CRITICAL ANALYSIS REPORT

2020-REG-WE-22

SLIIT (Case Study in Software Engineering)

Group Members

Group ID: 2020-REG-WE-22

| IT Number	Name	Email	Mobile
IT18097634	M.M.C. Ranasinghe	ranasinghemalith20@gmail.com	0718772549
IT18098174	Ahkam M.A.A.	maaahkam@gmail.com	0766747147
IT18134018	A.M.K. Bojitha Mindula	mindula.amk@gmail.com	0713579450
IT18142310	M.A.I.F. Isma	ishma.ish19@gmail.com	0776795592

Contents

INTRODUCTION

The Critical Evaluation Report is prepared based on a project which was designed by group 21. The short coming of the project logic were sorted according to the general project specification which was provided. The project was critically evaluated and the report has been prepared.

SHORTCOMINGS IN UML LOGIC
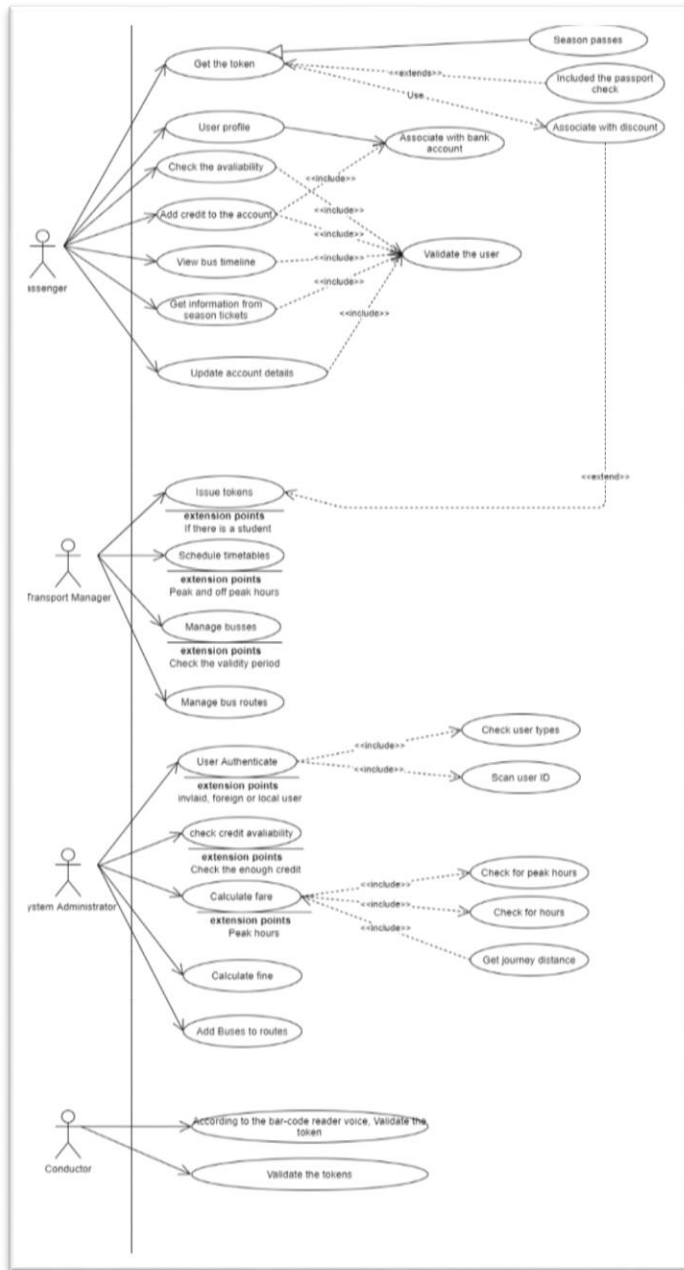
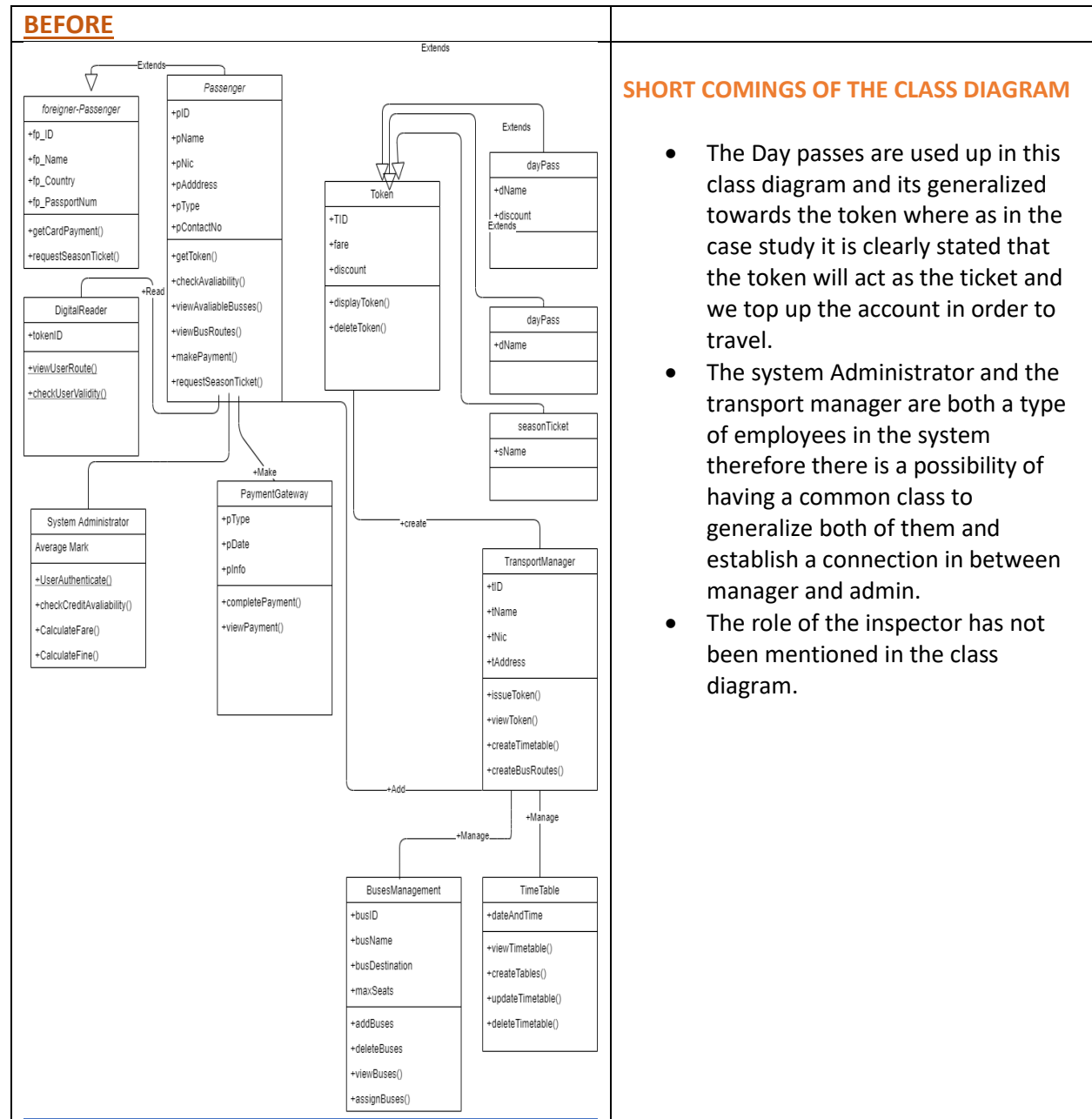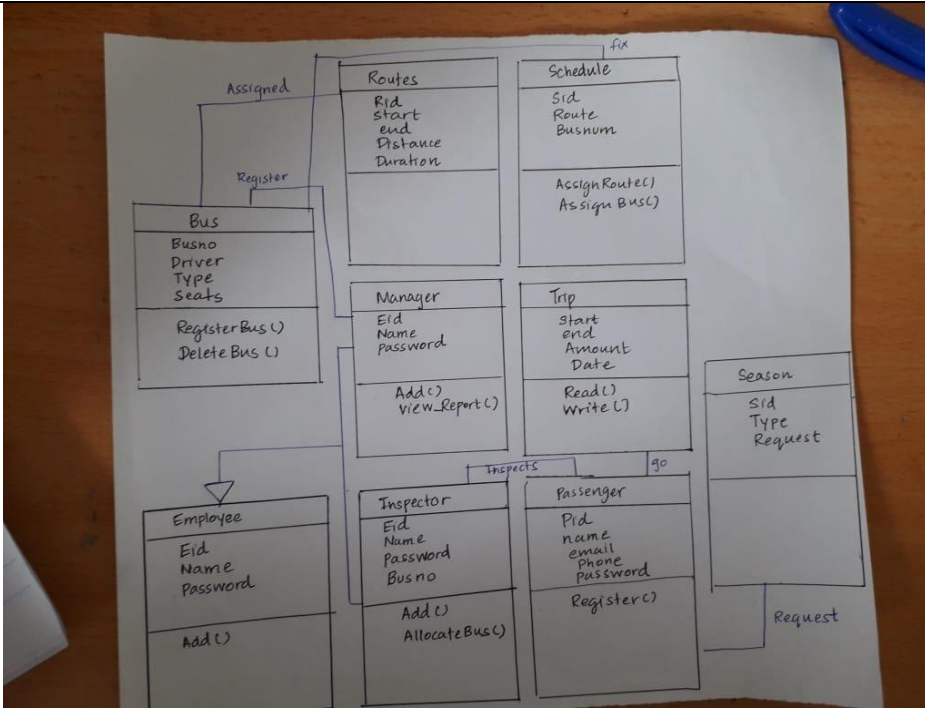| BEFORE	
	Shortcoming of the use case Diagram - The Passenger should be able to swipe the token once he starts or ends the journey. The use case diagram did not mention any such scenarios. - The Passenger shall add credits to the account where, the passenger should be able to top up the account using both mobile app and web app. Where this specific responsibility is missing. - The Passenger should be able to view the trip history this was not included in this diagram. - The Transport manager should be able to view the reports in addition to his managing task which have been assigned this task was also not mentioned. - This use case states that calculating fare, calculating fine and adding busses to routes are responsibilities of an inspector. - Where as the logic according to the case study states that calculating fare and fine are done by the machine in the bus according to the start and end of the journey.
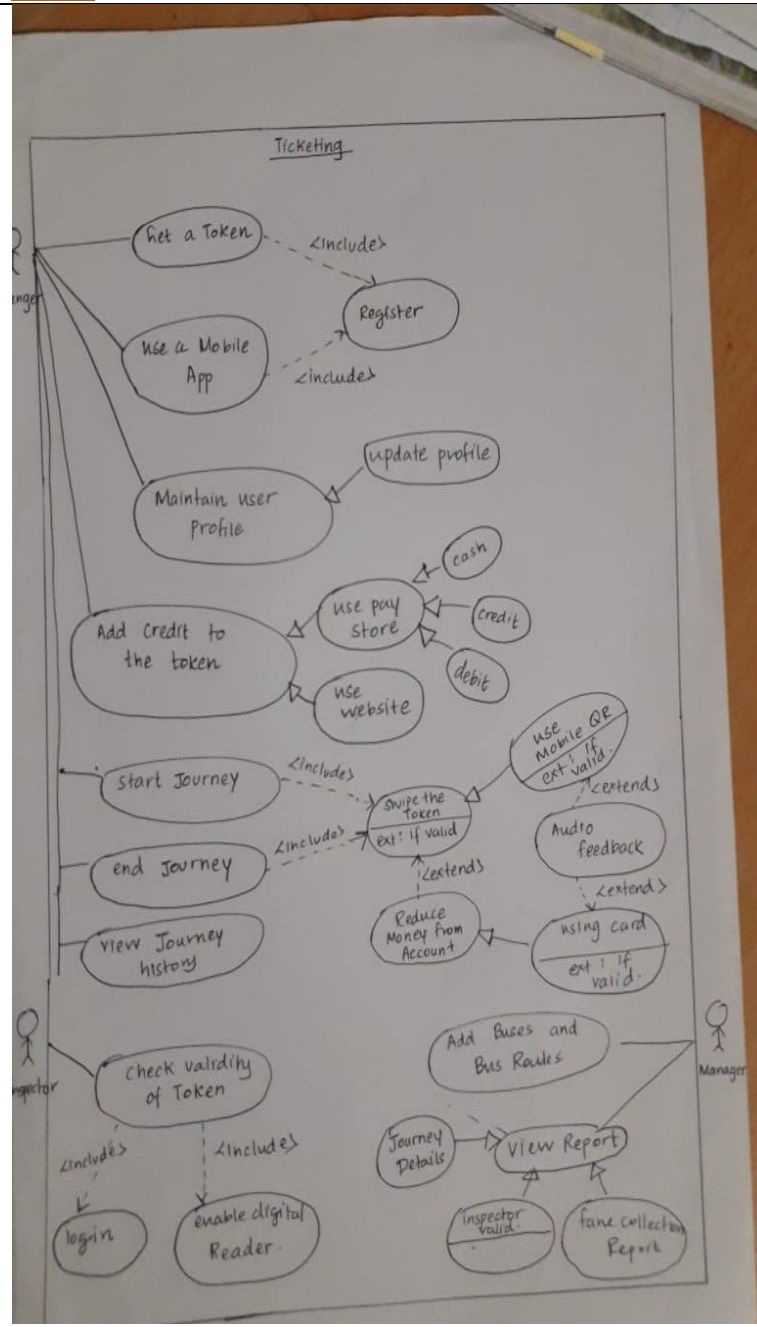
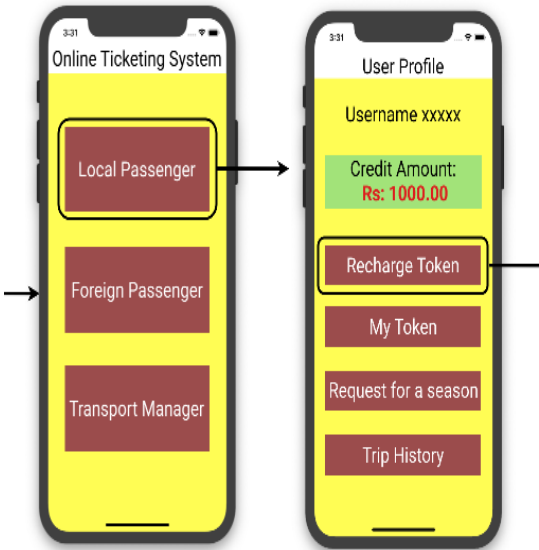| AFTER	
	THE UPDATED CONTENTS • In this Diagram the payment done in order to top up a token has been clearly elaborated. • The Token swiping mechanism has been shown in this diagram. • The inspector unwanted functionalities of the inspector has been removed in this. • The Manager being able to view the reports has been mentioned.

2.Class diagrams

SHORT COMINGS OF THE CLASS DIAGRAM

- The Day passes are used up in this class diagram and its generalized towards the token where as in the case study it is clearly stated that the token will act as the ticket and we top up the account in order to travel.
- The system Administrator and the transport manager are both a type of employees in the system therefore there is a possibility of having a common class to generalize both of them and establish a connection in between manager and admin.
- The role of the inspector has not been mentioned in the class diagram.

AFTER	
	MODIFIED CLASS DIAGRAM • An Employee class has been used as a base class for both manager and inspector. • There is no Admin specific task in the case study, therefore manager is taken up as the only person for managing certain task.

SHORTCOMINGS IN WIREFRAMES

BEFORE	AFTER
Shortcomings ✓ The login was not implementer in the Control flow diagrams. ✓ The dividing of the initial UI into 3 categories as Local passenger, Foreign passenger, Transport manager is not user friendly.	**Modifications** ✓ Any type of user should be able to login therefore implementing a common login and validating them at the backend would be a appropriate. ✓ No need to use a separate UI to view the user profile a side menu bar would be more appropriate and user friendly.

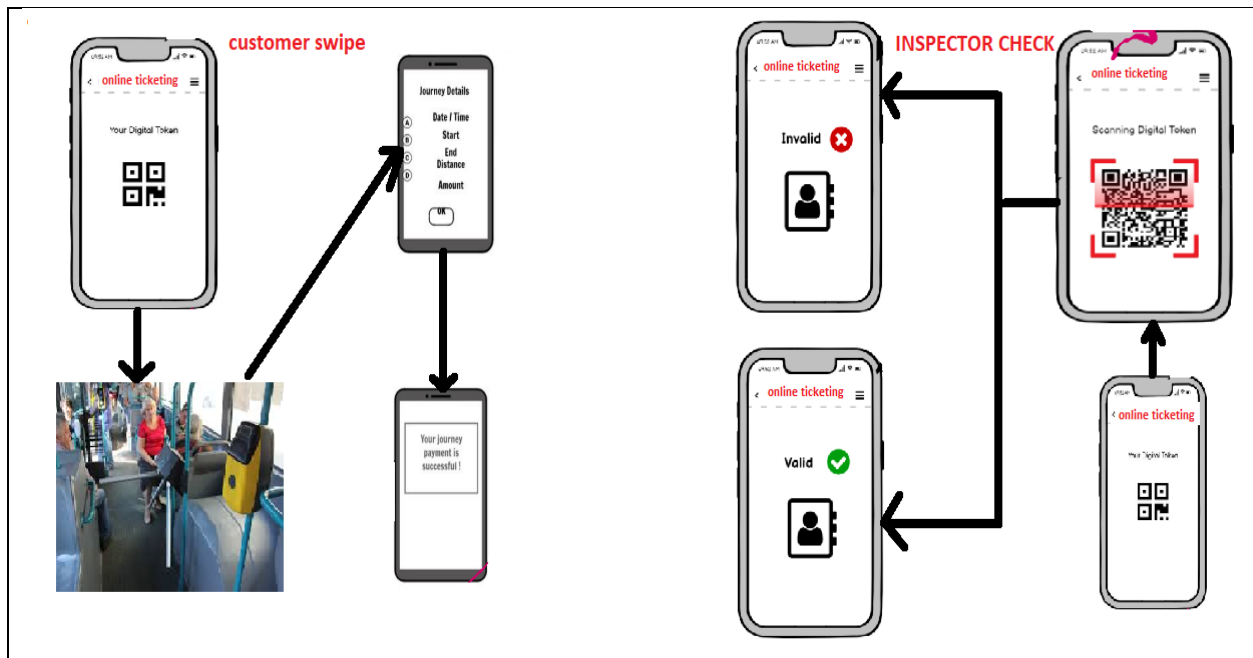| Before	After
Shortcomings	**Modifications**
✓ They have implanted a manual token where as a token implemented must be a digital one.	✓ Therefore, once the token is created it should be a QR or any other digital ticket.
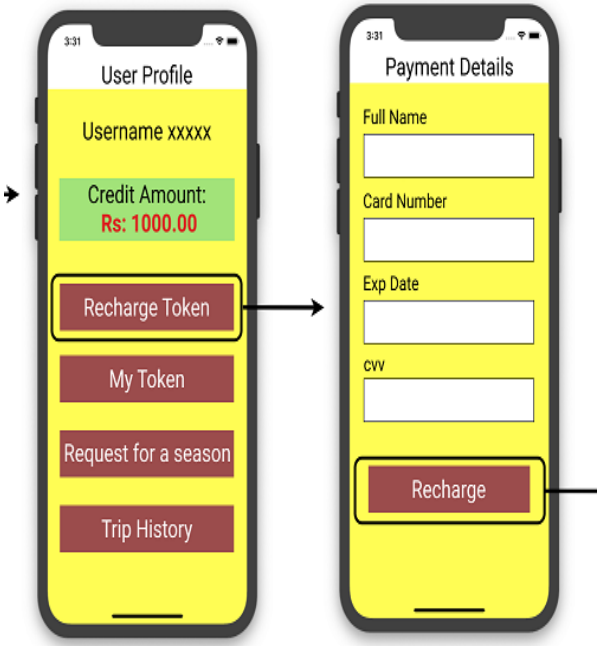
customer swipe

INSPECTOR CHECK

Un implemented functionality

- ✓ The involvement of QR was not considered in the previous version therefore we designed a set of QR flows for both the passenger and Inspector.
- ✓ Where as the passenger will extract the details of the start and end of the journey when the token is swiped on the bus scanner.
- ✓ The process in which the inspector been validated is not mentioned in any of the UI.

Previous	Modified
shortcomings	**Modified**
✓ The payment UI has been prepared and it the full name has been asked which is in appropriate. ✓ According to the UI implemented the passenger will have to pay the whole credit amount in one single payment therefore this was not logically withstanding the case study, where as the passenger should be able to pay the amount what he/she desires.	✓ The new payment UI was prepared omitting the full name . ✓ The passenger will be able to pay the amount which he/she wanted to pay. Paying multiple times in order to settle the full amount is ensured.
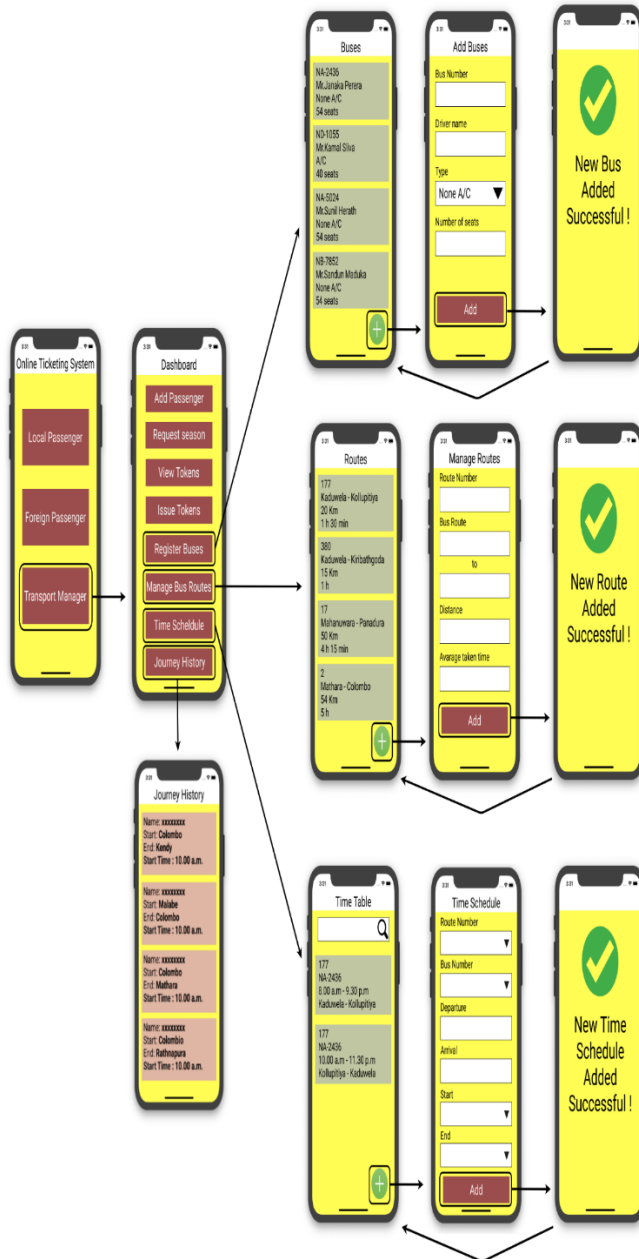
Previous	Shortcomings
	✓ In this system it is highly in convenient to maintain the whole bus system in a mobile therefore as per the case study we understood the fact that it is highly required for a web application which can be used. Hence most contents in the UI were not altered. The full system was switched to a web app from a mobile app.

Modified	Modifications
ONLINE TICKETING INSPECTOR MANAGEMENT NAME [NAME] BUS [Select ▼] [ADD INSPECTOR] id / name / Bus edit delete edit delete	✓ The inspector can be added using a web app instead ✓ The details added can be viewed in a table on side which was not available in the previous wireframe.
ONLINE TICKETING register bus Bus registeration number [WP-0191] Driver name [ ] Type [AC/ non-AC ▼] NO.OF.Seats [ ] [ADD BUS] id / bus NO / name / type / seats edit del edit del	✓ The bus can be registered using a web app instead The details added can be viewed in a table on side which was not available in the previous wireframe.

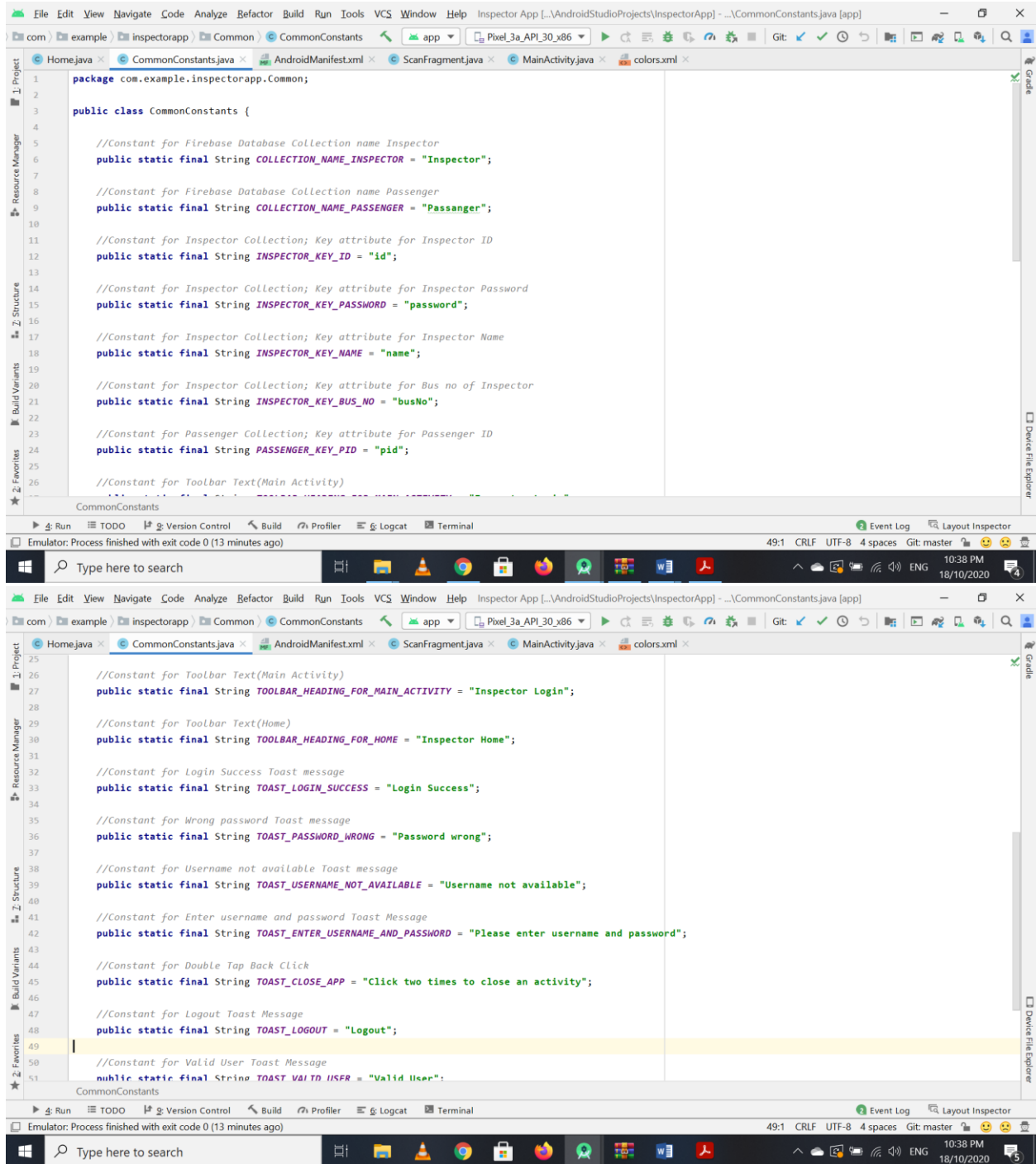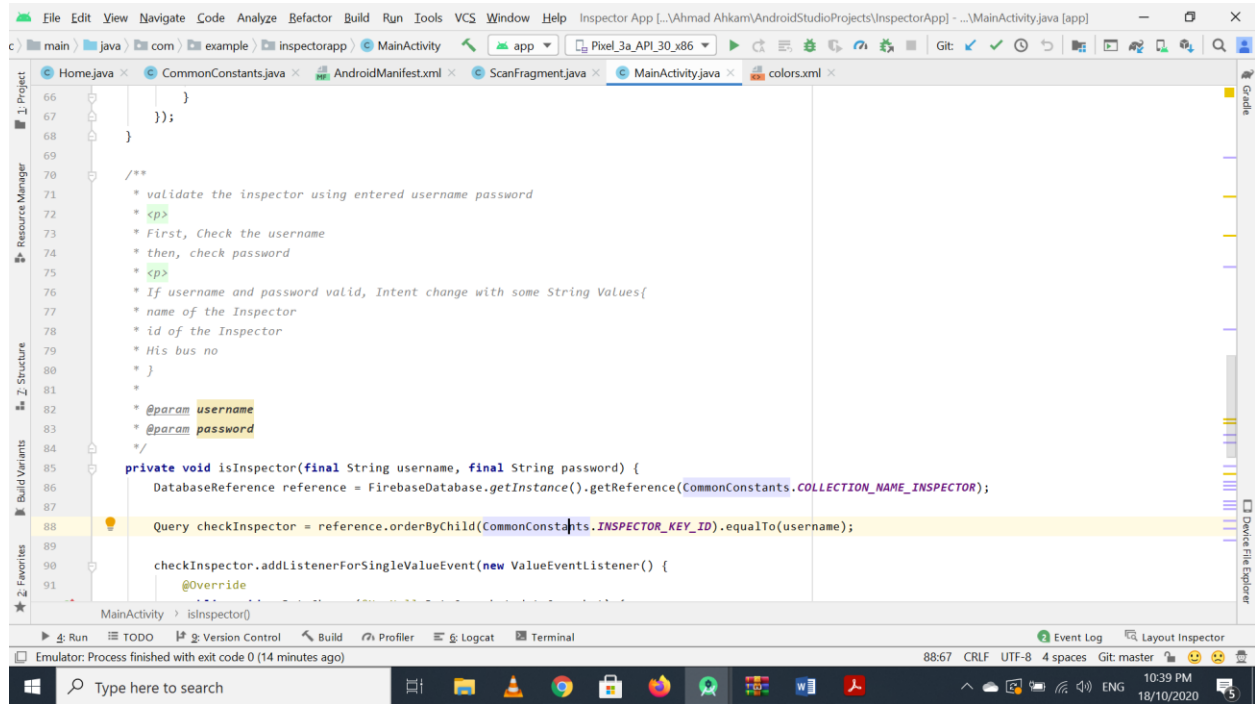|	
O O O ONLINE TICKETING Route registeration Bus Route From:- WP-0191 To:- Distance in KM Time Calculated **ADD ROUTE** Table: id | from | to | Distance | time | edit del edit del	✓ The route can be allocated using a web app instead The details added can be viewed in a table on side which was not available in the previous wireframe.
O O O ONLINE TICKETING Schedule Route ID id - xxxxxxxxxxxxxxxxxxx Bus number WP- 0190 **ADD SChedule** Table: id | Bus number | edit del edit del	✓ The schedule can be assigned using a web app instead The details added can be viewed in a table on side which was not available in the previous wireframe.

Code Refactors

Common Constants Class


```java
package com.example.inspectorapp.Common;

public class CommonConstants {

    //Constant for Firebase Database Collection name Inspector
    public static final String COLLECTION_NAME_INSPECTOR = "Inspector";

    //Constant for Firebase Database Collection name Passenger
    public static final String COLLECTION_NAME_PASSENGER = "Passanger";

    //Constant for Inspector Collection; Key attribute for Inspector ID
    public static final String INSPECTOR_KEY_ID = "id";

    //Constant for Inspector Collection; Key attribute for Inspector Password
    public static final String INSPECTOR_KEY_PASSWORD = "password";

    //Constant for Inspector Collection; Key attribute for Inspector Name
    public static final String INSPECTOR_KEY_NAME = "name";

    //Constant for Inspector Collection; Key attribute for Bus no of Inspector
    public static final String INSPECTOR_KEY_BUS_NO = "busNo";

    //Constant for Passenger Collection; Key attribute for Passenger ID
    public static final String PASSENGER_KEY_PID = "pid";

    //Constant for Toolbar Text(Main Activity)
```



```java
    //Constant for Toolbar Text(Main Activity)
    public static final String TOOLBAR_HEADING_FOR_MAIN_ACTIVITY = "Inspector Login";

    //Constant for Toolbar Text(Home)
    public static final String TOOLBAR_HEADING_FOR_HOME = "Inspector Home";

    //Constant for Login Success Toast message
    public static final String TOAST_LOGIN_SUCCESS = "Login Success";

    //Constant for Wrong password Toast message
    public static final String TOAST_PASSWORD_WRONG = "Password wrong";

    //Constant for Username not available Toast message
    public static final String TOAST_USERNAME_NOT_AVAILABLE = "Username not available";

    //Constant for Enter username and password Toast Message
    public static final String TOAST_ENTER_USERNAME_AND_PASSWORD = "Please enter username and password";

    //Constant for Double Tap Back Click
    public static final String TOAST_CLOSE_APP = "Click two times to close an activity";

    //Constant for Logout Toast Message
    public static final String TOAST_LOGOUT = "Logout";

    //Constant for Valid User Toast Message
    public static final String TOAST_VALID_USER = "Valid User";
```

Proper Comments


```java
            }
        });
    }

    /**
     * validate the inspector using entered username password
     * <p>
     * First, Check the username
     * then, check password
     * <p>
     * If username and password valid, Intent change with some String Values{
     * name of the Inspector
     * id of the Inspector
     * His bus no
     * }
     *
     * @param username
     * @param password
     */
    private void isInspector(final String username, final String password) {
        DatabaseReference reference = FirebaseDatabase.getInstance().getReference(CommonConstants.COLLECTION_NAME_INSPECTOR);

        Query checkInspector = reference.orderByChild(CommonConstants.INSPECTOR_KEY_ID).equalTo(username);

        checkInspector.addListenerForSingleValueEvent(new ValueEventListener() {
            @Override
```

Test Cases

Test Case ID	Test Scenario	Test Steps	Test Data	Expected Results	Actual Result
TU01	Check customer login with valid data	1. Go to mobile app 2. Enter username 3. Enter password 4. Click Login	Username=malith@gmail.com Password=malith123	User should login into an mobile application	As expected
TU02	Check customer login with invalid data	1. Go to mobile app 2. Enter invalid username 3. Enter invalid password 4. Click Login	Username=malith Password=123	User should not login into an mobile application	As expected
TU03	Recharge the token with valid data	1. Go to the mobile application 2. Enter Card Number 3. Enter Expire Date 4. Enter CCV 5. Enter Amount 6. Click Recharge	Card Number=451-1001-3452-16 Expire Date=20/12 CCV=444 Amount=100	Recharge the Token with Rs.100	As expected
TU04	Recharge the token with invalid data	1. Go to the mobile application 2. Enter invalid Card Number 3. Enter Expire Date 4. Enter CCV 5. Enter Amount 6. Click Recharge	Card Number=451-1001-3452-15S Expire Date=20/12 CCV=444 Amount=100	Not Recharge the Token with RS.100	As expected
TU05	Request the session with valid data	1. Go to the mobile application 2. Select the Session Type 3. Enter the special request 4. Click send	Session Type=office package Any special Request=every day I travel , so I want to session	User's session Request is successfully send.	As expected
TU06	Request the session with invalid data	1. Go to the mobile application 2. Not Select the Session Type 3. Enter the special request 4. Click send	Session Type=null Any special Request=every day I travel , so I want to session	User's session Request is not successfully send.	As expected
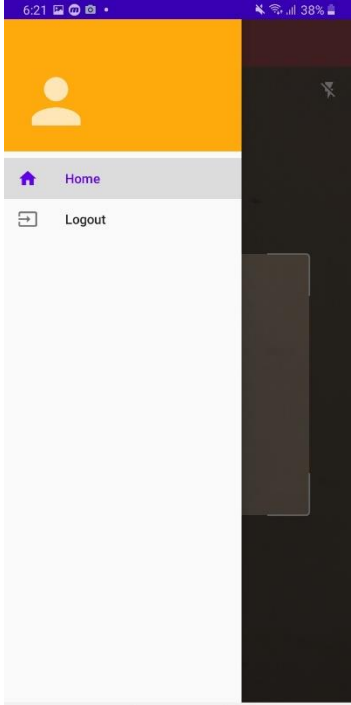
Test Case ID	Test Scenario	Test Steps	Test Data	Expected Results	Actual Result
TU07	Check Inspector login with valid data	1. Open Inspector app 2. Enter username 3. Enter password 4. Click Login	Username = "P025" Password = "perera123"	User should login into app and toast message prompt.	As expected
TU08	Check Inspector login with invalid data	1. Open Inspector app 2. Enter username 3. Enter password 4. Click Login	Username = "ABC12" Password = "123t1"	User can't login into app	As expected
TU09	Check valid qr code in Inspector app	1. Login to passenger app 2. Scan passenger qr code in inspector app	Passenger QR code	Valid passenger with a success message	As expected
TU10	Check invalid qr code in Inspector app	1. Login to passenger app 2. Scan passenger qr code in inspector app	Passenger QR code	Invalid passenger with a error message and "Beep" sound.	As expected

TU11	Check valid passenger when entering the bus and exit the bus.	1. Passenger open the token interface in phone. 2. Scan the token with the scanner in the bus.	Passenger QR code	Valid passenger with a "Welcome" Message tone. And when passenger gets down "good bye" tone played.	As expected
TU12	Check invalid passenger when entering the bus and exit the bus.	1. Passenger open the token interface in phone. 2. Scan the token with	Passenger QR code	Invalid passenger when entering or exit "Beep" Sound.	As expected
TU13	Check valid passenger with balance higher than the minimum amount when entering the bus .	1. Passenger scan the token. 2. Check for the valid passenger. 3. Check the balance.	Passenger QR code.	Valid passenger with the amount more than the initial minimum amount. "Welcome" Tone.	As expected
TU14	Check valid passenger with balance lower than the minimum amount when entering the bus .	1. Passenger scan the token. 2. Check for the valid passenger. 3. Check the balance.	Passenger QR code	Valid passenger with the amount lesser than the initial minimum amount. "Beep" Tone.	As Expected.
TU15	Check Admin Login with valid details	1. Check Admin Login with valid details	AdminEmail: hranasinghe97@gmail.com Password:gtgtgddt2	Admin Log in to the application	As expected

TU16	Add Bus to the system	1. Go to site 2.Go to Add new bus 3.Enter Valid details 4.Click add bus button	**Bus Name:Sinhagiri Bus Number:WQ-2030 Bus type:AC Inspector:Kasun Rajitha Route ID:177 Seat Capacity:45 Driver Name:Liyange Departure Time:12.00am Journey date:12/2/2020 Start Location:Matara End Location:Galle Dropping Points:Weligama**	Bus added to the system	As expected
TU17	Add Inspector	1.Go to inspectors 2.Click add inspector button 3.Enter inspector name 4.add inspector button click	*Name:Vikum gunasinghe*	Inspector added to the system	As expected
TU18	Check Admin Login with valid details	Check Admin Login with valid details	AdminEmail: hranasinghe97@gmail.com Password:llllllll	Admin Log in to the application	Admin can't log In to the system because password is wrong
TU19	Add Bus to the system	1.Go to site 2.Go to Add new bus 3.Enter Valid details 4.Click add bus button	**Bus Name:Sinhagiri Bus Number: Bus type:AC Inspector:Kasun Rajitha Route ID:177 Seat Capacity:45 Driver Name:Liyange Departure Time:12.00am Journey date:12/2/2020 Start Location:Matara End Location:Galle Dropping Points:Weligama**	Bus added to the system	Bus didn't add because bus number is empty

TU20	Add passenger in to the system	1.go to site 2.click add passenger 3.enter details 4.click add passenger	Name:kasun Ranasinghe Email:kasun@gmail.com PNumber:0718774656 NIC:987654321V Amount:1000 Password:1234 Confirm pass:12345678	Add passenger in to the system	Passenger did't add because password is not matching

Implemented UI

Topic	UI
Inspector Login	
Inspector Drawer	

Inspector QR Scanner	
Bus QR Scanner	
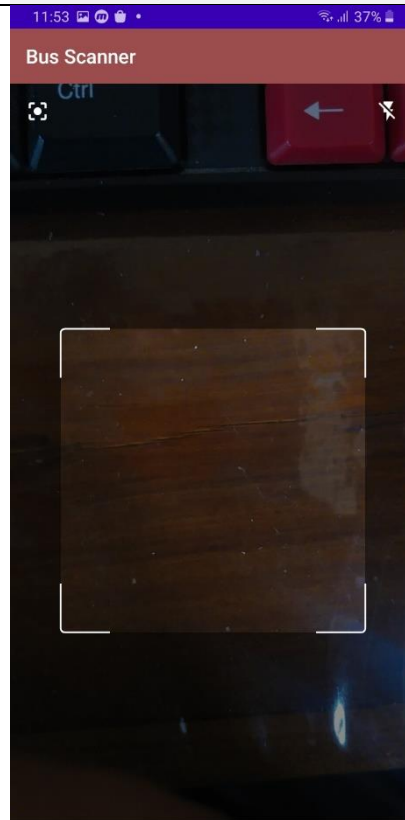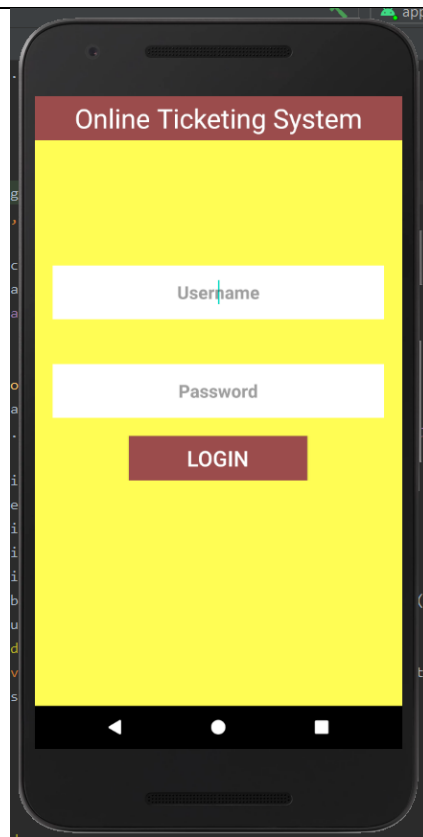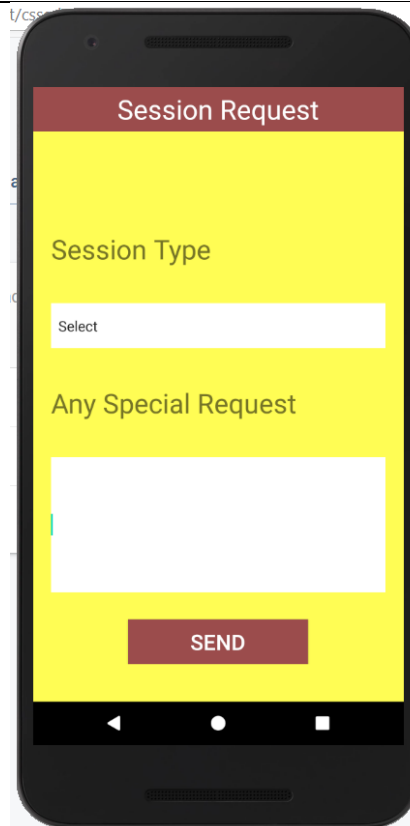
Passenger Login	
Passenger App Home	

Season Request	
Token Payment	

Trip History

Tip History

Date
From To
Start Time to End Time
Route Number
Amount

Date
From To
Start Time to End Time
Route Number
Amount

Add Bus

EasyBooking

Malith
superadmin

- Dashboard
- My Buses
- All Bookings
- My Bookings `new`
- People
- Locations
- Travels
- Logout

Logout

Add new bus (Primary details)

Name *

Jet Liner 01

Bus Number *

NC_5656

Enter in the format of ba-2-pa

☑ is Available

Bus type

AC

Travels

Jet Liners

Fare (Rs) *

323

Seat Capacity

33

Continue to next form

EasyBooking ☰   Logout

Malith
● superadmin   ✎

🚌 Dashboard

🚌 My Buses   ‹

◉ All Bookings

📅 My Bookings   new

👥 People   ‹

◉ Locations

📋 Travels

↪ Logout

Add new bus (Additional details)

Additional Features

Separate features with commas

Departure Time *

12:50 PM   🕐

Journey date

10/18/2020   📅

Start Location

Kandy   ⌄

End Location

Malabe   ⌄
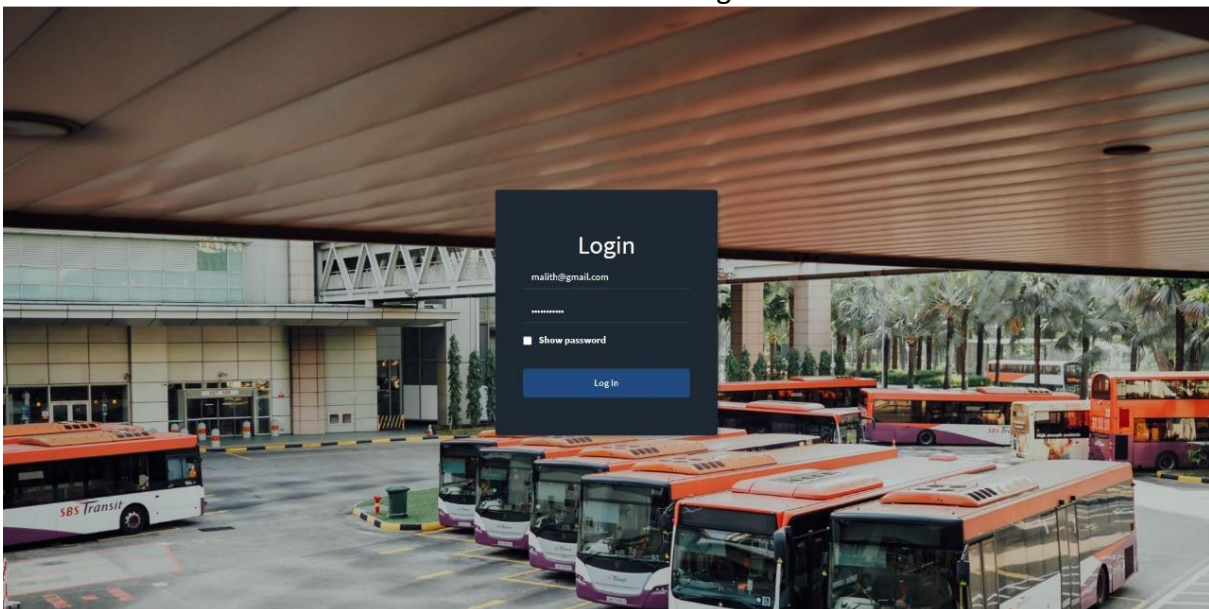
Boarding Points

Enter names separated by comma

Dropping Points

Enter names separated by comma

Back   Continue to submit

Service Provider Login

Login

malith@gmail.com

••••••••••

☐ Show password

Log In

Add Route

EasyBooking

Malith
● superadmin

- Dashboard
- My Buses
- All Bookings
- My Bookings `new`
- People
- Locations
- Travels
- Logout

Logout

Add Location

Name

Location Name

District

Ampara

Select District
Ampara
Anuradhapura
Badulla
Batticaloa
Colombo
Galle
Gampaha
Hambantota
Jaffna
Kalutara
Kandy
Kegalle
Kilinochchi
Kurunegala
Mannar
Matale
Matara
Moneragala
Mullaitivu

SLIIT Project

Add Inspector

← → C ① localhost:3000/travels

EasyBooking

- All Buses
- Add new bus
- All Bookings
- Passengers
- Routes
- Inspectors
- Logout

Travels

Add Inspector

Inspectors

Show 10 result per page

Search in records...

S.N	ID	Name	Action
1	5f8bc6e3a144ad491d184315	Kasun Ranasinghe	
2	5f8bba9da144ad491d184303	Pasindu	
3	5f8bd6e6a144ad491d184324	Perara	
4	5f8bdb7fa144ad491d184326	Viraj Sadaruwan	

Add Passenger