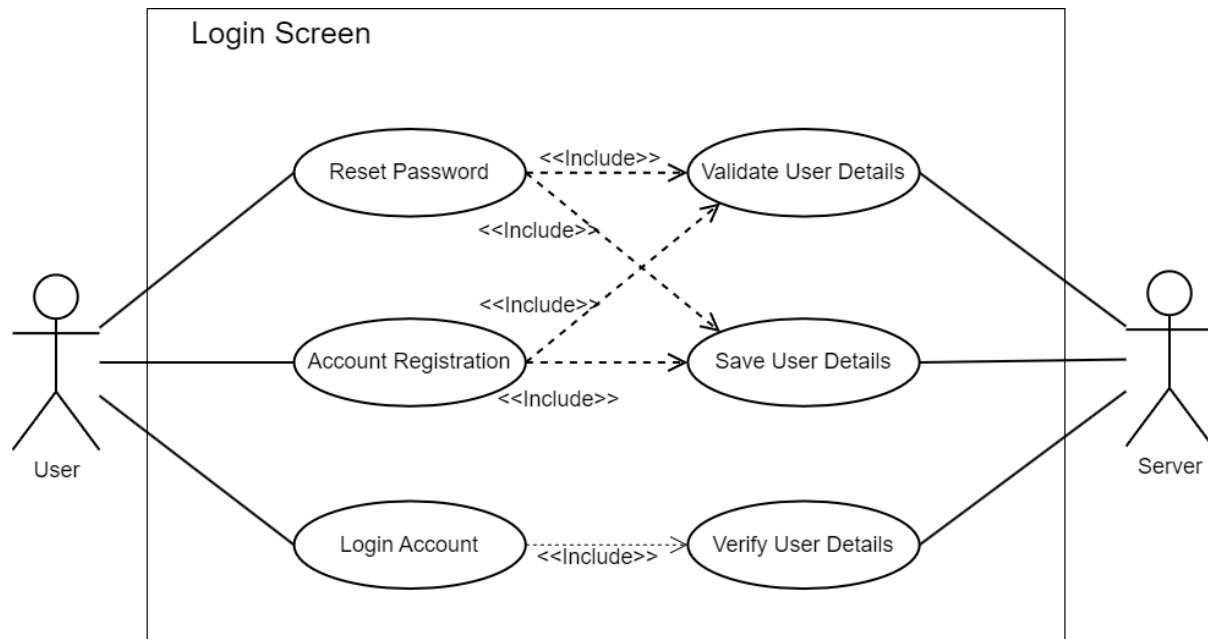


## Login Screen



## UC001 Account Registration

Use Case ID:	UC001		
Use Case Name:	Account Registration		
Created By:	Arun Ezekiel	Last Updated By:	Arun Ezekiel
Date Created:	01/02/2023	Date Last Updated:	01/02/2023

Actor:	User
Description:	Users have to register for an account if they do not possess one to use the functionalities of the application.
Preconditions:	User does not have a valid account and wants to register for one. The user must be on the application's start page.
Postconditions:	Account details are stored in the cloud database and the user will be directed to the application's home page.
Priority:	High
Frequency of Use:	Low
Flow of Events:	1. The user presses on the <b>&lt;Register&gt;</b> button on the application's start page. 2. The system displays the application's registration page. 3. The user enters a valid username, first name, last name, password, and email. 4. The user submits valid username, first name, last name, password, and email by pressing the <b>&lt;Sign Up&gt;</b> button. 5. The server validates the username, first name, last name and password. 6. The server saves the unique username, first name, last name and password on the cloud database. 7. The system displays the application's home screen. 8. The use case ends.

Alternative Flows:	<p>UC001.AC.1 Missing username, first name, last name, password, or email</p> <ol style="list-style-type: none"> <li>1. The system prompts the user for the missing inputs.</li> <li>2. Use case resumes at main flow step 3.</li> </ol> <p>UC001.AC.2 Invalid username, first name, last name, password, or email</p> <ol style="list-style-type: none"> <li>1. The system prompts the user for the invalid inputs.</li> <li>2. Use case resumes at main flow step 3.</li> </ol> <p>UC001.AC.3 Username taken</p> <ol style="list-style-type: none"> <li>1. The system displays a “Username taken” message.</li> <li>2. The system prompts the user for a username.</li> <li>3. Use case resumes at main flow step 3.</li> </ol> <p>UC001.AC.4 Email taken</p> <ol style="list-style-type: none"> <li>1. The system displays an “Email taken” message.</li> <li>2. The system prompts the user for an email.</li> <li>3. Use case resumes at main flow step 3.</li> </ol> <p>UC001.AC.5 Password does not meet requirements</p> <ol style="list-style-type: none"> <li>1. The system displays a “Password does not meet requirements” message.</li> <li>2. The system prompts the user for a password.</li> <li>3. Use case resumes at main flow step 3.</li> </ol>
Exceptions:	<p>UC001.EX.1 Cancellation of User Registration</p> <ol style="list-style-type: none"> <li>1. The user aborts the use case by pressing on the &lt;Back&gt; button.</li> <li>2. The system does not save any details.</li> <li>3. The user will be directed to the application's home page.</li> <li>4. The use case ends.</li> </ol>
Includes:	SVR.UC001, SVR.UC002
Special Requirements:	
Assumptions:	When the user enters the application’s registration page, the user must successfully create an account.
Notes and Issues:	

## UC002 Login Account

Use Case ID:	UC002		
Use Case Name:	Login Account		
Created By:	Arun Ezekiel	Last Updated By:	Arun Ezekiel
Date Created:	01/02/2023	Date Last Updated:	01/02/2023

Actor:	User
Description:	The user indicates intent to log in to his previously registered account to access the functionalities of the application.
Preconditions:	User has a valid account and is on the application’s start page.
Postconditions:	User successfully logs in and the application displays the home screen.
Priority:	High
Frequency of Use:	High

Flow of Events:	<ol style="list-style-type: none"> <li>1. The user presses on the &lt;Login&gt; button on the application's start page.</li> <li>2. The system displays the application's login page.</li> <li>3. The user enters a username and password.</li> <li>4. The user submits a username and password by pressing the &lt;Login&gt; button.</li> <li>5. The server verifies the username and password.</li> <li>6. The system displays the application's home screen.</li> </ol>
Alternative Flows:	<p>UC002.AC.01 Missing username and/or passwords</p> <ol style="list-style-type: none"> <li>1. The system prompts for a username and password.</li> <li>2. The use case resumes at main flow step 3.</li> </ol> <p>UC002.AC.02 Incorrect username and/or password</p> <ol style="list-style-type: none"> <li>1. The system displays an "Incorrect username and/or password" message.</li> <li>2. The system prompts for a username and password.</li> <li>3. The use case resumes at main flow step 3.</li> </ol>
Exceptions:	<p>UC002.EX.1 Cancellation of Account Login</p> <ol style="list-style-type: none"> <li>1. The user aborts the use case by pressing on the &lt;Back&gt; button.</li> <li>2. The system displays the application's start page.</li> <li>3. The use case ends.</li> </ol>
Includes:	SVR.UC003
Special Requirements:	
Assumptions:	When the user enters the application's account login page, the user must successfully login to their registered account.
Notes and Issues:	

## UC003 Reset Password

Use Case ID:	UC003		
Use Case Name:	Reset Password		
Created By:	Arun Ezekiel	Last Updated By:	Arun Ezekiel
Date Created:	01/02/2023	Date Last Updated:	01/02/2023

Actor:	User
Description:	The user indicates intent to reset their password as he may have forgotten his password.
Preconditions:	User has a valid account and is on the application's login page. The user remembers his registered email.
Postconditions:	User password successfully reset.
Priority:	Low
Frequency of Use:	Low
Flow of Events:	<ol style="list-style-type: none"> <li>1. The user presses on the &lt;Forgot Password?&gt; button on the application's login page.</li> <li>2. The system displays the application's reset password page.</li> <li>3. The user enters their email.</li> <li>4. The server sends the user a reset password link email.</li> <li>5. The user clicks the password reset link.</li> <li>6. The user is redirected to the reset password page on the application.</li> <li>7. The user enters a new password.</li> <li>8. The user submits the password.</li> </ol>

	9. The server validates the password. 10. The server saves the password. 11. The system displays the login screen. 12. The use case ends.
Alternative Flows:	UC003.AC.01 Invalid email <ol style="list-style-type: none"> <li>1. The system displays an “Invalid email” message.</li> <li>2. The system prompts for a valid email.</li> <li>3. The use case resumes at main flow step 3.</li> </ol> UC003.AC.02 Invalid password <ol style="list-style-type: none"> <li>1. The system displays an “Invalid password” message.</li> <li>2. The system prompts for a valid password.</li> <li>3. The use case resumes at main flow step 7.</li> </ol>
Exceptions:	UC003.EX.1 Cancellation of Password Reset <ol style="list-style-type: none"> <li>1. The user aborts the use case.</li> <li>2. The use case ends.</li> </ol>
Includes:	SVR.UC001, SVR.UC002
Special Requirements:	
Assumptions:	When the user enters the application’s reset password page, the user must successfully reset their password.
Notes and Issues:	

## SVR.UC001 Validate User Details

Use Case ID:	SVR.UC001		
Use Case Name:	Validate User Details		
Created By:	Arun Ezekiel	Last Updated By:	Arun Ezekiel
Date Created:	01/02/2023	Date Last Updated:	01/02/2023

Actor:	Server
Description:	The use case begins when the actor receives user details to validate. The use case ends when the username is not taken, email not taken, password meets requirements and first name/last name has been filled.
Preconditions:	Email entered, password entered, username entered, first name entered and last name entered.
Postconditions:	Accurately verified the user details and sends an appropriate message to the system.
Priority:	High
Frequency of Use:	High
Flow of Events:	1. The server receives user details. 2. The server validates the email is not taken. 3. The server validates the unique username is unique. 4. The server validates the first name is not missing. 5. The server validates the last name is not missing. 6. The server validates the password meets the requirements. 7. The server successfully validates all the user details. 8. The use case ends.
Alternative Flows:	SVR.UC001.AC1 Username Taken. 1. The server sends a “Username Taken” message to the system. 2. The use case ends.  SVR.UC001.AC2 Email Taken

	3. 1. The server sends an “Email Taken” message to the system. 4. 2. The use case ends.  SVR.UC001.AC3 First Name and/or Last Name is missing 5. 1. The server sends a “First Name and/or Last Name is missing” message to the system. 6. 2. The use case ends.  SVR.UC001.AC4 Password does not meet requirements 7. 1. The server sends a “Password does not meet requirements” message to the system. 8. 2. The use case ends.
Exceptions:	
Includes:	
Special Requirements:	
Assumptions:	
Notes and Issues:	

## SVR.UC002 Save User details

Use Case ID:	SVR.UC003		
Use Case Name:	Save User details		
Created By:	Arun Ezekiel	Last Updated By:	Arun Ezekiel
Date Created:	01/02/2023	Date Last Updated:	01/02/2023

Actor:	Server
Description:	The use case begins when the actor receives validated user details - email, username, password, first name and last name.
Preconditions:	Email validated, password validated, username validated, first name validated, and last name validated.
Postconditions:	User details are saved successfully.
Priority:	High
Frequency of Use:	High
Flow of Events:	1. The server receives validated user details. 2. The server saves the email. 3. The server saves the username. 4. The server saves the first name. 5. The server saves the last name. 6. The server saves the password. 7. The server successfully saved all user details. 8. The use case ends.
Alternative Flows:	
Exceptions:	SVR.UC002.EX1 Storage space full 1. The server aborts the case due to not enough space left in storage. 2. The server sends a “Insufficient Storage, please contact an admin” message to the system. 3. The use case ends.

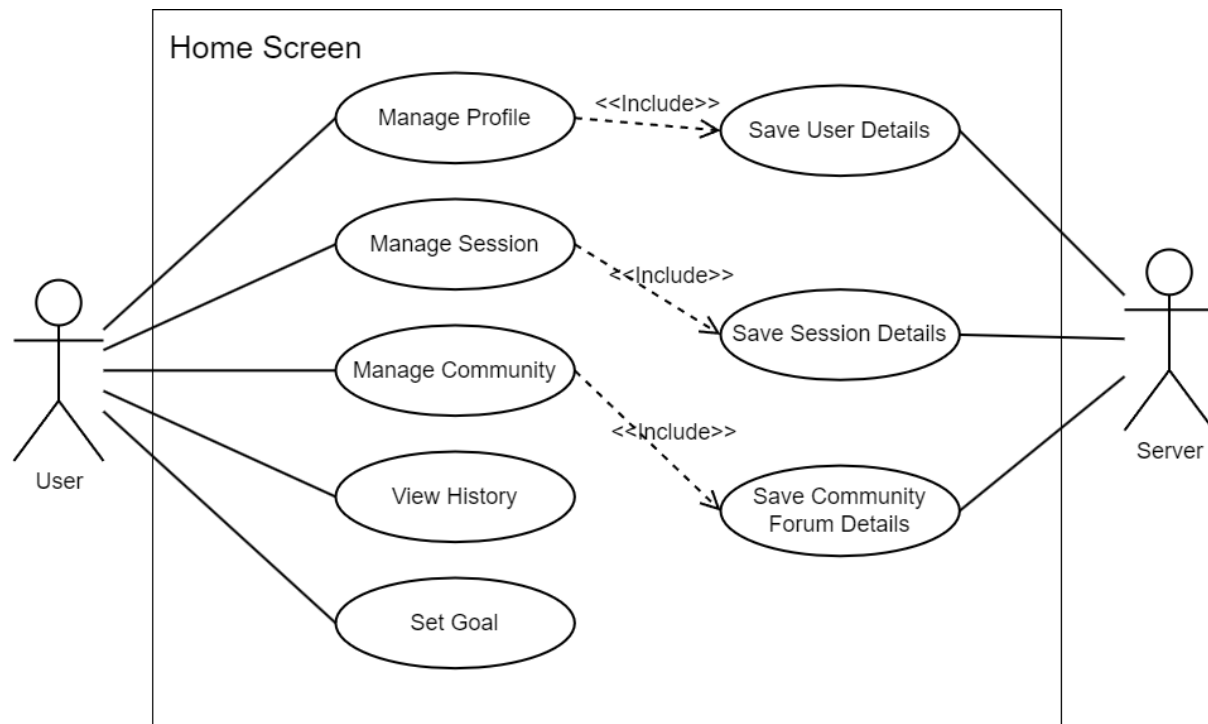
Includes:	
Special Requirements:	
Assumptions:	The user details will always be successfully saved after validation of details.
Notes and Issues:	

## SVR.UC003 Verify User details

Use Case ID:	SVR.UC003		
Use Case Name:	Verify User details		
Created By:	Arun Ezekiel	Last Updated By:	Arun Ezekiel
Date Created:	01/02/2023	Date Last Updated:	01/02/2023

Actor:	Server
Description:	The use case begins when the actor receives user details to verify and ends when it has been verified.
Preconditions:	Username entered, password entered.
Postconditions:	Accurately verifies the authenticity of username and password pair and server sends an appropriate message to the system.
Priority:	
Frequency of Use:	
Flow of Events:	<ol style="list-style-type: none"> <li>1 .The server receives a username and password pair.</li> <li>2. The server verifies the username is in the database and the password received matches with the password stored in the database.</li> <li>3. The server sends a “Verified” message to the system.</li> <li>4. The use case ends.</li> </ol>
Alternative Flows:	SVR.UC003.AC1 Username not found and/or Password mismatch <ol style="list-style-type: none"> <li>1. The server sends a “Username not found and/or Password mismatch” message to the system.</li> <li>2. The use case ends.</li> </ol>
Exceptions:	
Includes:	
Special Requirements:	
Assumptions:	
Notes and Issues:	

## Home Screen



## UC004 Manage Profile

Use Case ID:	UC004		
Use Case Name:	Manage Profile		
Created By:	Arun Ezekiel	Last Updated By:	Arun Ezekiel
Date Created:	01/02/2023	Date Last Updated:	01/02/2023

Actor:	User
Description:	The user indicates interest to view his profile. The profile will display his username, first name, last name, email, total distance to date and total hours to date on sessions. The user can edit his profile details.
Preconditions:	User is logged in and is on the application's home page.
Postconditions:	The system switches to an application's profile page and keeps the user's information updated.
Priority:	Mid
Frequency of Use:	Mid
Flow of Events:	<ol style="list-style-type: none"> <li>1. The user presses the &lt;Personal&gt; button on the application's home page.</li> <li>2. The system displays the application's profile page.</li> <li>2. Use case ends.</li> </ol>
Alternative Flows:	UC004.AC.01 Edit profile <ol style="list-style-type: none"> <li>1. The user can make changes to his profile by pressing the &lt;Edit&gt; icon on the application's profile page.</li> <li>2. The system displays the application's edit profile page.</li> <li>3. Users can update his profile by inputting new profile information.</li> <li>4. User can save his changes by pressing the &lt;Save&gt; button.</li> <li>5. User profile information will be updated on the cloud database.</li> </ol>

	6. The system displays the application's profile page. 7. The use case resumes at main flow step 2.  UC004.AC.02 Cancel edit 1. The user can revert any changes to his profile during current edit by pressing the <Cancel> button on the application's edit profile page. 2. The use case resumes at main flow step 2.
Exceptions:	UC004.EX.01 Invalid username, first name, last name and email 1. The system will display a "Invalid username, first name, last name and email" message. 2. The system will prompt the user to enter a username, first name, last name and email. 3. The use case resumes at sub flow UC004.AC.01 step 3.
Includes:	SVR.UC002
Special Requirements:	
Assumptions:	
Notes and Issues:	

## UC006 Manage Session

Use Case ID:	UC006			
Use Case Name:	Manage Session			
Created By:	Arun Ezekiel		Last Updated By:	Arun Ezekiel
Date Created:	01/02/2023		Date Last Updated:	01/02/2023

Actor:	User
Description:	The user indicates intent to record a session. The user can start, stop, pause and resume a session. The session will show the users his current session's details, including his current location on the map, the route from the initial start point of the user to the user's current location on the map, total distance traveled, time taken, and average speed.
Preconditions:	User is logged in and is on the application's home page.
Postconditions:	A session is stopped and stored on the cloud database. User's session history will be updated.
Priority:	High
Frequency of Use:	High
Flow of Events:	1. The user starts a session by either pressing the <Cycle> icon or the <Running> icon. 2. The system will start recording the session. 3. The system will constantly update the current session's details. 4. The user stops the current session by pressing the <Stop> button. 5. The system will stop recording the session. 6. The system will save the information on the cloud database. 7. The system will update the user's history. 8. The system displays the application's home page.
Alternative Flows:	UC006.AC.01 Pause and Resume



	<ol style="list-style-type: none"> <li>1. The user can pause a current session by pressing the <b>&lt;Pause&gt;</b> button after main flow step 2.</li> <li>2. The system will temporarily stop updating the session.</li> <li>3. The system will continue updating the session after the user presses the <b>&lt;Resume&gt;</b> button.</li> <li>4. The use case resumes at main flow step 3.</li> </ol>
Exceptions:	
Includes:	SVR.UC004
Special Requirements:	
Assumptions:	
Notes and Issues:	

## UC007 Manage Community

Use Case ID:	UC007		
Use Case Name:	Manage Community		
Created By:	Arun Ezekiel	Last Updated By:	Arun Ezekiel
Date Created:	01/02/2023	Date Last Updated:	01/02/2023

Actor:	User
Description:	The user can interact with other users through the community forum page by posting, liking posts, and commenting under posts. The user can also delete his posts. The user can end the use case by pressing the <b>&lt;Back&gt;</b> button.
Preconditions:	User is logged in and is on the application's home page.
Postconditions:	Post created, post deleted, post liked or post commented.
Priority:	Mid
Frequency of Use:	High
Flow of Events:	<ol style="list-style-type: none"> <li>1. The user selects the Community screen by pressing the <b>&lt;Community&gt;</b> icon.</li> <li>2. The user chooses to create a post by pressing the <b>&lt;Post&gt;</b> button.</li> <li>3. The user chooses a session to post.</li> <li>4. The user enters a caption.</li> <li>5. 5. The user submits the post.</li> <li>6. 6. The community forum will update with the user's post.</li> <li>7. 7. The use case ends.</li> </ol>
Alternative Flows:	<p>UC007.AC.01 Post Deleted</p> <ol style="list-style-type: none"> <li>1. The user searches for the post they created to delete after main flow step 1.</li> <li>2. The user chooses the post for deletion by pressing the <b>&lt;Delete&gt;</b> button beside the post.</li> <li>3. The system prompts the user for confirmation of deletion.</li> <li>4. The user confirms.</li> <li>5. The post is deleted.</li> <li>6. The use case ends.</li> </ol> <p>UC007.AC.02 Like Post</p> <ol style="list-style-type: none"> <li>1. The user searches for a post to like after main flow step 1.</li> <li>2. The user chooses a post to like by pressing the <b>&lt;Like&gt;</b> icon beside the post.</li> <li>3. The user likes the post.</li> <li>4. The use case ends.</li> </ol>

	<p>UC007.AC.03 Unlike Post</p> <ol style="list-style-type: none"> <li>1. The user searches for a post they liked to unlike after main flow step 1.</li> <li>2. The user chooses the post to unlike by pressing the <b>&lt;Like&gt;</b> icon beside the post.</li> <li>3. The user unlikes the post.</li> <li>4. The use case ends.</li> </ol> <p>UC007.AC.04 Comment under post</p> <ol style="list-style-type: none"> <li>1. The user searches for a post to comment under after main flow step 1.</li> <li>2. The user chooses a post to comment under by pressing the <b>&lt;Comment&gt;</b> button on a post.</li> <li>3. The user enters a comment.</li> <li>4. The user submits the comment.</li> <li>5. The use case ends.</li> </ol> <p>UC007.AC.05 Delete comment under post</p> <ol style="list-style-type: none"> <li>1. The user searches for the comment they created to delete after main flow step 1.</li> <li>2. The user chooses the comment for deletion by pressing the <b>&lt;Delete&gt;</b> button beside the comment.</li> <li>3. The system prompts the user for confirmation of deletion.</li> <li>4. The user confirms.</li> <li>5. The comment is deleted.</li> <li>6. The use case ends.</li> </ol> <p>UC007.AC.06 Abort Manage Community</p> <ol style="list-style-type: none"> <li>1. The user aborts the use case by pressing on the <b>&lt;Back&gt;</b> button.</li> <li>2. The user will be directed to the application's home page.</li> <li>3. The use case ends.</li> </ol>
Exceptions:	
Includes:	SVR.UC005
Special Requirements:	
Assumptions:	
Notes and Issues:	

## UC008 View History

Use Case ID:	UC008		
Use Case Name:	View History		
Created By:	Arun Ezekiel	Last Updated By:	Arun Ezekiel
Date Created:	01/02/2023	Date Last Updated:	01/02/2023

Actor:	User
Description:	The user indicates the intent to view details about his past sessions. The details include the date, start time, distance traveled, timing, average speed and cycling route of the session.
Preconditions:	User is logged in, has past sessions and is on the application's home page.
Postconditions:	The system displays the relevant information about the user's past sessions.

Priority:	High
Frequency of Use:	High
Flow of Events:	<ol style="list-style-type: none"> <li>1. The user can view his past sessions by pressing the <b>&lt;Sessions&gt;</b> button on the application's home page.</li> <li>2. The system displays the application's past session page.</li> <li>3. The user can press the sessions to view the route of the session, traced on a map.</li> <li>4. The use case ends.</li> </ol>
Alternative Flows:	
Exceptions:	
Includes:	
Special Requirements:	
Assumptions:	
Notes and Issues:	

## UC009 Set Goal

Use Case ID:	UC009		
Use Case Name:	Set Goal		
Created By:	Arun Ezekiel	Last Updated By:	Arun Ezekiel
Date Created:	01/02/2023	Date Last Updated:	01/02/2023

Actor:	User
Description:	The user indicates the intent to set a new monthly goal (in kilometers). The user can edit the monthly goal. The monthly goal will display.
Preconditions:	User is logged in and is on the application's session page.
Postconditions:	The monthly goal is successfully updated.
Priority:	Low
Frequency of Use:	Mid
Flow of Events:	<ol style="list-style-type: none"> <li>1. The user can set a new monthly goal by pressing the <b>&lt;Monthly Goal&gt;</b> icon on the application's sessions page.</li> <li>2. The user inputs the monthly goal (in kilometers).</li> <li>3. The user submits the monthly goal by pressing the <b>&lt;Confirm&gt;</b> button.</li> <li>4. The system validates the monthly goal.</li> <li>5. The system successfully displays the new monthly goal.</li> <li>6. The use case ends.</li> </ol>
Alternative Flows:	UC009.AC.1 Edit monthly goal <ol style="list-style-type: none"> <li>The system prompts the user to enter a monthly goal.</li> <li>The use case resumes at main flow step 2.</li> </ol>
Exceptions:	UC009.EX.1 Invalid Monthly goal <ol style="list-style-type: none"> <li>The system displays "Monthly goal must not exceed 999KM" message.</li> <li>The system prompts the user to enter a monthly goal.</li> <li>The use case resumes at main flow step 2.</li> </ol> UC009.EX.2 Cancellation of Goal Setting <ol style="list-style-type: none"> <li>User aborts the use case by pressing the <b>&lt;X&gt;</b> icon.</li> <li>The system displays the application's session page.</li> <li>The use case ends.</li> </ol>
Includes:	

Special Requirements:	
Assumptions:	When the user enters the application's goal setting page, the user must successfully edit or set a goal.
Notes and Issues:	

### SVR.UC004 Save Session details

Use Case ID:	SVR.UC004		
Use Case Name:	Save Session details		
Created By:	Arun Ezekiel	Last Updated By:	Arun Ezekiel
Date Created:	01/02/2023	Date Last Updated:	01/02/2023

Actor:	Server
Description:	The use case begins when the actor receives session details - Route, Total distance traveled, time taken and average speed.
Preconditions:	User is logged in and has stopped the session.
Postconditions:	User session details are saved successfully.
Priority:	High
Frequency of Use:	High
Flow of Events:	9. 1. The server receives user session details. 10. 2. The server saves the route. 11. 3. The server saves the total distance traveled. 12. 4. The server saves the time taken. 13. 5. The server saves the average speed. 14. 6. The use case ends.
Alternative Flows:	
Exceptions:	SVR.UC004.EX1 Storage space full 1. The server aborts the case due to not enough space left in storage. 2. The server sends a "Insufficient Storage, please contact an admin" message to the system. 3. The use case ends.
Includes:	
Special Requirements:	
Assumptions:	The user details will always be successfully saved.
Notes and Issues:	

### SVR.UC005 Save Community forum details

Use Case ID:	SVR.UC005		
Use Case Name:	Save Community forum details		
Created By:	Arun Ezekiel	Last Updated By:	Arun Ezekiel
Date Created:	01/02/2023	Date Last Updated:	01/02/2023

Actor:	Server
Description:	The use case begins when the actor receives community forum details - Post created, post deleted, post liked, post unliked, post commented, comment deleted.
Preconditions:	User is logged in.

Postconditions:	User community forum details are saved successfully.
Priority:	High
Frequency of Use:	High
Flow of Events:	15. 1. The server receives community forum details. 16. 2. The server saves the appropriate details. 17. 3. The use case ends.
Alternative Flows:	
Exceptions:	SVR.UC005.EX1 Storage space full 1. The server aborts the case due to not enough space left in storage. 2. The server sends a “Insufficient Storage, please contact an admin” message to the system. 3. The use case ends.
Includes:	
Special Requirements:	
Assumptions:	The community forum details will always be successfully saved.
Notes and Issues:	