



UNIVERSITY
OF WOLLONGONG
AUSTRALIA

JULY 2020

CSIT 314 Software Development Methodology
Shuttle Bus Central System

Group Unique

Group Members	UOW ID
Derrick Lim Jenn Yuen	6306615
Andreas Andi	6307073
Lum Zhi Yan	6340659
Zhang Yu Tong	6265728

Table of Contents

1.0 Project Overview	5
2.0 Group Structure, Roles and Responsibilities	6
3.0 Project Plan and Iteration	8
3.1 Gantt Chart	8
4.0 Tools and Techniques	9
5.0 System Architecture	12
5.1 Conceptual Diagram	12
6.0 Overall Description	13
6.1 Product Overview	13
6.2 User Characteristics	13
6.3 Constraint, Assumptions and Dependencies	14
7.0 User Stories & Taiga.io.....	15
7.1 Scrum Roles	15
7.2 User Stories & Sprint Tasks	15
7.2.1 Week 3 (9 Aug – 16 Aug).....	15
7.2.2 Week 4 (16 Aug – 23 Aug).....	16
7.2.3 Week 5 (23 Aug – 30 Aug).....	16
7.2.4 Week 6 (30 Aug – 6 Sep)	17
7.2.5 Week 7 (6 Sep – 13 Sep)	17
7.2.6 Week 8 (20 Sep – 27 Sep)	18
7.2.7 Week 9 (20 Sep – 27 Sep)	18
7.2.8 Week 10 (27 Sep – 4 Oct).....	19
7.2.9 Week 11 (4 Oct – 11 Oct).....	19
7.2.10 Week 12 (11 Oct – 18 Oct)	20
7.2.11 Week 13 (18 Oct – 25 Oct)	21
7.2.12 Week 14 (26 Oct – 2 Nov)	22
7.2.13 Week 15 (2 Nov – 8 Nov).....	23
8.0 Design and Implementation	24
8.1 Class Diagram.....	24
8.2 Use Case	25
8.2.1 Use Case Diagram	25
8.2.2 Use Case Description.....	27
8.3 Sequence Diagram.....	49
8.4 Data Persistence Diagram	59

8.4.1 Data Dictionary	60
9.0 User interface design and system manual	63
9.1 Windows System.....	63
9.1.1 System Menu and Navigation	63
9.1.2 User Authentication & User Level (Access Level).....	68
9.1.3 Create a Booking process	70
9.1.4 Register External Bus Owner	77
9.1.5 Create Driver	80
9.1.6 Create Bus.....	82
9.1.7 Account Management	85
9.1.8 View/Edit Customer Booking	88
9.1.9 View/Edit External Bus Owner	94
9.1.10 View/Edit Driver	97
9.1.11 View/Edit Bus.....	99
9.1.12 Mileage Record	102
9.1.13 Maintenance Record	105
9.2 Mobile Web Application	108
9.2.1 Login Page	108
9.2.2 Home Page	109
9.2.3 Driver Profile Page	111
9.2.4 Booking Request	116
9.2.5 Changed/Cancelled Booking.....	118
10.0 Unit Testing.....	119
10.1 Test Plan.....	119
10.2 Unit Testing Overview on Project	119
10.3 Test Data for each Unit Testing	120
10.3.1 Test Case 1	120
10.3.2 Test Case 2	121
10.3.3 Test Case 3	122
10.4 .Net Unit Test and Test Code.....	123
10.5 Unit Testing Results and Conclusions.....	124
11.0 Conclusion	125
12.0 Appendix	126
12.1 Meeting Minutes (Week 3 to Week 15)	126
12.1.1 Minutes of Meeting (Week 3).....	126

12.1.2 Minutes of Meeting (Week 4).....	128
12.1.3 Minutes of Meeting (Week 5).....	130
12.1.4 Minutes of Meeting (Week 6).....	132
12.1.5 Minutes of Meeting (Week 7).....	134
12.1.6 Minutes of Meeting (Week 8).....	136
12.1.7 Minutes of Meeting (Week 9).....	138
12.1.8 Minutes of Meeting (Week 10).....	140
12.1.9 Minutes of Meeting (Week 11).....	142
12.1.10 Minutes of Meeting (Week 12)	144
12.1.11 Minutes of Meeting (Week 13)	146
12.1.12 Minutes of Meeting (Week 14)	148
12.1.13 Minutes of Meeting (Week 15)	150
12.2 System Requirements and Installation	152
12.2.1 System Requirements	152
12.2.2 Installation	152
12.3 WhatsApp Screenshot	155
12.3 Google Docs Screenshots.....	158
12.4 Group Members Contribution Form	159

1.0 Project Overview

The main objective of this project is to develop a computerized booking system for Shuttle Bus Central (SBC) Sdn. Bhd. SBC was established in 2017 and its business is to provide bus/van service to the customers. In this project, our team members will create a new booking system to replace the paper documents and manual tracking of customer bookings which SBC has been doing all over the years. This system is designed with various functions capabilities that would allow SBC to manage the customers' bookings more effectively.

The development of the system is divided into two parts, a client Windows-based application and a mobile web application. SBC admin staff will use the client Windows-based application to keep track of customers booking requests, bus/van and driver details as well as external bus owners' information. On the other hand, external bus owners may handle customer bookings, check changed or cancelled requests and update drivers' profiles through the mobile web application.

To ensure that the project's objectives are achievable, a Gantt chart to show the overall project schedule, a UML class diagram with attributes to display the relationship among objects, a use case diagram with descriptions to summarize the interaction between users and system are all included in this report. Besides, the team members are also required to utilize the project management tool Taiga.io for coordination and communication. The whole project development is expected to be completed within 13 weeks.

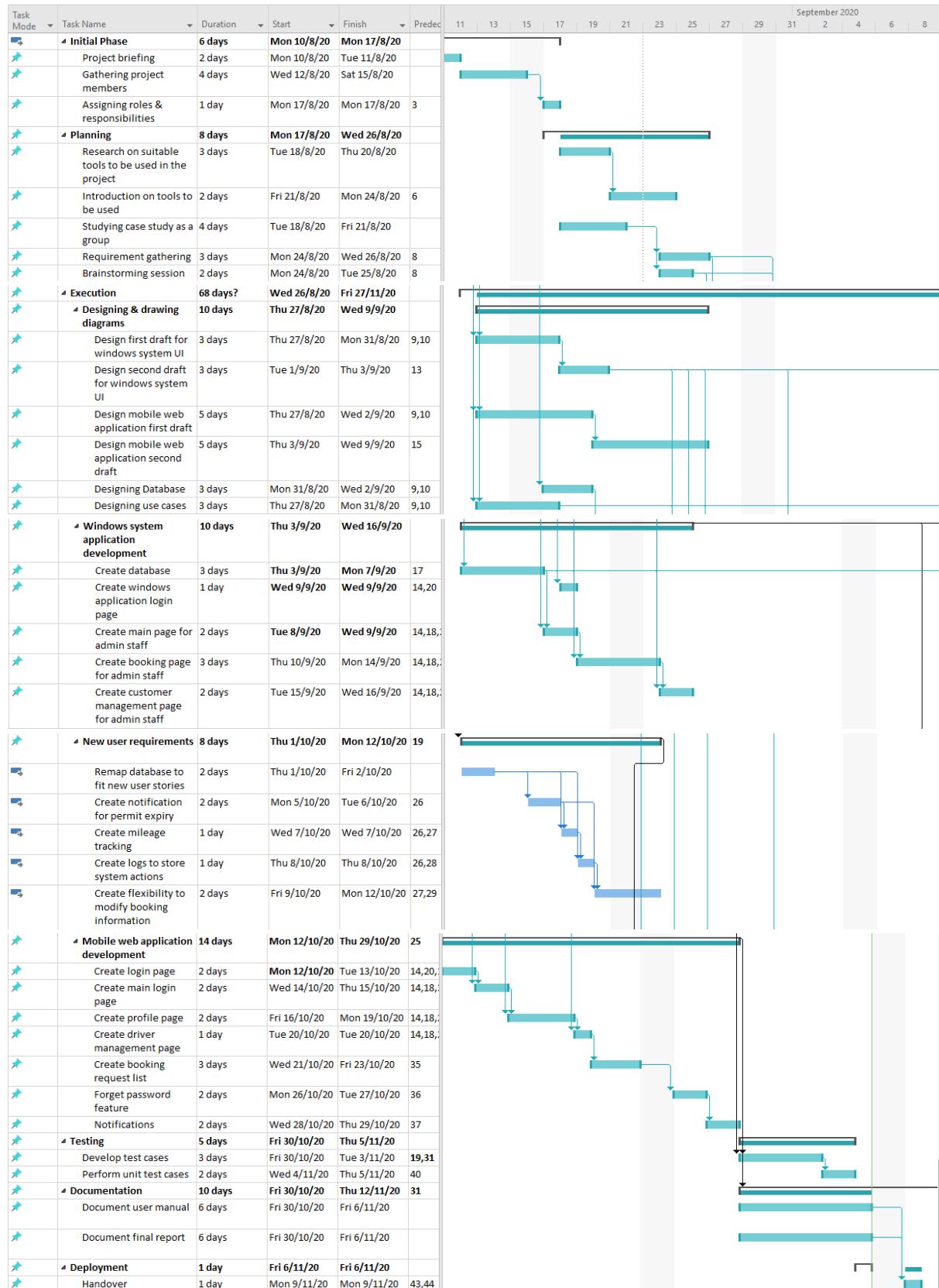
2.0 Group Structure, Roles and Responsibilities

Member	Roles	Responsibilities
Derrick Lim Jenn Yuen	<ul style="list-style-type: none"> o Project leader o Front-end programmer o Back-end programmer o Documenter 	<ul style="list-style-type: none"> · In charge of monitoring and controlling the project's progression · In charge of tasks delegation among team members · In charge of both front-end and back-end programming · In charge of creating the MySQL database · In charge of project documentation
Andreas Andi	<ul style="list-style-type: none"> o Programmer o Documenter 	<ul style="list-style-type: none"> · In charge of creating mobile web application · In charge of conducting unit testing · In charge of project documentation

Lum Zhi Yan	<ul style="list-style-type: none"> o Programmer o UI/UX Designer o Documenter 	<ul style="list-style-type: none"> · In charge of creating mobile web application · In charge of designing UI/UX of mobile web application · In charge of conducting unit testing · In charge of project documentation · In charge of the review and compilation of the final report
Zhang Yu Tong	<ul style="list-style-type: none"> o Programmer o UI/UX Designer o Documenter 	<ul style="list-style-type: none"> · In charge of creating Windows application · In charge of designing UI/UX of Windows application · In charge of conducting unit testing · In charge of project documentation

3.0 Project Plan and Iteration

3.1 Gantt Chart



4.0 Tools and Techniques

	Programming language	Description
1.	C# (.NET framework)	Windows Application Form .Net is a framework used to create Windows application and the framework is based on C# programming language
2.	PHP	PHP is a server scripting language for making interactive web development. PHP is selected by the team members to create the SBC Mobile Web Application due to its conformity with the requirements, and the compatibility with the required device.
3.	MySQL	MySQL is a relational database that uses Structured Query Language (SQL) to query a database. Team members use SQL to store, access, update, and manipulate the database.
4.	CSS	CSS is a designing and visual styling language used to style the HTML part of SBC Mobile Web Application. CSS is applied in the Mobile Web Application to enhance the product's user interface visualization and user experience.
5.	HTML	HTML is the standard markup language used to create structure and contents of the web page for Mobile Web Application.
6.	Javascript	Javascript is a web programming language that is able to update and manipulate CSS and HTML elements. It enables webpages to do more complex and sophisticated features.

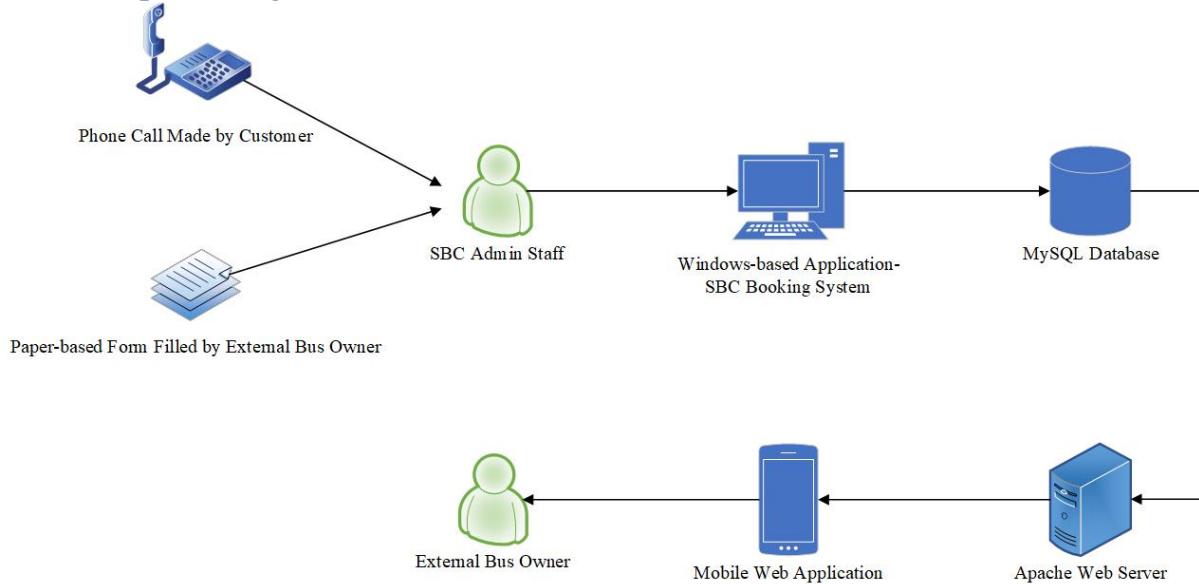
	Software used For Programming	Description
1.	Visual Studio 2019 .NET)	Visual Studio 2019 software is used in developing the Windows Application System for SBC. By using the .NET(C#) as the package solution, team members develop the Windows Application System using C# as the primary programming language
2.	Sublime Text	Sublime Text is a text code editor used in developing the Mobile Web Application for SBC. By utilizing PHP as the primary programming language team members create the Mobile Web Application for SBC.
3.	XAMPP	XAMPP is a stack package consisting of Apache, MariaDB, Perl and PHP. It is a free web server hosted on local machine to simulate the real webserver environment.

	Sites & Software used for Documentation	Description
1.	Google Drive	Google Drive allows members to store files and synchronize their work. Every member in this project can share and edit any files regarding the product creation or the documentation.
2.	Microsoft Word	Microsoft Word mainly used for the documentation and report generation.
3.	Microsoft Visio	Microsoft Visio is a diagramming and vector graphics application for drawing a variety of diagrams. Use case diagram, sequence diagram and architecture diagram in the project are drawn using Microsoft Visio.
4.	UMLet	UMLet software is used for creating the class diagram to define the types of objects in the system and the different types of relationships that exist among them.

5.	Taiga.io	Taiga.io is used to manage and track project progression. Every member role and task are defined within Taiga.io. All the members' tasks are updated on Taiga.io every week.
----	----------	--

5.0 System Architecture

5.1 Conceptual Diagram



The conceptual diagram above is showing an overview on the structure and behavior of SBC Booking System. The Window-based application is developed based on Windows Form .Net framework using C# programming language, whereas the mobile web application is developed with PHP, HTML, CSS and JavaScript. All data and information are stored in MySQL database which will be shared among the Windows application and the Apache Web Server.

Starting from a phone call made by a customer, the admin staff will register the customer details into the system and check the availability of the bus through a Windows-based application. SBC own internal buses will be first priority for any incoming booking that fulfils the request. If SBC is unable to fulfil the customer request, the request will be released to external providers where they can provide their service to customers through SBC with a small service charge.

The admin staff will also collect the paper-based form filled by the external bus owner and enter all the information into the system using the Window-based application. Admin staff may also use the Windows application to create, read and update any information in the database. On the other hand, the external bus owner may interact with the system using the mobile web application. The mobile web application is capable of confirming customer booking requests and assigning drivers and buses to that specific booking request. Other functionalities such as adding new drivers into to system can also be done by the external provider using the mobile web application.

6.0 Overall Description

6.1 Product Overview

SBC current rental operations are all handled manually using paper and form filling, which makes the process of booking complicated and time consuming. Thus, it is difficult for SBC to keep track of current internal data and registered external providers. SBC needs a faster and efficient way to streamline these business processes and to get confirmation from an external provider. Additionally, SBC is also looking for a way to manage company data such as customer information, business data and also external providers personal data. With the new Windows based SBC booking system in place, it is designed to solve the issues that SBC currently face.

6.2 User Characteristics

Admin Staff	This user interacts most with the system and runs the most of SBC rental operations. This user creates/manages customer booking and information, creates/manages external bus owner information and accounts, and creates/manages the bus driver/vehicle information.
Customer	This user request for booking through phone call and provides booking information.
External Bus Owner	This user can accept or decline the customer booking request for External provider. This user is also able to create/manage external bus driver and vehicle information within the system.
Internal / External Bus Driver	This user will be informed and assigned by the Internal / External owner regarding the customer booking request.

6.3 Constraint, Assumptions and Dependencies

Constraint

- Customer number must be in the database before create booking
- Admin need to manually update the booking status
- Only the Admin master able to manage admin account & view logs
- Only the Admin able to insert bus mileage and maintenance date
- Admin needs to create an internal driver & bus to assign customer booking to internal provider
- Admin need to fill in all the information requested on the system to create/update any records.
- Admin staff cannot assign the external bus driver to the customer booking

Assumption

- Assume all users are English literate
- Assume all users have IT knowledge
- Assume all users are not mentally disabled
- Assume all data entered into the system is accurate

Dependencies

- The booking can only be accepted if there's an available bus and driver
- Only Admin able to create and view internal driver/bus
- Only Admin able to register and view External bus owner account
- Mobile Web Application receive booking request after Admin create the booking in the Windows System and outsource to external provider.

7.0 User Stories & Taiga.io

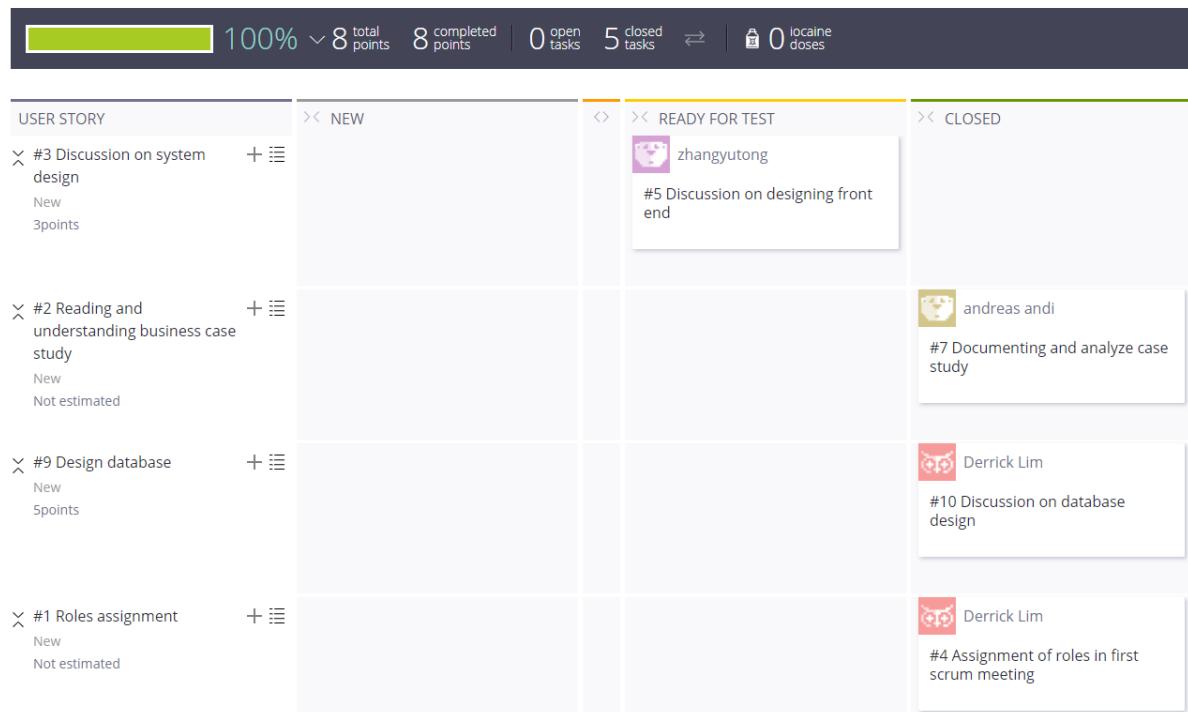
7.1 Scrum Roles

Name	Role
Mr. Faizal Alias	Product Owner
Derrick Lim Jenn Yuen	Scrum Master
Andreas Andi	Developer (Back-End)
Lum Zhi Yan	Developer (Front-End)
Zhang Yu Tong	Designer

7.2 User Stories & Sprint Tasks

7.2.1 Week 3 (9 Aug – 16 Aug)

WEEK3 CSIT314-JUL2020-U... 09 AUG 2020-16 AUG 2020



#3 Discussion on System Design

#5 Discussion on designing front end

#2 Reading and understanding business case study

#7 Documenting and analyze case study

#9 Design database

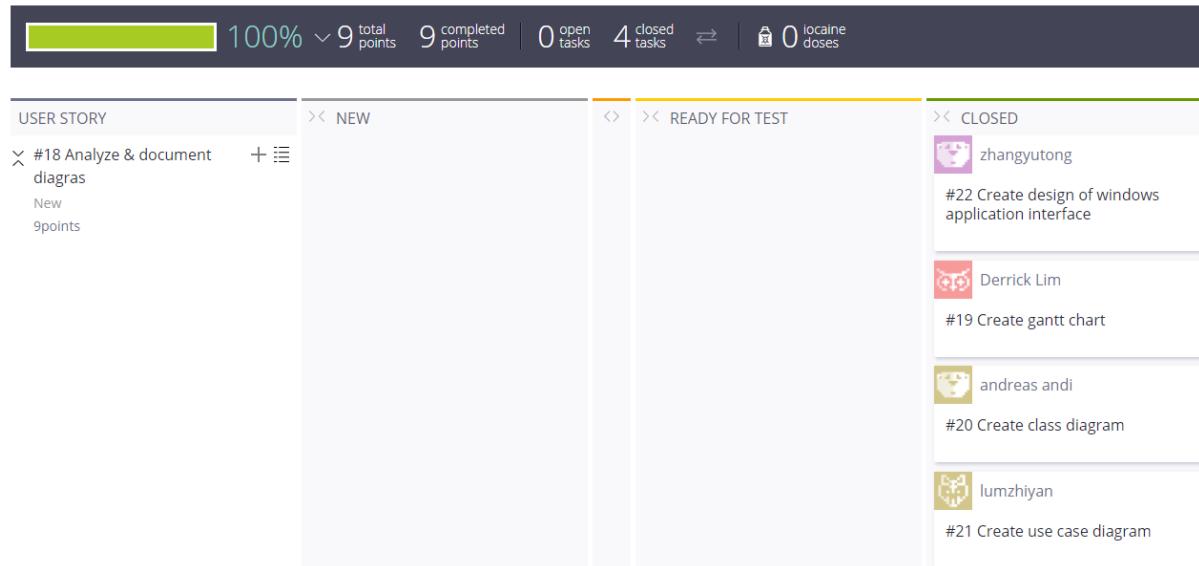
#10 Discussion on database design

#1 Roles Assignment

#4 Assignment of roles in first scrum meeting

7.2.2 Week 4 (16 Aug – 23 Aug)

WEEK4 CSIT314-JUL2020-U... 16 AUG 2020-23 AUG 2020



#18 Analyze & document diagrams

#22 Create design of windows application interface

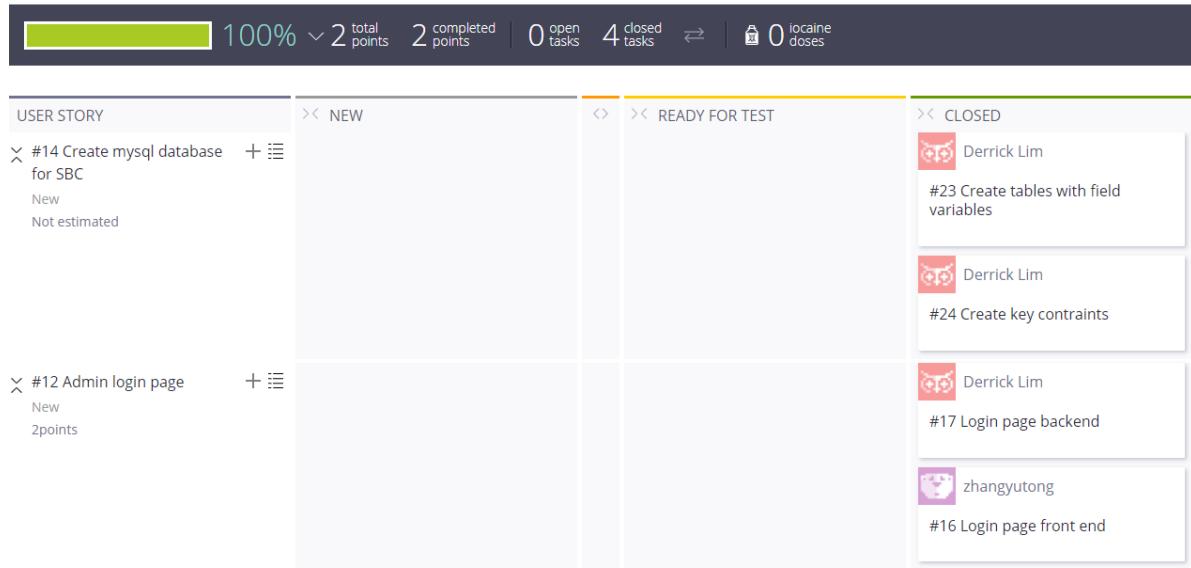
#19 Create gantt chart

#20 Create class diagram

#21 Create use case diagram

7.2.3 Week 5 (23 Aug – 30 Aug)

WEEK5 CSIT314-JUL2020-U... 23 AUG 2020-30 AUG 2020



#14 Create mysql database for SBC

#23 create tables with field variables

#24 Create key constraints

#12 Admin login page

#17 Login page backend

#16 Login page front end

7.2.4 Week 6 (30 Aug – 6 Sep)

WEEK6 CSIT314-JUL2020-U... 30 AUG 2020-06 SEP 2020

USER STORY				NEW	READY FOR TEST	CLOSED
✗ #25 Bus drivers to be able to view schedule	+ ⋮					 andreas andi #27 Check for authentication process
New Not estimated						 lumzhiyan #26 Mobile web login front end
✗ #28 Create customer bookings tab	+ ⋮					 Derrick Lim #31 Create customer booking frontend page
New 6points						

#25 Bus driver to be able to view schedule

#27 check for authentication process

#Mobile web login front end

#28 Create customer bookings tab

#31 Create customer booking frontend page

7.2.5 Week 7 (6 Sep – 13 Sep)

WEEK7 CSIT314-JUL2020-U... 06 SEP 2020-13 SEP 2020

USER STORY				NEW	READY FOR TEST	CLOSED
✗ Storyless tasks	+ ⋮					 Derrick Lim #52 Admin to enter customer telephone when booking

Storyless task

#52 Admin to enter customer telephone when booking

#51 Admin should be able to register customer

7.2.6 Week 8 (20 Sep – 27 Sep)

WEEK8 CSIT314-JUL2020-U... 13 SEP 2020-20 SEP 2020

The screenshot shows a project management board with four columns: USER STORY, NEW, READY FOR TEST, and CLOSED. In the NEW column, there is one task: #53 Allow admin to create bus owner. In the READY FOR TEST column, there are two tasks: #54 Create the form and #55 Insert data from front end to database bus owner table, both assigned to Derrick Lim. In the CLOSED column, there are no tasks.

USER STORY	NEW	READY FOR TEST	CLOSED
#53 Allow admin to create bus owner New Not estimated		#54 Create the form #55 Insert data from front end to database bus owner table	
X Storyless tasks + ⋮			

#53 Allow admin to create bus owner

#54 Create the form

#55 Insert data from front end to database bus owner table

7.2.7 Week 9 (20 Sep – 27 Sep)

WEEK 9 CSIT314-JUL2020-U... 20 SEP 2020-27 SEP 2020

The screenshot shows a project management board with four columns: USER STORY, NEW, READY FOR TEST, and CLOSED. In the NEW column, there are two tasks: #56 Admin to create and record bus information and #57 Admin to create and record driver information. In the READY FOR TEST column, there are four tasks: #59 Write bus information into database, #58 Create bus form, #61 Write driver information into SQL database, and #60 Create driver form page, all assigned to Derrick Lim. In the CLOSED column, there are no tasks.

USER STORY	NEW	READY FOR TEST	CLOSED
#56 Admin to create and record bus information New Not estimated		#59 Write bus information into database #58 Create bus form	
#57 Admin to create and record driver information New Not estimated		#61 Write driver information into sql database #60 Create driver form page	
X Storyless tasks + ⋮			

#56 Admin to create and record bus information

#59 Write bus information into database

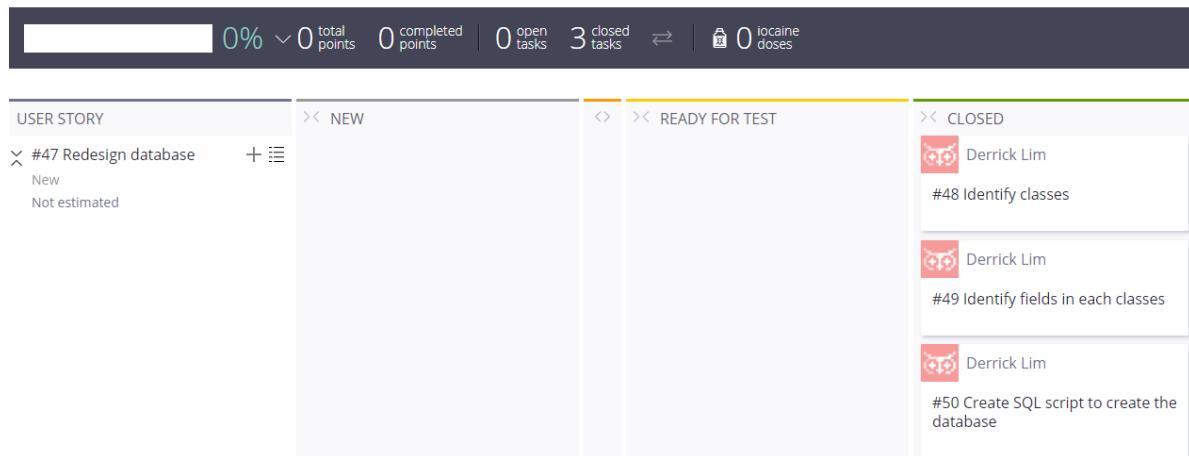
#58 Create bus form

#61 Write driver information into SQL database

#60 Create driver form page

7.2.8 Week 10 (27 Sep – 4 Oct)

WEEK10 CSIT314-JUL2020-U... 27 SEP 2020-04 OCT 2020



#47 Redesign database

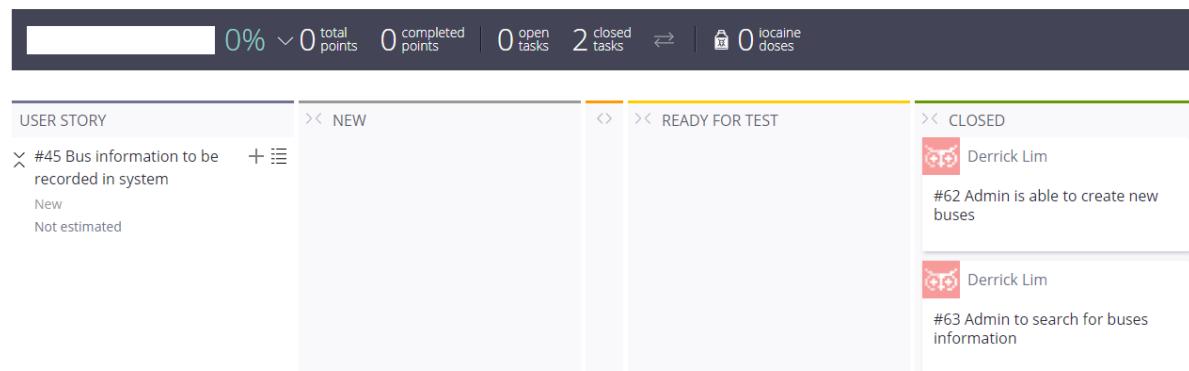
#48 identify classes

#49 identify fields in each class

#50 Create SQL script to create the database

7.2.9 Week 11 (4 Oct – 11 Oct)

WEEK11 CSIT314-JUL2020-U... 04 OCT 2020-11 OCT 2020



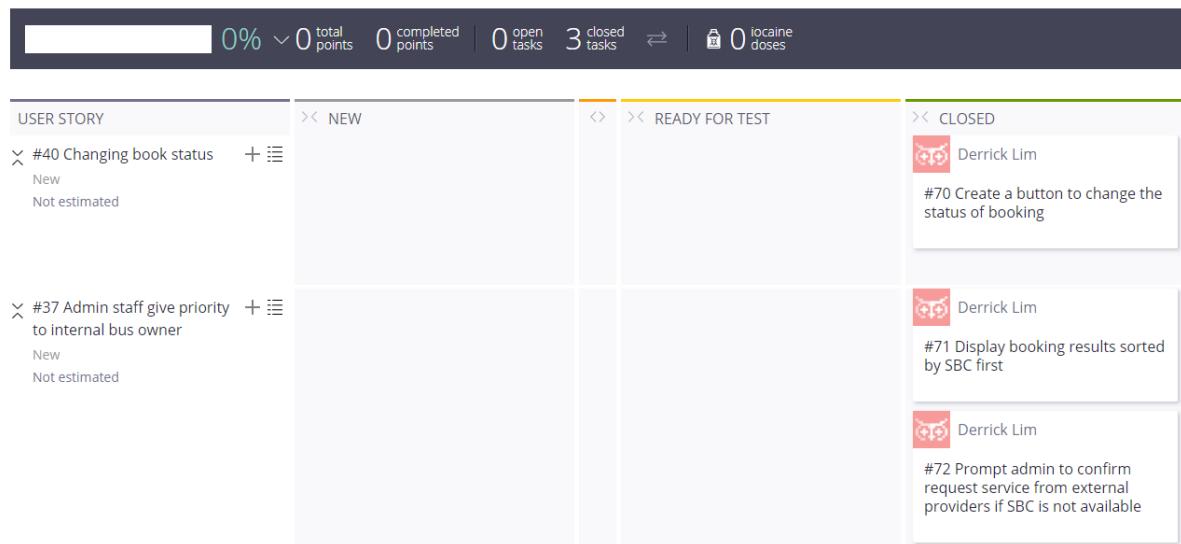
#45 Bus information to be recorded in system

#62 Admin is able to create new buses

#63 Admin to search for buses information

7.2.10 Week 12 (11 Oct – 18 Oct)

WEEK12 CSIT314-JUL2020-U... 11 OCT 2020-18 OCT 2020



#40 Changing book status

#70 Create a button to change the status of booking

#37 Admin staff give priority to internal bus owner

#71 Display booking results sorted by SBC first

#72 Prompt admin to confirm request service from external providers if SBC is not available

7.2.11 Week 13 (18 Oct – 25 Oct)

WEEK13 CSIT314-JUL2020-U... 18 OCT 2020-25 OCT 2020

USER STORY		NEW	READY FOR TEST	CLOSED
✗ #38 Admin staff check the availability of shuttle bus by the date request	+ ⋮			 Derrick Lim #69 Available buses to be shown according to date
✗ #42 External bus driver to choose customer by dates	+ ⋮			 Derrick Lim #68 Available bookings to be shown on booking page sorted
✗ #46 Customer booking should contain information	+ ⋮			 Derrick Lim #67 New page to enter customer booking information such as name, bus type, capacity, departure and arrival location, etc
				 Derrick Lim #66 Check for all available buses on the date, and display the buses sorted by SBC on top
				 Derrick Lim #65 Check if the customer has a previous record

#38 Admin staff check the availability of the shuttle bus by the date request

#69 Available busses to be shown according to date

#42 External bus driver to choose customer by dates

#68 Available booking to be shown on booking page sorted

#46 Customer booking should contain information

#67 New page to enter customer booking information such as name, bus type, capacity, departure and arrival location etc.

#66 Check for all available buses on the date, and display the buses sorted by SBC on the top

#65 Check if the customer has a previous record

7.2.12 Week 14 (26 Oct – 2 Nov)

WEEK14 CSIT314-JUL2020-U... 26 OCT 2020-02 NOV 2020

USER STORY		NEW	READY FOR TEST	CLOSED
✗ #39 Booking status should be "Book but not paid" after confirmation	+ ⏹			Derrick Lim #80 Booking status should be "Book but not paid" after creating booking
New				Derrick Lim #81 Admin to modify/update booking status manually
Not estimated				Derrick Lim #79 Allow master admin to modify admin passwords
✗ #77 Admin account management	+ ⏹			Derrick Lim #82 Create alert for admin for permit expiry
New				Derrick Lim #73 Write logs to database table for each transaction
Not estimated				Derrick Lim #78 Create a window to view transaction logs
✗ #43 Keep track of transaction with logs	+ ⏹			Derrick Lim #74 Display request bookings on external booking web
New				Derrick Lim #75 Providers can choose to accept bookings
Not estimated				
✗ #36 Shuttle bus owner can check shuttle service request via mobile web app	+ ⏹			
New				
Not estimated				

#39 Booking status should be “Book but not paid after confirmation”

#80 Booking status should be “booked but not paid” after creating booking

#81 Admin to modify/update booking status manually

#77 Admin account management

#79 Allow master admin to modify admin passwords

#76 Notifications upon login

#82 Create alert for admin for permit expiry

#43 Keep track of transaction with logs

#73 Write logs to database table for each transaction

#78 Create a window to view transaction logs

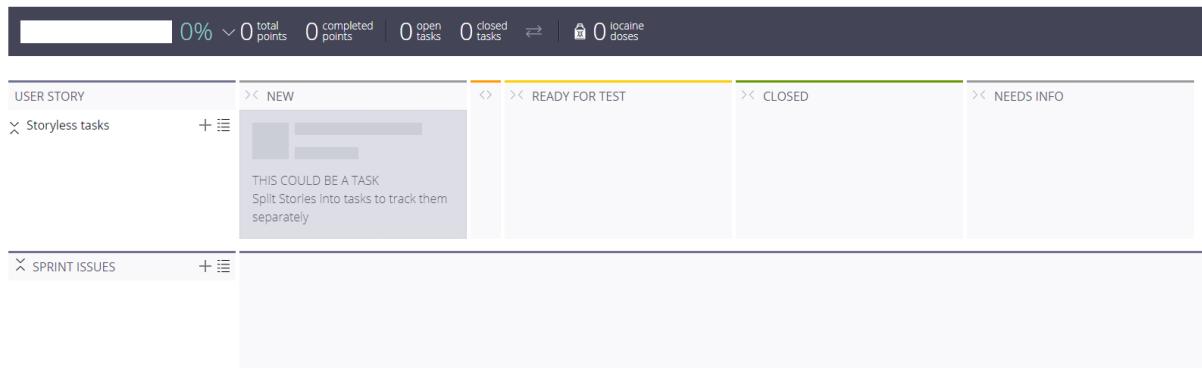
#36 Shuttle bus owner can check shuttle service request via mobile web app

#74 Display request bookings on external booking web

#75 Providers can choose to accept bookings

7.2.13 Week 15 (2 Nov – 8 Nov)

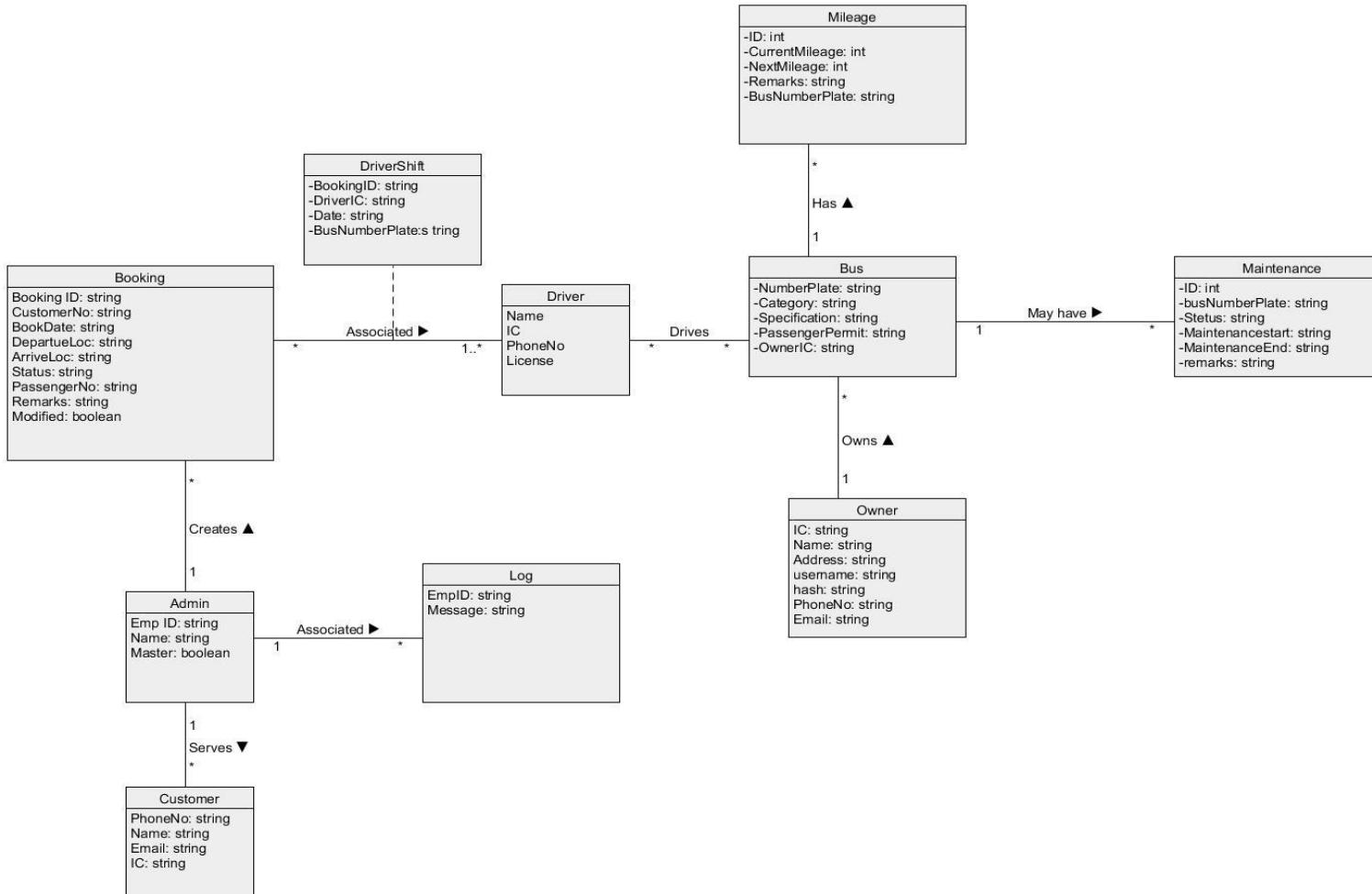
WEEK15 CSIT314-JUL2020-U... 02 NOV 2020-08 NOV 2020



Empty entries

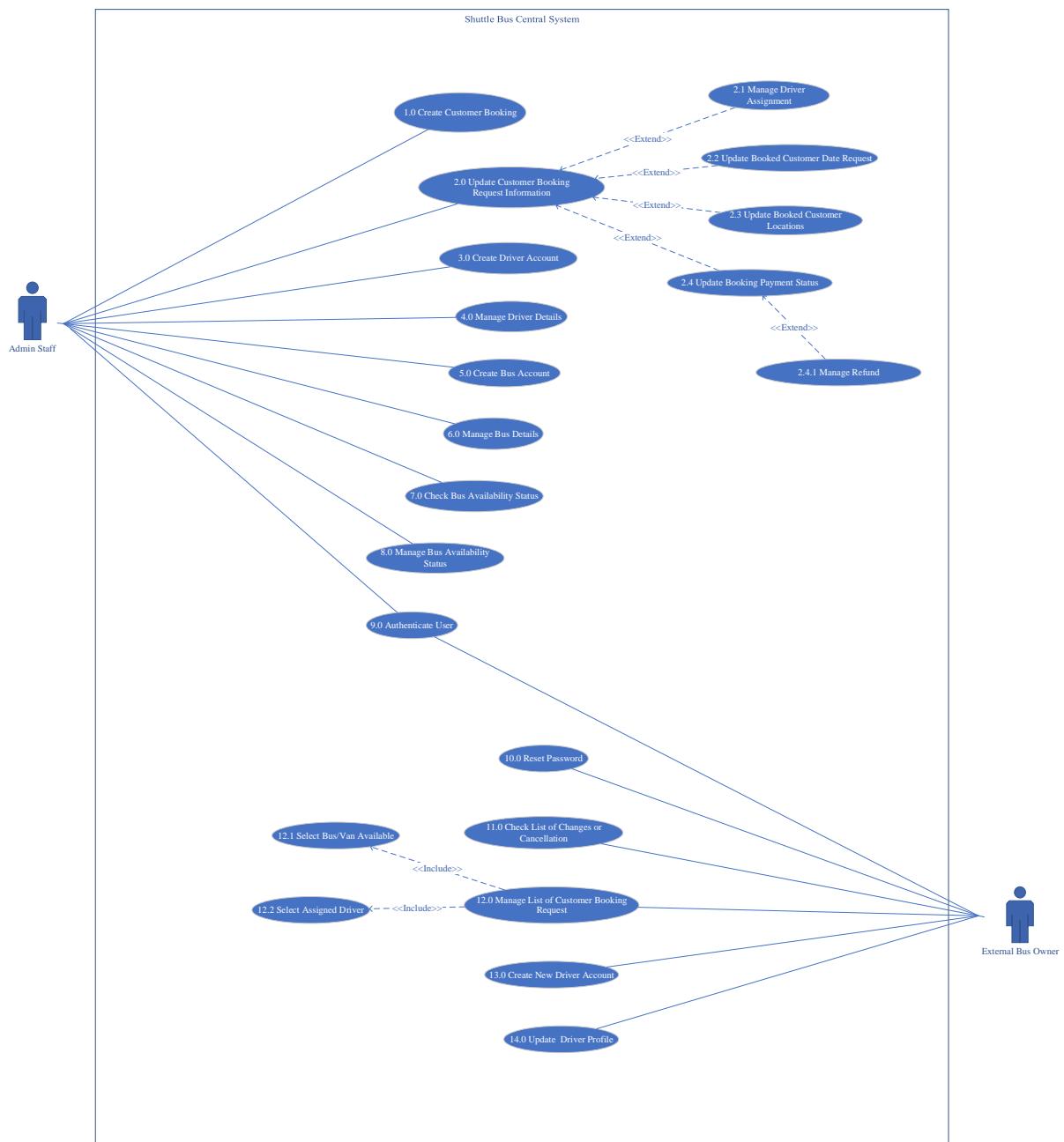
8.0 Design and Implementation

8.1 Class Diagram



8.2 Use Case

8.2.1 Use Case Diagram



The table below shows a summary of the use cases listed in the diagram above.

Actors	Use Cases
Admin Staff	1.0 Create Customer Booking 2.0 Update Customer Booking Request Information 2.1 Manage Driver Assignment 2.2 Update Booked Customer Date Request 2.3 Update Booked Customer Locations 2.4 Update Booking Payment Status 2.4.1 Manage Refund 3.0 Create Driver Account 4.0 Manage Driver Details 5.0 Create Bus Account 6.0 Manage Bus Details 7.0 Check Bus Availability Status 8.0 Manage Bus Availability Status 9.0 Authenticate User
External Bus Owner	10.0 Reset Password 11.0 Check List of Changes or Cancellation 12.0 Manage List of Customer Booking Request 12.1 Select Bus/Van Available 12.2 Select Assigned Driver 13.0 Create New Driver Account 14.0 Update Driver Profile

8.2.2 Use Case Description

Name of Use Case:	Create Customer Booking
Use Case No.:	1.0
Created By:	Lum Zhi Yan
Date Created:	17/10/20
Description:	The admin staff will create a booking in the system upon receiving booking requests and details from the customer through phone calls.
Actors:	Admin staff
Related Use Cases:	None
Preconditions:	1. Customer wants to make a booking.
Postconditions:	1. The details of customer booking requests are updated to the database.
Trigger:	Select create booking through the dropdown list of the manage customer booking.
Flow:	<ol style="list-style-type: none"> 1. Admin staff enters customer phone number into the system. 2. System displays create customer booking page. 3. Admin staff enters customer booking date into the system. 4. System displays available bus according to the date entered. 5. Admin staff select the bus and click confirm. 6. System displays form to collect booking information. 7. Admin staff enters booking information such as departure, arrival location, passenger number, bus category, etc and click confirm. 8. Booking is created and information will be stored inside database.
Exceptions:	1. No bus/van is available on the date requested by the customer.

Name of Use Case:	Update Customer Booking Request Information
Use Case No.:	2.0
Created By:	Zhang YuTong
Date Created:	25/10/20
Description:	Changing the booking information such as date or destination
Actors:	Admin staff
Related Use Cases:	2.1 Manage Driver Assignment 2.2 Update Booked Customer Date Request 2.3 Update Booked Customer Locations 2.4 Update Booking Payment Status 2.4.1 Manage Refund
Preconditions:	Booking information must be created and exist
Postconditions:	System will update the database for any updated booking information.
Trigger:	Clicking “View/Edit Booking Request” under the Customer Booking menu.
Flow:	1. Admin staff click on “View/Edit Booking Request” under Customer Booking menu. 2. System returns a new page to search for bookings based on a few fields such as customer name, booking ID, status or date. 3. Admin staff enter the information and click search. 4. System returns with the booking information. 5. Admin staff click on the booking and click the button “Edit”. 6. System return with a new window showing the booking information. 7. Admin staff can edit and change the fields in the new pop up window and click “Submit” when done. 8. System update the database for any modified fields. 9. System returns with a message showing successfully updated.
Exceptions:	1. Booking does not exist.

Name of Use Case:	Manage Driver Assignment
Use Case No.:	2.1
Created By:	Zhang YuTong
Date Created:	25/10/20
Description:	The admin staff assign drivers to booking request.
Actors:	Admin staff
Related Use Cases:	2.0 Update Customer Booking Request Information
Preconditions:	Customer wants to create a new booking request.
Postconditions:	Drivers assign will be updated in the database.
Trigger:	Assigning a driver while creating a new booking request.
Flow:	<p>1. Admin staff click on “Customer Booking” menu. 2. Admin staff clicks on “Create Booking” 3. System prompts for customer phone number. 4. Admin staff enters customer phone number. 5. System prompts for date of booking request 6. Admin staff enters the date for customer booking request and click search. 7. System returns with a list of available buses that can fulfil the request. 8. Admin staff click on the desired bus and click confirm 9. System returns with a form for admin staff to fill in the booking information 10. Admin staff fill in the booking information and click on the driver drop down list to assign driver to the booking request. 11. Admin staff click submit after all information is entered. 12. System collects information from the form page and update the database. 13. System returns with message showing booking saved.</p>
Exceptions:	<ol style="list-style-type: none"> 1. Booking information insufficient. 2. No buses are available for service. 3. No drivers are available for service.

Name of Use Case:	Update Booked Customer Date Request
Use Case No.:	2.2
Created By:	Zhang YuTong
Date Created:	25/10/20
Description:	Changing the initial date for booking request to a different date.
Actors:	Admin staff
Related Use Cases:	2.0 Update Customer Booking Request Information
Preconditions:	Booking must already be made.
Postconditions:	The date of the booking will be changed and updated in the database
Trigger:	Clicking on “Edit” button after searching for the booking record under “Customer Booking”>” View/Edit Booking”
Flow:	<ol style="list-style-type: none"> 1. Admin staff click on “View/Edit Booking Request” under Customer Booking menu. 2. System returns a new page to search for bookings based on a few fields such as customer name, booking ID, status or date. 3. Admin staff enter the information and click search. 4. System returns with the booking information. 5. Admin staff click on the booking and click the button “Edit”. 6. System return with a new window showing the booking information. 7. Admin staff can change the booking date by clicking on the calendar under “Booking Date”. 8. Admin staff click on “Submit” button. 9. System update the database for any modified fields. 10. System returns with a message showing successfully updated.
Exceptions:	Booking record does not exist.

Name of Use Case:	Update Booked Customer Locations
Use Case No.:	2.3
Created By:	ZhangYuTong
Date Created:	25/10/20
Description:	Admin staff to update the booking request customer location.
Actors:	Admin staff
Related Use Cases:	2.0 Update Customer Booking Request Information
Preconditions:	An existing booking record must be available.
Postconditions:	The booking location, departure and arrival location will be updated in the database.
Trigger:	Clicking on “Edit” button after searching for the booking record under “Customer Booking”>” View/Edit Booking”.
Flow:	<ol style="list-style-type: none"> 1. Admin staff click on “View/Edit Booking Request” under Customer Booking menu. 2. System returns a new page to search for bookings based on a few fields such as customer name, booking ID, status or date. 3. Admin staff enter the information and click search. 4. System returns with the booking information 5. Admin staff click on the booking and click the button “Edit”. 6. System return with a new window showing the booking information. 7. Admin staff can change the departure and arrival location by entering the new location. 8. Admin staff click on “Submit” button. 9. System update the database for any modified fields. 10. System returns with a message showing successfully updated.
Exceptions:	Booking record does not exist.

Name of Use Case:	Update Booking Payment Status
Use Case No.:	2.4
Created By:	ZhangYuTong
Date Created:	25/10/20
Description:	Admin check the system whether the payment has been made by the customer.
Actors:	Admin staff
Related Use Cases:	2.0 Update Customer Booking Request Information 2.4.1 Manage Refund
Preconditions:	Booking record must be available in the system.
Postconditions:	Booking payment status will be updated in the database.
Trigger:	Clicking on “Edit” button after searching for the booking record under “Customer Booking”>” View/Edit Booking” and changing the booking status to “Booked”.
Flow:	<ol style="list-style-type: none"> 1. Admin staff click on “View/Edit Booking Request” under Customer Booking menu. 2. System returns a new page to search for bookings based on a few fields such as customer name, booking ID, status or date. 3. Admin staff enter the information and click search. 4. System returns with the booking information. 5. Admin staff click on the booking and click the button “Edit”. 6. System return with a new window showing the booking information. 7. Admin staff select the booking status as “Booked” 8. Admin staff click on “Submit” button. 9. System update the database for any modified fields. 10. System returns with a message showing successfully updated.
Exceptions:	Booking record does not exist.

Name of Use Case:	Manage Refund
Use Case No.:	2.4.1
Created By:	ZhangYuTong
Date Created:	25/10/20
Description:	Admin staff enter customer bank account information whenever a refund is initiated.
Actors:	Admin staff
Related Use Cases:	2.0 Update Customer Booking Request Information 2.4 Update Booking Payment Status
Preconditions:	The booking record must be available and booked (paid).
Postconditions:	Customer bank account will be entered into the system and a refund will be initiated.
Trigger:	Clicking on “Edit” button after searching for the booking record under “Customer Booking”>” View/Edit Booking” and changing the booking status to “Cancelled”
Flow:	<ol style="list-style-type: none"> 1. Admin staff click on “View/Edit Booking Request” under Customer Booking menu. 2. System returns a new page to search for bookings based on a few fields such as customer name, booking ID, status or date. 3. Admin staff enter the information and click search. 4. System returns with the booking information. 5. Admin staff click on the booking and click the button “Edit”. 6. System return with a new window showing the booking information. 7. Admin staff select the booking status as “Cancelled” 8. Admin staff click on “Submit” button. 9. System update the database for any modified fields. 10. System will automatically calculate the refund amount based on the booking date. 11. System returns with a message showing successfully updated.
Exceptions:	<ol style="list-style-type: none"> 1. Booking record does not exist. 2. Booking is booked but not paid. 3. Booking is already cancelled.

Name of Use Case:	Create Driver Account
Use Case No.:	3.0
Created By:	Zhang YuTong
Date Created:	25/10/20
Description:	Admin staff creates a new internal or external drivers account attached to the bus providers.
Actors:	Admin staff
Related Use Cases:	None
Preconditions:	Bus owner must be registered with SBC in the system.
Postconditions:	New driver record is added into the system database.
Trigger:	Click on “Create Driver” under the Driver menu bar.
Flow:	<ol style="list-style-type: none"> 1. Admin staff click on “Create Driver” under Driver menu. 2. System displays a new window with empty form. 3. Admin staff fill in the driver information such as driver name, IC, phone number, license number and select the owner attached to. 4. Admin staff click submit once information is entered. 5. System will collect the information entered and update the database. 6. System will return a message showing driver has been added into the system.
Exceptions:	<ol style="list-style-type: none"> 1. The owner of the driver does not exist in the database. 2. Driver information field is empty.

Name of Use Case:	Manage Driver Details
Use Case No.:	4.0
Created By:	ZhangYuTong
Date Created:	25/10/20
Description:	Admin staff adjust and modify driver information.
Actors:	Admin staff
Related Use Cases:	None
Preconditions:	Driver information must already exist in the database.
Postconditions:	Driver information will be updated in the system database.
Trigger:	Click on “View/Edit Driver” under the Driver menu bar.
Flow:	<ol style="list-style-type: none"> 1. Admin staff click on “View/Edit Driver” under Driver menu. 2. System displays a new window to search for driver record using driver’s name or owner’s name. 3. Admin staff to enter the driver’s name and/or select the owner’s name and click “Search” button. 4. Admin staff select the driver record by highlighting the row and click “Edit” button. 5. The system will generate a new window page with the selected driver information shown. 6. Admin staff can select the fields and modify the driver information. 7. Admin staff clicks “Submit” when done. 8. System will extract the information from the form and update the system database. 9. System will return a message box stating driver information has been changed.
Exceptions:	No matching options.

Name of Use Case:	Create Bus Account
Use Case No.:	5.0
Created By:	ZhangYuTong
Date Created:	25/10/20
Description:	Admin staff create a new bus and upload the basic information of this bus.
Actors:	Admin staff
Related Use Cases:	None
Preconditions:	The owner of the bus must be available and registered under SBC.
Postconditions:	The information data of the bus is recorded in the system database.
Trigger:	Click on “Create Bus” under the Bus menu bar.
Flow:	<ol style="list-style-type: none"> 1. Admin staff click on “Create Bus” under Bus menu. 2. System displays a new window to create bus record. 3. Admin staff to enter bus information such as number plate, category, passenger permit, owner name, status, permit validity and bus specifications. 4. Admin staff clicks on “Submit” button when done. 5. System will extract the information from the form and update the system database. 6. System will return a message box stating driver information has been changed.
Exceptions:	Bus owner does not exist.

Name of Use Case:	Manage Bus Details
Use Case No.:	6.0
Created By:	ZhangYuTong
Date Created:	28/10/20
Description:	Admin staff update the bus information stored in the system.
Actors:	Admin staff
Related Use Cases:	None
Preconditions:	Bus record must already exist in the database.
Postconditions:	Bus information will be updated in the system database.
Trigger:	Select “View/Edit Bus” through the dropdown list of the Bus menu.
Flow:	<ol style="list-style-type: none"> 1. Admin staff click on “View/Edit Bus” under Bus menu. 2. System displays a new window to search for Bus record using bus number plate and/or owner’s name. 3. Admin staff to enter the bus number plate and/or select the owner’s name and click “Search” button. 4. Admin staff select the driver record by highlighting the row and click “Edit” button. 5. The system will generate a new window page with the selected bus information shown. 6. Admin staff can select the fields and modify the bus information. 7. Admin staff clicks “Submit” when done. 8. System will extract the information from the form and update the system database. 9. System will return a message box stating bus information has been changed.
Exceptions:	Bus record does not exist.

Name of Use Case:	Check Bus Availability Status
Use Case No.:	7.0
Created By:	ZhangYuTong
Date Created:	29/10/20
Description:	Admin staff view bus availability information on the view/edit bus interface.
Actors:	Admin staff
Related Use Cases:	None
Preconditions:	Bus record must already exist in the database.
Postconditions:	Admin staff to search the bus status and consult the customer for booking record.
Trigger:	Select “View/Edit Bus” through the dropdown list of the Bus menu.
Flow:	<ol style="list-style-type: none"> 1. Admin staff click on “View/Edit Bus” under Bus menu. 2. System displays a new window to search for Bus record using bus number plate and/or owner’s name. 3. Admin staff to enter the bus number plate and/or select the owner’s name and click “Search” button. 4. Admin staff select the driver record by highlighting the row and click “Edit” button. 5. The system will generate a new window page with the selected bus information shown. 6. Admin staff can view the bus status such as “Available” or “In workshop”.
Exceptions:	Bus record does not exist.

Name of Use Case:	Manage Bus Availability Status
Use Case No.:	8.0
Created By:	ZhangYuTong
Date Created:	31/10/20
Description:	Admin staff modify bus availability information on the view/edit bus interface.
Actors:	Admin staff
Related Use Cases:	None
Preconditions:	Bus record must exist in the database.
Postconditions:	Bus status availability is changed and updated in the system database.
Trigger:	Select “View/Edit Bus” through the dropdown list of the Bus menu.
Flow:	<ol style="list-style-type: none"> 1. Admin staff click on “View/Edit Bus” under Bus menu. 2. System displays a new window to search for Bus record using bus number plate and/or owner’s name. 3. Admin staff to enter the bus number plate and/or select the owner’s name and click “Search” button. 4. Admin staff select the driver record by highlighting the row and click “Edit” button. 5. The system will generate a new window page with the selected bus information shown. 6. Admin staff can update the bus status such as “Available” or “In workshop”. 7. Admin staff clicks on “Submit” button. 8. System will update the database on the modified fields. 9. System will return a message on the update status if successful or fail.
Exceptions:	Bus record does not exist.

Name of Use Case:	Manage Bus Availability Status
Use Case No.:	8.0
Created By:	ZhangYuTong
Date Created:	31/10/20
Description:	Admin staff modify bus availability information on the view/edit bus interface.
Actors:	Admin staff
Related Use Cases:	None
Preconditions:	Bus record must exist in the database.
Postconditions:	Bus status availability is changed and updated in the system database.
Trigger:	Select “View/Edit Bus” through the dropdown list of the Bus menu.
Flow:	<ol style="list-style-type: none"> 1. Admin staff click on “View/Edit Bus” under Bus menu. 2. System displays a new window to search for Bus record using bus number plate and/or owner’s name. 3. Admin staff to enter the bus number plate and/or select the owner’s name and click “Search” button. 4. Admin staff select the driver record by highlighting the row and click “Edit” button. 5. The system will generate a new window page with the selected bus information shown. 6. Admin staff can update the bus status such as “Available” or “In workshop” 7. Admin staff clicks on “Submit” button. 8. System will update the database on the modified fields. 9. System will return a message on the update status if successful or fail.
Exceptions:	Bus record does not exist.

Name of Use Case:	Authenticate User
Use Case No.:	9.0
Created By:	Lum Zhi Yan
Date Created:	17/10/20
Description:	Users are required to login to the system with their ID and password respectively.
Actors:	Admin staff, external bus owner
Related Use Cases:	None
Preconditions:	1. User must have an authorized ID and password given by SBC.
Postconditions:	1. User is able to log in to the system successfully.
Trigger:	Click on the “Login” button on the Windows application or mobile web application.
Flow:	<ol style="list-style-type: none"> 1. User enters ID to the Windows application or mobile web application. 2. User enters password to the Windows application or mobile web application. 3. User clicks on the “Login” button. 4. System validates user login credentials and grant access to the user. 5. User successfully log in to the system.
Exceptions:	<ol style="list-style-type: none"> 1. User enters an invalid ID. 2. User enters an invalid password.

Name of Use Case:	Reset Password
Use Case No.:	10.0
Created By:	Lum Zhi Yan
Date Created:	17/10/20
Description:	External bus owner is required to verify their identity with personal question in order to reset password.
Actors:	External bus owner
Related Use Cases:	None
Preconditions:	<ul style="list-style-type: none"> 1. User is unable to login to the Windows application or mobile web application. 2. Admin staff must be staff registered under SBC. 3. External bus owner has a dedicated ID given by SBC.
Postconditions:	<ul style="list-style-type: none"> 1. External bus owner is able to reset password.
Trigger:	Click on the “Forget password” on the mobile web application.
Flow:	<ul style="list-style-type: none"> 1. External bus owner clicks on the “forget password” on the mobile web application. 2. System prompts the external bus owner to enter the last 4 digits of the IC number. 3. External bus owner enters valid 4 last digits of IC number. 4. System validates external bus owner. 5. External bus owner may reset their password. 6. Password is reset.
Exceptions:	<ul style="list-style-type: none"> 1. External bus owner is unable to answer the question prompted by the system.

Name of Use Case:	Check List of Changes or Cancellation
Use Case No.:	11.0
Created By:	Lum Zhi Yan
Date Created:	17/10/20
Description:	External bus owner may check the details of changes or cancellations made by the customers on the mobile web application.
Actors:	External bus owner
Related Use Cases:	None
Preconditions:	1. Admin staff has updated the details of changes or cancellation of bookings on the Windows application.
Postconditions:	None
Trigger:	Click on the “Notification” button on the homepage of the mobile web application.
Flow:	<ol style="list-style-type: none"> 1. External bus owner clicks on the “Notification” button on the homepage of the mobile web application. 2. System displays the details of changes or cancelled booking made by the customers.
Exceptions:	None

Name of Use Case:	Manage List of Customer Booking Request
Use Case No.:	12.0
Created By:	Lum Zhi Yan
Date Created:	18/10/20
Description:	External bus owner can accept the booking request based on the dates in which he/she is able to satisfy the customers' requirements.
Actors:	External bus owner
Related Use Cases:	12.1 Select Bus / Van Available 12.2 Select Assigned Driver
Preconditions:	1. Admin staff has outsourced the customer booking request to external provider.
Postconditions:	1. The customer booking request is selected.
Trigger:	Click on the “Select” button in the Booking Request page.
Flow:	<ol style="list-style-type: none"> External bus owner clicks on the “Booking Request” button on the homepage of the mobile web application. System displays the details of booking requests made by the customers. External bus owner can select the customer booking request based on the dates in which he/she is able to satisfy the customers' requirements. Customer booking request is selected.
Exceptions:	<ol style="list-style-type: none"> The list of customer booking request is empty. External bus owner is unavailable for all the dates shown in the list of customer booking requests.

Name of Use Case:	Select Bus/Van Available
Use Case No.:	12.1
Created By:	Lum Zhi Yan
Date Created:	18/10/20
Description:	External bus owner can select the bus/van available based on the customers' requirements.
Actors:	External bus owner
Related Use Cases:	12.0 Manage List of Customer Booking Request
Preconditions:	<ul style="list-style-type: none"> 1. External bus owner has available bus/van on the selected dates. 2. Admin staff has outsourced the customer booking request to external provider.
Postconditions:	<ul style="list-style-type: none"> 1. The assigned bus/van will be updated in the database.
Trigger:	Click on the bus/van shown in the bus/van selection box.
Flow:	<ul style="list-style-type: none"> 1. System displays a list of bus/van available. 2. External bus owner selects the appropriate bus to satisfy the customers' requirements. 3. External bus owner clicks on the “Save” button. 4. System updates the selected bus/van in the database.
Exceptions:	<ul style="list-style-type: none"> 1. No available buses/vans are available at that time.

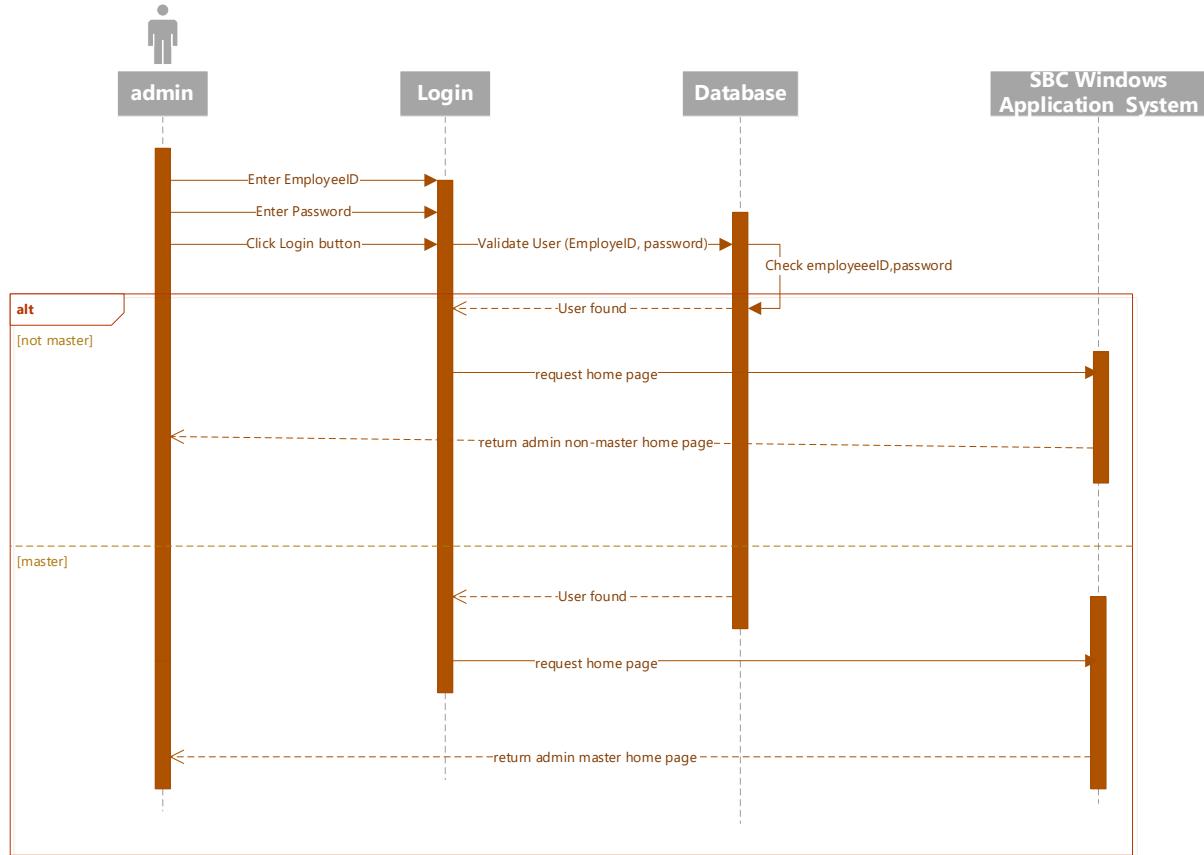
Name of Use Case:	Select Assigned Driver
Use Case No.:	12.2
Created By:	Lum Zhi Yan
Date Created:	18/10/20
Description:	External bus owners can assign any driver from their company that has registered with SBC to satisfy the customers' booking requests.
Actors:	External bus owner
Related Use Cases:	12.0 Manage List of Customer Booking Request
Preconditions:	<ul style="list-style-type: none"> 1. External bus owner has unassigned driver on the selected dates. 2. Driver from the external bus provider has registered with SBC.
Postconditions:	<ul style="list-style-type: none"> 1. The selected driver is updated to the database.
Trigger:	Click on the driver shown in the driver selection box.
Flow:	<ul style="list-style-type: none"> 1. External bus owner clicks on the "Selection" button shown in the assign bus/driver page. 2. System displays a list of available drivers. 3. External bus owner selects the appropriate drivers to satisfy customer booking request. 4. External bus owner clicks on "Save". 5. System updates the selected driver to the database.
Exceptions:	<ul style="list-style-type: none"> 1. No registered drivers are available at that time.

Name of Use Case:	Create New Driver Account
Use Case No.:	13.0
Created By:	Lum Zhi Yan
Date Created:	18/10/20
Description:	External bus owner can create a new driver account to be registered with SBC on the mobile web application.
Actors:	External bus owner
Related Use Cases:	None
Preconditions:	1. Driver has not registered with SBC.
Postconditions:	1. The new driver information is created in the database.
Trigger:	Click on the “Create New Driver” button under the Driver Profile tab on the mobile web application.
Flow:	<ol style="list-style-type: none"> 1. External bus owner clicks on the “Driver Profile” button from the menu. 2. External bus owner clicks on “Create New Driver” tab. 3. System displays an empty form that requires driver information. 4. External bus owner entered all the details of the new driver into the empty form. 5. External bus owner clicks on the “Save” button at the bottom of the page. 6. System returns with a successful confirmation. 7. System updates the new driver information to the database.
Exceptions:	None

Name of Use Case:	Update Driver Profile
Use Case No.:	14.0
Created By:	Lum Zhi Yan
Date Created:	18/10/20
Description:	External bus owner can update the personal details of any driver from their company that has registered with SBC on the mobile web application.
Actors:	External bus owner
Related Use Cases:	None
Preconditions:	1. The details of the external drivers are registered in the SBC system.
Postconditions:	1. The driver profile is updated to the database.
Trigger:	1. Click on the “Update Driver Profile” button under the Driver Profile tab on the mobile web application.
Flow:	<ol style="list-style-type: none"> 1. External bus owner clicks on the “Driver Profile” button from the menu. 2. External bus owner clicks on “Update Driver Profile” tab. 3. System displays a list of the registered driver profiles. 4. External bus owner selects the driver profile which he/she wants to check. 5. External bus owner clicks on the “Update” button shown beside driver profile. 6. System displays the details of the selected driver. 7. External bus owner can modify the fields in the driver details. 8. External bus owner clicks on “Save” to confirm the changes. 9. System returns with a successful confirmation. 10. System updates the driver information to the database.
Exceptions:	Driver profile does not exist.

8.3 Sequence Diagram

Admin Login



The diagram above depicts the sequence of the Admin login process into the SBC Windows Application System. Firstly, the Admin needs to input his/her EmployeeID and password to log into the system and press the login button. The login page would send the data input to the database to request data validation of the EmployeeID and the password once the database approves the authentication and found the user. The login system would directly display the Admin home page based on whether the employee is master or non-master.

Admin Create Customer Booking

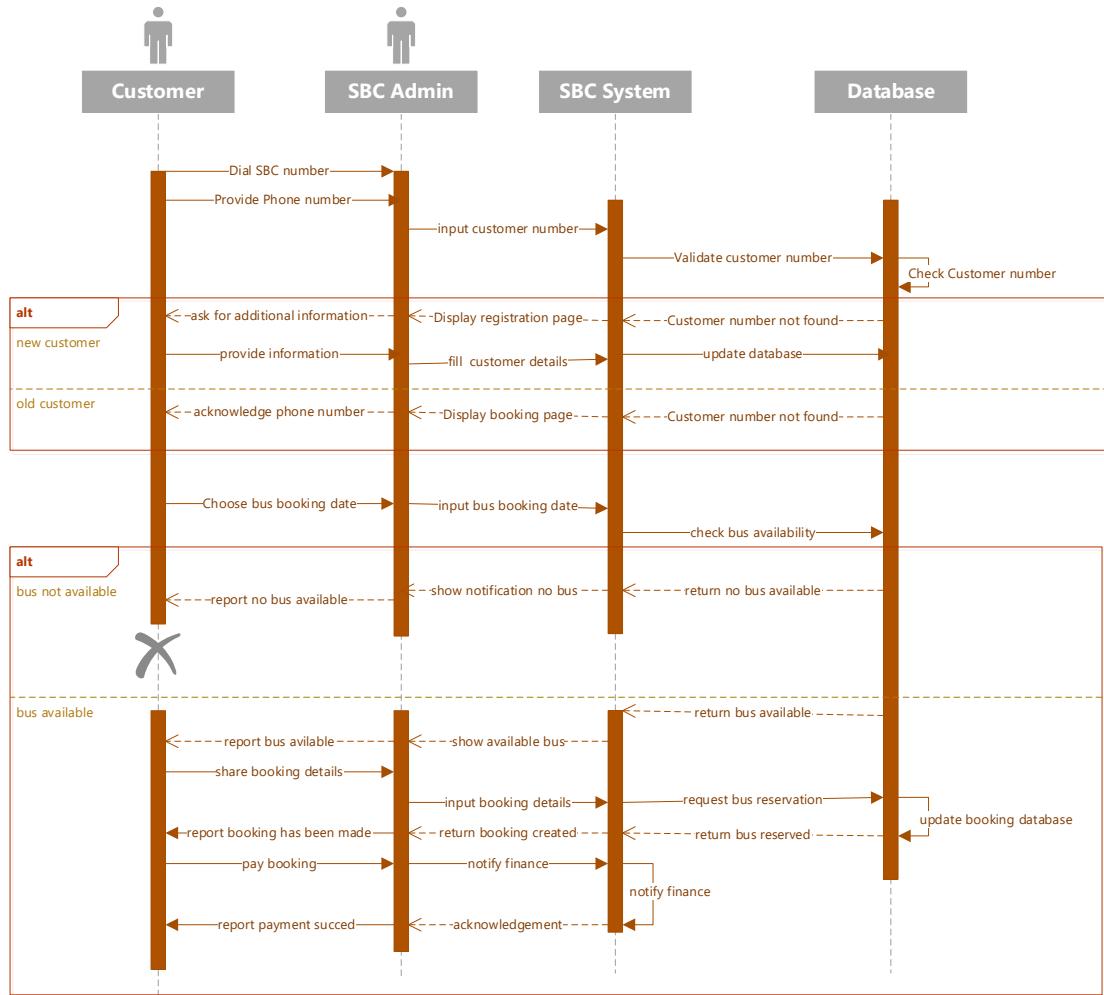
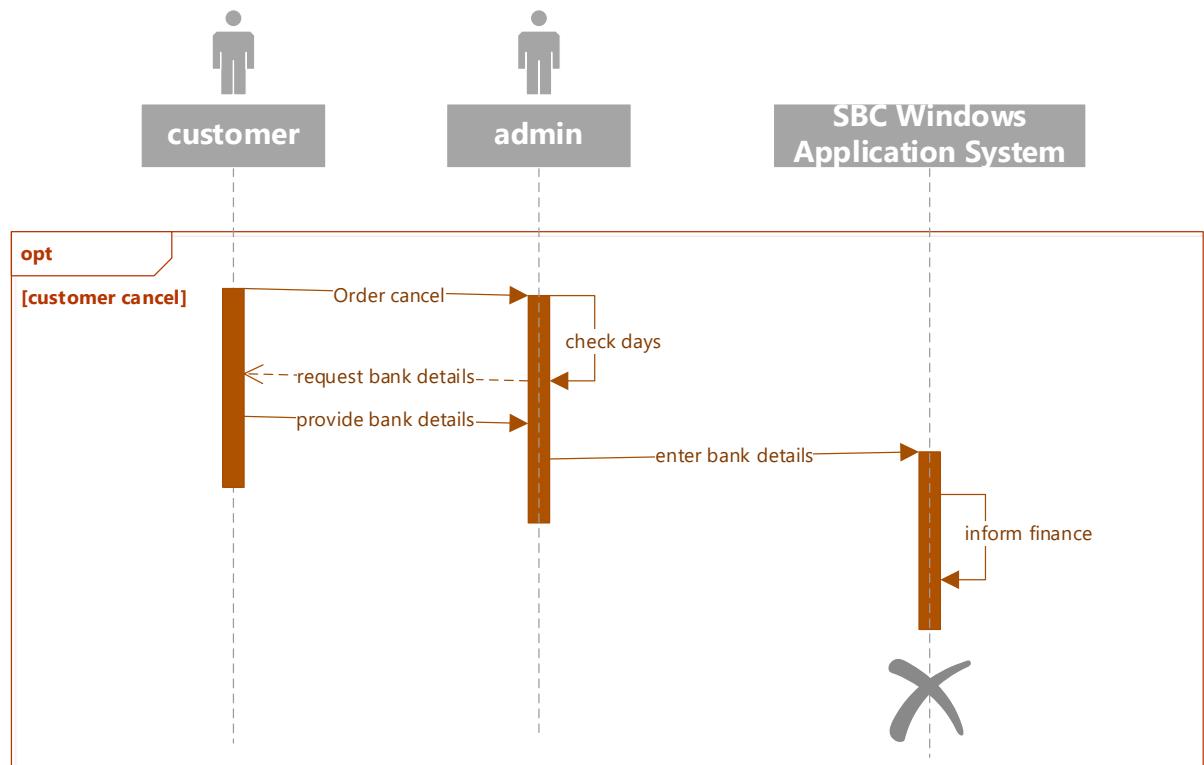


Diagram above shows how Admin creates customer booking. It begins with customers dial the SBC number and provides their contact number to SBC Admin. Additionally, Admin will input their customer number into the creates booking page. The system will then authenticate the customer number inside the database. If the customer s a new customer database will notify, there is no customer data found inside the database. It will automatically direct the page to the customer registration page where Admin could fill in the customer information. But If the customer number is already inside the database, the system would direct the page straight to the booking page. The customer then can choose the booking date, and Admin would check if there are any available busses. The system would check the database for bus availability. If there is no available bus during the day, the database will return no bus available. Thus, the system display there is no bus available. The Admin will inform the customer there is no bus available in which the customer can choose to cancel the bus reservation or choose another date.

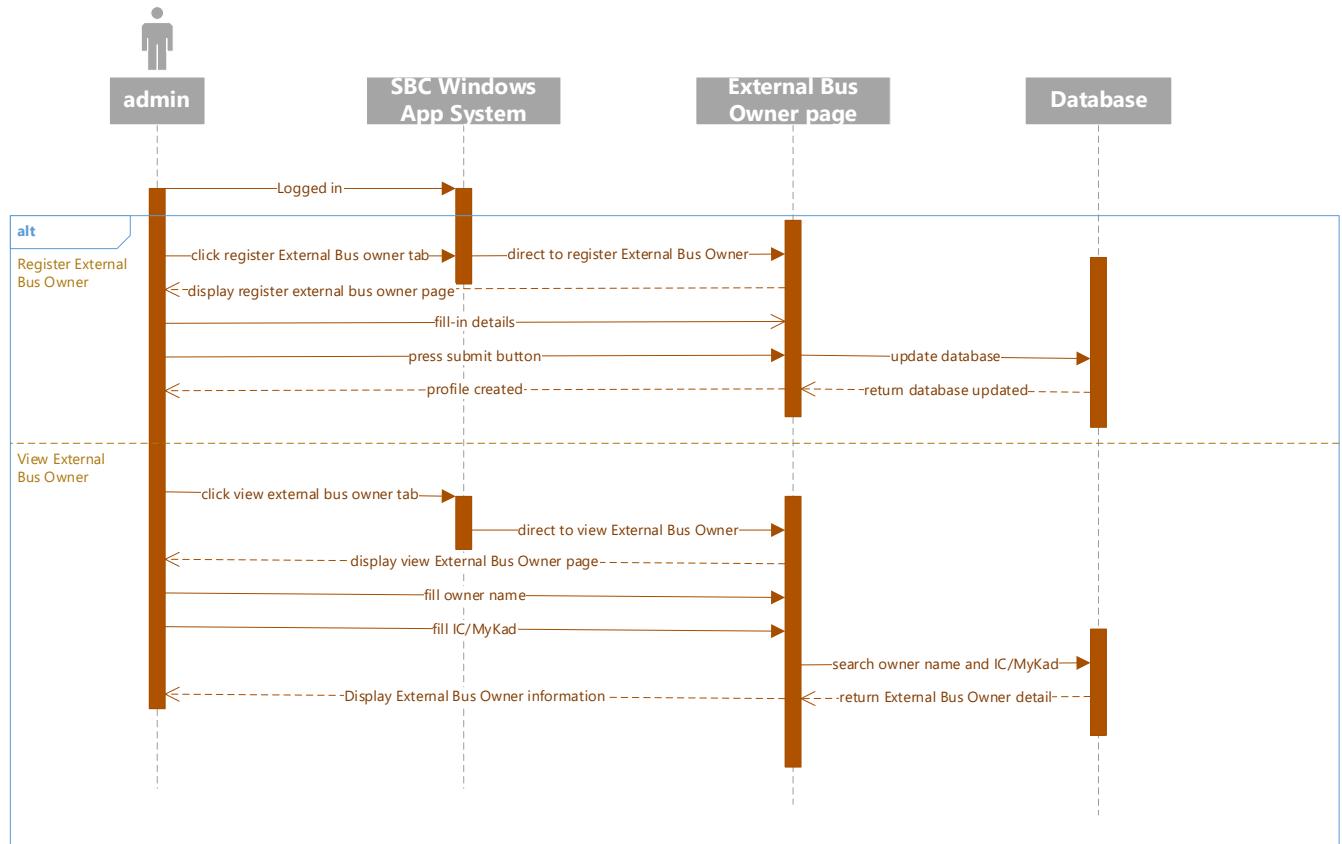
On the other hand, if there is a bus available during the day, Admin will inform the customer whether the bus is available. Additionally, the customer would give more of the booking information to Admin. The Admin then fills up all the booking details into the system, which then updates the database. The system then will acknowledge the booking and notified the Admin the booking had been made. The customer then may pay the booking fee, which Admin would notify the Finance department through the system. The system will display an invoice or acknowledgment if the payment successful, which the Admin would report back to the customer that their payment is successful.

Customer Cancel Order



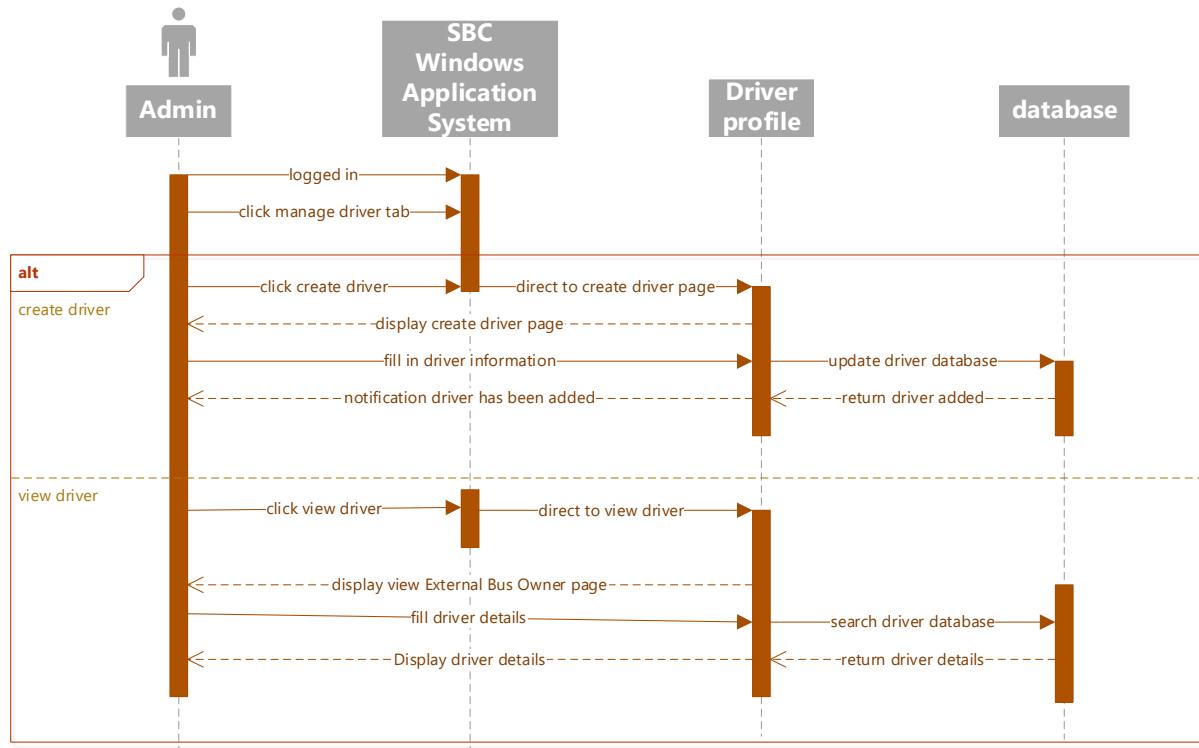
The figure above explains the customer cancellation sequence. If the customer decided to cancel the order after payment, The Admin would then check how many days have passed since the customer made the payment. The Admin would then ask for the customer refund bank details. The Admin would enter the bank details into the system. Additionally, the system would inform the finance about the refund process and let finance handle the rest of the refund process.

Admin Manage External Bus Owner



The diagram above explains how Admin manages the External Bus Owner account. Admin needs to login to the first before managing the External bus owner account. Admin can choose to register a new External Bus Owner or View External Bus Owner profile. If Admin decides to register External Bus Owner, Admin needs to click the register External Bus Owner tab in the system. It will direct the page to register the External Bus Owner page where Admin can fill-in the details of the new Owner. After filling in the External Bus Owner details, the admin press the submit button to update the database. The system then alerts the Admin that the Owner has been added. If Admin decides to view the External Bus Owner account, Admin needs to press the view External Bus Owner tab. Then the system would direct the page to view the External Bus Owner page. Admin then fills in Owner's name & IC or MyKad then press the search button. The submit button would directly send the information to the database to search for the Owner's information. The database then returns the External Bus Owner information then the system displays the information.

Admin Manage Driver



The Sequence diagram above explains how Admin manages the driver account. Admin needs to login to the system first before managing the driver account. Admin can choose to register a new driver or view driver. If Admin decides to register a new driver, Admin needs to click register the driver into the system. It will direct the page to register the driver page where Admin can fill-in the details of the new driver. Admin then needs to fill in the driver details and press the submit button. The system sends the details to the database to record and update the driver database. The system then alerts the Admin the driver has been added. If Admin decides to view the driver information, Admin needs to press the view driver tab. Admin then fills in driver information then press the search button. The system sends the data inputted to the database to search for the driver's information. The database returns the driver information to the system. Lastly, the system will display the driver information to Admin.

Admin Manage Bus

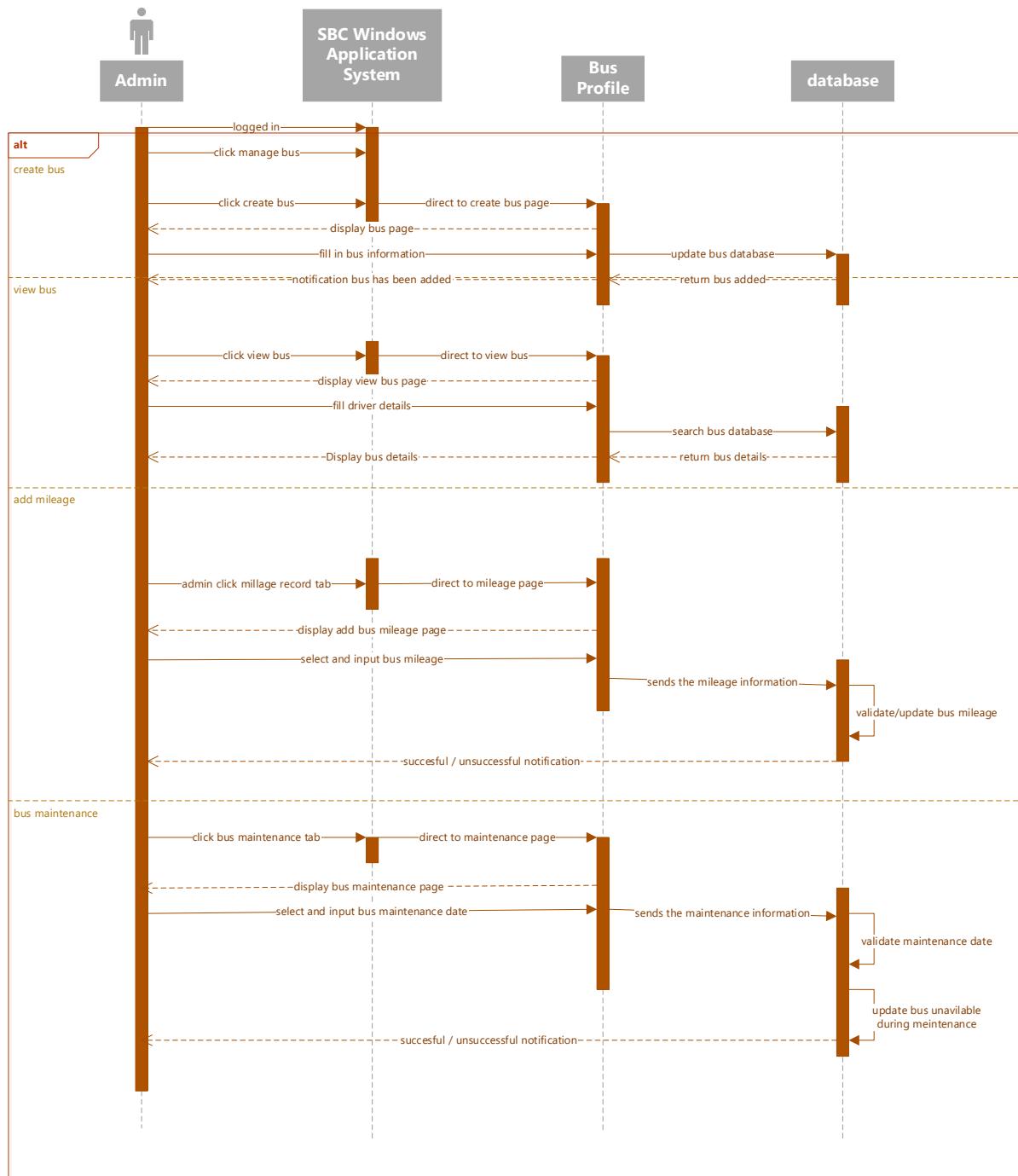
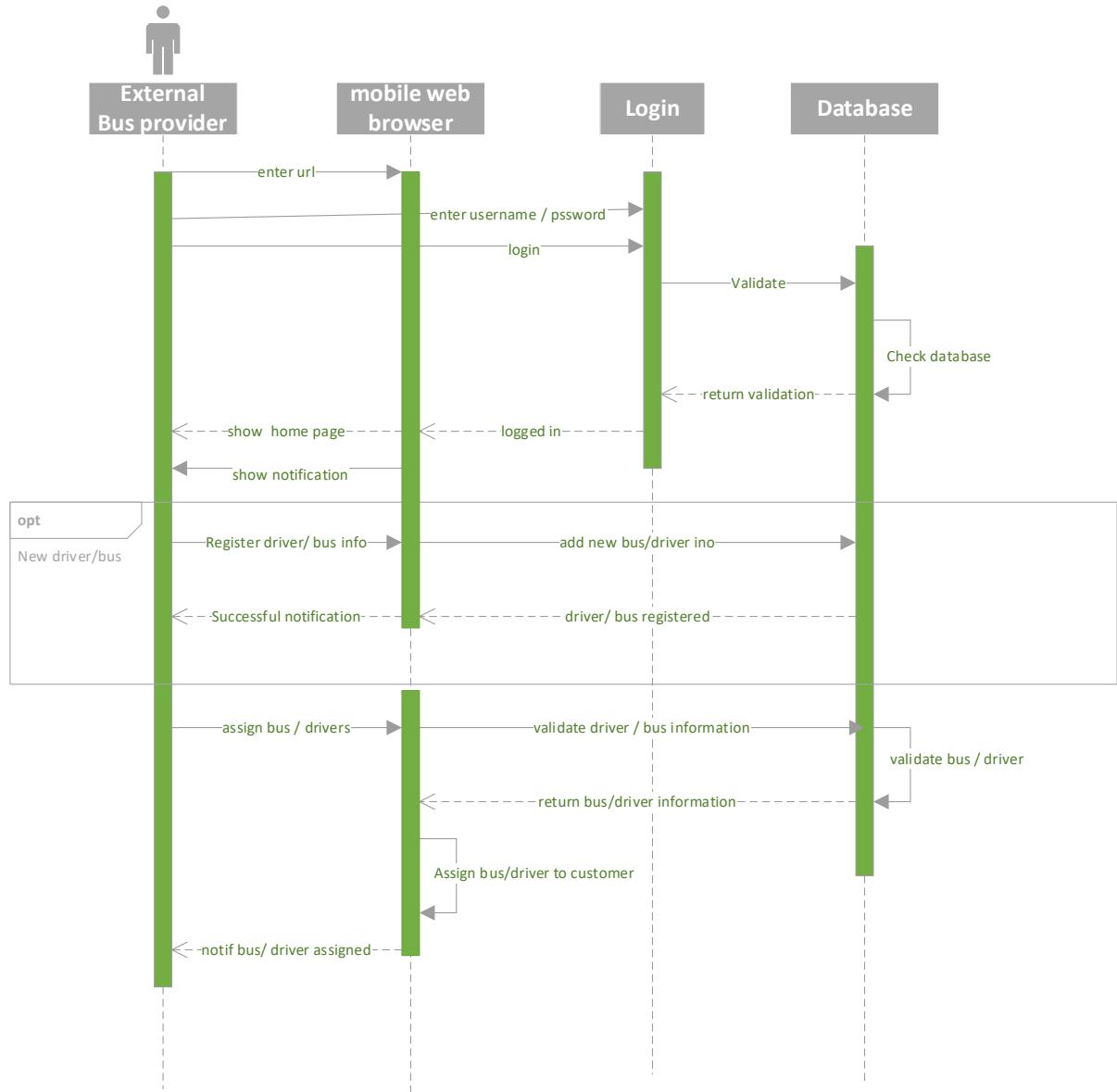


Figure above represents how admin register or view the bus profile. The Admin needs to log in first into the system then press the manage bus tab. Admin then had the option to register a new bus or view the registered bus information. If Admin decides to register a new bus, Admin needs to press the create bus tab and fill in the bus details. The system then updates the database and alerts the Admin if the registration process is successful. On the other hand, if the Admin decides to view the bus details. Admin needs to press the view bus tab and fill in the text box required to search for the details. Once filled in, Admin needs to press the submit button, then

the system will send the details to the database. The database searches the bus details and returns all the details to the system bus page where now the Admin able to see the bus information. Additionally, Admin able to add the mileage record for whichever bus in the system. By clicking into the mileage record tab, Admin will be directed to the mileage page where Admin can input the chosen bus mileage record. Later, the information will be sent to the database and update the bus information. Lastly, Admin had an option to create a maintenance schedule for both internal dan external bus. Admin needs to choose the bus maintenance tab and choose the bus which going to be in maintenance. After filling in the maintenance dates, Admin press the submit button to update the bus availability inside the database. Once the bus is in maintenance, the bus won't be available during the maintenance dates.

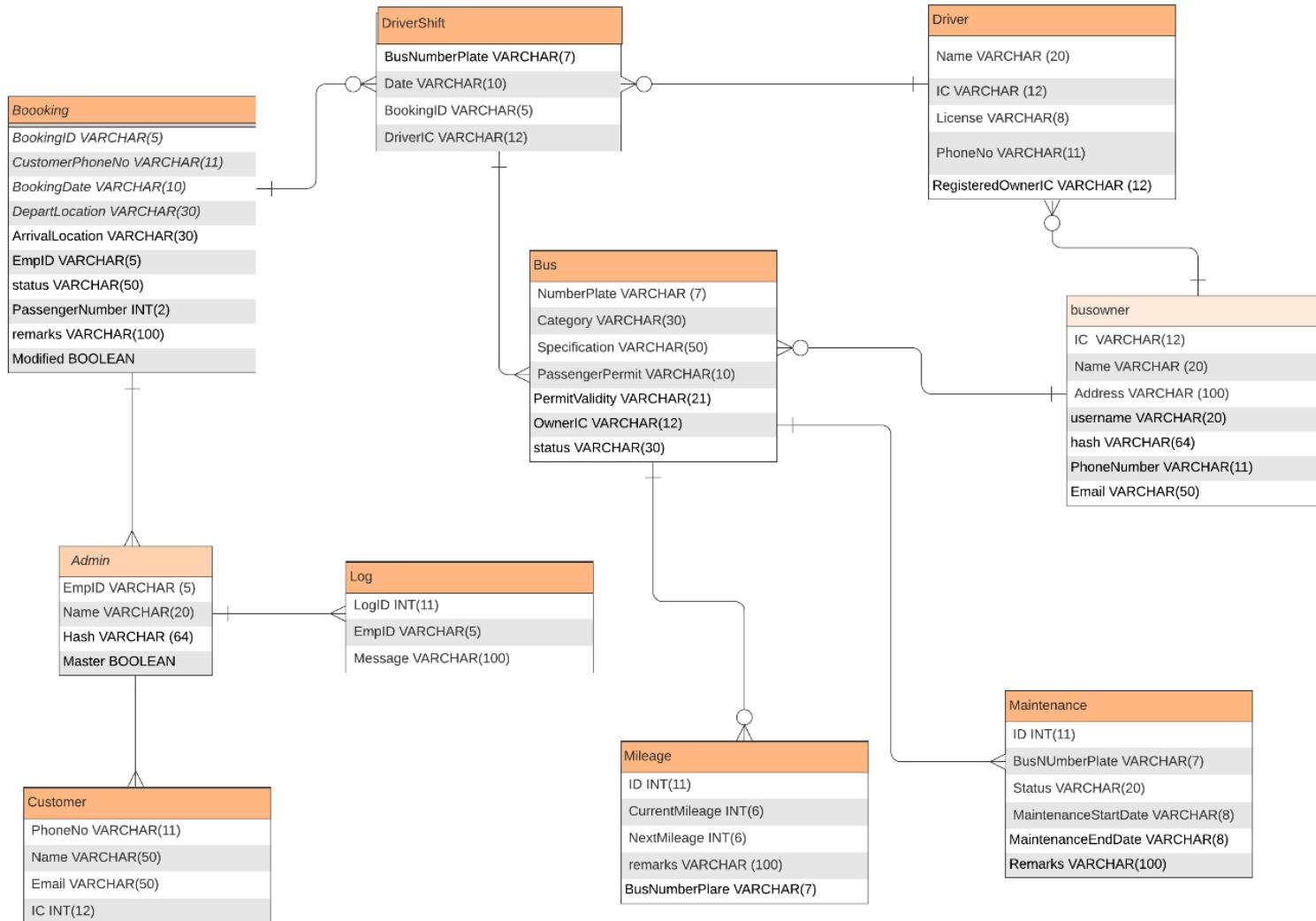
External Bus Owner Assign External Driver



The diagram shows the sequence of how the External Bus Owner assigns their driver to the customer. First, the External Bus Owner needs to enter the website URL and enter the username & password to log in to the site. The website then validates the owner information with the database and display the Owner's home page. The website then shows notification about the upcoming customer booking order. The Owner then had the option to register a new driver or assign the registered external driver to the customer booking. If the Owner decides to register, a new driver or vehicle Owner needs to fill in the new driver or bus information to insert the information into the database. Once the driver or bus is successfully registered, the Owner will receive notification about the successful registration. The Owner can assign which driver or bus to go for customer booking. The system then sends the details to validate the recorded

driver or bus information inside the database. After the validation process is completed, the system will then approve the assigned process and notifies the External Bus Owner.

8.4 Data Persistence Diagram



8.4.1 Data Dictionary

Table Name	Attributes Name	Description	Type	Size	Required	PK or FK	FK reference table
Booking	BookingID	ID belonging to the specific order	VARCHAR	5	Y	PK	Admin
	CustomerPhoneNo	Customer's phone number	VARCHAR	11	Y		
	BookingDate	Date of booking	VARCHAR	10	Y		
	DepartLocation	Departure Location	VARCHAR	30	Y		
	ArrivalLocation	Arrival Location	VARCHAR	30	Y		
	EmpID	Employee ID	VARCHAR	5	Y	FK	
	status	Booking status	VARCHAR	50	N		
	PassengerNumber	Number of passengers	INT	2	Y		
	remarks	Additional remarks	VARCHAR	100	N		
	Modified	Shows if the booking is modified at external providers have not acknowledged yet	BOOLEAN		Y		
DriverShift	BusNumberPlate	Bus number plate	VARCHAR	7	Y	FK	Bus Booking Driver
	Date	Date of booking	VARCHAR	10	Y		
	BookingID	Booking ID	VARCHAR	5	Y	PK FK	
	DriverIC	Driver's IC number	VARCHAR	12	Y	PK FK	
Driver	Name	Driver's name	VARCHAR	20	Y		busOwner
	IC	Driver's IC number	VARCHAR	12	Y	PK	
	License	Driver's license number	VARCHAR	8	Y		
	PhoneNo	Driver's phone number	VARCHAR	11	Y		
	RegisteredOwnerIC	The owner's IC the driver is registered to	VARCHAR	12	Y	FK	
Bus	NumberPlate	Bus number plate	VARCHAR	7	Y	PK	
	Category	Bus category	VARCHAR	30	Y		
	Specification	Bus specification	VARCHAR	50	Y		

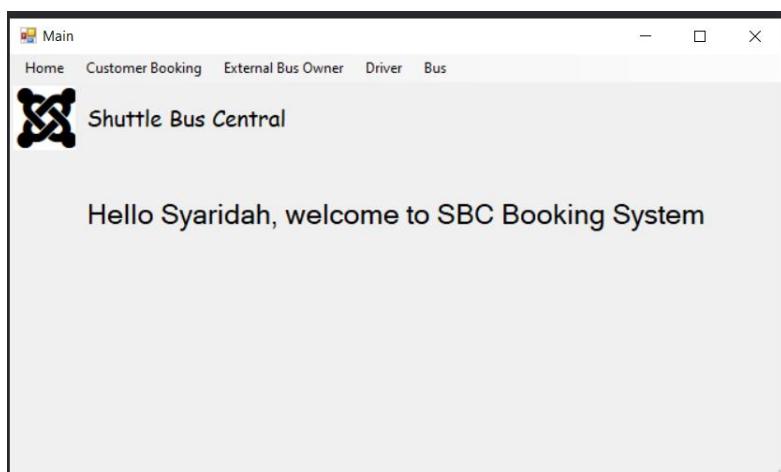
busOwner	PassengerPermit	Bus passenger permit number	VARCHAR	10	Y		busOwner
	PermitValidity	Bus permit validity	VARCHAR	21	Y		
	OwnerIC	Owner's IC the bus is registered to	VARCHAR	12	Y	FK	
	status	Bus's status	VARCHAR	30	Y		
	IC	Bus owner's IC	VARCHAR	12	Y	PK	
	Name	Bus owner's name	VARCHAR	20	Y		
	Address	Bus owner's address	VARCHAR	100	Y		
	username	username used to log into mobile web view	VARCHAR	20	Y		
	hash	MD5 hash used to log in to mobile web	VARCHAR	64	Y		
	PhoneNumber	Bus owner's phone number	VARCHAR	11	N		
	Email	Bus owner's email	VARCHAR	50	N		
Admin	EmpID	Admin's employee ID	VARCHAR	5	Y	PK	Admin
	Name	Admin's name	VARCHAR	20	Y		
	Hash	MD5 hash used to log into windows system	VARCHAR	64	Y		
	Master	Defines if the account is a master account	BOOLEAN		Y		
Log	LogID	Log ID	INT	11	Y	PK	Admin
	EmpID	Employee ID associated with the log	VARCHAR	5	Y	FK	
	Message	Message of the log	VARCHAR	100	Y		
Customer	PhoneNo	Customer's phone number	VARCHAR	11	Y	PK	
	Name	Customer's name	VARCHAR	50	Y		
	Email	Customer's email	VARCHAR	50	Y		
	IC	Customer's IC	INT	12	Y		
Mileage	ID	Mileage ID for every record	INT	11	Y	PK	
	CurrentMileage	The current mileage of the bus	INT	6	Y		

Bus	NextMileage	The next mileage limit for the bus	INT	6	Y		Bus
	remarks	Additional remarks	VARCHAR	100	N		
	BusNumberPlate	Bus number plate for the mileage record	VARCHAR	7	Y	FK	
Maintenance	ID	Maintenance ID	INT	11	Y	PK	Bus
	BusNumberPlate	Bus number plate of the vehicle involved	VARCHAR	7	Y	FK	
	Status	Maintenance status	VARCHAR	20	Y		
	MaintenanceStartDate	Maintenance start date	VARCHAR	8	Y		
	MaintenanceEndDate	Maintenance end date	VARCHAR	8	Y		
	remarks	Additional remarks	VARCHAR	100	N		

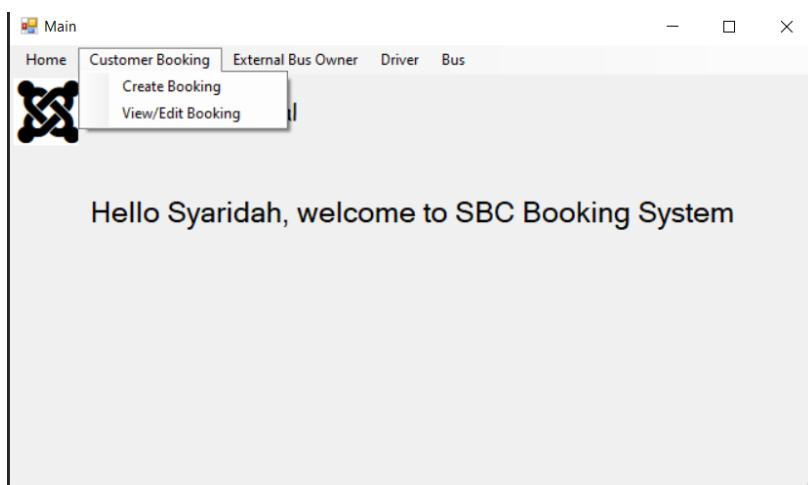
9.0 User interface design and system manual

9.1 Windows System

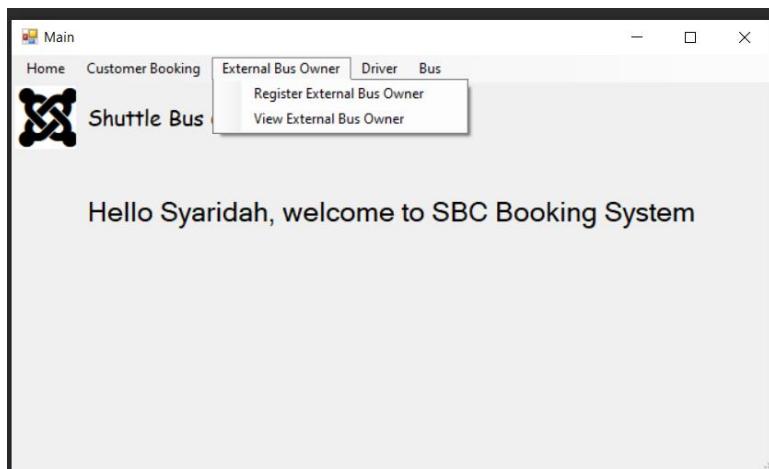
9.1.1 System Menu and Navigation



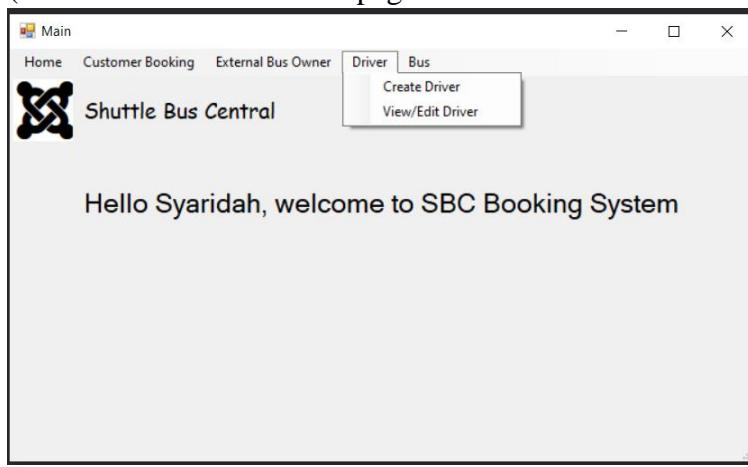
(SBC Windows Application System Non-Master Admin home page)



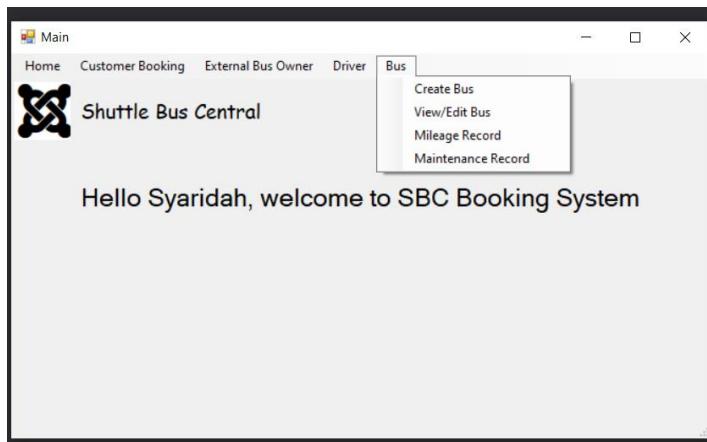
(Non-Master Admin home page – Customer Booking tab)



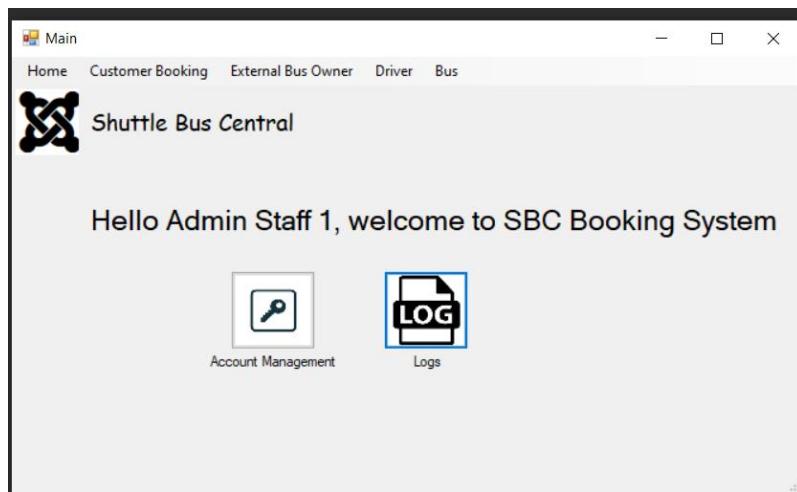
(Non-Master Admin home page – External Bus Owner tab)



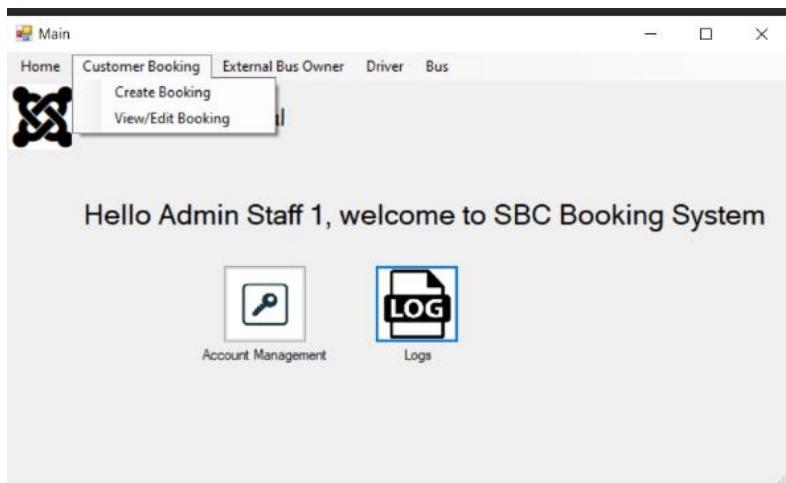
(Non-Master Admin home page – Driver tab)



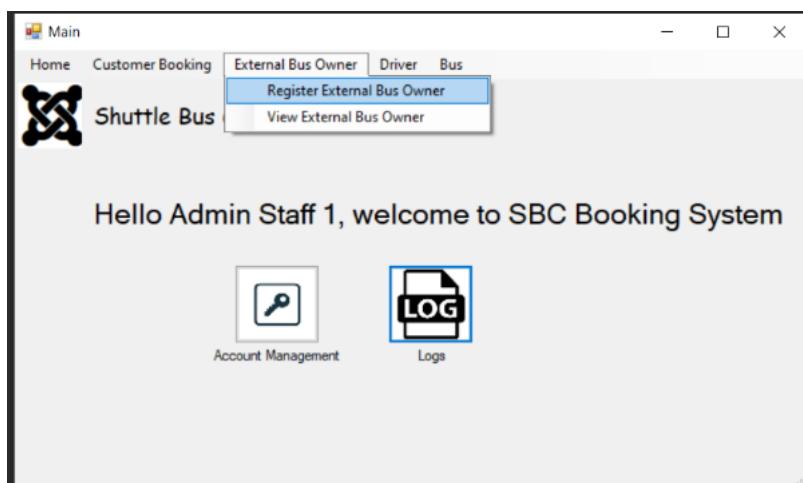
(Non-Master Admin home page –Bus tab)



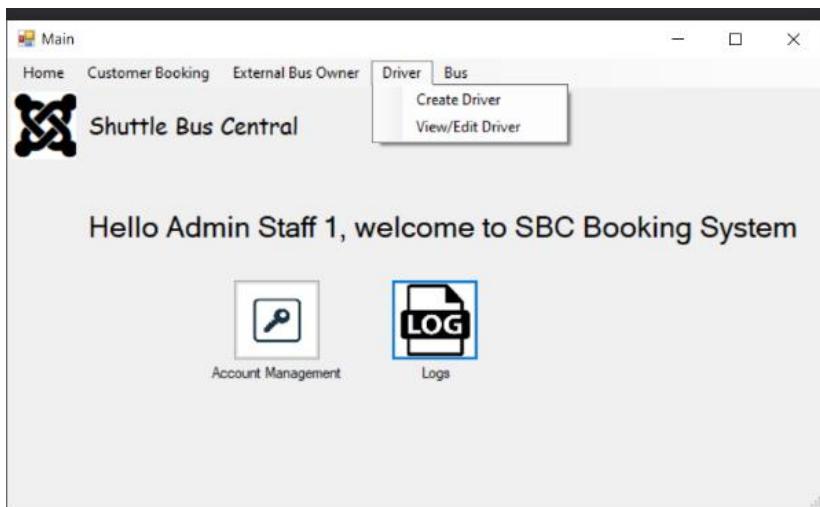
(SBC Windows Application System Master Admin home page)



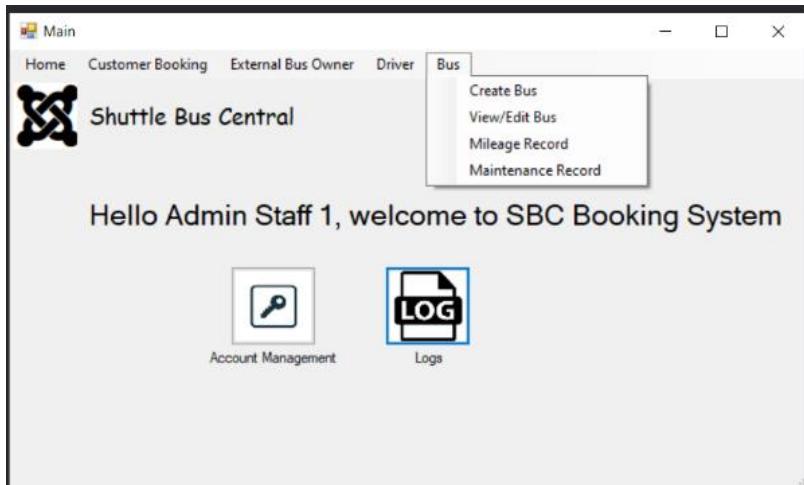
(Master Admin home page – Customer Booking tab)



(Master Admin home page – External Bus Owner tab)



(Master Admin home page – Driver tab)



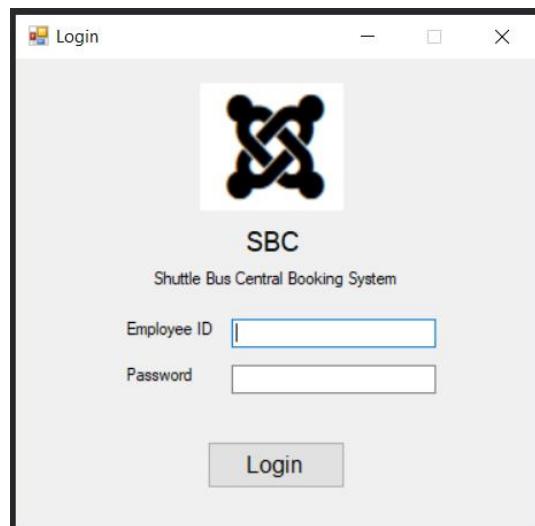
(Master Admin home page –Bus tab)

LogID	EmpID	Message
51	E0001	Assigned a bus &...
52	E0001	Assigned a bus &...
53	E0001	Driver IC 111111...
54	E0001	Added a new cus...
55	E0001	Assigned a bus &...
56	E0001	B0004 booking in...
57	E0001	B0004 booking in...
58	E0001	B0004 booking in...
59	E0001	Driver IC 111111...
60	E0001	B0002 booking in...
*		

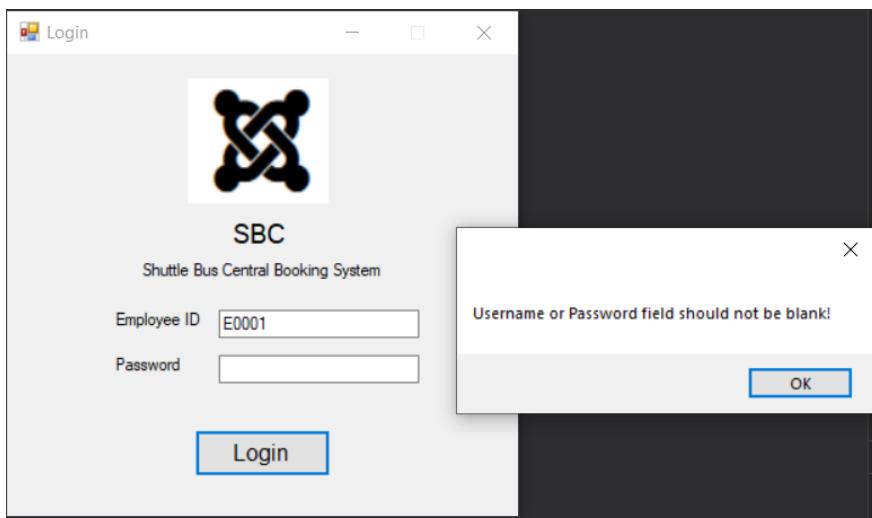
(Non-Master Admin home page – Logs)

Logs will record and show every new update or input done by the Admin. Only Master Admin able to view the Logs.

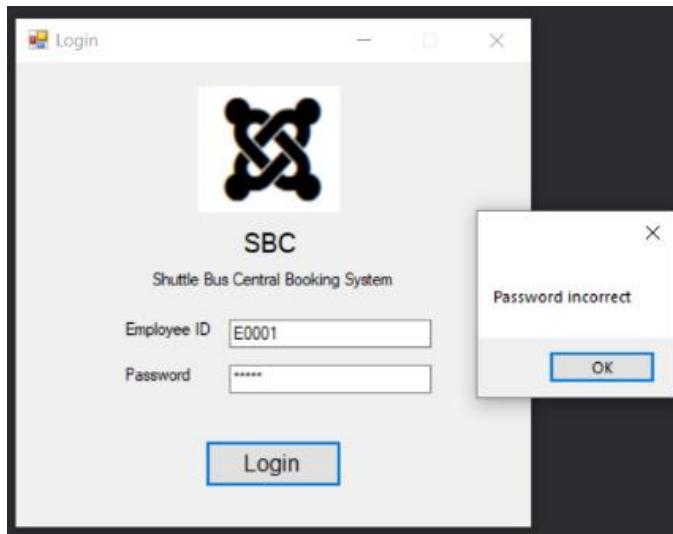
9.1.2 User Authentication & User Level (Access Level)



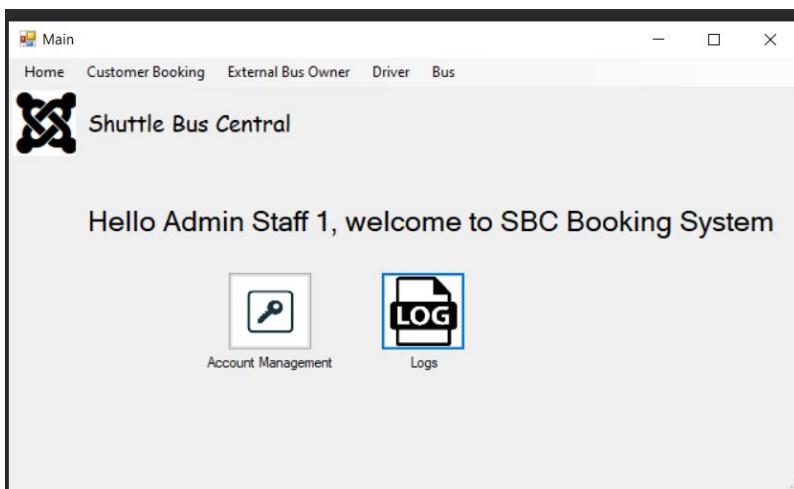
(SBC Windows Application System Login page)



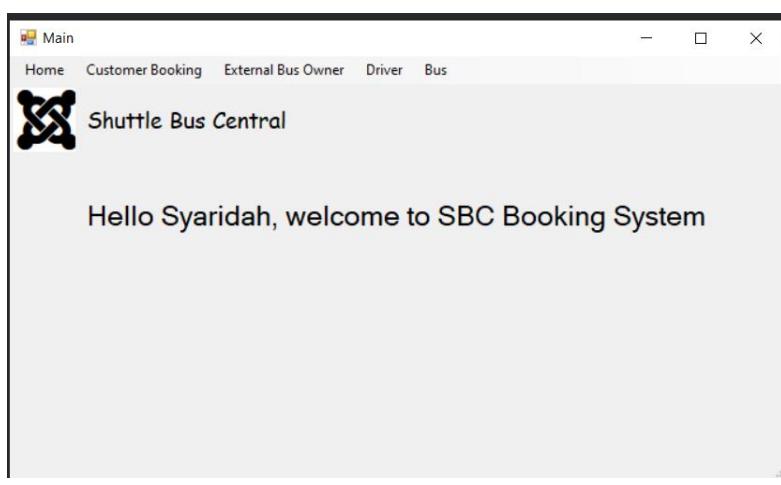
(Windows Login page – User input blank alert)



(Windows Login page – User validation incorrect alert)

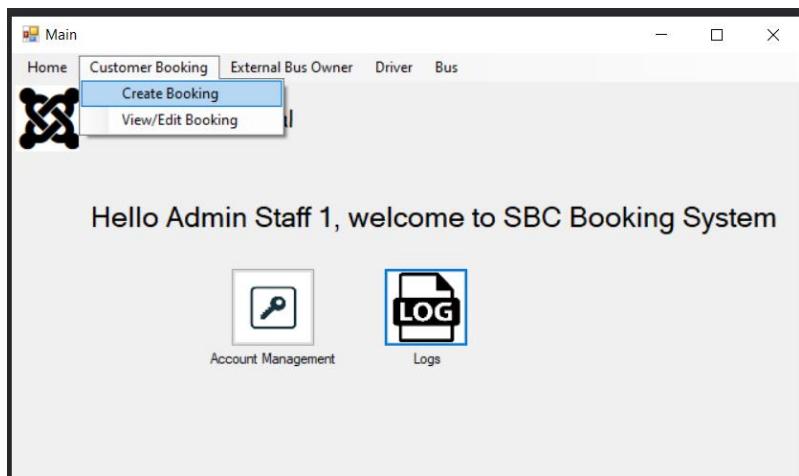


(Windows Admin Login – Admin master home screen)

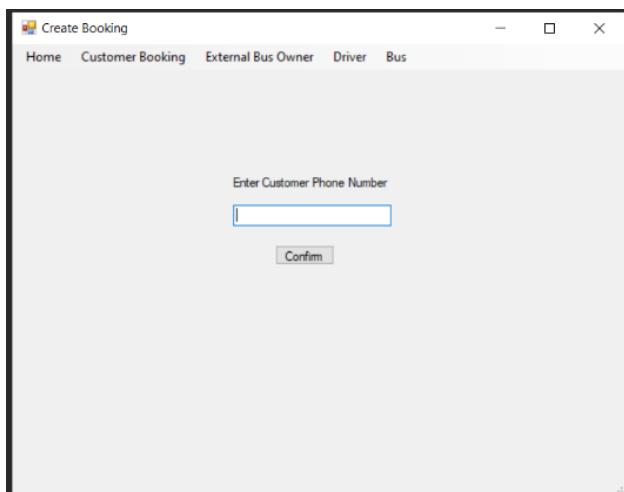


(Windows Admin Login – Admin non-master home screen)

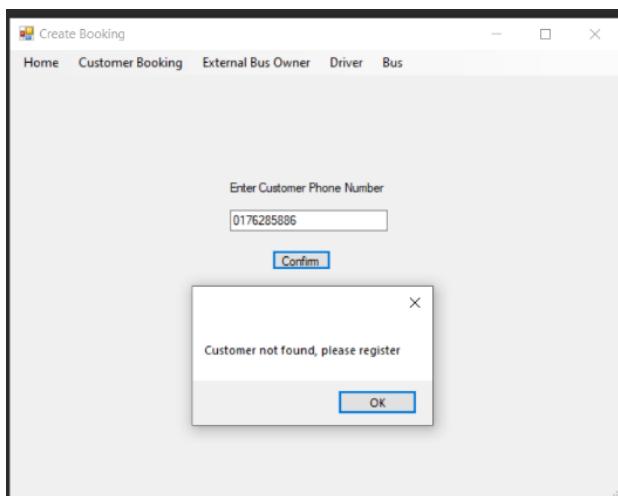
9.1.3 Create a Booking process



(Master Admin home page – Customer Booking tab)



(Create booking page – Enter Customer number)



(Create booking page – Customer number not within the database)

Registration Form

Home Customer Booking External Bus Owner Driver Bus

Shuttle Bus Central

Customer Registration Form

Customer Name

IC/MyKad

Phone Number

Email

Submit

(Create booking page – Customer registration form)

Registration Form

Home Customer Booking External Bus Owner Driver Bus

Shuttle Bus Central

Customer Registration Form

Customer Name

IC/MyKad

Phone Number

Email

Submit

Input must not be empty!

OK

(Create booking page – Customer registration blank input alert)

Registration Form

Home Customer Booking External Bus Owner Driver Bus

Shuttle Bus Central

Customer Registration Form

Customer Name

IC/MyKad

Phone Number

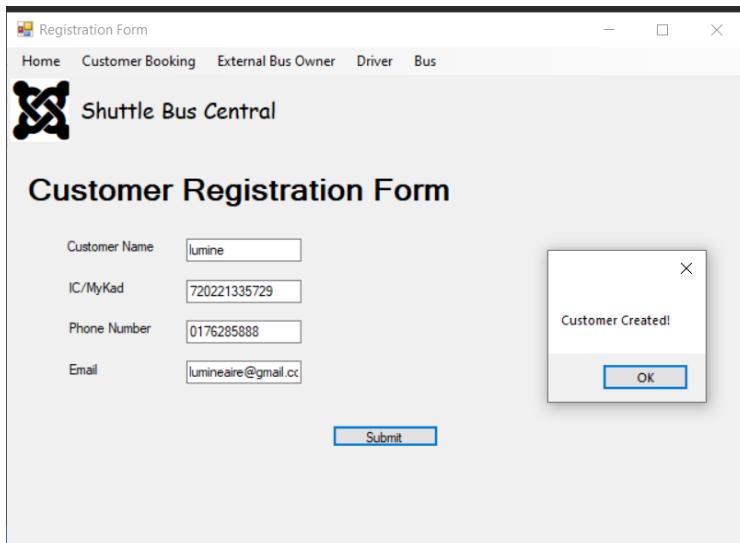
Email

Submit

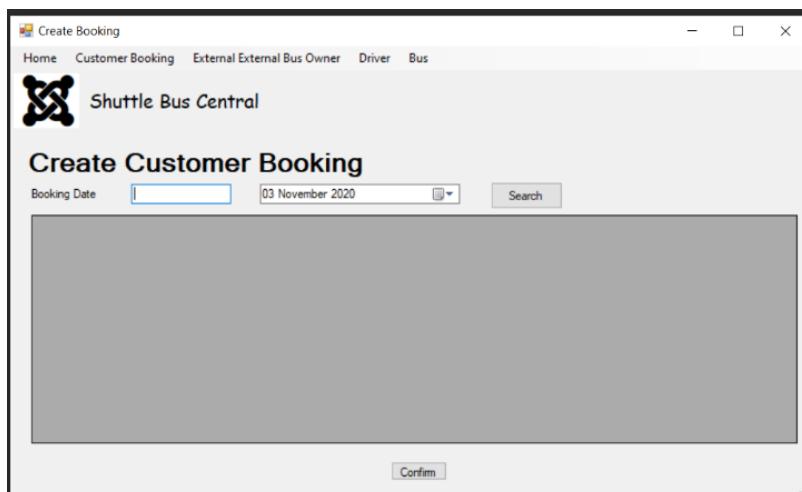
Please input IC correctly.

OK

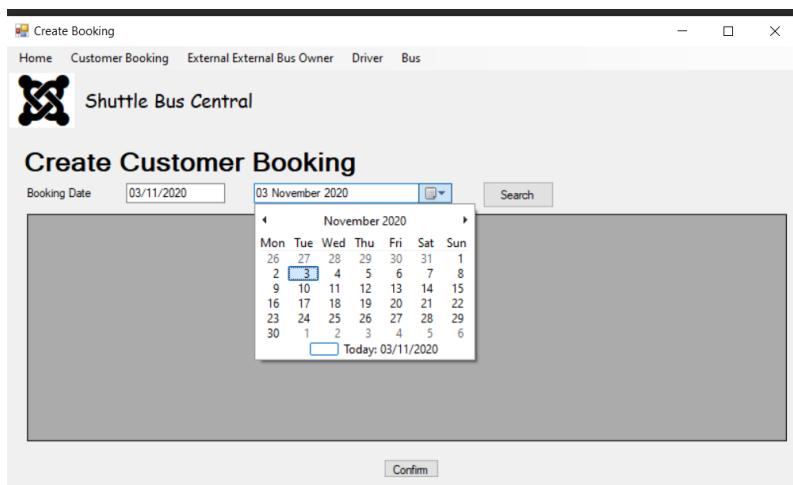
(Create booking page – Customer registration incorrect input format alert)



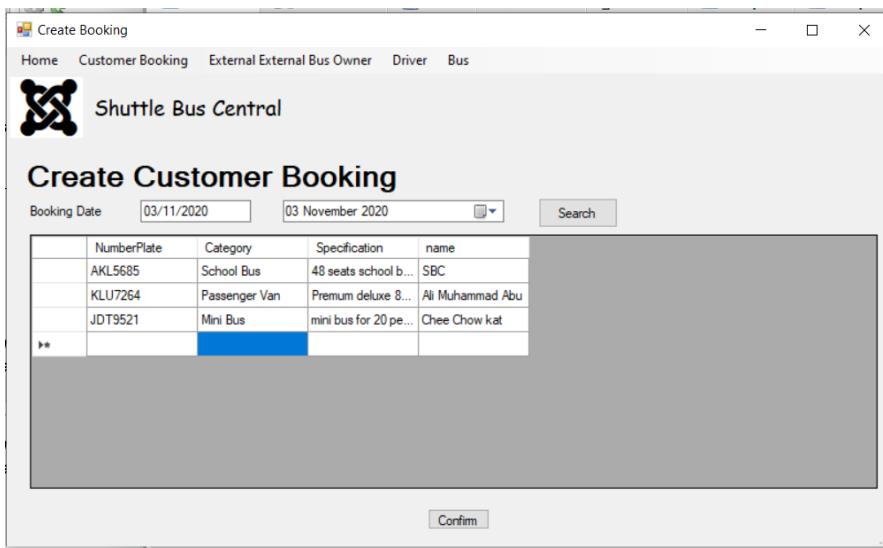
(Create booking page – Customer registration correct input alert)



(Create booking page –Available internal/external bus)

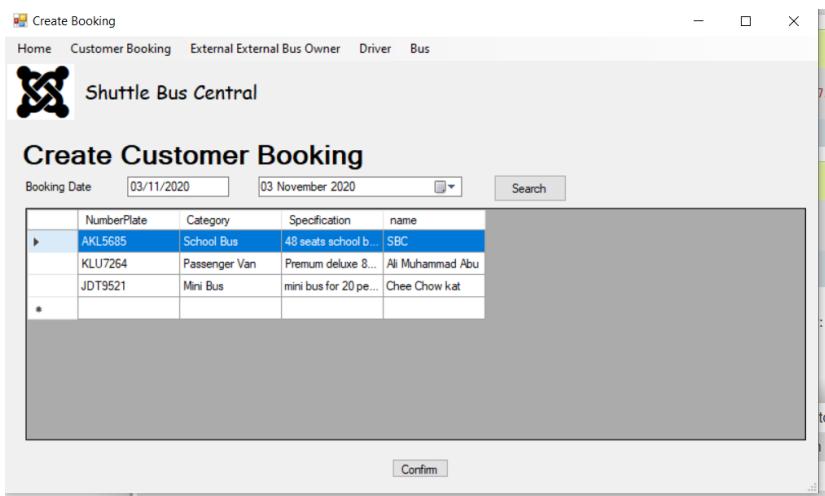


(Create booking page – Search bus by date)



(Create booking page – The available bus)

Note that if it is Internal Bus, it will be under the name SBC, however, if it is External bus, it will be under different owner name.



(Create booking page – Select the available bus)

This screenshot shows the 'Create Customer Booking' page with the 'Customer booking fill-in details' section visible. It includes fields for 'Customer Name' (set to 'shinde'), 'No. Passengers' (empty), 'Phone Number' (set to '0176285886'), 'Shuttle Type' (dropdown menu), 'Departure' (empty), 'Date' (set to '03/11/2020'), 'Arrival' (empty), 'Driver' (dropdown menu), 'Remarks' (text area), and a 'Submit' button at the bottom.

(Create booking page – Customer booking fill-in details)

Create Booking

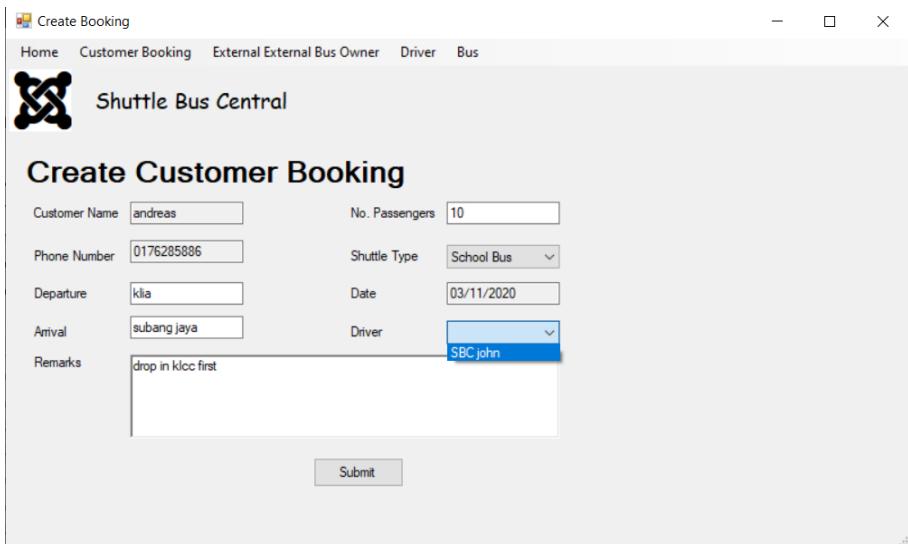
Home Customer Booking External External Bus Owner Driver Bus

Shuttle Bus Central

Create Customer Booking

Customer Name	andreas	No. Passengers	10
Phone Number	0176285886	Shuttle Type	School Bus
Departure	klia	Date	03/11/2020
Arrival	subang jaya	Driver	SBC john
Remarks	drop in klcc first		

Submit



(Create booking page – Customer booking details choose SBC driver)

Create Booking

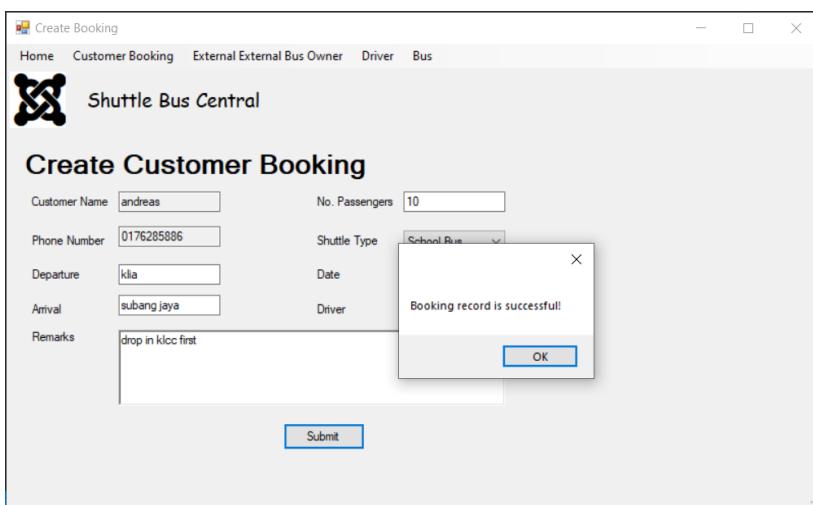
Home Customer Booking External External Bus Owner Driver Bus

Shuttle Bus Central

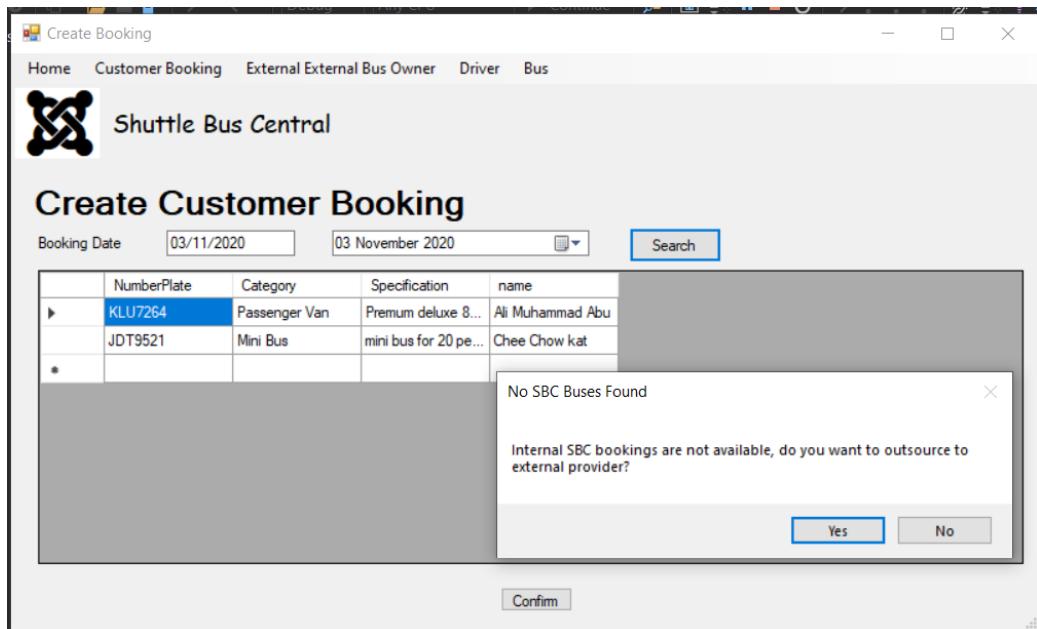
Create Customer Booking

Customer Name	andreas	No. Passengers	10
Phone Number	0176285886	Shuttle Type	School Bus
Departure	klia	Date	03/11/2020
Arrival	subang jaya	Driver	SBC john
Remarks	drop in klcc first		

OK



(Create booking page – Customer booking successful)

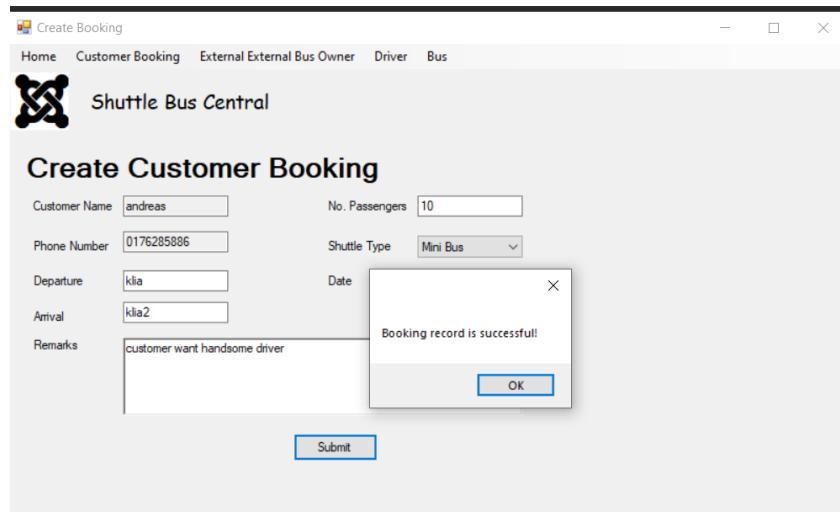


(Create booking page – Internal bus are not available alert)

After choosing the requested dates, if there is no available internal bus on the day. There will be a request alert to outsource the order to an external provider.

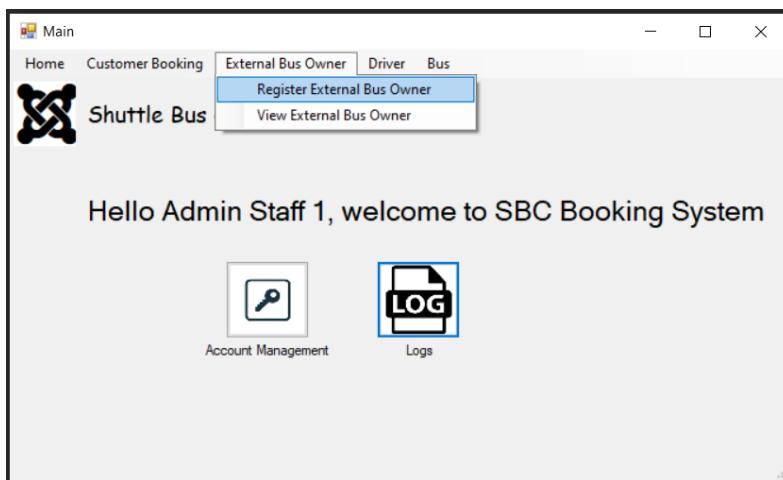
The screenshot shows the 'Create Customer Booking' interface with a filled-out form. The form fields include: Customer Name (andreas), No. Passengers (10), Phone Number (0176285886), Shuttle Type (Mini Bus), Departure (klia), Date (03/11/2020), Arrival (klia2), and Remarks (customer want handsome driver). A 'Submit' button is at the bottom right.

(Create booking page –External provider customer booking form)



(Create booking page – External provider customer booking successful)

9.1.4 Register External Bus Owner



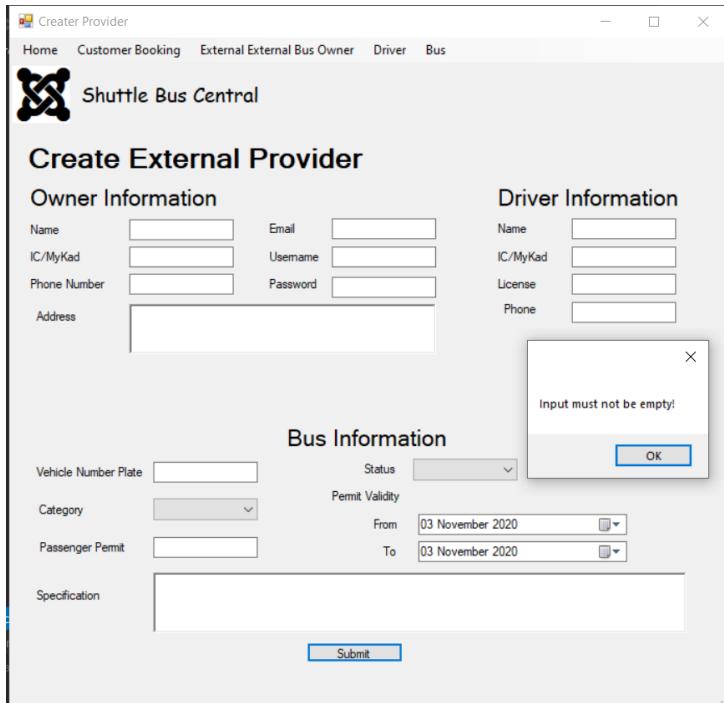
(Master Admin home page – Customer Booking tab)

A screenshot of a Windows application window titled "Create Provider". The top navigation bar includes "Home", "Customer Booking", "External External Bus Owner", "Driver", and "Bus". The "External External Bus Owner" tab is active. The main title is "Create External Provider". The form is divided into three sections: "Owner Information", "Driver Information", and "Bus Information".

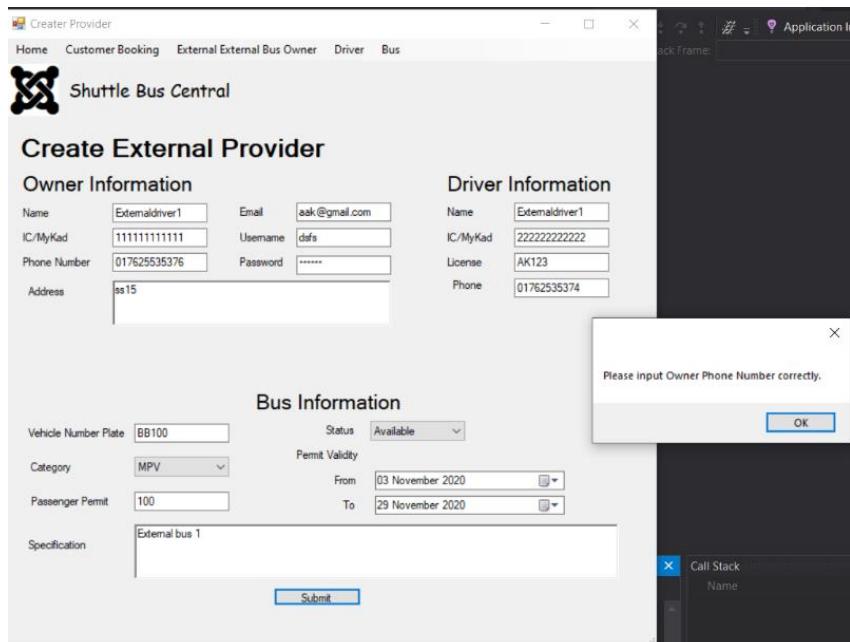
- Owner Information:** Fields include Name, Email, IC/MyKad, Username, Phone Number, Password, and Address.
- Driver Information:** Fields include Name, IC/MyKad, License, and Phone.
- Bus Information:** Fields include Vehicle Number Plate, Status (dropdown), Category (dropdown), Passenger Permit (dropdown), Permit Validity (From: 03 November 2020, To: 03 November 2020), and Specification (text area).

A "Submit" button is located at the bottom right of the form.

(Register External Bus Owner page – Create External Provider form)



(Register External Bus Owner page – Create External Provider form blank input alert)



(Register External Bus Owner page – Create External Provider form wrong input alert)

Shuttle Bus Central

Create External Provider

Owner Information

Name	Chee Chow kat.	Email	sechow@gmail.com
IC/MyKad	8212013357289	Username	Chee Chow
Phone Number	0178461886	Password	*****
Address	19, Jalan Pemberita U1/49, Kawasan Perindustrian Temasya, 40150 Shah Alam, Selangor		

Driver Information

Name	Silva
IC/MyKad	910211337728
License	L3745
Phone	0112645836

Bus Information

Vehicle Number Plate	JDT9521	Status	Available
Category	Mini Bus	Permit Validity	
Passenger Permit	P2364	From	01 November 2020
Specification	mini bus for 20 people with toilet		

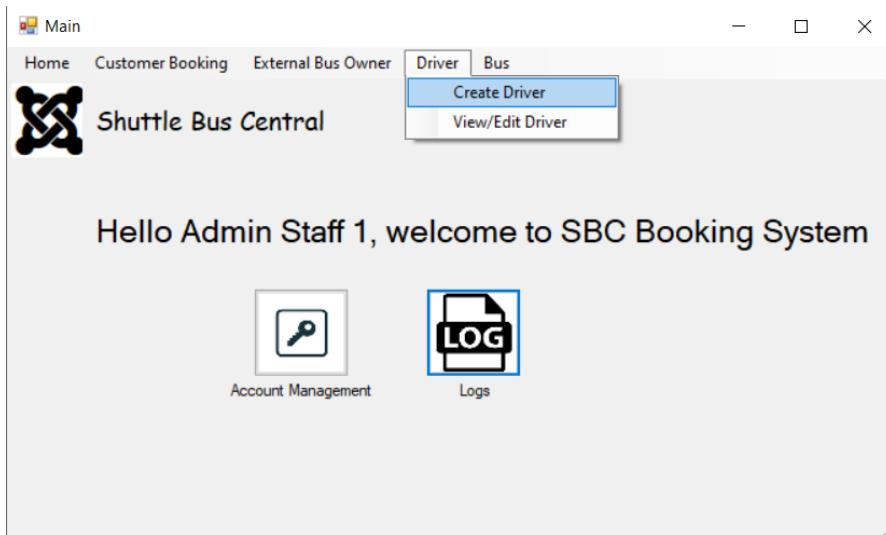
External provided added!

OK

Submit

(Register External Bus Owner page – Create External Provider form success)

9.1.5 Create Driver



(Master Admin home page – Create driver tab)

The screenshot shows a Windows application window titled "Create Driver". The menu bar includes "Home", "Customer Booking", "External External Bus Owner", "Driver", and "Bus". The "Driver" menu is open, showing "Create Driver" and "View/Edit Driver", with "Create Driver" highlighted. The main content area is titled "Create Driver". It contains fields for "Driver Name" (input field), "Owner" (dropdown menu), "IC/MyKad" (input field), "Phone Number" (input field), and "License" (input field). A "Submit" button is at the bottom.

(Create Drive page – Create driver form)

The screenshot shows the 'Create Driver' page. The 'Owner' dropdown menu is open, displaying 'SBC' at the top, followed by three external drivers: 'Ali Muhammad Abu', 'Chee Chow kat', and 'SBC' again. The rest of the form contains fields for Driver Name ('andy'), IC/MyKad ('980511335799'), Phone Number ('0175243775'), and License ('L7464'). A 'Submit' button is visible at the bottom.

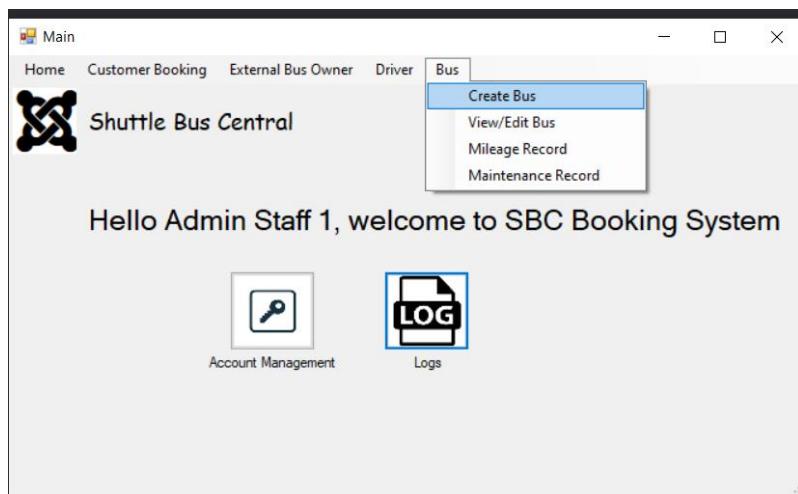
(Create Drive page – Choose driver Internal/External)

The screenshot shows the 'Create Driver' page. A confirmation dialog box is displayed in the center-right, stating 'Driver has been added!' with an 'OK' button. The rest of the form contains fields for Driver Name ('andy'), IC/MyKad ('980511335799'), Phone Number ('0175243775'), and License ('L7464'). A 'Submit' button is visible at the bottom.

(Create Drive page – Driver added)

Driver added as an Internal driver if Owner under SBC, Driver added as an External driver if under another name

9.1.6 Create Bus



(Master Admin home page – Create bus tab)

The screenshot shows the 'Create Bus' page. The title bar says 'Create Bus'. The page includes a navigation bar with Home, Customer Booking, External Bus Owner, Driver, and Bus. The main content area is titled 'Create Bus' and contains the following fields:

Number Plate	Status
<input type="text"/>	<input type="text"/>
Category	Permit Validity
<input type="text"/>	From: 03 November 2020 <input type="button"/>
<input type="text"/>	To: 03 November 2020 <input type="button"/>
Owner Name	
Specification	<input type="text"/>

A 'Submit' button is located at the bottom right.

(Create Bus page – Creating bus form)

The screenshot shows the 'Create Bus' page with the title 'Shuttle Bus Central'. The 'Category' dropdown menu is open, displaying options: Mini Bus, Factory Bus, School Bus, Express Bus, Coach, Passenger Van, and MPV. Other fields include Number Plate (BL2423), Status (Available), Permit Validity (From: 03 November 2020, To: 03 November 2020), and Specification (empty). A 'Submit' button is at the bottom.

(Create Bus page – Bus category)

The screenshot shows the 'Create Bus' page with the title 'Shuttle Bus Central'. The 'Owner Name' dropdown menu is open, displaying options: Ali Muhammad /, Ali Muhammad Abu, Chee Chow kat, and SBC. Other fields include Number Plate (BL2423), Status (empty), Category (Mini Bus), Permit Validity (From: 05 November 2020, To: 05 November 2020), and Specification (empty). A 'Submit' button is at the bottom.

(Create Bus page – Bus Internal/External Owner)

Create Bus

Home Customer Booking External Bus Owner Driver Bus

Shuttle Bus Central

Create Bus

Number Plate	BL2423	Status	Available																																																	
Category	Mini Bus	Permit Validity																																																		
Passenger Permit	P1293	From	03 November 2020																																																	
Owner Name	Ali Muhammad /	To	<table border="1"><tr><td>Mon</td><td>Tue</td><td>Wed</td><td>Thu</td><td>Fri</td><td>Sat</td><td>Sun</td></tr><tr><td>26</td><td>27</td><td>28</td><td>29</td><td>30</td><td>31</td><td>1</td></tr><tr><td>2</td><td>3</td><td>4</td><td>5</td><td>6</td><td>7</td><td>8</td></tr><tr><td>9</td><td>10</td><td>11</td><td>12</td><td>13</td><td>14</td><td>15</td></tr><tr><td>16</td><td>17</td><td>18</td><td>19</td><td>20</td><td>21</td><td>22</td></tr><tr><td>23</td><td>24</td><td>25</td><td>26</td><td>27</td><td>28</td><td>29</td></tr><tr><td>30</td><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td><td>6</td></tr></table>	Mon	Tue	Wed	Thu	Fri	Sat	Sun	26	27	28	29	30	31	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	1	2	3	4	5	6
Mon	Tue	Wed	Thu	Fri	Sat	Sun																																														
26	27	28	29	30	31	1																																														
2	3	4	5	6	7	8																																														
9	10	11	12	13	14	15																																														
16	17	18	19	20	21	22																																														
23	24	25	26	27	28	29																																														
30	1	2	3	4	5	6																																														
Specification	Big brwon 14 seater mini bus	Submit	Today: 05/11/2020																																																	

(Create Bus page – Bus Permit validity)

Create Bus

Home Customer Booking External Bus Owner Driver Bus

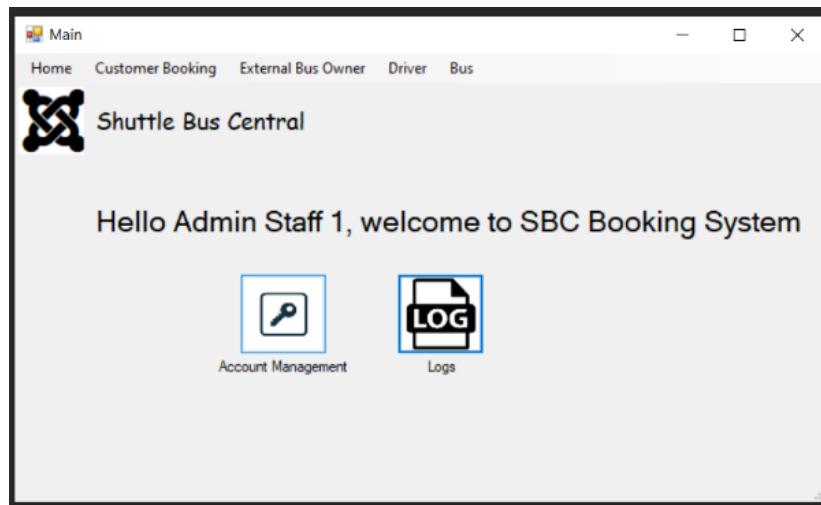
Shuttle Bus Central

Create Bus

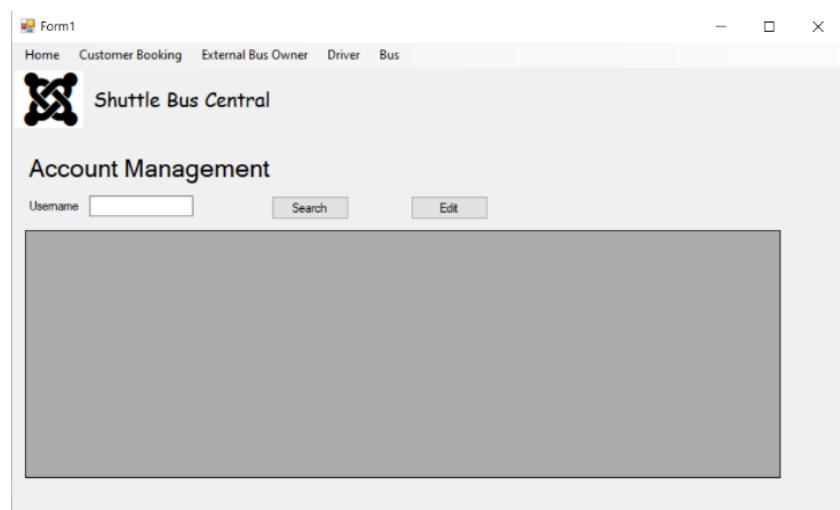
Number Plate	BL2423	Status	Available
Category	Mini Bus	Permit Validity	
Passenger Permit	P1293	From	03 Nov
Owner Name	Ali Muhammad /	To	21 Nov
Specification	Big brwon 14 seater mini bus	Submit	OK

(Create Bus page – Bus has been added)

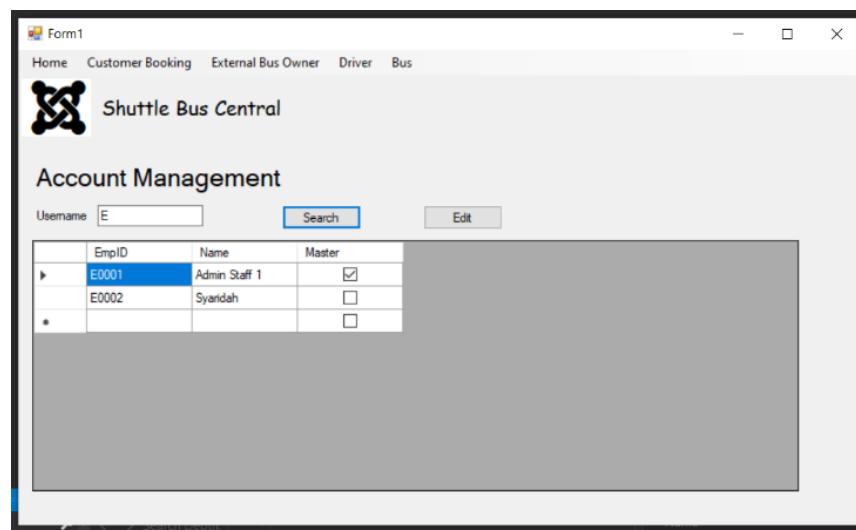
9.1.7 Account Management



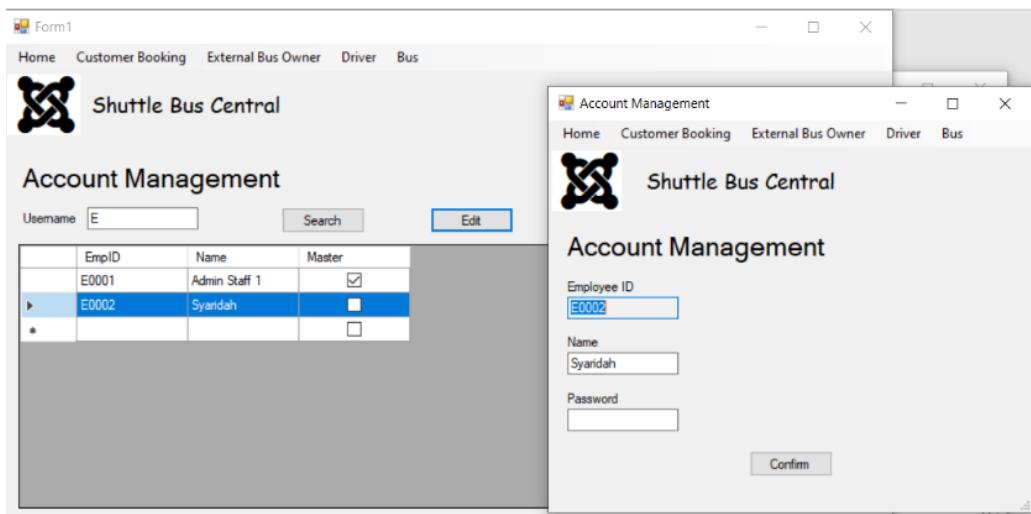
(Master Admin home page – Account Management)



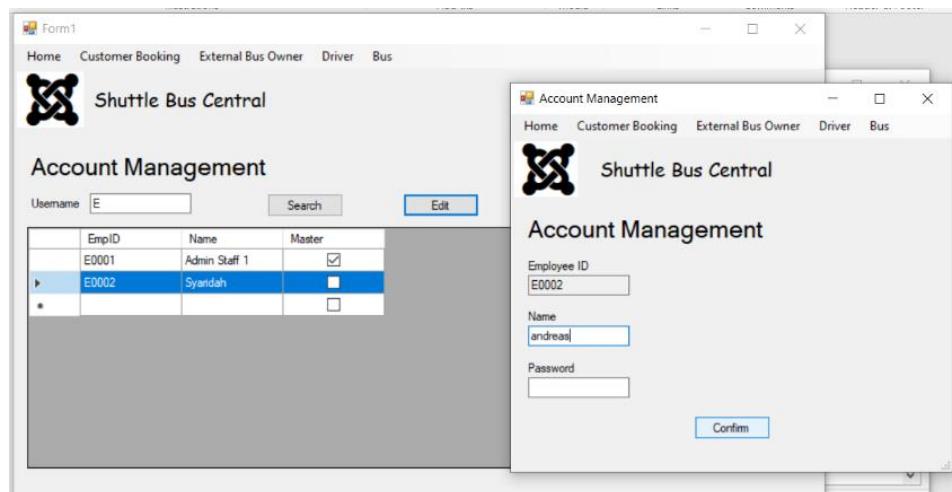
(Account Management page – Account Management home page)



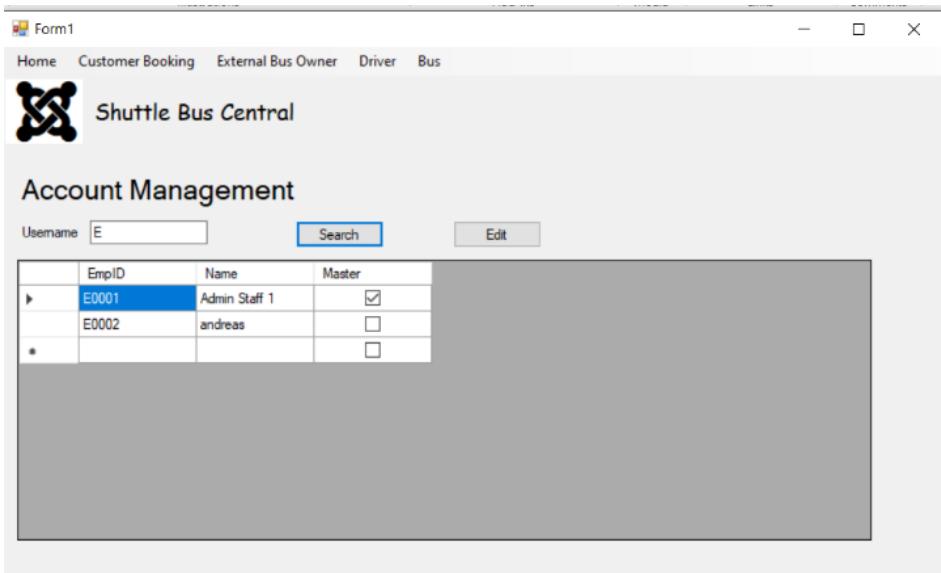
(Account Management page – Account search)



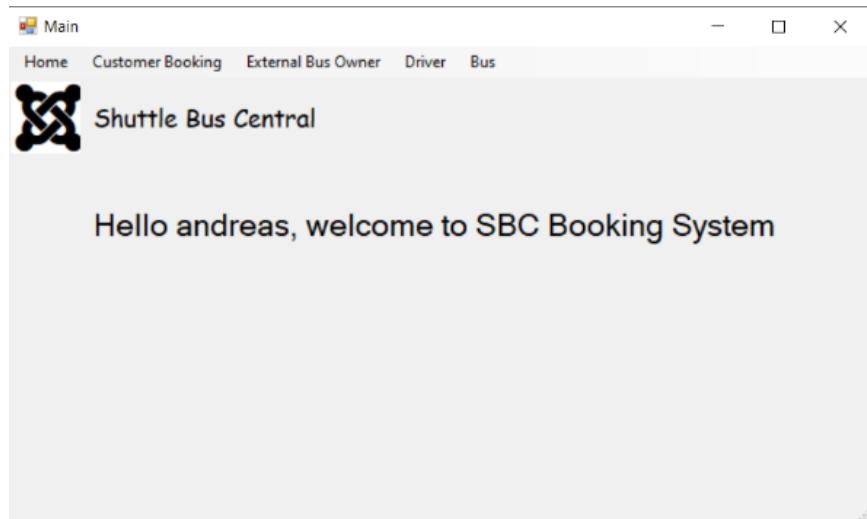
(Account Management page –Select account to edit)



(Account Management page – Edit Admin name)

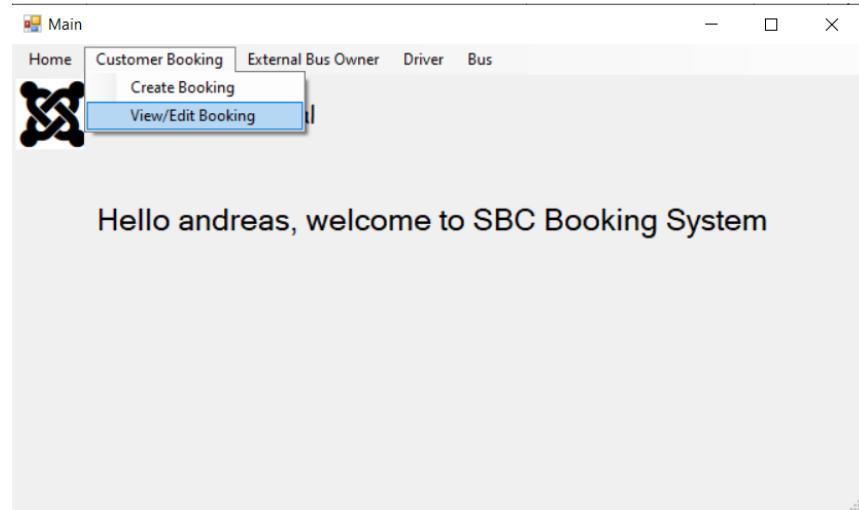


(Account Management page – Admin name changed)

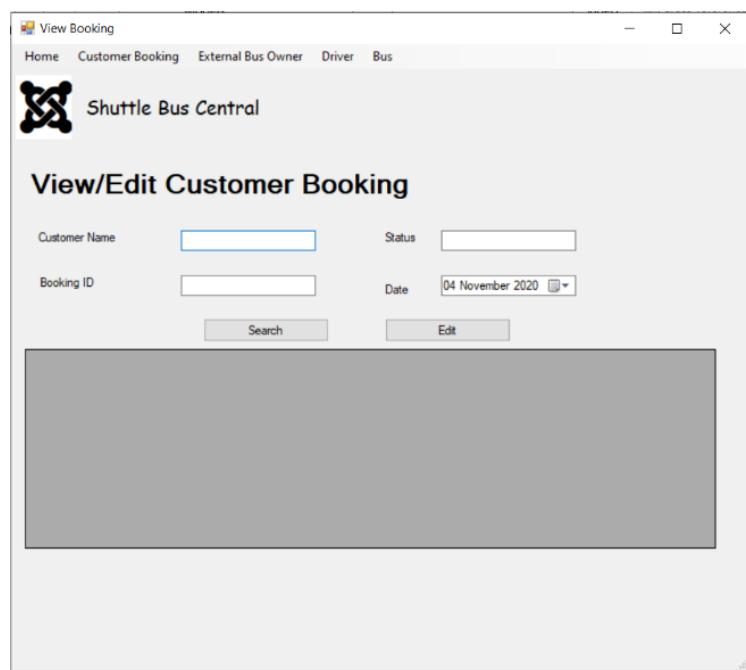


(Account Management page – Admin home page name changed)

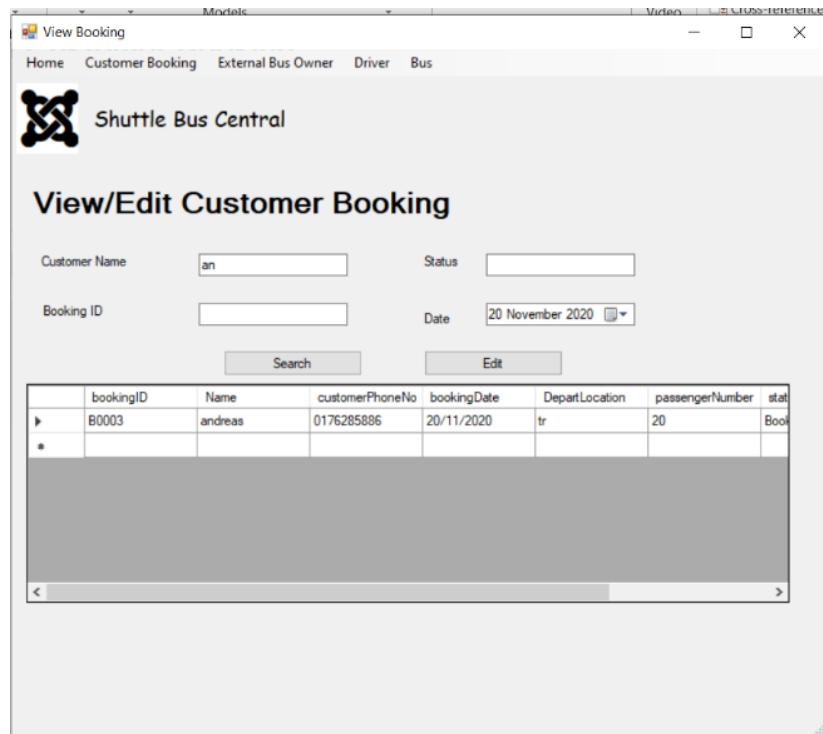
9.1.8 View/Edit Customer Booking



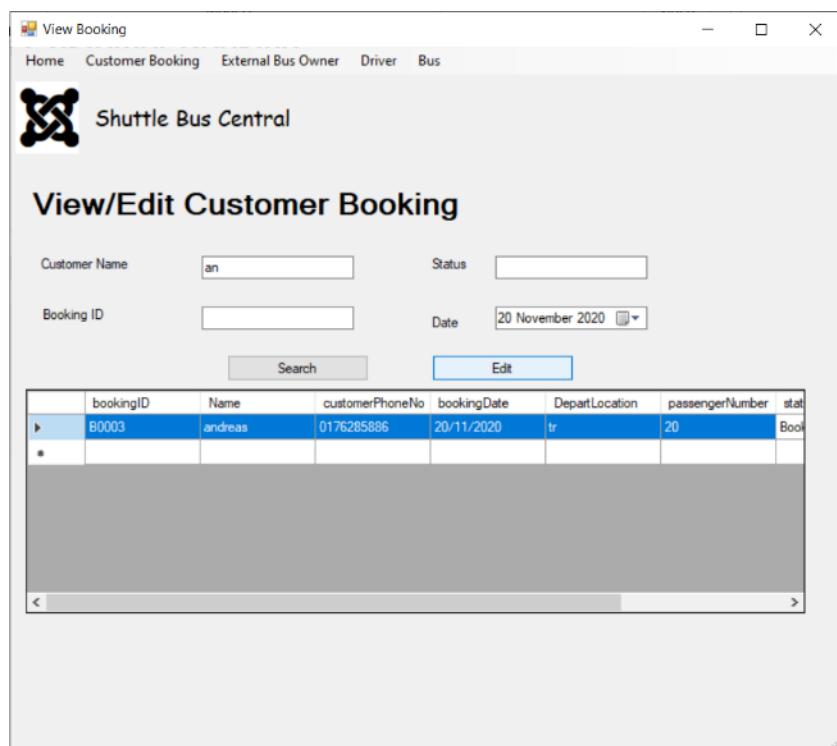
(Master Admin home page – View/Edit Booking)



(View/Edit Customer Booking page)



(View/Edit Customer Booking page – Search booking)



(View/Edit Customer Booking page – Select booking)

Edit Booking

Home Customer Booking External Bus Owner Driver Bus

Shuttle Bus Central

View/Edit Customer Booking

Booking ID	B0003	Status	<input type="text"/>
Customer Name	0176285886	Passenger Number	<input type="text"/>
Booking Date	20 November 2020	Remarks	<input type="text"/>
Departure	tr		
Arrival	ash		

Submit

(View/Edit Customer Booking page – Edit status)

Edit Booking

Home Customer Booking External Bus Owner Driver Bus

Shuttle Bus Central

View/Edit Customer Booking

Booking ID	B0003	Status	<input type="text"/> Booked but not paid
Customer Name	0176285886	Passenger Number	<input type="text"/> 20
Booking Date	20 November 2020	Remarks	<input type="text"/> 242
Departure	tr		
Arrival	ash		

Submit

Booking info changed!

OK

(View/Edit Customer Booking page – Booking info changed alert)

View Booking

Home Customer Booking External Bus Owner Driver Bus

Shuttle Bus Central

View/Edit Customer Booking

Customer Name	<input type="text"/> an	Status	<input type="text"/>																					
Booking ID	<input type="text"/>	Date	20 November 2020 <input type="button"/>																					
<input type="button"/> Search <input type="button"/> Edit																								
<table border="1"> <thead> <tr> <th></th> <th>customerPhoneNo</th> <th>bookingDate</th> <th>DepartLocation</th> <th>passengerNumber</th> <th>status</th> <th>remarks</th> </tr> </thead> <tbody> <tr> <td>></td> <td>0176285886</td> <td>20/11/2020</td> <td>tr</td> <td>20</td> <td>Booked but not p...</td> <td>242</td> </tr> <tr> <td>*</td> <td></td> <td></td> <td></td> <td></td> <td></td> <td>Booked but not paid</td> </tr> </tbody> </table>					customerPhoneNo	bookingDate	DepartLocation	passengerNumber	status	remarks	>	0176285886	20/11/2020	tr	20	Booked but not p...	242	*						Booked but not paid
	customerPhoneNo	bookingDate	DepartLocation	passengerNumber	status	remarks																		
>	0176285886	20/11/2020	tr	20	Booked but not p...	242																		
*						Booked but not paid																		

(View/Edit Customer Booking page – Booking info changed result)

Edit Booking

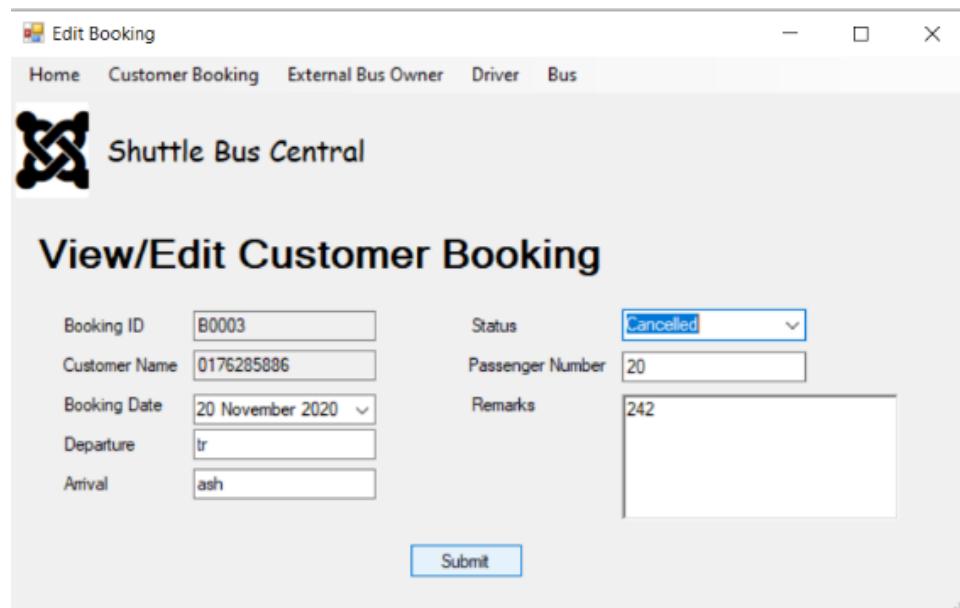
Home Customer Booking External Bus Owner Driver Bus

Shuttle Bus Central

View/Edit Customer Booking

Booking ID	B0003	Status	Cancelled
Customer Name	0176285886	Passenger Number	20
Booking Date	20 November 2020	Remarks	242
Departure	tr		
Arrival	ash		

Submit



(View/Edit Customer Booking page – Booking status cancelled)

Edit Booking

Home Customer Booking External Bus Owner Driver Bus

Shuttle Bus Central

View/Edit Customer Booking

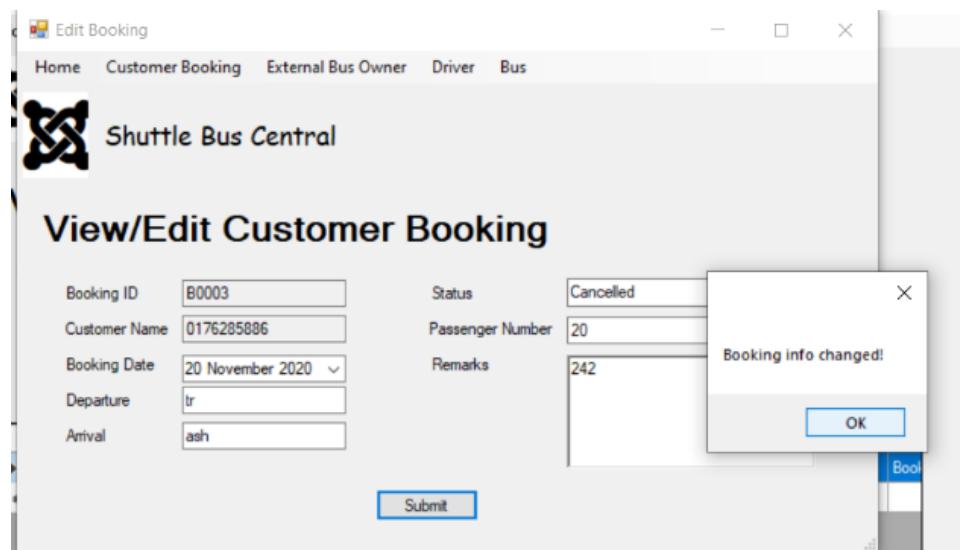
Booking ID	B0003	Status	Cancelled
Customer Name	0176285886	Passenger Number	20
Booking Date	20 November 2020	Remarks	242
Departure	tr		
Arrival	ash		

Booking info changed!

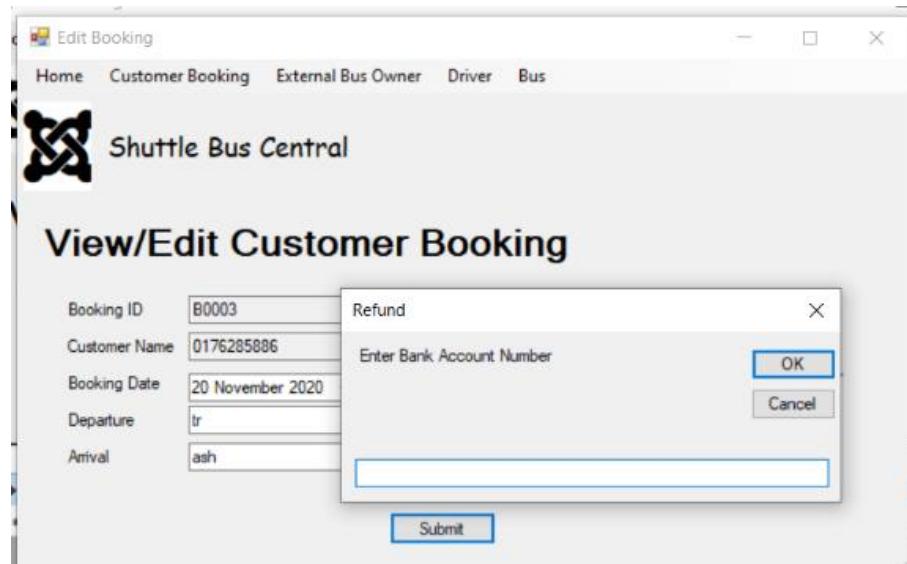
OK

Cancel

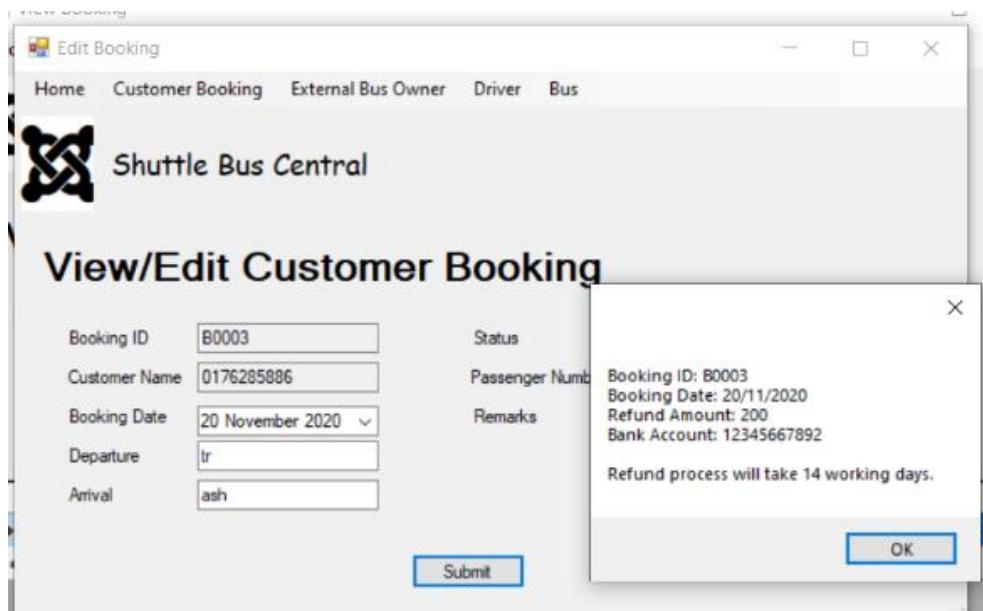
Submit



(View/Edit Customer Booking page – Booking status changed to cancelled)

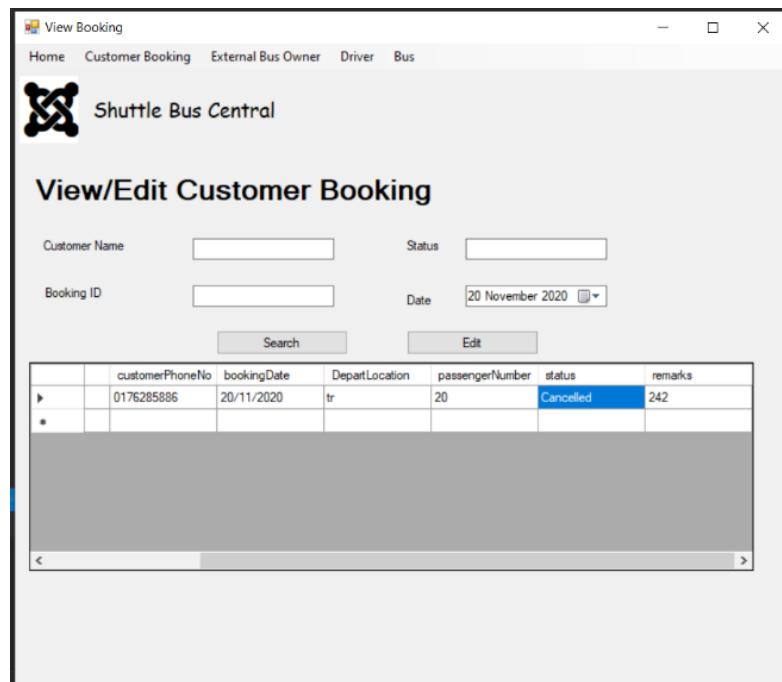


(View/Edit Customer Booking page – Refund process initiated)



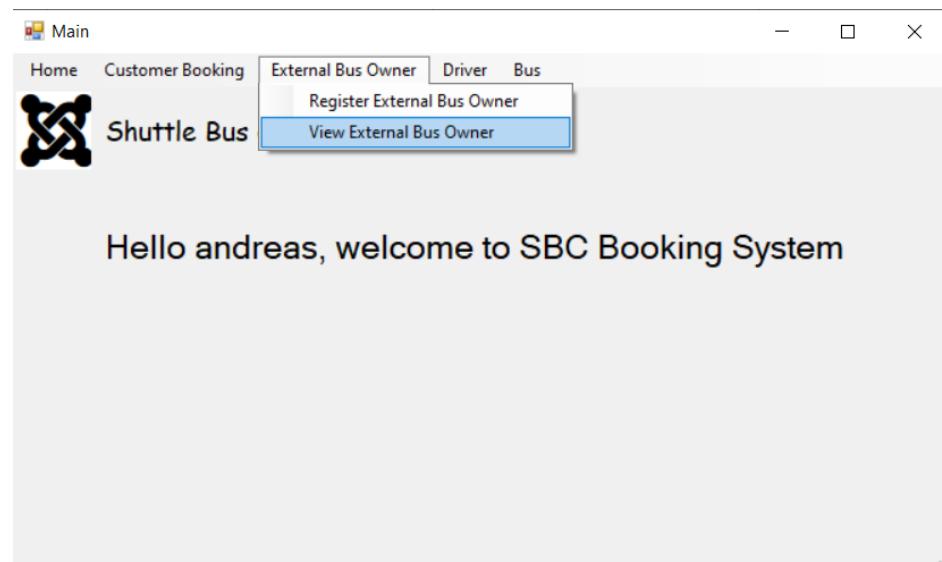
(View/Edit Customer Booking page – Refund details)

The refund amount will increase based on the days after booking

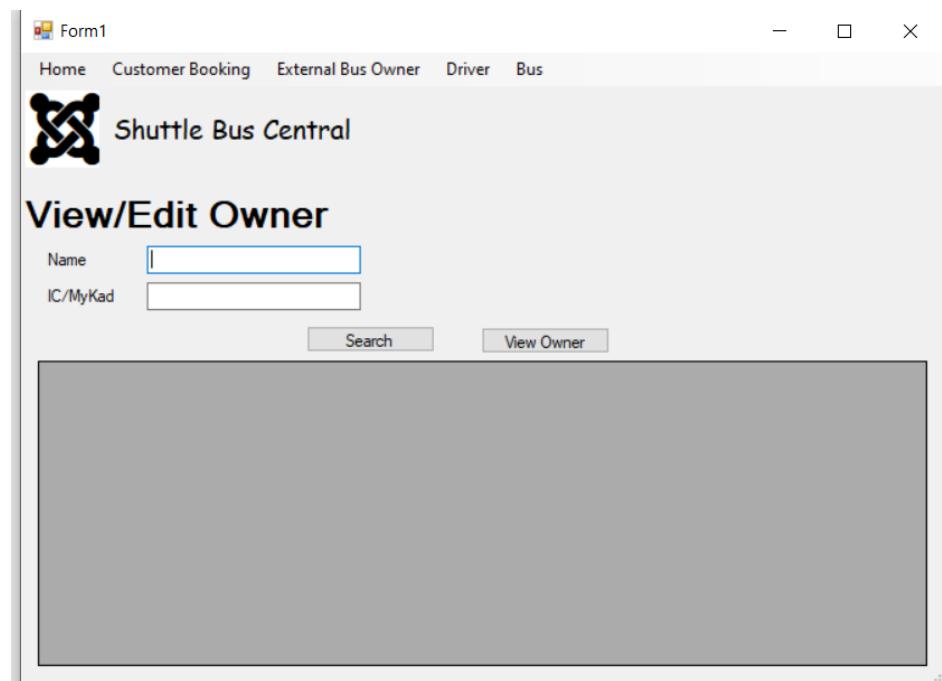


(View/Edit Customer Booking page – Status cancelled)

9.1.9 View/Edit External Bus Owner



(Admin home page – View/Edit External Bus Owner tab)



(View/Edit External Bus Owner page)

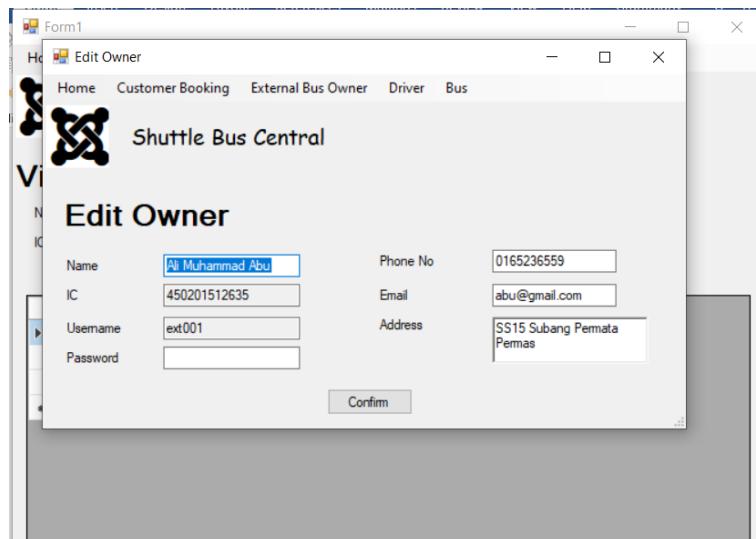
The screenshot shows a Windows application window titled 'Form1'. The menu bar includes 'Home', 'Customer Booking', 'External Bus Owner', 'Driver', and 'Bus'. The title bar features the 'Shuttle Bus Central' logo and the text 'View/Edit Owner'. Below the title bar are two input fields: 'Name' and 'IC/MyKad', each with an associated text input box. Underneath these fields are two buttons: 'Search' and 'View Owner'. A data grid table follows, displaying four rows of data:

	IC	Name	Address	Phone	Email
▶	450201512635	Ali Muhammad Abu	SS15 Subang Pe...	0165236559	abu@gmail.com
	821201335728	Chee Chow kat	19, Jalan Pember...	0178461886	cheechow@gmai...
*	990102065522	SBC	Jalan Klang	0125526633	SBC@internalSB...

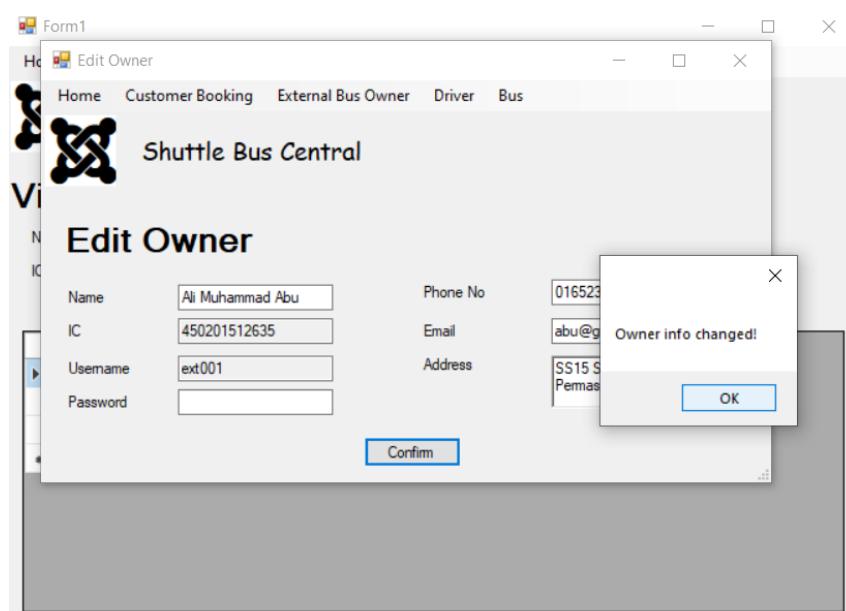
(View/Edit External Bus Owner page – Search External Bus Owner)

This screenshot is identical to the one above, showing the 'View/Edit Owner' page. The difference is that the first row of the data grid is now highlighted with a blue background, indicating it is selected.

(View/Edit External Bus Owner page –Select External Bus Owner)

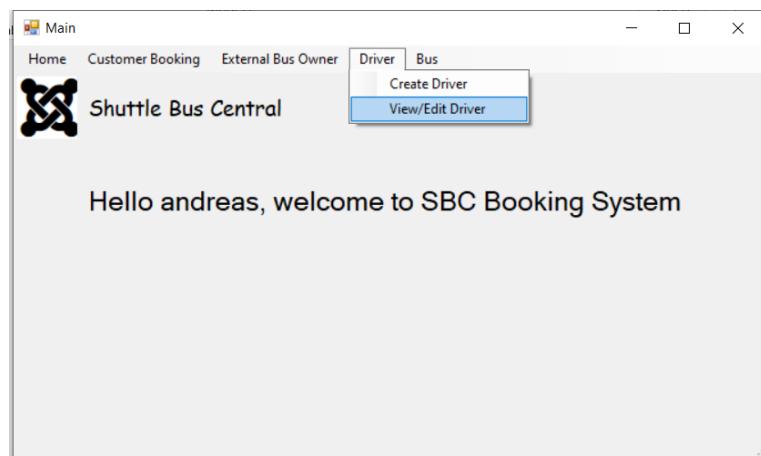


(View/Edit External Bus Owner page – Edit External Bus Owner)

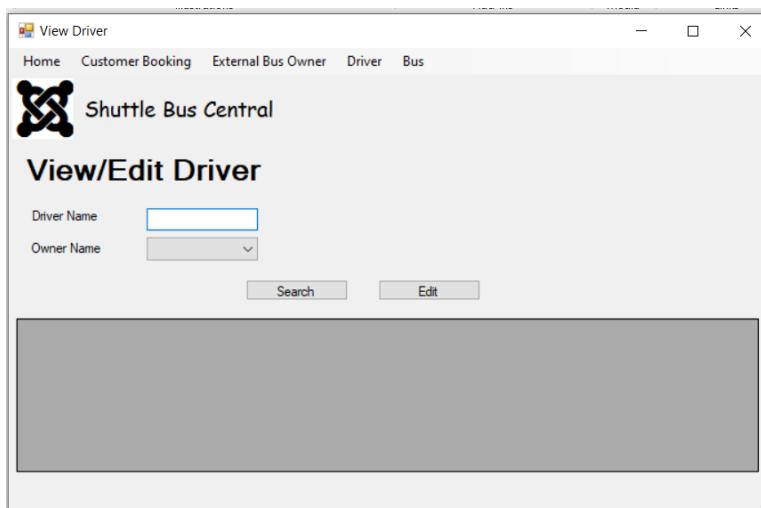


(View/Edit External Bus Owner page – External Bus Owner info changed)

9.1.10 View/Edit Driver



(Figure 9.4.4.1 Admin home page – View/Edit Driver tab)

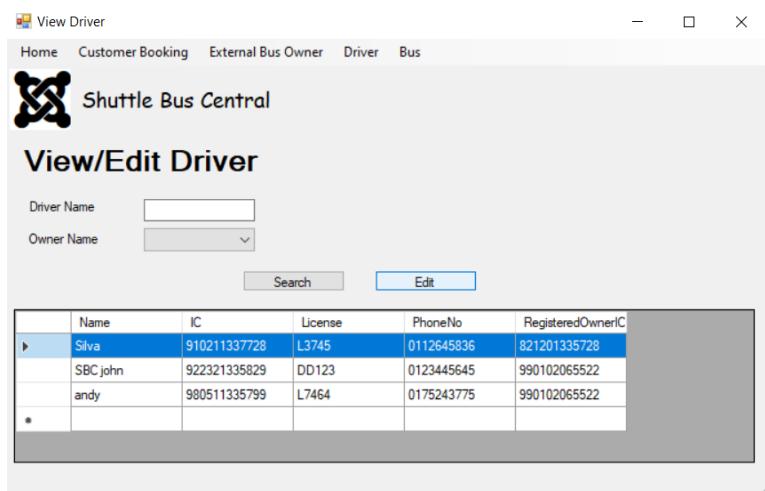


(View/Edit Driver page)

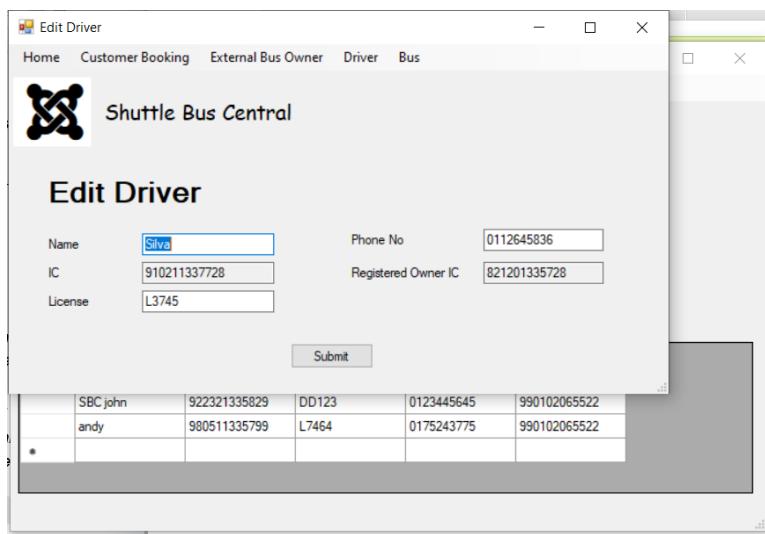
A screenshot of a Windows application window titled "View Driver". The menu bar includes "Home", "Customer Booking", "External Bus Owner", "Driver", and "Bus". The "Driver" menu item is highlighted. The main content area has a title "View/Edit Driver" and contains two search fields: "Driver Name" (with an empty input field) and "Owner Name" (with a dropdown menu). Below these fields are "Search" and "Edit" buttons. A table displays search results for drivers. The table has columns: Name, IC, License, PhoneNo, and RegisteredOwnerID. The rows show the following data:

	Name	IC	License	PhoneNo	RegisteredOwnerID
▶	Silva	910211337728	L3745	0112645836	821201335728
	SBC john	922321335829	DD123	0123445645	990102065522
*	andy	980511335799	L7464	0175243775	990102065522

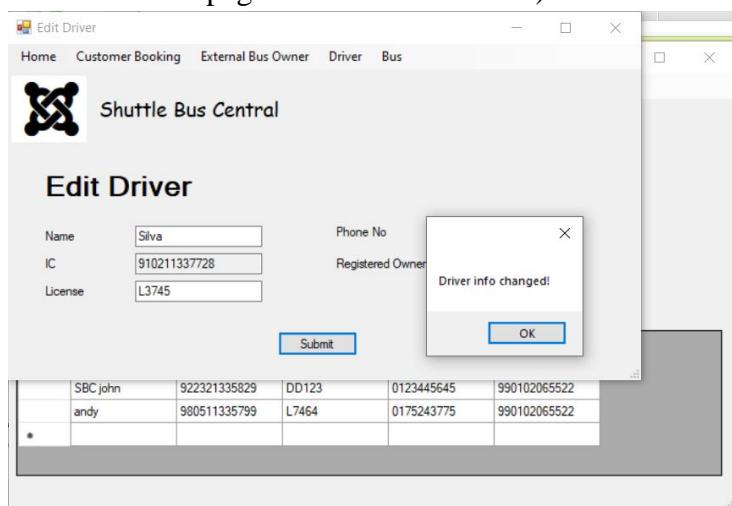
(View/Edit Driver page – Drivers search)



(View/Edit Driver page – Select Driver)

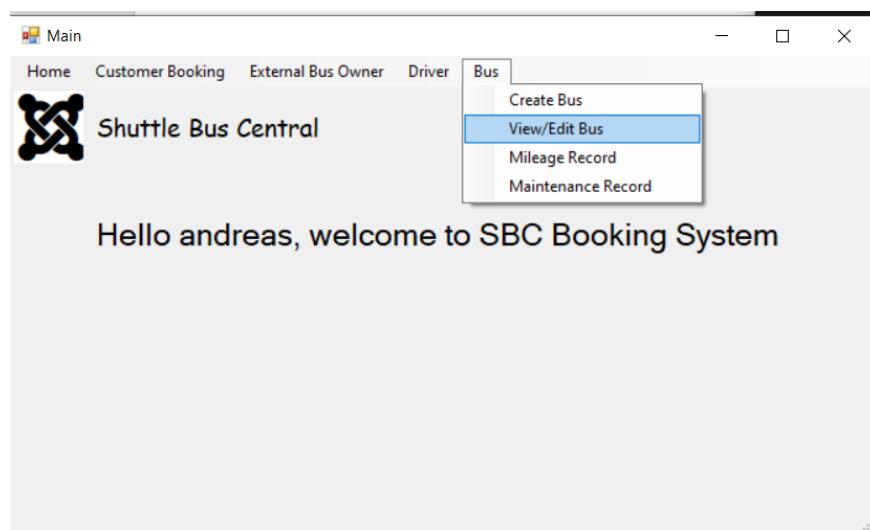


(View/Edit Driver page – Edit Driver details)

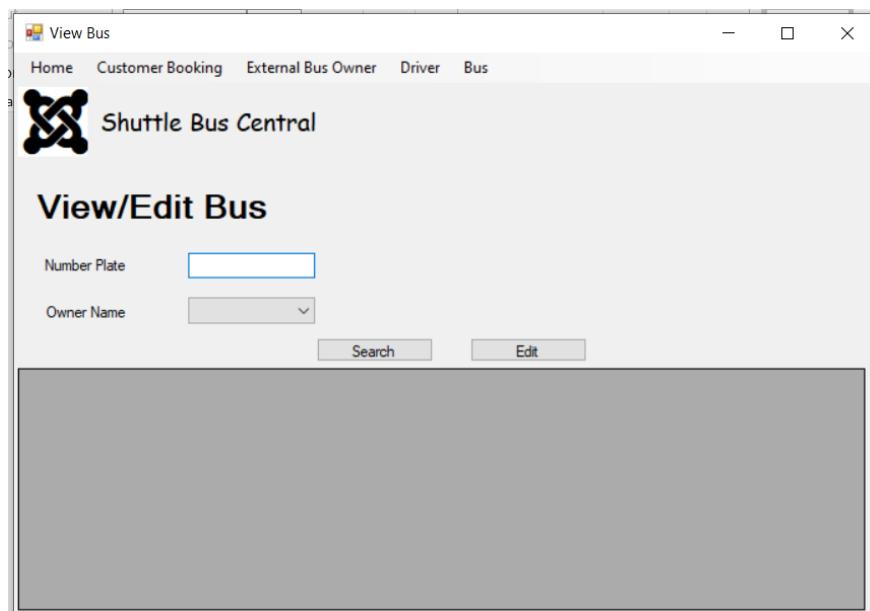


(View/Edit Driver page –Driver info changed)

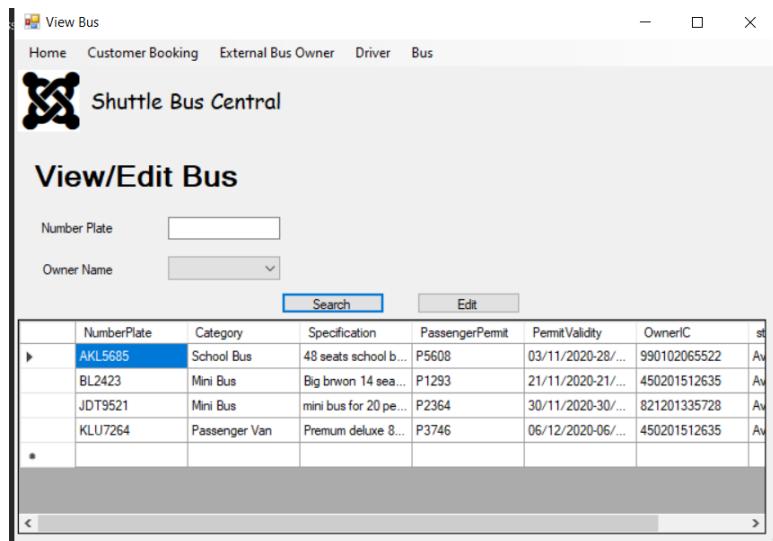
9.1.11 View/Edit Bus



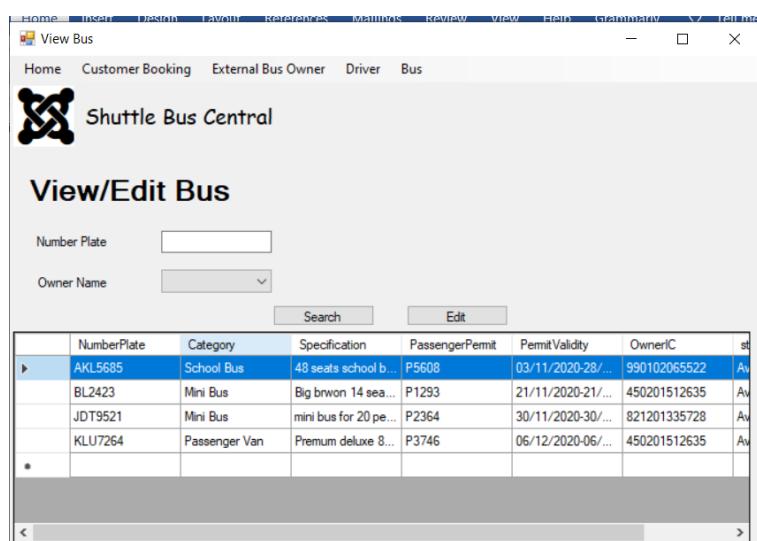
(Admin home page – View/Edit Bus tab)



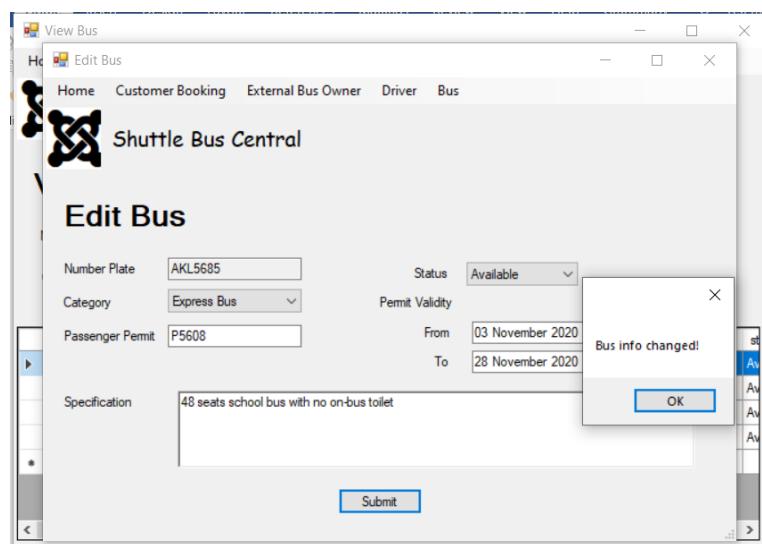
(View/Edit Bus page)



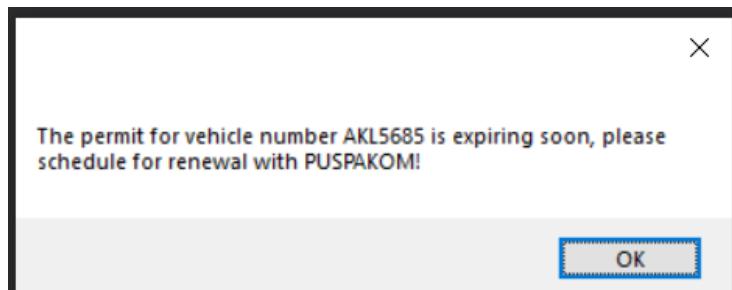
(View/Edit Bus page – Search bus)



(View/Edit Bus page – Edit bus)



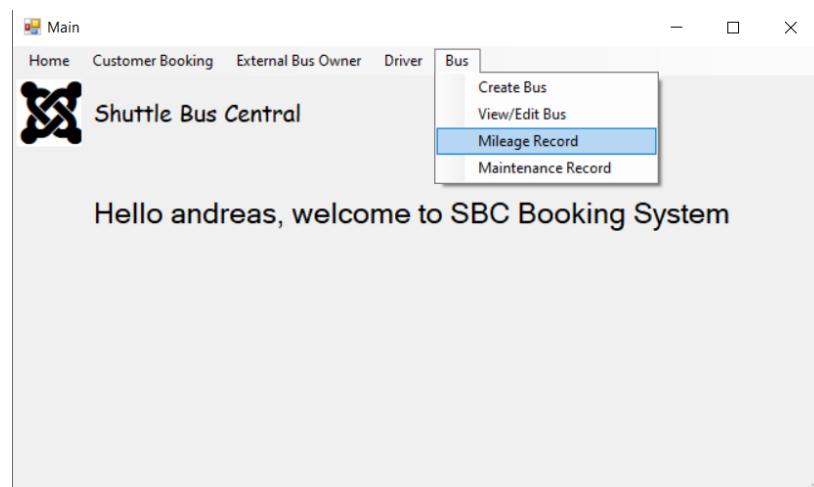
(View/Edit Bus page –Bus info changed alert)



(View/Edit Bus page –Bus Permit Validity expiring)

When the bus permit validity is less than a week, there will be an alert message whenever Admin logged into the system.

9.1.12 Mileage Record



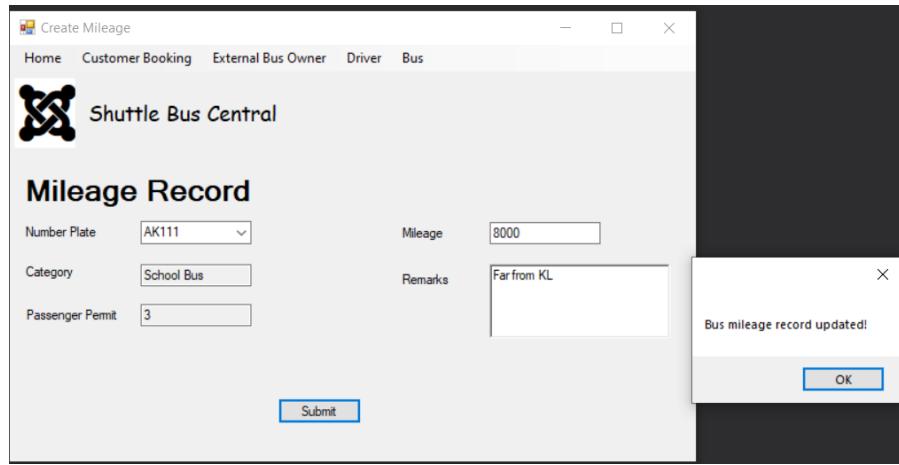
(Admin home page –Mileage Record tab)

A screenshot of a window titled "Create Mileage". The window has a menu bar with "Home", "Customer Booking", "External Bus Owner", "Driver", and "Bus". A dropdown menu for "Bus" is open, showing "Create Bus", "View/Edit Bus", "Mileage Record" (highlighted), and "Maintenance Record". The main area is titled "Mileage Record" and contains fields for "Number Plate" (dropdown), "Mileage" (text input), "Category" (dropdown), "Remarks" (text area), and "Passenger Permit" (text input). A "Submit" button is at the bottom.

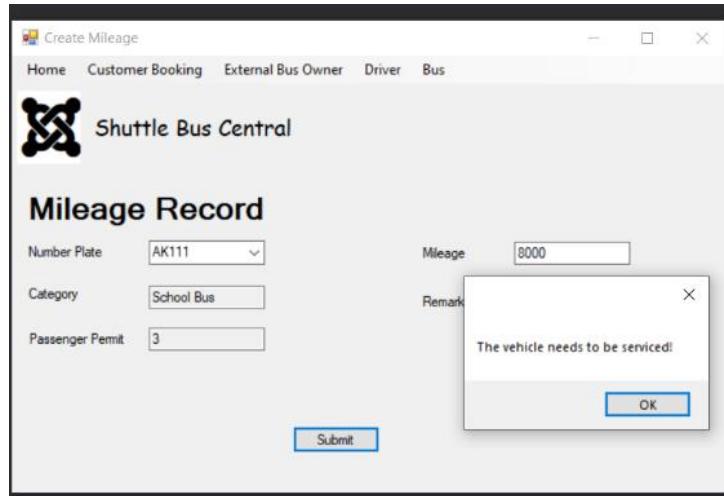
(View/Edit Bus tab)

A screenshot of the "Create Mileage" window with the "Mileage Record" tab selected. The "Number Plate" field contains "AK111", the "Mileage" field contains "8000", the "Category" field contains "School Bus", the "Remarks" field contains "Far from KL", and the "Passenger Permit" field contains "3". A "Submit" button is at the bottom.

(View/Edit Bus tab – Fill-in details)

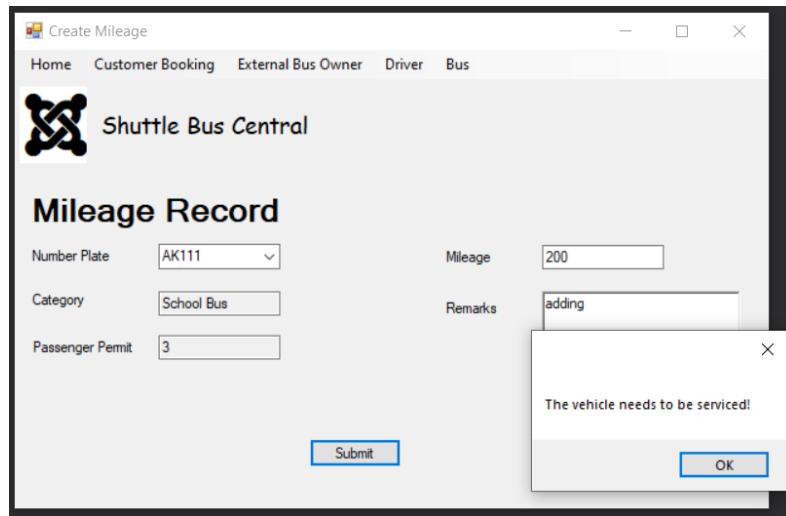


(View/Edit Bus tab –Bus Mileage Record updated)



(View/Edit Bus tab – Service alert)

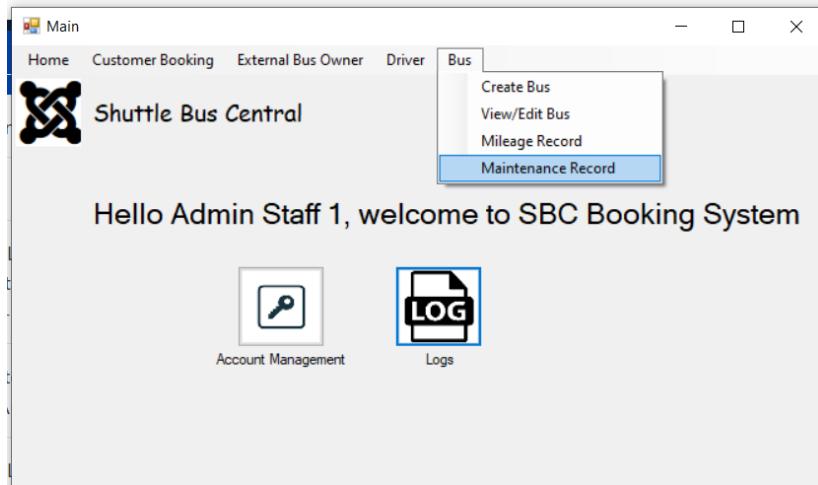
If the Bus mileage is above 8000, there will be alert to remind the bus need to be serviced



(View/Edit Bus tab – Service alert cumulative)

The Bus mileage is cumulative, any additional mileage that brings the total over 7000 will trigger the alert message to service the bus.

9.1.13 Maintenance Record



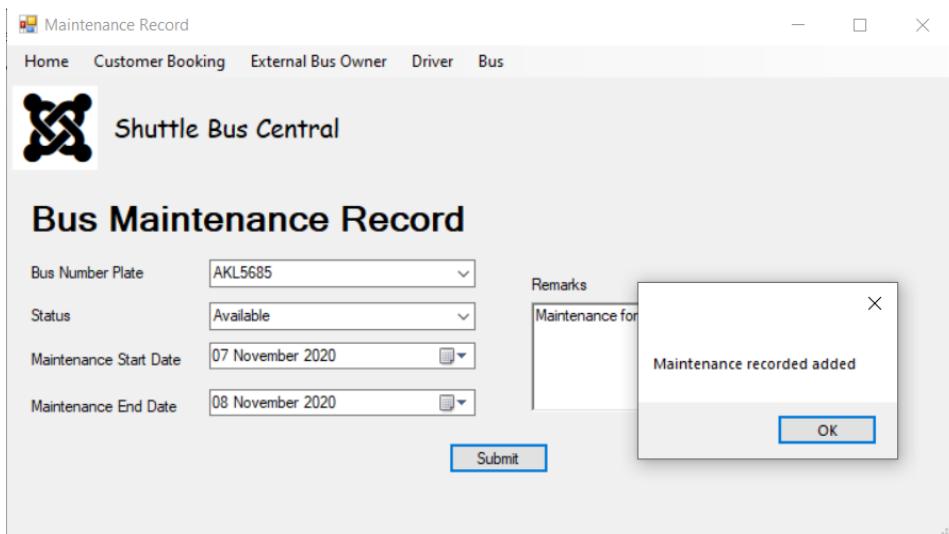
(Admin home page –Maintenance Record tab)

The screenshot shows the 'Maintenance Record' window. The title bar says 'Maintenance Record'. The main content area is titled 'Bus Maintenance Record'. It contains four input fields: 'Bus Number Plate' (dropdown), 'Status' (dropdown), 'Maintenance Start Date' (date picker set to 04 November 2020), and 'Maintenance End Date' (date picker set to 04 November 2020). To the right of these fields is a 'Remarks' text area. A 'Submit' button is located at the bottom right of the form.

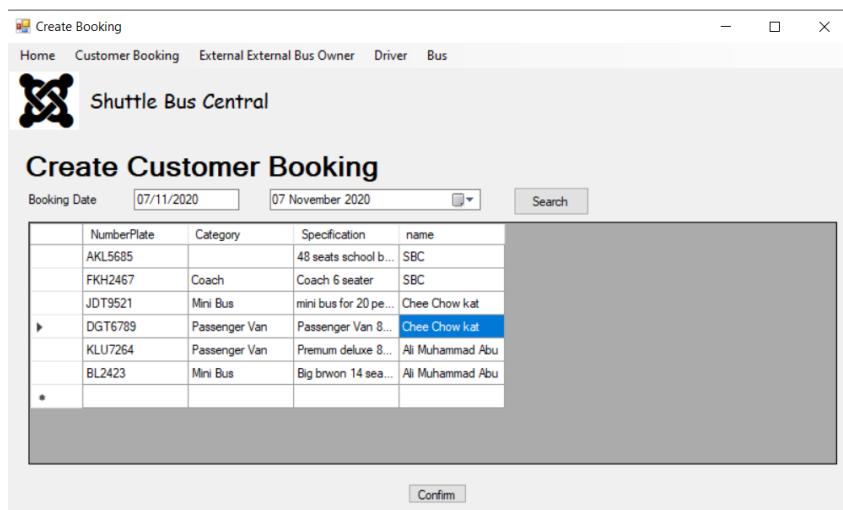
(Maintenance Record page)

The screenshot shows the 'Maintenance Record' window again. The title bar says 'Maintenance Record'. The main content area is titled 'Bus Maintenance Record'. The input fields are now populated: 'Bus Number Plate' is set to 'AKL5685', 'Status' is set to 'Available', 'Maintenance Start Date' is set to '07 November 2020', and 'Maintenance End Date' is set to '08 November 2020'. The 'Remarks' text area contains the text 'Maintenance for engine'. A 'Submit' button is located at the bottom right of the form.

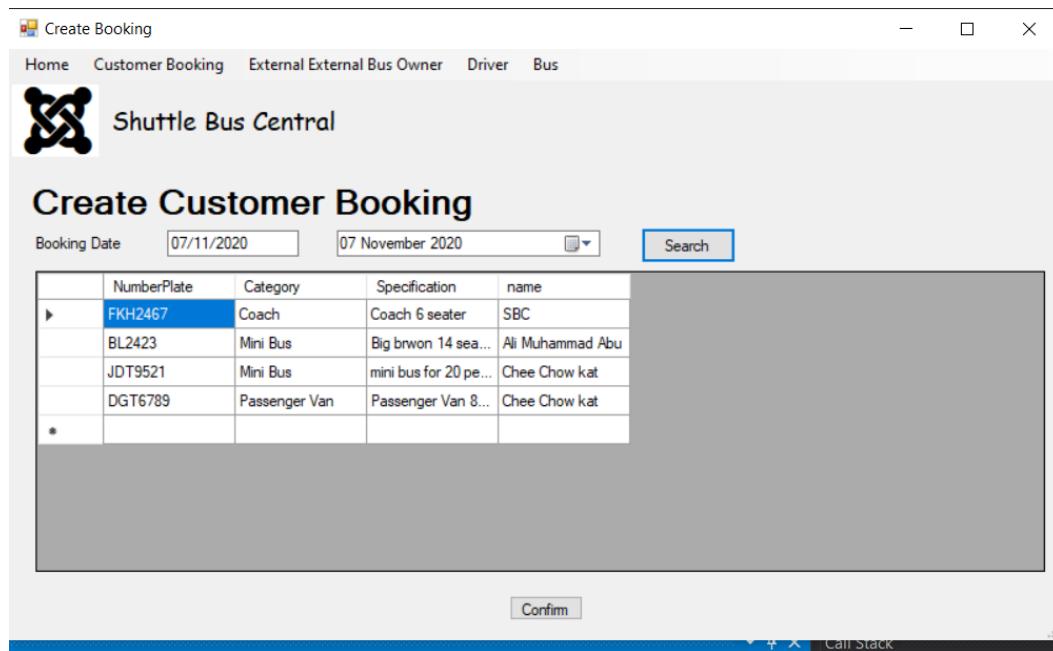
(Maintenance Record page – Bus Maintenance details)



(Maintenance Record page – Bus Maintenance record added)



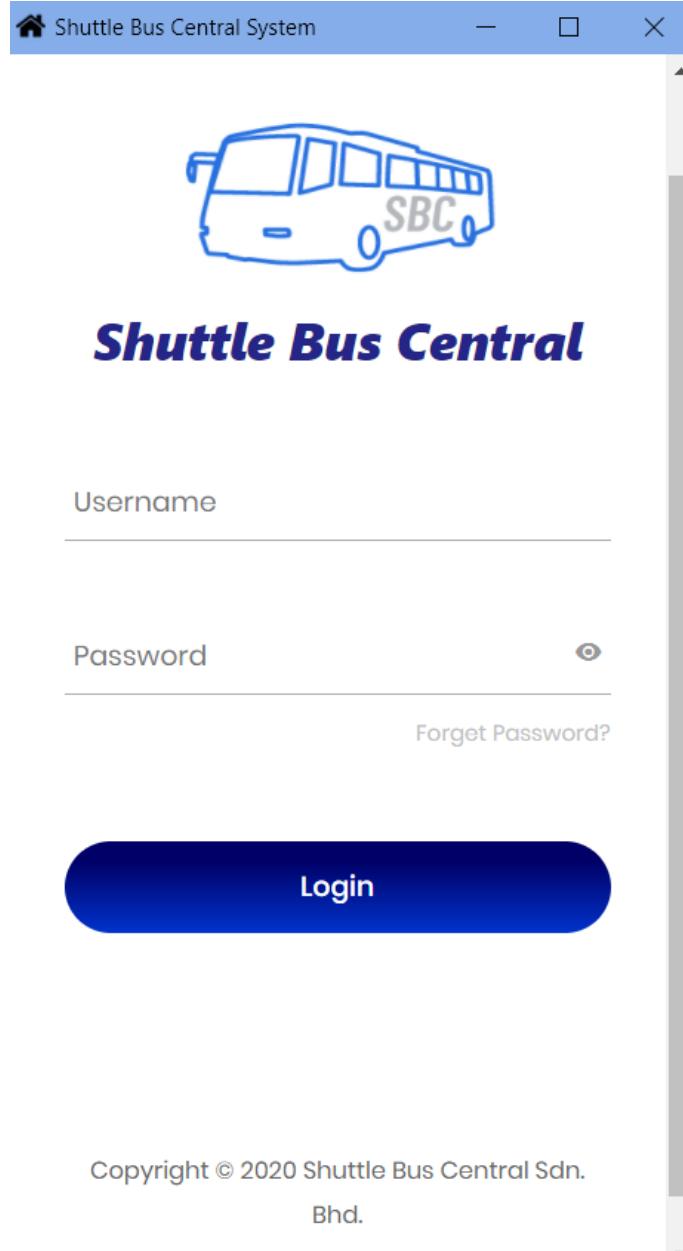
(Customer Booking page– Bus still available before maintenance added)
AKL5685 bus is still available



(Customer Booking page– Bus unavailable after maintenance added)
AKL5685 bus is not available, under maintenance

9.2 Mobile Web Application

9.2.1 Login Page



(SBC Mobile Web Login Page)

External bus provider may access to this login page with dedicated username and password given by SBC.

9.2.2 Home Page

The screenshot shows a web browser window titled "Shuttle Bus Central System". The header includes a user greeting "Hi, jimmy" and a menu icon. The main content features a banner with the text "Central Sdn. Bhd." and "We Provide The Best Service" over an image of a bus interior. Below the banner is a notification message: "!" Notification ! "Booking B0003 info has changed." A blue button labeled "Marked As Read" is visible. At the bottom, copyright information reads "Copyright © 2020 Shuttle Bus Central Sdn. Bhd."

Hi, jimmy

Central Sdn.
Bhd.

We Provide The Best Service

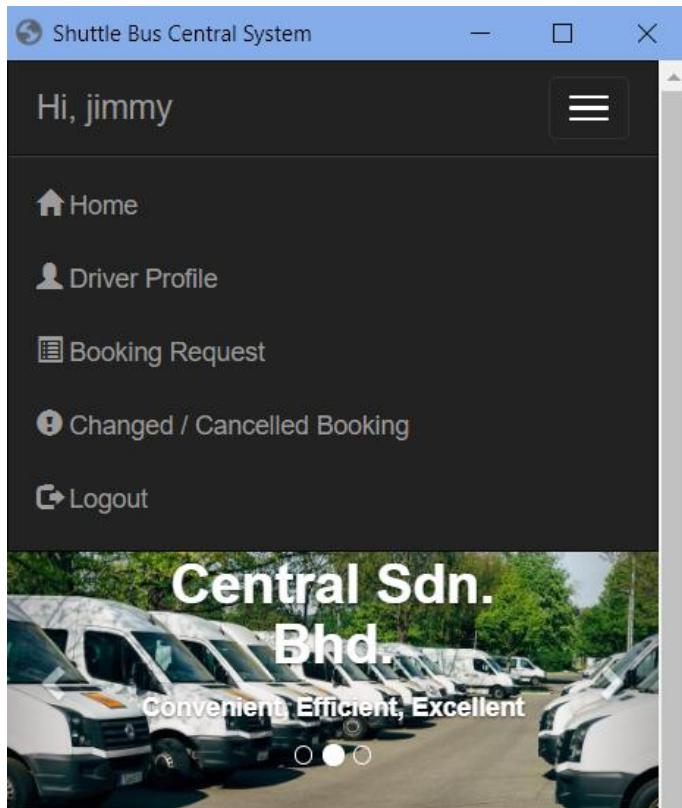
! Notification !

Booking B0003 info has changed.

Marked As Read

Copyright © 2020
Shuttle Bus Central Sdn.
Bhd.

(SBC Mobile Web Home Page)



! Notification !

"Booking B0003 info has changed."

Marked As Read

(SBC Mobile Web Dropdown Menu)

External bus provider may click on the burger menu to look for access to other pages. External bus provider will also receive notifications on the home page of the mobile web application regarding changed/cancelled bookings. External bus provider may also check the details of the changed or cancelled bookings from the Changed/Cancelled Booking page. Once the provider is notified, he/she may click on the "Marked As Read" button and the notification will disappear.

9.2.3 Driver Profile Page

The image displays two side-by-side screenshots of a mobile application interface for managing driver profiles.

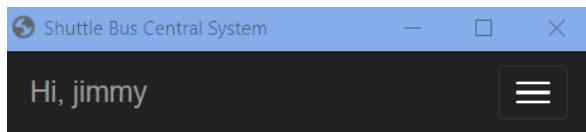
Left Screenshot (Main Driver Profile Page):

- Header: Shuttle Bus Central System
- Welcome message: Hi, jimmy
- Section title: SBC Driver Profile
- Buttons:
 - Create New Driver (highlighted with a box)
 - Update Driver Profile

Right Screenshot (Create New Driver Profile Tab):

- Header: Shuttle Bus Central System
- Section title: Create New Driver Profile
- Form fields:
 - Driver Name: Name
 - Driver IC Number: IC Number
 - Driver License Number: License Number
 - Driver Phone Number: Phone Number
- Save button

(SBC Mobile Web Driver Profile- Create New Driver Profile Tab)



SBC Driver Profile

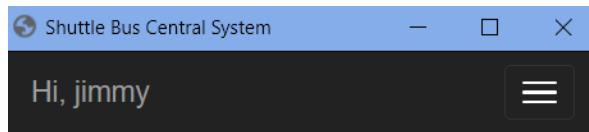
Create New Driver

Update Driver Profile

Update Driver Profile

Choose Your Driver:

Select



SBC Driver Profile

Create New Driver

Update Driver Profile

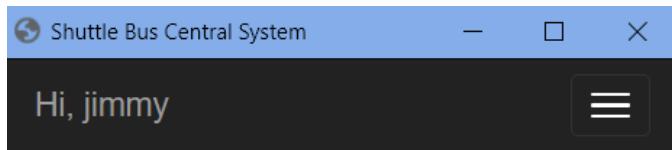
Update Driver Profile

Choose Your Driver:

Choose Your Driver:
Jensen Fernandez, 741225624593
Jimmy Choo, 750518441256
Brad Choi, 860423085113

Select

(SBC Mobile Web Driver Profile- Update Driver Profile Tab)



Update Driver Profile

Driver Name:

Jimmy Choo

Driver IC Number:

750518441256

Driver License Number:

L5521

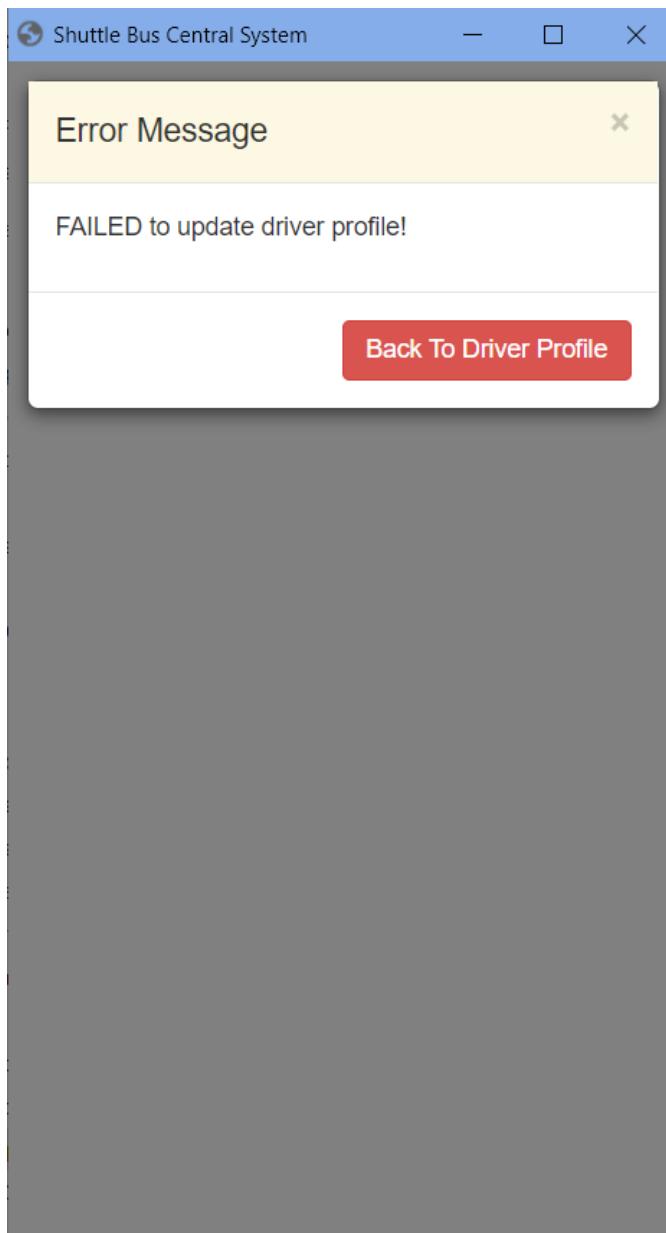
Driver Phone Number:

0145965523

Save

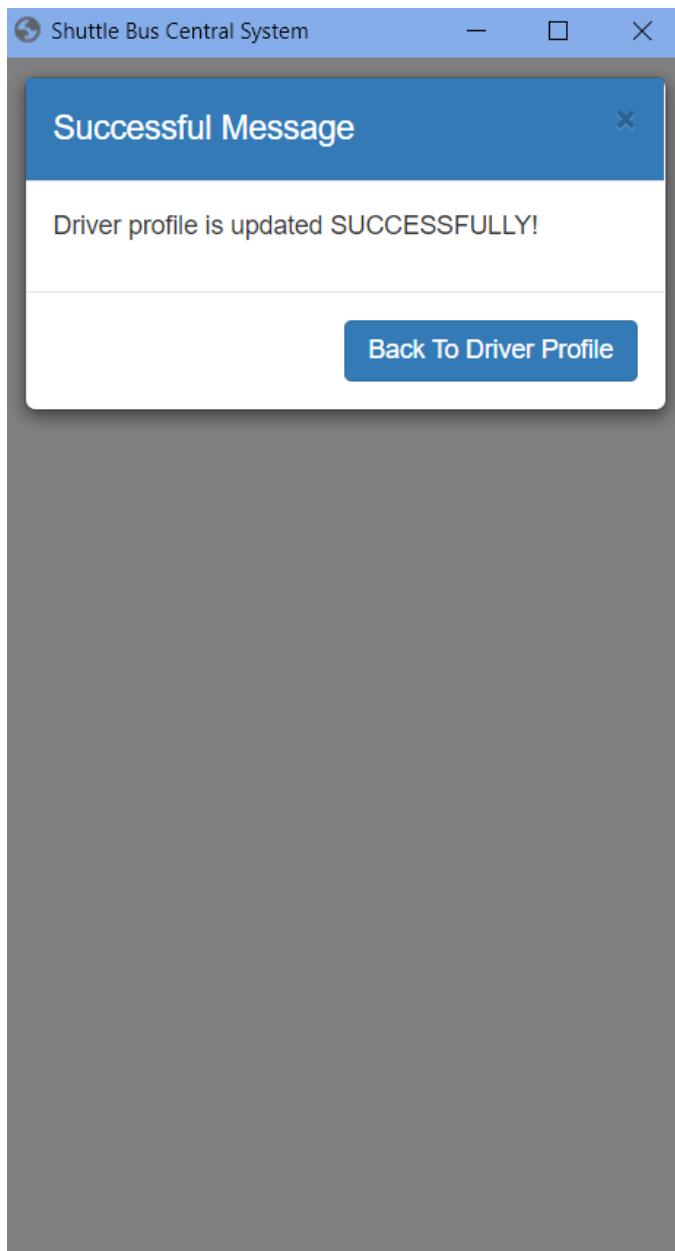
(SBC Mobile Web Driver Profile- Update Driver Profile Tab)

External bus provider may change the driver phone number only but not driver's name, IC number and license number in SBC mobile web application.



(SBC Mobile Web Driver Profile- Update Driver Profile Tab)

External bus provider will receive an error message if he/she enters inappropriate phone number and will be directed back to the driver profile page.



(SBC Mobile Web Driver Profile- Update Driver Profile Tab)

External bus provider will receive a successful message if the phone number is updated to the system successfully and will be directed back to the driver profile page.

9.2.4 Booking Request

The image displays three separate windows of the "Shuttle Bus Central System" application, all titled "SBC Booking Request". Each window shows a table with booking information.

BOOKING ID	CUSTOMER PHONE NO.	CUSTOMER NAME	DATE	DEPART LOCATION	ARRIVAL LOCATION	SBC STAFF ID	REQUESTER	PAYMENT STATUS	REMARKS
B0004	0115228896	Christan	06/11/2020	KLIA	KLCC	E0003		Requested	Carrying luggage along

(SBC Mobile Web Booking Request Page)

External bus provider will be able to check all the outsourced booking from the booking request page in SBC mobile web application. External bus owner may click on the “Select” button for whichever booking date that the external bus providers are able to satisfy.

The image displays two identical web pages from the "Shuttle Bus Central System". Both pages have a blue header bar with the system name and standard window control buttons (minimize, maximize, close). The main content area is white with a light gray border.

Please assign a bus/van and driver:

Current Booking ID: B0004

Bus / Van: Select bus / van

Driver: Select driver

Remarks: (Empty text input field)

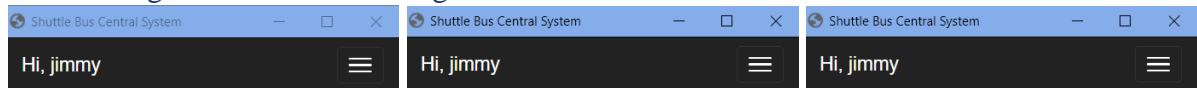
Save (Blue button)

The right-hand screenshot shows the same fields, but the "Bus / Van" dropdown is populated with "School Bus, KGH5226". The "Driver" dropdown is populated with "Brad Choi, 860423085113, 0164256632".

(SBC Mobile Web Booking Request Page- Assign Bus/Van/Driver Page)

External bus provider is directed to the assign bus/van and driver page to select the available bus/van and drivers to satisfy customers' booking requests. Once the external bus provider clicks on the "Save" button, he/she will be directed back to booking request page.

9.2.5 Changed/Cancelled Booking



**SBC Changed /
Cancelled Booking**

BOOKING ID	CUSTOMER PHONE NO	DRIVER NAME	DATE
B0003	0115228896	Jimmy Choo	06/11/2018
B0004	0115228896	Brad Choi	06/11/2018

**SBC Changed /
Cancelled Booking**

DEPART LOCATION	ARRIVAL LOCATION	SBC STAFF ID	PASSENGER NO.
Kelantan	Perak	E0002	10
KLIA	KLCC	E0003	6

**SBC Changed /
Cancelled Booking**

CARRIER ID	PASSENGER NO.	PAYMENT STATUS	REMARKS
002	10	Booked but not paid	2 stops during the journey
003	6	Booked but not paid	Carrying luggage along

(SBC Mobile Web Changed/Cancelled Booking Page)

External bus provider will be able to check the details of changed or cancelled booking from the changed/cancelled booking page in SBC mobile web application.

10.0 Unit Testing

10.1 Test Plan

Testing was part of the essential phase where the software will go through a series of test cases to understand if the system will behave in unexpected ways. During the testing phase, the project group designed a series of tests with actual data, together with the expected results to evaluate if the developed system will run as expected. In the test, 3 test cases were designed to ensure that the system would behave in an expected manner.

1. Testing if mileage would alert the admin staff to send the vehicle for service if the cumulated mileage exceeds the allowed mileage.
2. Changing the password with test data, this will test if the system will accept the legitimate data to change and update the password for the account.
3. Creating a bus with a given set of data. This will test if the code will create the bus as expected with the given data set.

10.2 Unit Testing Overview on Project

The unit testing for the project is designed in a way that no simple mistakes on the features such as changing passwords, creating bus records, and alerts on mileage would fail. The overall unit test was successfully executed with minor error in the CreateBusTest test case. Reason of the fail is because of the null object call in the project function itself. After removing the line of null object in the project function, the all 3 test cases is successfully passed.

10.3 Test Data for each Unit Testing

10.3.1 Test Case 1

Test Case 1

System: Windows Form Application

Executed By: Derrick Lim Jenn Yuen

Short Description:

To identify if the cumulate mileage function will warn the admin if the current mileage exceeds the allowed mileage.

Test Case Name: CreateMileageTest

Pre-Condition:

The mileage data must exist with the currentMilage larger than the allowedMileage

Test Data	Expected Results	Actual Response	Status
currentMilage • 15000	Alerts the admin since $15000 > 7000$	Pop up box alerts the admin	Success
allowedMilage • 7000			

10.3.2 Test Case 2

Test Case 2

System: Windows Form Application

Executed By: Derrick Lim Jenn Yuen

Short Description:

Test if the change password function works as intended and it will update the account password

Test Case Name: ChangePasswordTest

Pre-Condition:

A supply of username and MD5 hashed password would be supplied as test data.

Test Data	Expected Results	Actual Response	Status
name • Test User passwordHash • e16b2ab8d12314bf4efbd6203906ea6c	Successfully change password	Succeed changing password	Success

10.3.3 Test Case 3

Test Case 3

System: Windows Form Application

Executed By: Derrick Lim Jenn Yuen

Short Description:

Creates a bus object and record into database with the provided data for bus information

Test Case Name: CreateBusTest

Pre-Condition:

Bus related data such as number plate, categories, passenger permit, etc must be provided as test data.

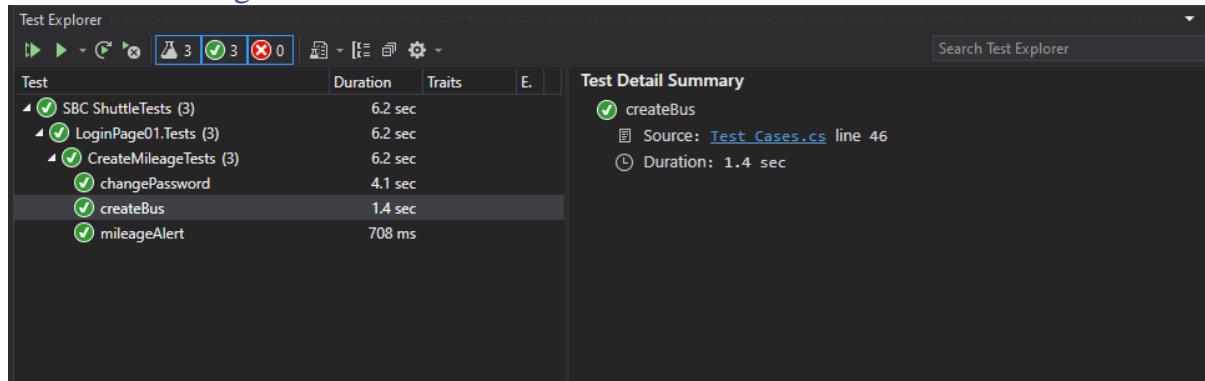
Test Data	Expected Results	Actual Response	Status
NumPlate <ul style="list-style-type: none">• JMK5520 Category <ul style="list-style-type: none">• Express Bus Specification <ul style="list-style-type: none">• Luxury bus with 25 seats at aircond at each seat Permit <ul style="list-style-type: none">• P0052 PermitValidity <ul style="list-style-type: none">• 20/02/2020-25/02/2022 OwnerIC <ul style="list-style-type: none">• 990102065522 Status <ul style="list-style-type: none">• Available	Bus will be created successfully since the data entered is all valid	Successfully created	Success

10.4 .Net Unit Test and Test Code

The screenshot shows the Visual Studio IDE with three code files open:

- SBC_ShuttleTests.cs**: A unit test class for the shuttle system. It includes imports for Microsoft.VisualStudio.TestTools.UnitTesting, LoginPage01, System, System.Collections.Generic, System.Linq, System.Text, System.Threading.Tasks, and System.Windows.Forms. The class `CreateMileageTests` contains a single test method `mileageAlert()`. The code within the method sets up test values for current and allowed mileage, creates a `CreateMileage` form, and compares the result of the `compareMileageLimit` method against an expected success value.
- LoginPage01.Tests.CreateMileageTests.cs**: A unit test class for the login page. It includes imports for Microsoft.VisualStudio.TestTools.UnitTesting and LoginPage01. The class contains two test methods: `changePassword()` and `createBus()`. The `changePassword()` method tests the `changePassword` method of the `EditAccount` class, asserting that it returns true for a given password hash and name. The `createBus()` method tests the `createBus` method of the `CreateBus` class, asserting that it returns true for specific bus parameters.
- CreateBus.cs**: A class for creating buses. It includes imports for System, System.Collections.Generic, System.Linq, System.Text, System.Threading.Tasks, and System.Windows.Forms. The class has a constructor and a method `createBus` which takes parameters for numPlate, category, specification, permit, valid date, owner ID, and status. It also includes an `Assert.AreEqual` statement for testing the method's success.

10.5 Unit Testing Results and Conclusions



After implementing the simple fix for null object reference in one of the test cases, the all 3 test results were successfully executed. After running the test cases, the project group gone an extra mile by testing the actual system again by executing the same steps from the admin perspective and running using the test data provided in this test results. As a result, all tests are success and the records have also been updated in the database.

11.0 Conclusion

The SBC project that was aimed to solve the issue of manual paperwork and complex operations to assign the buses and drivers was successfully executed and were delivered to the product owner at the end of the project timeline. The delivered system can efficiently handle SBC customers' request and assignment, and to outsource the bus services to external providers whenever SBC could not fulfil the customers' demand. The whole system is to streamline the processes and simplify the operation by providing a centralized system where SBC administrator can use to operate the booking systems. On the other hand, external providers can use the mobile web view to manage the customers' request that was handled by SBC. This include accepting customer's request, checking the schedule and updating the drivers registered to SBC.

While adopting the scrum methodology throughout the project, communication plays an important role in ensuring that the project is moving towards the correct path. The scrum master needs to have good leadership skills, sensitive and aware of the problems faced by each of the members and try to remove the obstacles. Collaboration between team members and the product owner is important to reduce the misconception of the requirements. If an error is not pinpointed out at the early stage, it will be more expensive to fix at the later stage due to the increased complexity of the system.

In conclusion, the project was delivered as promised within the timeline, fulfilling the requirements specified by the product owner.

12.0 Appendix

12.1 Meeting Minutes (Week 3 to Week 15)

12.1.1 Minutes of Meeting (Week 3)

Date: 16th August 2020

Time: 3pm

Venue: Google Hangout

Attendees:

1. Derrick Lim Jenn Yuen
2. Andreas Andi
3. Lum Zhi Yan
4. Zhang Yu Tong

Absentees: -

Last Meeting Follow-Up:

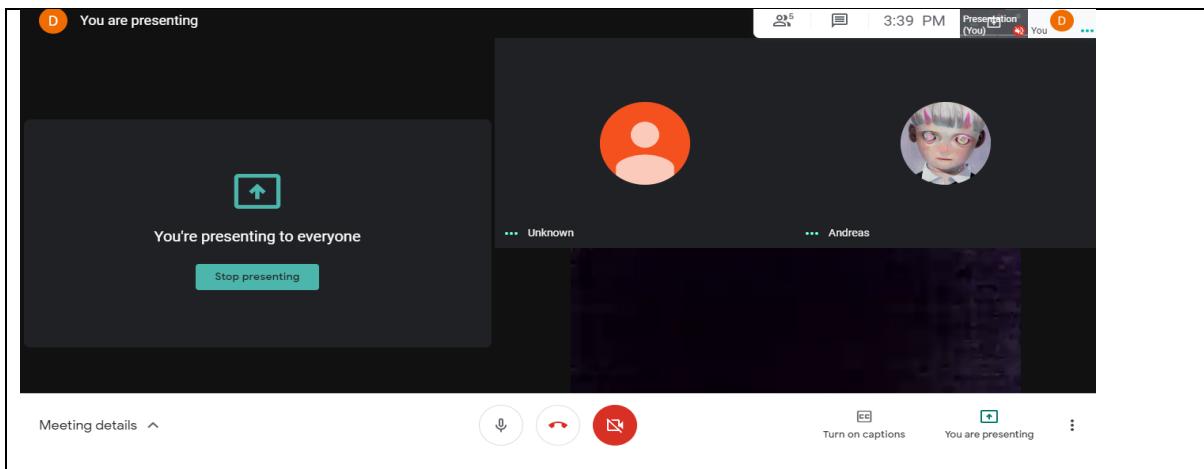
1.	-
----	---

Agenda:

1.	Roles assignment Project leader: Derrick Lim Jenn Yuen Lead programmer: Lum Zhi Yan Lead documentation: Andreas Andi Lead UX/UI designer: Zhang Yu Tong
2.	Analyze case study
3.	Discussion on system design (Window application) i. Design Window interface- Derrick, Zhang Yu Tong
4.	Discussion on mobile web tools i. Research and find out the most suitable tools for mobile web application- Lum Zhi Yan

Remarks: Next meeting to be held on 20/08/2020 to check on progress.

Meeting Screenshot:



12.1.2 Minutes of Meeting (Week 4)

Date: 20th August 2020

Time: 9pm

Venue: Google Hangout

Attendees:

5. Derrick Lim Jenn Yuen
6. Andreas Andi
7. Lum Zhi Yan
8. Zhang Yu Tong

Absentees: -

Last Meeting Follow-Up:

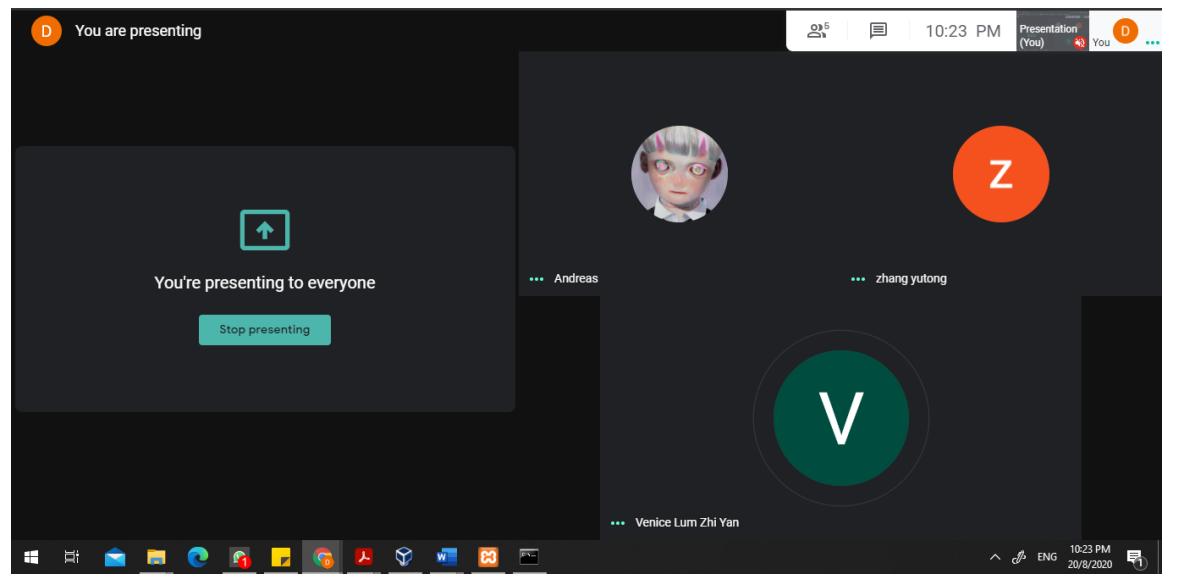
1.	Discussion on system design (Window application) i. Amend Window interface design
2.	Discussion on mobile web tools i. Decide php as mobile web application language tool

Agenda:

1.	Progress report task assignment i. Gantt chart- Derrick ii. Class Diagram- Andreas iii. Use Case Diagram- Lum Zhi Yan iv. Weekly minutes of meeting- Derrick
2.	Project leader explains to members how to use Taiga.io i. Members are required to update their tasks on Taiga.io every week

Remarks: -

Meeting Screenshot:



12.1.3 Minutes of Meeting (Week 5)

Date: 24th August 2020

Time: 5pm

Venue: Google Hangout

Attendees:

9. Derrick Lim Jenn Yuen
10. Andreas Andi
11. Lum Zhi Yan
12. Zhang Yu Tong

Absentees: -

Last Meeting Follow-Up:

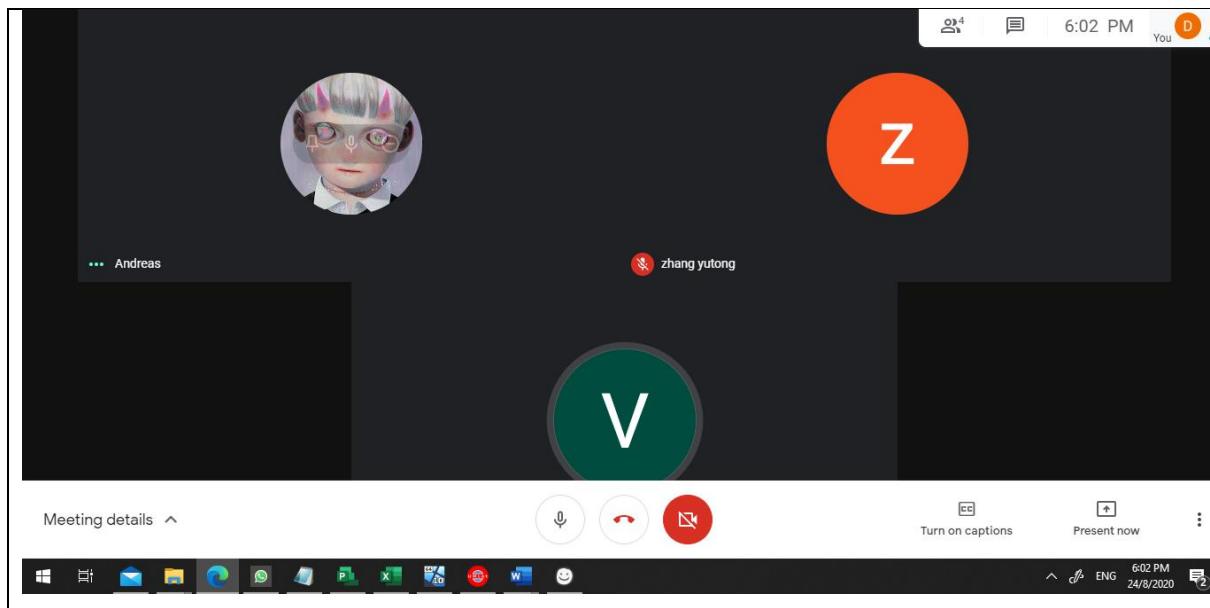
- | | |
|----|---|
| 1. | Discussion on progress report task assignments <ol style="list-style-type: none">i. Amend class diagramii. Amend use case diagramiii. Amend Gantt chart |
|----|---|

Agenda:

- | | |
|----|---|
| 1. | Create MySQL database for SBC- Derrick |
| 2. | Discussion on system design (mobile web application) <ol style="list-style-type: none">i. Design mobile web interface- Lum Zhi Yan, Andreas |

Remarks:

Meeting Screenshot:



12.1.4 Minutes of Meeting (Week 6)

Date: 1st September 2020

Time: 5pm

Venue: Google Hangout

Attendees:

13. Derrick Lim Jenn Yuen
14. Andreas Andi
15. Lum Zhi Yan
16. Zhang Yu Tong

Absentees: -

Last Meeting Follow-Up:

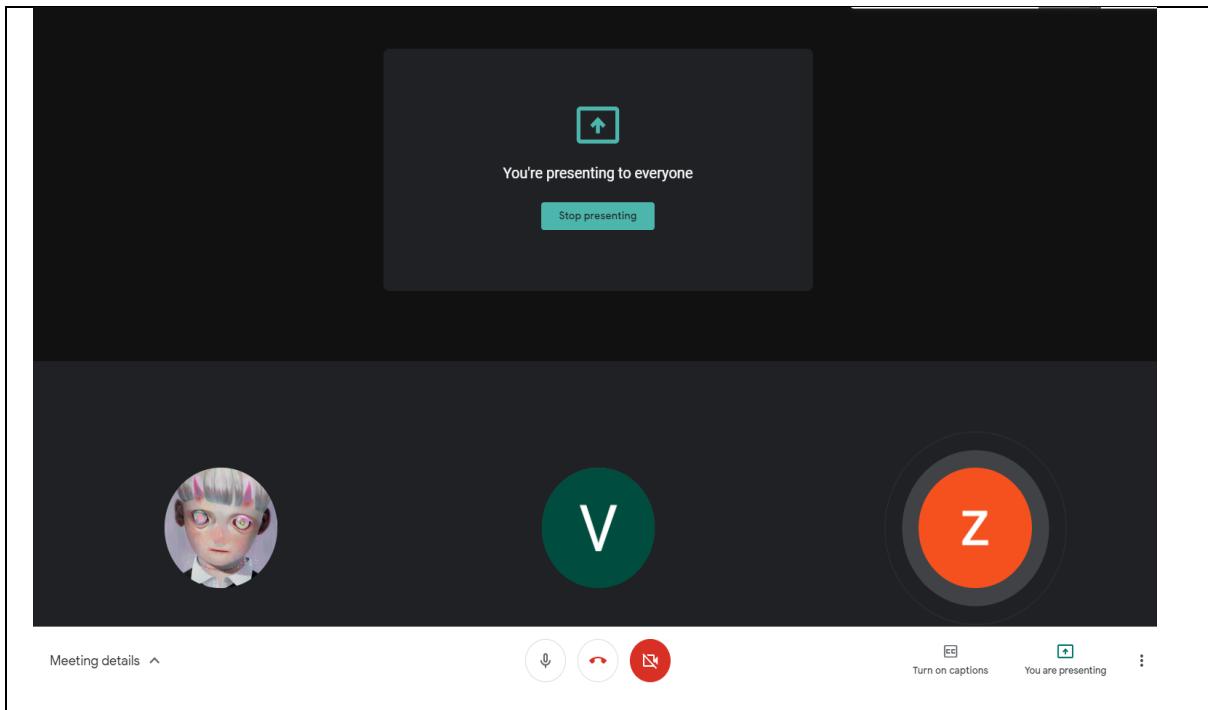
1.	Discussion on system design (mobile web application) i. Amend mobile web design
2.	Update MySQL database

Agenda:

1.	Update Google docs i. Upload all the progress report diagrams
2.	Create Window login page i. Front-end- Zhang Yu Tong ii. Back-end- Derrick

Remarks:

Meeting Screenshot:



12.1.5 Minutes of Meeting (Week 7)

Date: 7th September 2020

Time: 4.30pm

Venue: Google Hangout

Attendees:

17. Derrick Lim Jenn Yuen
18. Andreas Andi
19. Lum Zhi Yan
20. Zhang Yu Tong

Absentees: -

Last Meeting Follow-Up:

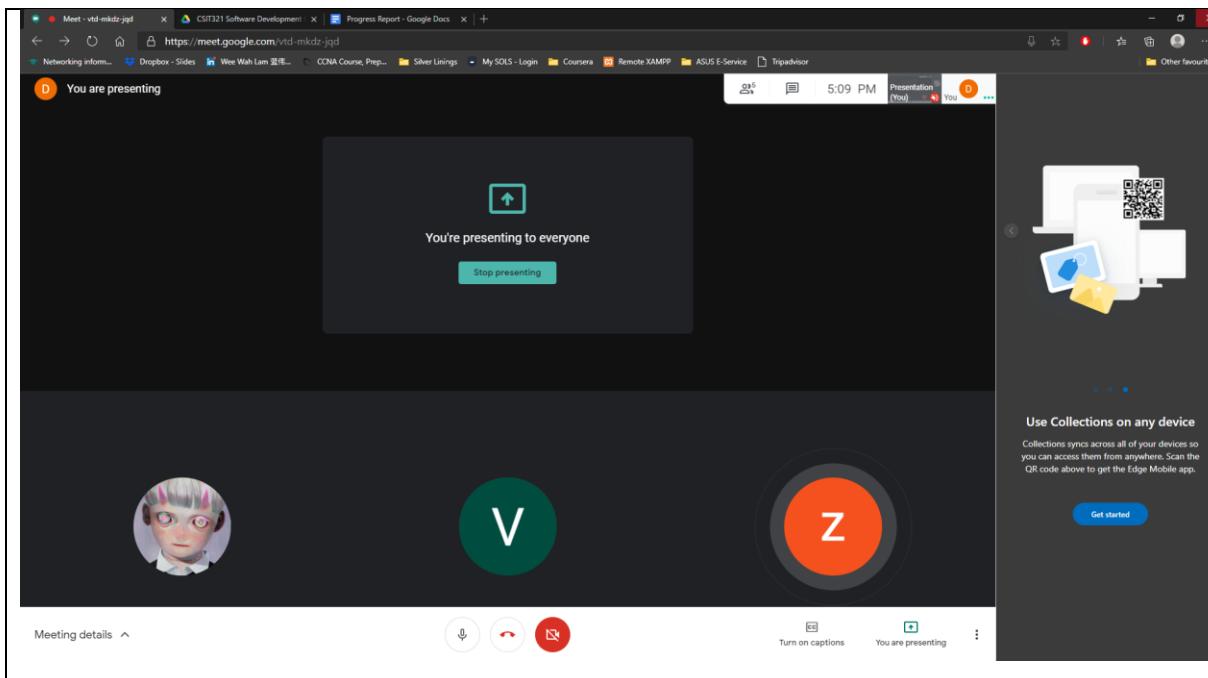
1.	Discussion on system design (Window application) i. Update Window application programming codes
2.	Discussion on system design (mobile web application) i. Update mobile web design

Agenda:

1.	Finalize progress report i. Update use case diagram ii. Update class diagram iii. Update Gantt chart
2.	Create mobile web application backend with php- Andreas

Remarks:

Meeting Screenshot:



12.1.6 Minutes of Meeting (Week 8)

Date: 14th September 2020

Time: 5pm

Venue: Google Hangout

Attendees:

21. Derrick Lim Jenn Yuen
22. Andreas Andi
23. Lum Zhi Yan
24. Zhang Yu Tong

Absentees: -

Last Meeting Follow-Up:

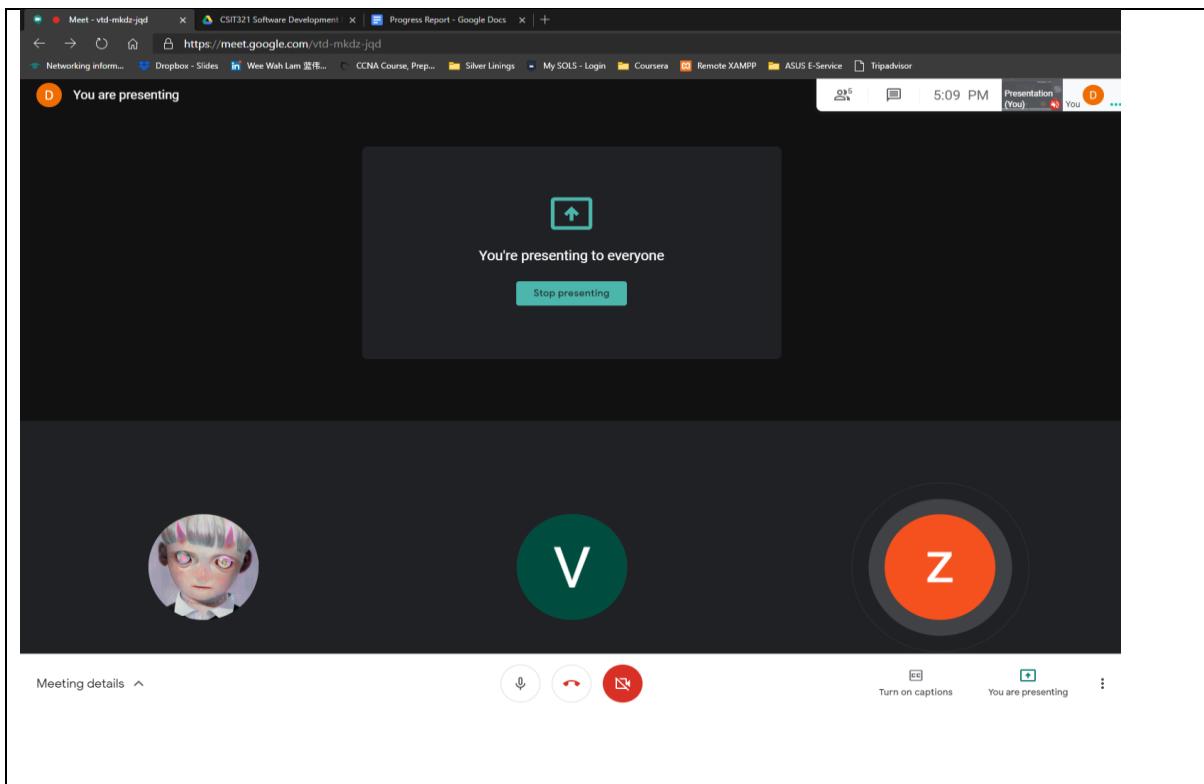
- | | |
|----|--|
| 1. | Discussion on system design (mobile web application)
i. Testing to create mobile web application backend with php |
|----|--|

Agenda:

- | | |
|----|---|
| 1. | Discussion on mobile web application
i. Create mobile web login page- Lum Zhi Yan, Andreas |
| 2. | Discussion on Window application
i. Create form to store external bus owner information- Derrick |

Remarks:

Meeting Screenshot:



12.1.7 Minutes of Meeting (Week 9)

Date: 22nd September 2020

Time: 3.30pm

Venue: Google Hangout

Attendees:

1. Derrick Lim Jenn Yuen
2. Andreas Andi
3. Lum Zhi Yan
4. Zhang Yu Tong

Absentees: -

Last Meeting Follow-Up:

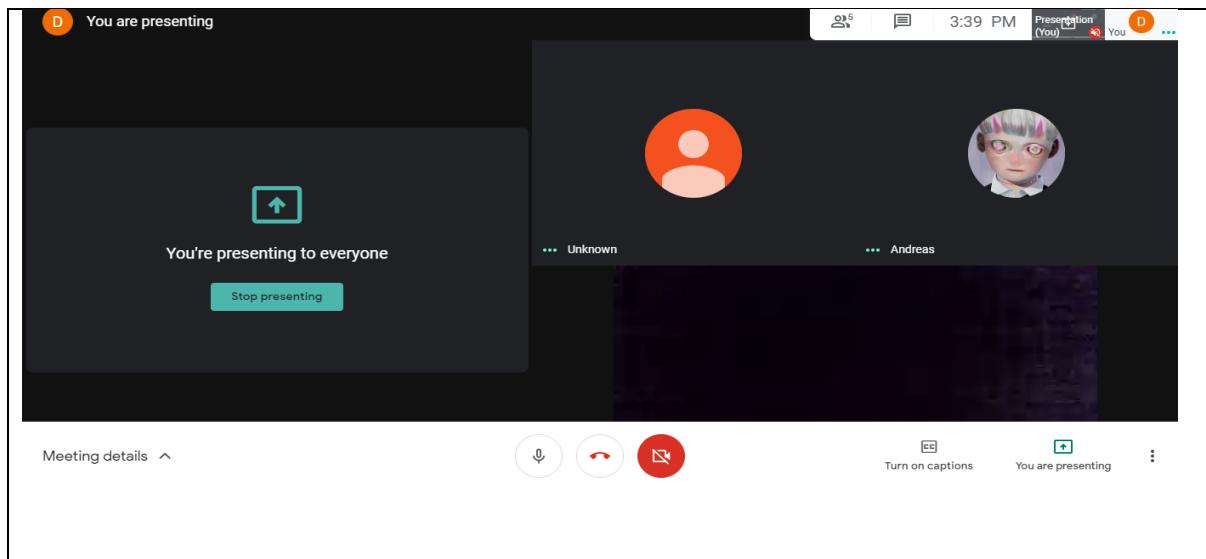
1.	Discussion on mobile web application
	i. Create mobile web login page
2.	Discussion on Window application
	i. Create form to store external bus owner information

Agenda:

1.	Discussion on new user stories.
2.	Report documentation assignment
	i. Class Diagram- Derrick
	ii. Use Case Diagram- Lum Zhi Yan
	iii. Sequence Diagram- Andreas
3.	Discussion on Window application
	i. Create form to store driver information- Derrick, Zhang Yu Tong
	ii. Create form to store bus information- Derrick, Zhang Yu Tong
4.	Discussion on mobile web application
	i. Create mobile web homepage- Andreas, Lum Zhi Yan

Remarks:

Meeting Screenshot:



12.1.8 Minutes of Meeting (Week 10)

Date: 30th September 2020

Time: 5pm

Venue: Google Hangout

Attendees:

1. Derrick Lim Jenn Yuen
2. Andreas Andi
3. Lum Zhi Yan
4. Zhang Yu Tong

Absentees: -

Last Meeting Follow-Up:

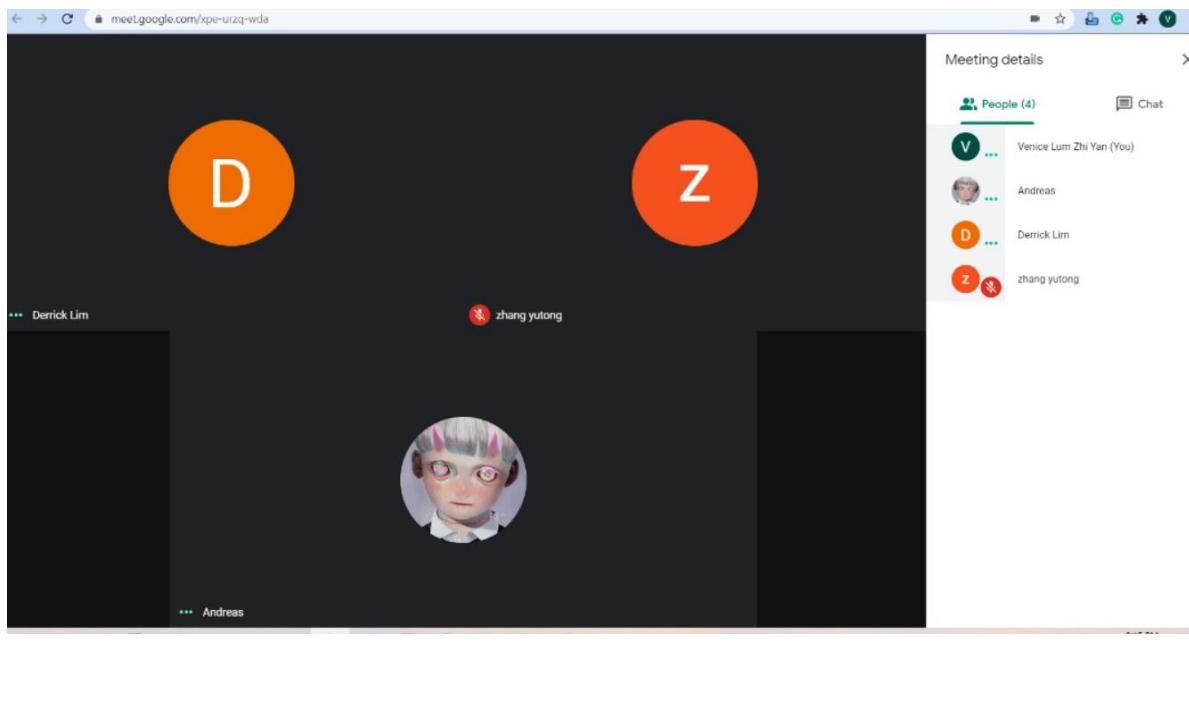
1.	Discussion on report documentation <ol style="list-style-type: none">i. Class Diagram- Derrickii. Use Case Diagram- Lum Zhi Yaniii. Sequence Diagram- Andreas
2.	Discussion on Window application bus and driver page
3.	Discussion on mobile web application homepage

Agenda:

1.	Report documentation assignment <ol style="list-style-type: none">i. Project Overview- Zhang Yu Tongii. Tools and Software Used in the Project- Andreasiii. System Architecture and Conceptual Diagram- Lum Zhi Yan
2.	Discussion on the system database. <ol style="list-style-type: none">i. Modify database according to new user stories.
3.	Discussion on Window application <ol style="list-style-type: none">i. Manage view/edit page for external bus owner page, bus page and driver page
4.	Discussion on mobile web application <ol style="list-style-type: none">i. Create mobile web driver profile page (new driver page and update driver profile page)- Andreas, Lum Zhi Yan

Remarks:

Meeting Screenshot:



12.1.9 Minutes of Meeting (Week 11)

Date: 5th October 2020

Time: 5pm

Venue: Google Hangout

Attendees:

1. Derrick Lim Jenn Yuen
2. Andreas Andi
3. Lum Zhi Yan
4. Zhang Yu Tong

Absentees: -

Last Meeting Follow-Up:

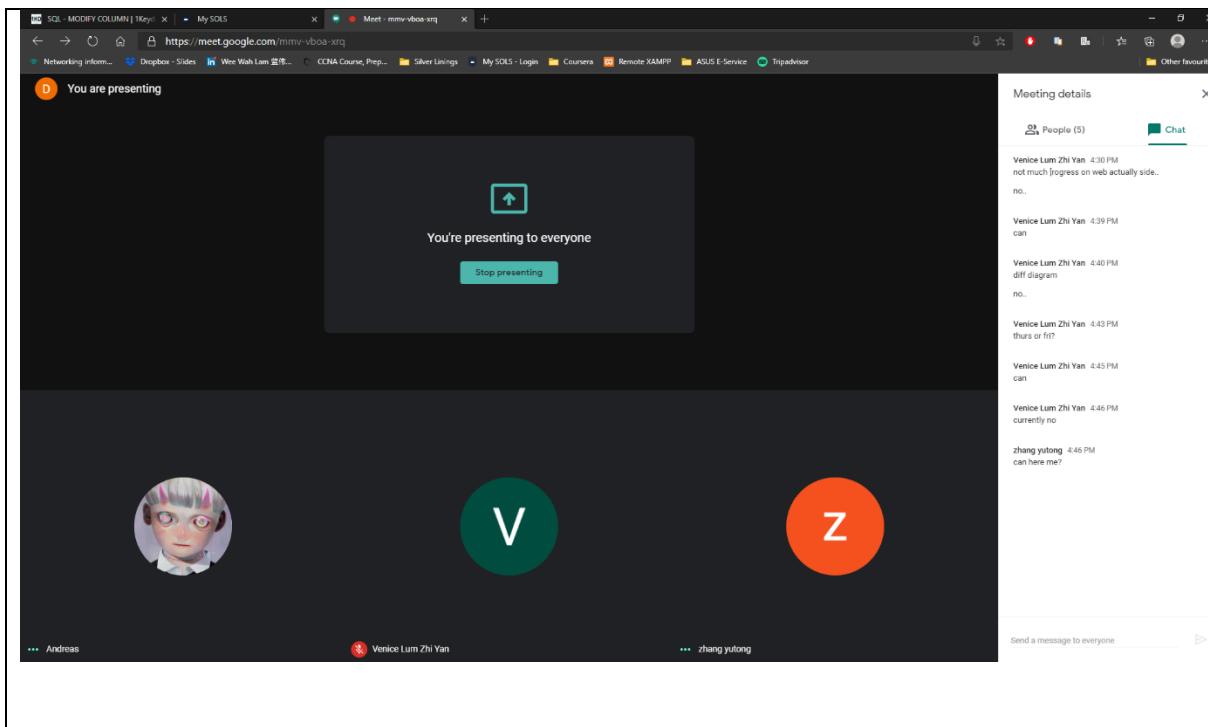
1.	Discussion and checking on report documentation <ol style="list-style-type: none">i. Project Overview- Zhang Yu Tongii. Tools and Software Used in the Project- Andreasiii. System Architecture and Conceptual Diagram- Lum Zhi Yan
----	---

Agenda:

1.	Report documentation assignment <ol style="list-style-type: none">i. Group Structure, Roles and Responsibilities- Zhang Yu Tongii. Product Overview- Andreasiii. System Architecture and Conceptual Diagram- Lum Zhi Yan
2.	Discussion on Window application <ol style="list-style-type: none">i. Manage search function (filter) for required pages- Derrickii. Manage mileage and maintenance record- Derrick, Zhang Yu Tong
3.	Discussion on mobile web application <ol style="list-style-type: none">i. Create mobile web booking request page (list of booking request)- Andreas, Lum Zhi Yan

Remarks:

Meeting Screenshot:



12.1.10 Minutes of Meeting (Week 12)

Date: 12nd October 2020

Time: 10pm

Venue: Google Hangout

Attendees:

1. Derrick Lim Jenn Yuen
2. Andreas Andi
3. Lum Zhi Yan
4. Zhang Yu Tong

Absentees: -

Last Meeting Follow-Up:

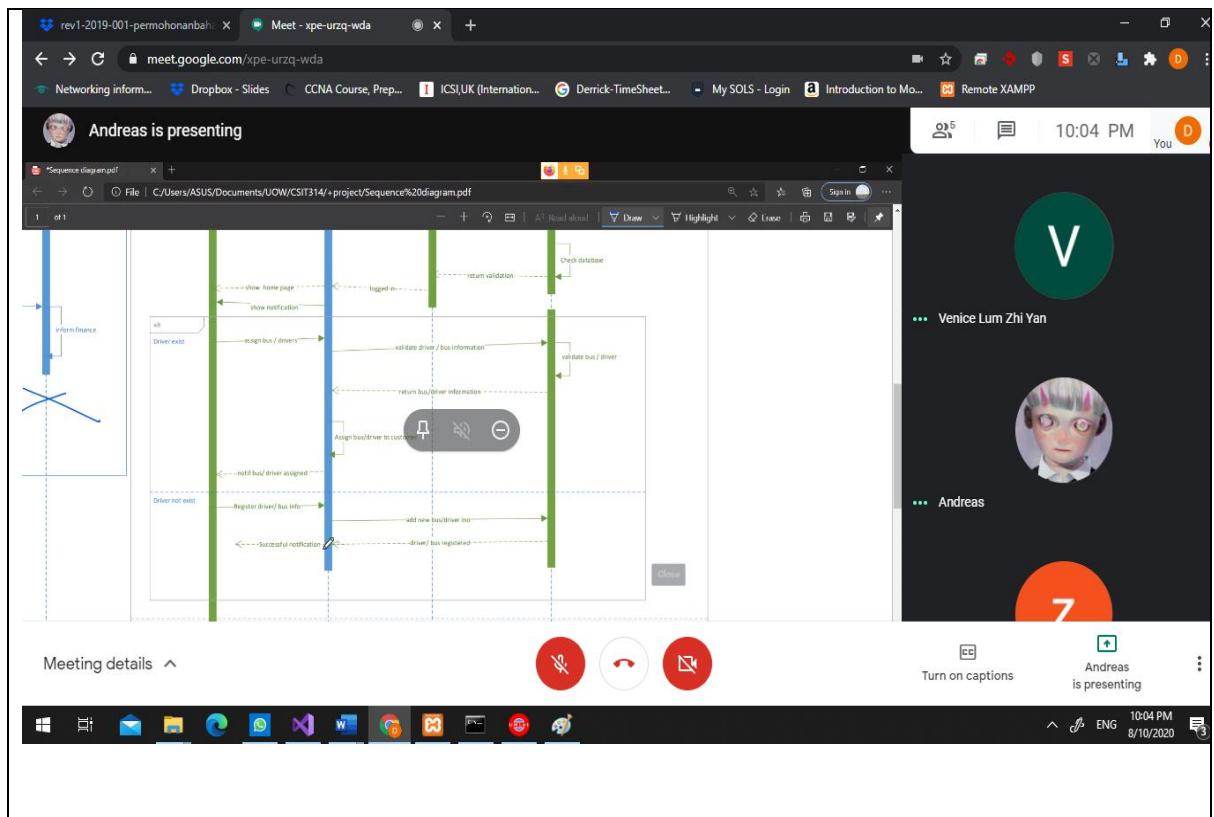
1.	Discussion and checking on report documentation <ol style="list-style-type: none">i. Group Structure, Roles and Responsibilities- Zhang Yu Tongii. Product Overview- Andreasiii. System Architecture and Conceptual Diagram- Lum Zhi Yan
----	--

Agenda:

1.	Report documentation assignment <ol style="list-style-type: none">i. Use Case Description- Lum Zhi Yan, Zhang Yu Tong, Derrickii. User Characteristics, Constraints, Assumptions and Dependencies- Andreas
2.	Discussion on Window application <ol style="list-style-type: none">i. Create button for customer booking status- Derrick, Zhang Yu Tongii. Manage sorting of bookings display (SBC first followed by external providers)- Derrickiii. Validate system and all forms- Derrick, Zhang Yu Tong
3.	Discussion on mobile web application <ol style="list-style-type: none">i. Update mobile web booking request page (list of booking request)- Andreas, Lum Zhi Yan

Remarks:

Meeting Screenshot:



12.1.11 Minutes of Meeting (Week 13)

Date: 19th October 2020

Time: 9pm

Venue: Google Hangout

Attendees:

1. Derrick Lim Jenn Yuen
2. Andreas Andi
3. Lum Zhi Yan
4. Zhang Yu Tong

Absentees: -

Last Meeting Follow-Up:

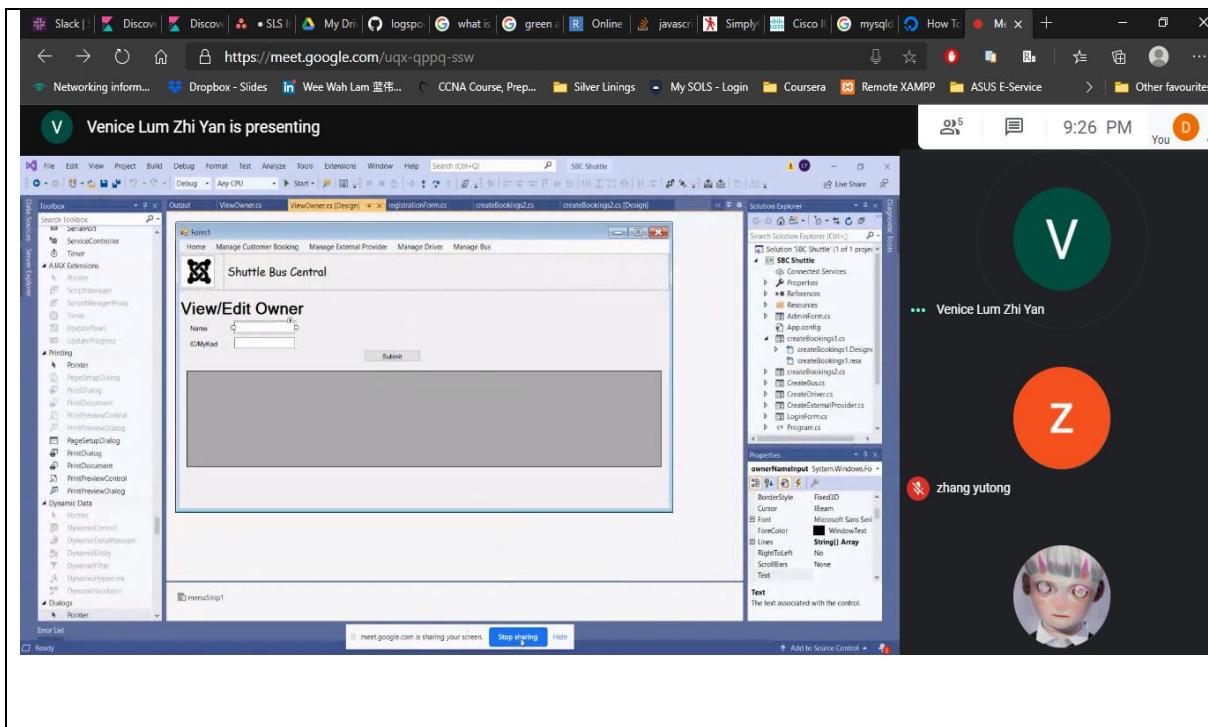
1.	<p>Discussion and checking on report documentation</p> <ol style="list-style-type: none">i. Use Case Description- Lum Zhi Yan, Zhang Yu Tong, Derrickii. User Characteristics, Constraints, Assumptions and Dependencies- Andreas
----	--

Agenda:

1.	<p>Report documentation assignment</p> <ol style="list-style-type: none">i. Update Use Case Description- Lum Zhi Yan, Derrickii. Update User Characteristics, Constraints, Assumptions and Dependencies- Andreas- Andreas
2.	<p>Discussion on Window application</p> <ol style="list-style-type: none">i. Check and discuss all bugs found in the Window system.ii. Update Window system and remove bugs found.iii. Update system validation.
3.	<p>Discussion on mobile web application</p> <ol style="list-style-type: none">i. Create and manage mobile web changed or cancelled booking (list of changed/cancelled booking)- Andreas, Lum Zhi Yanii. Validate system and all forms- Andreas, Lum Zhi Yan
4.	<p>Review requirements not implemented in the system.</p>

Remarks:

Meeting Screenshot:



12.1.12 Minutes of Meeting (Week 14)

Date: 26th October 2020

Time: 5pm

Venue: Google Hangout

Attendees:

1. Derrick Lim Jenn Yuen
2. Andreas Andi
3. Lum Zhi Yan
4. Zhang Yu Tong

Absentees: -

Last Meeting Follow-Up:

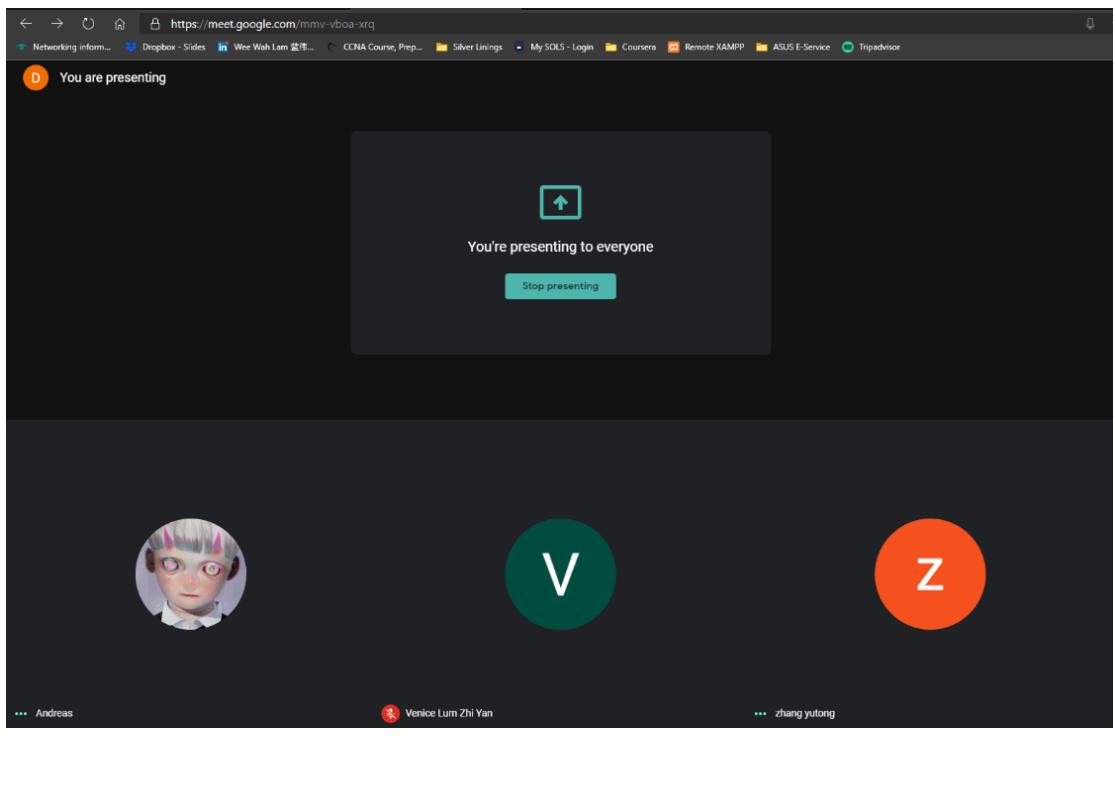
1.	Discussion and checking on report documentation <ul style="list-style-type: none">i. Update Use Case Description- Lum Zhi Yan, Derrickii. Update User Characteristics, Constraints, Assumptions and Dependencies- Andreas- Andreas
----	---

Agenda:

1.	Report documentation assignment <ul style="list-style-type: none">i. Data Persistence Diagram- Derrick, Zhang Yu Tongii. Window User Interface and System Manual- Andreasiii. Mobile Web User Interface and System Manual- Lum Zhi Yan
2.	Discussion on Window application <ul style="list-style-type: none">i. Create master account and log- Derrickii. Create alert notification- Derrick, Zhang Yu Tongiii. Update Window system and remove bugs found
3.	Discussion on mobile web application <ul style="list-style-type: none">i. Check and discuss all bugs found in the mobile web applicationii. Update mobile web application and remove bugs foundiii. Create alert notification in homepage- Andreas, Lum Zhi Yan
4.	Review and update requirements not implemented in the system.

Remarks:

Meeting Screenshot:



12.1.13 Minutes of Meeting (Week 15)

Date: 2nd November 2020

Time: 10pm

Venue: Google Hangout

Attendees:

1. Derrick Lim Jenn Yuen
2. Andreas Andi
3. Lum Zhi Yan
4. Zhang Yu Tong

Absentees: -

Last Meeting Follow-Up:

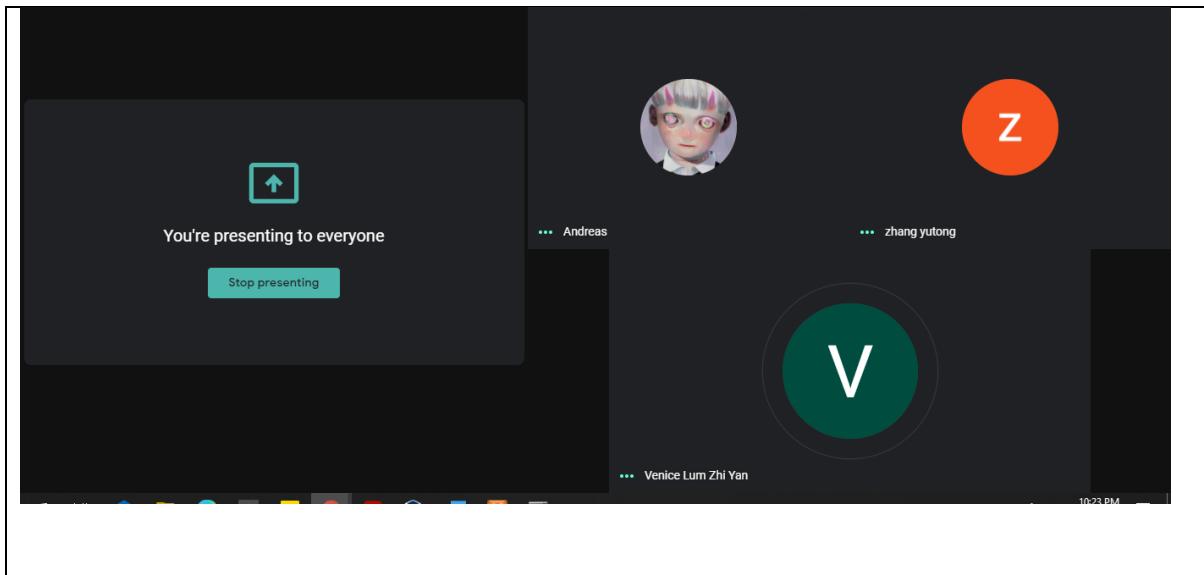
1.	Discussion and checking on report documentation <ul style="list-style-type: none">i. Data Persistence Diagram- Derrick, Zhang Yu Tongii. Window User Interface and System Manual- Andreasiii. Mobile Web User Interface and System Manual- Lum Zhi Yan
----	--

Agenda:

1.	Report documentation assignment <ul style="list-style-type: none">i. Unit Testing- Derrickii. Update Window User Interface and System Manual- Andreasiii. Update Mobile Web User Interface and System Manual- Lum Zhi Yan
2.	Discussion on Window and mobile web application testing results.
3.	Review on all the user stories and requirements.
4.	Video presentation assignment <ul style="list-style-type: none">i. Windows Application video- Derrickii. Mobile Web Application video- Lum Zhi Yaniii. Testing video- Andreas

Remarks:

Meeting Screenshot:



12.2 System Requirements and Installation

12.2.1 System Requirements

XAMPP is a stack package consisting of Apache, MariaDB, PHP and Perl interpreters. The XAMPP stack can run using minimal resources from as low as using an old Pentium processor with 128MB RAM and some storage space. The system is only developed using Apache Server to act as a PHP interpreter, and MySQL database to store centralized data.

12.2.2 Installation

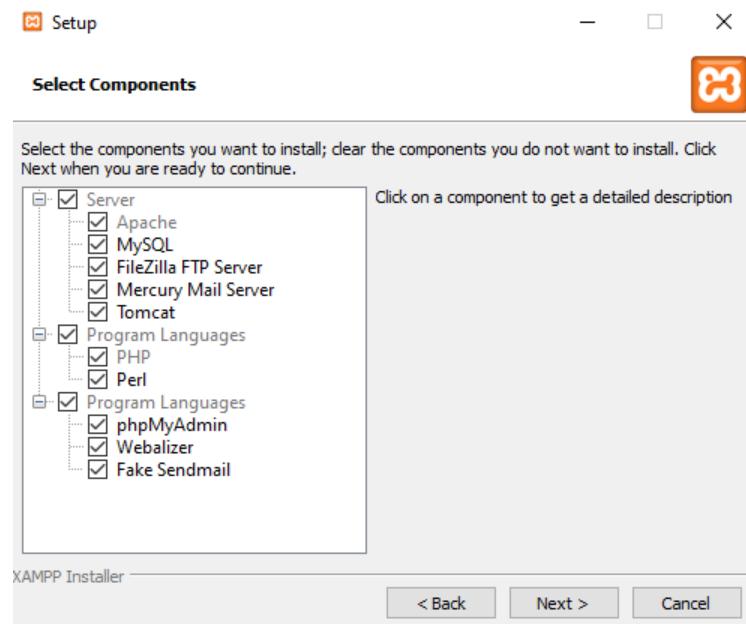
Step 1:

The screenshot shows the official XAMPP download page. On the left, there's a section titled "What is XAMPP?" with a brief description and a link to "Click here for other versions". To the right is a video player showing a thumbnail for "Introduction to XAMPP" with a play button. Below these are three download links: "XAMPP for Windows 7.4.6 (PHP 7.4.6)", "XAMPP for Linux 7.4.6 (PHP 7.4.6)", and "XAMPP for OS X 7.4.6 (PHP 7.4.6)".

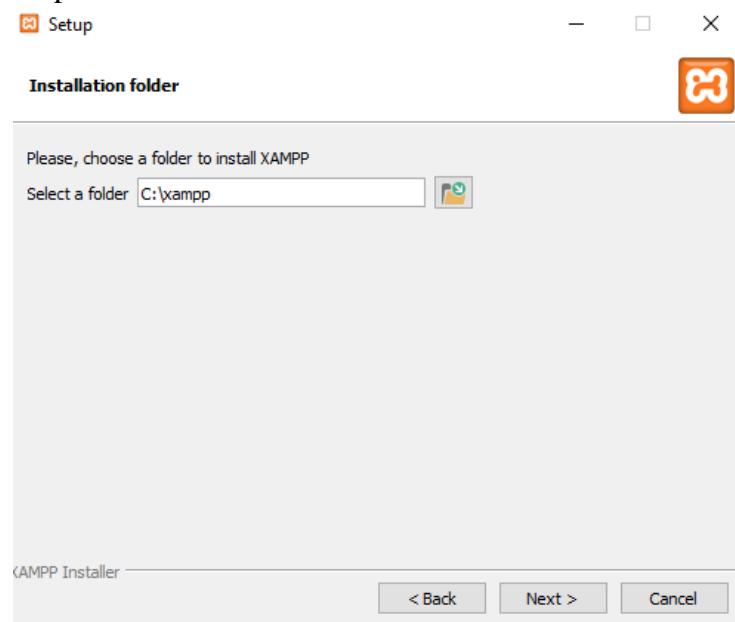
Step 2:

The screenshot shows the "Setup - XAMPP" window of the XAMPP Setup Wizard. It features a large Bitnami logo on the left and the text "Welcome to the XAMPP Setup Wizard." on the right. At the bottom, there are buttons for "< Back", "Next >", and "Cancel".

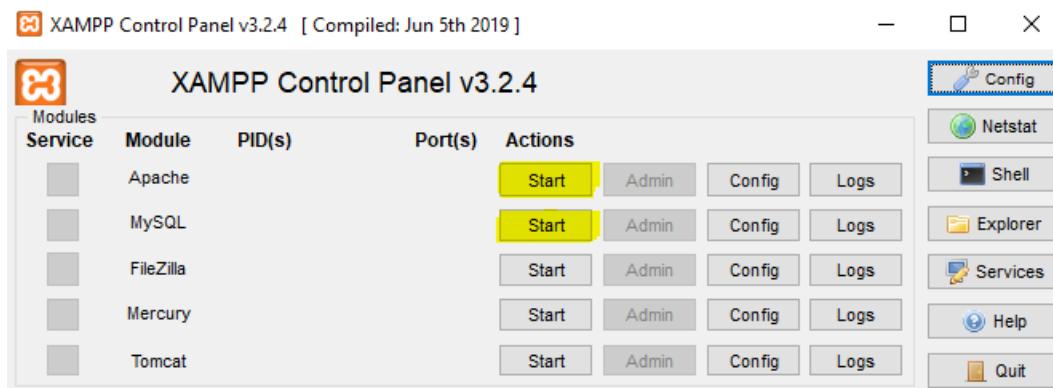
Step 3:



Step 4:

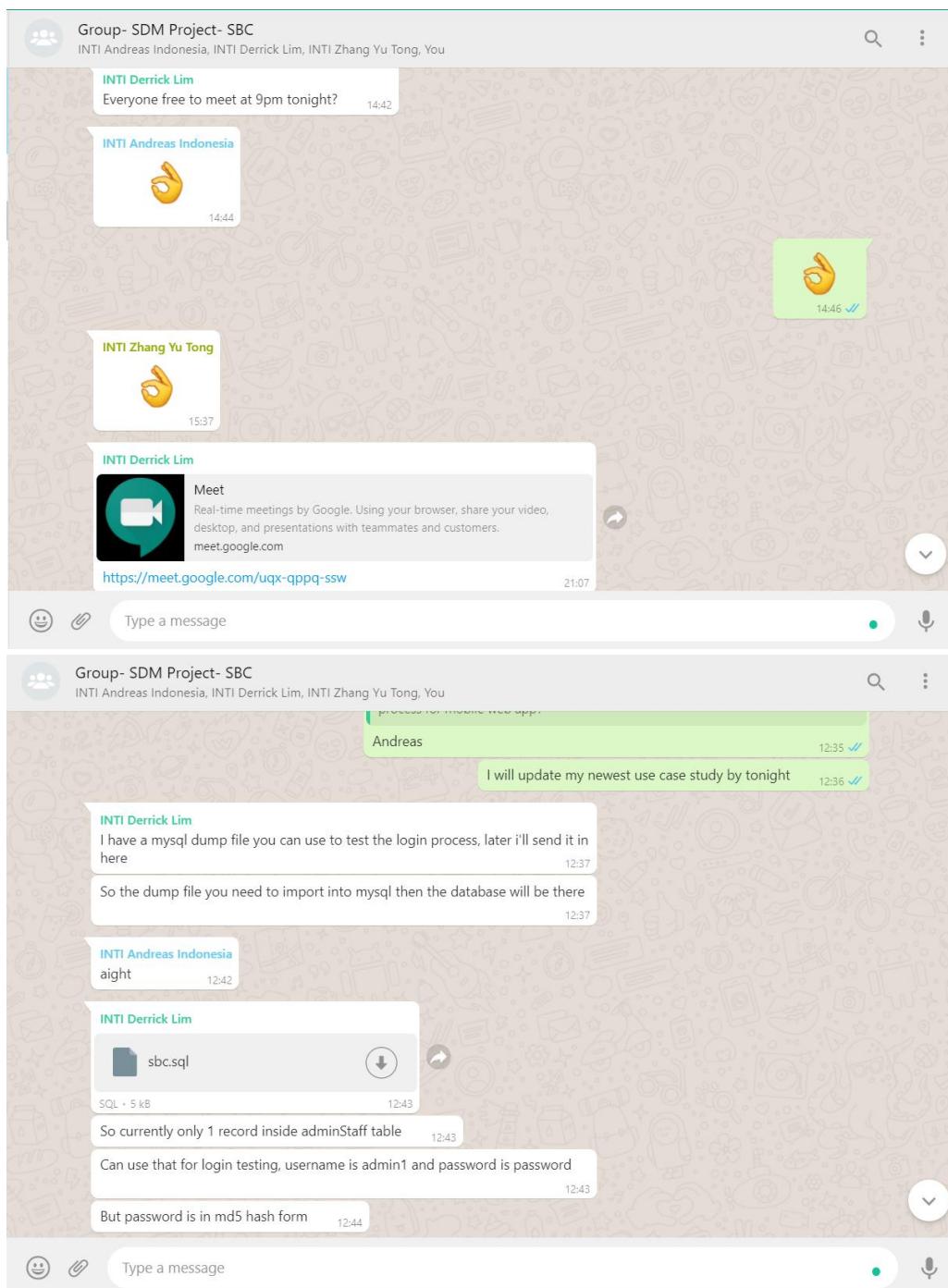


Step 5:



12.3 WhatsApp Screenshot

WhatsApp Name	Actual Name
INTI Zhang Yu Tong	Zhang Yu Tong
INTI Andreas Indonesia	Andreas Andi
INTI Derrick Lim	Derrick Lim
Green box	Lum Zhi Yan



Group- SDM Project- SBC
INTI Andreas Indonesia, INTI Derrick Lim, INTI Zhang Yu Tong, You

INTI Andreas Indonesia
INTI Derrick Lim
Send me a copy of your codes, I can help you have a look
can i send u tomorrow? i'll need to clean up some codes first 15:30

INTI Derrick Lim
Ok 15:31
Using you XAMPP right? 15:31

INTI Andreas Indonesia
yes 15:31

INTI Derrick Lim
Alright 15:31

9/12/2020

INTI Derrick Lim
We need to have contribution form inside our progress report also 16:05
Please update the contribution form on the google sheet 16:11

Type a message

Group- SDM Project- SBC
INTI Andreas Indonesia, INTI Derrick Lim, INTI Zhang Yu Tong, You

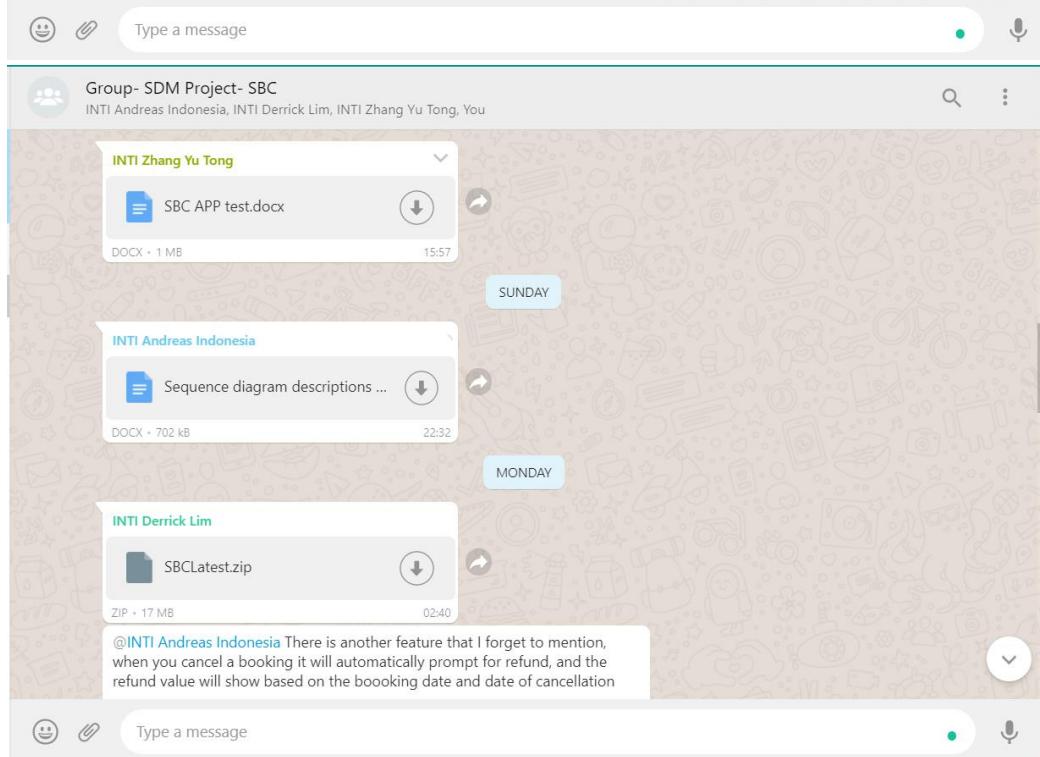
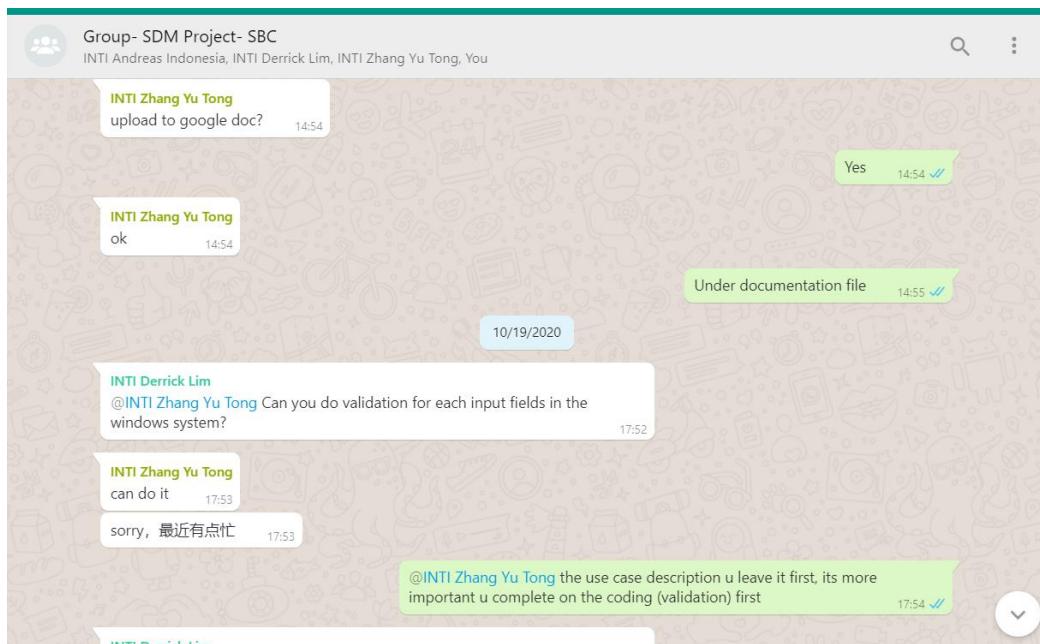
INTI Zhang Yu 15:37


INTI Derrick Lim
 Meet
Real-time meetings by Google. Using your browser, share your video, desktop, and presentations with teammates and customers.
meet.google.com/uqx-pppq-ssw 21:07

i have uploaded the documentation content here 22:06 

INTI Derrick Lim
Not sure if I mentioned this earlier, but not only the errors and bugs. Also see if

Type a message



12.3 Google Docs Screenshots

This screenshot shows the Google Drive interface for a folder named "CSIT314 Software Development Methodology". The left sidebar includes links for "My Drive", "Shared with me", "Recent", "Starred", and "Trash". A "Storage" section indicates 4.3 GB used out of 15 GB. The main area displays a list of files:

Name	Owner	Last modified	File size
Use case diagram	me	Aug 23, 2020	me
Testing	Andreas	Oct 16, 2020	Andreas
System Design	Derrick Lim	Sep 14, 2020	Derrick Lim
Sequence Diagram	Andreas	Oct 9, 2020	Andreas
Mobile Web Codes	me	Sep 16, 2020	me
Meeting minutes_v2	me	Sep 12, 2020	me
Meeting Minutes	Derrick Lim	Sep 12, 2020	Derrick Lim
Gantt	Derrick Lim	Oct 4, 2020	Derrick Lim
DOCUMENTATION	me	Nov 1, 2020	me
Class Diagram	Derrick Lim	Aug 30, 2020	Derrick Lim
UPDATED Progress Report.docx	Derrick Lim	Sep 12, 2020	Derrick Lim
Progress Report	Derrick Lim	Oct 17, 2020	zhang yutong

This screenshot shows the Google Drive interface for a folder named "DOCUMENTATION" within the "CSIT314 Software Development Methodology" folder. The left sidebar and storage information are identical to the previous screenshot. The main area displays a list of files:

Name	Owner	Last modified	File size
User interface design and system manual.docx	Andreas	Nov 4, 2020	Andreas
Use case description_v1- venice.docx	me	1:07 PM	Derrick Lim
Use case description completed-v1.docx	me	Nov 6, 2020	me
Tools and Techniques (Andreas).docx	Andreas	Oct 18, 2020	Andreas
TOC.docx	me	Nov 6, 2020	Derrick Lim
TestResults.docx	Derrick Lim	Nov 6, 2020	Derrick Lim
Sequence diagram descriptions v2.docx	Andreas	Nov 1, 2020	Andreas
SBC Gantt.mpp	Derrick Lim	Nov 4, 2020	Derrick Lim
project overview, group roles and responsibilities.docx	zhang yutong	Nov 3, 2020	Derrick Lim

12.4 Group Members Contribution Form

CSIT314 Software Development Methodologies Group Members Contribution Form				
Name:	Contribution Scope			
Derrick Lim Jenn Yuen	Planning & Management (Gantt Chart & Taiga.io entries)	Preparation of any diagrams & document	Group Meeting Attendance	System Development
Evaluation of other members:				
1. Andreas Andi	4	5	5	3
2. Lum Zhi Yan	4	5	5	4
3. Zhang Yu Tong	3	4	5	4
Name:	Contribution Scope			
Andreas Andi	Planning & Management (Gantt Chart & Taiga.io entries)	Preparation of any diagrams & document	Group Meeting Attendance	System Development
Evaluation of other members:				
1. Derrick Lim Jenn Yuen	5	4	5	5
2. Lum Zhi Yan	4	5	5	4
3. Zhang Yu Tong	3	3	5	4
Name:	Contribution Scope			
Lum Zhi Yan	Planning & Management (Gantt Chart & Taiga.io entries)	Preparation of any diagrams & document	Group Meeting Attendance	System Development
Evaluation of other members:				
1. Derrick Lim Jenn Yuen	5	5	5	5
2. Andreas Andi	4	5	5	4
3. Zhang Yu Tong	4	5	5	4
Name:	Contribution Scope			
Zhang Yu Tong	Planning & Management (Gantt Chart & Taiga.io entries)	Preparation of any diagrams & document	Group Meeting Attendance	System Development
Evaluation of other members:				
1. Derrick Lim Jenn Yuen	5	5	5	5
2. Andreas Andi	4	5	5	4
3. Lum Zhi Yan	4	5	5	4