<Supplement and Fitness Shop>

Use-Case Specification

Version <1.1>

Revision History

| **Date** | **Version** | **Description** | **Author** |
| --- | --- | --- | --- |
| 04/11/2024 | 1.0 | Overview of use-case | Hồ Trương Viết Long |
| 20/11/2024 | 1.1 | Add and adjust some use-case | Hồ Trương Viết Long |
|  |  |  |  |
|  |  |  |  |

Table of Contents

[**1. Use-case Model 4**](#_heading=h.gjdgxs)

[**2. Use-case Specifications 4**](#_heading=h.30j0zll)

[*2.1 Use-case: Sign in 4*](#_heading=h.1fob9te)

[*2.2 Use-case: Sign out 5*](#_heading=h.2et92p0)

[*2.3 Use-case: Sign up 6*](#_heading=h.3ibea8s9socv)

[*2.4 Use-case: Profile Editing 6*](#_heading=h.t3us7f6qxw5z)

[*2.5 Use-case: View Notification 7*](#_heading=h.4dqmy8avth98)

[*2.6 Use-case: View Product 8*](#_heading=h.580sdy4f9id7)

[*2.7 Use-case: Product Search 8*](#_heading=h.pisqx9yfidx7)

[*2.8 Use-case: Product Filter 9*](#_heading=h.oxvosxt2p29n)

[*2.9 Use-case: Add item to cart 9*](#_heading=h.ngpdqif6adu)

[*2.10 Use-case: View / Edit Cart 10*](#_heading=h.f6b45w6dsj8m)

[*2.11 Use-case: Checkout 10*](#_heading=h.ahh5i874kogz)

[*2.12 Use-case: Payment 11*](#_heading=h.94f355j0z4q6)

[*2.13 Use-case: Coupon Wallet 12*](#_heading=h.lwujf7j8qcmd)

[*2.14 Use-case: Payment with Coupon 12*](#_heading=h.5a2wm6ovb2a4)

[*2.15 Use-case: View Order History 13*](#_heading=h.c4j10kzegg5w)

[*2.16 Use-case: Product Review / Rating 14*](#_heading=h.dbz8ig165nh5)

[*2.17 Use-case: Add Favorite Products List 14*](#_heading=h.7f7918ux19pv)

[*2.18 Use-case: View Favorite Products List 15*](#_heading=h.efi2pidqyji)

[*2.19 Use-case: Hire Personal Trainer 15*](#_heading=h.oh9uusnceo5q)

[*2.20 Use-case: Rate Personal Trainer 16*](#_heading=h.lduhw0tqigl4)

[*2.21 Use-case: Forum Participation 17*](#_heading=h.pk18yc9h5qsw)

[*2.22 Use-case: Post Question / Answer in Forum 17*](#_heading=h.36kilmyyjwzy)

[*2.23 Use-case: Service / Product Uploading 18*](#_heading=h.cew5xj6dmq5i)

[*2.24 Use-case: Manage Inventory 19*](#_heading=h.9hex6bqctnbf)

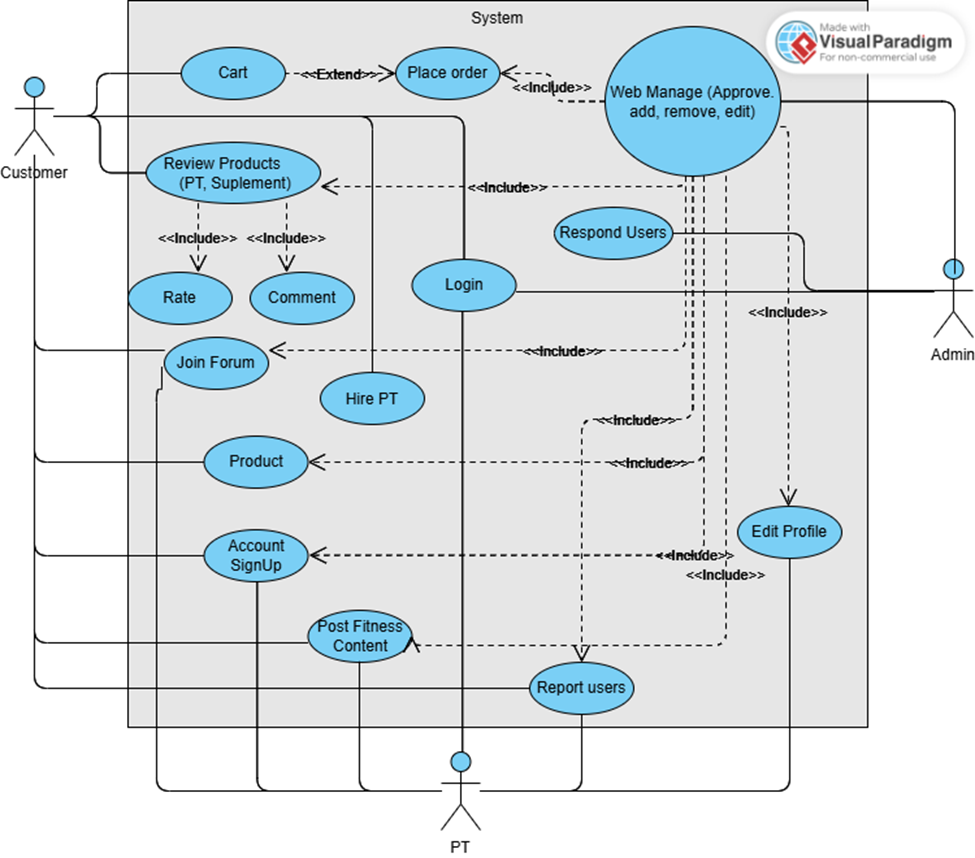
[*2.25 Use-case: Manage Orders 20*](#_heading=h.azw8dr1rt7ct)

[*2.26 Use-case: User Management 20*](#_heading=h.fwozxs6748kh)

[*2.27 Use-case: Manage Site Content 21*](#_heading=h.psjxejltd498)

[*2.28 Use-case: Support Chat 22*](#_heading=h.hcffn7c1lydb)

# Use-case Model



# Use-case Specifications

## Use-case: Sign in

| Use case Name | Sign in |
| --- | --- |
| Brief description | This use case describes how users sign in to the system. |
| Actors | Customer, Administrator, Personal Trainer, Vendor |
| Basic Flow | 1. User successfully accesses the website. 2. User clicks the ‘Sign in’ button. 3. System displays the sign-in page. 4. User inputs their username and password. 5. User clicks the sign-in button on the sign-in page. 6. System checks if the account exists and verifies credentials. 7. System redirects user to the homepage with their validated account. |
| Alternative Flows | **Alternative flow 1: Click Enter**   1. In Step #5, user can press "Enter" instead of clicking the sign-in button.     **Alternative flow 2: Shopping cart has already had the product**   1. If the account does not exist, or if username or password is incorrect, the system displays a message: “Wrong username or password.”   **Alternative flow 3: Blank Username or Password**   1. If user leaves username or password blank, they cannot press "Enter" or click the sign-in button. |
| Pre-conditions | User must have access to the website.  User must have a registered account. |
| Post-conditions | User logs in successfully and can browse products and interact with other features based on their permissions. |

## Use-case: Sign out

| Use case Name | Sign out |
| --- | --- |
| Brief description | This use case describes how users sign out. |
| Actors | Customer, Administrator, Personal Trainer, Vendor |
| Basic Flow | 1. User clicks the ‘Sign out’ button. 2. System logs user out and stops providing some features for a validated account. 3. Web gets back to the default condition. |
| Alternative Flows | **Alternative flow 1: Not logout when disconnect**   1. System will not log user out if user’s network has been corrupted |
| Pre-conditions | Users must have logged in to their accounts |
| Post-conditions | Log out successfully |

## Use-case: Sign up

| Use case Name | Sign up |
| --- | --- |
| Brief description | This use case describes how to sign up new customer account |
| Actors | Guest |
| Basic Flow | 1. Guests successfully access our website 2. Guest click sign up button 3. System show the sign up page 4. User will have to input required information 5. User click sign up button to create new account 6. System check if user information is valid 7. System show “Sign up successfully” 8. System moves back to sign in page and let the guest sign in |
| Alternative Flows | **Alternative flow 1: Not fully filled information**   1. In 4# if user does not fully fill information, the system will send a red notification below the box that is missing data.   **Alternative flow 2: Invalid character**   1. In 4#, if user does not use UTF-8 encoding characters in their information, user will require user to make a change to data.   **Alternative flow 3: Already existing username**   1. In 6#, if the system checked and found that username had already existed, the system would require user to make a change to username.   **Alternative flow 4: The password and confirm password do not match**   1. If 6# if the system checks the password and confirms the password does not match, the system will send a notification to user to refill the password information. |
| Pre-conditions | User must access our website and does not have a customer account |
| Post-conditions | User has a customer account and full access to our website’s features |

## Use-case: Profile Editing

| Use case Name | Profile Editing |
| --- | --- |
| Brief description | This use case describes how users edit their profiles. |
| Actors | Customer, Personal Trainer, Vendor |
| Basic Flow | 1. User clicks avatar button at top right corner 2. System moves to “Profile Editing” page 3. User makes changes 4. User clicks the save button 5. System notice saves successfully |
| Alternative Flows | **Alternative flow 1: Choose picture from device**   1. In 3# if users choose to change profile picture 2. System will let users upload their picture from the computer   **Alternative flow 2: Not filling name**   1. In 3# if users choose change name, but they do not fill fully, then when users click save 2. System will notice about “Have a blank information”   **Alternative flow 3: Change password**   1. In 3# if users choose change password 2. System will move to change password page 3. User will have to fill in current password 4. User will have to fill in new password 5. User will have to fill in confirmed password 6. System check if new password and confirmed pass are matched 7. User click save button.   **Alternative flow 4: Not save change**   1. In 4# if user does not want to save the changes, they can click cancel 2. System will not save changes |
| Pre-conditions | Users must have logged in to their accounts |
| Post-conditions | Users successfully save their new profile. |

## Use-case: View Notification

| Use case Name | View Notification |
| --- | --- |
| Brief description | This use case describes how users can see special notifications. |
| Actors | Customer, personal trainer |
| Basic Flow | 1. Users click on a belt button 2. System shows a container about new notifications 3. Users click on a new notification 4. System moves them to “Notification” on page 5. Users can read fully |
| Alternative Flows | **Alternative flow 1: View notification from “More”**   1. In 3# if user clicks “more”, the system will move them to the “Notification” page |
| Pre-conditions | Users must have logged in to their accounts |
| Post-conditions | Users can see special notifications. |

## Use-case: View Product

| Use case Name | View Product |
| --- | --- |
| Brief description | This use case describes how users view the product on a website. |
| Actors | Customer, Personal Trainer, Vendor, Administrator |
| Basic Flow | 1. User clicks the “Home” button at the upper of the page. 2. System will show some products. 3. User can view the products. |
| Alternative Flows | **Alternative flow 1: User can not find product** |
| Pre-conditions | Users must have logged in to their accounts |
| Post-conditions | User can view the product of the website |

## 

## Use-case: Product Search

| Use case Name | Product Search |
| --- | --- |
| Brief description | This use case describes how users search for specific products using the search bar by entering keywords. |
| Actors | Customer, Personal Trainer |
| Basic Flow | 1. User accesses the website and navigates to the product catalog or search bar. 2. User enters keywords in the search bar. 3. System processes the search query and displays relevant products matching the keywords. |
| Alternative Flows | **Alternative flow 1: No Matching Results**   1. If no products match the search or filter criteria, the system displays a “No results found” message and suggests similar categories or popular items. |
| Pre-conditions | User has access to the product catalog page with a search bar. |
| Post-conditions | Users successfully find a relevant product.  Users may add selected items to the cart or wishlist for future purchase. |

## Use-case: Product Filter

| Use case Name | Product Search and Filter |
| --- | --- |
| Brief description | This use case describes how users apply filters to refine product results within the product catalog. |
| Actors | Customer, Personal Trainer |
| Basic Flow | 1. User navigates to the product catalog page. 2. User selects one or more filter categories (e.g., product type, brand). 3. System displays products that meet the filter criteria. 4. Users can add additional filters (e.g., fitness goal, price range, rating) to refine the results further. 5. System updates the displayed products according to the selected filters. |
| Alternative Flows | **Alternative flow 1: No Matching Results**   1. If no products match the selected filters, the system displays a “No results found” message and suggests removing some filters or exploring popular items. |
| Pre-conditions | User has access to the product catalog page with filter options. |
| Post-conditions | Users successfully narrow down results to find products.  Users may add selected items to the cart or wishlist for future purchase. |

## Use-case: Add item to cart

| Use case Name | Add item to cart |
| --- | --- |
| Brief description | This use case describes how users can add products to their shopping cart. |
| Actors | Customer, Personal Trainer |
| Basic Flow | 1. User browses the product catalog and selects a product to view its details 2. System displays the product information, including available quantity and price. 3. User selects the desired quantity. 4. User clicks the "Add to Cart" button. 5. System confirms the addition by showing an “Added to Cart” notification. 6. System updates the cart icon or cart summary to reflect the new item and quantity. |
| Alternative Flows | **Alternative flow 1: Out of Stock**   1. At #2, if the product is out of stock, the system shows an “Out of Stock” message, and user cannot add it to the cart. |
| Pre-conditions | User is logged into the system.  The selected product is available in stock. |
| Post-conditions | User's cart is updated with the selected item and quantity.  Users can proceed to view the cart, continue shopping, or move to checkout. |

## Use-case: View / Edit Cart

| Use case Name | View and Edit Cart |
| --- | --- |
| Brief description | This use case describes how users can view items in their shopping cart, make adjustments to item quantities, or remove items before proceeding to checkout. |
| Actors | Customer, Personal Trainer |
| Basic Flow | 1. User clicks on the “Cart” icon or “View Cart” option from the navigation menu. 2. System displays a list of items currently in the cart, showing product names, quantities, prices, and a total cost summary. 3. Users can update the quantity of any item by selecting a new quantity or entering a desired number. 4. User clicks the "Update Cart" button to save changes, and the system recalculates the total price. 5. Users can also remove items by clicking a “Remove” or “Delete” button next to the unwanted product. 6. System updates the cart, reflecting the new quantities, removed items, and adjusted total cost. |
| Alternative Flows | **Alternative flow 1: Attempted Over-Limit Quantity**   1. At #3, if user tries to enter a quantity higher than the available stock, the system alerts them and suggests a valid quantity..   **Alternative flow 2: Empty Cart**   1. If the cart has no items, the system displays a message like “Your cart is empty” and suggests popular products or categories for shopping. |
| Pre-conditions | User has added one or more items to the cart.  The cart icon or summary is accessible from user’s current page. |
| Post-conditions | The cart accurately reflects user’s selected products and quantities.  Users can proceed to checkout with the updated cart contents or continue shopping. |

## Use-case: Checkout

| Use case Name | Checkout |
| --- | --- |
| Brief description | This use case describes how users finalize their purchases by confirming delivery details and initiating payment. |
| Actors | Customer, Personal trainer |
| Basic Flow | 1. User clicks the “Checkout” button on the cart page. 2. System displays a summary of cart items, total cost, applicable taxes, and shipping fees. 3. User provides or confirms their delivery address. 4. User proceeds to payment by clicking the "Continue to Payment" button. |
| Alternative Flows | **Alternative flow 1: Delivery Address Issue**   1. At #3, if the delivery address is incomplete or invalid, the system prompts the user to correct the information.   **Alternative flow 2: Cart Changes During Checkout**   1. If items in the cart become out of stock during the checkout process, the system notifies the user, removes the unavailable items, and updates the cart before allowing the user to proceed. |
| Pre-conditions | User has items in their cart.  User is logged in.  Delivery and payment options are available. |
| Post-conditions | User successfully proceeds to the payment phase with valid cart and delivery information. |

## Use-case: Payment

| Use case Name | Payment |
| --- | --- |
| Brief description | This use case describes how users complete payment for their purchase to finalize the order. |
| Actors | Customer, personal trainer |
| Basic Flow | 1. After proceeding from the checkout, the system displays available payment options (e.g., credit card, PayPal). 2. User selects a payment method. 3. User enters their payment details (e.g., card number, expiration date, CVV). 4. User clicks “Confirm Payment” to initiate the transaction. 5. System processes the payment: If successful, an order confirmation is generated with a unique order ID. |
| Alternative Flows | **Alternative flow 1: Payment Failure**   1. At #5, if the payment fails, the system displays an error message. 2. User is prompted to re-enter payment information or select a different payment method. |
| Pre-conditions | User has valid cart items and delivery information.  System supports the selected payment method. |
| Post-conditions | Payment is successfully processed.  The order is placed, and the user receives a confirmation via email and account details. |

## Use-case: Coupon Wallet

| Use case Name | Coupon Wallet |
| --- | --- |
| Brief description | This use-case allows users to view and manage their discount coupons. |
| Actors | Customer, personal trainer |
| Basic Flow | 1. User navigates to the Profile page. 2. System displays the "Coupon Wallet" section, listing:  * *Available Coupons:* Coupons that are valid and can be applied to purchases. * *Unavailable Coupons*: Expired or invalid coupons.  1. User selects a coupon to view its details, including terms, eligible products, and expiration date. 2. During checkout, the user can choose to apply an available coupon for discounts. |
| Pre-conditions | User is logged in.  System has coupons assigned to the user. |
| Post-conditions | Users can view their coupon details and apply them during checkout. |

## Use-case: Payment with Coupon

| Use case Name | Payment with Coupon |
| --- | --- |
| Brief description | This use case describes how users apply coupons during payment to avail discounts and complete their purchase. |
| Actors | Customer, personal trainer |
| Basic Flow | 1. After proceeding from the checkout, the system displays available payment options (e.g., credit card, PayPal) and an option to apply coupons. 2. System checks the user's coupon wallet for available coupons. 3. User selects a coupon to apply. 4. System validates the coupon and recalculates the total amount. 5. User selects a payment method. 6. User clicks “Confirm Payment” to finalize the transaction. 7. System processes the payment |
| Alternative Flows | **Alternative flow 1: Invalid Coupon**   1. At #4, if the coupon is invalid (e.g., expired or not applicable), the system displays an error message. 2. User can either select a valid coupon or proceed without one.   **Alternative flow 2: Payment Failure**   1. At #7, if the payment fails (e.g., insufficient funds, invalid details), the system displays an error message. 2. User is prompted to re-enter payment details or select another payment method. |
| Pre-conditions | User has valid cart items and delivery information.  System recognizes and validates the selected coupon. |
| Post-conditions | Payment is successfully processed with the applied discount.  The order is placed, and the user receives a confirmation via email and account details. |

## Use-case: View Order History

| Use case Name | View Order History |
| --- | --- |
| Brief description | This use case describes how users can view their past purchases, including details such as order dates, items purchased, delivery status, and total amounts. |
| Actors | Customer, personal trainer |
| Basic Flow | 1. User navigates to the “Order History” or “My Orders” section from their account dashboard. 2. System displays a list of past orders, with each entry showing the order ID, order date, status (e.g., "Processing," "Shipped," "Delivered"), and total amount. 3. User selects an order to view more details. 4. System shows detailed information about the selected order, including items purchased, individual item prices, quantities, shipping address, payment method, and delivery tracking information if applicable. |
| Alternative Flows | **Alternative flow 1: No Previous Orders**   1. If the user has no previous orders, the system displays a message like “You have no order history” and may suggest products or offer a shortcut to start shopping.   **Alternative flow 2: Order Detail Unavailable**   1. If an order detail is unavailable (due to a canceled order or a system issue), the system notifies the user and may suggest they contact customer support for further assistance. |
| Pre-conditions | User is logged into their account.  System has stored data on the user’s past purchases. |
| Post-conditions | User views the details of any selected past order.  User can track current delivery status or contact support if needed. |

## Use-case: Product Review / Rating

| Use case Name | Product Review / Rating |
| --- | --- |
| Brief description | This use case describes how users can submit reviews and ratings for products they have purchased. |
| Actors | Customer, personal trainer |
| Basic Flow | 1. User navigates to the product page of an item they have purchased. 2. System displays the product details, including existing reviews and average ratings. 3. User clicks on the “Write a Review” or “Rate this Product” button. 4. System prompts the user to enter their rating (e.g., 1 to 5 stars) and write a review. 5. User submits the review and rating. 6. System validates the submission (checking for profanity, length, etc.) and saves the review to the product’s review database. 7. System updates the product page to reflect the new review and recalculates the average rating. |
| Alternative Flows | **Alternative flow 1: Review Submission Error**   1. If the review does not meet the validation criteria (e.g., too short, inappropriate content), the system notifies the user with an error message and requests corrections. |
| Pre-conditions | User is logged into their account.  User must have purchased the product to leave a review. |
| Post-conditions | User’s review and rating are stored in the system. |

## Use-case: Add Favorite Products List

| Use case Name | Add Favorite Products List |
| --- | --- |
| Brief description | This use case describes how users add an item to their FPL. |
| Actors | Customer, personal trainer |
| Basic Flow | 1. User clicks the “Heart” icon at the bottom right corner of the post. 2. System notify user and save this post to user’s “Favorite Products List”. 3. Users can see the “Heart” icon will be highlighted which means this post has been added to “Favorite Products List”. |
| Alternative Flows | **Alternative flow 1: Double click “Heart” icon**   1. In #1 If the user clicks the “Heart” icon again, the user will cancel the emoticon for this post. 2. System remove this post from “Favorite Products List”. |
| Pre-conditions | User is logged into their account.  Users must have accessed to the post page |
| Post-conditions | Products are added to FPL. |

## Use-case: View Favorite Products List

| Use case Name | View Favorite Products List |
| --- | --- |
| Brief description | This use case describes how to show list which users have liked. |
| Actors | Customer |
| Basic Flow | 1. Users click the avatar of their accounts in the top right corner. 2. System will show the list of account’s features such as sign out, edit information, security account, View FPL, etc. 3. Users click the “View FPL” option. 4. System moves to the “Favorite Products List” page and shows the favorite items of the user's account added. |
| Alternative Flows | **Alternative flow 1: Remove item in Favorite Products List**   1. If the users click the “Heart” emotion of a post,this post will be removed after users refresh page or access to this page next time.   **Alternative flow 2: No favorite product found**   1. Users access this page. 2. System show notification “You have not loved any post. Let pick some favorite items”. 3. System show tutorial of “Love” items for adding them to Favorite Products List. |
| Pre-conditions | User is logged into their account. |
| Post-conditions | User can see their Favorite Products List and will recommend purchasing those items in future. |

## Use-case: Hire Personal Trainer

| Use case Name | Hire Personal Trainer |
| --- | --- |
| Brief description | This use case describes how users can hire personal trainers through the platform. |
| Actors | Customer |
| Basic Flow | 1. User navigates to the “Personal Trainer” section of the website. 2. System displays a list of available personal trainers, including their profiles, qualifications, specialties, and client ratings. 3. User uses filters (e.g., location, expertise, availability) to refine the list of trainers. 4. User selects a trainer to view more detailed information, including their biography, services offered, pricing, and customer reviews. 5. User clicks the “Hire” button option on the trainer’s profile. 6. System prompts the user to select the type of training session (e.g., one-time session, package deal) and schedule a date and time. 7. User confirms the booking and makes the payment. 8. System sends a confirmation email with the session details to the user and the selected trainer. |
| Alternative Flows | **Alternative flow 1: Trainer Not Available**   1. If the trainer is not available at the selected time, the system notifies the user and prompts them to choose a different time slot or select another trainer. |
| Pre-conditions | User is logged into their account.  System has a list of available personal trainers with up-to-date availability. |
| Post-conditions | User successfully hires a personal trainer and receives confirmation of the session.  The trainer’s schedule is updated to reflect the new booking. |

## Use-case: Rate Personal Trainer

| Use case Name | Rate Personal Trainer |
| --- | --- |
| Brief description | This use case describes how users can submit ratings and feedback for personal trainers. |
| Actors | Customer, Personal Trainer |
| Basic Flow | 1. User logs into their account and navigates to the “My Bookings” section of the website. 2. System displays a list of personal training sessions the user has completed. 3. User selects a specific session with a personal trainer they wish to rate. 4. System prompts the user to enter a rating (e.g., 1 to 5 stars) and write comments about their experience with the trainer. 5. User submits the rating and review. 6. System validates the submission (checking for inappropriate content, length, etc.) and saves the rating and review in the database. 7. System updates the personal trainer's profile to reflect the new rating and displays the average rating based on all user feedback. |
| Alternative Flows | **Alternative flow 1: Submission Error**   1. If the review does not meet the validation criteria (e.g., inappropriate language, insufficient length), the system notifies the user with an error message and requests corrections. |
| Pre-conditions | User is logged into their account.  User must have completed a session with the personal trainer they wish to rate. |
| Post-conditions | User’s rating and review are stored in the system.  The personal trainer’s profile is updated with the new rating, contributing to their overall feedback score. |

## Use-case: Forum Participation

| Use case Name | Forum Participation |
| --- | --- |
| Brief description | This use case describes how users can participate in discussions within the forum |
| Actors | Customer, Administrator, Personal Trainer |
| Basic Flow | 1. User logs into their account and navigates to the “Forum” section. 2. System displays a list of available discussion threads and topics. 3. User selects a thread they wish to contribute to or reads existing posts for context. |
| Alternative Flows | **Alternative flow 1: Report Inappropriate Content**   1. If a user encounters an inappropriate post, they can report it to the administrator, triggering a review process. |
| Pre-conditions | User is logged into their account.  The forum must be operational and accessible. |
| Post-conditions | User can access the Forum. |

## Use-case: Post Question / Answer in Forum

| Use case Name | Post Questions / Answers in Forum |
| --- | --- |
| Brief description | This use case describes how users can post questions and provide answers in the forum. |
| Actors | Customer, Administrator |
| Basic Flow | 1. User logs into their account and navigates to the “Forum” section. 2. System displays a list of existing discussion threads and a prompt to create a new post. 3. To ask a question, the user clicks on the “Ask a Question” button. 4. User enters a descriptive title for their question and provides detailed content in the text area. 5. User submits the question, and the system validates the content for appropriateness and relevance. 6. System saves the question as a new thread in the forum. 7. Other users can view the question and post answers by clicking the “Reply” button beneath the question. 8. Users providing answers follow the reply submission process, similar to the participation flow. |
| Alternative Flows | **Alternative flow 1: Search for Question / Answer**   1. Before posting a question, the user may search the forum using keywords to see if the question has already been answered, preventing duplicate posts.   **Alternative flow 2: Edit or Delete Question / Answer**   1. User may edit or delete their question within a specific time frame after posting if they wish to change the content or remove it altogether. |
| Pre-conditions | User is logged into their account.  The forum must be operational and accessible. |
| Post-conditions | User's question is saved in the forum and is visible to all members.  Other users can respond to the question. |

## Use-case: Service / Product Uploading

| Use case Name | Service / Product Uploading |
| --- | --- |
| Brief description | This use case describes how sellers can upload new products or services to the platform. |
| Actors | Personal Trainer, Administrator, Vendor |
| Basic Flow | 1. Seller logs into their account and navigates to the “Product Management” section. 2. System displays options for uploading new products or services. 3. Seller selects “Upload New Product/Service.” 4. System presents a form to fill in details such as product name, description, price, category, and images. 5. Seller enters all required information and uploads relevant images. 6. Seller reviews the entered information for accuracy. 7. Seller clicks the “Submit” button to upload the product or service. 8. System validates the data, saves the new product/service in the database, and displays a confirmation message. |
| Alternative Flows | **Alternative flow 1: Incomplete Information**   1. If the seller submits the form without filling in all required fields, the system prompts them to complete the missing information before submitting.   **Alternative flow 2: Image Upload Failure**   1. If the image files exceed size limits or are in unsupported formats, the system notifies the seller of the issue and allows for a retry. |
| Pre-conditions | Seller is logged into their account.  The seller must have the necessary permissions to upload products or services. |
| Post-conditions | The new product or service is successfully uploaded.  The inventory is updated in the system. |

## Use-case: Manage Inventory

| Use case Name | Manage Inventory |
| --- | --- |
| Brief description | This use case describes how Vendors can manage their inventory levels for each product listed on the platform. |
| Actors | Vendor |
| Basic Flow | 1. Vendor logs in to their account. 2. Vendor navigates to the Inventory Management section. 3. Vendor views the list of products they have uploaded. 4. Vendor updates stock quantities, sets notifications for low inventory, or removes products from listing if out of stock. 5. System saves updates and reflects them in product availability for customers. |
| Alternative Flows | **Alternative flow 1: Remove Products**   1. If a product is no longer available, the vendor or administrator can mark it as discontinued or delete it from the inventory.   **Alternative flow 2: Low Stock Alerts**   1. If a product’s stock falls below a set threshold, the system sends a low stock alert to the vendor/administrator to restock. |
| Pre-conditions | Vendor is logged into their account.  Products exist in the system to be managed or updated. |
| Post-conditions | Inventory details are updated. |

## Use-case: Manage Orders

| Use case Name | Manage Orders |
| --- | --- |
| Brief description | This use case describes how administrators can view, update, and manage customer orders. |
| Actors | Administrator |
| Basic Flow | 1. Administrator logs into their account and navigates to the “Order Management” section. 2. System displays a list of all customer orders, including order ID, customer name, order status, and date. 3. Administrator selects an order to view its details. 4. System displays detailed information about the selected order, including the items ordered, quantities, customer shipping information, and payment status. 5. Administrator can update the order status and add any notes. 6. Administrator can issue refunds or modify the order details. 7. Administrator saves any changes made to the order. 8. System confirms the changes and updates the order status. |
| Alternative Flows | **Alternative flow 1: Order Not Found**   1. If the administrator searches for an order that does not exist, the system notifies them that no matching order was found. |
| Pre-conditions | Administrator is logged into their account. |
| Post-conditions | The order status is updated in the system, and the customer is notified of any changes. |

## Use-case: User Management

| Use case Name | User Management |
| --- | --- |
| Brief description | This use case describes how administrators can manage user accounts. |
| Actors | Administrator |
| Basic Flow | 1. Administrator logs into their account and navigates to the “User Management” section. 2. System displays a list of all registered users, including usernames and account statuses. 3. Administrator can search for a specific user or apply filters to narrow down the list. 4. Administrator selects a user to view their profile. 5. System displays the user’s details, including their order history and account status. 6. Administrator can update user information, reset passwords, or change account statuses. 7. Administrator can delete user accounts. 8. System confirms the changes and updates the user information. |
| Alternative Flows | **Alternative flow 1: User Not Found**   1. If the administrator searches for a user that does not exist, the system notifies them that no matching user was found. |
| Pre-conditions | Administrator is logged into their account. |
| Post-conditions | User account information is updated, and changes are reflected in the system. |

## Use-case: Manage Site Content

| Use case Name | Manage Site Content |
| --- | --- |
| Brief description | This use case describes how administrators can manage the content displayed on the website, including product descriptions, blog posts, and other informational pages. |
| Actors | Administrator |
| Basic Flow | 1. Administrator logs into their account and navigates to the “Site Content Management” section. 2. System displays a list of all editable content. 3. Administrator selects a piece of content to edit. 4. System displays the current content in an editor interface. 5. Administrator makes changes to the content and saves the updates. 6. System confirms that the changes have been saved and updated on the live site. |
| Alternative Flows | **Alternative flow 1: Content Preview**   1. Administrator can preview changes before saving them to ensure accuracy. |
| Pre-conditions | Administrator is logged into their account. |
| Post-conditions | Updated content is reflected on the website. |

## Use-case: Support Chat

| Use case Name | Support Chat |
| --- | --- |
| Brief description | This use case describes how users can engage in real-time chat with customer support representatives. |
| Actors | Customer, Vendor, Administrator, Personal Trainer |
| Basic Flow | 1. User navigates to the “Support Chat” feature on the website. 2. User submits the information to initiate the chat. 3. System connects the user to an available support representative. 4. User and support representatives engage in real-time conversation to address questions or concerns. 5. User can end the chat session at any time. 6. System records the chat for future reference and quality assurance. |
| Alternative Flows | **Alternative flow 1: No Representatives Available**   1. If no support representatives are available, the system notifies the user and offers an option to leave a message. |
| Pre-conditions | User is logged into their account. |
| Post-conditions | User receives assistance. |