

Lab 1 Deliverables

- ☐ Documentation of functional and non-functional requirements (Darius, Jun Kiat)
- ☐ Data dictionary (Darius, Jun Kiat)
- ☐ Initial Use Case Model, consisting of Use Case diagram and Use Case descriptions (Javen, Zi Lun)
- ☐ [UI Mockups](#) (Zi Shen, Yi Rou)

Functional Requirements

1. User Authentication

1.1. Users must be able to create an account.

1.1.1. The system shall display text fields for users to enter user particulars.

1.1.2. All mandatory fields must be filled in before account creation.

1.1.2.1. The system must ensure that all fields are filled with valid data.

1.1.2.2. The system shall highlight fields filled with invalid data.

1.2. Users must be able to login

1.2.1. The system shall ask for username and password for login verification

1.2.2. The system shall disable login function for 3 minutes after 5 failed login attempts

1.3. Users must be able to logout

1.3.1. The system shall prompt the user for logout confirmation

1.4. User must be able to use Forgot password button to recover account if they have forgotten their password

1.4.1. The system shall ask for the user's email address

1.4.2. The system must send an email to the user's email address to verify and reset password

2. Route Discovery

2.1. Users shall be able to search for predefined running routes in their area by inputting their postal code or name location

- 2.1.1. The system shall display text fields to ask for starting location, ending location, and required distance
 - 2.1.1.1. The user may choose to only input selected fields
 - 2.1.1.2. The system will display recommended routes based on past inputs by user
 - 2.2. Users shall be able to search for predefined running routes with their GPS location
 - 2.2.1. The system shall display all routes with starting point within 1km of user's GPS location
 - 2.3. The system must display route details
 - 2.3.1. The system must display distance information
 - 2.3.2. The system must display terrain information
 - 2.3.3. The system must display estimated time of completion of routes.
 - 2.4. The system shall allow user to filter route details to their preference
 - 2.4.1. The system shall allow users to filter recommended routes based on terrain
 - 2.4.2. The system shall allow users to filter recommended routes based on popularity
 - 2.4.3. The system shall allow users to filter recommended routes based on Distance
- 3. Custom Route Creation
 - 3.1. Users shall be able to create a custom route by selecting a start point and end point only
 - 3.2. Users shall be able to create a custom route by selecting particular segment(s) to include in their route only
 - 3.3. Users shall be able to create a custom route by selecting a start point and end point and segment(s) to include in their route
 - 3.4. Users shall be able to store the custom route
 - 3.5. Users shall be able to share the custom route
 - 3.6. The system shall generate at least 1 running route suggestion based on these points (Route creation algorithm: A-star).

4. Route Rating and Comments

- 4.1. Users shall be able to rate a route on a scale from 1 to 5
- 4.2. Users shall be able to input up to 1000 characters as part of a comment for a particular route.
- 4.3. The system shall store these ratings and feedback in the database.
- 4.4. The system shall display comments about a particular route to other users when recommending routes
- 4.5. The system shall display average rating of a particular route to other users when recommending routes

5. Database Management for Routes

- 5.1. The system shall manage a database of routes,
- 5.2. The system shall manage a database of user ratings
- 5.3. The system shall manage a database of route ratings
- 5.4. The system shall manage a database of feedback

6. User Data Tracking

- 6.1. The system must securely store user data.
- 6.2. User data must be able to be retrieved as needed.
- 6.3. User data must be able to be updated as needed

7. Route Information Retrieval

- 7.1. The system must be able to display relevant information for chosen or created routes.
 - 7.1.1. The system must be able to display terrain information for all parts of the routes
 - 7.1.2. The system must be able to display terrain information for all parts of the routes

Non-Functional Requirements

1. Performance

- 1.1. The system shall have at most 1% down time on weekly average
- 1.2. The system shall be able to support at least 10 active users concurrently
- 1.3. The system shall return at least 5 recommended routes within 10 seconds of a user's search

2. Reliability

- 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
- 2.2. Distances indicated in the system shall be accurate up to 0.5km
- 2.3. Forget password email must be sent out within 1 minute of pressing the 'Send Email' button

3. Usability

- 3.1. 80% of users must be able to perform a route search within 5 minutes of use.
- 3.2. 80% of users must be able to upload a custom route within 5 minutes of use.
- 3.3. 80% of users must be able to use upload route rating and feedback within 5 minutes of use.
- 3.4. 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)

4. Security

- 4.1. The system shall have at least 2 measures to prevent unauthorized access and data breaches
- 4.2. The system must implement secure session management practices, including the generation of unique session identifiers (session IDs).
- 4.3. All input fields must be validated
- 4.4. Password stored must be encrypted by sha256
- 4.5. Password must be masked with '*' in the input field

5. Scalability

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

6. Maintainability

- 6.1. The system shall allow developers to implement and deploy updates to any module without affecting the operation of other modules
- 6.2. All system updates shall be deployable with no more than 2 hours of downtime per month

DATA DICTIONARY

<u>Term</u>	<u>Definition</u>
User	Individuals who have created an account
System	Running application
Route	Pedestrian path that is suitable for running
Terrain	Type of terrain (e.g., urban, trail)
Location	Physical geographical location
Distance	Total distance of the route in kilometers
Route Creation Algorithm (A-star)	A pathfinding algorithm used by the system to recommend the most efficient route from point A (start location) to point B (end location) based on various factors such as distance, terrain, and user preferences. The A* algorithm is known for its efficiency and accuracy in finding the shortest path, taking into account the cost of the route and any obstacles or conditions that may affect the traversal
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
Hash	Unique digital fingerprint of a piece of data (Hashed password for security)
User data	Saved user information (comments, previous routes)

sha256	A cryptographic hash function that generates a fixed-size, 256-bit (32-byte) hash value
Name Location	A specific place that can be identified and searched through an API, such as the Google Places API, based on user input.

Users Table		
Field Name	Data Type	Description
UserID	Integer	Unique identifier for each user
Username	Varchar(50)	User's chosen username
PasswordHash	Varchar(255)	Hashed password for security
Email	Varchar(100)	User's email address
CreatedAt	DateTime	Account creation date and time
LastLogin	DateTime	Last successful login date and time
Routes Table		
Field Name	Data Type	Description
RouteID	Integer	Unique identifier for each route
StartLocation	Varchar(255)	Text description or coordinates of start point
EndLocation	Varchar(255)	Text description or coordinates of end point
Distance	Decimal(5,2)	Total distance of the route in kilometers
Terrain	Varchar(50)	Type of terrain (e.g., urban, trail)
Popularity	Integer	Indicator of route's popularity
UserRoutes Table		
Field Name	Data Type	Description
UserRouteID	Integer	Unique identifier for each user-route record
UserID	Integer	Identifier for the user
RouteID	Integer	Identifier for the route
CreatedAt	DateTime	Date and time the route was saved/created
Feedback Table		
Field Name	Data Type	Description
FeedbackID	Integer	Unique identifier for each feedback entry
RouteID	Integer	Identifier for the route being reviewed

UserID	Integer	Identifier for the user providing feedback
Rating	Integer	Rating given to the route (1-5)
Comment	Text	Textual feedback provided by the user
CommentDate	DateTime	Date and time the feedback was provided
System Components and Data Specifications		
Field Name	Data Type	Description
Route Creation Algorithm (A-star):	Algorithm / Functionality	A pathfinding algorithm used by the system to recommend the most efficient route from point A (start location) to point B (end location) based on various factors such as distance, terrain, and user preferences. The A* algorithm is known for its efficiency and accuracy in finding the shortest path, taking into account the cost of the route and any obstacles or conditions that may affect the traversal
Name Location	API Resource / External Data Reference	A specific place that can be identified and searched through an API, such as the Google Places API, based on user input.