Use Cases

for

<SC2006 Project>

Version 1.01 approved

Prepared by <Leon>

<SCSE Group 7>

<24-02-2024>

Revision History

Name	Date	Reason For Changes	Version
Leon	11-02-2024	Added alternative flow for A1 use case based on suggestions by Jon Added alternative flow for A6 use case based on suggestions by Jon	1.01
Leon	24-02-2024	Updated Boundary, Entity and Control Classes based on our Conceptual Model.	1.1

Guidance for Use Case Template

Document each use case using the template shown in the Appendix. This section provides a description of each section in the use case template.

1. Use Case Identification

1.1. Use Case ID

Give each use case a unique numeric identifier, in hierarchical form: X.Y. Related use cases can be grouped in the hierarchy. Functional requirements can be traced back to a labeled use case.

1.2. Use Case Name

State a concise, results-oriented name for the use case. These reflect the tasks the user needs to be able to accomplish using the system. Include an action verb and a noun. Some examples:

- View part number information.
- Manually mark hypertext source and establish link to target.
- Place an order for a CD with the updated software version.

1.3. Use Case History

1.3.1 Created By

Supply the name of the person who initially documented this use case.

1.3.2 Date Created

Enter the date on which the use case was initially documented.

1.3.3 Last Updated By

Supply the name of the person who performed the most recent update to the use case description.

1.3.4 Date Last Updated

Enter the date on which the use case was most recently updated.

2. Use Case Definition

2.1. Actor

An actor is a person or other entity external to the software system being specified who interacts with the system and performs use cases to accomplish tasks. Different actors often correspond to different user classes, or roles, identified from the customer community that will use the product. Name the actor(s) that will be performing this use case.

2.2. Description

Provide a brief description of the reason for and outcome of this use case, or a high-level description of the sequence of actions and the outcome of executing the use case.

2.3. Preconditions

List any activities that must take place, or any conditions that must be true, before the use case can be started. Number each precondition. Examples:

1. User's identity has been authenticated.
2. User's computer has sufficient free memory available to launch task.

2.4. Postconditions

Describe the state of the system at the conclusion of the use case execution. Number each postcondition. Examples:

1. Document contains only valid SGML tags.
2. Price of item in database has been updated with new value.

2.5. Priority

Indicate the relative priority of implementing the functionality required to allow this use case to be executed. The priority scheme used must be the same as that used in the software requirements specification.

2.6. Frequency of Use

Estimate the number of times this use case will be performed by the actors per some appropriate unit of time.

2.7. Flow of Events

Provide a detailed description of the user actions and system responses that will take place during execution of the use case under normal, expected conditions. This dialog sequence will ultimately lead to accomplishing the goal stated in the use case name and description. This description may be written as an answer to the hypothetical question, "How do I <accomplish the task stated in the use case name>?" This is best done as a numbered list of actions performed by the actor, alternating with responses provided by the system.

2.8. Alternative Flows

Document other, legitimate usage scenarios that can take place within this use case separately in this section. State the alternative course, and describe any differences in the sequence of steps that take place. Number each alternative course using the Use Case ID as a prefix, followed by "AC" to indicate "Alternative Course". Example: X.Y.AC.1.

2.9. Exceptions

Describe any anticipated error conditions that could occur during execution of the use case, and define how the system is to respond to those conditions. Also, describe how the system is to respond if the use

case execution fails for some unanticipated reason. Number each exception using the Use Case ID as a prefix, followed by "EX" to indicate "Exception". Example: X.Y.EX.1.

2.10. Includes

List any other use cases that are included ("called") by this use case. Common functionality that appears in multiple use cases can be split out into a separate use case that is included by the ones that need that common functionality.

2.11. Special Requirements

Identify any additional requirements, such as nonfunctional requirements, for the use case that may need to be addressed during design or implementation. These may include performance requirements or other quality attributes.

2.12. Assumptions

List any assumptions that were made in the analysis that led to accepting this use case into the product description and writing the use case description.

2.13. Notes and Issues

List any additional comments about this use case or any remaining open issues or TBDs (To Be Determineds) that must be resolved. Identify who will resolve each issue, the due date, and what the resolution ultimately is.

Use Case ID:	A1		
Use Case Name:	User Login		
Created By:	Leon	Last Updated By:	Leon
Date Created:	07-02-2024	Date Last Updated:	24-02-2024

Actor:	User
Description:	This use case describes the process by which a user logs into the website.
Preconditions:	1. The user has access to a device with internet connectivity. 2. The user has registered an account on the website.
Postconditions:	1. The user gains access to their account dashboard upon successful login. 2. If login fails, appropriate error messaging is displayed.
Priority:	High
Frequency of Use:	Multiple times per day; everytime they wish to visit the website to access its services.
Flow of Events:	1. The user navigates to the Login Page Interface. 2. The user enters their username/email and password. 3. The system includes the Authentication Manager to verify the user's credentials. 4. If the credentials are valid: 4.1. The system logs the user in and redirects them to the Dashboard Interface. 4.2. The system creates a session for the user and stores session information. 5. If the credentials are invalid: 5.1. The system triggers the Error Handling Controller to display an error message on the Error Message Display. 5.2. The user can retry the login process.
Alternative Flows:	A1.AF-S1: Incorrect username or password 1. The system triggers the Error Handling Controller to display an error message on the Error Message Display. 2. The user clicks on the "Forgot Password?" link. 3. The system prompts the user to enter their registered email address. 4. The user enters their email address and submits the request. 5. The system sends a password reset link to the user's email address. 6. The user checks their email and clicks on the password reset link. 7. The system validates the link and prompts the user to create a new password. 8. The user enters a new password and confirms it.

	9. The system updates the user's password and confirms successful password reset. 10. The user can now log in with the new password. A1.AF-S3: More than 5 repeated failed attempts 1. The system displays an error message indicating repeated failed attempts: "You have exceeded the amount of attempts. Please try again later." 2. The user has to wait 10 minutes before the next attempt. 3. The user can reattempt for 5 more attempts to log in: 3.1 If the user is successful, proceed to the home page. 3.2 If the user is still unsuccessful, proceed to step 2 of the alternative flow A1.AF-S1.
Exceptions:	NA
Includes:	Verify Login
Special Requirements:	1. The system must ensure secure transmission of login credentials. 2. The system must implement measures to prevent brute force attacks. 3. The login page must have appropriate accessibility features.
Assumptions:	1. The user has a valid account registered on the website. 2. The website has adequate security measures in place to protect user data. 3. The user remembers the email account associated with the account.
Notes and Issues:	TBD: Implement password reset functionality

Use Case ID:	A2		
Use Case Name:	User Registration		
Created By:	Leon	Last Updated By:	Leon
Date Created:	08-02-2024	Date Last Updated:	24-02-2024

Actor:	User
Description:	This use case describes the process by which a user registers for an account on the website.
Preconditions:	1. The user has access to a device with internet connectivity. 2. The user navigates to the website's registration page.
Postconditions:	1. The user successfully registers an account and gains access to the website's features. 2. If registration fails, appropriate error messaging is displayed.
Priority:	High
Frequency of Use:	When User does not have an account, once per email
Flow of Events:	1. The user navigates to the Registration Page Interface. 2. The user enters their personal information, including name, email address, and desired username. 3. The system includes the Registration Manager to validate user input and check for duplicates. 4. If all required fields are filled correctly and the username is unique: 4.1. The system creates the user account. 4.2. The system sends a confirmation email to the user's provided email address. 4.3. The user confirms their email address by clicking on the confirmation link in the Confirmation Email Interface. 4.4. The system activates the user account. 4.5. The system displays a success message indicating successful registration. 5. If any required fields are missing or invalid, or if the username is not unique: 5.1. The system triggers the Error Handling Controller to display error messages on the Registration Page Interface. 5.2. The user corrects the errors and resubmits the registration form. 6.2.2. The user corrects the errors and resubmits the registration form.
Alternative Flows:	A2.AF-S1: User Already Registered 1. The user attempts to register with an email address or username that is already associated with an existing account. 2. The system checks the database to verify if the email address or username is already registered. 3. The system determines that the email address or username is already in use. 4. The system displays an error message to the user indicating that the email address or username is already registered. 5. The user is prompted to log in using their existing credentials or to recover their existing account. 6. The user follows the instructions provided in the error message to log in or recover their existing account.

Exceptions:	1. If the user's chosen username is not unique: 1.1. The system displays an error message indicating that the username is already taken. 1.2. The user chooses a different username and resubmits the registration form. 2. If the user does not confirm their email address within a specified time period: 2.1. The system deactivates the user account. 2.2. The user must request a new confirmation email to activate their account.
Includes:	NA
Special Requirements:	1. The system must ensure secure transmission and storage of user registration data. 2. The registration page must have appropriate accessibility features. 3. The system must implement measures to prevent automated or fraudulent registrations.
Assumptions:	1. The user has access to a valid email address for account verification. 2. The website has adequate security measures in place to protect user data.
Notes and Issues:	TBD: CAPTCHA for anti-spam measures

Use Case ID:	A3		
Use Case Name:	Manage Personal Profile		
Created By:	Leon	Last Updated By:	Leon
Date Created:	08-02-2024	Date Last Updated:	24-02-2024

Actor:	User
Description:	This use case describes the process by which a user views their personal profile, including their health information, on the website.
Preconditions:	1. The user has successfully logged into their account. 2. The user has navigated to the personal profile section of the website.
Postconditions:	1. The user is able to view their personal profile and health information. 2. The user is able to edit their personal profile and health information.
Priority:	Medium
Frequency of Use:	Occasional, whenever it is subject to change
Alternative Flows:	
Flow of Events:	1. The user navigates to the Personal Profile Interface. 2. The system retrieves the user's current personal profile information from the database. 3. The system displays the user's personal profile and health information on the Personal Profile Interface. 4. The user selects the option to manage their personal profile. 5. The system presents the user with options to update their personal information, including health-related data. 6. The user makes changes to their personal profile and health information as desired. 7. The user confirms the changes. 8. The system triggers the Profile Manager to validate user input and update the user's personal profile and health information. 9. If the update is successful: 9.1. The system triggers the Confirmation Message Controller to display a confirmation message indicating that the changes have been successfully saved. 10. If an error occurs during the update: 10.1. The system triggers the Error Handling Controller to display an error message indicating the issue. 10.2. The user may attempt to correct the error and resubmit the request.
Exceptions:	1. If the user encounters an error while managing their personal profile: 1.1. The system displays an error message indicating the issue. 1.2. The user may attempt to refresh the page or try again later.
Includes:	NA
Special Requirements:	1. The system must ensure that only authorised users have access to manage personal profiles.

	2. The personal profile management page must adhere to privacy and data protection regulations.
Assumptions:	1. The user has previously provided their health information during account setup or through a separate process. 2. The website has adequate security measures in place to protect user data.
Notes and Issues:	

Use Case ID:	A4		
Use Case Name:	Manage Fitness Goal		
Created By:	Leon	Last Updated By:	Leon
Date Created:	08-02-2024	Date Last Updated:	24-02-2024

Actor:	User
Description:	This use case describes the process by which a user manages their fitness goal on the website, including setting, modifying, or cancelling the goal.
Preconditions:	1. The user has successfully logged into their account. 2. The user has navigated to the fitness goal management section of the website.
Postconditions:	1. The user's fitness goal is updated or canceled as per their actions.
Priority:	Medium
Frequency of Use:	Once a day, or whenever they reach their current goal
Flow of Events:	1. The user navigates to the fitness goal management section of the website. 2. The system displays the current fitness goal settings, if any, for the user. 3. The user selects the option to manage their fitness goal. 4. The system presents the user with options to: 4.1. Set a new fitness goal. 4.2. Modify the existing fitness goal. 4.3. Cancel the fitness goal setting process. 5. The user selects the appropriate action: 5.1. If setting a new fitness goal: 5.1.1. The system triggers the Fitness Goal Manager to validate user input and update the user's fitness goal settings in the database. 5.1.2. The system displays a confirmation message indicating that the fitness goal has been successfully managed. 5.2. If modifying the fitness goal: 5.2.1. The system triggers the Fitness Goal Manager to validate user input and update the user's fitness goal settings in the database. 5.2.2. The system displays a confirmation message indicating that the fitness goal has been successfully managed. 5.3. If cancelling the fitness goal: 5.3.1. The system cancels the fitness goal setting process and returns the user to the previous page or dashboard. 6. The system updates the user's fitness goal settings in the database if applicable. 7. The system displays a confirmation message indicating that the fitness goal has been successfully managed.
Alternative Flows:	NA
Exceptions:	1. If the user encounters an error while managing their fitness goal: 1.1. The system triggers the Error Handling Controller to display an error message indicating the issue. 1.2. The user may attempt to refresh the page or try again later.
Includes:	Track Calories

	Track Progress
Special Requirements:	1. The system must ensure that only authorised users have access to manage fitness goals. 2. The fitness goal management page must provide guidance and options suitable for different fitness objectives. 3. The system should provide recommendations or guidance based on the user's input to help set realistic and achievable fitness goals.
Assumptions:	1. The user has basic knowledge of fitness terminology and concepts. 2. The website has adequate security measures in place to protect user data.
Notes and Issues:	TBD: Maybe can include notifications or reminders to go to Track Workout Progress use case.

Use Case ID:	A5

Use Case Name:	Track Progress		
Created By:	Leon	Last Updated By:	Leon
Date Created:	08-02-2024	Date Last Updated:	24-02-2024

Actor:	User
Description:	This use case describes the process by which a user tracks their workout progress on the website.
Preconditions:	1. The user has successfully logged into their account. 2. The user has navigated to the workout tracking section of the website. 3. The user has previously set a fitness goal.
Postconditions:	1. The user's workout progress is tracked and recorded in the system. 2. The user's progress towards their fitness goal is monitored and displayed.
Priority:	Medium
Frequency of Use:	2 to 3 times a day, whenever the website is opened.
Flow of Events:	1. The user navigates to the workout tracking section of the website. 2. The system displays options for the user to input their workout details, including type of exercise, duration, intensity, etc. 3. The user selects the type of exercise they performed. 4. The user enters the details of their workout session, such as duration, intensity, distance covered, etc. 5. The user confirms the inputted workout details. 6. The system triggers the Workout Progress Tracker to validate user input and record the user's workout progress in the database. 7. The system retrieves the user's fitness goal from the database. 8. The system analyses the user's workout progress in relation to their fitness goal. 9. The system compares the user's current progress towards their fitness goal with their set goal. 10. The system displays the user's progress towards their fitness goal to the user.
Alternative Flows:	A5.AF-S2: Incorrect input of details 1. The website displays the message “Incorrect input!”. 2. The input box turns red and the user is required to change the inputs accordingly.
Exceptions:	1. If the user encounters an error while tracking their workout progress: 1.1. The system displays an error message indicating the issue. 1.2. The user may attempt to refresh the page or try again later.
Includes:	NA

Special Requirements:	1. The system must ensure that only authorised users have access to track their workout progress. 2. The workout tracking page must provide options for various types of exercises and workout parameters. 3. The system should provide visualisation tools to display the user's workout progress over time. 4. The system should accurately track and display the user's progress towards their fitness goal.
Assumptions:	1. The user has basic knowledge of workout terminology and concepts. 2. The user has previously set a fitness goal on the website. 3. The website has adequate security measures in place to protect user data.
Notes and Issues:	TBD: Implement features to analyse workout progress in relation to their goals

Use Case ID:	A6		
Use Case Name:	Track Calories		
Created By:	Leon	Last Updated By:	Leon
Date Created:	08-02-2024	Date Last Updated:	24-02-2024

Actor:	User
Description:	This use case describes the process by which a user tracks their daily calorie intake and receives information on their maintenance calories, as well as the recommended calorie intake for weight gain or loss, on the website.
Preconditions:	1. The user has successfully logged into their account. 2. The user has navigated to the calorie tracking section of the website.
Postconditions:	1. The user's daily calorie intake is tracked and recorded in the system. 2. The user receives information on their maintenance calories, as well as the recommended calorie intake for weight gain or loss.
Priority:	Medium
Frequency of Use:	2-3 times a day, before each meal
Flow of Events:	1. The user navigates to the calorie tracking section of the website. 2. The system displays the user's daily calorie intake tracker. 3. The system triggers the Profile Management Controller to retrieve the user's profile information, including age, gender, weight, height, activity level, and fitness goals. 4. The system calculates the user's maintenance calories based on their profile information. 5. The system displays the user's maintenance calories to the user. 6. The user enters the details of their daily food intake, including meals and snacks consumed. 7. The system calculates the total calorie intake based on the user's input and triggers the Food Database Interface to retrieve calorie information for different foods. 8. The system compares the user's total calorie intake with their maintenance calories. 9. The system provides feedback to the user on whether they have reached their maintenance calories, exceeded them, or are below them. 10. If the user has specific weight gain or loss goals: 10.1. The system calculates the recommended calorie intake for achieving those goals. 10.2. The system displays the recommended calorie intake for weight gain or loss to the user.
Alternative Flows:	A6.AF-S3: System could not retrieve profile information 1. The system triggers the Error Handling Controller to display “Please input your profile information first!”. 2. A button pops up to direct the user to input their information in the Manage Personal Profile use case. A6.AF-S6: Incorrect input

	1. The system triggers the Error Handling Controller to display the message “Incorrect input!”. 2. The input box turns red, and the user is required to change the inputs accordingly. A6.AF-S6: User unsure of calorie intake 1. The system prompts the user to add in foods consumed. 2. The system retrieves information from the database on different foods and their calories. 3. The resultant calorie count is shown to the user and added to the calorie intake of the user accordingly. 4. The user gets a final calorie intake after inputting foods consumed for that day, and the normal flow continues.
Exceptions:	1. If the user encounters an error while tracking their daily calorie intake: 1.1. The system displays an error message indicating the issue. 1.2. The user may attempt to refresh the page or try again later.
Includes:	NA
Special Requirements:	1. The system must ensure that only authorised users have access to track their daily calorie intake. 2. The calorie tracking page must provide options for entering details of various food items and their calorie content. 3. The system should provide accurate calculations for maintenance calories and recommended calorie intake for weight gain or loss. 4. The system should provide nutritional information for common food items to assist users in tracking their calorie intake accurately.
Assumptions:	1. The user has basic knowledge of nutrition and calorie tracking concepts. 2. The user has provided accurate profile information, including age, gender, weight, height, activity level, and fitness goals. 3. The website has adequate security measures in place to protect user data.
Notes and Issues:	NA

Use Case ID:	A7
Use Case Name:	Manage Personalised Workout Plan

Created By:	Leon	Last Updated By:	Leon
Date Created:	08-02-2024	Date Last Updated:	24-02-2024

Actor:	User
Description:	This use case describes the process by which a user accesses their personalised workout plan on the website.
Preconditions:	1. The user has successfully logged into their account. 2. The user has navigated to the section of the website where personalised workout plans are available.
Postconditions:	1. The user gains access to their personalised workout plan. 2. The user can view, follow, and track their workouts as per the plan.
Priority:	Medium
Frequency of Use:	Regularly
Flow of Events:	1. The user navigates to the section of the website where personalised workout plans are available. 2. The system triggers the User Input Controller to retrieve the user's profile information, including age, gender, weight, height, fitness level, and fitness goals. 3. Based on the user's profile information, the Workout Plan Controller generates a personalised workout plan tailored to the user's needs and goals. 4. The system displays the personalised workout plan to the user, including details such as exercise routines, sets, reps, rest periods, and progression. 5. The user reviews the personalised workout plan to understand the exercises, schedule, and any specific instructions provided. 6. The user begins following the workout plan, completing the prescribed exercises, sets, and reps as per the schedule. 7. As the user progresses through the workout plan, they track their workouts and record their performance, such as weights lifted, reps completed, and any notes or feedback. 8. The system may provide feedback or suggestions based on the user's progress and performance, adjusting the workout plan accordingly to ensure continued progress towards their goals.
Alternative Flows:	A7.AF-S3: User wishes to create own workout plan 1. The user triggers the User Input Controller to search for workouts and presses the add button to add them into the workout plan. 2. The system displays the personalised workout plan to the user, including details such as exercise routines, sets, reps, rest periods, and progression. 3. The user reviews the personalised workout plan to understand the exercises, schedule, and any specific instructions provided. 4. The user begins following the workout plan, completing the prescribed exercises, sets, and reps as per the schedule. 5. As the user progresses through the workout plan, they track their workouts and record their performance, such as weights lifted, reps completed, and any notes or feedback.

	6. The system may provide feedback or suggestions based on the user's progress and performance, adjusting the workout plan accordingly to ensure continued progress towards their goals.
Exceptions:	1. If the user encounters an error while accessing or viewing their personalised workout plan: 1.1. The system displays an error message indicating the issue. 1.2. The user may attempt to refresh the page or try again later.
Includes:	NA
Special Requirements:	1. The system must ensure that only authorised users have access to their personalised workout plans. 2. The workout plan page must provide clear instructions and guidance on performing each exercise, suitable for users of different fitness levels. 3. The system should allow users to track their workouts and record their progress conveniently within the platform. 4. The system should provide options for users to customise or adjust their workout plans based on their preferences, capabilities, or changing goals.
Assumptions:	1. The user has basic knowledge of exercise techniques and terminology. 2. The user has provided accurate profile information, including age, gender, weight, height, fitness level, and fitness goals. 3. The website has adequate security measures in place to protect user data. 4. The user is able to follow instructions from the workout information.
Notes and Issues:	TBD: Notification system to workout

Use Case ID:	A8		
Use Case Name:	Access Recipe Book		
Created By:	Leon	Last Updated By:	

Date Created:	08-02-2024	Date Last Updated:	

Actor:	User
Description:	This use case describes the process by which a user accesses the recipe book feature on the website to view recommended recipes based on their fitness goals.
Preconditions:	1. The user has successfully logged into their account. 2. The user has navigated to the section of the website where the recipe book is available.
Postconditions:	1. The user gains access to the recipe book feature. 2. The user can view recommended recipes tailored to their fitness goals.
Priority:	Medium
Frequency of Use:	Regularly
Flow of Events:	Flow of Events: 1. The user navigates to the section of the website where the recipe book is available. 2. The system triggers the User Input Controller to retrieve the user's profile information, including age, gender, weight, height, fitness level, and dietary preferences. 3. Based on the user's profile information and fitness goals, the Recipe Book Controller generates a list of recommended recipes that align with their dietary preferences and nutritional requirements. 4. The system displays the list of recommended recipes to the user, including details such as recipe name, ingredients, cooking instructions, and nutritional information. 5. The user reviews the list of recommended recipes and selects a recipe of interest to view more details. 6. The system displays the detailed recipe information to the user, including ingredients, cooking instructions, nutritional breakdown, and any additional tips or notes. 7. The user may choose to save the recipe to their favorites, print it out, or follow the cooking instructions directly from the website.
Alternative Flows:	NA
Exceptions:	1. If the user encounters an error while accessing or viewing the recipe book: 1.1. The system displays an error message indicating the issue. 1.2. The user may attempt to refresh the page or try again later.
Includes:	NA
Special Requirements:	1. The system must ensure that only authorised users have access to the recipe book feature. 2. The recipe book page must provide options for filtering recipes based on dietary preferences, meal types, and other relevant criteria. 3. The system should prioritise recommending recipes that align with the user's fitness goals, such as weight loss, muscle gain, or overall health improvement. 4. The system should provide accurate nutritional information for each recipe, helping users make informed choices about their meals.

Assumptions:	1. The user has basic knowledge of cooking techniques and terminology. 2. The user has provided accurate profile information, including age, gender, weight, height, fitness level, and dietary preferences. 3. The website has adequate security measures in place to protect user data.
Notes and Issues:	TBD: Can consider incorporating options for customising recipes based on ingredient availability, dietary restrictions, or personal preferences.

Use Case ID:	A9		
Use Case Name:	Find Gym Location		
Created By:	Leon	Last Updated By:	Leon
Date Created:	08-02-2024	Date Last Updated:	25-02-2024

Actor:	User

Description:	This use case describes the process by which a user finds nearby gym locations on the website.
Preconditions:	1. The user has successfully logged into their account. 2. The user has navigated to the section of the website where gym locations are available.
Postconditions:	1. The user gains access to information about nearby gym locations and their respective capacities.
Priority:	Medium
Frequency of Use:	2-3 times a week depending on physical activity level
Flow of Events:	1. The user navigates to the section of the website where gym locations are available. 2. The system triggers the Gym Location Controller to retrieve the user's location information or prompts the user to enter their location. 3. The system queries the database for nearby gym locations based on the user's location. 4. The system displays a list of nearby gym locations to the user, including details such as gym name, address, operating hours, and current capacity. 5. The user reviews the list of nearby gym locations and selects a gym for more details. 6. The system displays additional information about the selected gym, such as amenities, available equipment, and any special offers. 7. The user may save selected gyms to the favourites tab.
Alternative Flows:	NA
Exceptions:	1. If the user encounters an error while finding gym locations: 1.1. The system displays an error message indicating the issue. 1.2. The user may attempt to refresh the page or try again later.
Includes:	Book Gym Slots
Special Requirements:	1. The system must ensure that only authorised users have access to gym location information. 2. The gym location page must provide options for filtering gym locations based on amenities, equipment, and other relevant criteria. 3. The system should provide accurate information about gym capacities in real-time.
Assumptions:	1. The user has basic knowledge of gym etiquette and facilities. 2. The user has provided accurate location information or granted permission for the system to access their location. 3. The website has adequate security measures in place to protect user data.
Notes and Issues:	TBD: Implement favourites tab

DISCONTINUED, not possible for us to do that with any API

Use Case ID:	A10		
Use Case Name:	Book Gym Slots		
Created By:	Leon	Last Updated By:	Leon
Date Created:	08-02-2024	Date Last Updated:	24-02-2024

Actor:	User
Description:	This use case describes the process by which a user books gym slots on the website.
Preconditions:	1. The user has successfully logged into their account. 2. The user has found a gym location they want to visit.
Postconditions:	1. The user successfully books a gym slot and reserves their spot for a workout session.
Priority:	Medium
Frequency of Use:	2-3 times a week depending on physical activity level
Flow of Events:	1. The user has already completed the "Find Gym Location" use case and selected a gym they want to visit. 2. The user selects a date and time for their workout session. 3. The system checks the availability of slots for the selected date and time at the chosen gym. 4. If slots are available, the system allows the user to proceed with the booking. 5. The system confirms the booking and reserves the slot for the user. 6. The system redirects the user to the booking page to complete the reservation process. 7. The user completes the booking by providing any necessary information and confirming the reservation.
Alternative Flows:	A10.AF-S4: No slots are available 1. The message is displayed “No slots available now!” to the user. 2. The message is displayed “Would you like to try other available gyms?” to the user and the user is redirected to Find Gym Location use case.
Exceptions:	1. If the user encounters an error while booking a gym slot: 1.1. The system displays an error message indicating the issue. 1.2. The user may attempt to refresh the page or try again later.
Includes:	NA
Special Requirements:	1. The system must ensure that only authorised users have access to book gym slots. 2. The booking process should be seamless and user-friendly, with clear instructions and prompts to guide the user through the process. 3. The system should provide accurate information about slot availability in real-time.
Assumptions:	1. The user has basic knowledge of gym booking procedures. 2. The user has completed the "Find Gym Location" use case to select a gym location. 3. The website has adequate security measures in place to protect user data.
Notes and Issues:	TBD: Feasibility in implement booking gym from the website itself or instead redirect the user to an app or website for booking.