



SC2006 – Software Engineering

Lab 2 Deliverables

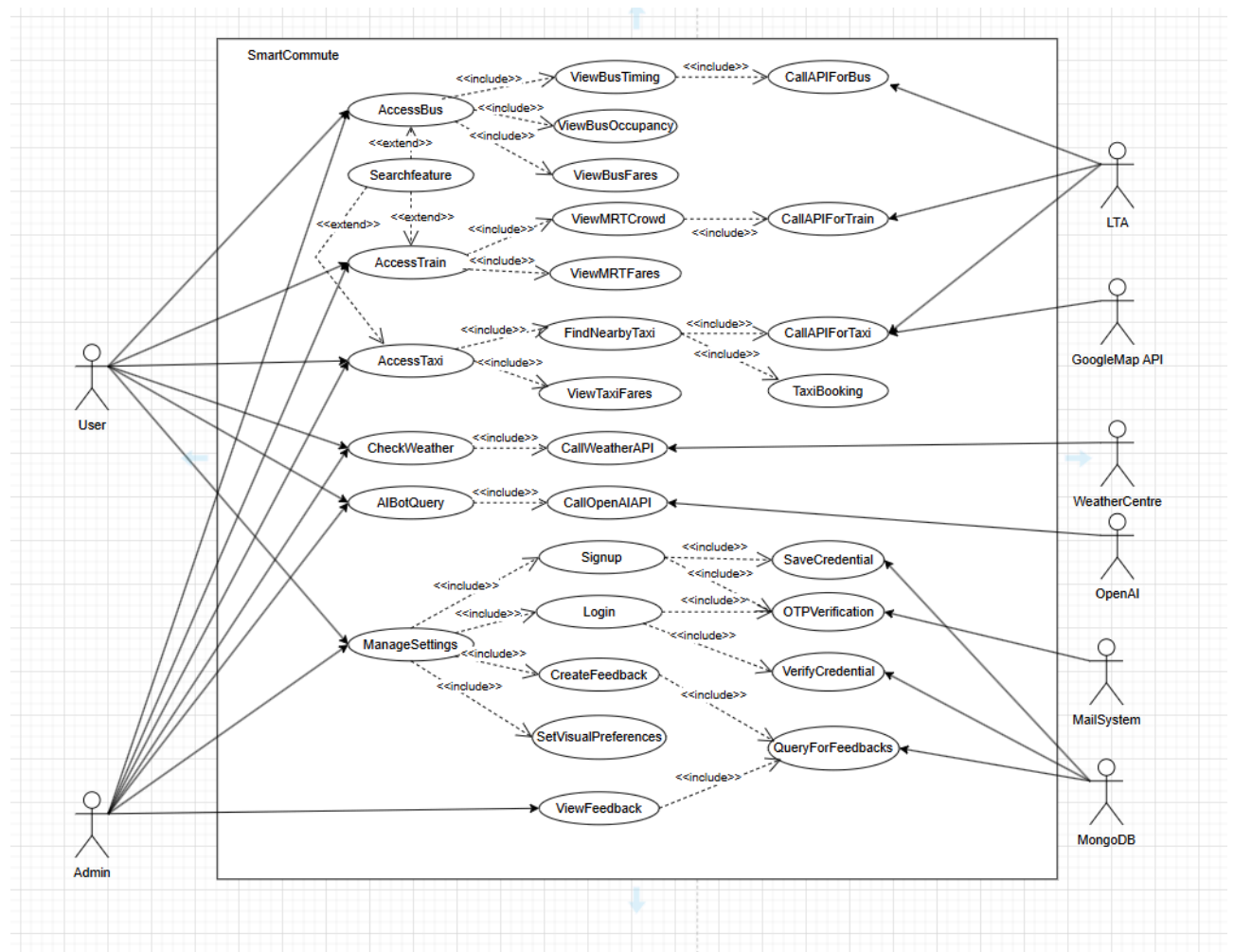
Lab Group	SCEX
Team	SmartCommute
Members	CHAN ZI HAO (U2222242B)
	AMANDA RAE JOSEPHINE (U2420764F)
	AW YONG WING KIAN, ALVIN (U2223300F)
	IVAN CHENG LI HAO (U2221078L)
	JACE SEOW WEN HUI (U2222469F)

Table of Contents

Refined Use Case Model, consisting of Use Case Diagram and Use Case Description.	3
A. Complete Use Case Diagram.....	3
B. Complete Use Case Description.....	4
Class Diagram of Entity Classes.....	21
Key Boundary Classes and Control Classes.....	22
Sequence Diagrams of Some Use Cases.....	23
Initial Dialog Map.....	27

Refined Use Case Model, consisting of Use Case diagram and Use Case descriptions

A. Complete Use Case Diagram



B. Complete Use Case Description

Use Case ID:	#1		
Use Case Name:	AccessBus		
Created By:	Ivan Cheng Li Hao	Last Updated By:	
Date Created:	6/2/2025	Date Last Updated:	

Actor:	User
Description:	This use case allows access to bus information
Preconditions:	.
Postconditions:	
Priority:	High
Frequency of Use:	High
Flow of Events:	1. User selects and views different bus information
Alternative Flows:	
Exceptions:	
Includes:	ViewBusTiming, ViewBusFares and ViewBusOccupancy
Special Requirements:	
Assumptions:	
Notes and Issues:	

Use Case ID:	#1.1		
Use Case Name:	ViewBusTiming		
Created By:	Ivan Cheng Li Hao	Last Updated By:	
Date Created:	6/2/2025	Date Last Updated:	

Actor:	User and Admin
Description:	This use case allows the user to view all bus stops within 3km radius and bus arriving time.
Preconditions:	1. User device has working WiFi or Cellular connection 2. Connection established between system and LTA(Bus). 3. LTA(Bus) server online
Postconditions:	1. Display a list of bus stops with their respective bus arriving time
Priority:	High
Frequency of Use:	High
Flow of Events:	1. User selects the bus stop he/she wants. 2. Check for the bus arriving time towards the selected bus stop.
Alternative Flows:	
Exceptions:	EX1: Failed to fetch API 1. An error message will pop up.
Includes:	CallAPIForBus
Special Requirements:	
Assumptions:	
Notes and Issues:	

Use Case ID:	#1.2		
Use Case Name:	ViewBusOccupancy		
Created By:	Ivan Cheng Li Hao	Last Updated By:	
Date Created:	6/2/2025	Date Last Updated:	

Actor:	User and Admin
Description:	This use case allows the user to view all bus stops within 3km radius and bus occupancy level.
Preconditions:	1. User device has working WiFi or Cellular connection 2. Connection established between system and LTA(Bus). 3. LTA(Bus) server online
Postconditions:	1. Display a list of bus stops with their respective bus occupancy level
Priority:	High
Frequency of Use:	High
Flow of Events:	1. User selects the bus stop he/she wants. 2. Check for arriving bus occupancy levels.
Alternative Flows:	
Exceptions:	EX1: Failed to fetch API 1. An error message will pop up.
Includes:	CallAPIForBus
Special Requirements:	
Assumptions:	
Notes and Issues:	

Use Case ID:	#1.2.1		
Use Case Name:	CallAPIForBus		
Created By:	Ivan Cheng Li Hao	Last Updated By:	
Date Created:	6/2/2025	Date Last Updated:	

Actor:	LTA
Description:	This use case manages connections between the system and LTA(Bus).
Preconditions:	1. User device has working WiFi or Cellular connection 2. Connection established between system and LTA(Bus). 3. LTA(Bus) server is online
Postconditions:	
Priority:	High
Frequency of Use:	High

Flow of Events:	1. Forward API request to LTA(Bus). 2. Both server received request and send acknowledgement and bus arriving time and occupancy level data back
Alternative Flows:	
Exceptions:	EX1: Unable to reach server 1. No acknowledgement send back to system which leads to exception in UseCase #1.1 and #1.2
Includes:	
Special Requirements:	
Assumptions:	
Notes and Issues:	

Use Case ID:	#1.3		
Use Case Name:	ViewBusFares		
Created By:	Ivan Cheng Li Hao	Last Updated By:	
Date Created:	6/2/2025	Date Last Updated:	

Actor:	User and Admin
Description:	This use case allows user to view all required bus fares (Age Category, Distance, Cash/Card, Bus Service Types and etc)
Preconditions:	
Postconditions:	1. Display bus fares table with different prices, distances, age groups and bus service types.
Priority:	Moderate
Frequency of Use:	Moderate
Flow of Events:	1. User will need to select the bus fare types (Student, Senior Citizen, Adult or Workforce Transport Concession Scheme) 2. After selection, it will display a table with column attributes like "Distance" and "Bus Service Type" with Price generated below cells.
Alternative Flows:	
Exceptions:	
Includes:	
Special Requirements:	
Assumptions:	
Notes and Issues:	

Use Case ID:	#2		
Use Case Name:	Searchfeature		
Created By:	Ivan Cheng Li Hao	Last Updated By:	
Date Created:	6/2/2025	Date Last Updated:	

Actor:	None
Description:	This use case allows users to search for common info such as “BusStopNames”, “MRT Maps”, “Taxi Hotlines”, “MRT Operating Hours”
Preconditions:	
Postconditions:	1. Display the necessary information based on the different selection.
Priority:	Low
Frequency of Use:	Low
Flow of Events:	1. User will scroll or search the common info they want. 2. The user will be required to select. 3. After selecting, it displays the necessary information.
Alternative Flows:	
Exceptions:	
Includes:	
Special Requirements:	
Assumptions:	
Notes and Issues:	

Use Case ID:	#3		
Use Case Name:	AccessTrain		
Created By:	Ivan Cheng Li Hao	Last Updated By:	
Date Created:	6/2/2025	Date Last Updated:	

Actor:	User and Admin
Description:	This use case allows access to train information
Preconditions:	
Postconditions:	
Priority:	High
Frequency of Use:	High
Flow of Events:	1. User selects and views different train information
Alternative Flows:	
Exceptions:	
Includes:	ViewMRTFares and ViewMRTCrowd
Special Requirements:	
Assumptions:	
Notes and Issues:	

Use Case ID:	#3.1		
Use Case Name:	ViewMRTCrowd		
Created By:	Ivan Cheng Li Hao	Last Updated By:	

Date Created:	6/2/2025	Date Last Updated:	
---------------	----------	--------------------	--

Actor:	User and Admin
Description:	This use case allows user to check different MRT Station Crowd indicated by 3 colors (Green, Orange and Red)
Preconditions:	1. User device has working WiFi or Cellular connection 2. Connection established between system and LTA(Train). 3. LTA(Train) server online
Postconditions:	1. Display MRT Stations Crowd level
Priority:	High
Frequency of Use:	High
Flow of Events:	1. The system will list all MRT stations with the nearest station at the top. 2. User can press each station to see the crowd level indicated (Green, Orange and Red)
Alternative Flows:	
Exceptions:	EX1: Failed to fetch API: 1. An error message will pop up.
Includes:	CallAPIForTrain
Special Requirements:	
Assumptions:	
Notes and Issues:	

Use Case ID:	#3.1.1		
Use Case Name:	CallAPIForTrain		
Created By:	Ivan Cheng Li Hao	Last Updated By:	
Date Created:	6/2/2025	Date Last Updated:	

Actor:	LTA
Description:	This use case manages connections between the system and LTA(Train).
Preconditions:	1. User device has working WiFi or Cellular connection 2. Connection established between system and LTA(Train). 3. LTA(Train) server online
Postconditions:	
Priority:	High
Frequency of Use:	High
Flow of Events:	1. Forward API request to LTA(Train). 2. LTA(Train) receives request and sends acknowledgement and train crowd data back
Alternative Flows:	
Exceptions:	EX1: Unable to reach server 1. 1. No acknowledgement send back to system which leads to exception in UseCase #3.1

Includes:	
Special Requirements:	
Assumptions:	
Notes and Issues:	

Use Case ID:	#3.2		
Use Case Name:	ViewMRTFares		
Created By:	Ivan Cheng Li Hao	Last Updated By:	
Date Created:	6/2/2025	Date Last Updated:	

Actor:	User and Admin		
Description:	This use case allows users to view all expected train fares based on Age Group, Distance, Time and etc.		
Preconditions:			
Postconditions:	1. Display train fares table with different prices, distances, age groups and timing		
Priority:	Moderate		
Frequency of Use:	Moderate		
Flow of Events:	1. User will need to select the train fare types (Student, Senior Citizen, Adult or Workforce Transport Concession Scheme) 2. After selection, it will display a table with column attributes like "Distance" and "Timing" with Price generated below cells.		
Alternative Flows:			
Exceptions:			
Includes:			
Special Requirements:			
Assumptions:			
Notes and Issues:			

Use Case ID:	#4		
Use Case Name:	AccessTaxi		
Created By:	Ivan Cheng Li Hao	Last Updated By:	
Date Created:	6/2/2025	Date Last Updated:	

Actor:	User and Admin		
Description:	This use case allows access to taxi information		
Preconditions:			
Postconditions:			
Priority:	Moderate		
Frequency of Use:	Low		
Flow of Events:	1. User selects and views different taxi information		

Alternative Flows:	
Exceptions:	
Includes:	ViewTaxiFares and FindNearbyTaxi
Special Requirements:	
Assumptions:	
Notes and Issues:	

Use Case ID:	#4.1		
Use Case Name:	FindNearbyTaxi		
Created By:	Ivan Cheng Li Hao	Last Updated By:	
Date Created:	6/2/2025	Date Last Updated:	

Actor:	User and Admin
Description:	This use case allows the user to see all the nearby taxis around him/her on a map.
Preconditions:	1. User device has working WiFi or Cellular connection 2. Connection established between system and LTA(Taxi) and GoogleMap API. 3. LTA(Taxi) and GoogleMap server online
Postconditions:	1. Display a map of your location coordinates with nearby taxis showing.
Priority:	Moderate
Frequency of Use:	Low
Flow of Events:	
Alternative Flows:	
Exceptions:	EX1: Failed to fetch API 1. An error message will pop up.
Includes:	CallAPIForTaxi and TaxiBooking
Special Requirements:	
Assumptions:	
Notes and Issues:	

Use Case ID:	#4.1.1		
Use Case Name:	CallAPIForTaxi		
Created By:	Ivan Cheng Li Hao	Last Updated By:	
Date Created:	6/2/2025	Date Last Updated:	

Actor:	LTA
--------	-----

Description:	This use case manages connection between the system, LTA(Taxi) and GoogleMap API
Preconditions:	1. User device has working WiFi or Cellular connection 2. Connection established between system and LTA(Taxi) and GoogleMap API. 3. LTA(Taxi) server online
Postconditions:	
Priority:	Moderate
Frequency of Use:	Low
Flow of Events:	1. Forward API request to LTA(Taxi) and GoogleMap API. 2. Both servers receive request and send acknowledgement, TaxiCoordinates and map back.
Alternative Flows:	
Exceptions:	EX1: Unable to reach server 1. 1. No acknowledgement send back to system which leads to exception in UseCase #4.1
Includes:	
Special Requirements:	
Assumptions:	
Notes and Issues:	

Use Case ID:	#4.1.2		
Use Case Name:	TaxiBooking		
Created By:	Ivan Cheng Li Hao	Last Updated By:	
Date Created:	6/2/2025	Date Last Updated:	

Actor:	LTA
Description:	This use case allows users to call taxi service provider
Preconditions:	1. User device has working WiFi or Cellular connection
Postconditions:	
Priority:	Moderate
Frequency of Use:	Low
Flow of Events:	1.User selects booking button 2.After selecting, it will display an option to call the taxi service provider.
Alternative Flows:	
Exceptions:	
Includes:	
Special Requirements:	
Assumptions:	
Notes and Issues:	

Use Case ID:	#4.2		
Use Case Name:	ViewTaxiFares		
Created By:	Ivan Cheng Li Hao	Last Updated By:	
Date Created:	6/2/2025	Date Last Updated:	

Actor:	User and Admin
Description:	This use case allows user to view all expected taxi fares such as “Flag-down Fare”, “Peak Periods”, “Location Surcharges” and “Booking Fees”.
Preconditions:	
Postconditions:	1. Display Taxi Fare Table with its respective prices
Priority:	Moderate
Frequency of Use:	Low
Flow of Events:	1.User will need to select the type of fare they are looking to find from the dropdown. 2. After selecting, the fare table will be displayed.
Alternative Flows:	
Exceptions:	
Includes:	
Special Requirements:	
Assumptions:	
Notes and Issues:	

Use Case ID:	#5		
Use Case Name:	CheckWeather		
Created By:	Ivan Cheng Li Hao	Last Updated By:	
Date Created:	6/2/2025	Date Last Updated:	

Actor:	User and Admin
Description:	This use case creates a weather panel for display.
Preconditions:	1. User device has working WiFi or Cellular connection 2. Connection established between system and WeatherCentre 3. WeatherCentre server online
Postconditions:	1. Display Weather panel with today’s temperatures and weather conditions.
Priority:	Low
Frequency of Use:	Low
Flow of Events:	
Alternative Flows:	
Exceptions:	EX1: Failed to fetch API 1. An error message will pop up.

Includes:	CallWeatherAPI
Special Requirements:	
Assumptions:	
Notes and Issues:	

Use Case ID:	#5.1		
Use Case Name:	CallWeatherAPI		
Created By:	Ivan Cheng Li Hao	Last Updated By:	
Date Created:	6/2/2025	Date Last Updated:	

Actor:	WeatherCentre		
Description:	This use case manages connection between the system and WeatherCentre.		
Preconditions:	1. User device has working WiFi or Cellular connection 2. Connection established between system and WeatherCentre. 3. WeatherCentre server online		
Postconditions:			
Priority:	Low		
Frequency of Use:	Low		
Flow of Events:	1. Forward Request to WeatherCentre using API. 2. WeatherCentre sends acknowledgement and weather data back to the system.		
Alternative Flows:			
Exceptions:	EX1: Unable to reach server 1. No acknowledgement send back to system which leads to exception in UseCase #5		
Includes:			
Special Requirements:			
Assumptions:			
Notes and Issues:			

Use Case ID:	#6		
Use Case Name:	AIBotQuery		
Created By:	Ivan Cheng Li Hao	Last Updated By:	
Date Created:	18/2/2025	Date Last Updated:	

Actor:	User and Admin		
Description:	This use case utilizes an AI Chatbot for the user to ask AIBot enquiries.		
Preconditions:	1. User device has working WiFi or Cellular connection 2. Connection established between system and OpenAI.		

	3. OpenAI server online
Postconditions:	1. Display answers to the questions.
Priority:	Low
Frequency of Use:	Low
Flow of Events:	1. List of example questions will be displayed upon entering the chatroom 2. Users ask any questions listed, AI Bot will reply based on the input prompt.
Alternative Flows:	
Exceptions:	EX1: Failed to communicate with API 1. An error message will pop up.
Includes:	CallOpenAIAPI
Special Requirements:	
Assumptions:	
Notes and Issues:	

Use Case ID:	#6.1		
Use Case Name:	CallOpenAIAPI		
Created By:	Ivan Cheng Li Hao	Last Updated By:	
Date Created:	18/2/2025	Date Last Updated:	

Actor:	User
Description:	This use case manages the AI Chatbot communication. By Processing send and reply messages between the system and OpenAI
Preconditions:	1. User device has working WiFi or Cellular connection 2. Connection established between system and OpenAI. 3. OpenAI server online
Postconditions:	
Priority:	Low
Frequency of Use:	Low
Flow of Events:	1. Questions will be sent to the OpenAI server. 2. Based on the input, OpenAI will give the answer. 3. Result will be sent back for display.
Alternative Flows:	
Exceptions:	EX1: Unable to reach server 1. No response send back to system which leads to exception in UseCase #6
Includes:	
Special Requirements:	
Assumptions:	
Notes and Issues:	

Use Case ID:	#7		
Use Case Name:	ManageSettings		
Created By:	Ivan Cheng Li Hao	Last Updated By:	

Date Created:	18/2/2025	Date Last Updated:	
---------------	-----------	--------------------	--

Actor:	User and Admin
Description:	This use case manages settings which consists of Signup, Login, writing Feedbacks and setting VisualPreferences
Preconditions:	
Postconditions:	
Priority:	Low
Frequency of Use:	Low
Flow of Events:	1. The user selects the specific setting he/she wants to do. 2. The user will be redirected to new page for specific settings
Alternative Flows:	
Exceptions:	
Includes:	Signup, Login, CreateFeedback and SetVisualPreferences
Special Requirements:	
Assumptions:	
Notes and Issues:	

Use Case ID:	#7.1		
Use Case Name:	Signup		
Created By:	Ivan Cheng Li Hao	Last Updated By:	
Date Created:	18/2/2025	Date Last Updated:	

Actor:	User
Description:	This use case allows user to sign up for an account
Preconditions:	1. User device has working WiFi or Cellular connection 2. Connection established between system, MongoDB and MailSystem 3. MongoDB and MailSystem servers are online 4. The user should have a registered email.
Postconditions:	
Priority:	Mid
Frequency of Use:	Low
Flow of Events:	1. User enters its own registered email and preferred username and password. 2. After pressing “Sign up”, the user will receive OTP from email. 3. User keys in OTP Code into the system. 4. User successfully signed up and will be redirected to the Login Page.
Alternative Flows:	
Exceptions:	EX1: Invalid or missing required information 1. Error Message will be displayed
Includes:	SaveCredentials and OTPVerification
Special Requirements:	System needs to validate input information
Assumptions:	
Notes and Issues:	

Use Case ID:	#7.1.1		
Use Case Name:	SaveCredential		
Created By:	Ivan Cheng Li Hao	Last Updated By:	
Date Created:	6/2/2025	Date Last Updated:	

Actor:	MongoDB
Description:	This use case allows user credential to be saved in database
Preconditions:	1. User device has working WiFi or Cellular connection 2. Connection established between system and MongoDB 3. MongoDB server is online
Postconditions:	
Priority:	High
Frequency of Use:	Low
Flow of Events:	1. Signup will send registered information to MongoDB 2. MongoDB will store the information data in a table. 3. Returns acknowledgement
Alternative Flows:	
Exceptions:	
Includes:	
Special Requirements:	
Assumptions:	Sufficient Data Storage in MongoDB
Notes and Issues:	

Use Case ID:	#7.2		
Use Case Name:	Login		
Created By:	Ivan Cheng Li Hao	Last Updated By:	
Date Created:	18/2/2025	Date Last Updated:	

Actor:	User and Admin
Description:	This use case allows users to log into the system
Preconditions:	1. User should have created an account 2. User device has working WiFi or Cellular connection 3. Connection established between system, MongoDB and MailSystem. 4. MongoDB and MailSystem servers online
Postconditions:	
Priority:	High
Frequency of Use:	High
Flow of Events:	1. User will fill in their Username and Password 2. Selects Login and proceed to OTP Verification 3. User keys in OTP Code obtained from email 4. The user will successfully log into the system.
Alternative Flows:	

Exceptions:	EX1: User provides invalid credentials 1. Wrong Username or Password message will display EX2: User login without Username or Password 1. Please Enter Username or Password message will display
Includes:	OTPVerification and VerifyCredential
Special Requirements:	
Assumptions:	
Notes and Issues:	

Use Case ID:	#7.2.1		
Use Case Name:	OTPVerification		
Created By:	Ivan Cheng Li Hao	Last Updated By:	
Date Created:	18/2/2025	Date Last Updated:	

Actor:	MailSystem		
Description:	This use case manages the send and request OTP process between the system and mail		
Preconditions:	1. User device has working WiFi or Cellular connection 2. Connection established between system and MailSystem. 3. MailSystem server is online 4. The user already have a registered email account		
Postconditions:			
Priority:	High		
Frequency of Use:	High		
Flow of Events:	1. OTP Requests from the system will be sent to MailSystem. 2. MailSystem will send the OTP message to the user mailbox.		
Alternative Flows:			
Exceptions:	EX1: Failed to receive OTP message 1. No OTP received, user will need to resend request		
Includes:			
Special Requirements:			
Assumptions:			
Notes and Issues:			

Use Case ID:	#7.2.2		
Use Case Name:	VerifyCredential		
Created By:	Ivan Cheng Li Hao	Last Updated By:	
Date Created:	18/2/2025	Date Last Updated:	

Actor:	MongoDB		
Description:	This use case does user account security verification.		
Preconditions:	1. User should have created an account 2. User device has working WiFi or Cellular connection 3. Connection established between system and MongoDB		

	4. MongoDB server is online
Postconditions:	
Priority:	High
Frequency of Use:	High
Flow of Events:	1. After login, it will check if credential exists in the database then return acknowledgement to Login
Alternative Flows:	
Exceptions:	EX1: Credential does not exists 1. Returns error acknowledgment to Login
Includes:	
Special Requirements:	
Assumptions:	
Notes and Issues:	

Use Case ID:	#7.3		
Use Case Name:	CreateFeedback		
Created By:	Ivan Cheng Li Hao	Last Updated By:	
Date Created:	18/2/2025	Date Last Updated:	

Actor:	User and Admin
Description:	This use case allows the user to create feedback about the system
Preconditions:	1. The user should be logged in. 2. User device has working WiFi or Cellular connection 3. Connection established between system and MongoDB. 4. MongoDB server online
Postconditions:	
Priority:	Low
Frequency of Use:	Low
Flow of Events:	1. The user write feedback on the provided textbox 2. Select Send Feedback
Alternative Flows:	
Exceptions:	EX1: MongoDB server not online 1. Error message “Unable to send feedback”
Includes:	QueryForFeedbacks
Special Requirements:	
Assumptions:	
Notes and Issues:	

Use Case ID:	#7.3.1		
Use Case Name:	QueryForFeedbacks		
Created By:	Ivan Cheng Li Hao	Last Updated By:	
Date Created:	18/2/2025	Date Last Updated:	

Actor:	MongoDB
--------	---------

Description:	This use case manages feedback data between the system and MongoDB
Preconditions:	1. User device has working WiFi or Cellular connection 2. Connection established between system and MongoDB. 3. MongoDB server online
Postconditions:	
Priority:	Low
Frequency of Use:	Low
Flow of Events:	1. Receive data send request from system. 2. Save feedback data into the database. 3. Returns acknowledgement after storing
Alternative Flows:	AF-Step1: Receive data get request from system 1. Get feedback data from the database. 2. Return feedback data and acknowledgement to the system
Exceptions:	
Includes:	
Special Requirements:	
Assumptions:	
Notes and Issues:	

Use Case ID:	#7.4		
Use Case Name:	SetVisualPreferences		
Created By:	Ivan Cheng Li Hao	Last Updated By:	
Date Created:	18/2/2025	Date Last Updated:	

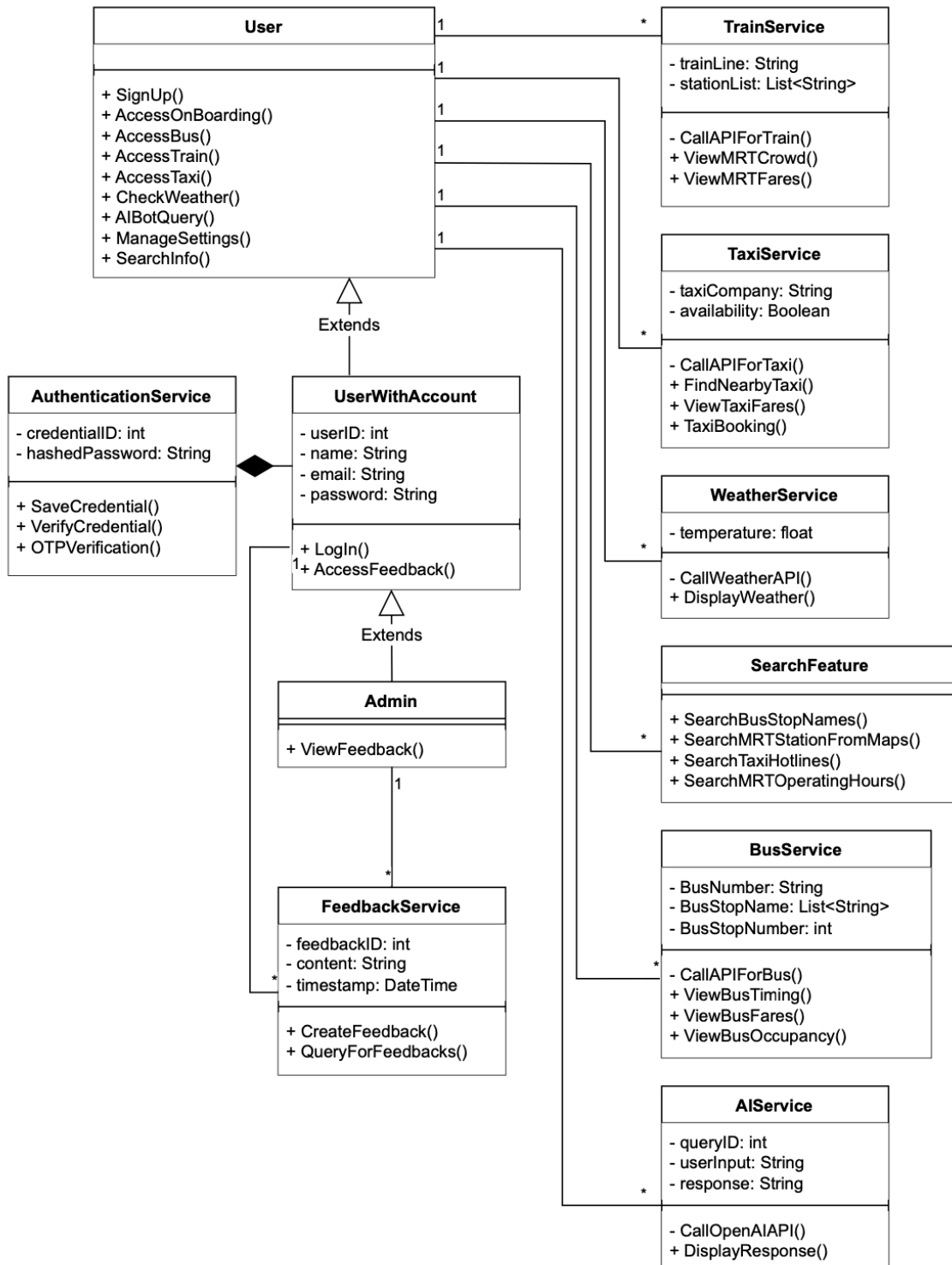
Actor:	User and Admin
Description:	This use case allows the user to set app settings like Font Size, Theme Color and Dark/Light Mode
Preconditions:	
Postconditions:	1. Application visuals will change
Priority:	Low
Frequency of Use:	Low
Flow of Events:	1. List of setting tabs displayed 2. The user will select the setting he/she wants to set.
Alternative Flows:	
Exceptions:	
Includes:	
Special Requirements:	
Assumptions:	
Notes and Issues:	

Use Case ID:	#8		
Use Case Name:	ViewFeedback		
Created By:	Ivan Cheng Li Hao	Last Updated By:	
Date Created:	18/2/2025	Date Last Updated:	

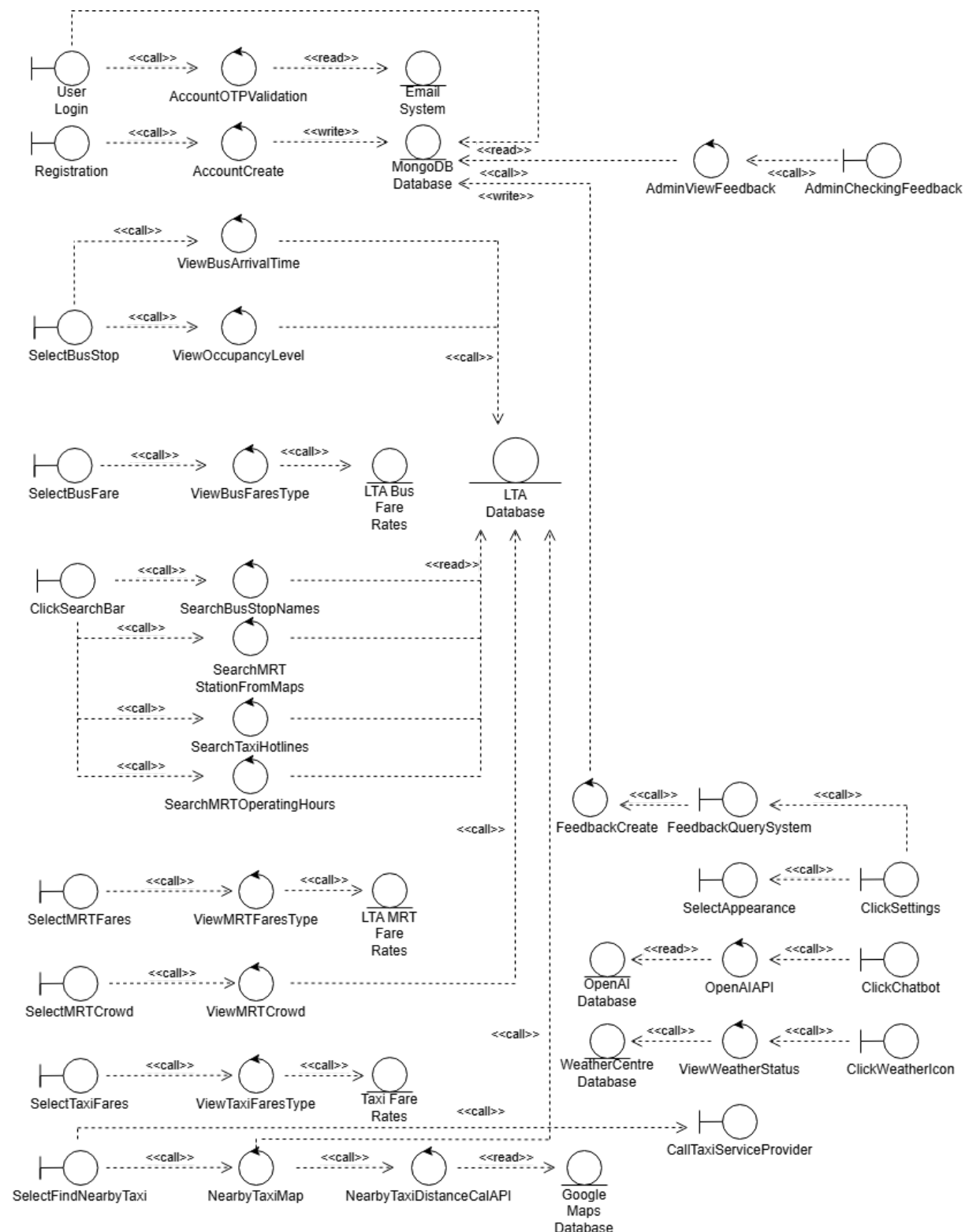
Actor:	Admin
--------	-------

Description:	This use case allows the admin to view feedback about the system from users.
Preconditions:	<ol style="list-style-type: none"> 1. The user should have logged in as admin 2. Admin device has working WiFi or Cellular connection 3. Connection established between system and MongoDB. 4. MongoDB server online
Postconditions:	1. Display a list of feedbacks
Priority:	Low
Frequency of Use:	Low
Flow of Events:	
Alternative Flows:	
Exceptions:	EX1: MongoDB server not online <ol style="list-style-type: none"> 1. Error message “Unable to retrieve feedback”
Includes:	QueryForFeedbacks
Special Requirements:	
Assumptions:	
Notes and Issues:	

Class Diagram

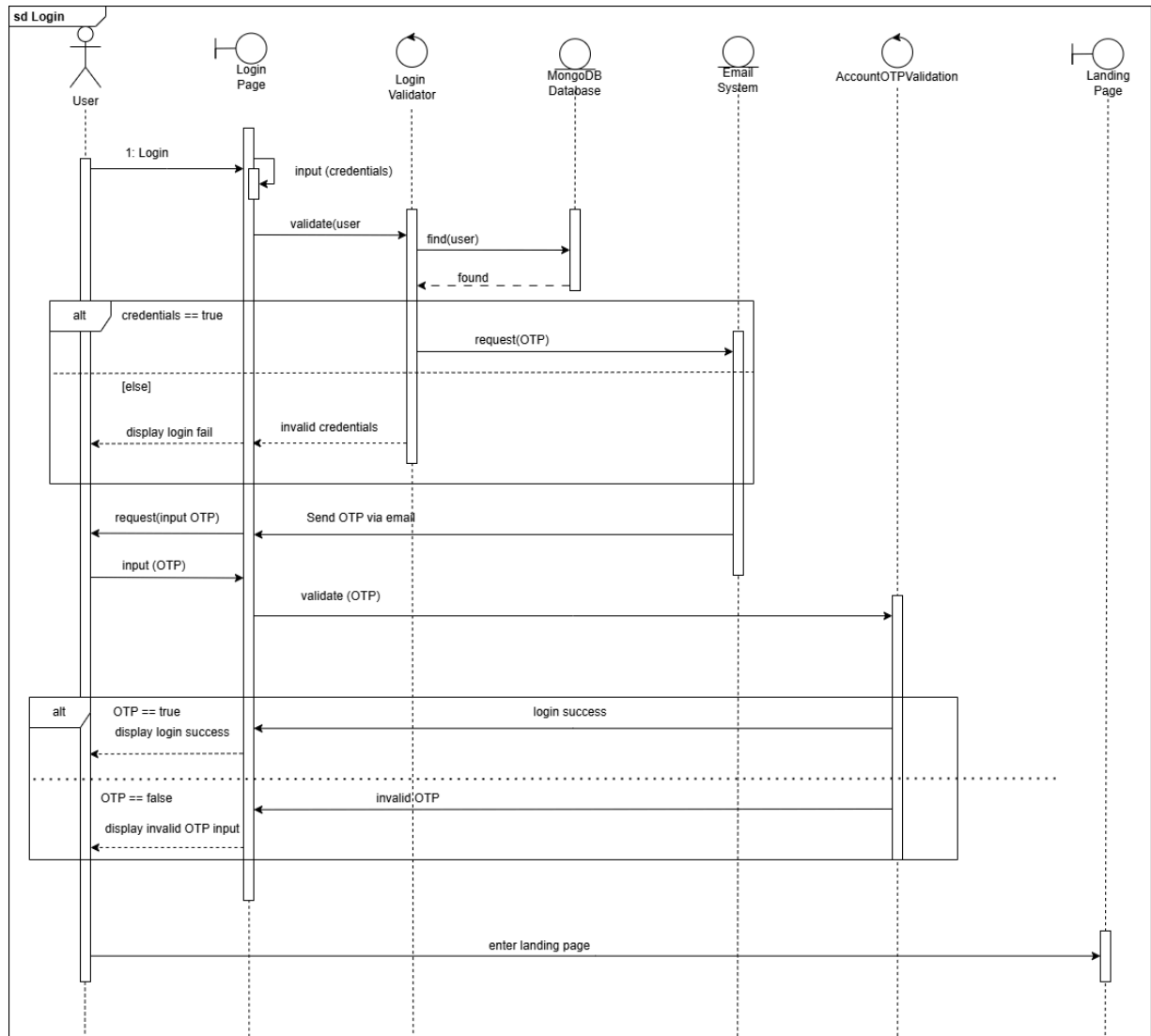


Key Boundary Classes and Control Classes

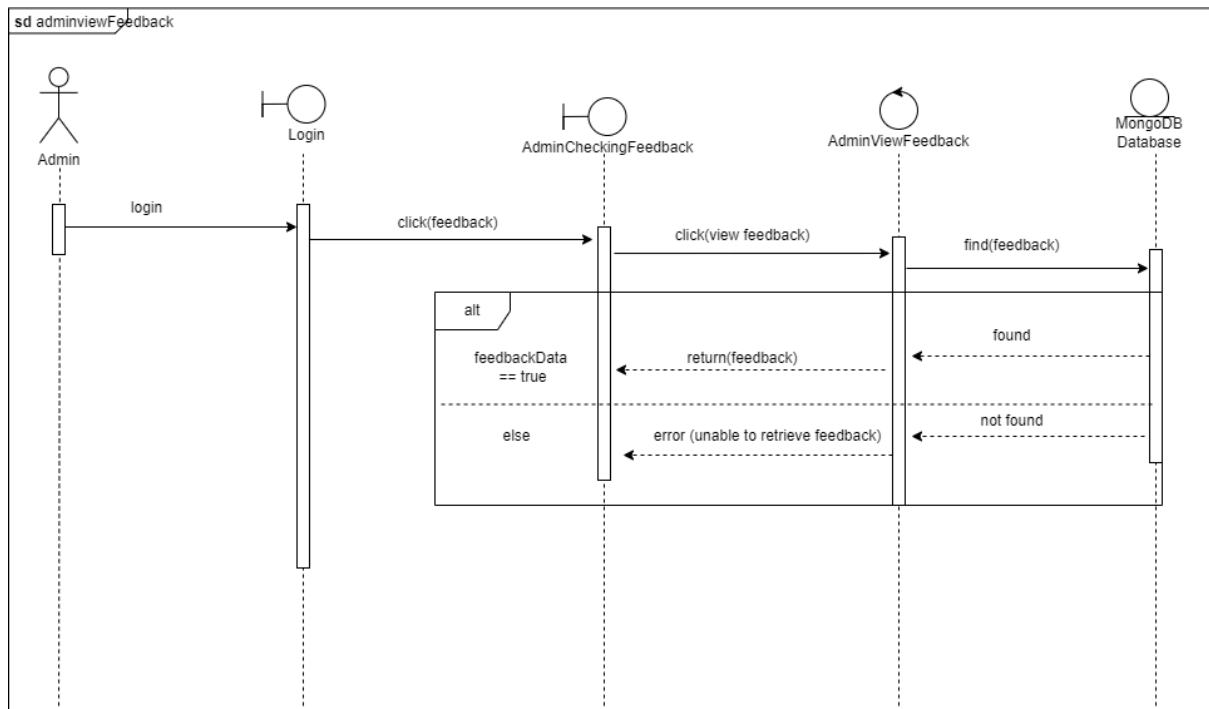


Sequence Diagram

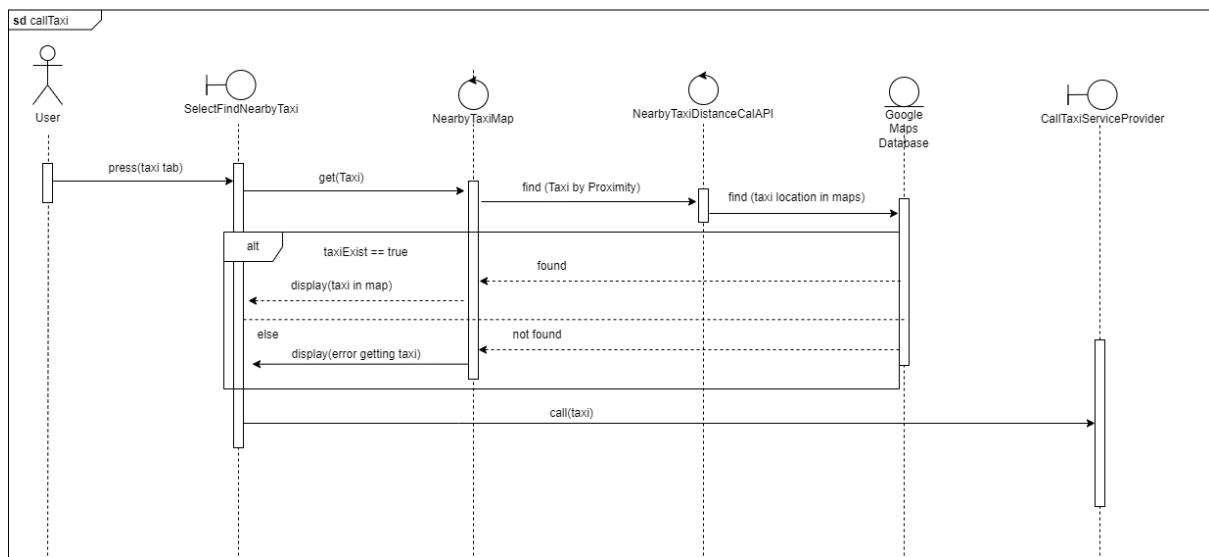
Login



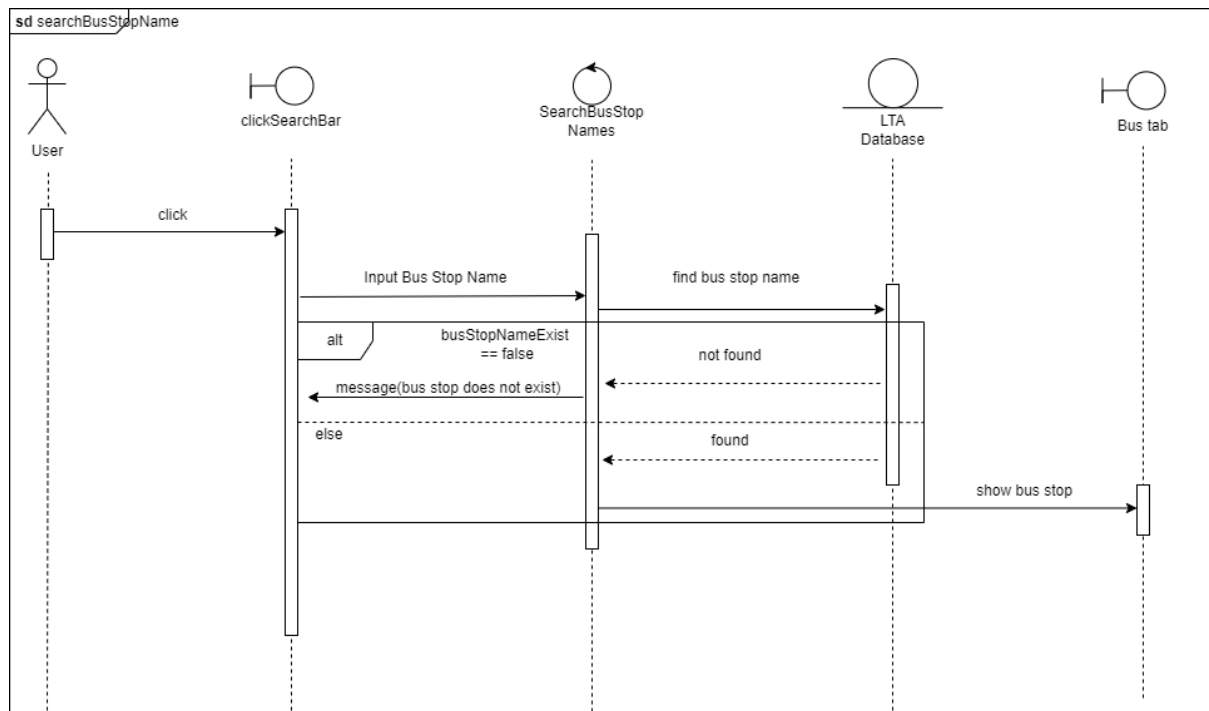
Admin View Feedback



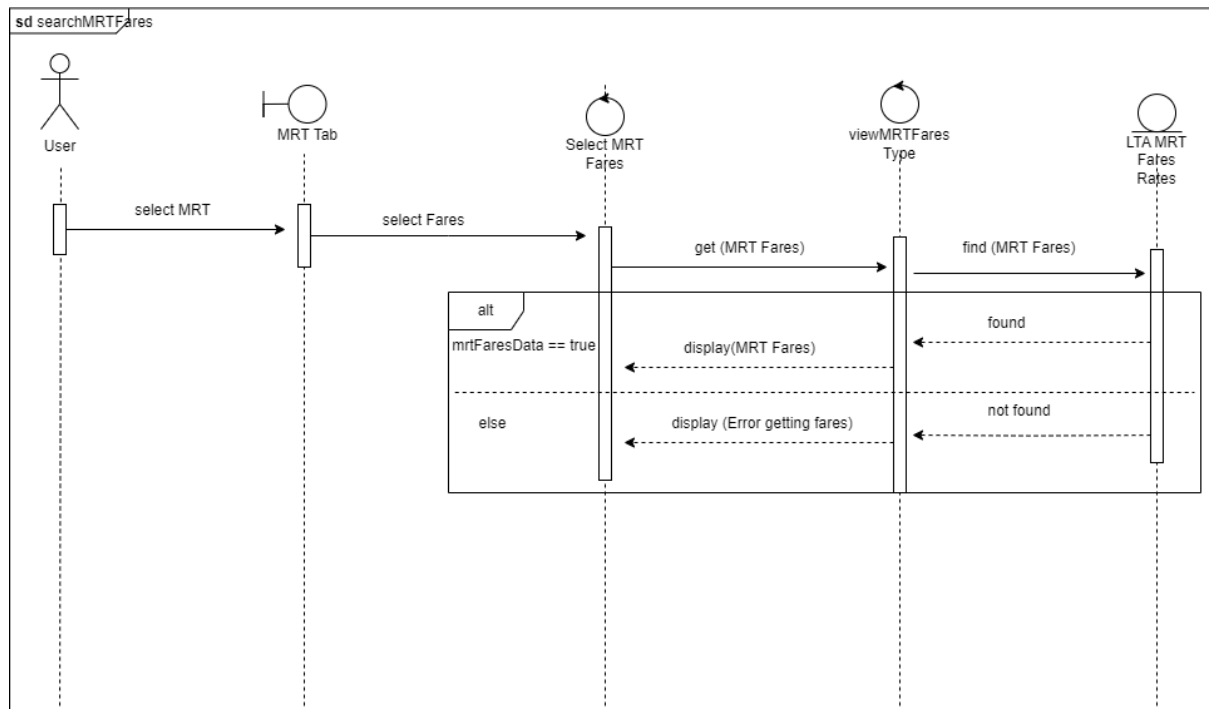
User Call Taxi



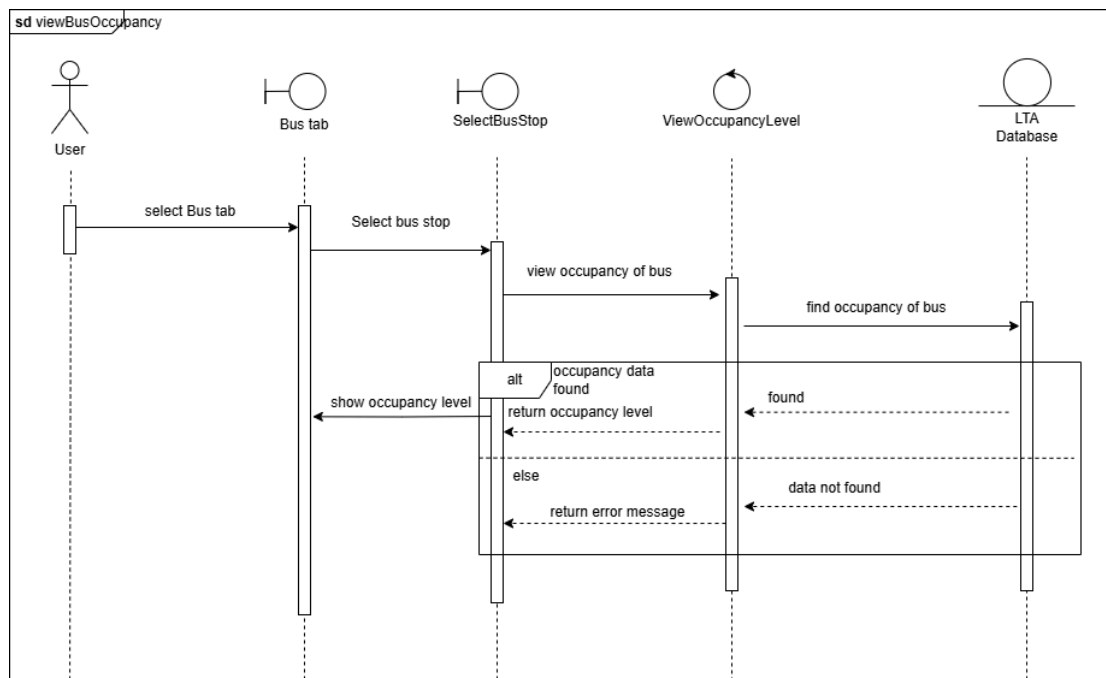
Search Bus Stop by Name



Search MRT Fares



View Bus Occupancy



*Please refer to the external file for a clearer illustration of the Initial Dialog Map

