Software Requirements Specification

for

Route Rover

Version 1.5 approved

Prepared by Group: number One
Lab Group: MACS2

Members:
Javen Ong Jing Hian (U2240610E)
Wong Zi Lun (U2240220D)
Oon Yi Rou (U2240133L)
Tan Jun Kiat (U2240182K)
Lim Jingheng, Darius (U2222219L)
Chan Zi Shen (U2240160K)

Table of Contents

Revision History

Name	Date	Reason For Changes	Version
Javen Ong Jing Hian Wong Zi Lun	13/02/2024	Functional & Non-Functional Requirements Use Case Diagram and Description Data Dictionary Initial UI Mockup	1.0
Javen Ong Jing Hian Wong Zi Lun	27/02/2024	Class Diagram Sequence Diagrams Dialog Map	1.1
Javen Ong Jing Hian Wong Zi Lun	12/03/2024	System Architecture Sequence Diagram Edits	1.2
Javen Ong Jing Hian Wong Zi Lun	10/4/2024	Introduction Overall Description	1.3
Lim Jingheng, Darius	11/4/2024	Test Cases	1.4
Javen Ong Jing Hian Wong Zi Lun	19/4/2024	Final Revision	1.5

1. Introduction

1.1. Purpose

This document outlines and specifies the software requirements for the web application Route Rover, version 1.5. This document encompasses a summary of the system, which includes the functional and non-functional requirements, possible use cases, the data dictionary, various diagrams such as class diagrams, sequence diagrams, dialog maps, and system architecture diagrams, and the user interface of the application.

1.2. Document Conventions

This document features the usage of the font Arial of size 11. It must be noted that each new header uses Times of size 18 and is stylized to be bold. Each subheading uses Times of size 16 and is set to be stylised in bold. In numbering our detailed requirements and use cases, we employ a nested (indented) numbering system where a title numbered 1 has subheadings 1.1, 1.2 and so on.

1.3. Intended Audience and Reading Suggestions

This document is intended for the following types of readers:

1. Project Managers
2. Current and future developers
3. Quality Assurance (QA) personnel who are also designing test cases and performing tests on the application
4. Testers
5. Users

The document is organised as follows: Section 1 introduces the structure of the document and provides a brief overview of the software system being developed. Section 2 gives an overview of the product, including its perspectives, functions, target users, operating environment, design and implementation constraints and assumptions. Section 3 gives the interfaces of the product. Section 4 introduces the system features, including the functional requirements, use case model, sequence diagrams, class diagrams, system architecture, design patterns and dialogue map. Section 5 describes the non-functional requirements of the system. Section 6 showcases the acceptance tests conducted for the system.

Project managers should read through all sections of the document to get a comprehensive understanding of the project, its overview, requirements, features, system design and acceptance tests involved.

Developers should primarily focus their attention on the overview, system features, and non-functional requirements, and system design. They may also read through the tests involved to better target their efforts towards passing the testing done.

For quality assurance personnel, they should focus on the system features, non-functional requirements and testing to carry out their duties in enforcing a certain level of quality in the code and performance of the system.

Testers should focus on the system features, non-functional requirements and testing to be aware of the tests that need to be carried out or have been carried out. They can then figure which tests can be improved, redone, modified, or added.

Users might be interested in examining the main functionality of the application. As such, users can mainly read the overview and the system features sections of the report.

1.4. Product Scope

Route Rover is a web-based application that enables users to create, browse and review running routes. The intention is to provide an application which

1.5. References

The following are references for relevant frameworks, packages and libraries used in the application:

Syntactically Awesome Style Sheets (SASS) - https://sass-lang.com/documentation/
Node.js - https://nodejs.org/
Chart.js - https://www.chartjs.org/
Leaflet API - https://leafletjs.com/reference.html
Firebase - https://firebase.google.com/docs

For specific version numbers, please refer to Section 2.4

2. Overall Description

2.1. Product Perspective

Route Rover aims to be a hub of running routes. Its 3 main functions are to create, browse and review running routes. The goal of Route Rover is to give users an easy way to find running routes according to their preferences.

2.2. Product Functions

The following is the high-level summary of the functions of Route Rover:

1. Primary Function: Helps users find curated running routes in Singapore, created by both the app team and other passionate runners.
2. User-Created Content: Provides tools for users to build and share their own running routes.
3. Gamification Element: Features a gamified leaderboard to enhance social interaction and engagement among users.
4. Social Features: Encourages social connections through competition and shared experiences in running.

2.3. User Classes and Characteristics

Route Rover's user base is runners of all levels who are seeking new routes or wanting to share their favourite routes with a community. The user classes are as follows:

- **Exploration of Curated Routes:** Initially, beginners may not know many running routes in their area. Route Rover helps them start exploring with a curated list of beginner-friendly routes. Users can browse these routes, which come with detailed descriptions, including distance, terrain, and scenic points of interest.
- **Progression to Custom Routes:** As users gain more experience and confidence, Route Rover offers the ability to create custom routes. This feature allows them to design routes that suit their specific preferences or training needs, perhaps adding more challenging distances or terrains.
- **MyRun Dashboard:** In the MyRun section, users can track all their saved routes and monitor their progress over time. This personal dashboard is essential for setting goals and reflecting on improvements in distance, speed, and overall running capacity.
- **Competition and Community Engagement:** Route Rover enhances motivation through a leaderboard feature within the MyRun section. Runners can compare their progress and compete with others in the community, which adds a fun and competitive element to their training.
- **Progress Tracking and Analytics:** The app provides comprehensive tracking features that allow users to see how far they've progressed in their running journey. This includes analytics on total distance run, average pace, calories burned, and other relevant metrics.

2.4. Operating Environment

The section will provide the description of the environment in which the software will operate including the hardware platform, operating system and versions together with other software components that must peacefully coexist.

Product Environment of Route Rover: The web application will need to be run using common web browsers like Google Chrome or FireFox

Development Environment of Route Rover:

Development Environment	Description
Front-end: Syntactically Awesome Style Sheets (SASS) (version 1.5.4)	https://sass-lang.com/documentation/ SASS is a CSS preprocessor that simplifies and enhances the way stylesheets are written. It extends CSS with features like variables, mixins, nested rules, and more, enabling developers to write cleaner, more organised, and maintainable code. SASS streamlines the styling process for web development projects, offering increased efficiency and flexibility.
Back-end: Node.js (version 10.5.0)	https://nodejs.org/ Node.js is an open-source, cross-platform JavaScript runtime environment that executes JavaScript code server-side. It allows developers to build scalable and high-performance applications, leveraging the event-driven, non-blocking I/O model. Its extensive package ecosystem, facilitated by npm (Node Package Manager), provides access to countless libraries and frameworks, fostering rapid development.
Back-end (graphing): Chart.js (version 4.0.0)	https://www.chartjs.org/docs/latest/ Chart.js is a JavaScript library for creating interactive and customizable charts and graphs on web pages. It offers a simple yet powerful API for developers to generate a variety of chart types, including line, bar, pie, and more. Chart.js is lightweight, responsive, and easily customizable, making it suitable for a wide range of data visualisation needs.
API: Leaflet (version 1.9.4)	https://leafletjs.com/reference.html Leaflet is an open-source JavaScript library for mobile-friendly interactive maps. It provides a simple yet powerful API for developers to embed maps into web pages and customise them with various layers, markers, and interactions. Leaflet is lightweight, fast, and highly extensible, making it good for building responsive and interactive mapping applications.

Database: Cloud Firestore and Firebase Storage by Firebase (version 10.10.0)	https://firebase.google.com/docs Firebase is a comprehensive web application development platform provided by Google. It offers Cloud Firestore for real-time NoSQL database storage and Cloud Storage for storing user-generated content like images. Firestore supports real-time data synchronisation and offline data persistence. Cloud Storage provides secure and reliable cloud storage infrastructure, facilitating efficient file uploads, downloads, and management.

2.5. Design and Implementation Constraints

Component Occlusion Due to Display Size/Type
The developers and testers are limited in the variety of screen sizes and types of screens that they have access to, so not all sizes of displays and types of displays can be tested to ensure that there is no occlusion issue. This issue is more pertinent in recent years as smartphone makers are experimenting with various screen sizes/types (foldable display, hole-punch display, notch display, etc.). Laptop and monitor makers are also making available newer displayers with various aspect ratios. For the devices that developers and testers have access to, it can be guaranteed that there are no occlusion issues.

Hosting Constraints
The primary database for Route Rover is Cloud Firestore hosted by Firebase. The free version of Cloud Firestore imposes a maximum data size constraint of 1GiB. This constraint should be considered when managing data storage within the application to avoid potential data loss or service interruptions. Regular monitoring and optimisation of data storage practices should be performed to stay within the constraints of the hosting environment.

2.6. User Documentation

A demo video on the flow of the application will be attached and shown to provide users with explanations on the navigation of the application.

To further ease developers in setting up the web application, README files with specific instructions to set up the environment for both front-end and back-end would be provided for the users.

2.7. Assumptions and Dependencies

Assumptions
There is an implicit assumption that users must be connected to the internet to access the application, as it is a web application after all. Depending on the strength of the internet connection, some images may not be able to load properly due to bandwidth constraints. If there is no connectivity, then the application will not be functional.

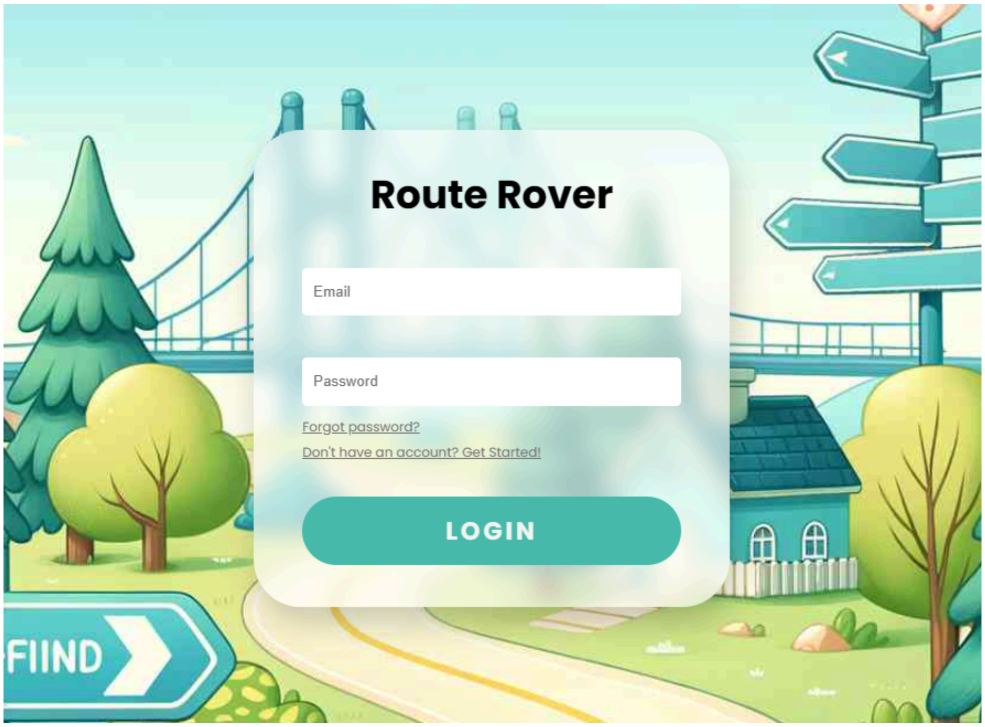
Dependencies

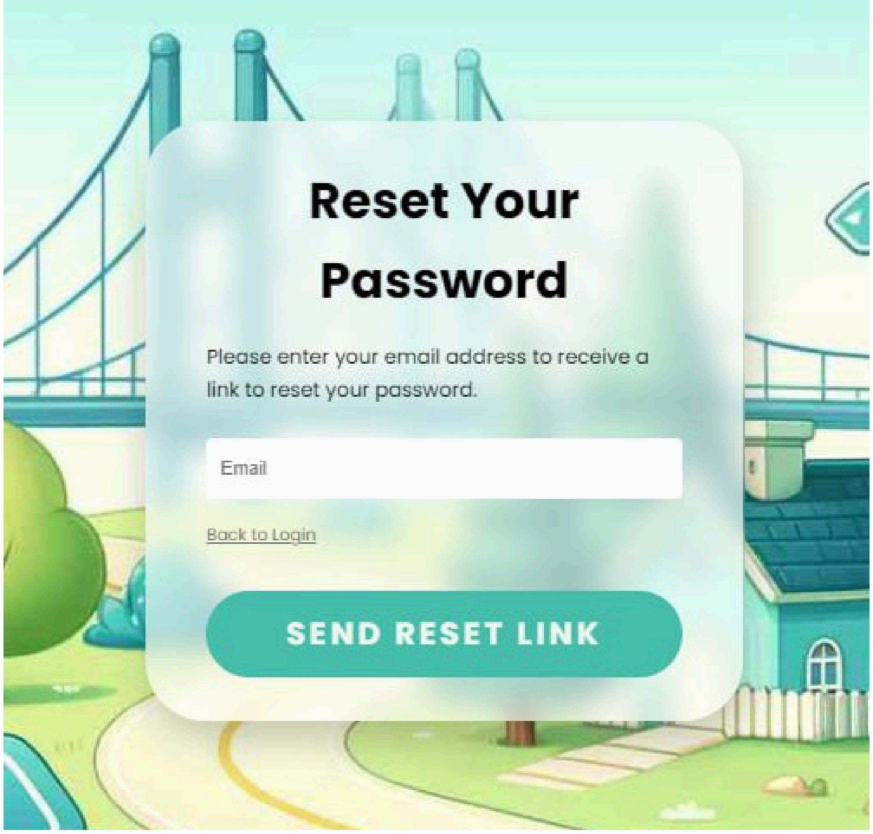
1. The system relies on Leaflet.js to render interactive maps. Leaflet.js provides the foundation for displaying maps and handling user interactions like zooming and panning. Any modifications to Leaflet.js that affect its compatibility or functionality may impair the system's ability to display maps correctly.
2. The system generates routes between points on the map using the Leaflet Routing Machine. This module is necessary for presenting users with route information and allows for addition, amendment, and removal of waypoints, as well as other core functionalities. Any changes or updates to the Leaflet Routing Machine module may require system modifications to preserve routing capabilities and waypoint management functionalities.

3. External Interface Requirements

3.1. User Interfaces

Route Rover allows users to access its functionalities through graphical and textual means. Its UI interfaces include elements such as buttons, text fields, sliders and menus that enable users to input commands and data.

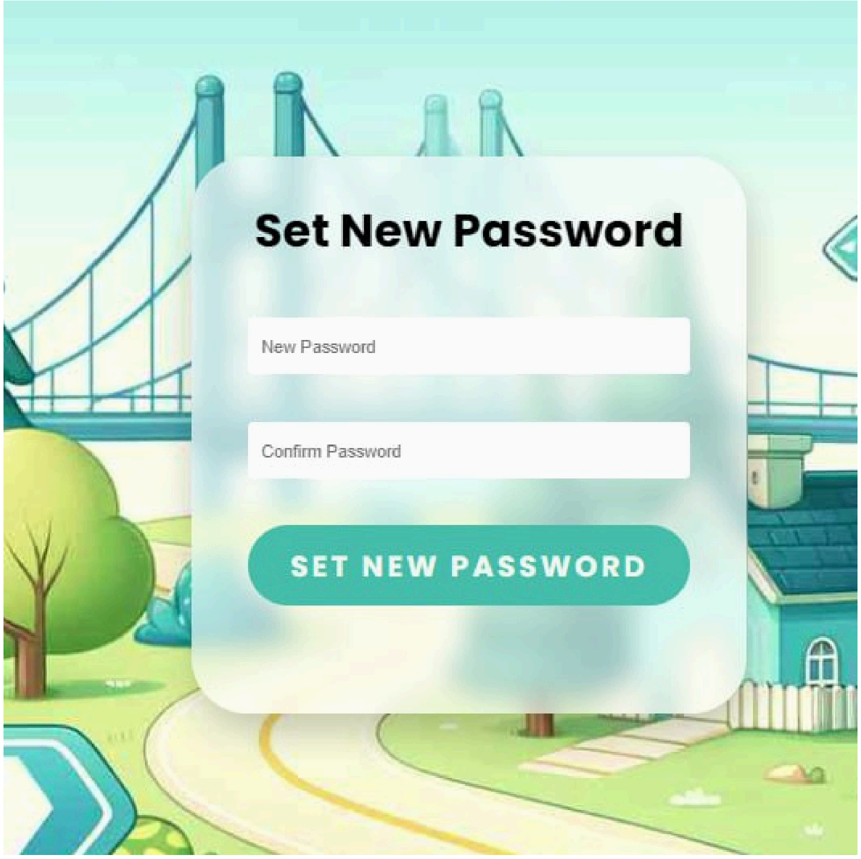




Reset Your Password

Please enter your email address to receive a link to reset your password.

Email



Set New Password

New Password

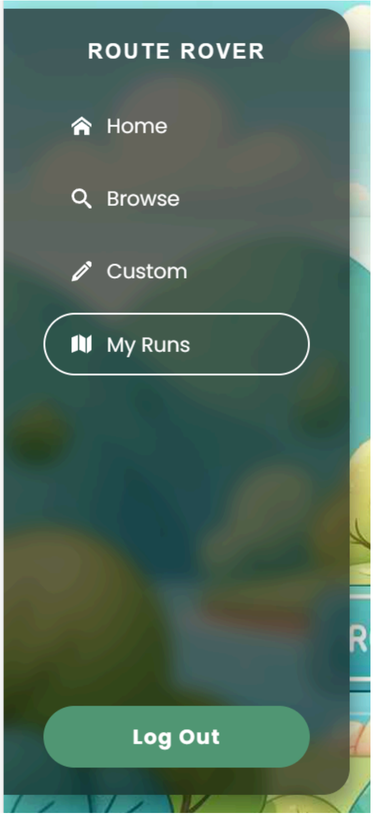
Confirm Password

3.1.1. Buttons

- Used to trigger actions or submit forms.

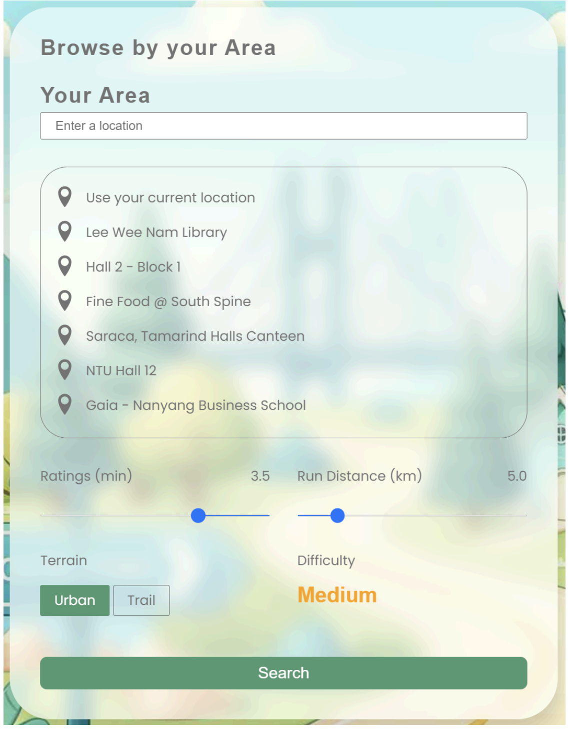
3.1.2: Text Fields

- Allow users to input text or numbers.



3.1.3 Menu

- Provide a list of options for users to choose from.



Browse by your Area

Your Area

Enter a location

- Use your current location
- Lee Wee Nam Library
- Hall 2 - Block 1
- Fine Food @ South Spine
- Saraca, Tamarind Halls Canteen
- NTU Hall 12
- Gaia - Nanyang Business School

Ratings (min) 3.5

Run Distance (km) 5.0

Terrain

Urban Trail

Difficulty

Medium

Search

3.1.4 Sliders

- Enable users to select a value from a range by dragging a handle along a track.

3.1.5 Checkboxes

- Used to toggle a binary option on or off.

3.2. Hardware Interfaces

There are no hardware interfaces beyond the use of personal computers or smartphones to access Route Rover. If the device being used to access Route Rover has a modern browser (Chrome, Safari, etc.) and an internet connection through wired or wireless means, Route Rover will function as intended.

3.3. Software Interfaces

Maps JavaScript API

Maps JavaScript API is a tool by Google that allows the embedding of Google Maps into web pages using JavaScript code. Maps JavaScript is used to create the interactive map in Route Rover with Google Maps.

Leaflet API

Leaflet is a JavaScript library used to create interactive maps. It is used to create the interactive map in Route Rover for users to visualise pre-existing running routes and to create new running routes. Leaflet is used to add lines on the interactive map to represent the running routes on the interactive map.

Firebase - Cloud Firestore
Cloud Firestore is a NoSQL database that is hosted on the cloud by Firebase by Google that acts as the primary database for Route Rover. The data stored within the database are records of each account with its username, password and email and routes with routeID, route information and route review.

The communication between the database and Route Rover is done through Javascript code to fetch/update data from the database depending on which webpage is used.

3.4. Communications Interfaces

HTTP Communication (Client to Web App and Frontend to Backend)

- HTTP is a protocol for fetching resources such as HTML documents. It is the foundation of any data exchange on the Web and it is a client-server protocol, which means requests are initiated by the recipient, usually the Web browser. A complete document is reconstructed from the different sub-documents fetched, for instance, text, layout description, images, videos, scripts, and more.

4. System Features

4.1. Account Management

4.1.1 Description and Priority

1. Allows users to create an account
2. Allows users to log in to their account
3. Allows users to log out of their account
4. Allows users to change their account password (Not sure if keeping)

Priority: High

4.1.2 Stimulus/Response Sequences

Use Case ID:	1.1		
Use Case Name:	Signup		
Created By:	Wong Zi Lun	Last Updated By:	Wong Zi Lun
Date Created:	07/02/2024	Date Last Updated:	25/02/2024

Actor:	User (Initiating Actor), User DB
Description:	Allows users to create an account
Preconditions:	The User does not already have an Account with User DB
Postconditions:	The User creates an Account or The User receives an explanation from the System why an Account could not be created
Priority:	High
Frequency of Use:	Frequent
Flow of Events:	1. The User selects the option “Don't have an account? Get Started!” 2. The System opens a Signup page 3. The System requests the input of the User’s particulars – Email and Password. 4. The User enters their particulars into all mandatory text fields – Email and Password. 5. The User submits the Registration Form by selecting the option ‘Sign Up’.

	6. The System verifies that all mandatory text fields are filled in. 7. The System verifies that all text fields are filled with valid data. 8. The System verifies that the User does not already have an account in User DB 9. The System creates an Account with the corresponding particulars for the User
Alternative Flows:	AF-S6: If a mandatory text field is empty 1. The System displays the message “Please enter both email and password” 2. The System returns to step 3 AF-S7: If the text fields contain invalid data 1. The System displays the message “Invalid email or password” 2. The System returns to step 3 AF-S8: If the User already has an Account in User DB 1. The System displays the message “Invalid email or password” 2. The System returns to step 3
Exceptions:	
Includes:	
Special Requirements:	
Assumptions:	
Notes and Issues:	

Use Case ID:	1.2		
Use Case Name:	Login		
Created By:	Wong Zi Lun	Last Updated By:	Wong Zi Lun
Date Created:	07/02/2024	Date Last Updated:	31/03/2024

Actor:	User (Initiating Actor), User DB
Description:	Allows the User to login to their Account
Preconditions:	The User has created an Account before
Postconditions:	The User is logged into their Account
Priority:	High

Frequency of Use:	Frequent
Flow of Events:	1. The User launches the Route Rover application 2. The System opens the Login page 3. The System requests the input of the User's Account's email and password 4. The User enters the email and password of their Account 5. The User selects the option 'Login' 6. The System verifies that the email and password matches an Account's details in User DB 7. The System logs the User into their Account 8. The System calls Use Case 2.1 to enter the Home page
Alternative Flows:	AF-S6: The username or password the User entered is invalid 1. The System displays the message "Invalid email or password" 2. The System returns to step 3
Exceptions:	
Includes:	
Special Requirements:	
Assumptions:	
Notes and Issues:	

Use Case ID:	1.3		
Use Case Name:	Log Out		
Created By:	Wong Zi Lun	Last Updated By:	Wong Zi Lun
Date Created:	07/02/2024	Date Last Updated:	25/02/2024

Actor:	User (Initiating Actor)
Description:	Allows the User to logout of their Account
Preconditions:	The User is logged into their Account
Postconditions:	The User is logged out of their Account
Priority:	High
Frequency of Use:	Frequently

Flow of Events:	1. The user selects the option 'Log Out' 2. The system logs the user out of their Account 3. The system displays the login page
Alternative Flows:	
Exceptions:	
Includes:	
Special Requirements:	
Assumptions:	
Notes and Issues:	

Use Case ID:	1.4		
Use Case Name:	Forget Password		
Created By:	Wong Zi Lun	Last Updated By:	Wong Zi Lun
Date Created:	07/02/2024	Date Last Updated:	25/02/2024

Actor:	User (Initiating Actor), UserDB
Description:	Allows the User to reset the password of their Account
Preconditions:	The User has created an Account before
Postconditions:	The User's Account password is changed
Priority:	High
Frequency of Use:	Occasionally
Flow of Events:	1. The User selects the option 'Forget password' 2. The System opens the Forget Password Page. 3. The System requests the input of the User's Account's email address 4. The User inputs their Account's email address 5. The User selects 'Submit' 6. The System verifies that the email address is valid 7. The System sends a link to the User's email address to reset their password 8. The User selects the link in the email 9. The System requests the input of the User's Account's new password and confirmation of their new password

	10. The User inputs their new password, and re-enters their new password for confirmation 11. The System verifies the input of the User's new password matches their confirmation 12. The System changes the password of the User's Account to the new password
Alternative Flows:	AF-S6: The email address is invalid 1. The System displays the message "Invalid email address! Please try again!" 2. The System returns to step 3 AF-S11: The new password and the confirmation do not match 1. The System displays the message "Password does not match! Please try again!" 2. The System returns to step 9
Exceptions:	
Includes:	
Special Requirements:	
Assumptions:	
Notes and Issues:	

4.1.3 Functional Requirements

1. Users must be able to create an account.
    1.1. The system will display text fields for users to enter user particulars including: Email and Password
    1.2. All mandatory fields must be filled in before account creation.
        1.2.1. The system must ensure that all fields are filled.
            1.2.1.1. The system will prompt the user to "Please enter both email and password" when there are unfilled fields
            1.2.1.2. The system must ensure that all fields are filled with valid data
                1.2.1.2.1. The system will prompt "Invalid email or password" when there is invalid data in the fields.
2. Users must be able to log in to their accounts
    2.1. The system will display text fields for users to enter user particulars including: Email and Password
        2.1.1. The system must ensure that all fields are filled with valid account details
            2.1.1.1. The system will prompt "Invalid email or password" when there is invalid account details in the fields.
3. Users must be able to log out of their accounts
4. User must be able to use Forgot password button to recover account if they have forgotten their password
    4.1. The system shall ask for the user's email address

4.2. The system must send an email to the user’s email address to verify and send a reset password link

4.3. After user clicks the link the system will then show a reset password page with 2 text fields to enter a new password and re-enter the new password

4.3.1. User cannot use previously used passwords

4.2. Home Page

4.2.1 Description and Priority

1. Home page of the system which curates routes for the user and displays progress graph of the user’s runs
2. Changing the statistic of the progress graph in home page

Priority: Medium

4.2.2 Stimulus/Response Sequences

Use Case ID:	2.1		
Use Case Name:	Go to Home Page		
Created By:	Javen Ong Jing Hian	Last Updated By:	Javen Ong Jing Hian
Date Created:	08/02/2024	Date Last Updated:	31/03/2024

Actor:	User (Initiating Actor), Runs DB, Route DB
Description:	Home page of the system which curates routes for the user and displays progress graph of the user’s runs
Preconditions:	The User is logged into their Account
Postconditions:	System displays the user's total runs, total running time in hour, minute, second format, total distance ran in kilometres, average running speed in minute, second per km format and fastest running speed in minute, second per km format. The system will display a graph of the user’s average speed of each run System displays 6 curated routes for the user
Priority:	High
Frequency of Use:	Frequent

Flow of Events:	1. The User clicks on the option ‘Home’ on the system sidebar user interface 2. The System retrieves the User’s Username from User DB. 3. The System retrieves the statistics of the User’s runs from Runs DB. 4. The System generates a curated list of routes for the User from Route DB. 5. The System calculates and displays the User’s run progress statistics – Total runs, Running Time (in hours, minutes, seconds format), Distance ran (in kilometres and to 2 decimal points), Average speed (in minutes and seconds per kilometre) and Fastest run (in seconds per kilometre) 6. The System displays a progress graph that can show the Average Speed (in minute per kilometre), Distance (in kilometre) or Duration (in hours) of the User’s past runs. 7. The System displays the 6 curated routes with their details – image attached to the route, name of the route, username of the creator of the route, number of users that have finished the route, distance information of the route in kilometres, terrain information (urban or trail) of the route and estimated time of completion of the route in minutes.
Alternative Flows:	
Exceptions:	EX1: The User is logging in to their account 1. The System jumps to step 2
Includes:	
Special Requirements:	
Assumptions:	
Notes and Issues:	

Use Case ID:	2.2		
Use Case Name:	Toggle progress graph		
Created By:	Javen Ong Jing Hian	Last Updated By:	Javen Ong Jing Hian
Date Created:	08/02/2024	Date Last Updated:	31/03/2024

Actor:	User (Initiating Actor), Runs DB, Route DB

Description:	Changing the statistic of the progress graph in home page
Preconditions:	User is at the Home page
Postconditions:	The system will display a graph of the user's run statistic depending on the choice
Priority:	Medium
Frequency of Use:	Occasionally
Flow of Events:	1. The User selects the statistic to display – Average Speed (in minute per kilometre), Distance (in kilometre) or Duration (in hours) of the User's past runs. 2. The System displays the progress graph of the chosen statistic.
Alternative Flows:	AF-S1: The User selects an already active statistic on the progress graph 1. The System does not do anything
Exceptions:	
Includes:	3 statistics to choose from: Average Speed (in minute per kilometre), Distance (in kilometre) or Duration (in hours) of the User's past runs.
Special Requirements:	
Assumptions:	
Notes and Issues:	

4.2.3 Functional Requirements

1. The system will display a welcome back message with the user's username
2. The system will display the user's run progress statistics
   - 2.1. The system will display the user's total runs.
   - 2.2. The system will display the user's total running time in hour, minute, second format.
   - 2.3. The system will display the user's total distance ran in kilometres.
   - 2.4. The system will display the user's average running speed in minute, second per km format.
   - 2.5. The system will display the user's fastest running speed in minute, second per km format.
   - 2.6. The system will display a graph of the user's runs.
     - 2.6.1. The system will display an option to display the user's average running speed in minutes per km by runs.
     - 2.6.2. The system will display an option to display the user's distance ran in km by runs.

2.6.3. The system will display an option to display the user's run durations in hours by runs.

3. The system shall display a list of 6 curated routes for the user.

3.1. The system will display the details of each route.

3.1.1. The system will display the image attached to the route.

3.1.2. The system will display the name of the route.

3.1.3. The system will display the username of the creator of the route.

3.1.4. The system will display the number of users that have finished the route.

3.1.5. The system will display the distance information of the route in kilometres.

3.1.6. The system will display the terrain information (urban or trail) of the route.

3.1.7. The system will display the estimated time of completion of the route in minutes.

3.2. Users can select the route to view more information.

3.2.1. The system will open the route in the Browse Routes page.

4.3. Browse Routes

4.3.1 Description and Priority

1. Gives a list of routes based on the location and specifications user has input

Priority: High

4.3.2 Stimulus/Response Sequences

Use Case ID:	3		
Use Case Name:	Browse routes		
Created By:	Javen Ong Jing Hian	Last Updated By:	Wong Zi Lun
Date Created:	08/02/2024	Date Last Updated:	31/03/2024

Actor:	User (Initiating Actor), Route DB
Description:	Gives a list of routes based on the location and specifications user has input
Preconditions:	User is logged into their account
Postconditions:	User gets a list of routes according to search
Priority:	High
Frequency of Use:	Frequent
Flow of Events:	1. The User selects the option 'Browse' on the system sidebar user interface

	2. The System creates a Search Form 3. The System displays the options for the User to choose a location by input into a text field, using their current location or selecting a location from a list of suggested locations based on past User inputs 4. The System displays an option to toggle the minimum rating (from 0.0 to 5.0 in intervals of 0.5) of the running routes to search for. 5. The System displays an option to toggle the maximum distance (from 0.5 km to 30 km in intervals of 0.5 km) of the running routes to search for. 6. The System displays the difficulty of the run the user is searching for as Easy (0.5 km to 2.5 km), Medium (3.0 km to 8.0 km), Hard (8.5km to 15.0 km) or Expert (above 15.5km) based on the maximum distance chosen. 7. The System displays an option for the User to choose the terrain type (urban or trail) of the running routes to search for. 8. The System displays an option to 'Search' for the routes. 9. The System displays an interactive map from Leaflet Map API. 10. The User chooses the location, minimum rating, maximum run distance and terrain type of the route they want 11. The User selects the option "Search" 12. The System verifies that a valid location is selected 13. The System retrieves a list of routes that matches the user's requirements and their names, usernames of their creator, average review ratings, number of users that have finished the route, distance information in kilometres, terrain information (urban or trail) and estimated time of completion from Route DB 14. The System displays the list of routes retrieved and their respective details
Alternative Flows:	AF-S13: A valid location is not chosen 1. The System displays the message "invalid location" 2. The System returns to step 10
Exceptions:	EX1: The User is returning back to the Browse page from viewing a route 1. The System jumps to step 2
Includes:	
Special Requirements:	
Assumptions:	
Notes and Issues:	

4.3.3 Functional Requirements

1. Users can choose a location to search for running routes.

1.1. The system will display a text field for users to enter a location.

1.1.1. The system will suggest locations to autocomplete the user's current text input in the text field

1.1.2. The user will be able to select a suggested location to autocomplete the text in the text field.

1.2. The system will display an option for users to use their current location.

1.2.1. The system will prompt users if they allow the system to use their current GPS location.

1.2.2. The system shall show an error message "Unable to locate" when the user does not allow the system to access his/her location.

1.3. The system will display a list of suggested locations based on past inputs by the user.

2. Users can choose filters to apply to the running routes they search for.

2.1. The system will display filters for the users to specify the restrictions of the routes to search for.

2.1.1. The system will display a slider to toggle the minimum rating (from 0.0 to 5.0 in intervals of 0.5) of the running routes to search for.

2.1.2. The system will display a slider to toggle the maximum distance (from 0.5 km to 30 km in intervals of 0.5 km) of the running routes to search for.

2.1.2.1. The system will display the difficulty of the run the user is searching for as Easy, Medium, Hard or Expert.

2.1.2.1.1. The system will display "Easy" difficulty when the maximum distance ranges from 0.5 km to 2.5 km.

2.1.2.1.2. The system will display "Medium" difficulty when the maximum distance ranges from 3.0 km to 8.0 km.

2.1.2.1.3. The system will display "Hard" difficulty when the maximum distance ranges from 8.5km to 15.0 km.

2.1.2.1.4. The system will display "Expert" difficulty when the maximum distance is above 15.5km.

2.1.3. The system will display a radio button to toggle the terrain type (urban or trail) of the running routes to search for.

3. The system will display the interactive map. (refer to Functionality 8. Interactive Map)

3.1. The map will display a marker of the location chosen by the user on the map.

4. The system will display a "search" button for users to use to initiate the search for running routes.

4.1. The system will display a list of running routes that matches the input location and chosen filters.

4.1.1. The system will display the routes in order of distance from location entered

4.1.2. The system will display the name of the route, the username of the creator of the route, the average review rating of the route, the number of users that have finished the route, distance information in kilometres of the route, terrain information (urban or trail) of the route, and estimated time of completion of the route.

4.1.3. The system will display the reviews on the route.

4.1.3.1. The system will display the reviewer's username, the photo they uploaded and their comments on the route.

4.1.3.1.1. The system will generate a blank string for blank comment reviews.

4.1.3.1.2. The system will generate a default photo for the review if the reviewer did not upload a photo for the review.

4.2. The system will show an error message "Invalid location" when the user does not input a valid location to search for.

4.3. The system will show an error message "Location unavailable" when the user clicks on "Use current location" but does not allow the system to access their GPS location.

4.4. View Route

4.4.1 Description and Priority

1. Views a route in more detail

Priority: High

4.4.2 Stimulus/Response Sequences

Use Case ID:	4.1		
Use Case Name:	View Additional Route Information (Browse)		
Created By:	Javen Ong Jing Hian	Last Updated By:	Javen Ong Jing Hian
Date Created:	08/02/2024	Date Last Updated:	31/03/2024

Actor:	User (Initiating Actor), Route DB
Description:	Views additional information about a route
Preconditions:	User is logged into their account and User is in the Browse page and has already searched for routes
Postconditions:	User is shown additional details of the route they selected

Priority:	High
Frequency of Use:	Frequent
Flow of Events:	1. The User clicks on a route to view more information. 2. The System retrieves the details of the route from Route DB. 3. The System displays the leaderboard of fastest run times of the route. 4. The System displays the route and marks the start, end and stop-by points on the interactive map 5. The System displays an option to add the route to the User's list of runs, 'Add to My Runs' 6. The System displays an option to start running the route, 'Start Running'. 7. The User selects the option 'Add to My Runs' 8. The System calls Use Case 7.4 to add the route to the User's list of runs 9. The System brings the User to the My Runs page
Alternative Flows:	AL-S7: The User selects the option 'Start Running' 1. The System calls Use Case 6 for the User to review the route after they have completed the route
Exceptions:	
Includes:	
Special Requirements:	
Assumptions:	
Notes and Issues:	

Use Case ID:	4.2		
Use Case Name:	View run on interactive map		
Created By:	Javen Ong Jing Hian	Last Updated By:	Javen Ong Jing Hian
Date Created:	08/02/2024	Date Last Updated:	31/03/2024

Actor:	User (Initiating Actor), Route DB, Leaflet Map API
Description:	Displays the route chosen on the interactive map

Preconditions:	The User is logged into their Account and The User is in ‘My Runs’ page or Browse page
Postconditions:	The route chosen is displayed on the interactive map
Priority:	Medium
Frequency of Use:	Occasionally
Flow of Events:	1. The User selects the route to display on the interactive map 2. The System retrieves the details of the route from Route DB 3. The System displays the route and marks the start, end and stop-by points on the interactive map
Alternative Flows:	
Exceptions:	EX1: The User selects to view a curated route from Home page or selects a run to ‘Add to My Runs’ in Browse page 1. The System jumps to step 2 using the route selected
Includes:	
Special Requirements:	
Assumptions:	
Notes and Issues:	

Use Case ID:	4.3		
Use Case Name:	View Route		
Created By:	Javen Ong Jing Hian	Last Updated By:	Javen Ong Jing Hian
Date Created:	08/02/2024	Date Last Updated:	31/03/2024

Actor:	User (Initiating Actor), Route DB
Description:	Views a route and its details
Preconditions:	User is logged into their account and

	User selects to view a curated route from Home page or selects a run to 'Add to My Runs' in Browse page
Postconditions:	User is shown the details of the route they chose to view
Priority:	High
Frequency of Use:	Frequent
Flow of Events:	1. The User clicks on a route. 2. The System retrieves the route and its details from Route DB. 3. The System navigates to the Browse page 4. The System displays the route and its details – name, username of the creator, rating, reviews, leaderboard of fastest run times, number of users that have finished the route, distance information of the route in kilometres, terrain information (urban or trail) and estimated time of completion of the route in minutes 5. The System calls Use Case 4.2 to display the route on the interactive map 6. The System displays an option to add the route to the User's list of runs 7. The System displays an option to go back to the Browse page 8. The User selects the option 'Add to My Runs' 9. The System calls Use Case 7.4 to add the route to the User's list of runs
Alternative Flows:	AL-S8: The User selects the option 'Back' 1. The System brings the User to the Browse page and calls Use Case 3
Exceptions:	
Includes:	
Special Requirements:	
Assumptions:	
Notes and Issues:	

4.4.3 Functional Requirements

1. When user selects the route after clicking on a route in browse page. The system will show the same route details shown on the browse page in addition to the following features:
   - 1.1. The system will display the leaderboard of fastest run timings of users for the route.
     - 1.1.1. The system will display the usernames of the top 3 fastest user's run timings.
     - 1.1.2. The system will display the run timings of the top 3 fastest user's run timings.

1.1.2.1. The run timings will be displayed in hh:mm:ss format.

1.2. The system will show the whole route on the interactive map

1.3. The user has the option to add the route to MyRuns

1.4. The user has the option to start their run on that route immediately to input their review

2. When the user selects the route after clicking on a route in MyRuns Page. The system will show the same route details shown on the MyRuns page in addition to the following features:

2.1. The system will show the whole route on the interactive map (refer to Functionality 8. Interactive Map)

2.2. When the user toggles to the browse page the following will happen:

2.2.1. The system will show the same route details shown on the MyRuns page after clicking on the route in the MyRuns Page

2.2.2. The system will have the option to add the route to MyRuns

4.5. Create Route

4.5.1 Description and Priority

1. User creates own custom route

Priority: High

4.5.2 Stimulus/Response Sequences

Use Case ID:	5		
Use Case Name:	Make custom route		
Created By:	Javen Ong Jing Hian	Last Updated By:	Wong Zi Lun
Date Created:	08/02/2024	Date Last Updated:	31/03/2024

Actor:	User (Initiating Actor), Route DB, Leaflet Map API
Description:	User creates a custom route
Preconditions:	User is logged into their account
Postconditions:	User’s created route is stored in Route DB and added to Runs DB
Priority:	High
Frequency of Use:	Frequent
Flow of Events:	1. The User selects the option ‘Custom’ on the system sidebar user interface 2. The System displays a text field for the User to enter a starting address.

	3. The System displays a text field for the User to enter a destination address. 4. The System displays an interactive map from Leaflet Map API. 5. The System displays an option to search for a custom route, ‘Search’ 6. The User enters a starting address and destination address. 7. The User selects ‘Search’ 8. The System verifies that valid starting and destination addresses have been input. 9. The System displays the route on the interactive map. 10. The System displays a text field for the User to enter additional stop-by points for the custom route. 11. The System displays an option to add additional stop-by points for the custom route, ‘Add Point’ 12. The System displays a text field for the User to enter a route name for the custom route. 13. The System displays an option for the User to choose the terrain type of the custom route 14. The System displays an option to create the custom route, ‘Add New Route’ 15. The User enters additional stop-by point addresses. 16. The User selects ‘Add Point’ 17. The System verifies that a valid stop-by point address has been input 18. The System adds the stop-by point into the custom route 19. The User enters the route name and terrain type of the custom route 20. The User selects ‘Add New Route’ 21. The System verifies that a valid route name has been input. 22. The System generates the running route based on the User’s inputs using the map api’s route creation algorithm 23. The System stores the route to Route DB 24. The System stores the route to Runs DB 25. The System brings the User to the My Runs page
Alternative Flows:	AF-S9: The User did not input valid starting and destination addresses. 1. The System displays the message “invalid address. Please use location provided in dropbox” 2. The System returns to step 6 AF-S17: The User does not input any additional stop-by point addresses 1. The System jumps to step 20 AF-S19: The User does not input a valid additional stop-by point

	addresses 1. The System does not add the stop-by point into the custom route AF-S23: The User did not input valid name for the route 1. The System displays the message “Please enter a name for the route” 2. The System returns to step 16 AF-S23: The User has already created another route with the same name 1. The System displays the message “You already have another route with the same name! Please enter another name for the new route” 3. The System returns to step 16
Exceptions:	
Includes:	
Special Requirements:	
Assumptions:	
Notes and Issues:	

4.5.3 Functional Requirements

1. The system will display the interactive map. (refer to Functionality 8. Interactive Map)
2. Users will be able to create a custom route.
    - 2.1. The system will display a text field for users to enter a starting address.
        - 2.1.1. The system will suggest locations to autocomplete the user’s current text input in the text field
        - 2.1.2. The user will be able to select a suggested location to autocomplete the text in the text field.
    - 2.2. The system will display a text field for users to enter a destination address.
        - 2.2.1. The system will suggest locations to autocomplete the user’s current text input in the text field
        - 2.2.2. The user will be able to select a suggested location to autocomplete the text in the text field.
    - 2.3. The system will display a button for users to search for routes with the provided.
        - 2.3.1. The system will display the route on the interactive map.
            - 2.3.1.1. The system will display the start and end points of the route with markers.
            - 2.3.1.2. The system will highlight the body of the route.
            - 2.3.1.3. The user will be able to modify the address of the start and end points by dragging the markers on the interactive map.
        - 2.3.2. The user will be able to select additional stop-by points for the route.

2.3.2.1. The system will display a text field for users to enter a stop-by point address.

2.3.2.2. The user will be able to create additional stop-by points by clicking on the interactive map.

2.3.2.3. The system will display additional stop-by points on the interactive map with markers.

2.3.2.4. The user will be able to modify the address of additional stop-by points by dragging the markers on the interactive map.

2.3.3. The system will display a text field for users to enter a name for their custom route.

2.3.3.1. The system will show an error message “Please enter a name for the route” when the user does not input a valid name for the route.

2.3.4. The system will display a radio button for the user to select the terrain type (urban or trail) of the custom route.

2.3.5. The system will display a button “Add New Route” to create the custom route.

2.3.5.1. The system will generate a running route based on the user’s specifications in 2.1, 2.2, 2.3.2, 2.3.3 and 2.3.4 from view route system feature.

2.3.5.2. The system will store the route into the database of routes.

2.3.5.3. The system will add the route to the user’s ‘My Runs’ page.

2.3.5.4. The system will redirect the user to their ‘My Runs’ page.

4.6. Route Review

4.6.1 Description and Priority

1. Give a review for a route after completing it

Priority: High

4.6.2 Stimulus/Response Sequences

Use Case ID:	6		
Use Case Name:	Give Route Review		
Created By:	Javen Ong Jing Hian	Last Updated By:	Javen Ong Jing Hian
Date Created:	08/02/2024	Date Last Updated:	31/03/2024

Actor:	User (Initiating Actor), Review DB
Description:	Give a review for a route after completing it
Preconditions:	User is logged into their account

	and User has completed a route and selected to give a review
Postconditions:	User’s review is saved to the route
Priority:	High
Frequency of Use:	Frequent
Flow of Events:	1. The User chooses to review route 2. The System displays the Review user interface 3. The System displays an input field for users to input their rating for a route. 4. The System displays an input field for users to input their time taken for a route. 5. The System displays a text field for users to input their comments for a route. 6. The System displays a file upload field for users to upload an image for their route. 7. The System displays an option for users to escape from the review page. 8. The System displays an option for users to submit their review, ‘Submit’. 9. The User inputs their rating, time taken, comments and image for the route 10. The User selects the option ‘Submit’ 11. The System verifies that a valid rating has been input 12. The System verifies that a valid time taken has been input 13. The System stores the review into Review DB
Alternative Flows:	AL-S12: The rating the User entered is invalid 1. The System displays the message “Invalid input. Please rate the route” 2. The System returns to step 9 AL-S13: The time taken the User entered is invalid 1. The System displays the message “Invalid input. Please enter a valid duration” 2. The System returns to step 9
Exceptions:	EX1: The User closes the review form 1. The System returns the User to the previous page
Includes:	

Special Requirements:	
Assumptions:	
Notes and Issues:	

4.6.3 Functional Requirements

1. The system will display an input field for users to input their rating for a route.
    1.1. Users must rate a route on a scale from 1 to 5 in intervals of 1.
    1.2. The system will show an error message “Invalid input. Please rate the route” when the user does not input a valid rating for the route.
2. The system will display an input field for users to input their time taken for a route.
    2.1. Users must provide the duration of their run for the route in hours, minutes and seconds.
    2.2. The system will show an error message “Invalid input. Please enter a valid duration” when the user does not input a valid duration (0h 0m 0s) for the route.
3. The system will display a text field for users to input their comments for a route.
    3.1. Users can input a maximum of 1000 characters for their comment.
4. The system will display a file upload field for users to upload an image for their route.
    4.1. Users can upload a photo relating to the route by clicking on “Upload an image”.
    4.2. The file uploaded must be an image file.
5. The system will display a button for users to escape from the review page.
    5.1. Users can cancel the review before submitting it.
6. The system will display a button for users to submit their review.
    6.1. The system will store the review into the database of reviews.

4.7. MyRuns

4.7.1 Description and Priority

1. Adds route from browse/custom route page to user’s MyRun as an unreviewed route
2. System displays list of user’s My Runs according to filter

Priority: High

4.7.2 Stimulus/Response Sequences

Use Case ID:	7.1		
Use Case Name:	Go to My Runs page		
Created By:	Javen Ong Jing Hian	Last Updated By:	Javen Ong Jing Hian
Date Created:	08/02/2024	Date Last Updated:	31/03/2024

Actor:	User (Initiating Actor), Runs DB, Route DB
Description:	System displays User’s My Runs page

Preconditions:	The User is logged into their Account
Postconditions:	User’s My Runs page is displayed
Priority:	Medium
Frequency of Use:	Occasionally
Flow of Events:	1. The User clicks on the option ‘My Runs’ on the system sidebar user interface 2. The System retrieves the User’s runs and run details from Runs DB. 3. The System displays the User’s list of runs with their details – name, username of the creator, rating, number of users that have finished the route, distance information in kilometres, terrain information (urban or trail) and estimated time of completion in minutes. 4. The System displays an option to mark the routes in the User’s ‘My Runs’ as favourite. 5. The System displays a checkmark on each route in the User’s ‘My Runs’ to indicate whether the route has been favourited (grey = unfavourited, blue = favourited) 6. The System retrieves the reviews of the User’s list of runs from Route DB. 7. The System displays the reviews of the routes the user has already reviewed with their details – photo uploaded, reviewer’s username, rating and comments. 8. The System displays the option to review the routes in the User’s ‘My Runs’ that are unreviewed by the User 9. The System displays an interactive map from Leaflet Map API. 10. The System displays an option to filter the type of routes (All Routes, Reviewed Routes, Unreviewed Routes, Favourited Routes and Created Routes) to be displayed.
Alternative Flows:	
Exceptions:	
Includes:	
Special Requirements:	
Assumptions:	
Notes and Issues:	

Use Case ID:	7.2		
Use Case Name:	Favourite a run		
Created By:	Javen Ong Jing Hian	Last Updated By:	Javen Ong Jing Hian
Date Created:	08/02/2024	Date Last Updated:	31/03/2024

Actor:	User (Initiating Actor), Runs DB
Description:	Mark a run as favourited
Preconditions:	The User is logged into their Account and The User is in ‘My Runs’ page
Postconditions:	User’s chosen run is favourited
Priority:	Medium
Frequency of Use:	Occasionally
Flow of Events:	1. The User selects the option to favourite a run. 2. The System saves the route as favourited in Runs DB 3. The System refreshes the My Runs page by calling Use Case 7.1
Alternative Flows:	
Exceptions:	
Includes:	
Special Requirements:	
Assumptions:	
Notes and Issues:	

Use Case ID:	7.3		
Use Case Name:	Filtering runs displayed in My Runs		
Created By:	Javen Ong Jing Hian	Last Updated By:	Javen Ong Jing Hian

Date Created:	08/02/2024	Date Last Updated:	31/03/2024

Actor:	User (Initiating Actor)
Description:	Filters the list of runs displayed in My Runs
Preconditions:	The User is logged into their Account and The User is in ‘My Runs’ page
Postconditions:	The filtered list of runs is displayed
Priority:	Medium
Frequency of Use:	Occasionally
Flow of Events:	1. The User selects the option to filter the runs displayed 2. The User selects the filter to use (All Routes, Reviewed Routes, Unreviewed Routes, Favourited Routes or Created Routes) 3. The System removes the runs on display that do not match the filter applied
Alternative Flows:	
Exceptions:	
Includes:	
Special Requirements:	
Assumptions:	
Notes and Issues:	

Use Case ID:	7.4		
Use Case Name:	Add to My Runs		
Created By:	Javen Ong Jing Hian	Last Updated By:	Javen Ong Jing Hian
Date Created:	08/02/2024	Date Last Updated:	31/03/2024

Actor:	User (Initiating Actor), Runs DB
Description:	Adds a route to the User's list of runs
Preconditions:	The User is logged into their Account and The User has selected a route to save
Postconditions:	The selected route is saved to the User's list of runs
Priority:	High
Frequency of Use:	Frequent
Flow of Events:	1. The User selects a route to save. 2. The System saves the route to the User's list of runs in Runs DB 3. The System calls Use Case 7.1 and opens the My Runs page
Alternative Flows:	
Exceptions:	
Includes:	
Special Requirements:	
Assumptions:	
Notes and Issues:	

4.7.3 Functional Requirements

1. The system will display the interactive map.
2. The system shall display a dropbox field with choices of filters to filter the list of routes to be displayed.
   2.1. The "All Routes" filter is the default filter and will display the list of all routes.
   2.2. The "Reviewed Routes" filter will display only the list of routes that have been reviewed by the user.
   2.3. The "Unreviewed Routes" filter will display only the list of routes that are unreviewed by the user.
   2.4. The "Favourited Routes" filter will display only the list of routes that are favourited by the user.

2.5. The “Created Routes” filter will display only the list of routes that are created by the user.

3. The system shall display the list of routes the user has added to MyRuns.
    - 3.1. The system will display the details of each route.
        - 3.1.1. The system will display the image attached to the route.
        - 3.1.2. The system will display the name of the route.
        - 3.1.3. The system will display the username of the creator of the route.
        - 3.1.4. The system will display the number of users that have finished the route.
        - 3.1.5. The system will display the distance information of the route in kilometres.
        - 3.1.6. The system will display the terrain information (urban or trail) of the route.
        - 3.1.7. The system will display the estimated time of completion of the route.
    - 3.2. The system will display additional information for the routes the user has already reviewed.
        - 3.2.1. The system will display the reviews of the route.
            - 3.2.1.1. The system will display the reviewer’s username.
            - 3.2.1.2. The system will display the photo uploaded for the review.
                - 3.2.1.2.1. The system will generate a default photo for the review if the reviewer did not upload a photo for the review.
            - 3.2.1.3. The system will display the rating for the route.
            - 3.2.1.4. The system will display the comments on the route.
                - 3.2.1.4.1. The system will generate a blank string for blank comment reviews.
        - 3.2.2. The system will display the leaderboard of fastest run timings of users for the route.
            - 3.2.2.1. The system will display the usernames of the top 3 fastest user’s run timings.
            - 3.2.2.2. The system will display the run timings of the top 3 fastest user’s run timings.
                - 3.2.2.2.1. The run timings will be displayed in hh:mm:ss format.
    - 3.3. The system will display a button to review the routes the user has not reviewed.
        - 3.3.1. The system will call the review route functionality when the button is clicked. (refer to Functionality 6. Review Route)
4. The system will display a plus shaped button attached to each route to favourite the route.
    - 4.1. Users can mark a route as favourited by clicking the plus button of the route.
    - 4.2. The system will mark the route as favourited.
5. The system will display a check mark symbol attached to each route to indicate whether the route has been favourited.
    - 5.1. The colour of the check mark will be grey by default and indicates the route has not been favourited.
    - 5.2. The colour of the check mark will be blue if the route has been favourited.
6. The system will allow the user to add routes to MyRuns when they view routes through various means as mentioned in the view route system feature.

5. Other Nonfunctional Requirements

5.1. Performance Requirements

5.1.1.The system shall have at most 1% down time on weekly average

Rationale: Prevents as many as possible instances where users are not able to access the system, which would lead to lower user satisfaction.

5.1.2.The system shall be able to support at least 10 active users concurrently

Rationale: Prevents as many as possible instances where users are not able to access the system, which would lead to lower user satisfaction.

5.1.3.The system shall return at least 5 recommended routes within 10 seconds of a user's search

Rationale: Reducing loading times on the user's side to improve user satisfaction when using the system.

5.2. Reliability Requirements

5.2.1.The system shall deliver identical results for identical requests, with a variance of no more than 0.01% between executions, under the same operating conditions.

Rationale: The requests from users must be mostly identical so that the user can reproduce and search for the same results again with the same search, helping to instil confidence of the system to the users. A vastly different result from identical requests would result in users losing confidence and lower user satisfaction

5.2.2.Distances indicated in the system shall be accurate up to 0.5km

Rationale: The distances indicated in the system must be correct to instil confidence to the user so that users will come back to use it again or recommend it to their peers. A very inaccurate indication of distance would result in users losing confidence and lower user satisfaction.

5.2.3.Forget password email must be sent out within 1 minute of pressing the 'Send Email' button

Rationale: The email must be sent out very quickly so that users will be able to quickly recover their account, increasing user satisfaction of the system.

5.3. Usability Requirements

5.3.1.80% of users must be able to perform any main function of the system within 5 minutes of use.

Rationale: Being able to use the main functions of the application immediately is what users expect.

5.3.2.The system must ensure that the user interface automatically adjusts to the screen size and resolution of both mobile devices (screen widths from 320px to 640px) and desktop monitors (screen widths from 1024px to 1920px)

Rationale: With this application being about running routes, making it portable and easy to use across multiple devices is what should be expected from users, with the application catering to a majority of popular devices.

5.3.3.The system must have a consistent user interface and design language throughout

Rationale: Having a consistent user interface style and design language would give users a sense of uniformity when using the different components of the system, allowing for better user satisfaction. A consistent design language would ensure that components seen on screen which look similar perform similar actions, preventing situations where the software does something that the user does not expect.

5.4. Security Requirements

5.4.1.The system must implement secure session management practices, including the generation of unique session identifiers.

Rationale: Unique session identifiers help prevent the web application from being hacked and exploited by hackers who want to take user data relating to their routes.

5.4.2.All input fields must be validated.

Rationale: Prevents unnecessary blank input fields which may cause errors in the system.

5.4.3.Password stored must be encrypted

Rationale: Keeps passwords of users safe from being easily extracted and used by hackers.

5.4.4.Password must be masked with '*' in the input field

Rationale: Keeps users safe from people peeking over to see their password while the user is logging in.

5.5. Scalability Requirements

5.5.1.The system should be able to handle an increase in users and data without performance degradation

Rationale: Having the ability to scale up the application for more users and data without any performance slow downs helps maintain user satisfaction with the application. Moreover, with the application having the ability to handle more users and data in the future, we can have a robust and massive hub of satisfied users using the application to create, browse and review their runs.

5.6.Maintainability Requirements

5.6.1.The system shall allow developers to implement and deploy updates to any module without affecting the operation of other modules

Rationale: Being able to update any module without disruption of any other functionality helps keep the application up for users to use and can easily maintain user satisfaction and retain users. Moreover, with no disruption to operations of other modules, developers of other modules can continue to easily work on their intended updates.

5.6.2.All system updates shall be deployable with no more than 2 hours of downtime per month

Rationale: Having very less downtime for any updated module helps the developers bring out their intended updates quickly and also brings about quicker bug testing of major updates. Lower downtimes also help to maintain user satisfaction.

6. Testing

6.1. Blackbox Testing

6.1.1 Create Account

Test Case No.	Inputs	Expected Output	Test Output	Result
1	Email : <Empty> Password : <Empty>	Fail to sign up due to empty field	Fail to sign up	Passed
2	Email : <Empty> Password : 123456	Fail to sign up due to empty field	Fail to sign up	Passed
3	Email : aaron@gmail.com Password : <Empty>	Fail to sign up due to empty field	Fail to sign up	Passed
4	Email : aaro@gmail.com Password : helloworld	Fail to sign up due to existing account	Fail to sign up	Passed
5	Email : aaron Password : helloworld	Fail to sign up due to invalid email	Fail to sign up	Passed
6	Email : aaron@gmail.com Password : helloworld	Successfully signed up	Successfully signed up	Passed
7	Email : jason@gmail.com Password : helloworld	Successfully signed up	Successfully signed up	Passed

6.1.2 Log In Account

Test Case No.	Inputs	Expected Output	Test Output	Result
1	Email : <Empty> Password : <Empty>	Fail to log in due to empty field	Fail to log in	Passed
2	Email : <Empty> Password : 123456	Fail to log in due to empty field	Fail to log in	Passed
3	Email : aaron@gmail.com Password : <Empty>	Fail to log in due to empty field	Fail to log in	Passed

Test Case No.	Inputs	Expected Output	Test Output	Result
4	Email : test Password : 123456	Fail to log in due to invalid email	Fail to log in	Passed
5	Email : aaron@gmail.com Password : 123456	Fail to log in due to wrong password	Fail to log in	Passed
6	Email : charlie@gmail.com Password : helloworld	Fail to log in due to wrong email	Fail to log in	Passed
7	Email : test@gmail.com Password : 123456	Successfully logged in	Successfully logged in	Passed

6.1.3 Log Out Account

Test Case No.	Inputs	Expected Output	Test Output	Result
1	After logging in, click on “Log out” button	Successful log out	Successful log out	Passed

6.1.4 Remain Logged In After Refresh

Test Case No.	Inputs	Expected Output	Test Output	Result
1	After logging in, refresh the page by clicking the refresh button.	User is still logged in	User is still logged in	Passed

6.1.5 Home Page Functionality

Test Case No.	Inputs	Expected Output	Test Output	Result
1	Home Page	Display statistics/ recommendation of routes	Display statistics/ recommendation of routes	Passed
2	Click on the recommended routes	Brings user to browse and prompt display route information	Brings user to browse and prompt display route information	Passed

3	Home Page Welcome Back, header should be the <email> without @gmail.com	Welcome Back, AARON	Welcome Back, AARON	Passed

6.1.6 Browse Page Functionality

Test Case No.	Inputs	Expected Output	Test Output	Result
1	Browse Page	Input field of location with recommended location and filters	Input field of location with recommended location and filters	Passed
2	Click on the search bar, and enter an address or select recommendation e.g. Nanyang Drive, NTU North Spine Plaza, Singapore	Display the list containing the search query	Display the list containing the search query	Passed
3	Combinations of Ratings, Distance, Terrain and Elevation	Output list should be filtered accordingly	Output list should be filtered accordingly	Passed
4	Scroll down the list	The list shows the best match first then subsequently less matching routes	The list shows the best match first then subsequently less matching routes	Passed
5	Select one of the route in the list	Route information displayed with map and Leaderboard	Route information displayed with map and Leaderboard	Passed
6	Select add to My Runs on desired route	Route added to My Runs page, and user directed to My Runs page	Route added to My Runs page, and user directed to My Runs page	Passed
7	Enter a location: <Empty> Rating: 0 Distance: 5 Elevation: 5 Terrain: Urban	Alert: Request to input location	Alert: Request to input location	Passed

8	Enter a location : 2 Chome-29-1 Dogenzaka, Shibuya City, Tokyo 150-0043, Japan Rating :0 Distance :5 Elevation :5 Terrain :Urban	Prompt user for valid location (in Singapore)	Prompt user for valid location (in Singapore)	Passed

6.1.7 Custom Page Functionality

Test Case No.	Inputs	Expected Output	Test Output	Result
1	Custom Page	Request 2 location fields to make a route	Request 2 location fields to make a route	Passed
2	Starting Address: <Empty> Destination Address: <Empty>	Request to input location (location cannot be empty)	Request to input location (location cannot be empty)	Passed
3	Starting Address: Nanyang Drive, NTU North Spine Plaza, Singapore Destination Address: <Empty>	Request to input location (location cannot be empty)	Request to input location (location cannot be empty)	Passed
4	Starting Address: <Empty> Destination Address: Nanyang Drive, NTU North Spine Plaza, Singapore	Request to input location (location cannot be empty)	Request to input location (location cannot be empty)	Passed
5	Starting Address: Nanyang Drive, NTU North Spine Plaza, Singapore Destination Address: Nanyang Avenue, Hall 11 - Block 54, Singapore	Display route on map with further prompts to add point, name route, set terrain and add route to My runs	Display route on map with further prompts to add point, name route, set terrain and add route to My runs	Passed
6	User change input Starting/Destination Address after pressing search	Display route on map will be updated with new location	Display route on map will be updated with new location	Passed

7	Input Stop-by Point address for user to add 3rd point	Display route on map will be updated with new location now with 3 markers	Display route on map will be updated with new location now with 3 markers	Passed
8	User drag can drag all markers to customs location	Route display will be updated with new drawn route on marker	Route display will be updated with new drawn route on marker	Passed
9	Starting Address: Nanyang Drive, NTU North Spine Plaza, Singapore Destination Address: Nanyang Avenue, Hall 11 - Block 54, Singapore Stop-by Point Address :<Empty> Route Name :<Empty> Terrain :Urban	Prompt user to enter Route name (cannot be empty)	Prompt user to enter Route name (cannot be empty)	Passed
10	Starting Address: Nanyang Drive, NTU North Spine Plaza, Singapore Destination Address: Nanyang Avenue, Hall 11 - Block 54, Singapore Stop-by Point Address : Nanyang Drive, Yunnan Garden, Singapore Route Name :<Empty> Terrain :Urban	Prompt user to enter Route name (cannot be empty)	Prompt user to enter Route name (cannot be empty)	Passed
11	Starting Address: Nanyang Drive, NTU North Spine Plaza, Singapore Destination Address: Nanyang Avenue, Hall 11 - Block 54, Singapore Stop-by Point Address : Nanyang Drive, Yunnan Garden, Singapore Route Name :NTU Route Terrain :Urban	NTU Route will be added to My Runs page, Map will display 3 points	NTU Route will be added to My Runs page, Map will display 3 points	Passed
12	Starting Address:	NTU Route will be added to My	NTU Route will be added to My	Passed

	Nanyang Drive, NTU North Spine Plaza, Singapore Destination Address: Nanyang Avenue, Hall 11 - Block 54, Singapore Stop-by Point Address :<Empty> Route Name :NTU Route Terrain :Urban	Runs page, Map will display 2 points	Runs page, Map will display 2 points	
13	User tries to name another route with existing name	Error message telling user to input another route name	Error message telling user to input another route name	Passed

6.1.8 My Runs Page Functionality

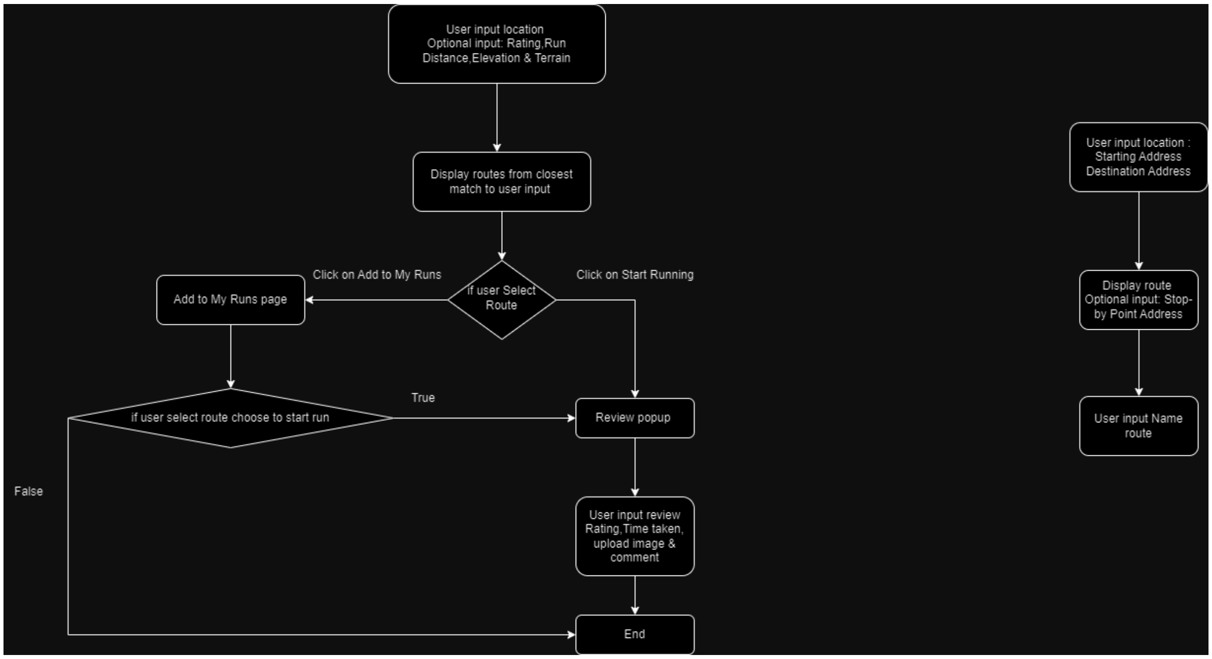
Test Case No.	Inputs	Expected Result	Test Result	Result
1	My Runs	Load user history reviewed and to be reviewed routes, if route is reviewed show user rating and comments, if not show button to review route	Load user history reviewed and to be reviewed routes, if route is reviewed show user rating and comments, if not show button to review route	Passed
2	Click on reviewed route	Display route on map, and leaderboard	Display route on map, and leaderboard	Passed
3	Click on to be reviewed route	Display route on map	Display route on map	Passed
4	Clicking review route	Prompt user input rating, comment, duration, and upload image	Prompt user input rating, comment, duration, and upload image	Passed
5	How was your experience: 0 Time Taken:01:00:00 Comment :<Empty> Image: <uploaded>	Prompt user input 1 to 5 stars	Prompt user input 1 to 5 stars	Passed

6	How was your experience: 1 Time Taken: 02:00:00 Comment :<Empty> Image: <uploaded>	Route will be reviewed and page reloads to show route with review and image	Route will be reviewed and page reloads to show route with review and image	Passed
7	How was your experience: 5 Time Taken:00:11:00 Comment : Great route Image: <uploaded>	Route will be reviewed and page reloads to show route with review and image	Route will be reviewed and page reloads to show route with review and image	Passed
8	Select filter option and scroll down the list (Reviewed, Unreviewed, Favourited, Created)	The list shows all routes based on the filtered option	The list shows all routes based on the filtered option	Passed

6.1.9 Others Functionality

Test Case No.	Inputs	Expected Result	Test Result	Result
1	Using Any Web Browser	Success	Success	Passed
2	Sidebar can be use to refresh selected page	Success	Success	Passed
3	Sidebar navigation to be linked and have same format throughout website	Success	Success	Passed

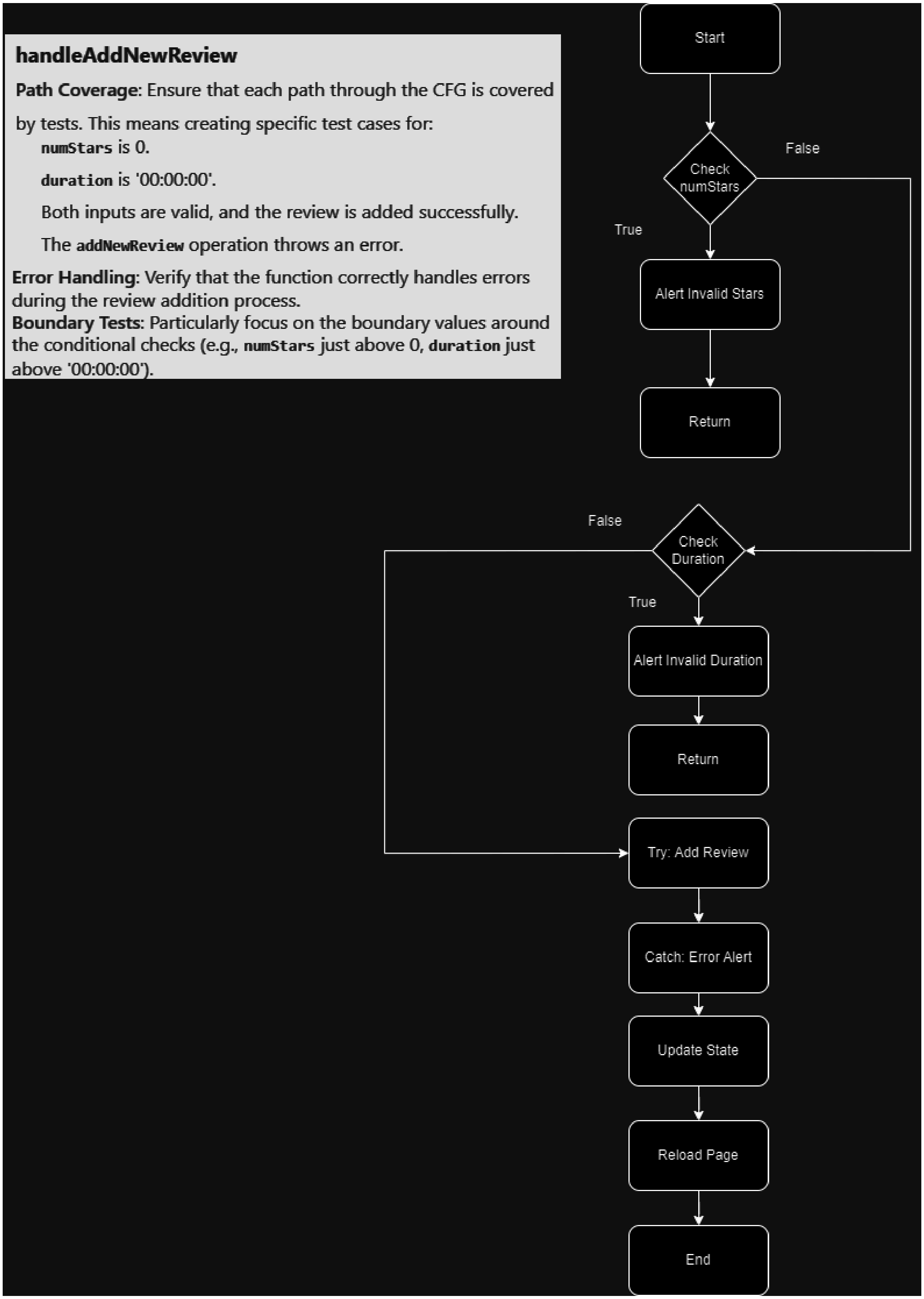
6.1.10 State based testing


User input location
Optional input: Rating,Run Distance,Elevation & Terrain
Display routes from closest match to user input
Click on Add to My Runs
if user Select Route
Click on Start Running
Add to My Runs page
True
if user select route choose to start run
Review popup
False
User input review Rating,Time taken, upload image & comment
End
User input location : Starting Address Destination Address
Display route Optional input: Stop-by Point Address
User input Name route

This shows the control flow for browse and custom routes. This is used to test if the logic implemented is as intended.

6.2 Whitebox Testing

6.2.1 Control Flow Graph (CFG)

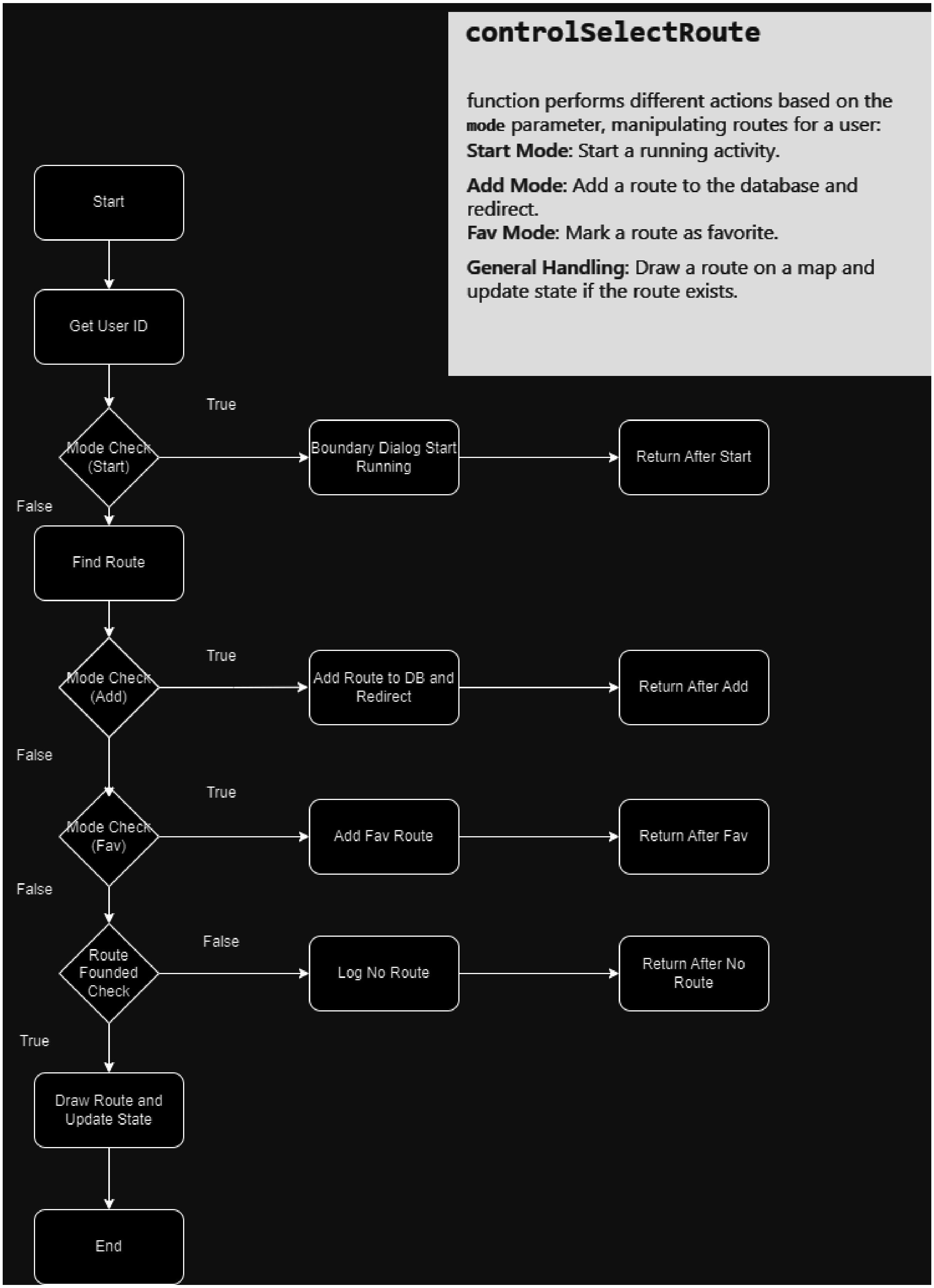


handleAddNewReview

Path Coverage: Ensure that each path through the CFG is covered by tests. This means creating specific test cases for:

- `numStars` is 0.
- `duration` is '00:00:00'.
- Both inputs are valid, and the review is added successfully.
- The `addNewReview` operation throws an error.

Error Handling: Verify that the function correctly handles errors during the review addition process.
Boundary Tests: Particularly focus on the boundary values around the conditional checks (e.g., `numStars` just above 0, `duration` just above '00:00:00').



controlSelectRoute

function performs different actions based on the `mode` parameter, manipulating routes for a user:

Start Mode: Start a running activity.

Add Mode: Add a route to the database and redirect.

Fav Mode: Mark a route as favorite.

General Handling: Draw a route on a map and update state if the route exists.

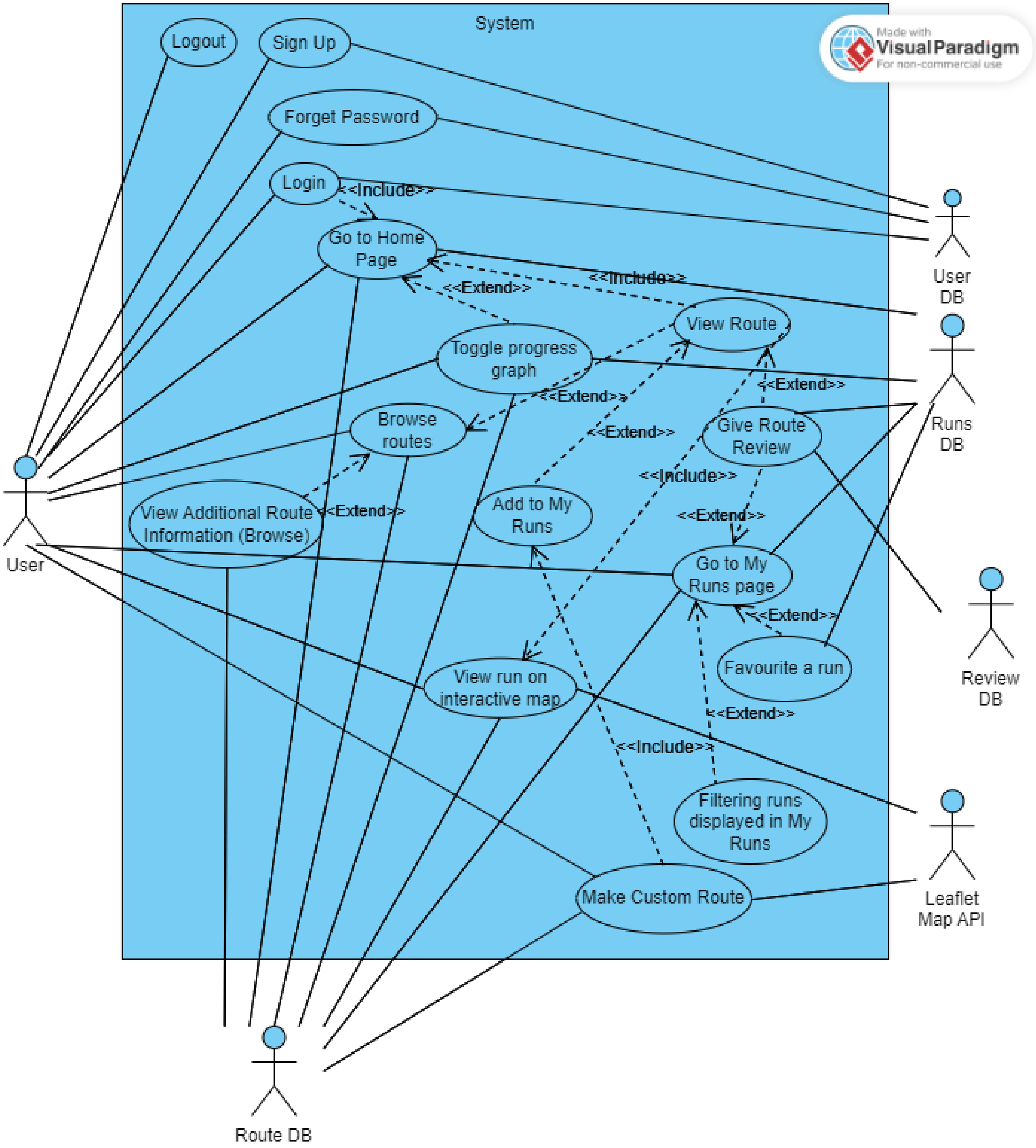
Appendix A: Glossary

Term	Definition
User	Individuals who have created an account
System	Running application
Route	Pedestrian path that is suitable for running
Creator	User who creates a route
Terrain	Type of terrain (e.g., urban, trail)
Location	Physical geographical location
Distance	Total distance of the route in kilometres
Review	User feedback of a route, consisting of rating, comment, time taken (for a run), image (of the route)
Rating	Score from 1 to 5 out of 5, representing how much a person likes a particular route
Comment	Text input representing a person's opinion of a route
Favourite	Marking a route as favourite to allow all favourited routes to be access together
Route Information	Terrain, Distance, Rating, Comments, Estimated completion time, that is specific to a route
Statistics	User run information collated into a summary statistic and graph
Hash	Unique digital fingerprint of a piece of data (Hashed password for security)
User data	Saved user information (comments, previous routes)
Name Location	A specific place that can be identified and searched through an API, such as the Google Places API, based on user input.
Difficulty	Base on total distance of the route in kilometres

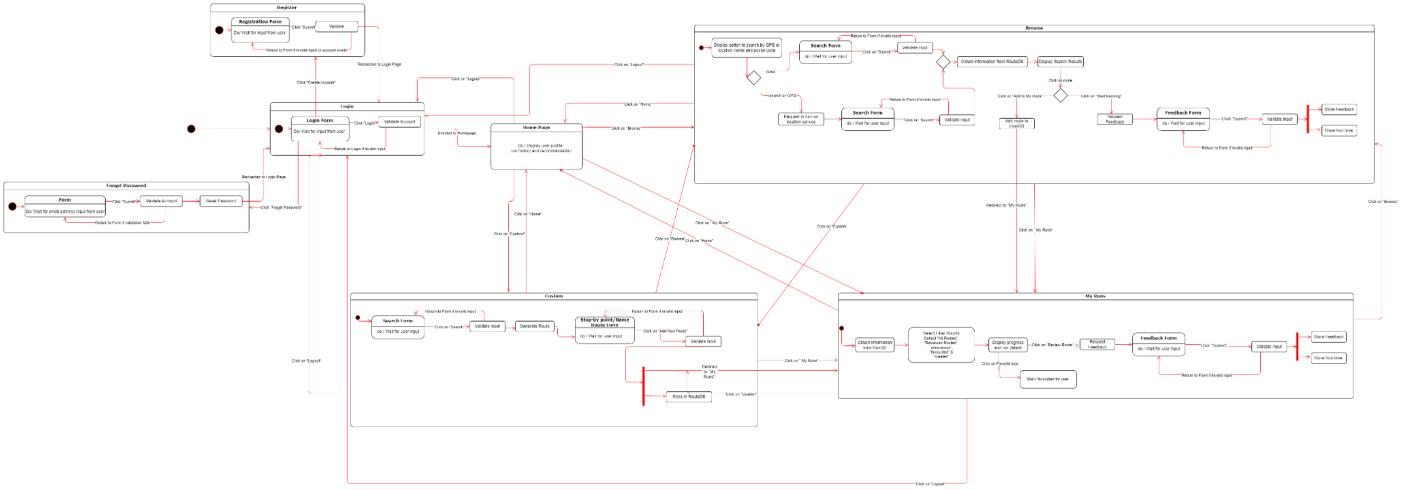
Appendix B: Analysis Models

This segment will consist of all analysis models. Clearer view of the picture are attached in the folder:

- Use Case Diagram

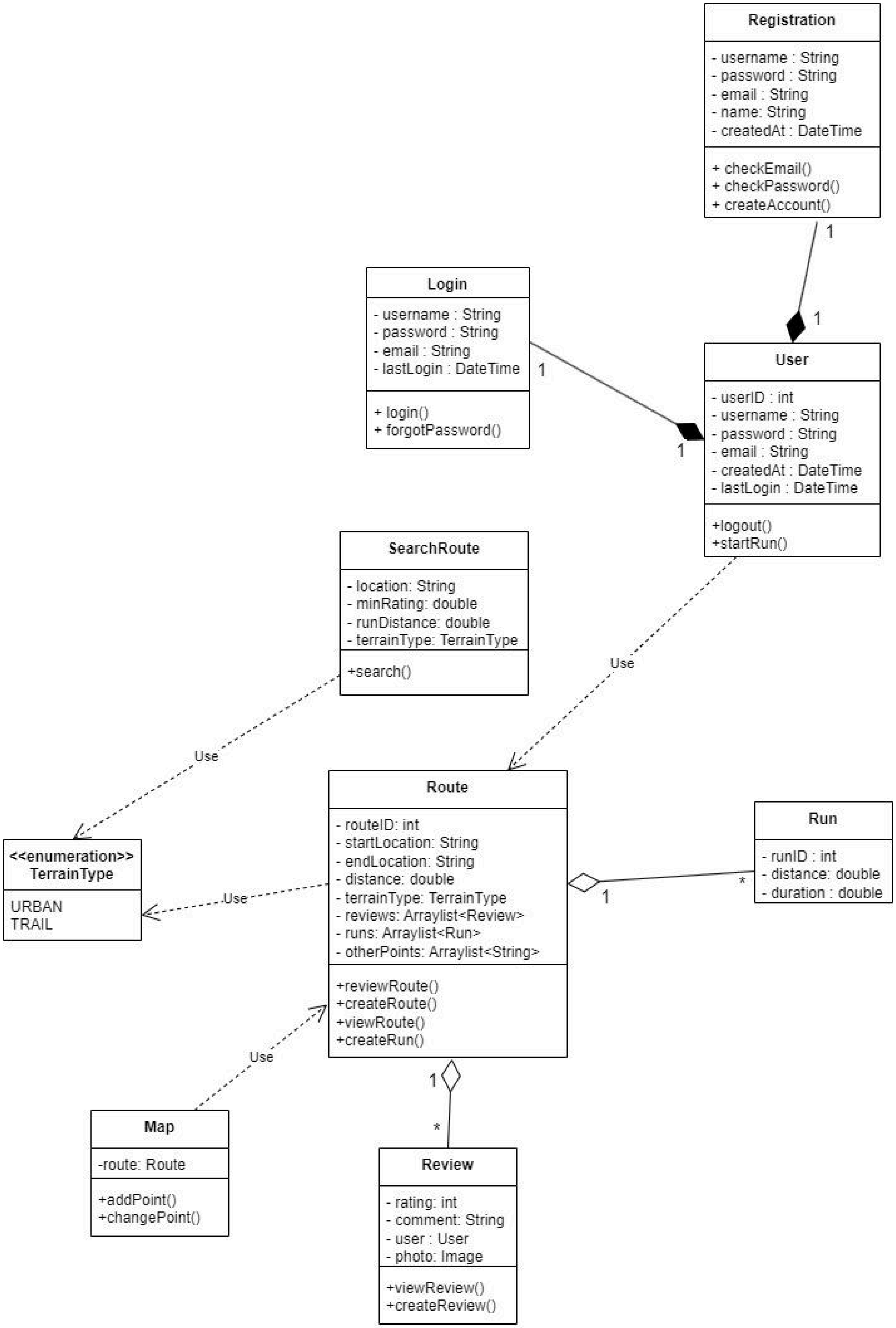


- Dialog Map (State Machine Diagram)

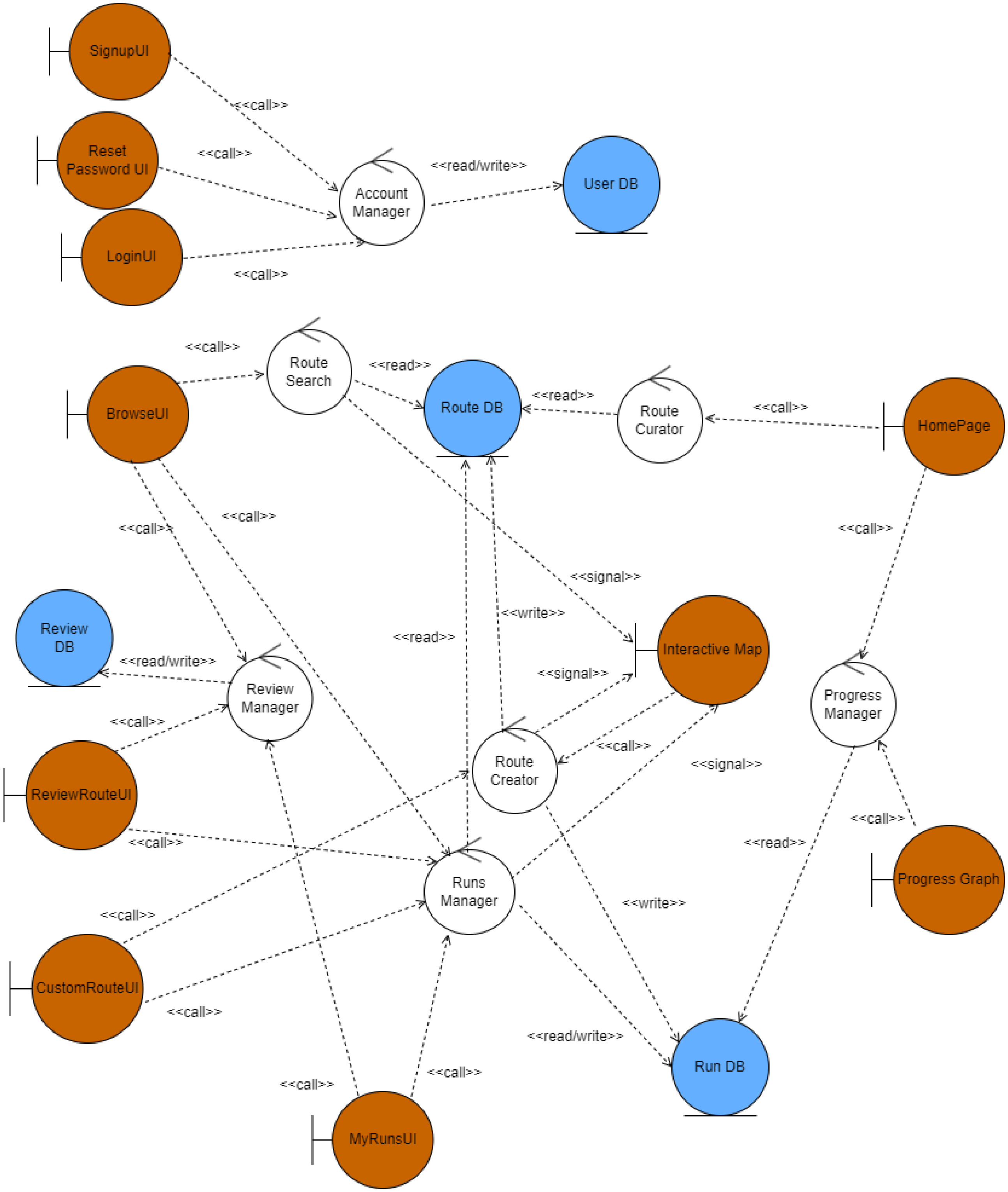


A higher res version is in the lab 5 repository

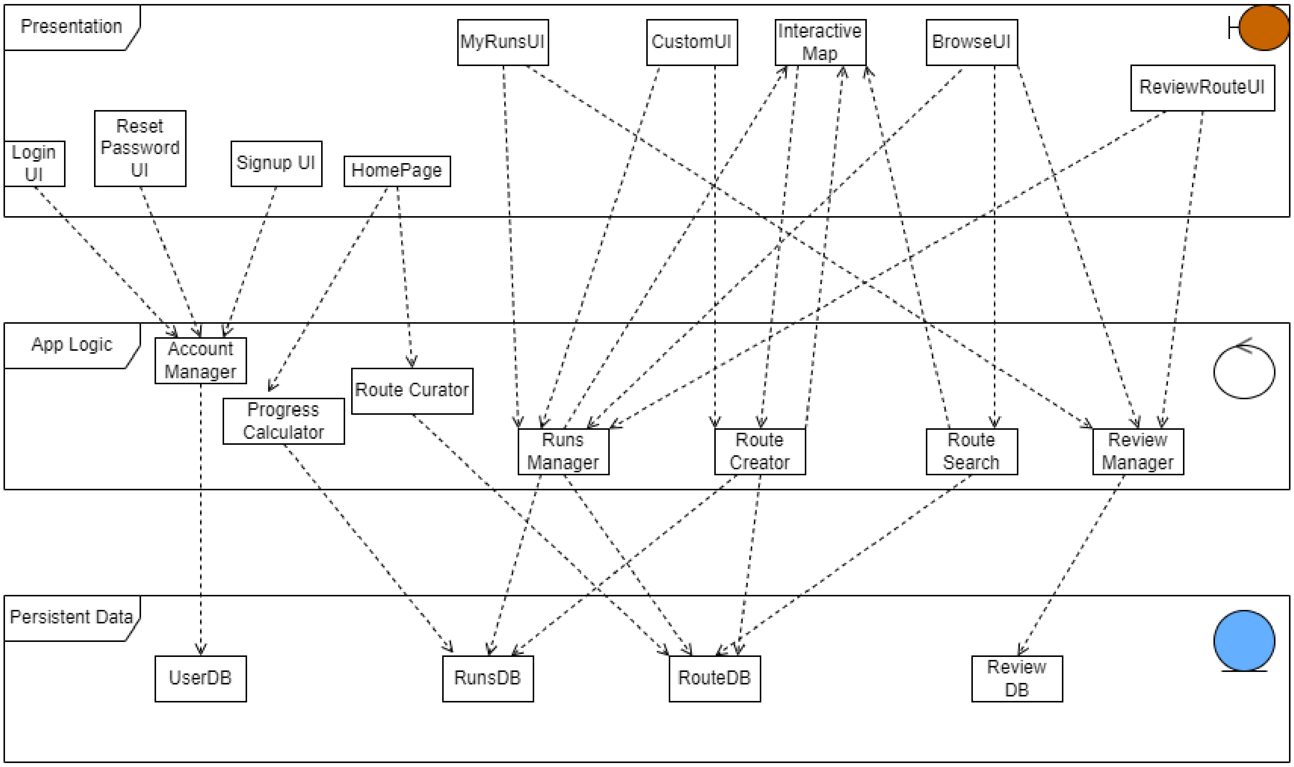
- UML Class Diagram



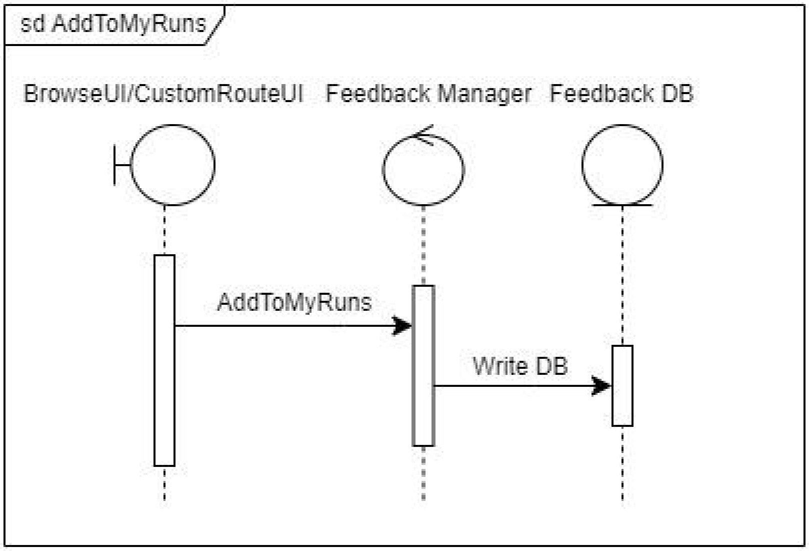
- Key Boundary Classes and Control Classes



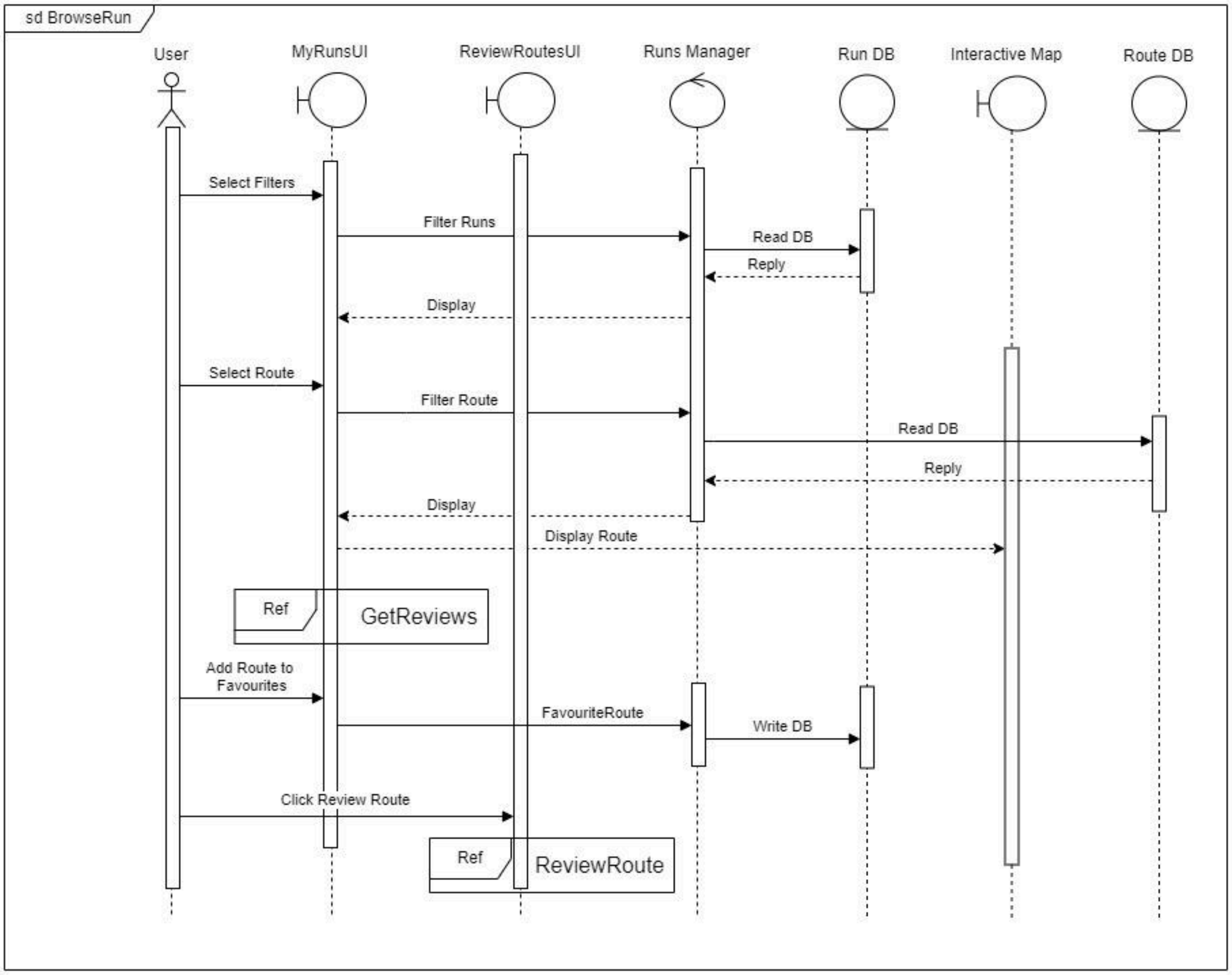
- Software Architecture Model



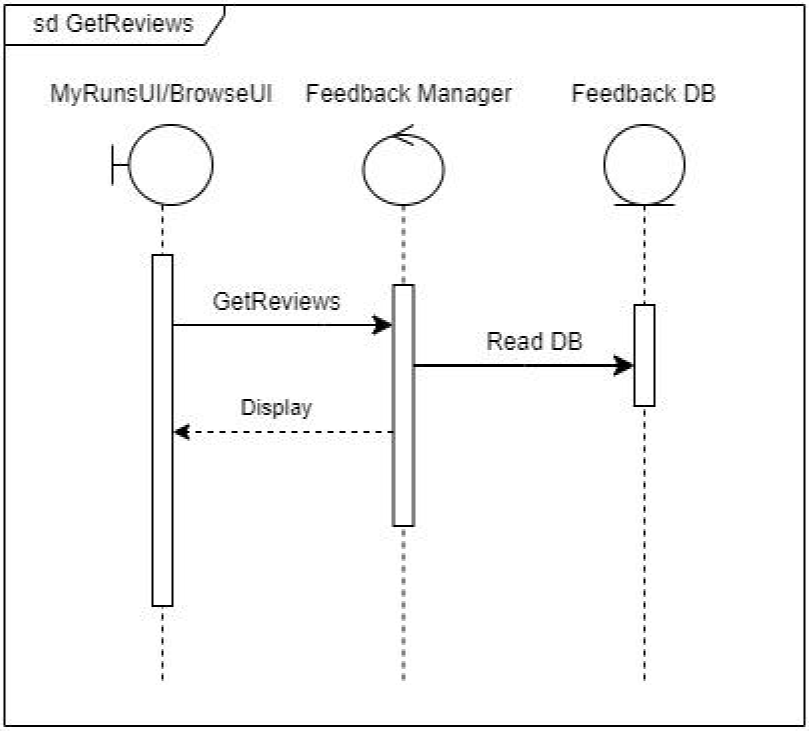
- Sequence Diagram
    - AddToMyRuns



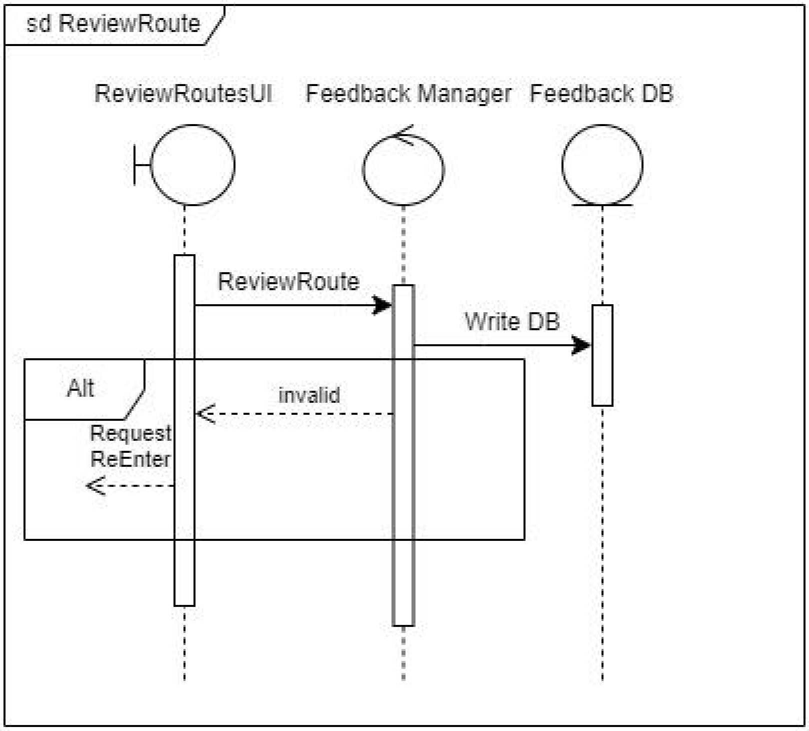
- BrowseRun


sd BrowseRun
User
MyRunsUI
ReviewRoutesUI
Runs Manager
Run DB
Interactive Map
Route DB
Select Filters
Filter Runs
Read DB
Reply
Display
Select Route
Filter Route
Read DB
Reply
Display
Display Route
Ref
GetReviews
Add Route to Favourites
FavouriteRoute
Write DB
Click Review Route
Ref
ReviewRoute

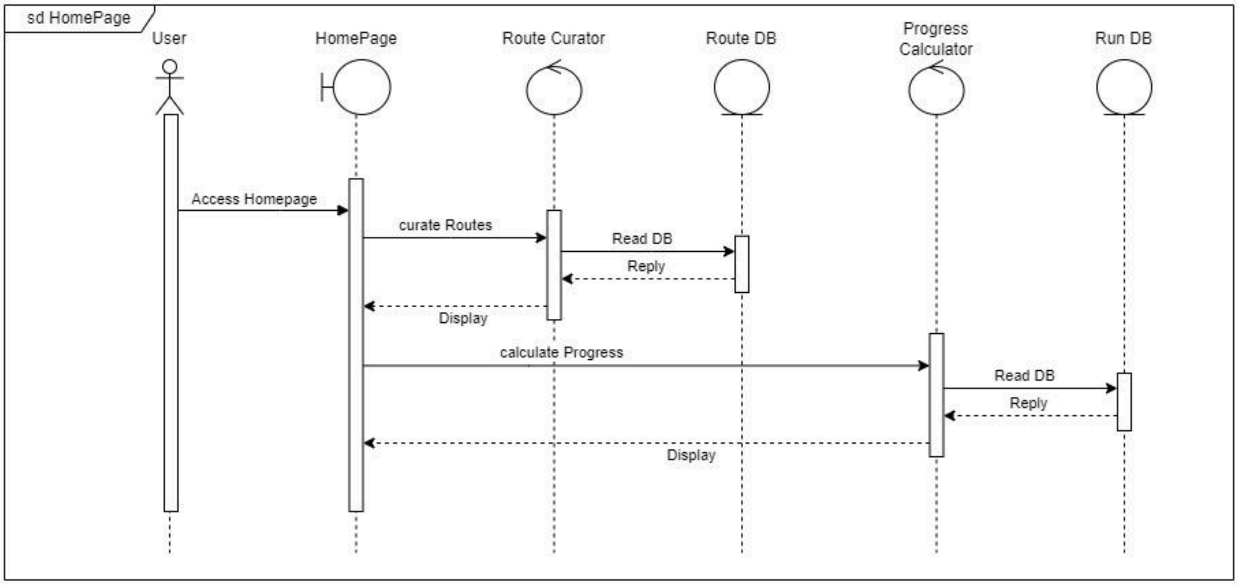
- GetReviews



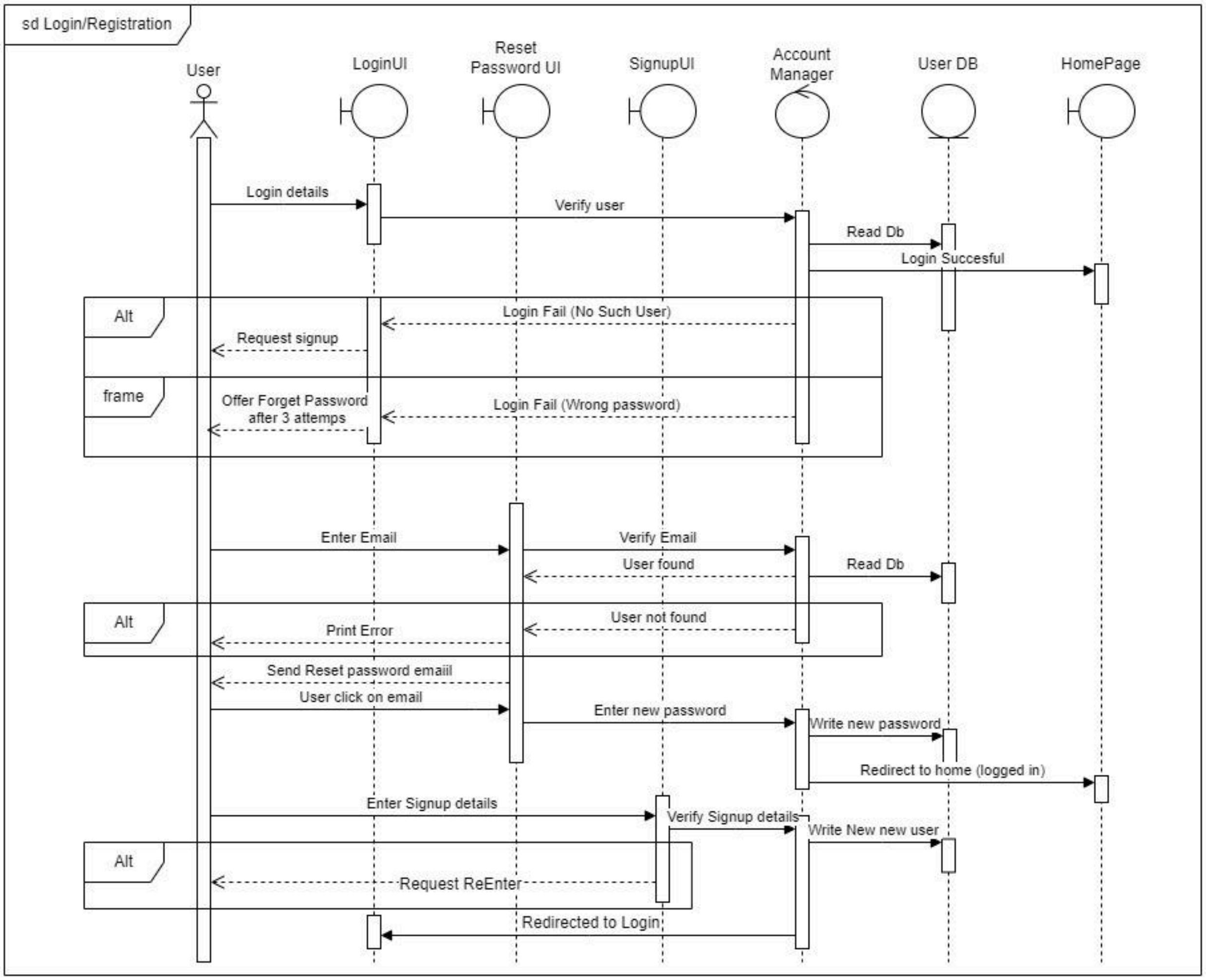
- ReviewRoute



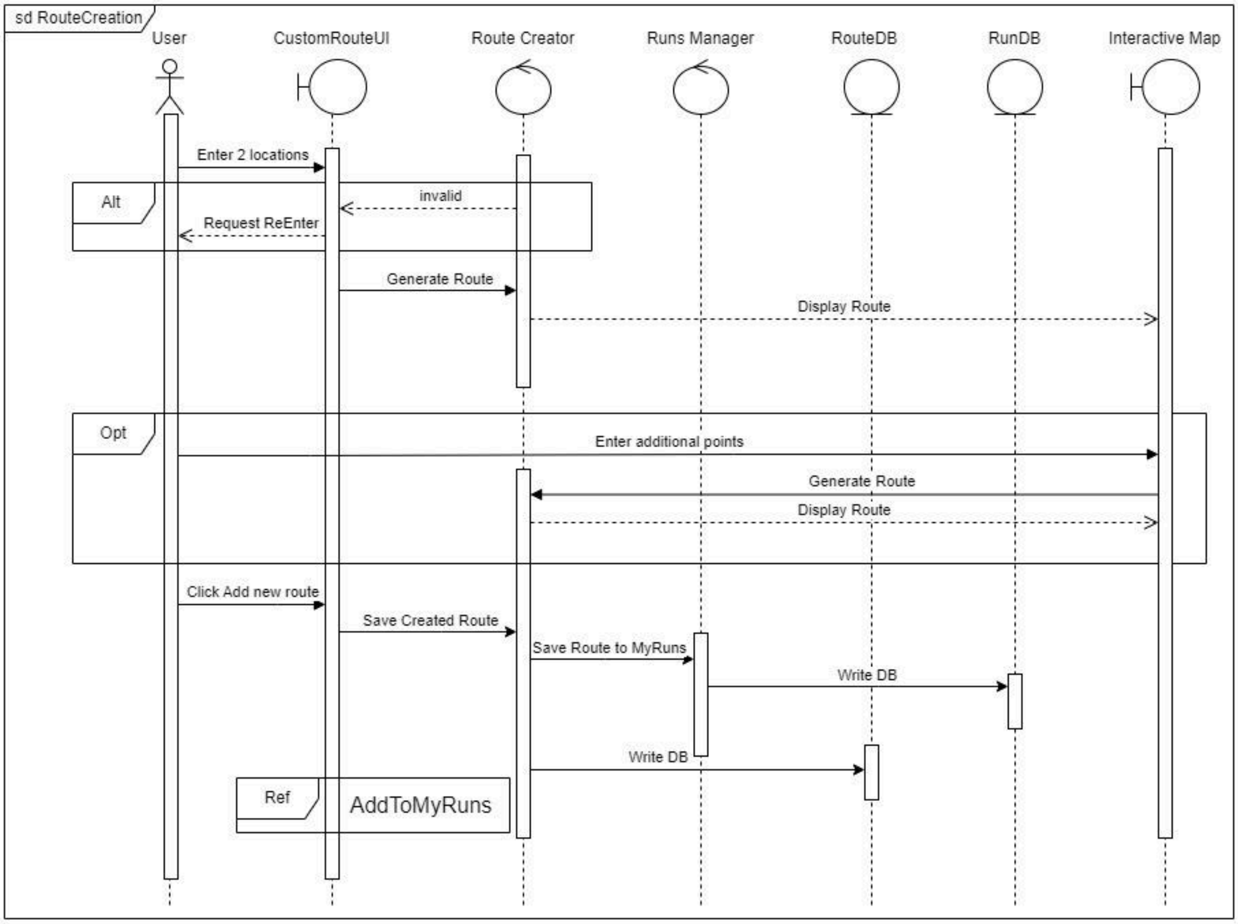
- HomePage


sd HomePage
User
HomePage
Route Curator
Route DB
Progress Calculator
Run DB
Access Homepage
curate Routes
Read DB
Reply
Display
calculate Progress
Read DB
Reply
Display

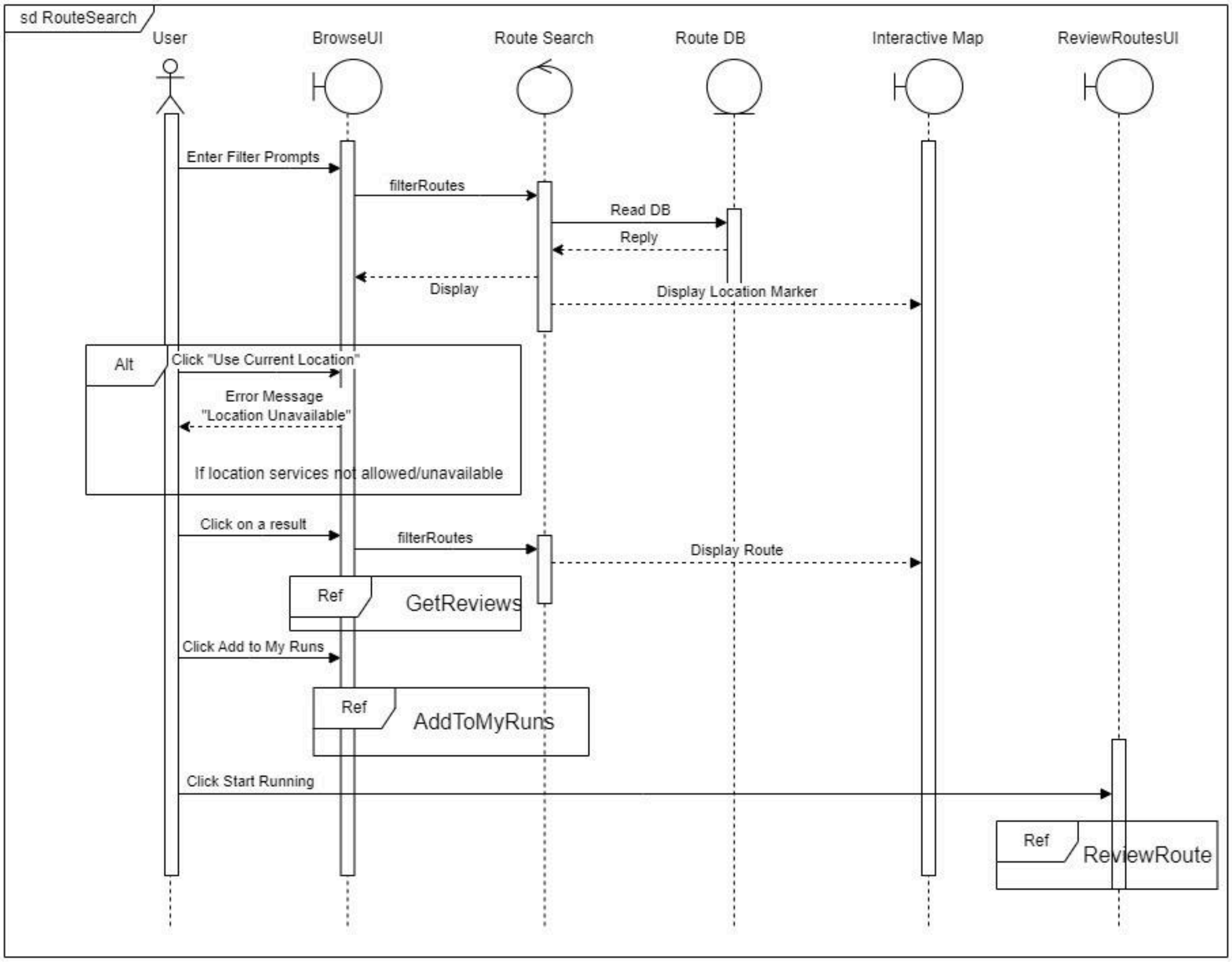
- LoginRegistration



- RouteCreation



- RouteSearch



Source: http://www.frontiernet.net/~kwiegers/process_assets/srs_template.doc