Hold driver's information	Driver drive bus.
Customer	Customer's information	Customers make bookings and payment.
Notification	Notification's information	Notifications are generated for each booking, including details like the booking is approved or rejected.
Booking	Booking's information	Bookings made by customers for bus rentals, including pickup and drop-off locations, rental date, passenger count and generate payment.
Bus	Bus's information	Buses for rentals with details like model, seat capacity, rental price and availability updated by staff.
Payment	Payment's information	Payment generated after the customers made booking and payment being made by the customers.
CardPayment	Hold payment information when paying by credit card	One of the selections available to customers when making payment.
CashPayment	Hold payment information when paying by cash	One of the selections available to customers when making payment.

IncomeAnalysis	IncomeAnalysis's information	Analysis of income and expenses associated with bookings and payments.

Description of Relationship

Entity	Multiplicity	Relationship	Multiplicity	Entity
Driver	1…1	Drive	1…*	Bus
Admin	1…1	Manage	1…*	Booking
Customer	1…1	Make	0…*	Booking
Booking	1…1	Trigger	0…*	Notification
Booking	1…1	Has	0…1	Payment
Bus	1…1	Has	0…*	Booking
IncomeAnalysis	1…1	Has	1…*	Payment

Description Attributes

Entity	Attribute	Description	Data Type	Null	Multi valued
Staff	StaffID	Uniquely identifies a Staff (PK)	VARCHAR(8)	No	No
	StaffName	Name of the Staff	VARCHAR(60)	No	No
	SfirstName	First name of Staff	VARCHAR(30)	No	No
	SlastName	Last name of Staff	VARCHAR(30)	No	No
	ContactNo	Contact number of the Staff	VARCHAR(20)	No	Yes
	StaffPassword	Password for Staff account	VARCHAR(15)	No	No
Driver	LicenseType	Type of license that is owned by the driver	VARCHAR(20)	No	Yes
	ExperienceYears	The year of experience of the driver	INT(2)	No	No
	LicenseExpireDate	The expire date of the license that is owned by the driver	DATE	No	No
Admin	Department	Department that the admin belongs to	VARCHAR(20)	No	No
	Role	Role of the admin	VARCHAR(20)	No	No
Customer	CustomerID	Uniquely identifies a Customer (PK)	VARCHAR(8)	No	No

	CustName	Name of the Customer	VARCHAR(60)	No	No
	CfirstName	First name of Customer	VARCHAR(30)	No	No
	ClastName	Last name of Customer	VARCHAR(30)	No	No
	Password	Secure password for Customer account	VARCHAR(15)	No	No
	Email	Email of the Customer	VARCHAR(30)	No	No
	SecurityQues	Security question for Customer account	VARCHAR(50)	No	No
	SecurityAns	Answer for the security question	VARCHAR(10)	No	No
	Age	Age of the Customer	INT(3)	No	No
	Gender	Gender of the Customer	CHAR(1)	No	No
	Address	Address of the Customer	VARCHAR(100)	No	No
	Street	Street Address of Customer	VARCHAR(50)	No	No
	City	City Address of Customer	VARCHAR(40)	No	No
	Postcode	Postcode Address of Customer	VARCHAR(10)	No	No
	CustPhone	Phone Number of the Customer	VARCHAR(20)	No	Yes
	CustType	Customer Type	VARCHAR(20)	No	No
	DiscountRate	DiscountRate	FLOAT(2,2)	No	No

		given to customer			
Notification	NotiID	Uniquely identifies a notification (PK)	VARCHAR(8)	No	No
	BookingID	Foreign key of Booking which uniquely identifies a booking (FK)	VARCHAR(8)	No	No
	Message	Gives a message to notify the customer	VARCHAR(100)	No	No
Booking	BookingID	Uniquely identifies a booking (PK)	VARCHAR(8)	No	No
	CustomerID	Foreign key of Customer which uniquely identifies a customer (FK)	VARCHAR(8)	No	No
	BusID	Foreign key of Bus which uniquely identifies a bus (FK)	VARCHAR(8)	No	No
	StaffID	Foreign key of Staff which uniquely identifies a staff (FK)	VARCHAR(8)	No	No
	BookingDate	Date the bus is booked	DATE	No	No
	PickUpLocation	Pickup location for the bus	VARCHAR(50)	No	No
	DropOffLocation	Drop off location for the bus	VARCHAR(50)	No	No

	NumOfPassesngers	Number of passengers for the bus	INT(3)	No	No
	TotalPrice	Total price for the booking	DECIMAL(8,2)	No	No
	Status	Status of the booking for example: approved, rejected	VARCHAR(10)	No	No
Bus	BusID	Uniquely identifies a Bus (PK)	VARCHAR(8)	No	No
	StaffID	Foreign key of Staff which uniquely identifies a Staff (FK)	VARCHAR(8)	No	No
	Model	Model of the Bus	VARCHAR(50)	No	No
	SeatCapacity	The seating capacity of a bus	INT(2)	No	No
	BookingPrice	The price of a bus for booking	DECIMAL(8,2)	No	No
	Availability	The status of availability of a bus to rent	VARCHAR(15)	No	No
Payment	PaymentID	Uniquely identifies a payment (PK)	VARCHAR(8)	No	No
	BookingID	Foreign key of Booking which uniquely identifies a booking (FK)	VARCHAR(8)	No	No
	PaymentMethod	Method of payment, for example: cash,	VARCHAR(50)	No	No

		e-wallet			
	PaymentAmount	Total amount paid	DECIMAL(8,2)	No	No
	PaymentDate	Date of payment	DATE	No	No
	PaymentStatus	Status of the payment, for example: successful, pending, failed	VARCHAR(10)	No	No
CardPayment	CardNumber	The number of the credit card	BIGINT	No	No
	ExpiryDate	Expiry date of the credit card	DATE	No	No
	CVV	The Card Verification Value of the credit card	INT(3)	No	No
CashPayment	ReceiptNumber	The receipt number of the payment made	INT(15)	No	No
IncomeAnalysis	AnalysisID	Uniquely identifies an income analysis (PK)	VARCHAR(8)	No	No
	TotalIncome	Total income generated	DECIMAL(8,2)	No	No
	TotalExpenses	Total expenses incurred	DECIMAL(8,2)	No	No
	Profit	Calculated profit	DECIMAL(8,2)	No	No

Relations for the Bus Booking System

STAFF(StaffID, StaffName, ContactNo, StaffPassword) **Primary Key** StaffID	CUSTOMER (CustomerID, CustName, Password, Email, SecurityQues, SecurityAns, Age, Gender, Address, CustPhone, CustType, DiscountRate) **Primary Key** CustomerID
DRIVER (StaffID, LicenseType, ExperienceYears, LicenseExpireDate) **Foreign Key** StaffID **references** STAFF (StaffID)	ADMIN (StaffID, Department, Role) **Foreign Key** StaffID **references** STAFF (StaffID)
BOOKING(BookingID, CustomerID, BusID, StaffID, BookingDate, PickUpLocation, DropOffLocation, NumOfPassesngers, TotalPrice, Status) **Primary Key** BookingID **Foreign Key** CustomerID **reference** Customer (CustomerID) **Foreign Key** BusID **reference** Bus(BusID) **Foreign Key** StaffID **reference** Staff (StaffID)	BUS (BusID, StaffID, Model, SeatCapacity, BookingPrice, Availability) **Primary Key** BusID **Foreign Key** StaffID **reference** Staff (StaffID)
PAYMENT (PaymentID, BookingID, PaymentMethod, PaymentAmount, PaymentDate, PaymentStatus) **Primary Key** PaymentID **Foreign Key** BookingID **reference** Booking(BookingID)	INCOMEANALYSIS (AnalysisID, TotalIncome, TotalExpenses, Profit) **Primary Key** AnalysisID **Derived** Profit (TotalIncome - TotalExpenses)
CARDPAYMENT (PaymentID, CardNumber, ExpiryDate, CVV) **Foreign Key** PaymentID **reference** Payment(PaymentID)	CASHPAYMENT (PaymentID, ReceiptNumber) **Foreign Key** PaymentID **reference** Payment(PaymentID)
NOTIFICATION (NotiID, BookingID, Message) **Primary Key** NotiID **Foreign Key** BookingID **reference** Booking(BookingID)	

4.3 Normalization

1. STAFF(StaffID, StaffName, ContactNo, StaffPassword)
   FD1: StaffID→ StaffName, ContactNo, StaffPassword
   1NF&2NF&3NF&BNCF:
   STAFF(<u>StaffID,</u> StaffName, ContactNo, StaffPassword)

2. DRIVER(StaffID, LicenseType, ExperienceYears, LicenseExpireDate)
   FD1: StaffID→ LicenseType, ExperienceYears, LicenseExpireDate
   1NF&2NF&3NF&BNCF:
   DRIVER(<u>StaffID,</u> LicenseType, ExperienceYears, LicenseExpireDate)

3. ADMIN(StaffID, Department, Role)
   FD1: StaffID→ Department, Role
   1NF&2NF&3NF&BNCF:
   ADMIN(<u>StaffID</u>, Department, Role)

4. CUSTOMER(CustomerID, CustName, Password, Email, SecurityQues, SecurityAns, Age, Gender, Address, CustPhone, CustType, DiscountRate)
   FD1: CustomerID → CustName, Password, Email, SecurityQues, SecurityAns, Age, Gender, Address, CustPhone, CustType, DiscountRate
   1NF&2NF&3NF&BNCF:
   CUSTOMER(CustName, Password, Email, SecurityQues, SecurityAns, Age, Gender, Address, CustPhone, CustType, DiscountRate)

5. NOTIFICATION(NotiID, BookingID, Message)
   FD1: NotiID → BookingID, Message
   1NF&2NF&3NF&BNCF:
   NOTIFICATION(<u>NotiID</u>, BookingID, Message)

6. BOOKING(BookingID, CustomerID, BusID, StaffID, BookingDate, PickUpLocation, DropOffLocation, NumOfPassengers, TotalPrice, Status)

FD1: BookingID → CustomerID, BusID, StaffID, BookingDate, PickUpLocation, DropOffLocation, NumOfPassengers, TotalPrice, Status

1NF&2NF&3NF&BNCF:

BOOKING(<u>BookingID</u>, CustomerID, BusID, StaffID, BookingDate, PickUpLocation, DropOffLocation, NumOfPassengers, TotalPrice, Status)

7. BUS(BusID, StaffID, Model, SeatCapacity, BookingPrice, Availability)
   FD1: BusID → StaffID, Model, SeatCapacity, BookingPrice, Availability
   1NF&2NF&3NF&BNCF:
   BUS(<u>BusID</u>, StaffID, Model, SeatCapacity, BookingPrice, Availability)

8. PAYMENT(PaymentID, BookingID, PaymentMethod, PaymentAmount, PaymentDate, PaymentStatus)
   FD1: PaymentID → BookingID, PaymentMethod, PaymentAmount, PaymentDate, PaymentStatus
   1NF&2NF&3NF&BNCF:
   CARDPAYMENT(<u>PaymentID</u>, BookingID, PaymentMethod, PaymentAmount, PaymentDate, PaymentStatus)

9. CARDPAYMENT(PaymentID, CardNumber, ExpiryDate, CVV)
   FD1: PaymentID → CardNumber, ExpiryDate, CVV
   1NF&2NF&3NF&BNCF:
   CARDPAYMENT(<u>PaymentID</u>, CardNumber, ExpiryDate, CVV)

10. CASHPAYMENT(PaymentID, ReceiptNumber)
    FD1: PaymentID → ReceiptNumber
    1NF&2NF&3NF&BNCF:
    CASHPAYMENT(<u>PaymentID</u>, ReceiptNumber)

11. INCOMEANALYSIS(AnalysisID, TotalIncome, TotalExpenses, Profit)
    FD1: AnalysisID → TotalIncome, TotalExpenses, Profit

1NF&2NF&3NF&BNCF:

INCOMEANALYSIS(AnalysisID, TotalIncome, TotalExpenses, Profit)

*Remark: Primary Key (PK) is underlined.

5.0 Relational DB Schemas (after normalization)

The relational database schema for Bus Booking System database is a set of relation schema which consist of:

Staff	(StaffID, StaffName, ContactNo, StaffPassword)
Driver	(StaffID, LicenseType, ExperienceYears, LicenseExpireDate)
Admin	(StaffID, Department, Role)
Customer	(CustomerID, CustName, Password, Email, SecurityQues, SecurityAns, Age, Gender, Address, CustPhone, CustType, DiscountRate)
Notification	(NotiID, BookingID, Message)
Booking	(BookingID, CustomerID, BusID, StaffID, BookingDate, PickUpLocation, DropOffLocation, NumOfPassengers, TotalPrice, Status)
Bus	(BusID, StaffID, Model, SeatCapacity, BookingPrice, Availability)
Payment	(PaymentID, BookingID, PaymentMethod, PaymentAmount, PaymentDate, PaymentStatus)
CardPayment	(PaymentID, CardNumber, ExpiryDate, CVV)
CashPayment	(PaymentID, ReceiptNumber)
IncomeAnalysis	(AnalysisID, TotalIncome, TotalExpenses, Profit)

*Remark: Primary Key (PK) is underlined.

Staff

StaffID	StaffName	ContactNo	StaffPassword

Driver

StaffID	LicenseType	ExperienceYears	LicenseExpireDate

Admin

StaffID	Department	Role

Customer

CustomerID	CustName	Password	Email	SecurityQues	SecurityAns	Age	Gender

Address	CustPhone	CustType	DiscountRate

Notification

NotiID	BookingID	Message

Booking

BookingID	CustomerID	BusID	StaffID	BookingDate	PickUpLocation

DropOffLocation	NumOfPassengers	TotalPrice	Status

Bus

BusID	StaffID	Model	SeatCapacity	BookingPrice	Availability

Payment

PaymentID	BookingID	PaymentMethod	PaymentAmount	PaymentDate	PaymentStatus

CardPayment

PaymentID	CardNumber	ExpiryDate	CVV

CashPayment

PaymentID	ReceiptNunber

IncomeAnalysis

AnalysisID	TotalIncome	TotalExpenses	Profit

6.0 SQL Statements (DDL & DML)

6.1 DDL:

```
-- staff table
CREATE TABLE STAFF (
  StaffID VARCHAR(8) PRIMARY KEY,
  StaffName VARCHAR(60) NOT NULL,
  ContactNo VARCHAR(20),
  StaffPassword VARCHAR(15) NOT NULL
);

-- driver table
CREATE TABLE DRIVER (
  StaffID VARCHAR(8) PRIMARY KEY,
  LicenseType VARCHAR(20) NOT NULL,
  ExperienceYears INT(2) NOT NULL,
  LicenseExpireDate DATE NOT NULL,
  FOREIGN KEY (StaffID) REFERENCES STAFF(StaffID)
);

-- Admin table
CREATE TABLE ADMIN (
  StaffID VARCHAR(8) PRIMARY KEY,
  Department VARCHAR(20) NOT NULL,
  Role VARCHAR(20) NOT NULL,
  FOREIGN KEY (StaffID) REFERENCES STAFF(StaffID)
);

-- Customer Table
CREATE TABLE CUSTOMER (
  CustomerID VARCHAR(8) PRIMARY KEY,
  CustName VARCHAR(60) NOT NULL,
  Password VARCHAR(15) NOT NULL,
  Email VARCHAR(30) NOT NULL,
  SecurityQues VARCHAR(50) NOT NULL,
  SecurityAns VARCHAR(10) NOT NULL,
```

```
  Age INT(3) NOT NULL,
  Gender CHAR(1) NOT NULL,
  Address VARCHAR(100) NOT NULL,
  CustPhone VARCHAR(20) NOT NULL,
  CustType VARCHAR(20) NOT NULL,
  DiscountRate FLOAT(2,2) NOT NULL
);

-- Booking table
CREATE TABLE BOOKING (
  BookingID VARCHAR(8) PRIMARY KEY,
  CustomerID VARCHAR(8) NOT NULL,
  BusID VARCHAR(8) NOT NULL,
  StaffID VARCHAR(8) NOT NULL,
  BookingDate DATE NOT NULL,
  PickUpLocation VARCHAR(50) NOT NULL,
  DropOffLocation VARCHAR(50) NOT NULL,
  NumOfPassengers INT(3) NOT NULL,
  TotalPrice DECIMAL(8,2) NOT NULL,
  Status VARCHAR(10) NOT NULL,
  FOREIGN KEY (CustomerID) REFERENCES CUSTOMER(CustomerID),
  FOREIGN KEY (BusID) REFERENCES BUS(BusID),
  FOREIGN KEY (StaffID) REFERENCES STAFF(StaffID)
);

-- Notification Table
CREATE TABLE NOTIFICATION (
  NotiID VARCHAR(8) PRIMARY KEY,
  BookingID VARCHAR(8) NOT NULL,
  Message VARCHAR(100) NOT NULL,
  FOREIGN KEY (BookingID) REFERENCES BOOKING(BookingID)
);

-- Bus Table
CREATE TABLE BUS (
  BusID VARCHAR(8) PRIMARY KEY,
```

```
  StaffID VARCHAR(8) NOT NULL,
  Model VARCHAR(50) NOT NULL,
  SeatCapacity INT(2) NOT NULL,
  BookingPrice DECIMAL(8,2) NOT NULL,
  Availability VARCHAR(15) NOT NULL,
  FOREIGN KEY (StaffID) REFERENCES STAFF(StaffID)
);

-- payment table
CREATE TABLE PAYMENT (
  PaymentID VARCHAR(8) PRIMARY KEY,
  BookingID VARCHAR(8) NOT NULL,
  PaymentMethod VARCHAR(50) NOT NULL,
  PaymentAmount DECIMAL(8,2) NOT NULL,
  PaymentDate DATE NOT NULL,
  PaymentStatus VARCHAR(10) NOT NULL,
  FOREIGN KEY (BookingID) REFERENCES BOOKING(BookingID)
);

-- Cardpayment Table
CREATE TABLE CARDPAYMENT (
  PaymentID VARCHAR(8) PRIMARY KEY,
  CardNumber BIGINT  NOT NULL,
  ExpiryDate DATE NOT NULL,
  CVV INT(3) NOT NULL,
  FOREIGN KEY (PaymentID) REFERENCES PAYMENT(PaymentID)
);

-- CashPayment Table
CREATE TABLE CASHPAYMENT (
  PaymentID VARCHAR(8) PRIMARY KEY,
  ReceiptNumber INT(15) NOT NULL,
  FOREIGN KEY (PaymentID) REFERENCES PAYMENT(PaymentID)
);

-- IncomeAnalysis table
```

```
CREATE TABLE INCOMEANALYSIS (
  AnalysisID VARCHAR(8) PRIMARY KEY,
  TotalIncome DECIMAL(8,2) NOT NULL,
  TotalExpenses DECIMAL(8,2) NOT NULL,
  Profit DECIMAL(8,2) GENERATED ALWAYS AS (TotalIncome - TotalExpenses) STORED
);
```

6.2 DML:

Inserting into tables:

```
—- INSERTING DATA

-- Insert into STAFF Table
INSERT INTO STAFF (StaffID, StaffName, ContactNo,
StaffPassword)
VALUES ('S001', 'Ali Ibrahim', '0123456789',
'password123');
INSERT INTO STAFF (StaffID, StaffName, ContactNo,
StaffPassword)
VALUES ('S002', 'Noraini Binti Ahmad', '0198765432',
'mypassword');
INSERT INTO STAFF (StaffID, StaffName, ContactNo,
StaffPassword)
VALUES ('S003', 'Mohd Zaki Bin Salleh', '0182345678',
'securepass');
INSERT INTO STAFF (StaffID, StaffName, ContactNo,
StaffPassword)
VALUES ('S004', 'Ahmad Zainal', '0198765432',
'staffpass1');
INSERT INTO STAFF (StaffID, StaffName, ContactNo,
StaffPassword)
```

```
VALUES ('S005', 'Nur Aisyah Binti Mohd', '0187654321',
'staffpass2');

-- Insert into DRIVER Table
INSERT INTO DRIVER (StaffID, LicenseType,
ExperienceYears, LicenseExpireDate)
VALUES ('S001', 'Class D', 5, '2027-12-31');
INSERT INTO DRIVER (StaffID, LicenseType,
ExperienceYears, LicenseExpireDate)
VALUES ('S002', 'Class E', 3, '2025-06-30');
INSERT INTO DRIVER (StaffID, LicenseType,
ExperienceYears, LicenseExpireDate)
VALUES ('S003', 'Class D', 6, '2026-11-01');
INSERT INTO DRIVER (StaffID, LicenseType,
ExperienceYears, LicenseExpireDate)
VALUES ('S004', 'Class D', 4, '2026-06-30');
INSERT INTO DRIVER (StaffID, LicenseType,
ExperienceYears, LicenseExpireDate)
VALUES ('S005', 'Class B', 5, '2027-09-15');

-- Insert into ADMIN Table
INSERT INTO ADMIN (StaffID, Department, Role)
VALUES ('S001', 'Operations', 'Manager');
INSERT INTO ADMIN (StaffID, Department, Role)
VALUES ('S002', 'HR', 'Assistant');
INSERT INTO ADMIN (StaffID, Department, Role)
VALUES ('S003', 'Sales', 'Supervisor');

-- Insert into CUSTOMER Table
INSERT INTO CUSTOMER (CustomerID, CustName, Password,
Email, SecurityQues, SecurityAns, Age, Gender, Address,
CustPhone, CustType, DiscountRate)
```

```
VALUES ('C001', 'Aisyah Binti Abdullah', 'pass1234',
'aisyah@example.com', 'Mother\'s maiden name', 'Salmah',
22, 'F', 'Jalan Bukit Indah, Johor Bahru 81300',
'0129876543', 'Regular', 0.10);
INSERT INTO CUSTOMER (CustomerID, CustName, Password,
Email, SecurityQues, SecurityAns, Age, Gender, Address,
CustPhone, CustType, DiscountRate)
VALUES ('C002', 'John Tan', 'mypassword',
'john@example.com', 'First pet\'s name', 'Buddy', 30,
'M', 'Taman Johor Jaya, Johor Bahru 81300', '0135678923',
'VIP', 0.15);
INSERT INTO CUSTOMER (CustomerID, CustName, Password,
Email, SecurityQues, SecurityAns, Age, Gender, Address,
CustPhone, CustType, DiscountRate)
VALUES ('C003', 'Rina Mohamed', 'securepass',
'rina@example.com', 'Favourite teacher', 'Mr. Lim', 25,
'F', 'Jalan Setia, Johor Bahru 81300', '0162345789',
'Regular', 0.05);
INSERT INTO CUSTOMER (CustomerID, CustName, Password,
Email, SecurityQues, SecurityAns, Age, Gender, Address,
CustPhone, CustType, DiscountRate)
VALUES ('C004', 'Farah Azizah', 'farahpass',
'farah@example.com', 'First teacher\'s name', 'Ms.
Zainab', 28, 'F', 'Taman Melawati, Kuala Lumpur 53300',
'0171234567', 'VIP', 0.20);
INSERT INTO CUSTOMER (CustomerID, CustName, Password,
Email, SecurityQues, SecurityAns, Age, Gender, Address,
CustPhone, CustType, DiscountRate)
VALUES ('C005', 'Hassan Ali', 'hassan2025',
'hassan@example.com', 'First car model', 'Toyota', 35,
'M', 'Jalan Anggerik, Penang 10000', '0147654321',
'Regular', 0.10);
```

```sql
-- Insert into BUS Table
INSERT INTO BUS (BusID, StaffID, Model, SeatCapacity,
BookingPrice, Availability)
VALUES ('B001', 'S001', 'RapidKL', 15, 100.00,
'Available');
INSERT INTO BUS (BusID, StaffID, Model, SeatCapacity,
BookingPrice, Availability)
VALUES ('B002', 'S002', 'GoKL', 20, 120.00,
'Unavailable');
INSERT INTO BUS (BusID, StaffID, Model, SeatCapacity,
BookingPrice, Availability)
VALUES ('B003', 'S003', 'Penang Bus Service', 12, 80.00,
'Available');
INSERT INTO BUS (BusID, StaffID, Model, SeatCapacity,
BookingPrice, Availability)
VALUES ('B004', 'S004', 'KTM Bus', 25, 150.00,
'Available');
INSERT INTO BUS (BusID, StaffID, Model, SeatCapacity,
BookingPrice, Availability)
VALUES ('B005', 'S005', 'Express Bus', 30, 180.00,
'Available');

-- Insert into BOOKING Table
INSERT INTO BOOKING (BookingID, CustomerID, BusID,
StaffID, BookingDate, PickUpLocation, DropOffLocation,
NumOfPassengers, TotalPrice, Status)
VALUES ('BK001', 'C001', 'B001', 'S001', '2025-01-10',
'Johor Bahru 81300', 'Kuala Lumpur', 2, 150.00,
'Confirmed');
```

```
INSERT INTO BOOKING (BookingID, CustomerID, BusID,
StaffID, BookingDate, PickUpLocation, DropOffLocation,
NumOfPassengers, TotalPrice, Status)
VALUES ('BK002', 'C002', 'B002', 'S002', '2025-01-11',
'Johor Bahru 81300', 'Penang', 1, 100.00, 'Pending');
INSERT INTO BOOKING (BookingID, CustomerID, BusID,
StaffID, BookingDate, PickUpLocation, DropOffLocation,
NumOfPassengers, TotalPrice, Status)
VALUES ('BK003', 'C001', 'B003', 'S003', '2025-01-12',
'Johor Bahru 81300', 'Kuala Lumpur', 3, 180.00,
'Cancelled');

-- Insert into NOTIFICATION Table
INSERT INTO NOTIFICATION (NotiID, BookingID, Message)
VALUES ('N001', 'BK001', 'Your booking is confirmed for
2025-01-10.');
INSERT INTO NOTIFICATION (NotiID, BookingID, Message)
VALUES ('N002', 'BK002', 'Your booking is pending
approval.');
INSERT INTO NOTIFICATION (NotiID, BookingID, Message)
VALUES ('N003', 'BK003', 'Your booking has been cancelled
due to unavailability.');

-- Insert into PAYMENT Table
INSERT INTO PAYMENT (PaymentID, BookingID, PaymentMethod,
PaymentAmount, PaymentDate, PaymentStatus)
VALUES ('P001', 'BK001', 'Credit Card', 150.00,
'2025-01-10', 'Completed');
INSERT INTO PAYMENT (PaymentID, BookingID, PaymentMethod,
PaymentAmount, PaymentDate, PaymentStatus)
VALUES ('P002', 'BK002', 'Cash', 100.00, '2025-01-11',
'Pending');
```

```sql
INSERT INTO PAYMENT (PaymentID, BookingID, PaymentMethod,
PaymentAmount, PaymentDate, PaymentStatus)
VALUES ('P003', 'BK003', 'Credit Card', 180.00,
'2025-01-12', 'Pending');

-- Insert into CARDPAYMENT Table
INSERT INTO CARDPAYMENT (PaymentID, CardNumber,
ExpiryDate, CVV)
VALUES ('P001', 1234567890123456, '2027-12-31', 123);
INSERT INTO CARDPAYMENT (PaymentID, CardNumber,
ExpiryDate, CVV)
VALUES ('P002', 2345678901234567, '2026-06-30', 456);
INSERT INTO CARDPAYMENT (PaymentID, CardNumber,
ExpiryDate, CVV)
VALUES ('P003', 3456789012345678, '2025-01-15', 789);

-- Insert into CASHPAYMENT Table
INSERT INTO CASHPAYMENT (PaymentID, ReceiptNumber)
VALUES ('P001', 10001);
INSERT INTO CASHPAYMENT (PaymentID, ReceiptNumber)
VALUES ('P002', 10002);
INSERT INTO CASHPAYMENT (PaymentID, ReceiptNumber)
VALUES ('P003', 10003);

-- Insert into INCOMEANALYSIS Table
INSERT INTO INCOMEANALYSIS (AnalysisID, TotalIncome,
TotalExpenses, Profit)
VALUES ('A001', 100000.00, 75000.00, 25000.00);
INSERT INTO INCOMEANALYSIS (AnalysisID, TotalIncome,
TotalExpenses, Profit)
VALUES ('A002', 200000.00, 150000.00, 50000.00);
```

```
INSERT INTO INCOMEANALYSIS (AnalysisID, TotalIncome,
TotalExpenses, Profit)
VALUES ('A003', 150000.00, 120000.00, 30000.00);
```

Tables after inserting the data:

Staff

```
SELECT * FROM STAFF;
```

StaffID	StaffName	ContactNo	StaffPassword
S001	Ali Ibrahim	0123456789	password123
S002	Noraini Binti Ahmad	0198765432	mypassword
S003	Mohd Zaki Bin Salleh	0182345678	securepass
S004	Ahmad Zainal	0198765432	staffpass1
S005	Nur Aisyah Binti Mohd	0187654321	staffpass2

Driver

```
SELECT * FROM DRIVER;
```

StaffID	LicenseType	ExperienceYears	LicenseExpireDate
S001	Class D	5	2027-12-31
S002	Class E	3	2025-06-30
S003	Class D	6	2026-11-01
S004	Class D	4	2026-06-30
S005	Class B	5	2027-09-15

Admin

```
SELECT * FROM ADMIN;
```

StaffID	Department	Role
S001	Operations	Manager
S002	HR	Assistant
S003	Sales	Supervisor

Customer:

```
SELECT * FROM CUSTOMER;
```

CustomerID	CustName	Password	Email	SecurityQues	SecurityAns	Age	Gender	Address	CustPhone	CustType	DiscountRate
C001	Aisyah Binti Abdullah	pass1234	aisyah@example.com	Mother's maiden name	Salmah	22	F	Jalan Bukit Indah, Johor Bahru 81300	0129876543	Regular	0.10
C002	John Tan	mypassword	john@example.com	First pet's name	Buddy	30	M	Taman Johor Jaya, Johor Bahru 81300	0135678923	VIP	0.15
C003	Rina Mohamed	securepass	rina@example.com	Favourite teacher	Mr. Lim	25	F	Jalan Setia, Johor Bahru 81300	0162348789	Regular	0.05
C004	Farah Azizah	farahpass	farah@example.com	First teacher's name	Ms. Zainab	28	F	Taman Melawati, Kuala Lumpur 53300	0171234567	VIP	0.20
C005	Hassan Ali	hassan2025	hassan@example.com	First car model	Toyota	35	M	Jalan Anggerik, Penang 10000	0147654321	Regular	0.10

Bus:

```
SELECT * FROM BUS;
```

BusID	StaffID	Model	SeatCapacity	BookingPrice	Availability
B001	S001	RapidKL	15	100.00	Unavailable
B002	S002	GoKL	20	120.00	Available
B003	S003	Penang Bus Service	12	80.00	Available

Booking:

```
SELECT * FROM BOOKING;
```

BookingID	CustomerID	BusID	StaffID	BookingDate	PickUpLocation	DropOffLocation	NumOfPassengers	TotalPrice	Status
BK001	C001	B001	S001	2025-01-10	Johor Bahru 81300	Kuala Lumpur	2	150.00	Confirmed
BK002	C002	B002	S002	2025-01-11	Johor Bahru 81300	Penang	1	100.00	Pending
BK003	C001	B003	S003	2025-01-12	Johor Bahru 81300	Kuala Lumpur	3	180.00	Cancelled

Notification:

```
SELECT * FROM NOTIFICATION;
```

NotiID	BookingID	Message
N001	BK001	Your booking is confirmed for 2025-01-10.
N002	BK002	Your booking is pending approval.
N003	BK003	Your booking has been cancelled due to unavailability.

Payment:

```
SELECT * FROM PAYMENT;
```

PaymentID	BookingID	PaymentMethod	PaymentAmount	PaymentDate	PaymentStatus
P001	BK001	Credit Card	150.00	2025-01-10	Completed
P002	BK002	Cash	100.00	2025-01-11	Pending
P003	BK003	Credit Card	180.00	2025-01-12	Pending

CardPayment:

```
SELECT * FROM CARDPAYMENT;
```

PaymentID	CardNumber	ExpiryDate	CVV
P001	1234567890123456	2027-12-31	123
P002	2345678901234567	2026-06-30	456
P003	3456789012345678	2025-01-15	789

CashPayment:

```
SELECT * FROM CASHPAYMENT;
```

PaymentID	ReceiptNumber
P001	10001
P002	10002
P003	10003

IncomeAnalysis:

```
SELECT * FROM INCOMEANALYSIS;
```

AnalysisID	TotalIncome	TotalExpenses	Profit
A001	100000.00	75000.00	25000.00
A002	200000.00	150000.00	50000.00
A003	150000.00	120000.00	30000.00

1. Customer Booking a Bus

Customer C004 (Aisyah Binti Abdullah) books a bus (B003) for the trip from Johor Bahru to Kuala Lumpur

Adding a new booking entry to the BOOKING table:

```
INSERT INTO BOOKING (BookingID, CustomerID, BusID,
StaffID, BookingDate, PickUpLocation, DropOffLocation,
NumOfPassengers, TotalPrice, Status)
VALUES ('BK004', 'C004', 'B003', 'S001', '2025-01-15',
'Johor Bahru 81300', 'Kuala Lumpur', 12, 80.00,
'Confirmed');
```

2. Updating Bus availability to 'Unavailable' once it's booked

```
UPDATE BUS
SET Availability = 'Unavailable'
WHERE BusID = 'B003';
```

3. Notification for the Booking

A notification is sent to the customer confirming the booking status

Adding a new notification to the NOTIFICATION table:

```
INSERT INTO NOTIFICATION (NotiID, BookingID, Message)
VALUES ('N004', 'BK004', 'Your booking is confirmed for
2025-01-15.');
```

4. Payment for Booking

Customer C001 (Aisyah Binti Abdullah) proceeds to pay for the booking via Credit Card

Adding a new payment entry to the PAYMENT table:

```
INSERT INTO PAYMENT (PaymentID, BookingID, PaymentMethod,
PaymentAmount, PaymentDate, PaymentStatus)
VALUES ('P004', 'BK004', 'Credit Card', 80.00,
'2025-01-15', 'Completed');
```

5. Card Payment Details

Since the payment method is Credit Card, we add the card details to the CARDPAYMENT table

Adding a new entry for the payment in the CARDPAYMENT table:

```
INSERT INTO CARDPAYMENT (PaymentID, CardNumber,
ExpiryDate, CVV)
VALUES ('P004', 4123456789012345, '2027-12-31', 123);
```

Show the updated Bus table:

```
SELECT * FROM BUS;
```

BusID	StaffID	Model	SeatCapacity	BookingPrice	Availability
B001	S001	RapidKL	15	100.00	Unavailable
B002	S002	GoKL	20	120.00	Available
B003	S003	Penang Bus Service	12	80.00	Unavailable

Show the updated Booking table with the new booking entry:

```
SELECT * FROM BOOKING;
```

BookingID	CustomerID	BusID	StaffID	BookingDate	PickUpLocation	DropOffLocation	NumOfPassengers	TotalPrice	Status
BK001	C001	B001	S001	2025-01-10	Johor Bahru 81300	Kuala Lumpur	2	150.00	Confirmed
BK002	C002	B002	S002	2025-01-11	Johor Bahru 81300	Penang	1	100.00	Pending
BK003	C001	B003	S003	2025-01-12	Johor Bahru 81300	Kuala Lumpur	3	180.00	Cancelled
BK004	C004	B003	S001	2025-01-15	Johor Bahru 81300	Kuala Lumpur	12	80.00	Confirmed

Show the updated Notification table with the new Notification entry:

```
SELECT * FROM NOTIFICATION;
```

NotiID	BookingID	Message
N001	BK001	Your booking is confirmed for 2025-01-10.
N002	BK002	Your booking is pending approval.
N003	BK003	Your booking has been cancelled due to unavailability.
N004	BK004	Your booking is confirmed for 2025-01-15.

Show the updated Payment table with the new Payment entry:

```
SELECT * FROM PAYMENT;
```

PaymentID	BookingID	PaymentMethod	PaymentAmount	PaymentDate	PaymentStatus
P001	BK001	Credit Card	150.00	2025-01-10	Completed
P002	BK002	Cash	100.00	2025-01-11	Pending
P003	BK003	Credit Card	180.00	2025-01-12	Pending
P004	BK004	Credit Card	80.00	2025-01-15	Completed

Show the updated CardPayment table with the new CardPayment entry:

```
SELECT * FROM CARDPAYMENT;
```

PaymentID	CardNumber	ExpiryDate	CVV
P001	1234567890123456	2027-12-31	123
P002	23456789012345 67	2026-06-30	456
P003	3456789012345678	2025-01-15	789
P004	4123456789012345	2027-12-31	123

Cancelling a Booking:

Update the status of the payment in the PAYMENT table to 'Cancelled'

```
UPDATE PAYMENT
SET PaymentStatus = 'Cancelled'
WHERE BookingID = 'BK003'
  AND PaymentStatus != 'Cancelled';
```

PaymentID	BookingID	PaymentMethod	PaymentAmount	PaymentDate	PaymentStatus
P001	BK001	Credit Card	150.00	2025-01-10	Completed
P002	BK002	Cash	100.00	2025-01-11	Pending
P003	BK003	Credit Card	180.00	2025-01-12	Cancelled
P004	BK004	Credit Card	80.00	2025-01-15	Completed

Remove the corresponding CARDPAYMENT entry

```
DELETE FROM CARDPAYMENT
WHERE PaymentID = 'P003';
```

PaymentID	CardNumber	ExpiryDate	CVV
P001	1234567890123456	2027-12-31	123
P002	2345678901234567	2026-06-30	456
P004	4123456789012345	2027-12-31	123

Update the booking status in the BOOKING table to 'Cancelled'

```
UPDATE BOOKING
SET Status = 'Cancelled'
WHERE BookingID = 'BK003';
```

BookingID	CustomerID	BusID	StaffID	BookingDate	PickUpLocation	DropOffLocation	NumOfPassengers	TotalPrice	Status
BK001	C001	B001	S001	2025-01-10	Johor Bahru 81300	Kuala Lumpur	2	150.00	Confirmed
BK002	C002	B002	S002	2025-01-11	Johor Bahru 81300	Penang	1	100.00	Pending
BK003	C001	B003	S003	2025-01-12	Johor Bahru 81300	Kuala Lumpur	3	180.00	Cancelled
BK004	C004	B003	S001	2025-01-15	Johor Bahru 81300	Kuala Lumpur	12	80.00	Confirmed

Insert a notification for the cancellation of the booking in the NOTIFICATION table

```
INSERT INTO NOTIFICATION (NotiID, BookingID, Message)
VALUES ('N005', 'BK003', 'Your booking has been
cancelled.');
```

NotiID	BookingID	Message
N001	BK001	Your booking is confirmed for 2025-01-10.
N002	BK002	Your booking is pending approval.
N003	BK003	Your booking has been cancelled due to unavailability.
N004	BK004	Your booking is confirmed for 2025-01-15.
N005	BK003	Your booking has been cancelled.

6.3 Test Query

6.3.1 View Staff Table

-- View Staff Table

SELECT * FROM STAFF;

StaffID	StaffName	ContactNo	StaffPassword
S001	Ali Ibrahim	0123456789	password123
S002	Noraini Binti Ahmad	0198765432	mypassword
S003	Mohd Zaki Bin Salleh	0182345678	securepass

6.3.2 View Driver Table

-- View Driver Table

SELECT * FROM DRIVER;

StaffID	LicenseType	ExperienceYears	LicenseExpireDate
S001	Class D	5	2027-12-31
S002	Class E	3	2025-06-30
S003	Class D	6	2026-11-01

6.3.3 View Admin Table

-- View Admin Table

SELECT * FROM ADMIN;

StaffID	Department	Role
S001	Operations	Manager
S002	HR	Assistant
S003	Sales	Supervisor

6.3.4 View Customer Table

-- View Customer Table

SELECT * FROM CUSTOMER;

CustomerID	CustName	Password	Email	SecurityQues	SecurityAns	Age	Gender	Address	CustPhone	CustType	DiscountRate
C001	Aisyah Binti Abdullah	pass1234	aisyah@example.com	Mother's maiden name	Salmah	22	F	Jalan Bukit Indah, Johor Bahru 81300	0129876543	Regular	0.10
C002	John Tan	mypassword	john@example.com	First pet's name	Buddy	30	M	Taman Johor Jaya, Johor Bahru 81300	0135678923	VIP	0.15
C003	Rina Mohamed	securepass	rina@example.com	Favourite teacher	Mr. Lim	25	F	Jalan Setia, Johor Bahru 81300	0162345789	Regular	0.05

6.3.5 View Booking Table

```
-- View Booking Table
SELECT * FROM BOOKING;
```

BookingID	CustomerID	BusID	StaffID	BookingDate	PickUpLocation	DropOffLocation	NumOfPassengers	TotalPrice	Status
BK001	C001	B001	S001	2025-01-10	Johor Bahru 81300	Kuala Lumpur	2	150.00	Confirmed
BK002	C002	B002	S002	2025-01-11	Johor Bahru 81300	Penang	1	100.00	Pending
BK003	C001	B003	S003	2025-01-12	Johor Bahru 81300	Kuala Lumpur	3	180.00	Cancelled
BK004	C001	B001	S001	2025-01-15	Johor Bahru 81300	Kuala Lumpur	2	200.00	Confirmed

6.3.6 View Notification Table

```
-- View Notification Table
SELECT * FROM NOTIFICATION;
```

NotiID	BookingID	Message
N001	BK001	Your booking is confirmed for 2025-01-10.
N002	BK002	Your booking is pending approval.
N003	BK003	Your booking has been cancelled due to unavailability.
N004	BK004	Your booking is confirmed for 2025-01-15.
N005	BK003	Your booking has been cancelled.

6.3.7 View Bus Table

```
-- View Bus Table
SELECT * FROM BUS;
```

BusID	StaffID	Model	SeatCapacity	BookingPrice	Availability
B001	S001	RapidKL	15	100.00	Available
B002	S002	GoKL	20	120.00	Unavailable
B003	S003	Penang Bus Service	12	80.00	Available

6.3.8 View Payment Table

```
-- View Payment Table
SELECT * FROM PAYMENT;
```

PaymentID	BookingID	PaymentMethod	PaymentAmount	PaymentDate	PaymentStatus
P001	BK001	Credit Card	150.00	2025-01-10	Completed
P002	BK002	Cash	100.00	2025-01-11	Pending
P003	BK003	Credit Card	180.00	2025-01-12	Cancelled
P004	BK004	Credit Card	200.00	2025-01-15	Completed

6.3.9 View Card Payment Table

```
-- View Card Payment Table
SELECT * FROM CARDPAYMENT;
```

PaymentID	CardNumber	ExpiryDate	CVV
P001	1234567890123456	2027-12-31	123
P002	2345678901234567	2026-06-30	456
P004	4123456789012345	2027-12-31	123

6.3.10 View Cash Payment Table

```
-- View Cash Payment Table
SELECT * FROM CASHPAYMENT;
```

PaymentID	ReceiptNumber
P001	10001
P002	10002
P003	10003

6.3.11 View Income Analysis Table

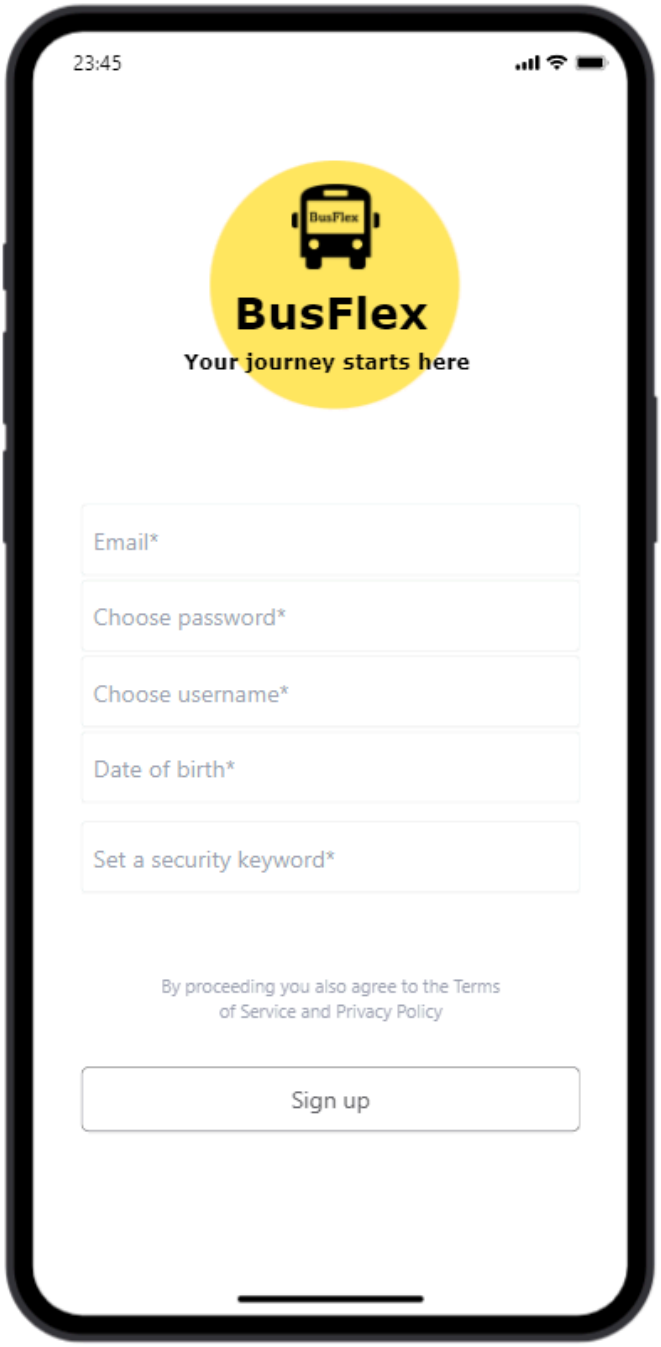
```
-- View Income Analysis Table
SELECT * FROM INCOMEANALYSIS;
```

AnalysisID	TotalIncome	TotalExpenses	Profit
A001	100000.00	75000.00	25000.00
A002	200000.00	150000.00	50000.00
A003	120000.00	120000.00	0.00

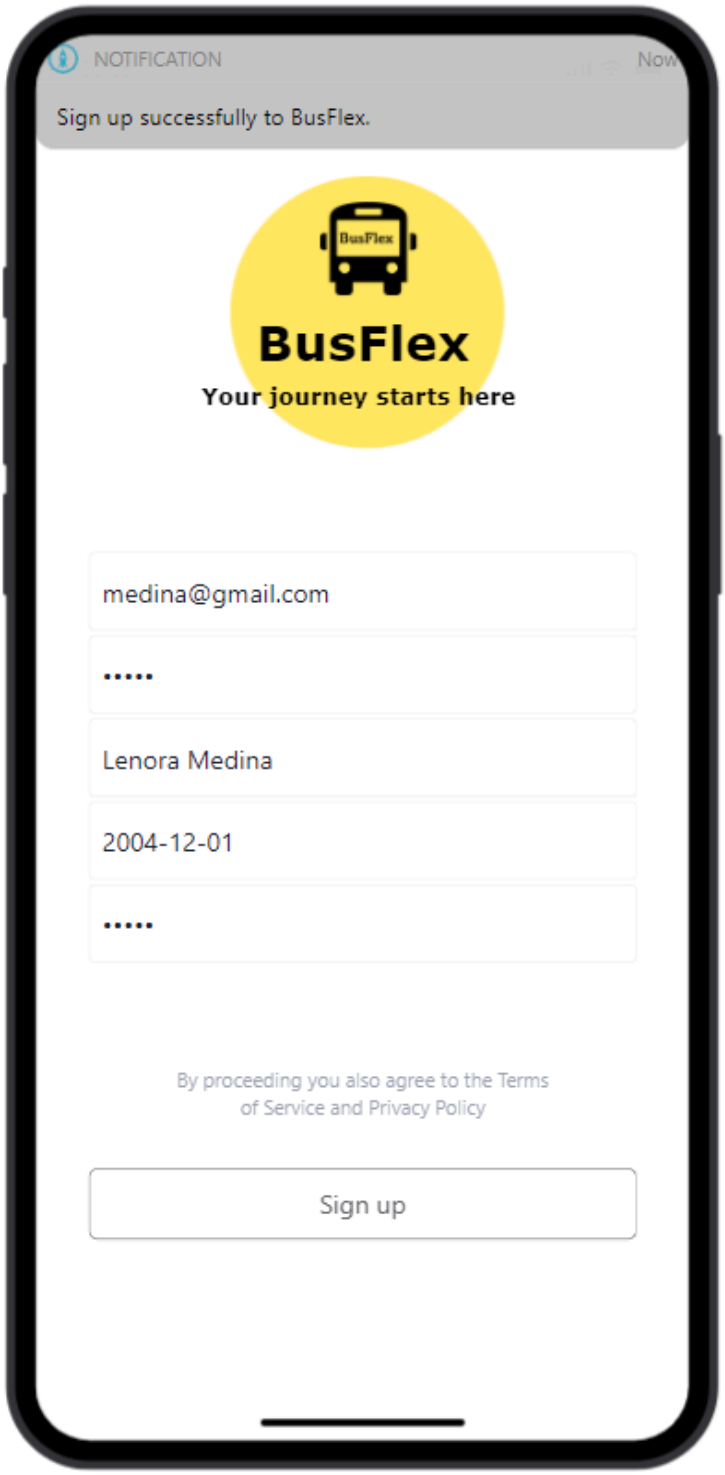
7.0 Interface

7.1 Welcome Page

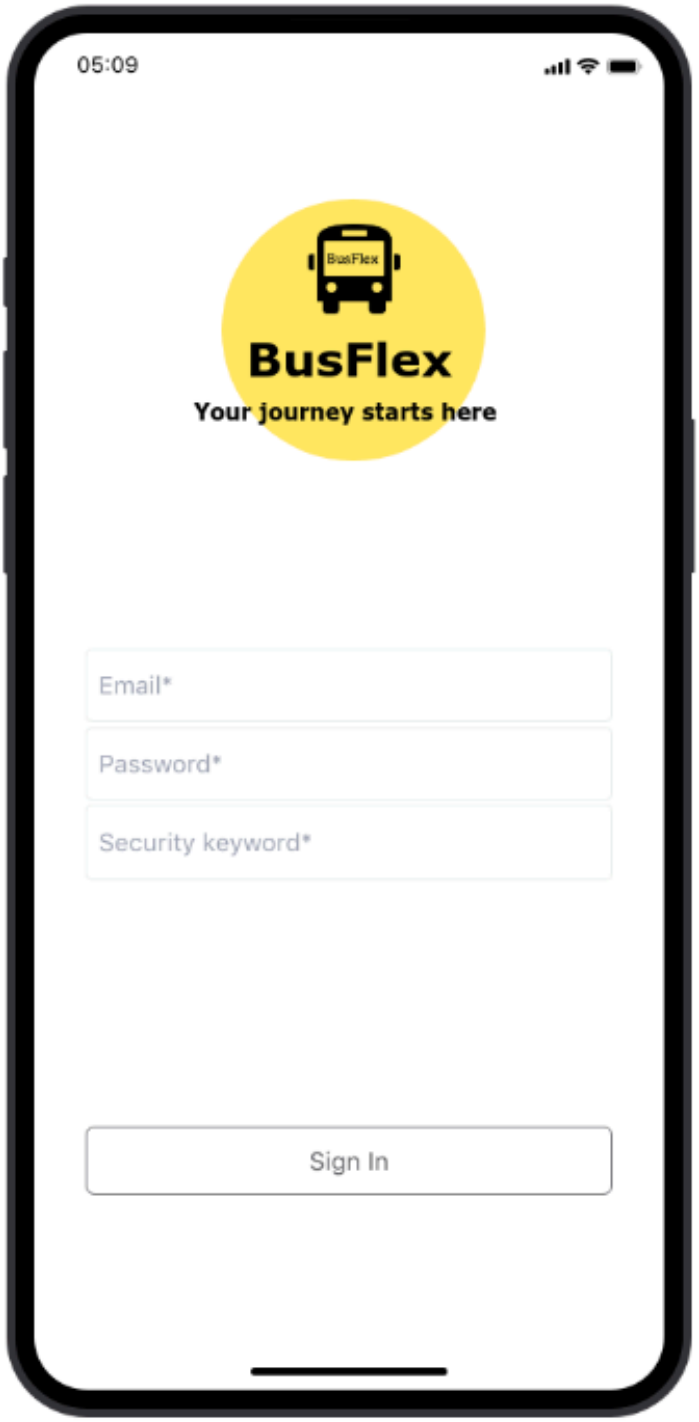
7.2 User Sign Up Page



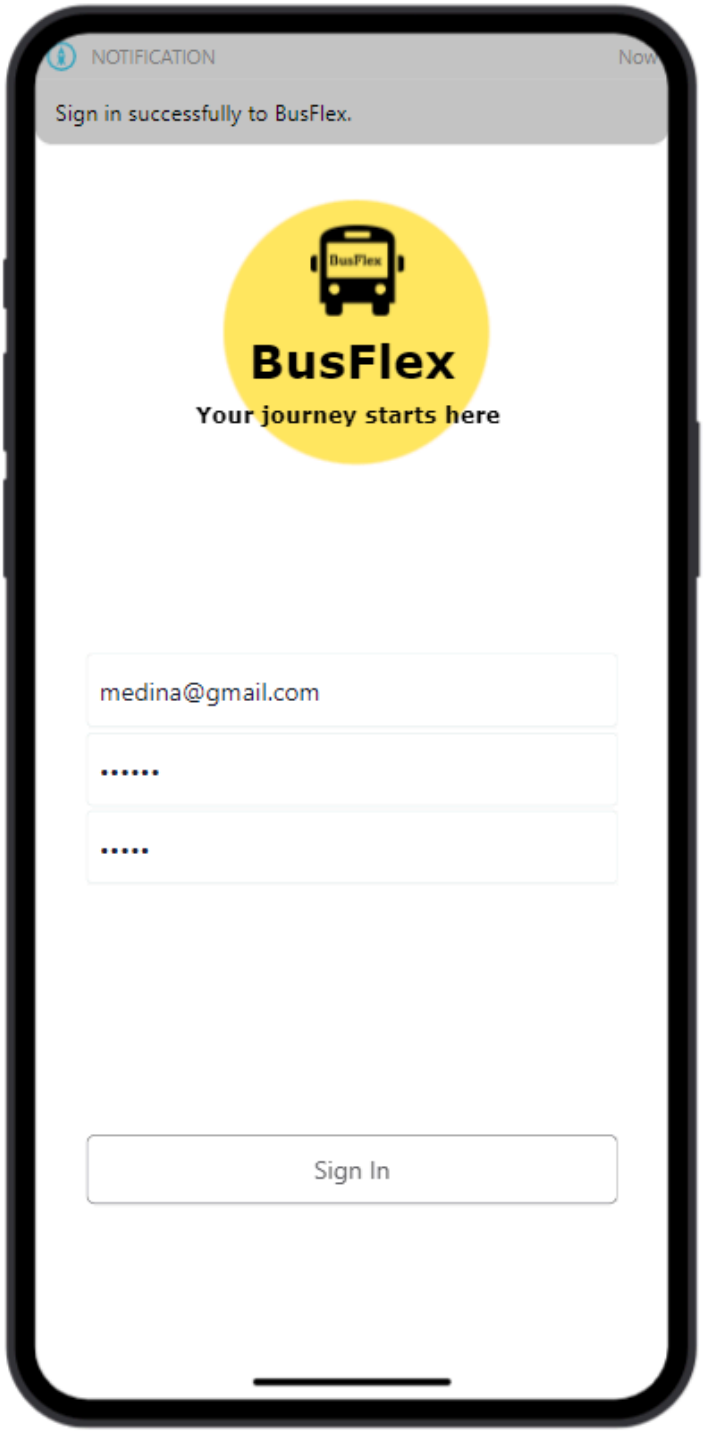
7.3 User Sign Up Notification



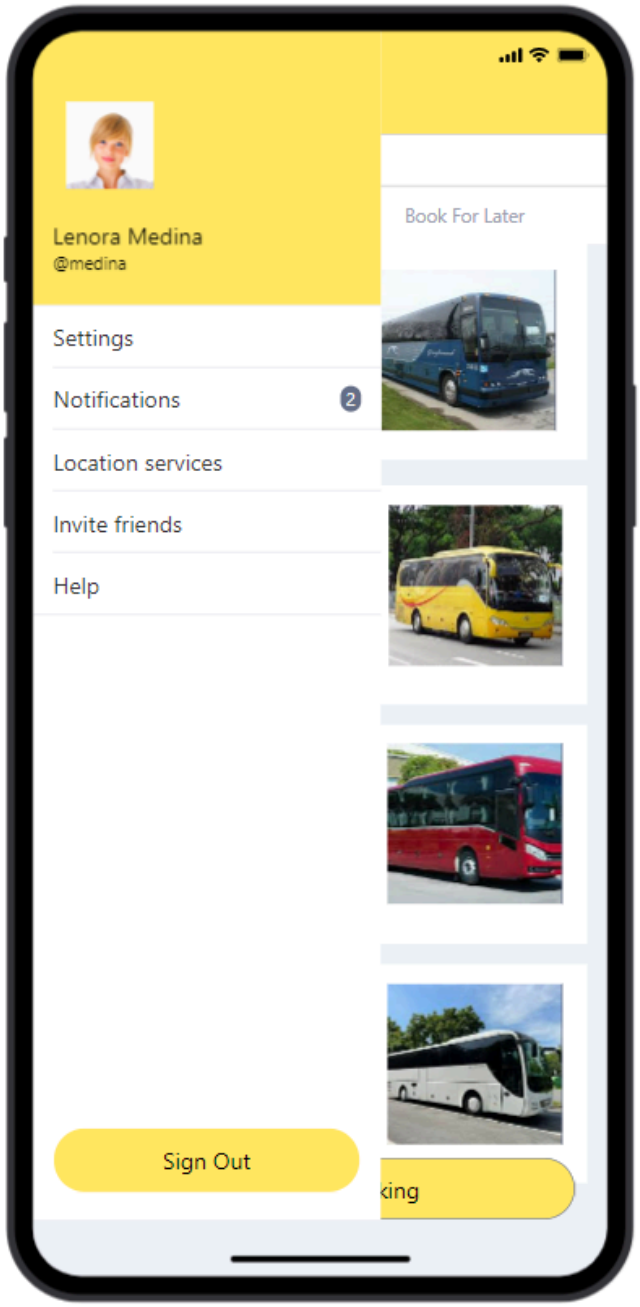
7.4 User Sign In Page



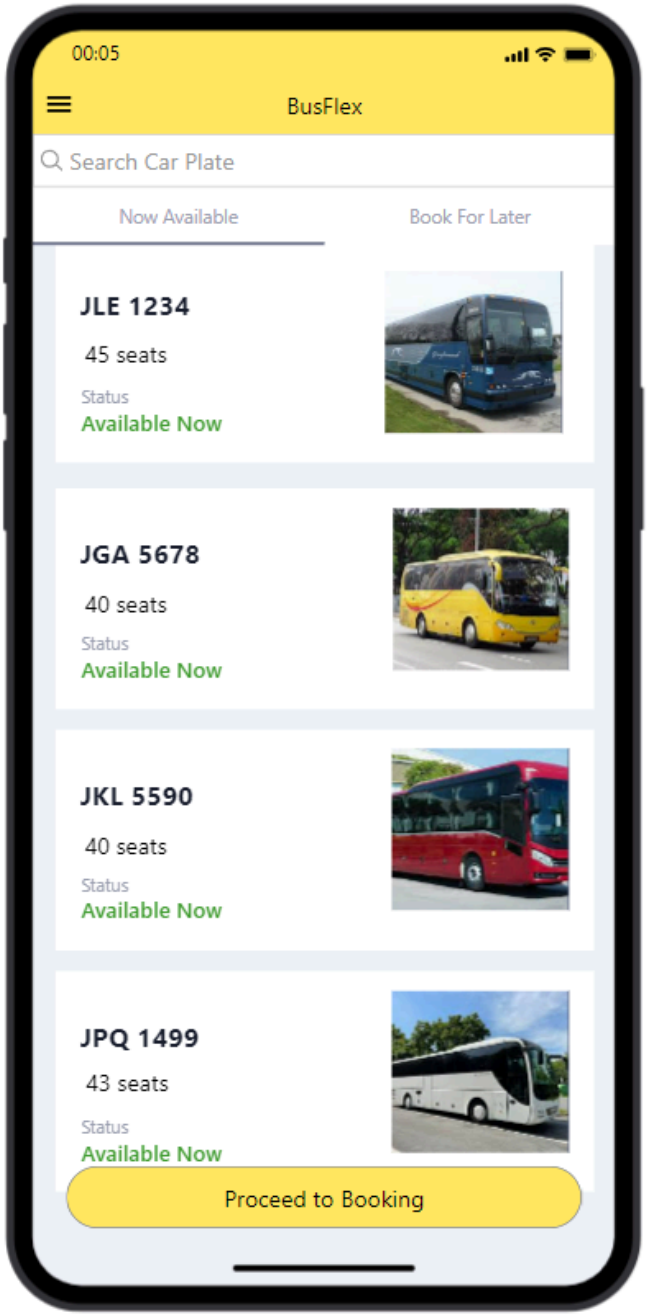
7.5 User Sign In Notification



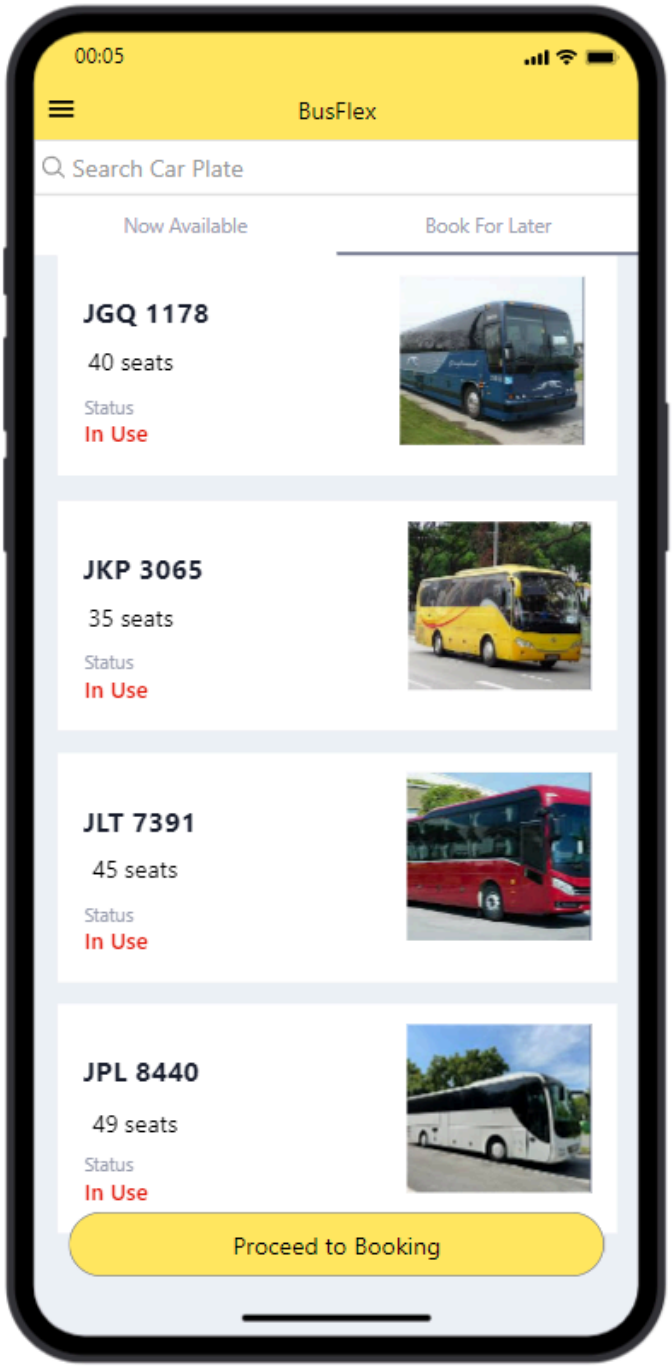
7.6 User Profile



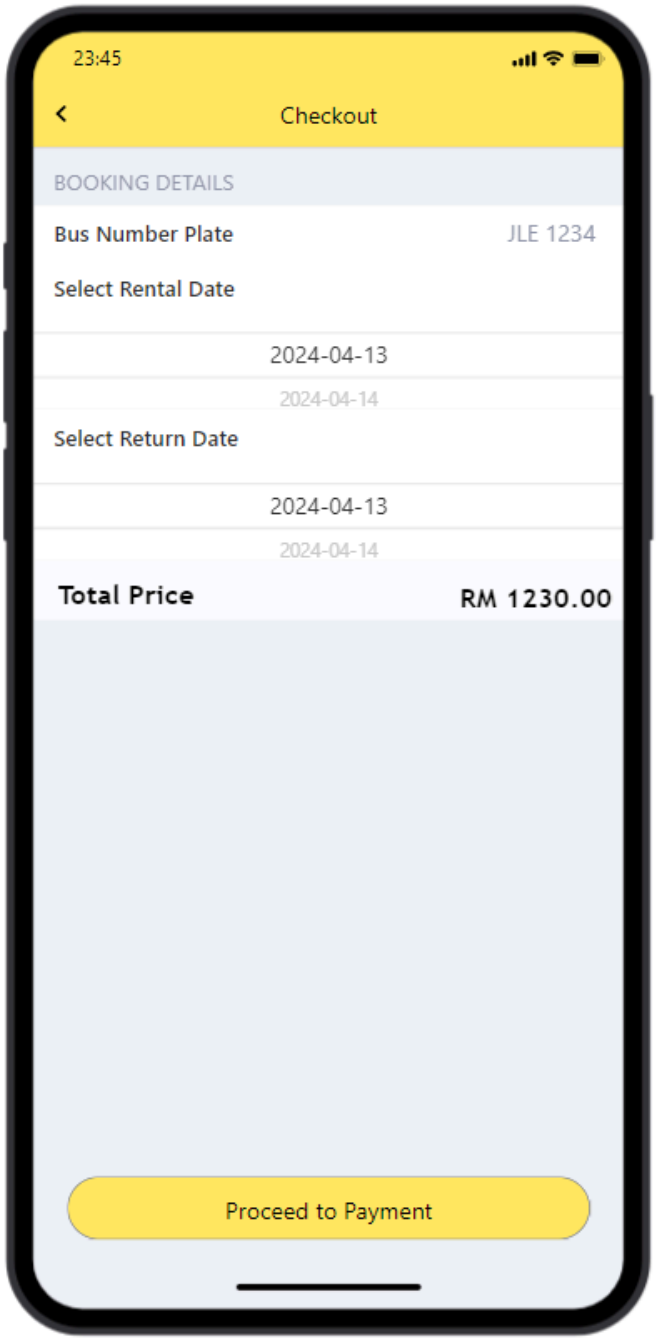
7.7 User Choose Available Buses Page



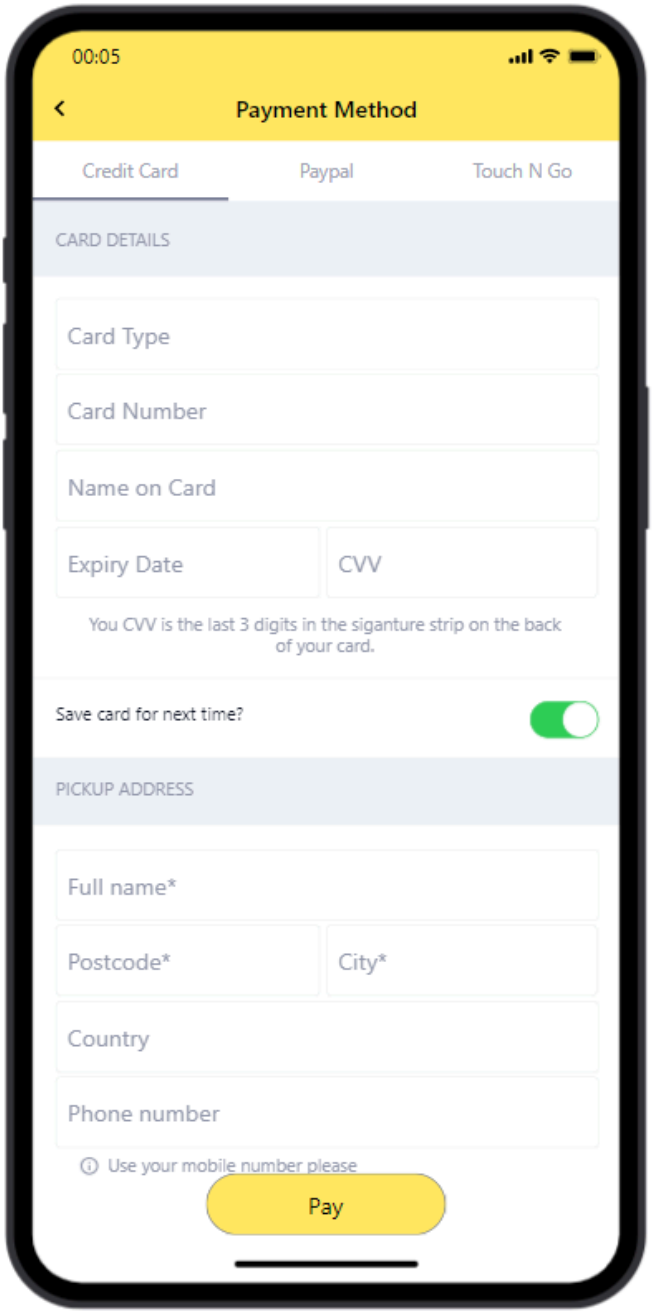
7.8 User Check In-Use Buses Page



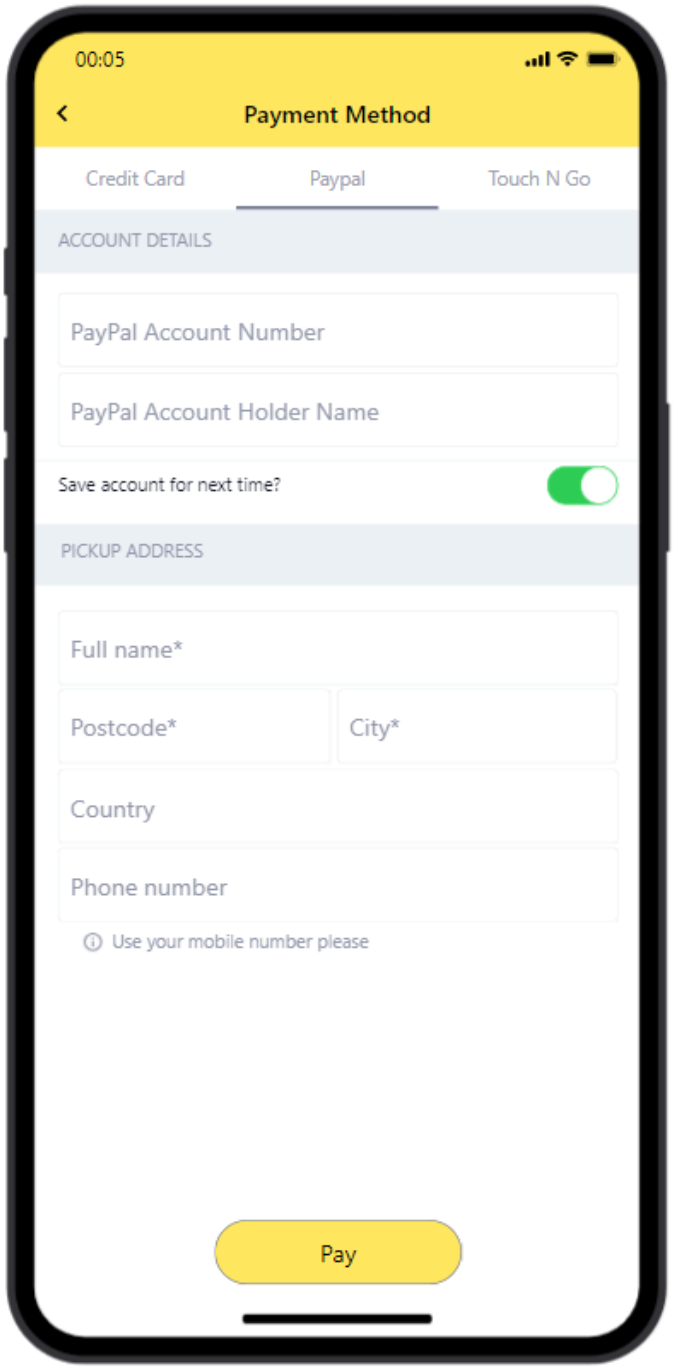
7.9 User Make Booking Page



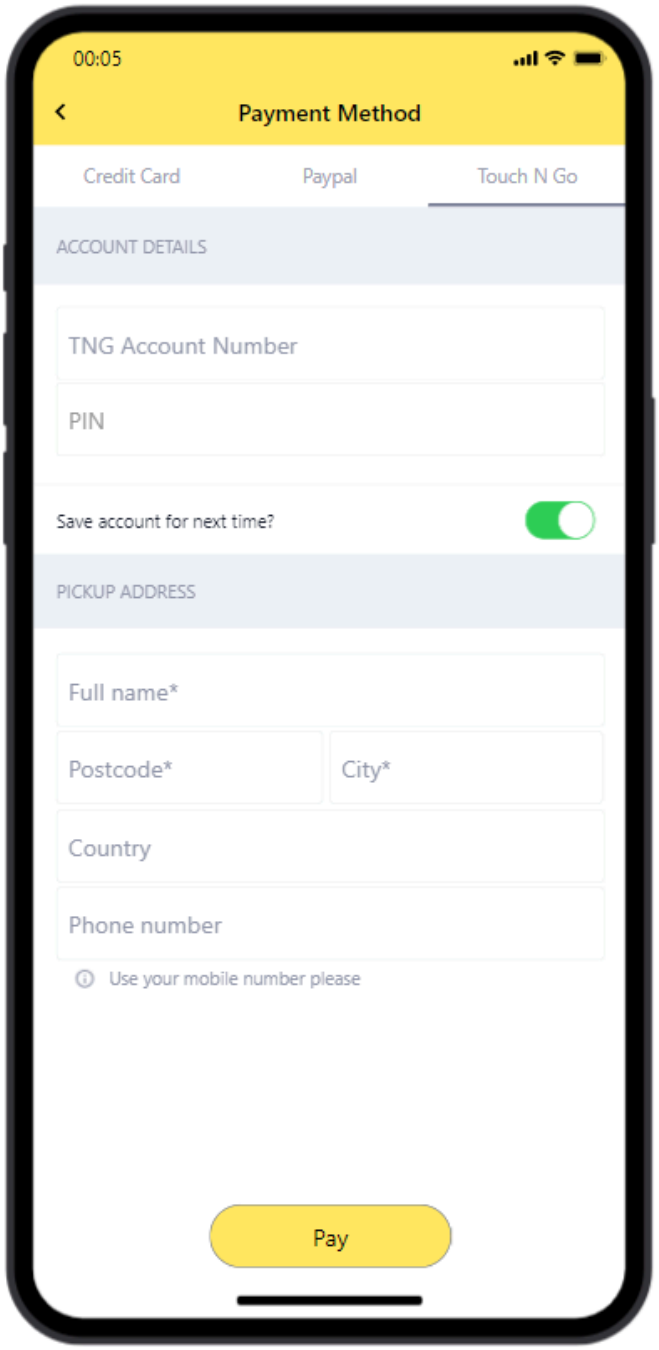
7.10 Payment Page - Credit Card



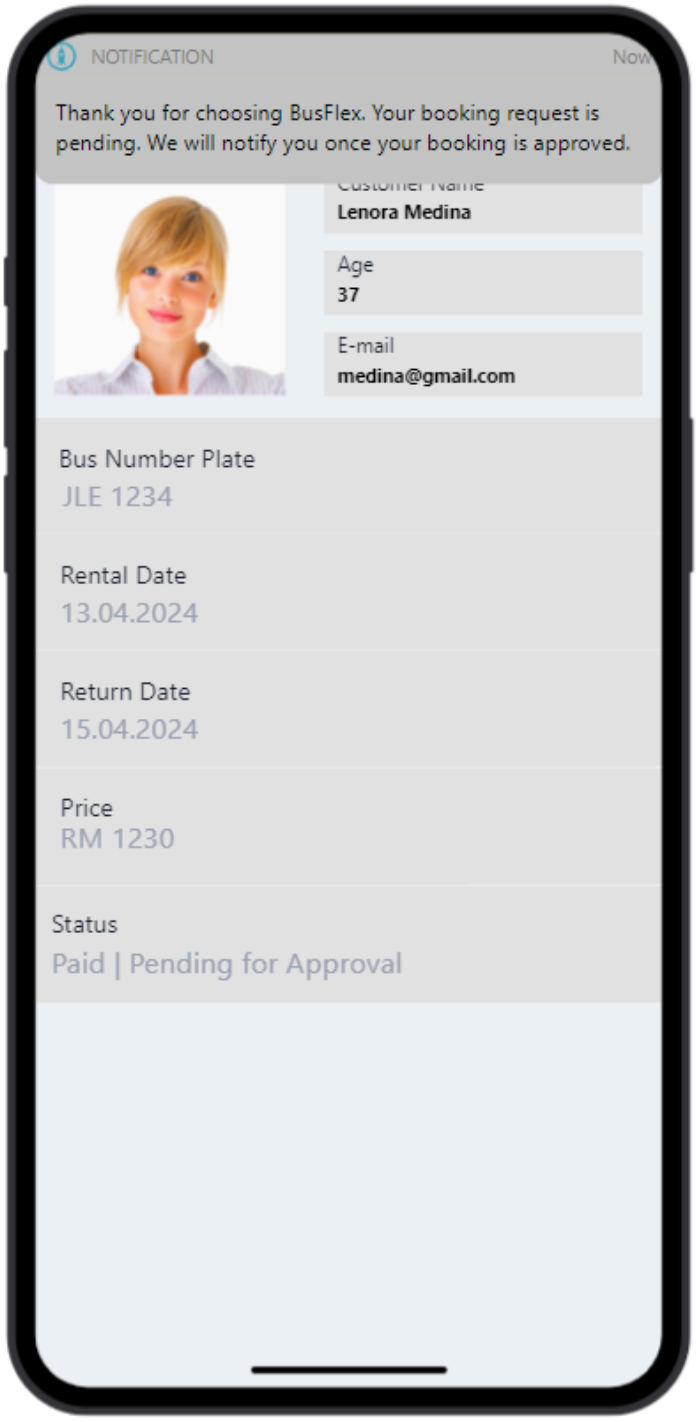
7.11 Payment Page - PayPal



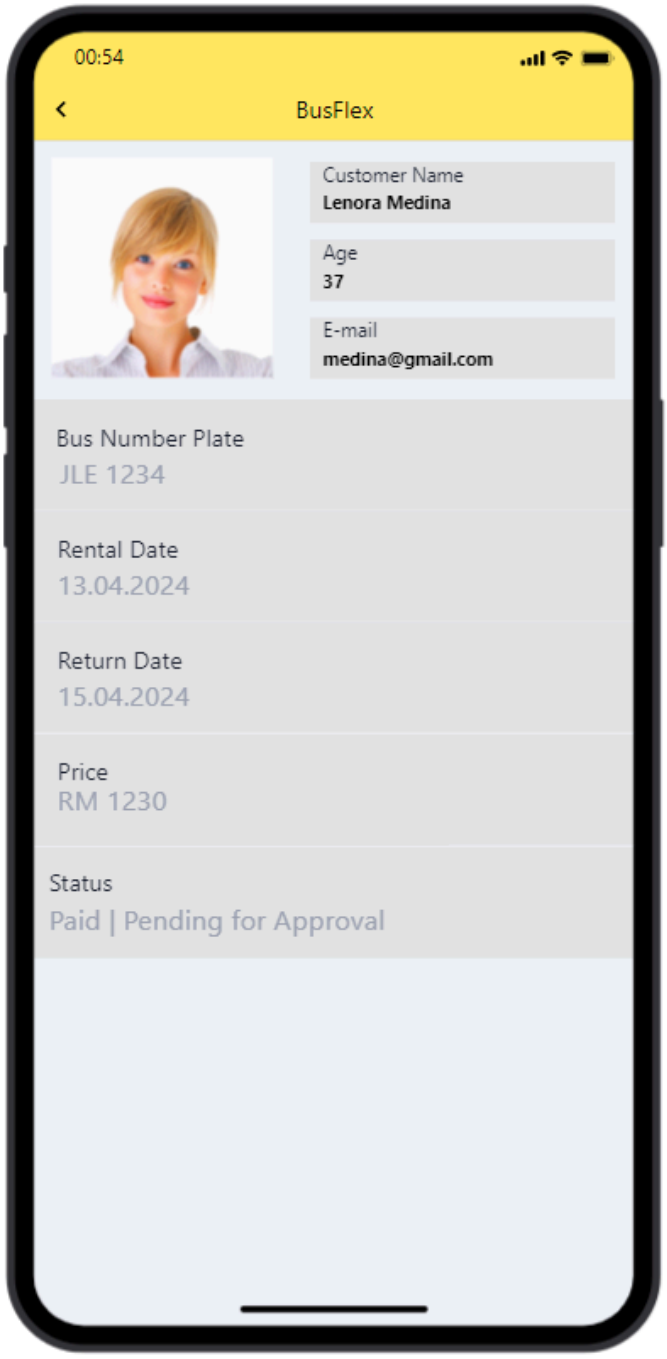
7.12 Payment Page - Touch N Go



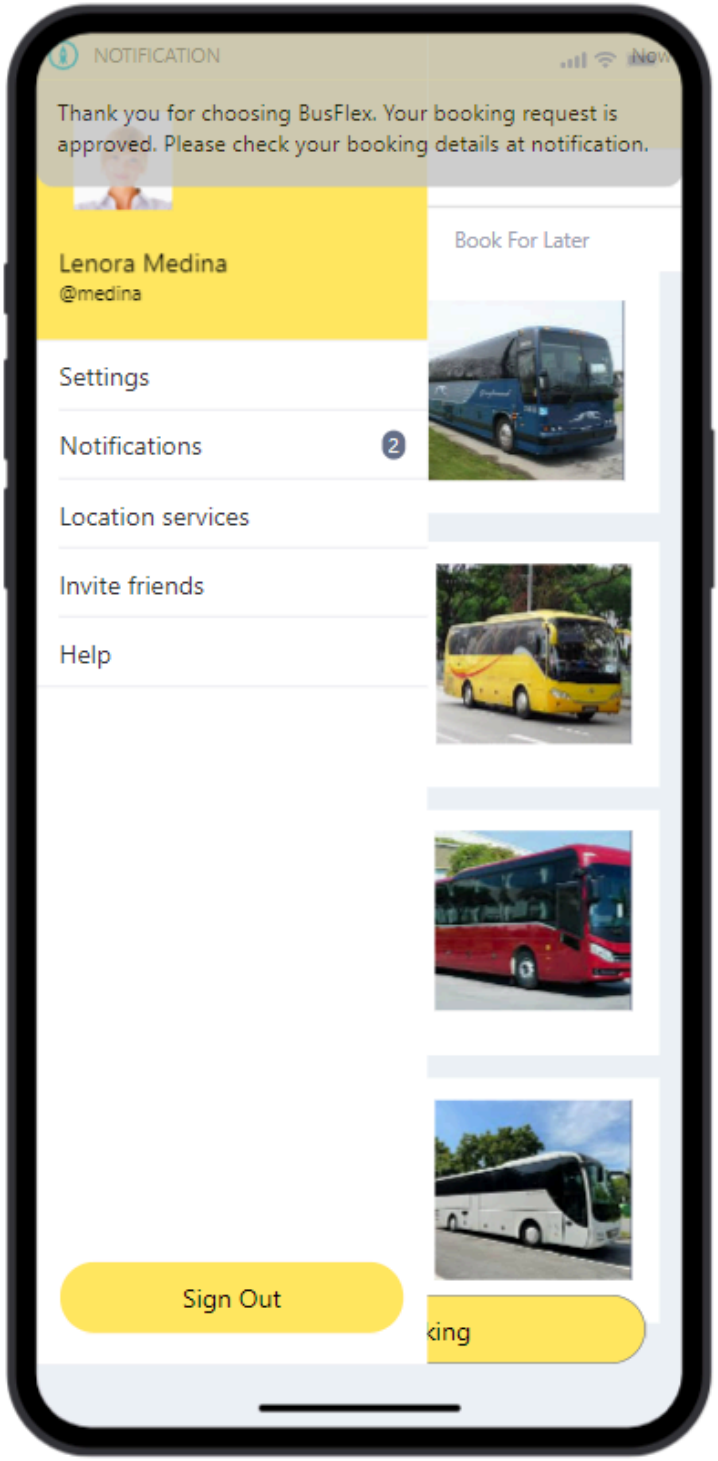
7.13 Booking Pending Notification



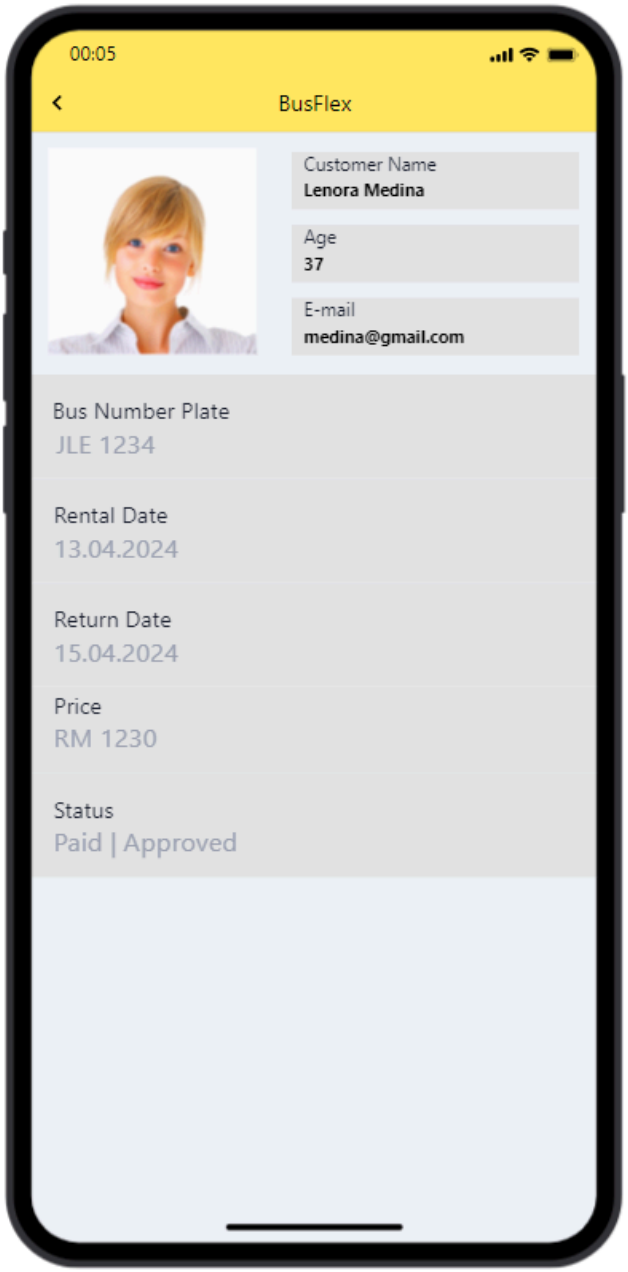
7.14 Booking Pending Page



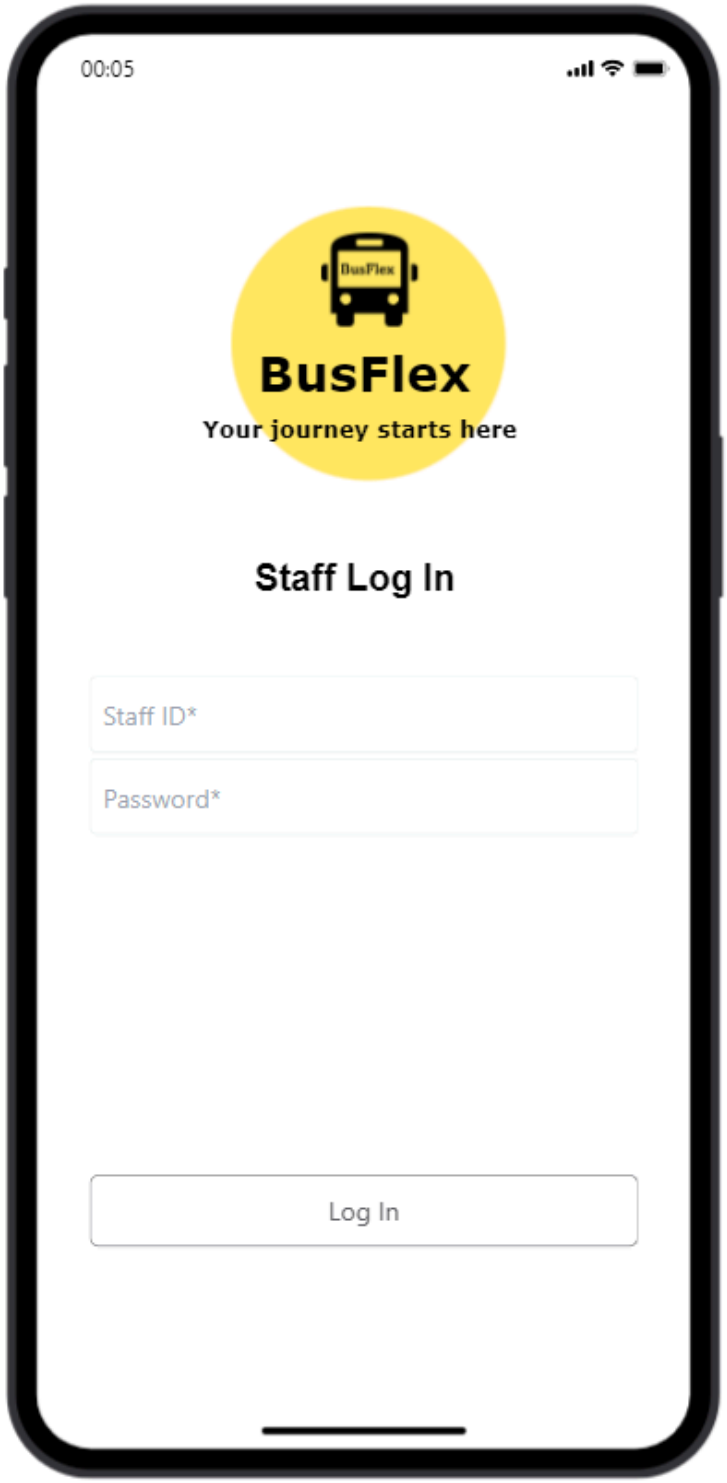
7.15 Notification after booking is approved



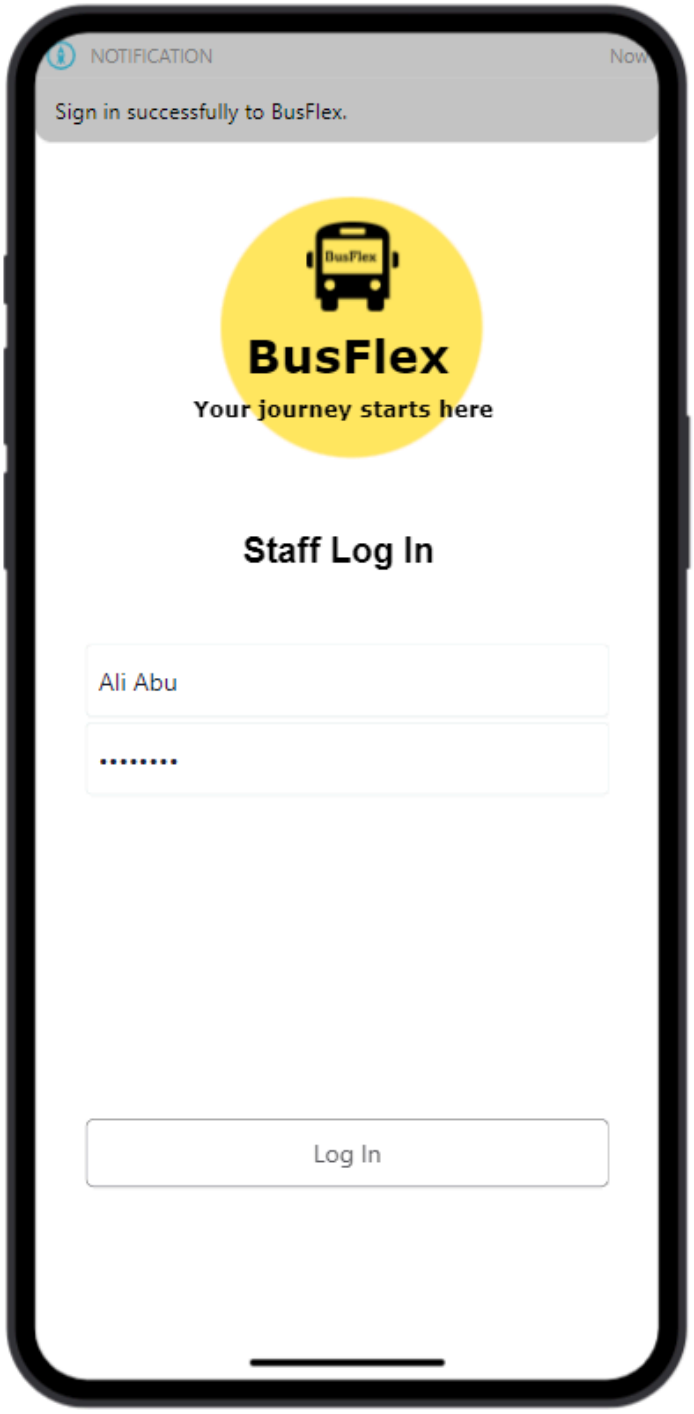
7.16 Approved Booking Page



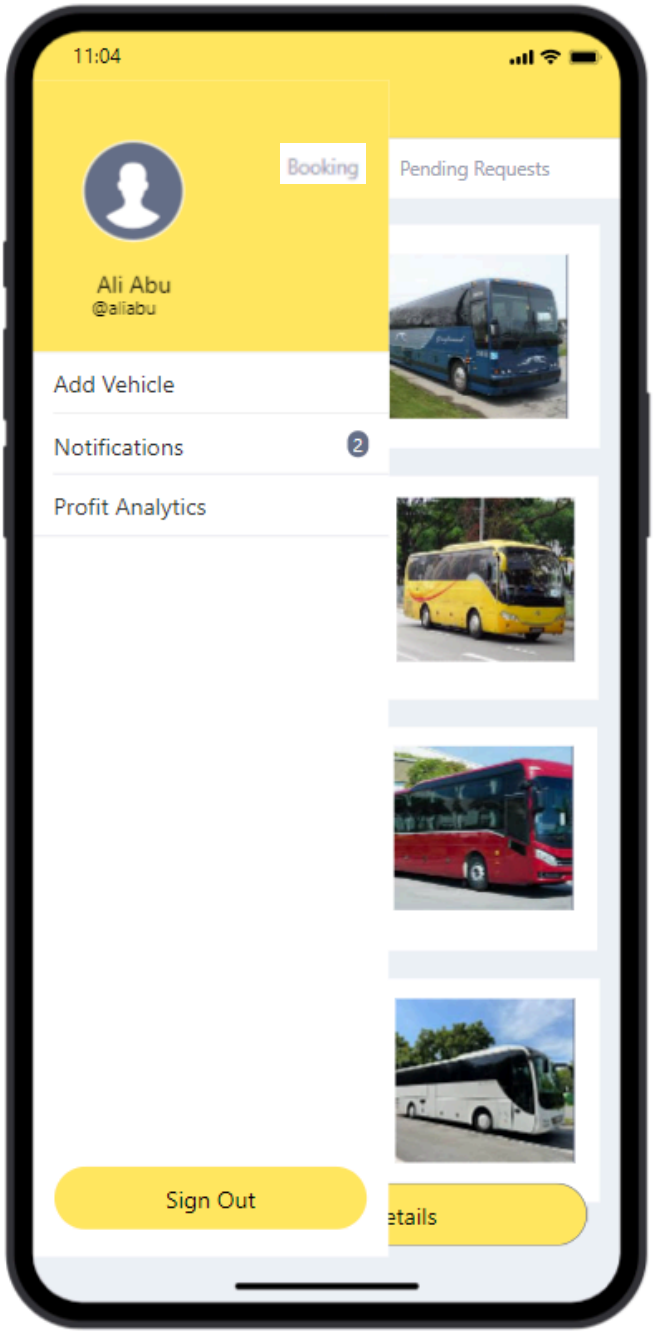
7.17 Staff Sign In Page



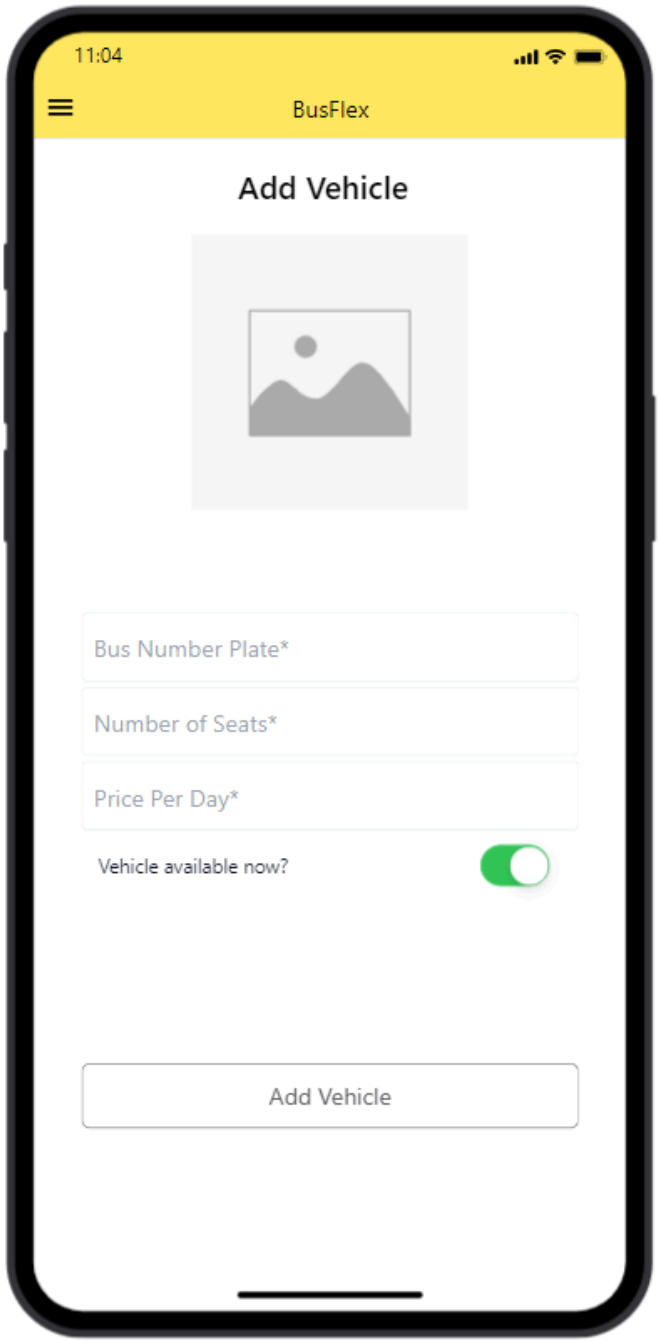
7.18 Staff Sign In Notification



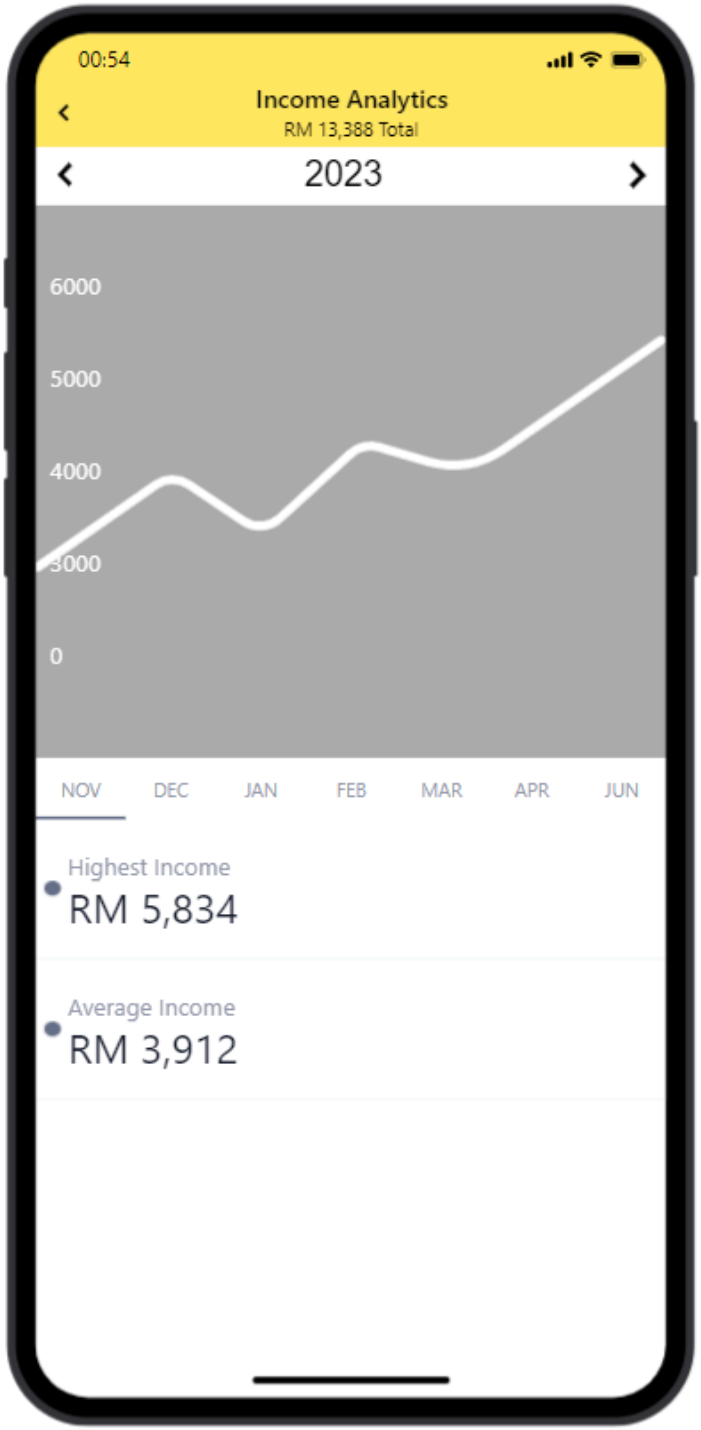
7.19 Staff Profile



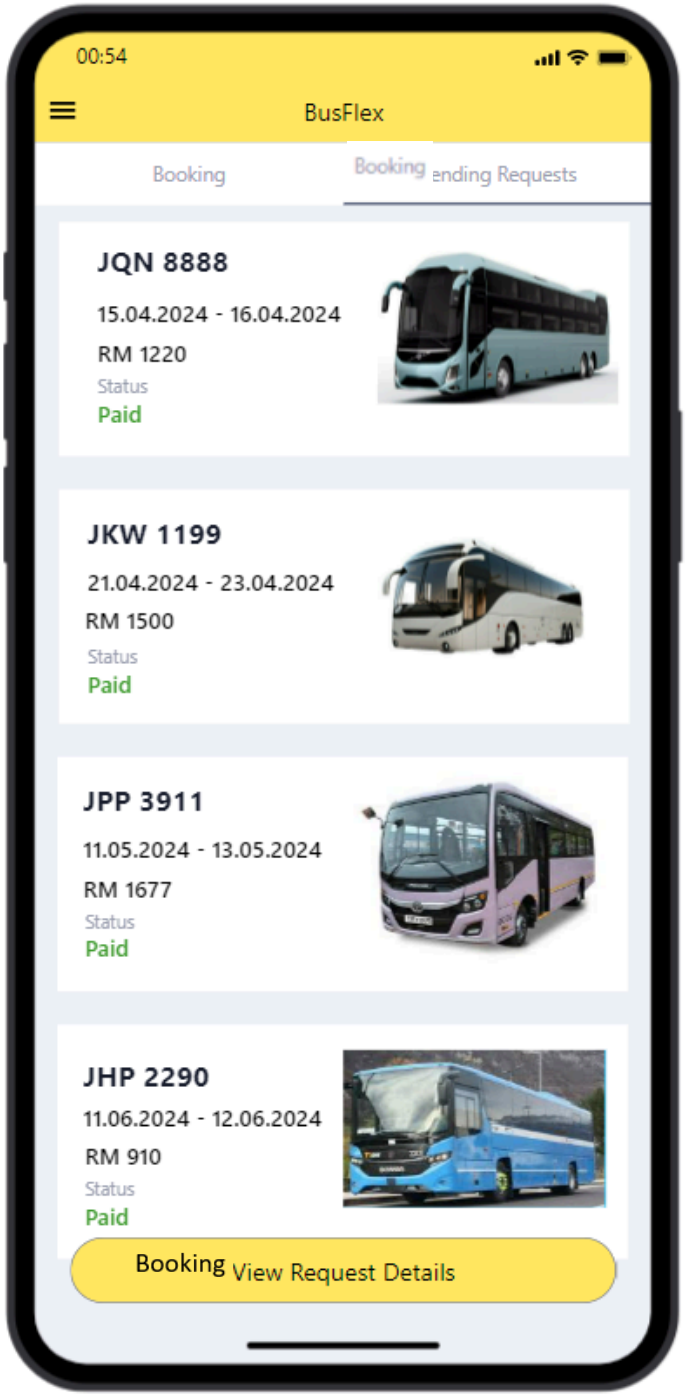
7.20 Add Vehicle Page



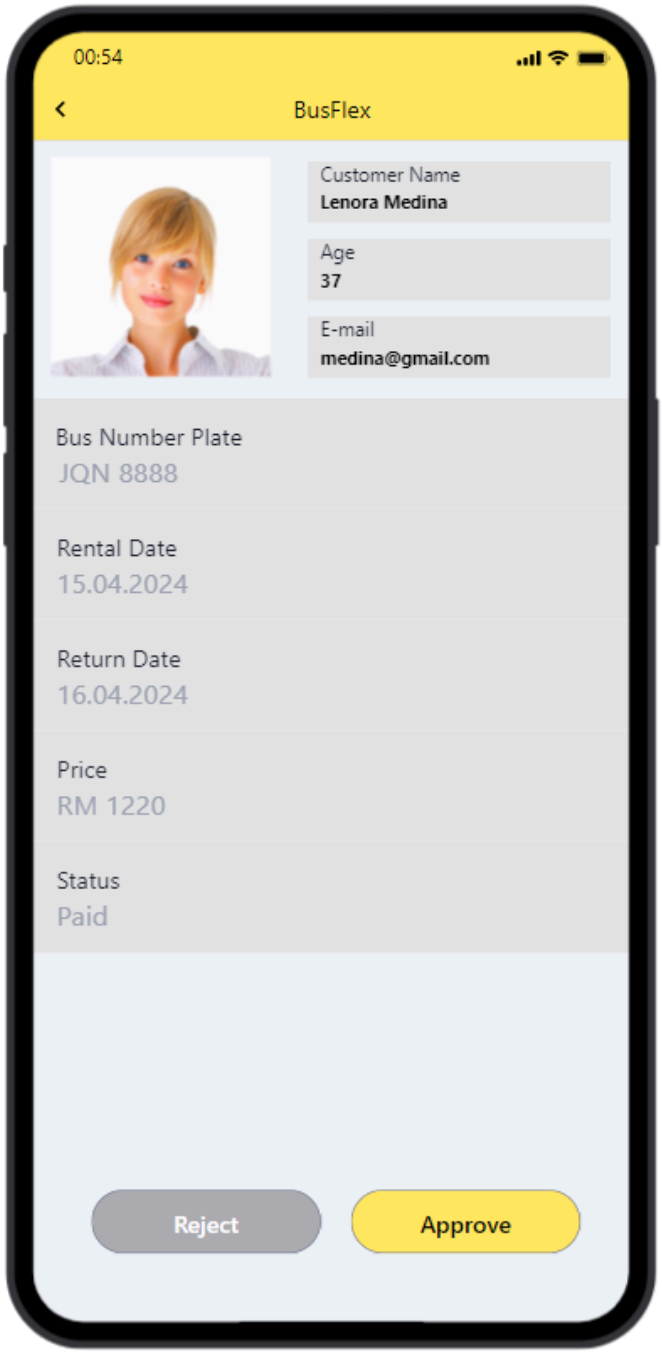
7.21 Profit Analytics Page



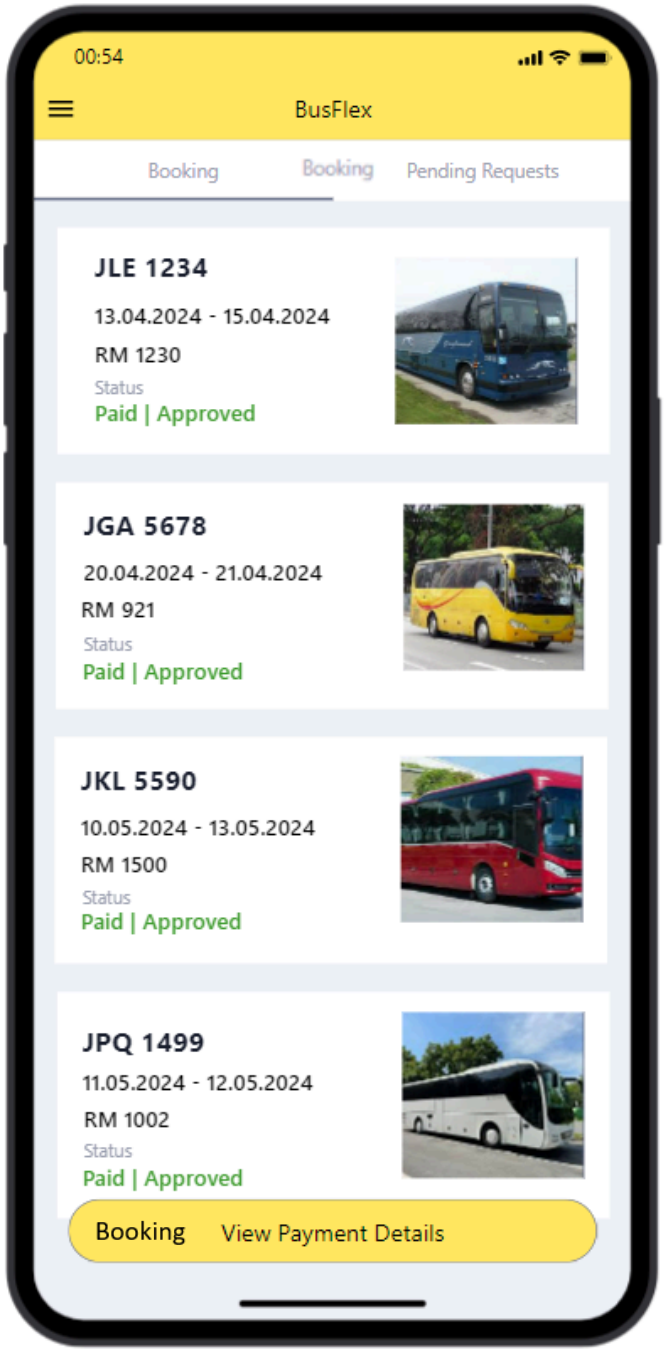
7.22 Staff View Pending Booking



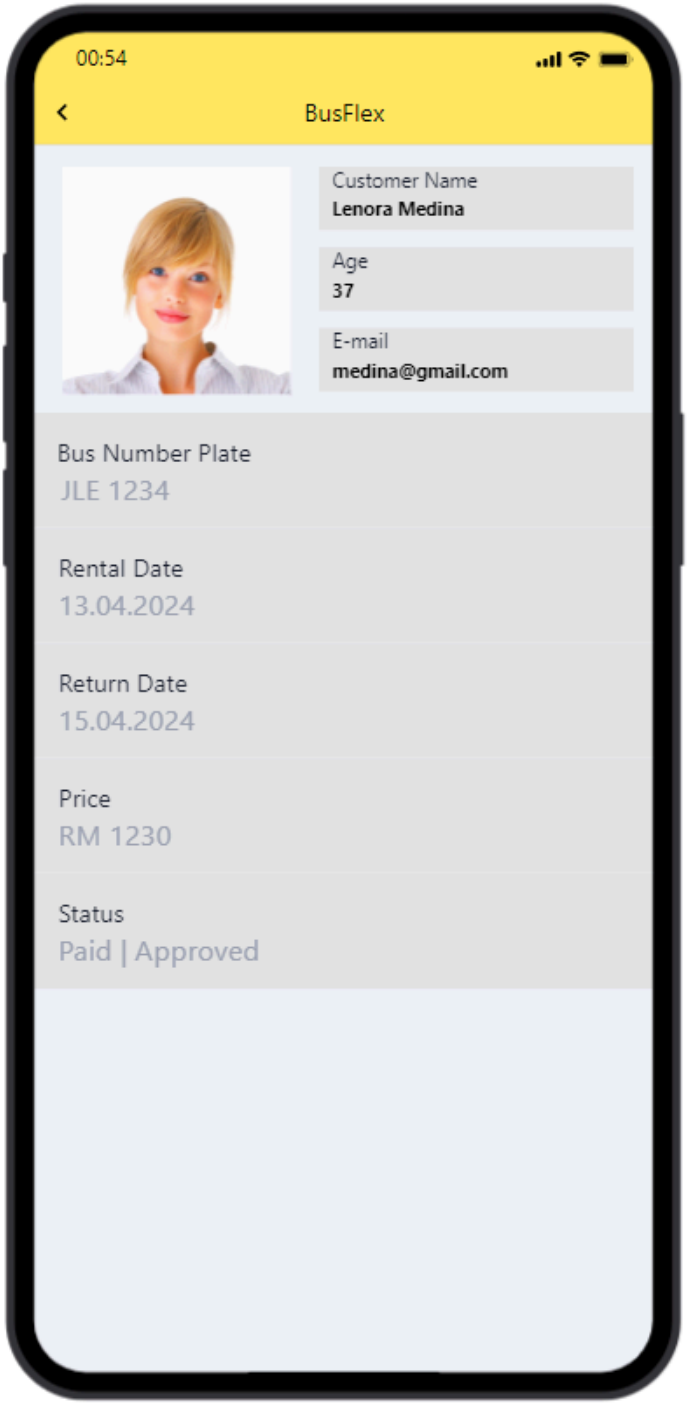
7.23 Staff Approve Pending Booking



7.24 Staff View Approved Booking Page



7.25 Staff View Approved Booking Details Page



8.0 Summary

During this phase, we carefully transformed our Conceptual Entity Relationship Diagram (ERD) into a Logical ERD, tailoring it specifically to meet the requirements of an efficient bus booking system. This process involved removing non-relational features and refining relationships to ensure that our database strictly adhered to relational principles.

We also finalized the relational schema by converting each entity, such as passenger, booking, bus, driver, and payment, into tables with well-defined attributes and primary keys. To meet the Boyce-Codd Normal Form (BCNF), we normalized the database to eliminate redundancies and dependencies, ensuring the integrity of passenger details, booking information, and associated transactions.

The creation of the final Logical ERD serves as a visual representation of our refined database structure, showcasing the relationships between entities and offering a clear roadmap for efficient data retrieval and management. Additionally, the data dictionary was updated to reflect the normalized structure, ensuring consistency and accuracy in the stored information.

In conclusion, this phase taught us how to translate conceptual ideas into well-organized and normalized database structures. This forms the foundation for an efficient and reliable bus booking system, streamlining the booking process, reducing errors, and improving the overall experience for passengers.