

**NANYANG
TECHNOLOGICAL
UNIVERSITY**
SINGAPORE

SC2006 – Software Engineering

Lab 2 Deliverables

Lab Group	SCSI-30
Team	TEAM 5
Members	Tan Wei Song Russell Bryan (U2421844K)
	Justin Woon Thean Woon (U244641B)
	Ethan Jared Chong Rui Zhi (U2421895B)
	Evelyn Theresia Cuaca (U2320523L)
	Parvez Kurniawan Wijaya (U2423845G)

Table of Contents

1. Use Case diagram and Use Case descriptions

A. Use Case Diagram

B. Use Case Descriptions

I. For Functional Requirement #1

I.I SignUp

I.II Login

I.III AuthenticateUser

II. For Functional Requirement #2

II.I ManageSavedPlaces

II.II ViewMap

II.III SearchDestination

II.IV SearchSuggestion

II.V CompareTransportation

II.VI Report

II.VII SuggestRoute

III. For Functional Requirement #3

III. I View Driving Metrics

III. II View Public Transport Metrics

III. III View Walking Metrics

III. IV View Cycling Metrics

III. V Bike Rental Information

IV. For Functional Requirement #4

IV.I Fetch Traffic Data

IV.II Manage Account User

IV.III ManageObstructionReport

IV.IV ManageSystemReport

IV.V Manage Suggested Route

V. For Functional Requirement #5

V.I External System Integration

2. Class Diagram of Entity Classes

3. Key Boundary and Control Classes

4. Sequence Diagrams of Use Cases

I. For Use Cases Under I

I.I SignUp

[I.II](#) Login

II. For Use Cases Under II

II.I SearchRouteToDestination

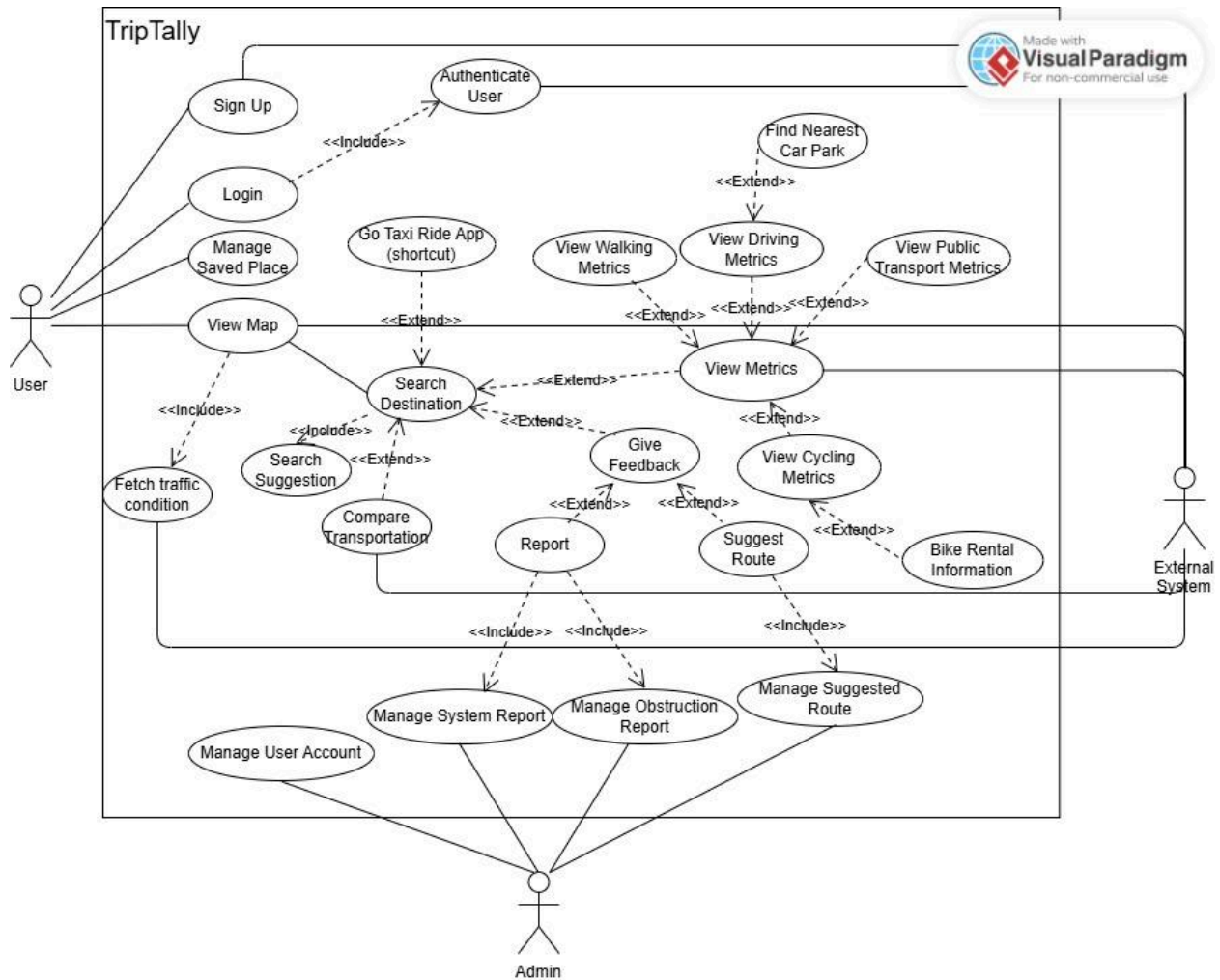
[II.II](#) ManageSavedPlaces

- II.III DeleteSavedLocations
 - [II.IV](#) ViewMap
 - II.V SearchSuggestion
- III. For Use Cases Under III
 - III.I ViewDrivingMetrics
 - [III.II](#) ViewPublicTransportMetrics
 - III.III ViewWalkingMetrics
 - [III.IV](#) ViewCyclingMetrics
- IV. For Use Cases Under IV
 - IV.I ViewReports
 - [IV.II](#) ManageReport
 - IV.III ViewSuggestedRoute
 - [IV.IV](#) ManageUserSuggestedRoute

5. Initial Dialog Diagram

1. Use Case diagram and Use Case descriptions

A. Use Case Diagram



B. Use Case Descriptions

I. For Functional Requirement #1

I.I SignUp

Use Case ID:	#1-1		
Use Case Name:	SignUp		
Created By:	Russell Tan	Last Updated By:	Russell Tan
Date Created:	01/09/2025	Date Last Updated:	01/09/2025
Actor:	User		
Description:	Creation of a new TripTally account of a user		
Preconditions:	<ol style="list-style-type: none">1. App must be installed and the user's email/contact number has not been registered.2. Internet connection must be available.		
Postconditions:	<ol style="list-style-type: none">1. New account created and verified.2. Confirmation message sent to the users email.		
Priority:	Low		
Frequency of Use:	High		
Flow of Events:	<ol style="list-style-type: none">1. User selects Sign Up2. User enters name, email, password and contact number. User can also sign up through Apple ID or Google.3. System validates that the email exists and is valid and checks the password strength.4. User agrees to the Terms and Services of the application.5. Account is then created and confirmation is sent to the email that has created the account.		
Alternative Flows:	AF-1: Sign up available with Apple/Google.		
Exceptions:	<p>EX-1: If there is a duplicate email account, system will prompt that account already exists and prompts user login</p> <p>EX-2: A weak password would result in the</p>		

	system prompting to change the password to have at least 8 characters with a mix of alphabets, numbers and special characters. EX-3: No internet would result in a retry option.
Includes:	1. Login 2. AuthenticateUser
Special Requirements:	Passwords must be a minimum of length 8, containing both alphabets, numbers and at least one special character.
Assumptions:	None
Notes and Issues:	None

I.II Login

Use Case ID:	#1-2		
Use Case Name:	Login		
Created By:	Russell Tan	Last Updated By:	Russell Tan
Date Created:	01/09/2025	Date Last Updated:	01/09/2025
Actor:	User		
Description:	Start a user session with email and the corresponding password.		
Preconditions:	1. User has a registered account and clicks the Login button. 2. Internet connection must be available		
Postconditions:	1. User has been authenticated and secure session active		
Priority:	Low		
Frequency of Use:	Low		
Flow of Events:	1. User clicks Login button. 2. User enters email/contact number with the correct password. 3. AuthenticateUser is called. 4. User is redirected to the Home page of the application if AuthenticateUser is successful.		

Alternative Flows:	<p>AF-1: Wrong password/ username</p> <ul style="list-style-type: none"> • If User enters their password incorrectly for 5 tries, their account would be locked for an hour. • Whenever the User gets their credentials wrong before exceeding 5 tries, a message saying “invalid login/password” will be displayed instead. • The system will prompt user to re-enter their username and password <p>AF-2: Network issues would bring back the user to the sign in page.</p>
Exceptions:	None
Includes:	AuthenticateUser
Special Requirements:	None
Assumptions:	Assume user enters the right password on the first try
Notes and Issues:	None

I.III AuthenticateUser

Use Case ID:	#1-3		
Use Case Name:	AuthenticateUser		
Created By:	Russell Tan	Last Updated By:	Russell Tan
Date Created:	01/09/2025	Date Last Updated:	01/09/2025
Actor:	ExternalSystem		
Description:	Process of verifying that the user's account exists and matches the password before logging them in.		
Preconditions:	<ol style="list-style-type: none"> 1. Login has been initiated. 2. Internet connection must be available. 		
Postconditions:	<ol style="list-style-type: none"> 1. User securely authenticated and login successful. 		
Priority:	High		
Frequency of Use:	High		

Flow of Events:	<ol style="list-style-type: none"> 1. System receives the login credentials from the User. 2. System fetches data matching the account. 3. System checks the status of the account and ensures that it is not locked. 4. System then verifies the user's password. 5. System resets the retry fail count. 6. System returns a true and user is authenticated
Alternative Flows:	AF-1: If the account is locked, system displays "Your account has been locked" to the User.
Exceptions:	EX-1: If user cancels at the account selection, display a retry option
Includes:	<ol style="list-style-type: none"> 1. SignUp 2. Login
Special Requirements:	None
Assumptions:	Account that the user is logging in exists.
Notes and Issues:	None

II. For Functional Requirement #2

II.I ManageSavedPlaces

Use Case ID:	#2-1		
Use Case Name:	ManageSavedPlaces		
Created By:	Russell Tan	Last Updated By:	Russell Tan
Date Created:	01/09/2025	Date Last Updated:	01/09/2025
Actor:	User		
Description:	User have the ability to manage their saved locations. (Home, Work, Favourites)		
Preconditions:	<ol style="list-style-type: none"> 1. User must be logged in. 2. Internet connection must be available 		

Postconditions:	1. User has successfully added/edited/deleted the saved location.
Priority:	Low
Frequency of Use:	Low
Flow of Events:	<ol style="list-style-type: none"> 1. User selects the “Saved Places” button. 2. User is able to add, edit or delete the location selected by the user. 3. System updates database and syncs profile. 4. Newly added, edited or deleted location is displayed for the user.
Alternative Flows:	AF-1: Duplicate places shall prompt a rename of the location to the user.
Exceptions:	EX-1: If invalid location is selected, an error will be displayed and bring the user back to the home page.
Includes:	None
Special Requirements:	None
Assumptions:	User logged in.
Notes and Issues:	None

II.II ViewMap

Use Case ID:	#2-2		
Use Case Name:	ViewMap		
Created By:	Russell Tan	Last Updated By:	Russell Tan
Date Created:	01/09/2025	Date Last Updated:	01/09/2025
Actor:	User, ExternalSystem		
Description:	Displays interactive map that is centered on the user’s location.		
Preconditions:	<ol style="list-style-type: none"> 1. User is logged in. 2. Internet connection must be available. 		

Postconditions:	1. Map is displayed with the current location displayed to the user.
Priority:	High
Frequency of Use:	High
Flow of Events:	<ol style="list-style-type: none"> 1. User opens the home page/ clicks on the explore button. 2. System fetches current location and GoogleMaps API. 3. System displays an interactive map with context.
Alternative Flows:	AF-1: User can manually input their address when GPS is off.
Exceptions:	None
Includes:	None
Special Requirements:	None
Assumptions:	User mobile devices have granted permission to view the user's current location.
Notes and Issues:	None

II.III SearchRouteToDestination

Use Case ID:	#2-3		
Use Case Name:	SearchDestination		
Created By:	Russell Tan	Last Updated By:	Russell Tan
Date Created:	01/09/2025	Date Last Updated:	01/09/2025
Actor:	User, ExternalSystem		
Description:	Displays the selected location to the user.		
Preconditions:	<ol style="list-style-type: none"> 1. Origin and Destination must be known. 2. Internet connection must be available. 		

Postconditions:	1. The selected location is displayed on the map for the user with the information regarding the location.
Priority:	High
Frequency of Use:	High
Flow of Events:	<ol style="list-style-type: none"> 1. System displays cached recent searches. 2. User keys in their destination in the search bar. 3. System queries geolocation API to search for the destination. 4. Location is displayed with its information and directions for the user. 5. Users can then select mode of transportation for the directions to their destination. 6. Route calculation is triggered. 7. Multiple routes to the users destination is displayed
Alternative Flows:	AF-1: Destination picked is from recent or Saved Places
Exceptions:	EX-1: Geocoding API is down. EX-2: Invalid location is inputted.
Includes:	1. ViewMap
Special Requirements:	None
Assumptions:	Interactive map is loaded.
Notes and Issues:	None

II.IV SearchSuggestion

Use Case ID:	#2-4		
Use Case Name:	SearchSuggestion		
Created By:	Russell Tan	Last Updated By:	Russell Tan
Date Created:	01/09/2025	Date Last Updated:	01/09/2025
Actor:	User, ExternalSystems		
Description:	Display live search suggestions as the user types into the search bar.		
Preconditions:	<ol style="list-style-type: none"> 1. User has keyed in a location in the search bar. 2. Internet connection must be available. 		
Postconditions:	<ol style="list-style-type: none"> 1. Search options displayed for users to select. 		
Priority:	High		
Frequency of Use:	High		
Flow of Events:	<ol style="list-style-type: none"> 1. User keys in a destination in the search bar. 2. System queries an external search API. 3. Suggestions based on the user's search are displayed. 4. User is able to select one of the suggestions that is displayed. 		
Alternative Flows:	None		
Exceptions:	EX-1: Location/API unavailable would display a "No match/ Try again" text to the user.		
Includes:	<ol style="list-style-type: none"> 1. ViewMap 2. SearchDestination 		
Special Requirements:	None		
Assumptions:	User has keyed in a valid location in the search bar.		
Notes and Issues:	None		

II.V CompareTransportation

Use Case ID:	#2-5		
Use Case Name:	CompareTransportation		
Created By:	Russell Tan	Last Updated By:	Russell Tan
Date Created:	01/09/2025	Date Last Updated:	01/09/2025
Actor:	User, ExternalSystem		
Description:	Side by side comparison of two modes of transportation. (Time, Cost, Carbon Footprint)		
Preconditions:	<ol style="list-style-type: none"> 1. Origin and destination must be known. 2. Internet connection must be available 		
Postconditions:	<ol style="list-style-type: none"> 1. User able to see side by side comparison of the 2 modes of transportation selected. 		
Priority:	High		
Frequency of Use:	High		
Flow of Events:	<ol style="list-style-type: none"> 1. User select the 2 modes of transportation that they would like to compare. 2. System fetches data of the different modes of transportation from their various APIs. 3. System compares the selected modes of transportation (Driving, Public Transport, Walking, Cycling) 4. System calculates the ETA, distance, cost and carbon footprint. 5. This data is then displayed to the user for comparison. 		
Alternative Flows:	<ol style="list-style-type: none"> 1. AF-1: User is able to view a brief comparison of all 4 metrics if the compare button is clicked without any selection of the 2 modes. 		
Exceptions:	<p>EX-1: If one mode is down due to the API being down, display historical data and inform users that data is based on past information.</p>		
Includes:	<ol style="list-style-type: none"> 1. ViewMap 2. SearchDestination 		
Special Requirements:	None		
Assumptions:	Origin and destination known.		
Notes and Issues:	None		

II.VI Report

Use Case ID:	#2-6		
Use Case Name:	Report		
Created By:	Russell Tan	Last Updated By:	Russell Tan
Date Created:	01/09/2025	Date Last Updated:	01/09/2025
Actor:	User		
Description:	User can submit report about incident on road or system error faced while using the app		
Preconditions:	<ol style="list-style-type: none"> 1. The user is logged in. 2. Network is available. 		
Postconditions:	<ol style="list-style-type: none"> 1. User is saved in the system with "Submitted" status 2. Admin are automatically notified and the report is placed in the review queue 		
Priority:	High		
Frequency of Use:	Medium		
Flow of Events:	<ol style="list-style-type: none"> 1. User open submit report option 2. Systems prompt user to choose type of report (road incident/system error) 3. User select reporting road incident or system error 4. The system displays the form where users can describe the issue and upload relevant supporting material such as picture/video. 5. System validate, store the report and confirm submission. 6. System notify admin queue for review. 		
Alternative Flows:	<p>AF-1:Road incident report</p> <ol style="list-style-type: none"> 1. Users input the location of the incident. 2. System prompt user to select incident categories. 3. Users can optionally add description and photos/videos. 4. Once submitted, report is saved with location, time, and marked as submitted 5. Admins are notified and incident is placed into the review queue <p>AF-2:System error report</p> <ol style="list-style-type: none"> 1. The User selects the category of error (fare calculation error, route error, data error etc.) 		

	2. The User is prompted to provide a description of the error. 3. User can optionally provide photo/video
Exceptions:	EX-1: Missing required information- System will highlight the required fields and prevent submission. EX-2: File upload error- System provide option to submit report without
Includes:	1. ManageObstructionReport 2. ManageSystemReport
Special Requirements:	Status of Report includes: Submitted, Pending, Resolved
Assumptions:	None
Notes and Issues:	Verified incidents will affect the routing.

II.VII SuggestRoute

Use Case ID:	#2-7		
Use Case Name:	SuggestRoute		
Created By:	Russell Tan	Last Updated By:	Russell Tan
Date Created:	01/09/2025	Date Last Updated:	01/09/2025
Actor:	User		
Description:	User submits different walking /cycling routes from one location to another than the ones generated by the system.		
Preconditions:	1. User is logged in to a valid account 2. Internet connection must be available		
Postconditions:	1. User gets confirmation that their suggested route has been successfully uploaded.		
Priority:	Low		
Frequency of Use:	Low		

Flow of Events:	<ol style="list-style-type: none"> 1. The User selects the option to upload a new route. 2. The User selects from a range of tags that accurately describe the type of route they are uploading. 3. The User is then prompted to search for their start and end points of their destination using the search bar. 4. Once both points are confirmed, the User will tap on the screen to place “checkpoint” pins on the map to show exactly where the route will lead. 5. At each checkpoint the User will have the option to indicate whether the route will continue indoors or outdoors, and also upload helping images. 6. Once all the checkpoints and details are confirmed, the User is prompted to give the route a name and short written description. 7. The User will then tap on the “upload” button to upload the route to the forum.
Alternative Flows:	None
Exceptions:	EX-1: If the forum’s servers are down, return a message stating “The servers are currently down, please come back later”.
Includes:	<ol style="list-style-type: none"> 1. ViewMap 2. SearchDestination
Special Requirements:	None
Assumptions:	The User is not banned from posting routes.
Notes and Issues:	None

III. For Functional Requirement #3

III. I View Driving Metrics

Use Case ID:	#3-1		
Use Case Name:	ViewDrivingMetrics		
Created By:	Ethan Jared Chong Rui Zhi	Last Updated By:	Ethan Jared Chong Rui Zhi
Date Created:	1/8/2025	Date Last Updated:	1/8/2025

Actor:	User
Description:	User is shown metrics for driving to their destination.
Preconditions:	1. User has searched for a destination and selected the “driving” mode of transportation.
Postconditions:	1. The User gets shown the corresponding metrics for driving.
Priority:	High
Frequency of Use:	Medium
Flow of Events:	<ol style="list-style-type: none"> 1. User searches for destination 2. User confirms destination 3. User selects “driving” as mode of transportation 4. Retrieve ERP pricing data, destination parking lot data (call FindNearestCarpark), fuel consumption data, fuel price data, vehicle CO2 emissions data, travel duration. 5. Calculate total cost. 6. Display drop-down table that shows CO2 emission, travel time and total cost.
Alternative Flows:	None
Exceptions:	<p>EX-1: If driving cannot reach the destination, the option to select “driving” will be disabled.</p> <p>EX-2: If some required information cannot be fetched, “note: error retrieving live data” message will be displayed in the corresponding table row and display historical data.</p>
Includes:	FindNearestCarpark
Special Requirements:	User’s phone location services are enabled for TripTally.
Assumptions:	None
Notes and Issues:	None

III. II View Public Transport Metrics

Use Case ID:	#3-2		
Use Case Name:	ViewPublicTransportMetrics		
Created By:	Ethan Jared Chong Rui Zhi	Last Updated By:	Ethan Jared Chong Rui Zhi
Date Created:	1/8/2025	Date Last Updated:	1/8/2025
Actor:	User		
Description:	User is shown metrics for taking public transport to their destination.		
Preconditions:	1. User has searched for a destination and selected the "public transport" mode of transportation.		
Postconditions:	1. The User gets shown the corresponding metrics for taking public transportation.		
Priority:	High		
Frequency of Use:	High		
Flow of Events:	<ol style="list-style-type: none"> 1. User searches for destination 2. User confirms destination 3. User selects "public transport" as mode of transportation 4. Retrieve total trip fares, CO2 emissions data, travel duration. 5. Calculate total cost. 6. Display drop-down table that shows CO2 emission, travel time and total cost. 		
Alternative Flows:	None		
Exceptions:	<p>EX-1: If any required information is missing or invalid, an error message is displayed.</p> <p>EX-2: If public transportation cannot reach the destination, the option to select "public transportation" will be disabled.</p> <p>EX-3: If public transportation is out of operating hours, display "Public Transport currently out of operating hours."</p>		
Includes:	None		
Special Requirements:	User's phone location services are enabled for TripTally.		
Assumptions:	None		
Notes and Issues:	None		

III. III View Walking Metrics

Use Case ID:	#3-3		
Use Case Name:	View Walking Metrics		
Created By:	Ethan Jared Chong Rui Zhi	Last Updated By:	Ethan Jared Chong Rui Zhi
Date Created:	1/8/2025	Date Last Updated:	1/8/2025
Actor:	User		
Description:	User is shown metrics for walking to their destination.		
Preconditions:	1. User has searched for a destination and selected the “walking” mode of transportation.		
Postconditions:	1. The User gets shown the corresponding metrics for walking to their destination.		
Priority:			
Frequency of Use:			
Flow of Events:	1. User searches for destination 2. User confirms destination 3. User selects “walking” as mode of transportation 4. Retrieve calories burnt and time taken 5. Display drop-down table that shows CO2 emission (zero), travel time and total cost (zero).		

Alternative Flows:	None
Exceptions:	None
Includes:	None
Special Requirements:	User’s phone location services are enabled for TripTally.

Assumptions:	None
Notes and Issues:	None

III. IV View Cycling Metrics

Use Case ID:	#3-4		
Use Case Name:	ViewCyclingMetrics		
Created By:	Ethan Jared Chong Rui Zhi	Last Updated By:	Ethan Jared Chong Rui Zhi
Date Created:	1/8/2025	Date Last Updated:	1/8/2025
Actor:	User		
Description:	User is shown metrics for walking to their destination.		
Preconditions:	1. User has searched for a destination and selected the “cycling” mode of transportation.		
Postconditions:	1. The User gets shown the corresponding metrics for cycling to their destination.		
Priority:	High		
Frequency of Use:	Medium		
Flow of Events:	<ol style="list-style-type: none"> 1. User searches for destination 2. User confirms destination 3. User selects “cycling” as mode of transportation 4. Retrieve calories burnt and time taken 5. Display drop-down table that shows CO2 emission (zero), travel time and total cost (zero). 		
Alternative Flows:	None		
Exceptions:	None		
Includes:	None		
Special Requirements:	User’s phone location services are enabled for TripTally.		
Assumptions:	None		
Notes and Issues:	None		

III. V Bike Rental Information

Use Case ID:	#3-5		
Use Case Name:	BikeRentalInformation		
Created By:	Ethan Jared Chong Rui Zhi	Last Updated By:	Ethan Jared Chong Rui Zhi
Date Created:	1/8/2025	Date Last Updated:	1/8/2025
Actor:	User		
Description:	Shows all available bike-sharing spots near the User.		
Preconditions:	<ol style="list-style-type: none"> 1. The User has selected “cycling” as their chosen mode of transportation. 2. The User has opted to show the available bike-sharing points. 		
Postconditions:	<ol style="list-style-type: none"> 1. Successfully display availability of bikes at the various spots. 2. Successfully display the locations of possible return points. 		
Priority:	Medium		
Frequency of Use:	High		
Flow of Events:	<ol style="list-style-type: none"> 1. Query bike-sharing API(s) 2. Show nearby bike-sharing points 3. Show the estimated number of available bicycles 4. Retrieve hourly/half-hourly rental cost and calculate cost to get to destination based on time taken to walk to bike-sharing point and then reach the destination. 5. Provide redirect to bike sharing application 		
Alternative Flows:	None		
Exceptions:	None		
Includes:	None		
Special Requirements:	User’s phone location services are enabled for TripTally.		
Assumptions:	None		

Notes and Issues:	None
-------------------	------

IV. For Functional Requirement #4

IV.I Fetch Traffic Data

Use Case ID:	#4-1		
Use Case Name:	Fetch Traffic Data		
Created By:	Evelyn Theresia Cuaca	Last Updated By:	Evelyn Theresia Cuaca
Date Created:	01/09/2025	Date Last Updated:	01/09/2025
Actor:	External System		
Description:	The system is able to receive and process real-time traffic incidents such as accidents, closures, or roadworks from an external feed.		
Preconditions:	<ol style="list-style-type: none"> 1. Connection to traffic alert API is active. 2. The user is viewing a map, searching, or navigating. 		
Postconditions:	<ol style="list-style-type: none"> 1. Incident data ingested and displayed in the app. 2. Routes and ETAs updated accordingly. 3. Logs and metrics recorded for monitoring. 		
Priority:	High		
Frequency of Use:	High		
Flow of Events:	<ol style="list-style-type: none"> 1. The Traffic Alert System (External system) alerts the system about traffic conditions (accident, road closure). 2. The system normalizes and shows the data. 3. The incident is stored in the TripTally database including the timestamp and validity period 4. Routing engine checks whether the incident affects: <ul style="list-style-type: none"> ○ Active trips in progress. ○ New trip requests. 5. If an active route is affected, drivers receive a soft reroute suggestion. 6. Map view automatically overlays icons for incidents (warning symbols, red markers, etc.). 		

Alternative Flows:	<p>AF-1: API Failure: If the traffic feed is unavailable, the system falls back to the last known incidents .User is warned that traffic data may be outdated.</p> <p>AF-2: High Latency: If data fetching takes more than 60s, the routing service uses default routes and applies the update once available.</p>
Exceptions:	None
Includes:	None
Special Requirements:	Access to traffic API
Assumptions:	The System has access to traffic alert API
Notes and Issues:	None

IV.II Manage Account User

Use Case ID:	#4-2		
Use Case Name:	Manage Account User		
Created By:	Evelyn Theresia Cuaca	Last Updated By:	Evelyn Theresia Cuaca
Date Created:	01/09/2025	Date Last Updated:	01/09/2025
Actor:	Admin		
Description:	Admin can view, deactivate, delete account		

Preconditions:	<ol style="list-style-type: none"> 1. Admin must be authenticated with proper role/ privileges 2. The target user account exist
Postconditions:	<ol style="list-style-type: none"> 1. User account data is successfully updated in the database. 2. All changes are recorded with timestamp, admin ID, and reason for change 3. If the account is deactivated/ deleted, the user losses access immediately
Priority:	High
Frequency of Use:	Medium
Flow of Events:	<ol style="list-style-type: none"> 1. Admin logs into the system 2. Admin navigates to the User Account Management panel 3. Admin searches for the target user account by ID, email, or name 4. System retrieves and displays user profile and account status. 5. Admin selects an action: <ul style="list-style-type: none"> • Edit user details • Deactivate account • Delete account 6. Admin confirms action 7. System validates the request and applies changes
Alternative Flows:	AF-1: Reset Password: Instead of editing details, Admin triggers a password reset email/SMS.
Exceptions:	None
Includes:	Login
Special Requirements:	<ol style="list-style-type: none"> 1. Deactivated users must immediately lose access to the system.
Assumptions:	Only authorized admins can access this feature.
Notes and Issues:	<ol style="list-style-type: none"> 1. Account deletion is irreversible. 2. Sensitive data (like passwords) is never directly editable, only resettable.

IV.III ManageObstructionReport

Use Case ID:	#4-3		
Use Case Name:	ManageObstructionReport		
Created By:	Parvez Kurniawan Wijaya	Last Updated By:	Parvez Kurniawan Wijaya
Date Created:	01/09/2025	Date Last Updated:	01/09/2025
Actor:	Admin		
Description:	Admin verifies road incidents that are reported by Users.		
Preconditions:	<ol style="list-style-type: none"> 1. Admin received information of road incident reported by users 2. Incident report by the User is stored in the system as Unverified 		
Postconditions:	<ol style="list-style-type: none"> 1. Incident record is updated with a status: Verified, Rejected, or Pending Review 2. Verified incidents are shown on the user's map and factored into routing 3. Rejected incidents are excluded from the routing system 		
Priority:	High		
Frequency of Use:	Medium		
Flow of Events	<ol style="list-style-type: none"> 1. The user submits an incident report through the app . 2. The system stores the incident as Unverified by default. 3. Admin reviews and updates the status: Verified / Rejected / Pending. 4. The system updates the incident record. 5. If Verified, the incident is factored into ETA and rerouting logic. 6. The user is notified of the outcome (e.g., "Your report has been verified"). 		
Alternative Flow	None		
Exceptions	None		
Include	<ol style="list-style-type: none"> 1. Login 2. ViewMap 		
Special Requirement	Only verified incidents affect routing and reroute notifications		
Assumptions	Not all API alerts are accurate, some require Admin judgement		

Notes and Issues	None
------------------	------

IV.IV ManageSystemReport

Use Case ID:	#4-4		
Use Case Name:	ManageSystemReport		
Created By:	Evelyn Theresia Cuaca	Last Updated By:	Evelyn Theresia Cuaca
Date Created:	01/09/2025	Date Last Updated:	01/09/2025
Actor:	Admin		
Description:	Admin can view, categorize, assign, resolve, or archive reports to ensure that issues are addressed promptly		
Preconditions:	<ol style="list-style-type: none"> 1. Admin must be authenticated with the correct privileges. 2. At least one report must exist in the system. 		
Postconditions:	<ol style="list-style-type: none"> 1. Reports are updated with new statuses 2. All admin actions are recorded with timestamp and admin ID. 3. Resolved reports no longer appear in the report queue. 		
Priority:	Medium		
Frequency of Use:	Medium		
Flow of Events:	<ol style="list-style-type: none"> 1. Admin logs into the system 2. Admin navigates to the Reports Management Module 3. System displays a list of reports 4. Admin selects a report from the list 5. Admin chooses an action: <ul style="list-style-type: none"> • Resolve the report • Assign report to another admin/ technical team • Archive report if it is invalid or duplicate 6. Admin's action is recorded by the system 7. Confirmation message shown to Admin 		
Alternative Flows:	AF-1: Escalation Flow: If the issue is critical, Admin hand it over to higher-level support or incident response team.		
Exceptions:	EX-1: Report Not Found: Report may have been deleted or already resolved by another admin.		
Includes:	None		

Special Requirements:	<ol style="list-style-type: none"> 1. Reports must always have one of the following status values: Submitted, Pending, Resolved. 2. Critical reports should trigger notifications to Admin
Assumptions:	<ol style="list-style-type: none"> 1. Reports follow a linear lifecycle (Submitted → Pending → Resolved).
Notes and Issues:	<ol style="list-style-type: none"> 1. “Submitted” is the default state for new reports. 2. Once marked “Resolved,” reports can still be reviewed in the history/archive. 3. Users will receive notifications when their report changes status.

IV.V Manage Suggested Route

Use Case ID:	#4-5		
Use Case Name:	ManageSuggestedRoute		
Created By:	Evelyn Theresia Cuaca	Last Updated By:	Evelyn Theresia Cuaca
Date Created:	01/09/2025	Date Last Updated:	01/09/2025
Actor:	Admin		
Description:	Admin can review and validates user-suggested routes against traffic data and safety constraints, and either approves, modifies, or rejects them		
Preconditions:	<ol style="list-style-type: none"> 1. The user has submitted a suggested route to the forum. 2. Admin is authenticated and authorized. 3. Suggested route data (start, end, waypoints) is stored in the system. 		
Postconditions:	<ol style="list-style-type: none"> 1. Approved routes are integrated into the routing database or flagged for testing. 2. Verification of suggested route in the forum 		
Priority:	Low		
Frequency of Use:	Medium		

Flow of Events:	<ol style="list-style-type: none"> 1. Admin logs into the system 2. Navigates to the Suggested Routes Forum 3. The system displays a list of user-submitted routes with detailed information(origin–destination, date submitted, status). 4. Admin selects a suggested route for review 5. The system shows details 6. Admin evaluates the suggestion against: <ul style="list-style-type: none"> • Safety (road closures, traffic rules). • Efficiency (time, cost). • Relevance (popular demand) 7. Admin takes one of the following actions <ul style="list-style-type: none"> • Approve : verify the suggested route • Reject : mark invalid with reason. 8. The Status of suggested route is updated (Verified, Rejected)
Alternative Flows:	AF-1: Auto-Approval: Low-risk routes may be auto-verified.
Exceptions:	EX-1: Invalid Data - Suggested route is missing waypoints or corrupted EX-2: Duplicate Suggestion- Route already exists in Forum
Includes:	None
Special Requirements:	<ol style="list-style-type: none"> 1. Suggested routes must have a status (Submitted, Pending Review, Verified, Rejected). 2. Status of verification must be shown in the Forum. 3. Users must download and load alternate routes from the Forum before use. 4. If no alternate route is loaded, the app defaults to the normal system-generated route. 5. Approved routes must not violate traffic/safety rules.
Assumptions:	None
Notes and Issues:	Forum acts as a shared library for alternate user-suggested routes

V. For Functional Requirement #5

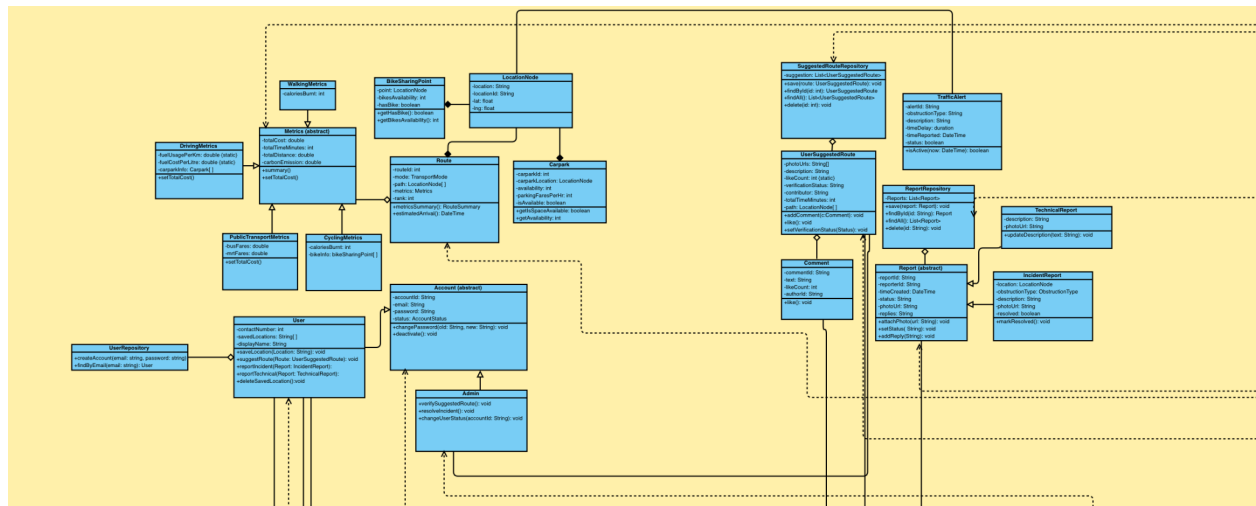
V.I External System Integration

Use Case ID:	#5-1		
Use Case Name:	ExternalSystemIntegration		
Created By:	Evelyn Theresia Cuaca	Last Updated By:	Evelyn Theresia Cuaca
Date Created:	01/09/2025	Date Last Updated:	01/09/2025
Actor:	External System / Data Provider		
Description:	Integrate with external systems (data provider) to get real-time data like routes, traffic incidents, ERP rates, carpark availability, and fuel prices,etc.		
Preconditions:	<ol style="list-style-type: none">1. Valid API keys/credentials for each external provider2. Network access to provider endpoints.3. Data structure and mappings are defined		
Postconditions:	<ol style="list-style-type: none">1. Data retrieved from external systems is made available to TripTally service		
Priority:	High		
Frequency of Use:	High		
Flow of Events:	<ol style="list-style-type: none">1. Mapping<ul style="list-style-type: none">• System calls Google Map API to identify user position• The API fetch available driving, walking, cycling, and public transportation routes• Map APIs will render preview of the routes2. Traffic and Incidents<ul style="list-style-type: none">• TripTally get real-time traffic conditions and road closures from external traffic APIs• Incident data is applied to update travelling time and rerouting logic.3. ERP integration<ul style="list-style-type: none">• The system retrieves ERP gantry locations and rate tables by time/day from LTA ERP providers.• ERP costs calculated per driving route.		

	<p>4. Parking Integration</p> <ul style="list-style-type: none"> • Carpark APIs provide real-time availability and rates • The system will update parking charges and lot availability. <p>5. Fuel Prices</p> <ul style="list-style-type: none"> • TripTally retrieves current fuel price data from providers. • Driving cost calculations is updated based on user's vehicle profile <p>6. Carbon Footprint</p> <ul style="list-style-type: none"> • The average emission factors per transport mode is retrieved from carbon services. • The system calculates estimated CO₂ footprint per trip. <p>7. Public Transport Data</p> <ul style="list-style-type: none"> • Bus and train schedules retrieved from LTA DataMall. • Fare tables pulled from fare providers, travelling times and costs are computed. <p>8. Calorie API</p> <ul style="list-style-type: none"> • Retrieve calorie expenditure estimates for active travel modes (walking, cycling). • Use external health/fitness APIs to compute calories burnt. <p>9. Ride-Hailing Links</p> <ul style="list-style-type: none"> • Taxi/ride-hailing APIs provide service availability . • The user is presented with options (Grab, CDG), tapping will redirect to app or app store.
Alternative Flows:	AF-1: Data not available
Exceptions:	EX-1: Timeout - Return fallback values after threshold exceeded.
Includes:	<ol style="list-style-type: none"> 1. ViewMap 2. ViewMetrics 3. CompareTransportation
Special Requirements:	<ul style="list-style-type: none"> • Critical feeds must be updated within $\leq 60s$.

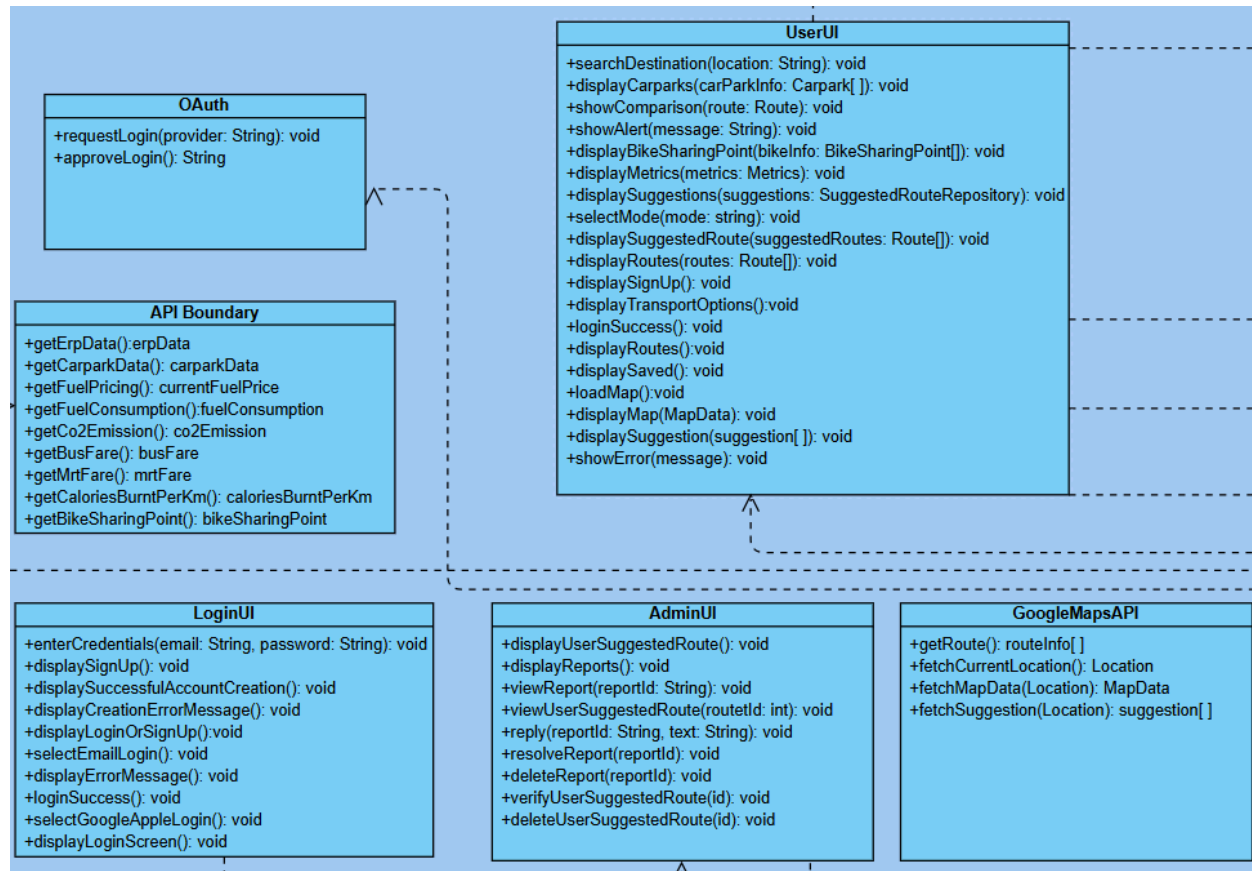
	<ul style="list-style-type: none"> • API responses must integrate into the app within $\leq 2.5s$
Assumptions:	<ol style="list-style-type: none"> 1. External providers maintain reliable APIs and SLAs. 2. TripTally has access to provider data. 3. Users may need to install third-party ride-hailing apps for deep linking.
Notes and Issues:	<p>System must support multi-provider fallback</p> <p>Some private car parks may lack reliable data</p>

2. Class Diagram of Entity Classes

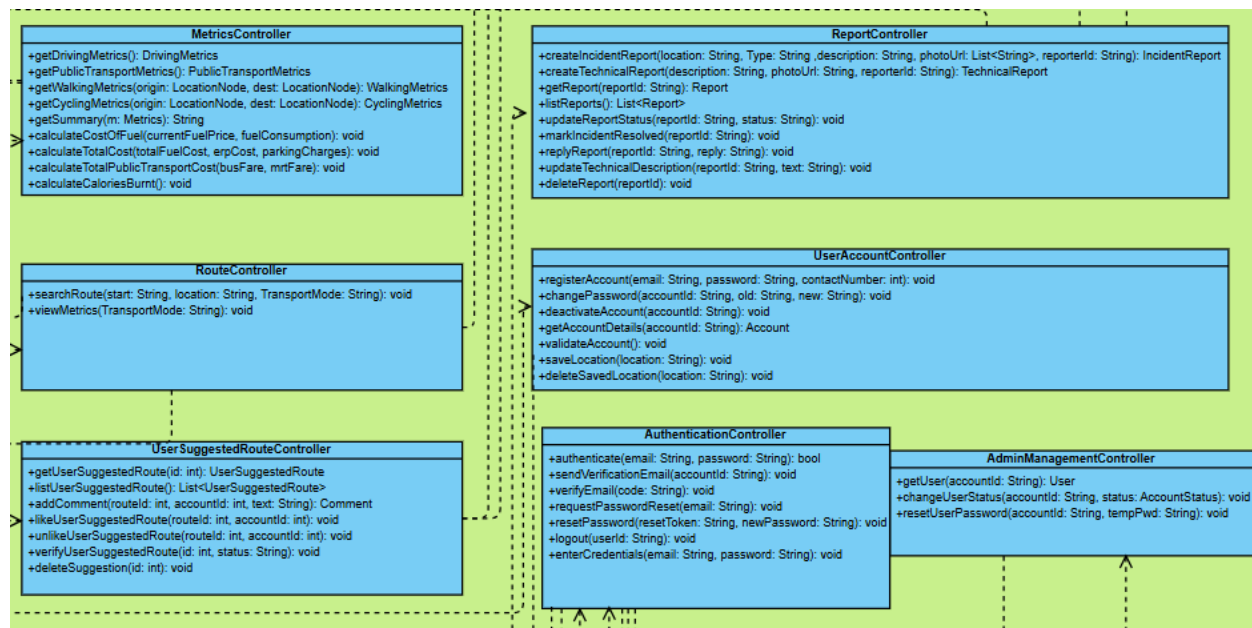


3. Key Boundary Classes and Control Classes

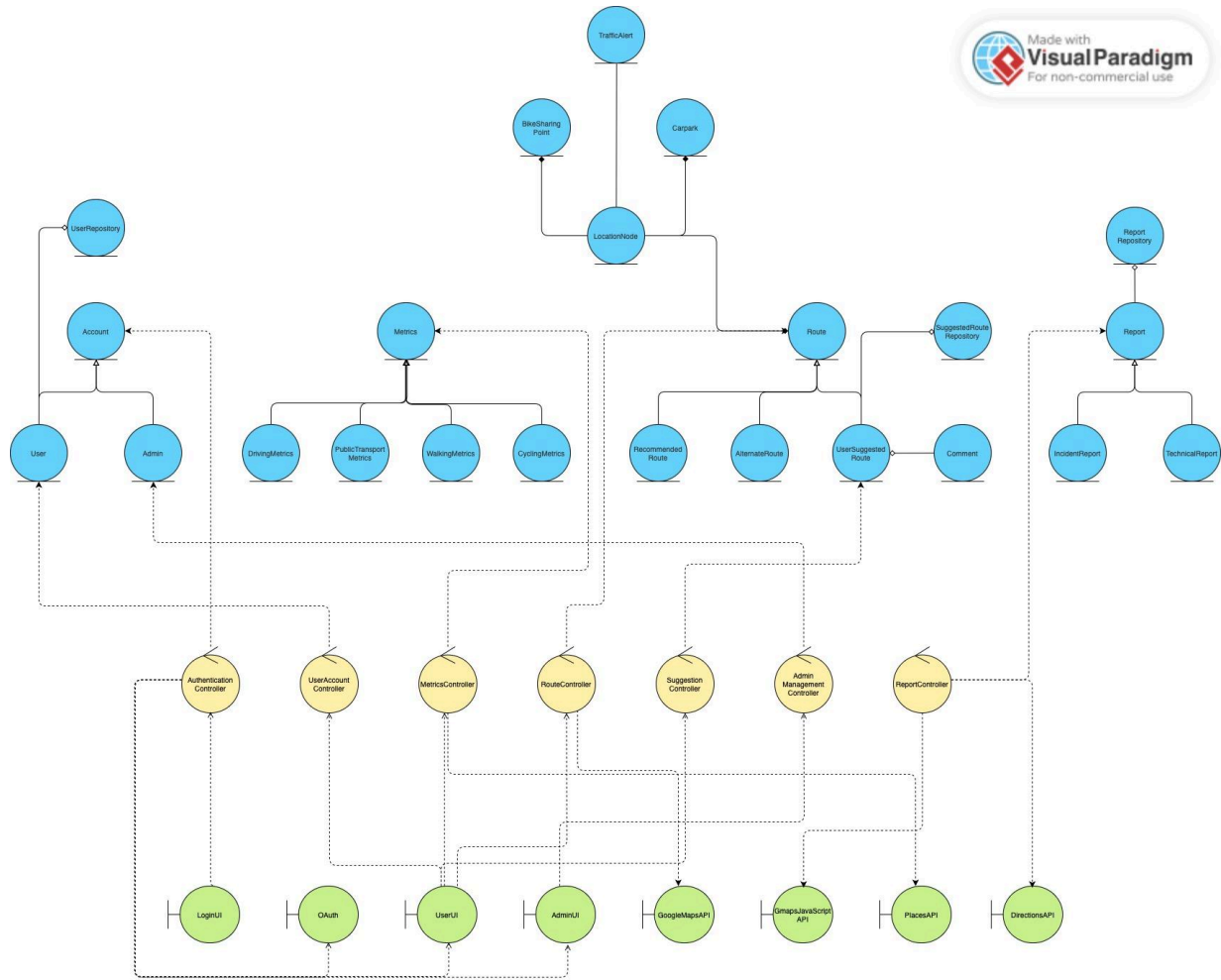
Boundary Class



Control Class



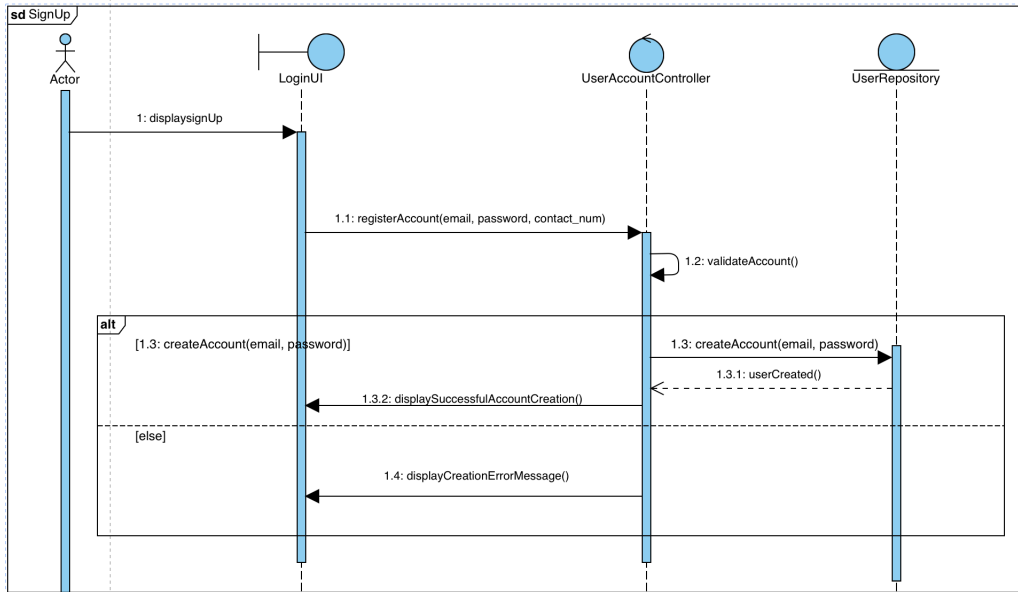
Stereotype Class Diagram



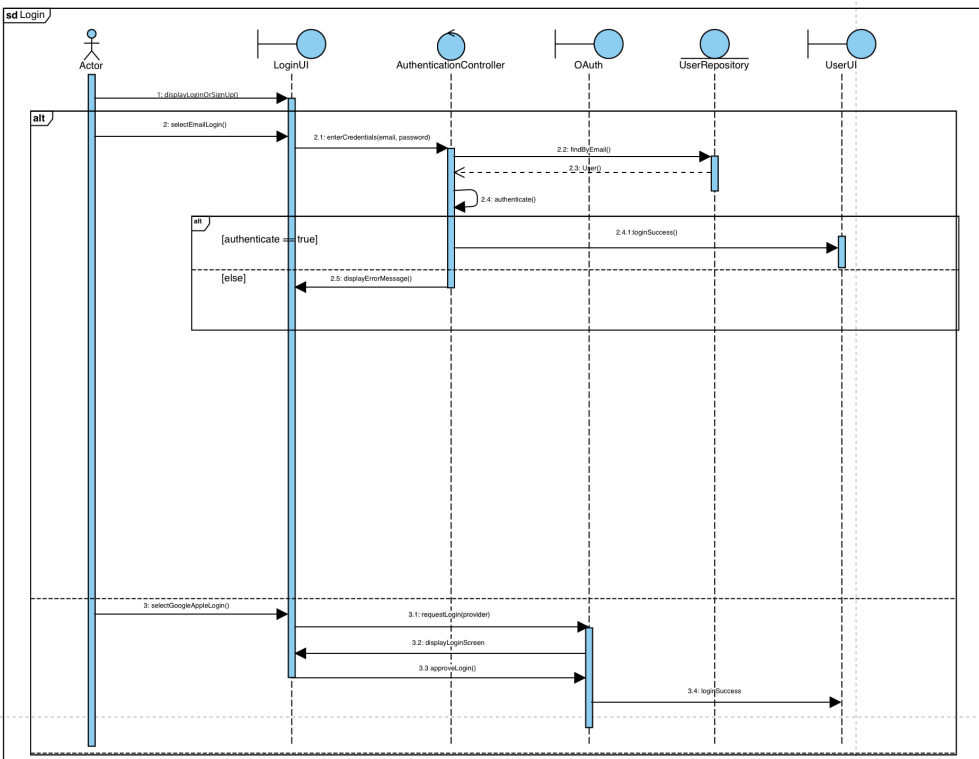
4. Sequence Diagrams of Use Cases

I. For Use Cases under I

I.I SignUp

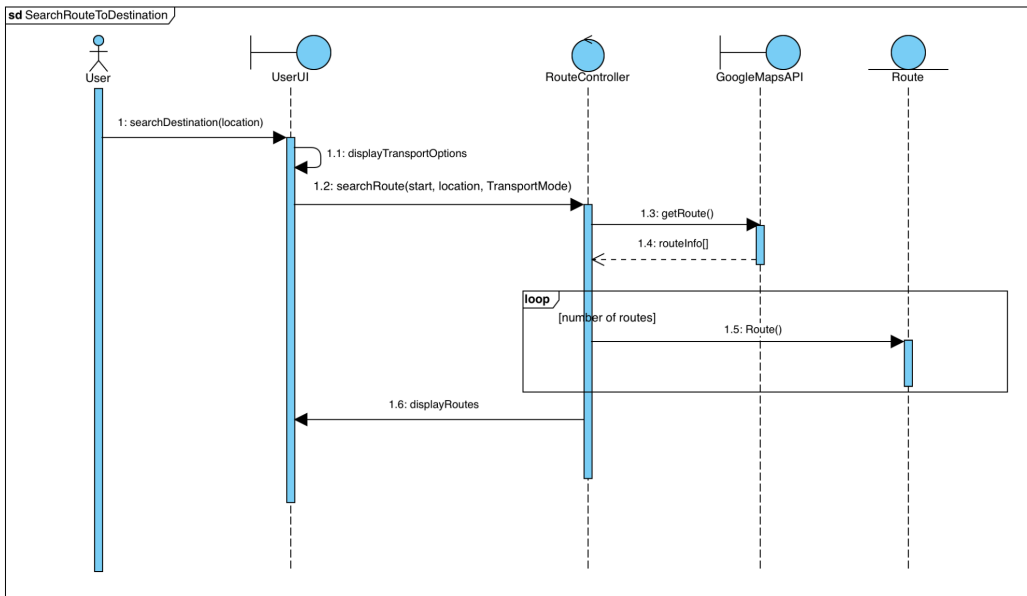


I.II Login

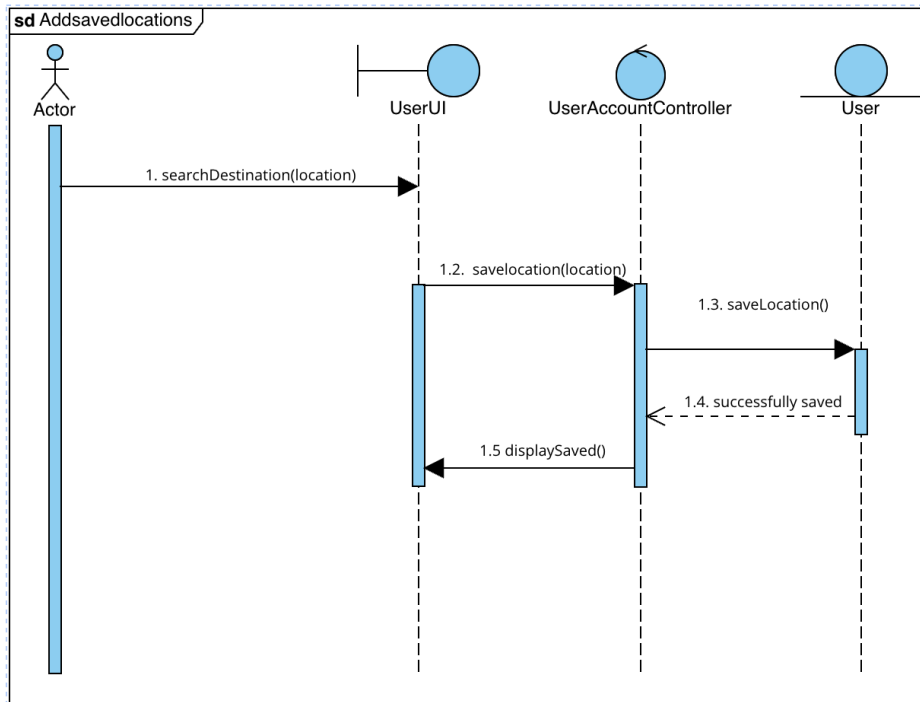


II. For Use Cases Under II

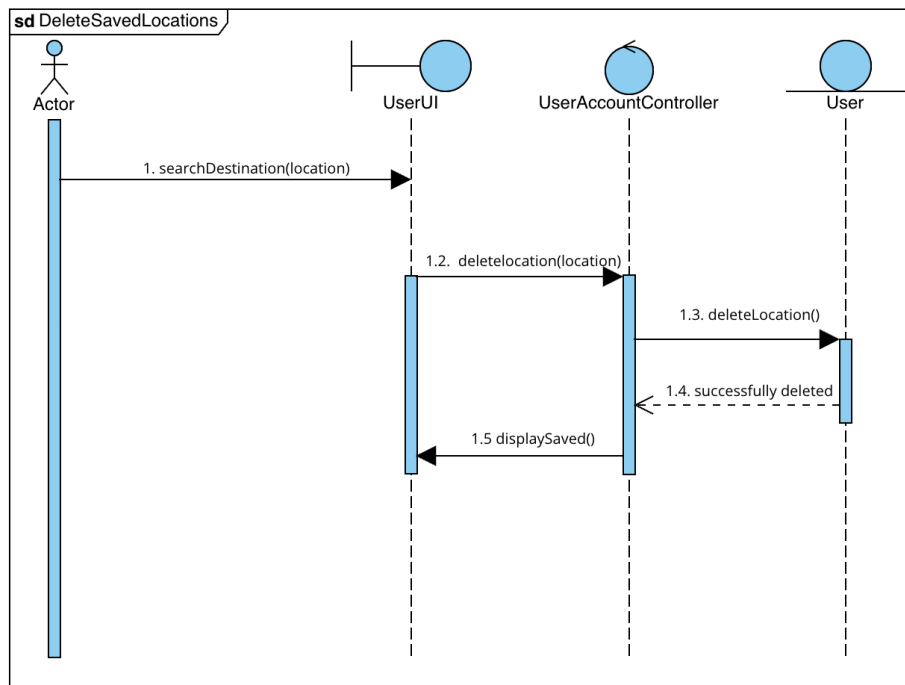
II.I SearchRouteToDestination



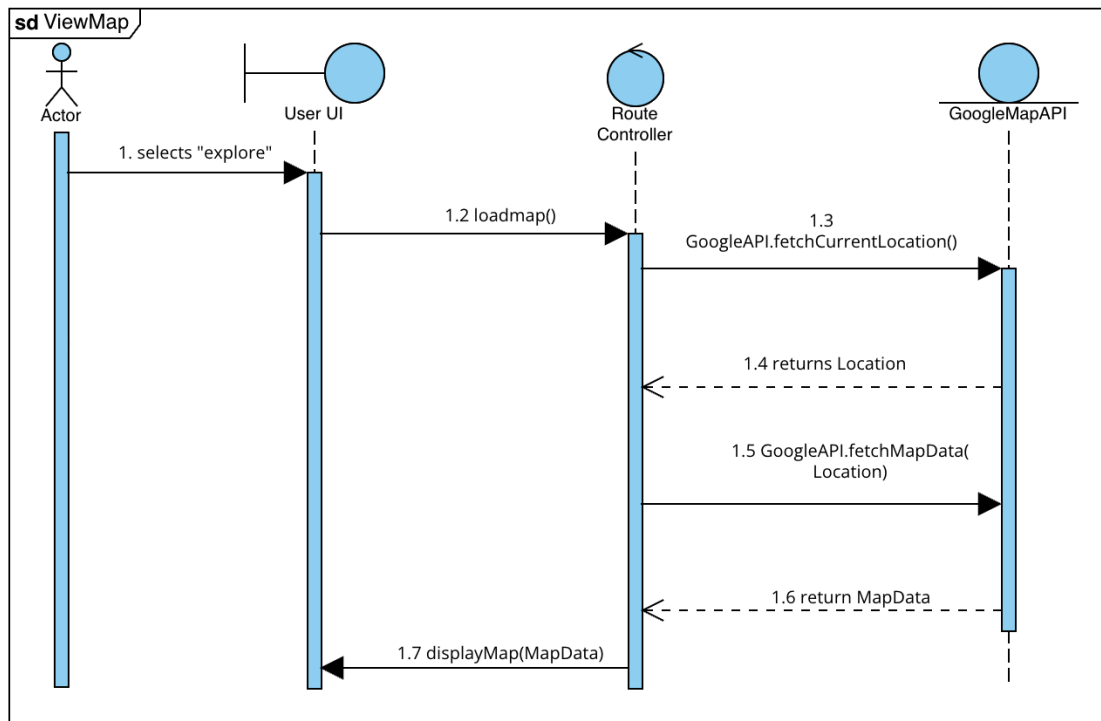
II.II ManageSavedPlaces



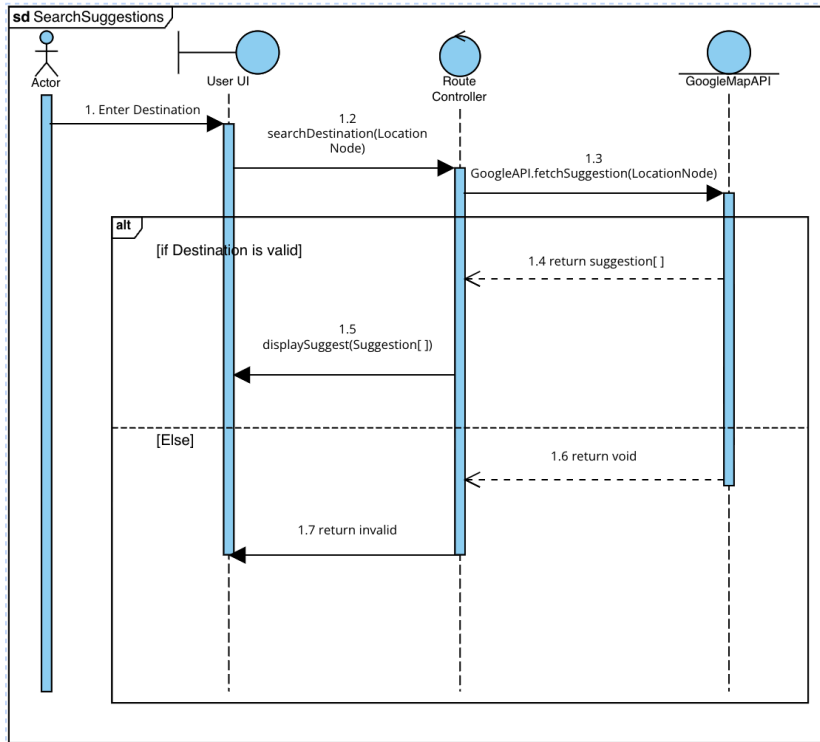
II.III DeleteSavedLocations



II.IV View map

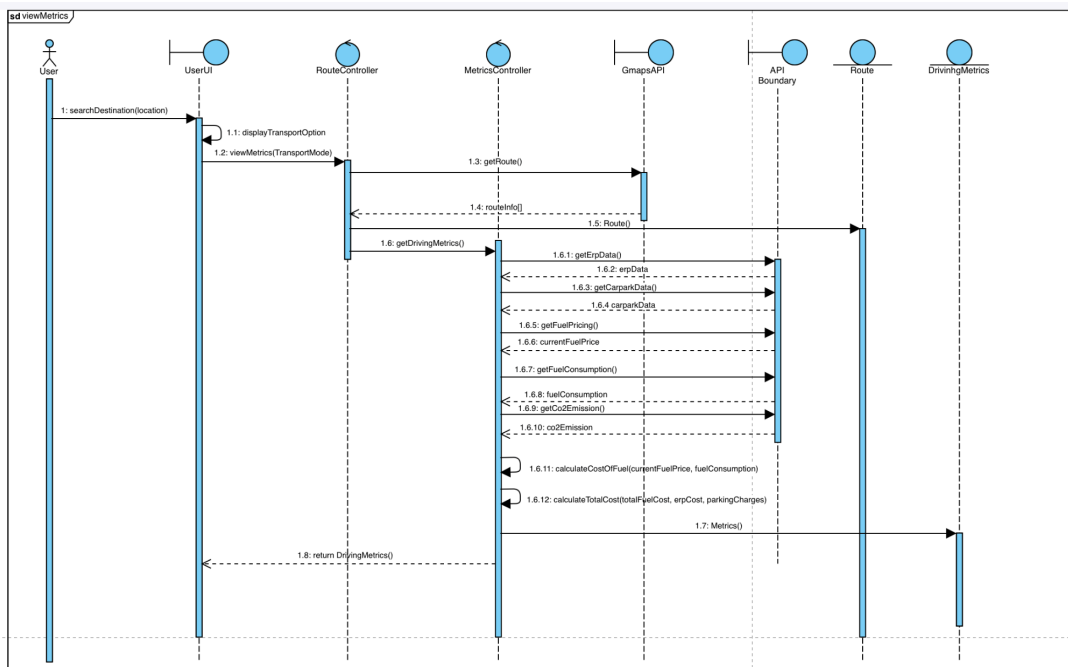


II.V SearchSuggestion

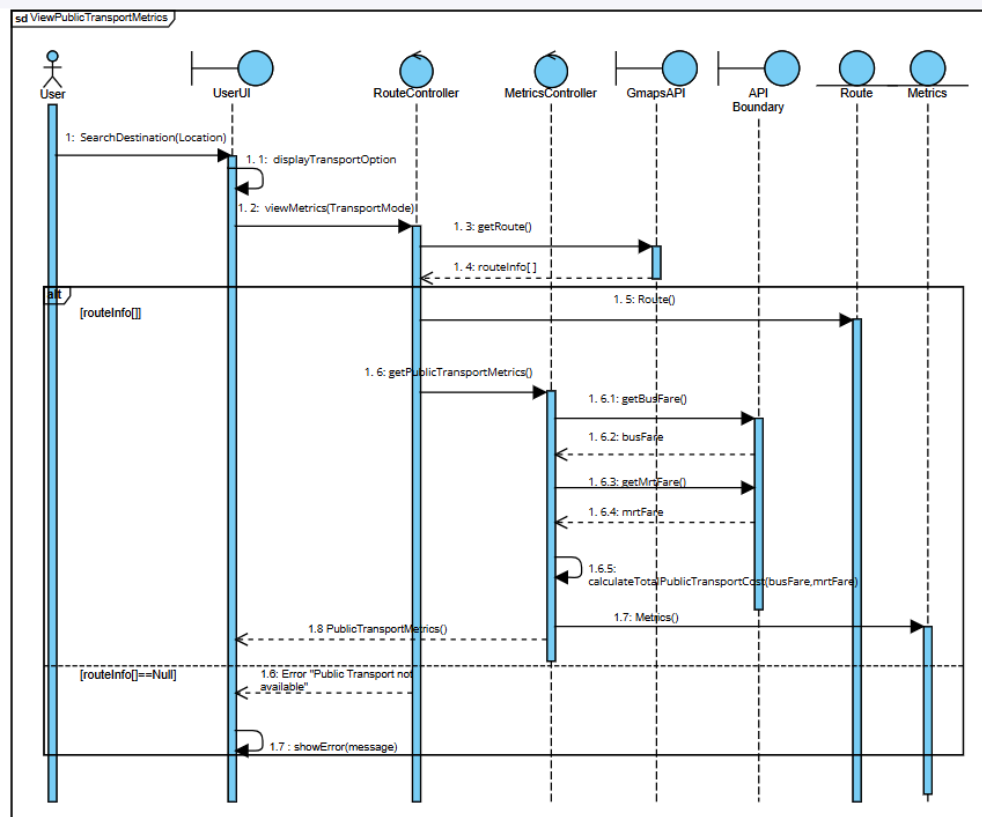


III. For Use Cases Under III

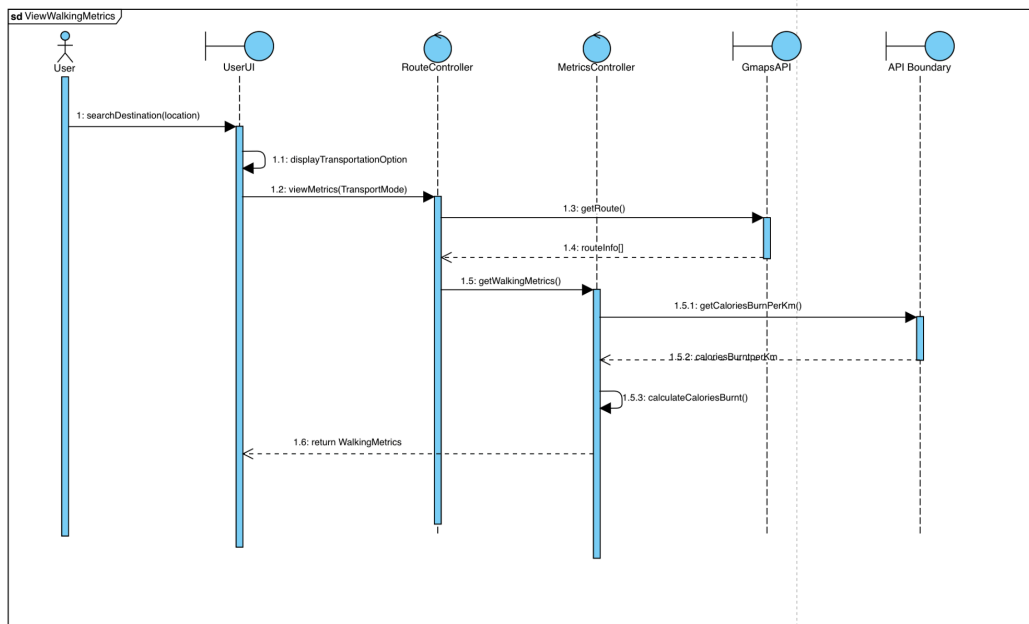
III.I ViewDrivingMetrics



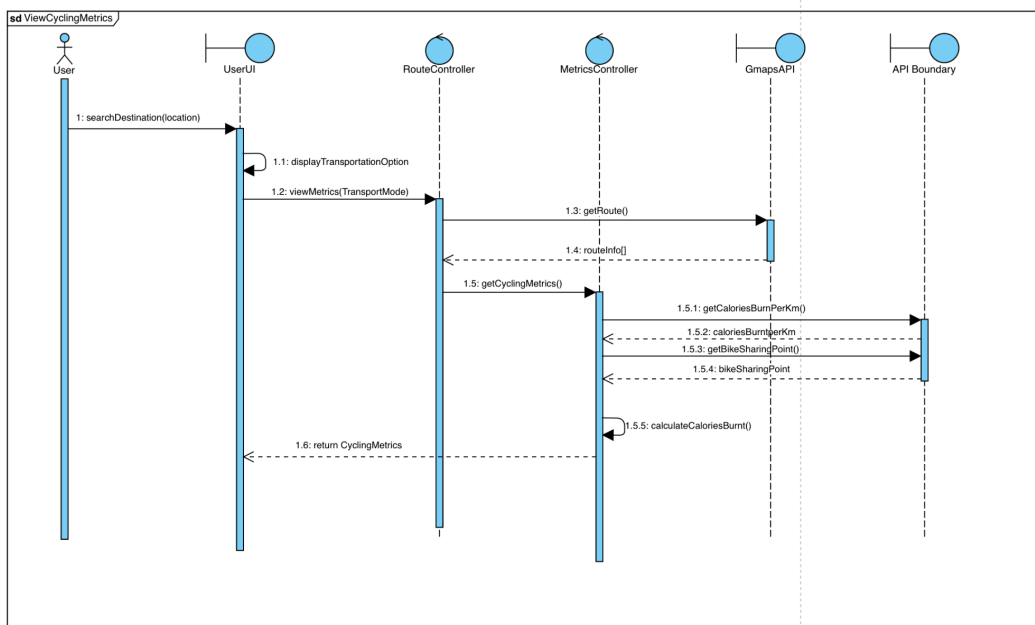
III.II ViewPublicTransportMetrics



III.III ViewWalkingMetrics

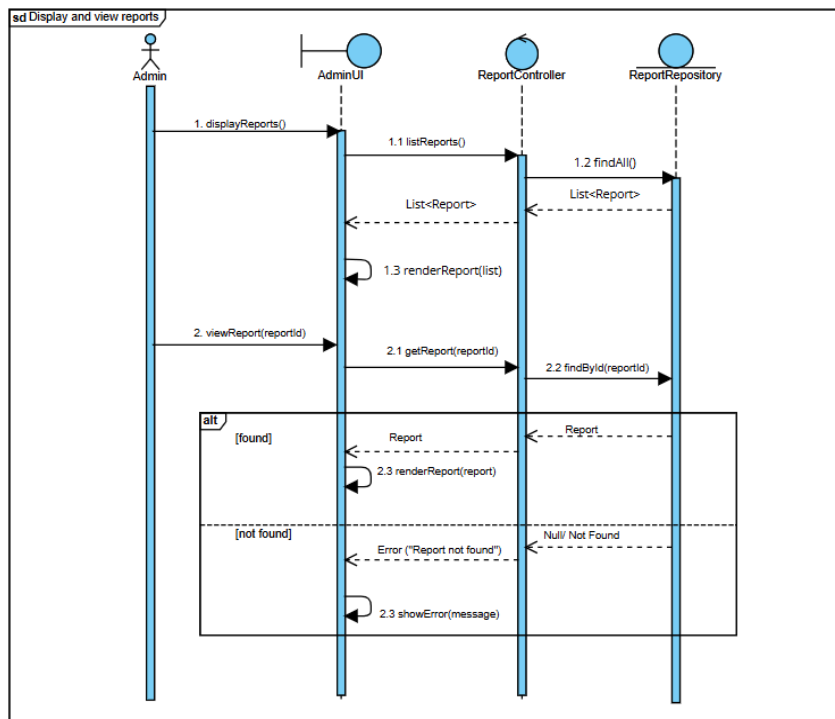


III.IV ViewCyclingMetrics

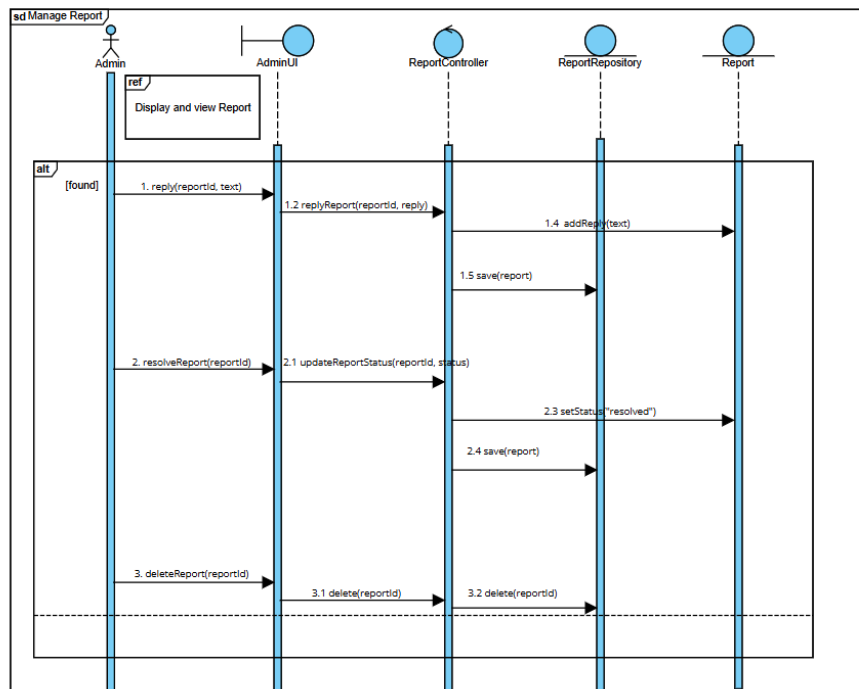


IV. For Use Cases Under IV

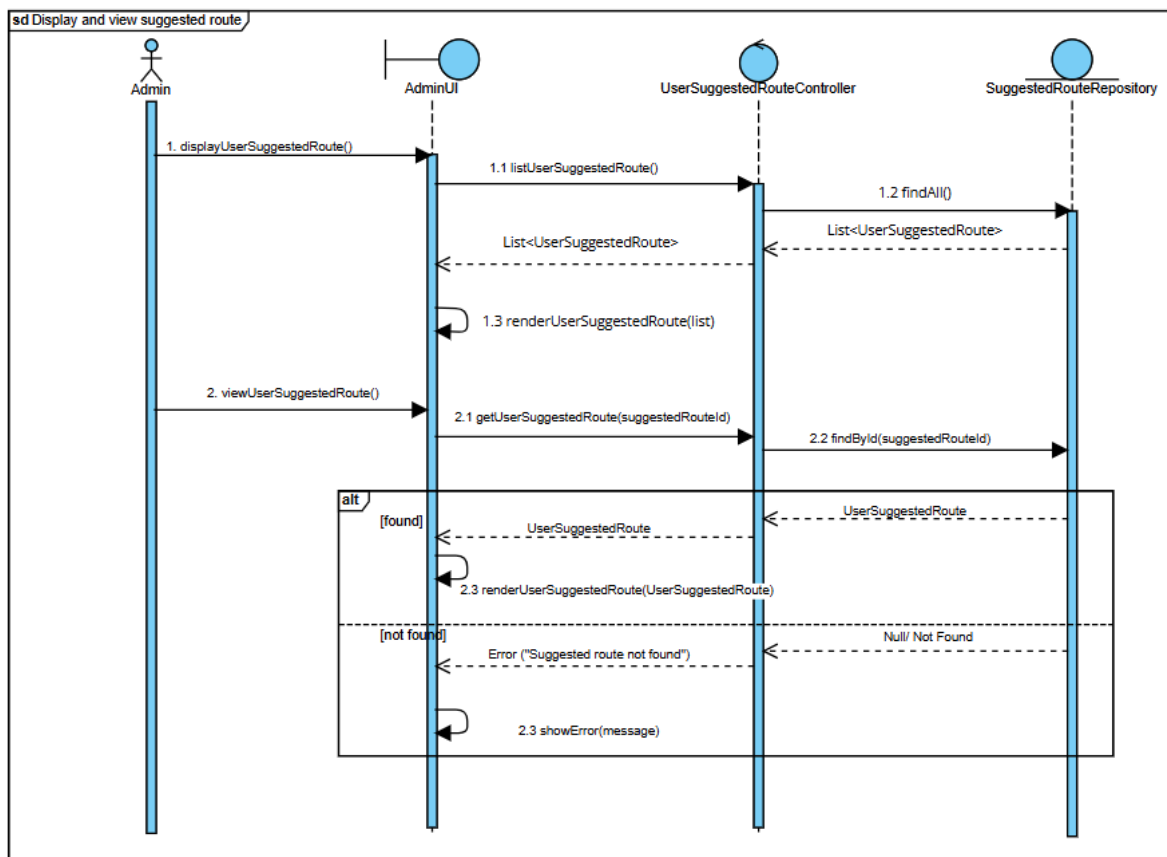
IV.I ViewReports (Admin)



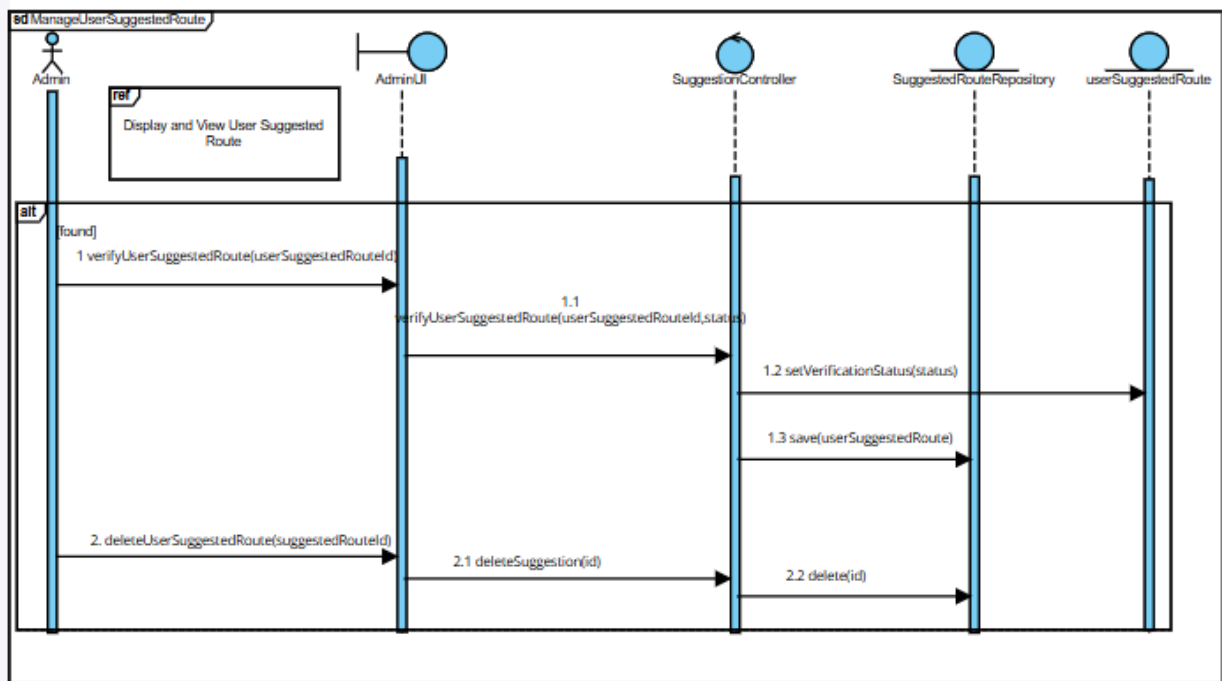
IV.II ManageReport (Admin)



IV.III ViewSuggestedRoute (Admin)



IV.IV ManageUserSuggestedRoute (Admin)



5. Initial Dialog Diagram

