



FPT UNIVERSITY

KOI CARE SYSTEM

Software Requirement Document

Lecturer: Tôn Thất Hoàng Minh

Class: SE1856

Member:

Lê Đức Lộc (Leader)

Nguyễn Trung Kiên

Lưu Thế Vinh

Nguyễn Văn Minh Thoại

Đào Công An Phước

– Ho Chi Minh, October 2024 –

Table Of Contents

Definition and Acronyms	5
I. Introduction	5
1. Product Background	5
2. Existing Systems	5
2.1 Koi Fish Pond Management System at Home	5
3. Business Opportunity	7
4. Software Product Vision	7
5. Major Features	7
6. Limitations and Exclusions	9
7. Project Scope	9
7.1. Software Introduction	9
7.2. Purpose	9
7.3. Links to User and Business Goals	9
7.4. Feature Scope	10
II. Overall Description	11
1. Product Overview	11
2. Business Process	16
3. Check Payment (Credit & Invoicing)	17
4. Payment by VNPay (Credit & Invoicing)	17
5. Is Payment Successful? (Credit & Invoicing)	17
6. Cancel Order (Shop)	17
7. Prepare Invoice (Credit & Invoicing)	17
8. Send Invoice (Credit & Invoicing)	17
9. Receive Invoice (Member)	17
10. Delivery (Shop)	17
11. Receive Orders (Member)	17
III. User Requirements	19
1. Actors	19
2. Use Cases	22
2.1 Diagram	22
2.2 Descriptions	23
3. Business Rules	96
IV. Functional Requirements	98

1. System Functional Overview	98
1.1 Screen Flow	98
1.2 Screen Descriptions	98
1.3 Screen Authorization	104
1.4 Non-Screen Functions	109
2. Data Requirements	110
2.1 Logical Data Model	110
2.2 Data Dictionary	110
2.3 Reports	125
2.4 State transition diagram	136
3. Home Page	137
4. Login Page	138
5. Dashboard Member	142
6. My Koi	144
7. My Pond	155
8. Pond Log	158
9. Water Parameters	162
10. Reminders	167
10.2 FR-23: Add Reminders	168
11. Recommendations	170
12. News and Blogs	173
13. Cart and Checkout	177
FR-28: Cart	177
FR-29: Checkout	178
14. Food Calculator	180
15. Salt Calculator	182
16. Statistics	183
FR-32: Statistics - View Statistics About Koi Ponds and Koi Fish	183
17. Member information	186
FR-33: Member Information - View and Update Member Profile	186
18. Dashboard Shop	187
19. Manage User	189
FR-35: Manage User - View List of Users	189
FR-36: View User Profile	191
FR-37: View User's Payment Details	192
FR-38: View User's Order Details	194
20. Manage Order	196
FR-39: Manage Order	196
21. Manage Payment	198
FR-40: Manage Payment - View and Manage Payment Details	198
22. Manage Blog Tag	201
FR-41: Manage Blog Tag - List and Manage Blog Tags	201
FR-42: Create Tag	202
FR-43: Update Tag - Update Existing Blog Tags	204

23. Manage Blog	205
FR-44: Manage Blog	205
FR-45: Create Blog	207
FR-46: Update Blog	208
FR-47: View Blog	210
24. Manage Supplier	211
FR-48: Manage Supplier - List Suppliers	211
FR-49: Create Supplier	213
FR-50: Edit Supplier	214
25. Manage Promotion	216
FR-51: Manage Promotion - List Promotions	216
FR-52: Create Promotion	218
FR-53: Edit Promotion	220
26. Manage Category	221
FR-54: Manage Category	221
FR-55: Create Category	223
FR-56: Edit Category	224
27. Manage Image	226
FR-57: Manage Image	226
FR-58: Create Image	227
FR-59: Edit Image - Edit Product Image Details	229
27. Manage Product	230
FR-60: Manage Product	230
FR-61: Create Product	232
FR-62: Edit Product	234
28. Dashboard Admin	236
FR-63: Dashboard Admin	236
FR-64: Promotion Management	237
FR-65: Manage Shop Staff	238
FR-66: Create Shop Staff Account	239
FR-67: Change Shop Account Password	240
FR-68: Upgrade Premium for Member	241
FR-69: About	242
V. Non-Functional Requirements	243
1. External Interface Requirements	243
1.1 User Interfaces	243
1.2 Software Interfaces	244
1.3 Hardware Interfaces	246
1.4 Communications Interfaces	246
2. Quality Attributes	247
2.1 Usability	247
2.2 Performance	247
2.3 Security	247
2.4 Safety	247

2.5 Availability	247
2.6 Reliability	248
2.7 Design Constraints	248

Definition and Acronyms

Acronym	Definition
BA	Business Analysis
BR	Business Rule
ERD	Entity Relationship Diagram
GUI	Graphical User Interface
PM	Project Manager
SDD	Software Design Description
SPMP	Software Project Management Plan
SRS	Software Requirement Specification
UAT	User Acceptance Test
UC	Use Case
API	Application Program Interface

I. Introduction

1. Product Background

The Koi Care System is designed to streamline the management of Koi ponds for home-based users. Koi care requires regular monitoring of water parameters, feeding schedules, and overall fish health. Without an efficient management system, hobbyists may struggle to maintain optimal conditions for their fish. Manual methods are prone to errors and do not provide real-time tracking of important metrics such as water quality, fish growth, and pond conditions.

2. Existing Systems

2.1 Koi Fish Pond Management System at Home

Description: A Koi Fish Pond Management System is a software solution that allows homeowners to efficiently manage their Koi fish ponds and monitor the health and growth of their Koi fish, ensuring optimal living conditions.

System Actors:

- **Admin:** Oversees the entire system, sets permissions, and analyzes reports.
- **Shop Staff:** Uses the system for customer service, point-of-sale, and order management.
- **Customers:** Can access online portals for account management, purchase history, and loyalty programs (optional).

Features:

- **Pond Management:** Allows users to manage detailed information about their Koi pond, including the pond's name, images, size, depth, volume, number of drainage outlets, pump power, and other characteristics.
- **Koi Fish Management:** Provides detailed management of each Koi fish, including attributes like name, image, body shape, age, size, weight, gender, breed, origin, selling price, and the pond they are housed in. Users can update the growth metrics of the fish over time and view trends with growth charts.
- **Water Parameters Monitoring:** Tracks water quality parameters for each pond at various measurement times (e.g., temperature, salt levels, pH, O₂, NO₂, NO₃, PO₄). The system can recommend necessary adjustments if any values fall outside ideal conditions.
- **Food Calculation:** Calculates the required amount of food for each Koi fish based on its growth stage, ensuring proper nutrition at all times.
- **Salt Calculation:** Determines the amount of salt needed for each pond to maintain ideal water conditions for Koi fish health.
- **Product Ordering:** Allows users to order products to improve water parameters or treat Koi fish health conditions.
- **News & Blog:** Provides users access to articles, tips, and shared experiences related to Koi fish care.

Pros:

- Ensures a healthy environment for Koi fish by tracking water conditions and fish growth.
- Simplifies care with automated calculations for food and salt requirements.
- Helps prevent costly health issues by recommending timely interventions.
- Conveniently integrates e-commerce for purchasing needed products.

Cons:

- Requires initial setup and training for effective use.
- Dependence on software may result in challenges during technical disruptions.
- Reliance on technology can lead to disruptions in case of system outages.
- Data security is crucial and requires proper measures to prevent breaches.

Link: <https://koicontrol.com>

3. Business Opportunity

The Koi fish industry is growing as customers seek professional and effective solutions to ensure the health and beauty of their Koi fish. A Koi Fish Pond Management System can revolutionize how pond owners manage their ponds and care for their Koi, benefiting both the fish and the owners. By optimizing the process of monitoring water parameters, managing fish health, and providing necessary recommendations, this system addresses current challenges and offers a competitive advantage in the Koi fish care market.

4. Software Product Vision

Our Vision is to transform Koi pond management into a:

- Seamless
- Efficient
- Customer-centric experience

With our software, pond owners and caretakers will be able to easily track the growth of Koi fish, manage water parameters, and monitor the health of the pond with precision, ensuring smooth and accurate care routines. The system supports both manual input and automated sensor integration, simplifying the monitoring process, while real-time data calculations guarantee transparency and accuracy.

Maintenance schedules and feeding plans will be effortlessly managed, and the system will handle inventory for supplies like food and water treatment products efficiently, fostering customer trust and satisfaction by ensuring optimal care for Koi fish.

It will support personalized care recommendations based on each fish's growth stage, improving fish health and longevity. Additionally, it will track the performance of pond maintenance activities, monitor each pond's water quality, and generate reports for both the pond owner and caretakers, providing valuable insights for operational improvements.

The system will include customizable settings for care routines, comprehensive tracking of fish development, and product management capabilities for fish and pond-related supplies.

Ultimately, our vision is to empower Koi fish owners to manage their ponds at peak efficiency, delivering exceptional care and fostering a healthy, thriving Koi population. Our software will align with the needs of both hobbyists and professionals, not only meeting current demands but also anticipating and adapting to future trends in the Koi fish industry, setting a new standard in pond management.

5. Major Features

Shop:

- Dashboard Shop
- Manager User
- Manage News
 - News
 - Tag
- Manage Shop
 - Supplier
 - Promotion
 - Category
 - Images
 - Product
- Manage Report
 - Manage Order
 - Manage Payment
- Profile Shop

Admin:

- Dashboard Admin
- News
- Manage Shop:
 - Supplier
 - Promotion
 - Product
- Manage Users
 - Shop
 - Customer
- Manage Report
 - Manage Order
 - Manage Payment

Member:

- Dashboard Member
- My Koi

- My Pond
- Water Parameters
- Reminders
- Recommendations
- Food Calculator
- Salt Calculator
- Statistics
- News And Blogs
- About
- Shop Carts
- Wishlist
- Payment
- Profile

6. Limitations and Exclusions

LE-1: The system will not support integration with external water quality sensors or automation systems for feeding and maintenance. It is designed primarily for manual data entry and monitoring by the user.

LE-2: In-depth reports on fish breeding trends, financial costs, or advanced analytics for fish health and pond conditions are not included. Detailed data analysis to identify the best-performing ponds, fish, or equipment is not planned.

7. Project Scope

7.1. Software Introduction

The Koi Fish Pond Management Software is an application designed to meet the needs of pond owners in managing fish health, pond water quality, and overall pond maintenance, with an integrated feature for recommending products related to Koi care and pond maintenance.

7.2. Purpose

- Optimize the care and management process for Koi ponds, reducing the time required for monitoring and maintenance.
- Accurately track the growth, health, and well-being of Koi fish over time.
- Record and manage water quality parameters and other pond details for optimal fish care.
- Provide detailed reports on pond conditions, fish development, and offer recommendations for improvements.

- Recommend related products (food, water treatments, health supplements) for purchase directly from the shop to enhance pond conditions and fish care.

7.3. Links to User and Business Goals

- **For users (pond owners/caretakers):**
 - Minimize errors in monitoring water quality and fish care.
 - Save time in tracking pond conditions and calculating food and treatment needs.
 - Easily monitor the growth and health history of each Koi fish.
 - Conveniently purchase recommended products for pond and fish care based on real-time data.
 -
- **For businesses:**
 - Tightly control pond conditions and fish health to avoid losses.
 - Improve customer service by offering detailed and personalized care recommendations.
 - Build effective maintenance and care strategies based on comprehensive data analysis.
 - Increase revenue by promoting the sale of recommended products tailored to the needs of the pond and fish.

7.4. Feature Scope

Shop

- **Dashboard:** Monitor total users, total orders, total payments, etc. Top products, general charts about the above sections.
- **Manage Users:** Monitor the status of each user in the system and their current information.
- **Manage Payment:** Oversee and process payments, transactions.
- **Manage Orders:** Oversee the order process, from creation to fulfillment.
- **Manage Product:** Add, update, view and delete product information (name, images, brand, price,...).
- **Manage Category for Product:** Add, update, view and delete categories (name) for product.
- **Manage Image for Product:** Add, update, view and delete image for product.
- **Manage Supplier:** Add, update, view and delete Supplier information.
- **Manage Promotions:** Create, update, view and manage promotional campaigns and apply to products.
- **Manage News:** Create blogs sharing experiences, treatments, and current news about koi fish.
- **Manage Tag for News:** Create, view, update, delete Tag (Name, Description) for News.

Admin

- **Statistics Dashboard:** Monitor order to day, premium user, subscribes today, revenue shop, total users, total orders, total payments, etc. Top products, general charts about the above sections.
- **Manage News :** View, delete News .

- **Manage Supplier:** Add, update, view and delete Supplier information.
- **Manage Promotions:** Oversee and manage promotional activities.
- **Manage Product:** Add, update, view and delete product information (name, images, brand, price,...).
- **Manage Shop:** Create staff shop, view update and delete account shop.
- **Manage Users:** View all users and user details (view payment details, order details, profile,...).

Users

- **Dashboard Member :** Monitor ponds, koi, water parameter, reminders, orders currently available, overview (recent transaction, payment details, report).
- **Create, View, Update, Delete Koi:** Handle information my koi in pond, views information, growth history, update information and growth history, add remarks for koi.
- **Create, View, Cancel, Save Sell Orders:** Handle sales orders, including creation, viewing, cancellation, and saving.
- **Create, View, Update, Delete ponds :** Handle information about my pond, view information, create, update, view, delete pond log and notification issue in the aquarium and take it to the product page.
- **Create, View, Update, Delete water Reminders:** Handle information about reminders, reminders notifications.
- **Create, View, Update, Delete water parameters:** Handle information about parameters in the pond, display color according to input field standards.
- **Create, View, Cancel, Save Buy Orders:** Manage buy orders, including creation, viewing, cancellation, and saving.
- **View recommendations:** View products available in the shop, add to wish list, add shop cart, and payment, feedback in detailed products.
- **Food Calculator:** Take the total weight of fish in the pond, perform the calculation formula and return the result as food per day.
- **Salt Calculator:** Take the volume of water in the lake, perform the calculation formula and return the result as the amount of water that needs to be changed and the amount of salt that needs to be added.
- **Statistics:** View a chart of changes in water components over time and growth history belong to koi.
- **News and blog:** View blogs sharing fish care experiences and new information about koi fish.

II. Overall Description

1. Product Overview

Koi-Care-System emerges as a sophisticated software solution designed to streamline the intricate operations involved in koi fish care and retail. It serves as a comprehensive platform, providing a suite of tools meticulously crafted for inventory management, customer relationship management (CRM), order processing, and insightful reporting. The primary goal of **Koi-Care-System** is to enhance operational efficiency, improve customer service, and deliver invaluable insights to promote business growth.

Target Users

The target users of **Koi-Care-System** encompass various stakeholders within the koi care ecosystem:

Shop:

- **User-Friendly Dashboard:** Monitors total users, orders, payments, and top products efficiently.
- **Effective Inventory Management:** Displays real-time stock levels and product availability.
- **Order Management:** Oversees the entire order process from creation to fulfillment, ensuring timely delivery.
- **Promotion Management:** Enables the application of promotional campaigns, discounts, and loyalty programs.
- **Mobile Access:** Facilitates servicing customers anytime, anywhere through a mobile-friendly interface.

Admin:

- **Statistics Dashboard:** Monitors key metrics such as daily orders, premium user subscriptions, total revenue, and more through customizable reports.
- **User Management:** Controls and configures user accounts, ensuring appropriate access levels based on roles.
- **Supplier Management:** Adds, updates, views, and deletes supplier information for effective stock management.
- **Promotional Oversight:** Oversees and manages promotional activities, ensuring alignment with business goals.
- **Analytics and Reporting:** Supports data-driven decision-making through comprehensive reporting tools and business intelligence.

User (Customer):

- **Personalized Dashboard:** Monitors pond health, koi information, and current orders while providing insights into recent transactions and payment details.
- **Koi Management:** Handles information about koi in ponds, including growth history and health updates.
- **Order Handling:** Manages buy and sell orders, allowing users to create, view, and cancel orders effortlessly.
- **Water Parameter Management:** Tracks water parameters and receives reminders for necessary maintenance tasks.
- **Product Recommendations:** Accesses curated product suggestions based on individual preferences and pond conditions.

Data Security and Compliance:

- **Robust Security Protocols:** Implements stringent security measures to protect user data and ensure compliance with industry standards.

- **System Maintenance and Support:** Provides tools for system upkeep, updates, and responsive customer support.

Reporting and Analytics:

- **Performance Tracking:** Delivers insights into system usage and operational performance, aiding continuous improvement.

A. Primary Users

Shop

The shop plays a crucial role in selling koi care products, processing orders, and managing promotions. Key functionalities include:

- **Product Management:** Add, update, and manage koi care products, including food, equipment, and pond accessories.
- **Order Management:** Process and fulfill customer orders for koi care products. Manage order statuses, apply promotions, and track deliveries.
- **Supplier Management:** Manage relationships with suppliers, add new products, and ensure stock availability.
- **Payment Handling:** Accept and process payments for products and services.

Admin

The admin oversees the overall system, managing user accounts, security settings, and business operations. Key functionalities include:

- **User Management:** Control and configure user accounts with role-based access permissions.
- **System Settings:** Configure system-wide settings for shop, orders, and user operations.
- **Security and Compliance:** Ensure data security and compliance with privacy policies and legal standards.
- **Reporting and Analytics:** Generate reports on sales, user activity, product performance, and revenue.

User (Customer)

The user is the primary customer interacting with the system to buy koi care products, manage their ponds, and monitor their koi. Key functionalities include:

- **Dashboard:** Monitor pond health, water parameters, and orders for koi care products.
- **Product Purchases:** Browse, add to cart, and purchase koi care products such as food, equipment, and pond treatments.
- **Order Tracking:** View and manage orders, including cancellation and tracking delivery status.
- **Pond and Koi Management:** Input and track data related to koi health, pond conditions, and water quality.

B. System:

The **Koi-Care-System** serves as a central hub for managing interactions between users, shops, and the system itself, facilitating the processing of transactions and the management of koi care. The data flows represent how information is exchanged within the system to ensure efficient operations.

Orders

Orders in the system are categorized into buy and sell orders, specifically related to koi care products such as food, treatments, and pond supplies.

- **Sell Orders**

- **Creation:** The shop creates sell orders when customers purchase koi care products. Each order includes product details, quantities, and any applicable promotions.
- **Data Flow:**
 - **Input:**
 - Product details (name, SKU, price).
 - Customer information (name, address, contact).
 - Order quantity and applied promotions.
 - **Output:**
 - Order confirmation sent to the customer.
 - Updated inventory levels reflecting sold products.
 - Shipping and tracking details provided once the order is dispatched.

- **Buy Orders**

- **Creation:** Users create buy orders to purchase koi care products from the shop. This includes selecting products, specifying quantities, and entering payment information.
- **Data Flow:**
 - **Input:**
 - Selected product details (name, price).
 - Quantity and shipping information.
 - Payment method (credit card, PayPal).
 - **Output:**
 - Order confirmation and receipt emailed to the user.
 - Updates to the order status (pending, shipped, completed).
 - Inventory updates reflecting new purchases.

Payments

Payments are processed by the shop when users complete purchases of koi care products. This flow ensures secure and accurate transaction handling.

- **Data Flow:**

- **Input:**
 - Payment information (credit card details, billing address).
 - Order details (order ID, total amount).
 - Customer information (name, contact).

- **Output:**
 - Payment confirmation sent to the customer via email.
 - Transaction records stored for accounting and reporting purposes.
 - Receipt generation for user reference.

Reports

Admins generate reports on various aspects of the business, including sales performance, user activity, and inventory levels, to support decision-making.

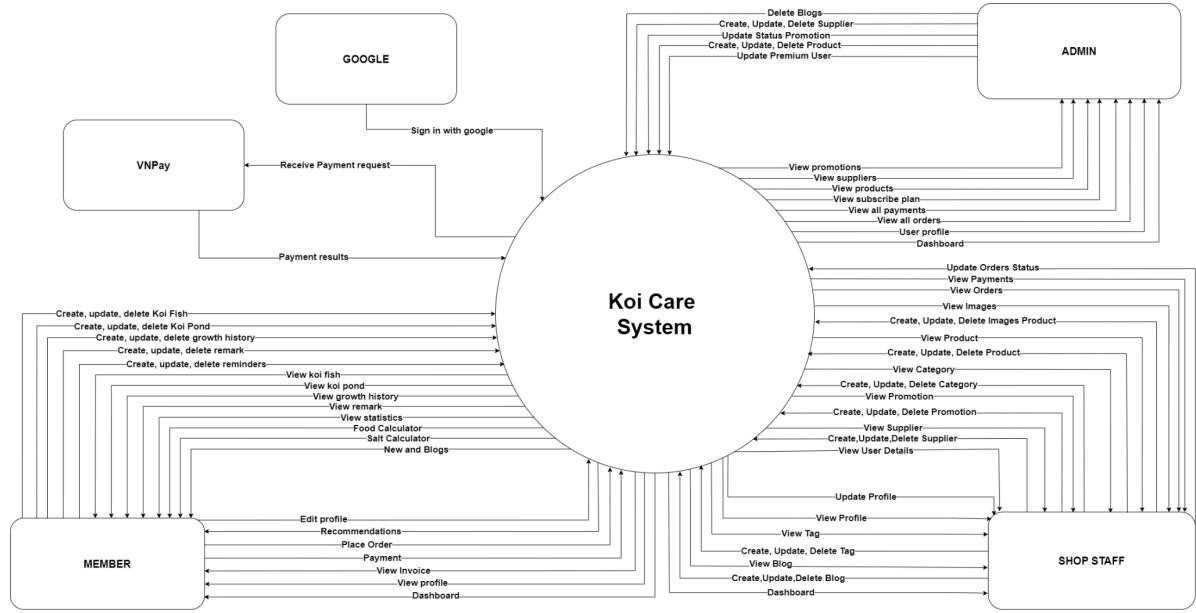
- **Data Flow:**
 - **Input:**
 - Sales data (total revenue, number of transactions).
 - User activity data (logins, order history).
 - Product performance metrics (best-selling items, stock levels).
 - **Output:**
 - Comprehensive reports for sales performance, user engagement, and inventory management.
 - Visual analytics (graphs and charts) for easy interpretation of data trends.
 - Insights and recommendations for marketing strategies based on sales data.

User Management

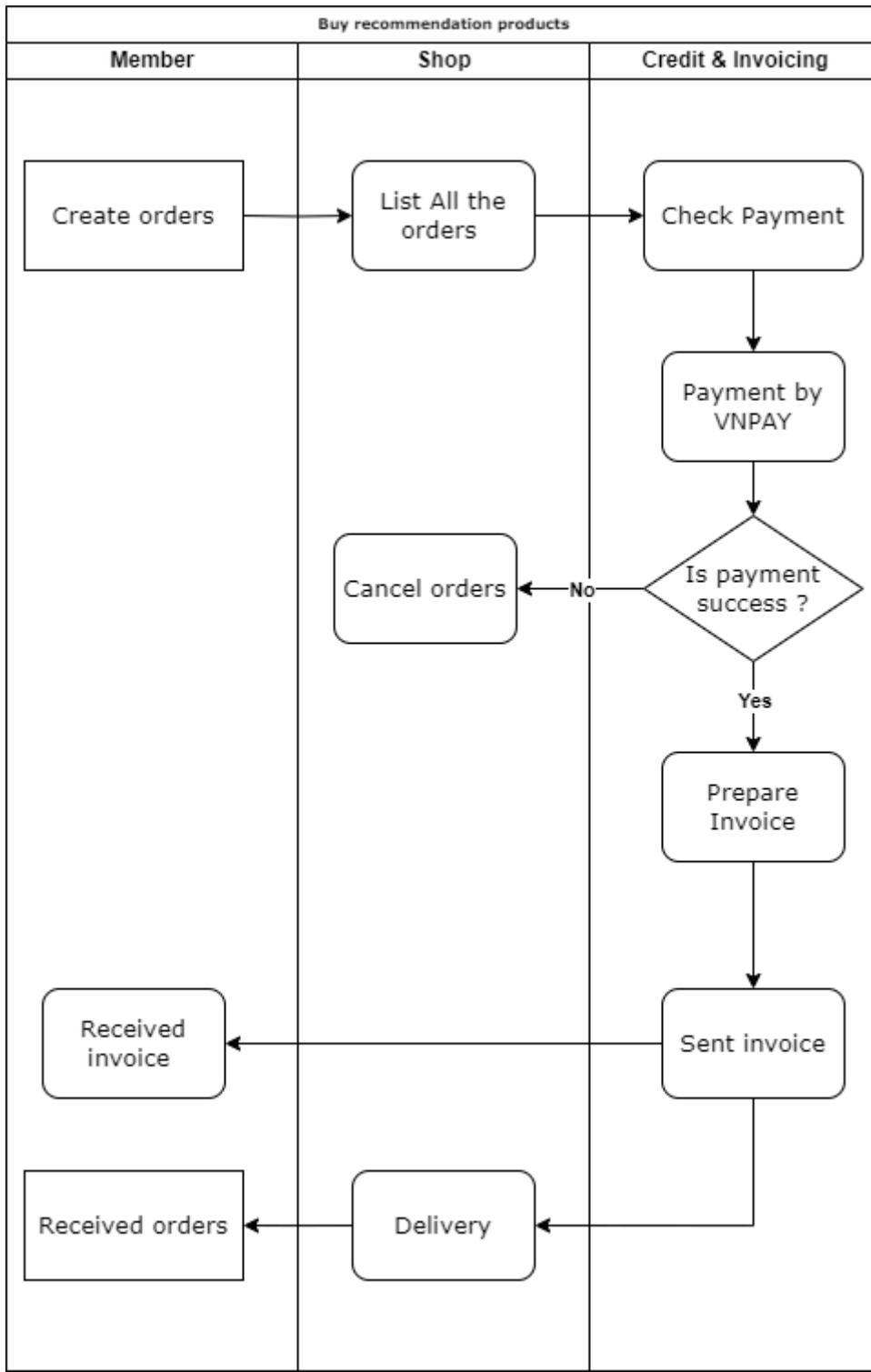
Admins manage user accounts, overseeing permissions and ensuring appropriate access to various functionalities based on roles.

- **Data Flow:**
 - **Input:**
 - User account details (name, email, role).
 - System configuration settings (security protocols, access levels).
 - Policy details (privacy policy, terms of service).
 - **Output:**
 - Creation or modification of user accounts and roles.
 - Notifications sent to users regarding account status changes.
 - Audit logs detailing user activities and access patterns for security compliance.

C. Context diagram:



2. Business Process



Process Steps and Descriptions for buy recommendation products:

1. Create Order (Member)

- The member initiates the order process by selecting recommendation products and adding them to their shopping cart. Once the products are added, the order creation is completed on the system.

2. List All the Orders (Shop)

- The shop system lists all the member's orders, allowing shop staff or the backend to review and monitor the orders that have been placed.

3. Check Payment (Credit & Invoicing)

- The credit and invoicing team checks the payment status for the order. They verify if the payment method has been chosen and if it is ready for processing.

4. Payment by VNPay (Credit & Invoicing)

- The member selects VNPay as the payment method, and the system redirects them to VNPay for processing the payment.

5. Is Payment Successful? (Credit & Invoicing)

- After processing the payment through VNPay, the system checks whether the payment was successful:
 - **Yes:** If the payment is successful, the process moves to the next step to prepare the invoice.
 - **No:** If the payment fails, the system cancels the order.

6. Cancel Order (Shop)

- If the payment is not successful, the shop system cancels the order and informs the member that the transaction did not go through.

7. Prepare Invoice (Credit & Invoicing)

- If the payment is successful, the credit and invoicing team prepares the invoice for the member's order. The invoice includes all details of the products purchased and the payment confirmation.

8. Send Invoice (Credit & Invoicing)

- The system sends the prepared invoice to the member, confirming that the order has been successfully placed and payment completed.

9. Receive Invoice (Member)

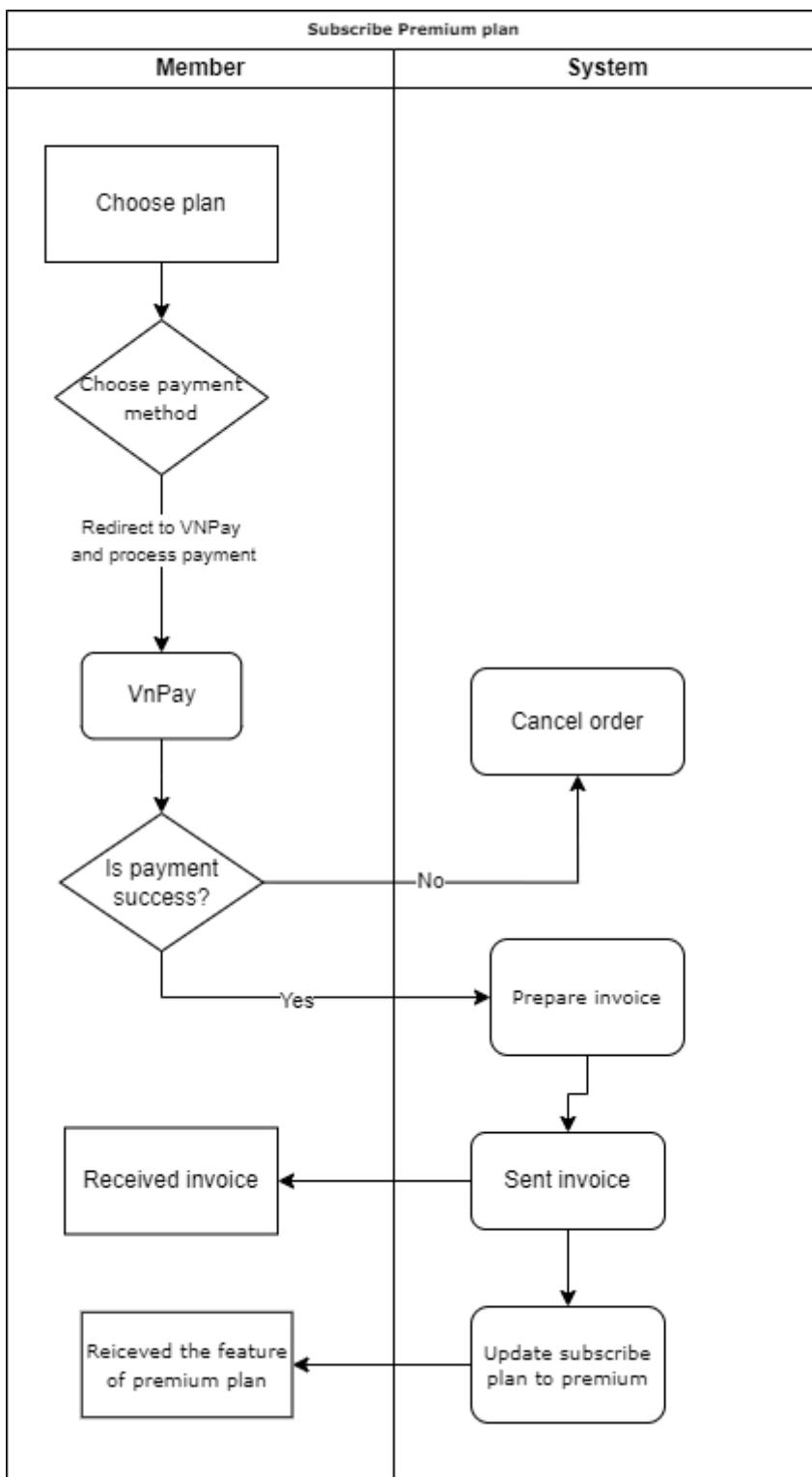
- The member receives the invoice through the system (via email or account) as confirmation of their successful purchase.

10. Delivery (Shop)

- The shop arranges the delivery of the ordered products to the member. The system updates the delivery status as the products are dispatched to the member's provided address.

11. Receive Orders (Member)

- The member receives the products they ordered, marking the completion of the purchase and delivery process. Any relevant warranties or loyalty cards are included with the delivery.



Process Steps and Descriptions for subscribe premium plan:

1. Choose Plan (Member)

- The member initiates the subscription process by selecting a premium plan they want to subscribe to.

2. Choose Payment Method (Member)

After selecting a plan, the member chooses their preferred payment method to complete the subscription.

3. Redirect to VNPay (System)

The system redirects the member to VNPay to handle the payment transaction.

4. Is Payment Successful? (System)

After VNPay processes the payment, the system checks whether the payment was successful:

- **Yes:** If the payment is successful, the process continues to prepare the invoice.
- **No:** If the payment fails, the system cancels the order.

5. Cancel Order (System)

If the payment is not successful, the system cancels the order and informs the member that the transaction did not go through.

6. Prepare Invoice (System)

If the payment is successful, the system prepares an invoice for the member's subscription to the premium plan.

7. Send Invoice (System)

The system sends the invoice to the member, confirming the successful payment and subscription.

8. Receive Invoice (Member)

The member receives the invoice, typically via email or in their account, confirming the subscription.

9. Receive Premium Features (Member)

The member is granted access to the features of the premium plan, marking the completion of the subscription process.

III. User Requirements

1. Actors

Primary Actor	Use Cases
Shop	<ol style="list-style-type: none">1. View Member2. Create Blog3. Update Blog4. Delete Blog5. View Blog6. Create Tag7. Update Tag8. Delete Tag9. View Tag10. View Product11. Create Product

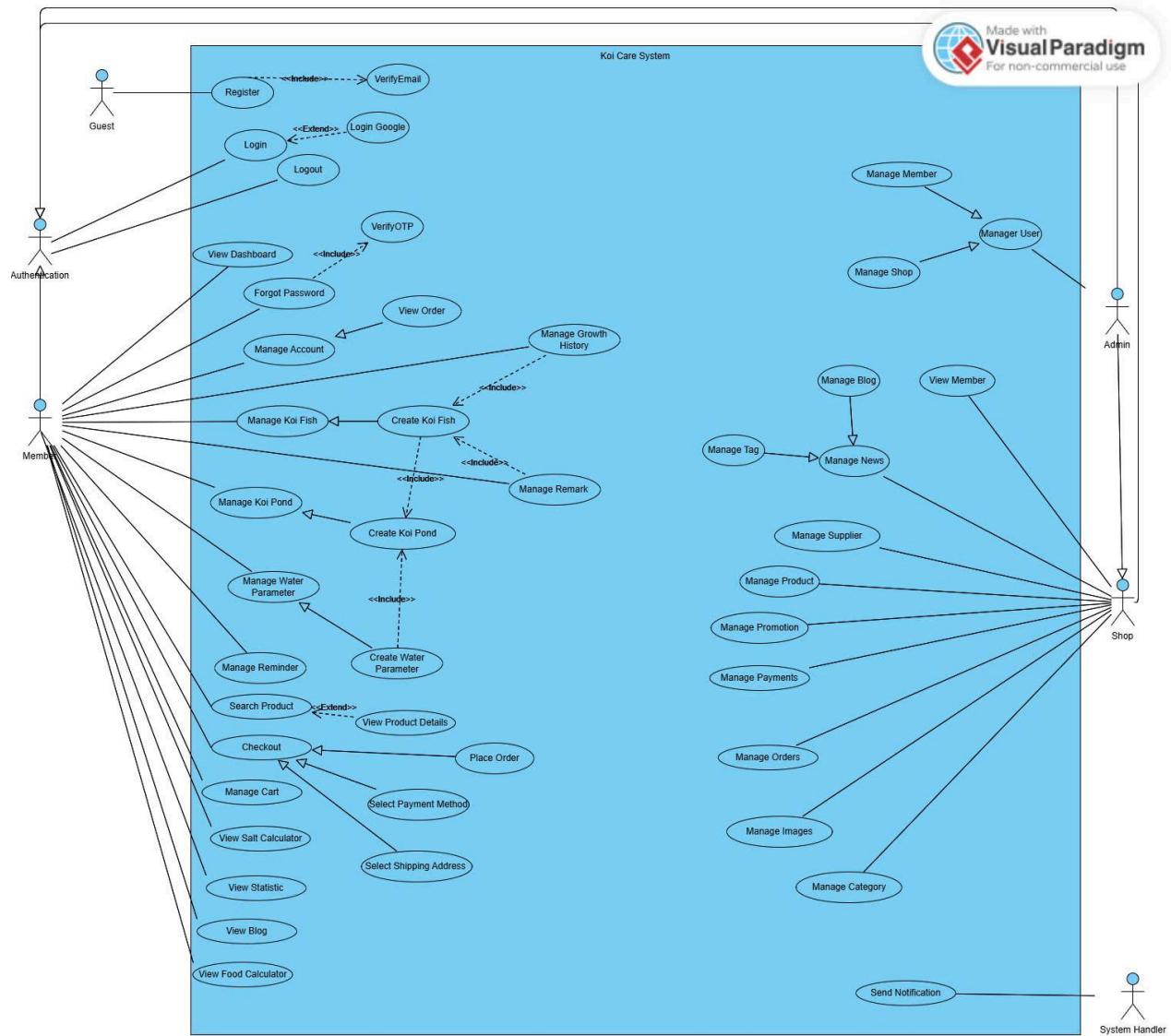
	<ul style="list-style-type: none"> 12. Update Product 13. Delete Product 14. Create Supplier 15. Update Supplier 16. Delete Supplier 17. View Supplier 18. Create Promotion 19. Update Promotion 20. View Promotion 21. Delete Promotion 22. View Payment 23. View Orders 24. Update Orders 25. Create Category 26. Update Category 27. Delete Category 28. View Category 29. View Dashboard
Member	<ul style="list-style-type: none"> 1. View Dashboard 2. Forgot Password 3. Verify 4. View Profile 5. View Order 6. Update Profile 7. Update Koi Fish 8. Create Koi Fish 9. Delete Koi Fish 10. View Koi Fish 11. Update Growth History 12. Create Growth History 13. Delete Growth History 14. View Growth History 15. Update Remark 16. Create Remark 17. Delete Remark 18. View Remark 19. Create Koi Pond 20. View Koi Pond 21. Update Koi Pond 22. Delete Koi Pond

	23. Update Water Parameter 24. View Water Parameter 25. Create Water Parameter 26. Delete Water Parameter 27. Update Reminder 28. Create Reminder 29. Delete Reminder 30. View Reminder 31. Search Product 32. View Product Details 33. Update Cart 34. View Cart 35. Add to Cart 36. Place Order for checkout 37. Select Payment method 38. Select Shipping Address 39. View Food Salt Calculator 40. View Statistic 41. View Blog
Admin	1. View Member 2. Create Blog 3. Update Blog 4. Delete Blog 5. View Blog 6. View Product 7. Create Product 8. Update Product 9. Delete Product 10. Create Supplier 11. Update Supplier 12. Delete Supplier 13. View Supplier 14. Create Promotion 15. Update Promotion 16. View Promotion 17. Delete Promotion 18. View Payment 19. View Orders 20. Update Orders 21. Update Member 22. Delete Member

- | | |
|--|---|
| | <p>23. View Staff
 24. Create Staff
 25. Update Staff
 26. Delete Staff</p> |
|--|---|

2. Use Cases

2.1 Diagram



2.2 Descriptions

ID and Name:	UC-01 View Member		
Created By:	Nguyễn Văn Minh Thoại	Date Created:	22/10/24
Primary Actor:	Shop	Secondary Actors:	Admin
Description:	Shop views the details of a member.		
Trigger:	Shop wants to view member information.		
Preconditions:	Shop must be logged in.		
Postconditions:	Member details are displayed.		
Normal Flow:	<ol style="list-style-type: none"> 1. Shop navigates to the member section. 2. Shop views the selected member's details. 		
Alternative Flows:	None		
Exceptions:	E1: Member does not exist. The system displays an error message.		
Priority:	Medium		
Frequency of Use:	Regular		
Business Rules:	BR-18		
Other Information:	None		
Assumptions:	None		

--	--

ID and Name:	UC-02 Create Blog		
Created By:	Nguyễn Văn Minh Thoại	Date Created:	22/10/24
Primary Actor:	Shop	Secondary Actors:	None
Description:	Shop creates a new blog post.		
Trigger:	Shop wants to create a new blog post.		
Preconditions:	Shop wants to create a new blog post.		
Postconditions:	Shop wants to create a new blog post.		
Normal Flow:	<ol style="list-style-type: none"> 1. Shop navigates to the blog creation page. 2. Shop enters the blog details and publishes the post. 		
Alternative Flows:	None		
Exceptions:	E1: Required blog information is missing. The system displays an error message.		
Priority:	Medium		
Frequency of Use:	Regular part of sales operations.		
Business Rules:	BR-15, BR-16		
Other Information:	None		
Assumptions:	None		

--	--

ID and Name:	UC-03: Update Blog		
Created By:	Nguyễn Văn Minh Thoại	Date Created:	22/10/24
Primary Actor:	Shop	Secondary Actors:	None
Description:	Shop updates an existing blog post.		
Trigger:	Shop wants to update a blog post.		
Preconditions:	Shop must be logged in.		
Postconditions:	Blog post is updated.		
Normal Flow:	<ol style="list-style-type: none"> 1. Shop navigates to the blog post. 2. Shop edits and saves the updated blog content. 		
Alternative Flows:	None		
Exceptions:	E1: Blog post does not exist. The system displays an error message.		
Priority:	Medium		
Frequency of Use:	Regular part of sales operations.		
Business Rules:	BR-15, BR-16		
Other Information:	None		
Assumptions:	None		

ID and Name:	UC-04 Delete Blog		
Created By:	Nguyễn Văn Minh Thoại	Date Created:	22/10/24
Primary Actor:	Shop	Secondary Actors:	Admin
Description:	Shop deletes an existing blog post.		
Trigger:	Shop deletes an existing blog post.		
Preconditions:	Shop must be logged in.		
Postconditions:	Shop deletes an existing blog post.		
Normal Flow:	<ol style="list-style-type: none"> 1. Shop navigates to the blog post. 2. Shop deletes the blog post. 		
Alternative Flows:	None		
Exceptions:	E1: Blog post does not exist. The system displays an error message.		
Priority:	Medium		
Frequency of Use:	Regular		
Business Rules:	BR-15		
Other Information:	None		
Assumptions:	None		

ID and Name:	UC-05: View Blog		
Created By:	Nguyễn Văn Minh Thoại	Date Created:	22/10/24
Primary Actor:	Shop	Secondary Actors:	Member,admin
Description:	Shop views a blog post.		
Trigger:	Shop wants to view a blog post.		
Preconditions:	Shop must be logged in.		
Postconditions:	Blog post is displayed.		
Normal Flow:	<ol style="list-style-type: none"> 1. Shop navigates to the blog section. 2. Shop views the blog post. 		
Alternative Flows:	None		
Exceptions:	E1: Blog post does not exist. The system displays an error message.		
Priority:	Medium		
Frequency of Use:	Regular		
Business Rules:	BR-16		
Other Information:	None		
Assumptions:	None		

ID and Name:	UC-06 Create Tag		
Created By:	Nguyễn Văn Minh Thoại	Date Created:	22/10/24
Primary Actor:	Shop	Secondary Actors:	None
Description:	Shop creates a new tag for categorizing content.		
Trigger:	Shop wants to create a tag.		
Preconditions:	Shop must be logged in.		
Postconditions:	A new tag is created.		
Normal Flow:	<ol style="list-style-type: none"> 1. Shop navigates to the tag creation page. 2. Shop enters the tag details and saves the tag. 		
Alternative Flows:	None		
Exceptions:	E1: Required tag information is missing. The system displays an error message.		
Priority:	Medium		
Frequency of Use:	Regular		
Business Rules:	BR-17		
Other Information:	None		
Assumptions:	None		

ID and Name:	UC-07 Update Tag		
Created By:	Nguyễn Văn Minh Thoại	Date Created:	22/10/24
Primary Actor:	Shop	Secondary Actors:	None
Description:	Shop updates an existing tag.		
Trigger:	Shop wants to update a tag.		
Preconditions:	Shop must be logged in.		
Postconditions:	Tag is updated.		
Normal Flow:	<ol style="list-style-type: none"> 1. Shop navigates to the tag section. 2. Shop updates the tag details and saves the changes. 		
Alternative Flows:	None		
Exceptions:	E1: Tag does not exist. The system displays an error message.		
Priority:	Medium		
Frequency of Use:	Regular		
Business Rules:	BR-17		
Other Information:	None		
Assumptions:	None		

ID and Name:	UC-08 Delete Tag		
Created By:	Nguyễn Văn Minh Thoại	Date Created:	22/10/24
Primary Actor:	Shop	Secondary Actors:	None
Description:	Shop wants to delete a tag.		
Trigger:	Shop wants to delete a tag.		
Preconditions:	Shop must be logged in.		
Postconditions:	Tag is deleted.		
Normal Flow:	<ol style="list-style-type: none"> 1. Shop navigates to the tag section. 2. Shop deletes the tag. 		
Alternative Flows:	None		
Exceptions:	E1: Tag does not exist. The system displays an error message.		
Priority:	Medium		
Frequency of Use:	Regular		
Business Rules:	BR-17		
Other Information:	None		
Assumptions:	None		

ID and Name:	UC-09: View Tag
--------------	------------------------

Created By:	Nguyễn Văn Minh Thoại	Date Created:	22/10/24
Primary Actor:	Seller	Secondary Actors:	None
Description:	Seller applies a customer's accumulated loyalty points to a purchase, verifying available points and determining the applicable discount.		
Trigger:	A customer wants to use accumulated points for a purchase.		
Preconditions:	<p>The seller must be logged into the system.</p> <p>Customer's accumulated points must be available.</p>		
Postconditions:	The customer's points are applied, and the total purchase amount is adjusted.		
Normal Flow:	<ol style="list-style-type: none"> 1. The seller verifies the customer's accumulated points. 2. The seller applies the points to the purchase. 3. The system adjusts the total purchase amount and updates the points balance. 		
Alternative Flows:	None		
Exceptions:	E1: Insufficient points. The system displays an error message.		
Priority:	Medium		
Frequency of Use:	Regular part of sales operations.		
Business Rules:	BR-17		
Other Information:	None		
Assumptions:	None		

--	--

ID and Name:	UC-10: View Product		
Created By:	Nguyễn Văn Minh Thoại	Date Created:	22/10/24
Primary Actor:	Shop	Secondary Actors:	Admin, Member
Description:	Shop views the details of a product.		
Trigger:	Shop wants to view product information.		
Preconditions:	Shop must be logged in.		
Postconditions:	Product details are displayed.		
Normal Flow:	<ol style="list-style-type: none"> 1. Shop navigates to the product section. 2. Shop views the product details. 		
Alternative Flows:	None		
Exceptions:	E1: Product does not exist. The system displays an error message.		
Priority:	High		
Frequency of Use:	Regular		
Business Rules:	BR-3		
Other Information:	None		
Assumptions:	None		

--	--

ID and Name:	UC-11: Create Product		
Created By:	Nguyễn Văn Minh Thoại	Date Created:	22/10/24
Primary Actor:	Shop	Secondary Actors:	Admin
Description:	Shop creates a new product for the inventory.		
Trigger:	Shop wants to add a new product to the inventory.		
Preconditions:	Shop must be logged in.		
Postconditions:	A new product is created.		
Normal Flow:	<ol style="list-style-type: none"> 1. Shop navigates to the product creation page. 2. Shop enters the product details and saves the product. 		
Alternative Flows:	None		
Exceptions:	E1: Required product information is missing. The system displays an error message.		
Priority:	Medium		
Frequency of Use:	Regular		
Business Rules:	BR-21, BR-27		
Other Information:	None		
Assumptions:	None		

--	--

ID and Name:	UC-12: Update Product		
Created By:	Nguyễn Văn Minh Thoại	Date Created:	22/10/24
Primary Actor:	Shop	Secondary Actors:	Admin
Description:	Shop updates an existing product in the inventory.		
Trigger:	Shop wants to update a product.		
Preconditions:	Shop must be logged in.		
Postconditions:	Product is updated.		
Normal Flow:	<ol style="list-style-type: none"> 1. Shop navigates to the product section. 2. Shop updates the product details and saves the changes. 		
Alternative Flows:	None		
Exceptions:	E1: Product does not exist. The system displays an error message.		
Priority:	Medium		
Frequency of Use:	Regular		
Business Rules:	BR-21, BR-27		
Other Information:	None		
Assumptions:	None		

--	--

ID and Name:	UC-13: Delete Product		
Created By:	Nguyễn Văn Minh Thoại	Date Created:	22/10/24
Primary Actor:	Shop	Secondary Actors:	Admin
Description:	Shop deletes an existing product from the inventory.		
Trigger:	Shop wants to delete a product.		
Preconditions:	Shop must be logged in.		
Postconditions:	Product is deleted.		
Normal Flow:	<ol style="list-style-type: none"> 1. Shop navigates to the product section. 2. Shop deletes the product. 		
Alternative Flows:	None		
Exceptions:	E1: Printer is not available. The system displays an error message.		
Priority:	High		
Frequency of Use:	Regular part of sales operations.		
Business Rules:	BR-21		
Other Information:	None		
Assumptions:	None		

--	--

ID and Name:	UC-14: Create Supplier		
Created By:	Nguyễn Văn Minh Thoại	Date Created:	22/10/24
Primary Actor:	Shop	Secondary Actors:	Admin
Description:	Shop creates a new supplier.		
Trigger:	Shop wants to add a new supplier.		
Preconditions:	Shop must be logged in.		
Postconditions:	A new supplier is created.		
Normal Flow:	<ol style="list-style-type: none"> 1. Shop navigates to the supplier creation page. 2. Shop enters the supplier details and saves the supplier. 		
Alternative Flows:	None		
Exceptions:	E1: Required supplier information is missing. The system displays an error message.		
Priority:	Medium		
Frequency of Use:	Regular		

Business Rules:	BR-21
Other Information:	None

Assumptions:	None
--------------	------

ID and Name:	UC-15: Update Supplier		
Created By:	Nguyễn Văn Minh Thoại	Date Created:	22/10/24
Primary Actor:	Shop	Secondary Actors:	Admin
Description:	Shop updates an existing supplier.		
Trigger:	Shop wants to update a supplier.		
Preconditions:	Shop must be logged in.		
Postconditions:	Supplier information is updated.		
Normal Flow:	<ol style="list-style-type: none"> 1. Shop navigates to the supplier section. 2. Shop updates the supplier details and saves the changes. 		
Alternative Flows:	None		
Exceptions:	E1: Supplier does not exist. The system displays an error message.		
Priority:	Medium		
Frequency of Use:	Regular		
Business Rules:	BR-21		
Other Information:	None		

Assumptions:	None
--------------	------

ID and Name:	UC-16: Delete Supplier		
Created By:	Nguyễn Văn Minh Thoại	Date Created:	22/10/24
Primary Actor:	Shop	Secondary Actors:	Admin
Description:	Shop views details of a supplier.		
Trigger:	Shop wants to see a supplier.		
Preconditions:	Shop must be logged in.		
Postconditions:	Supplier details are displayed.		
Normal Flow:	<ol style="list-style-type: none"> 1. Shop navigates to the supplier section. 2. Shop views the supplier details. 		
Alternative Flows:	None		
Exceptions:	E1: Supplier does not exist. The system displays an error message.		
Priority:	Medium		
Frequency of Use:	Regular		
Business Rules:	BR-21		
Other Information:	None		

Assumptions:	None
--------------	------

ID and Name:	UC-17: View Supplier		
Created By:	Nguyễn Văn Minh Thoại	Date Created:	22/10/24
Primary Actor:	Shop	Secondary Actors:	admin
Description:	Shop views details of a supplier.		
Trigger:	Shop wants to view a supplier.		
Preconditions:	Shop must be logged in.		
Postconditions:	Supplier details are displayed.		
Normal Flow:	<ol style="list-style-type: none"> 1. Shop navigates to the supplier section. 2. Shop views the supplier details. 		
Alternative Flows:	None		
Exceptions:	E1: Supplier does not exist. The system displays an error message.		
Priority:	Medium		
Frequency of Use:	Regular.		
Business Rules:	BR-21		
Other Information:	None		

Assumptions:	None
--------------	------

ID and Name:	UC-18: Create Promotion		
Created By:	Nguyễn Văn Minh Thoại	Date Created:	22/10/24
Primary Actor:	Shop	Secondary Actors:	None
Description:	Shop creates a new promotion.		
Trigger:	Shop wants to add a new promotion for customers.		
Preconditions:	Shop must be logged in.		
Postconditions:	A new promotion is created.		
Normal Flow:	<ol style="list-style-type: none"> 1. Shop navigates to the promotion creation page. 2. Shop enters the promotion details and saves it. 		
Alternative Flows:	None		
Exceptions:	E1: Required promotion information is missing. The system displays an error message.		
Priority:	Medium		
Frequency of Use:	Regular		
Business Rules:	BR-5, BR-21		
Other Information:	None		

Assumptions:	None

ID and Name:	UC-19: Update Promotion		
Created By:	Nguyễn Văn Minh Thoại	Date Created:	22/10/24
Primary Actor:	Shop	Secondary Actors:	None
Description:	Shop updates an existing promotion.		
Trigger:	Shop wants to update a promotion.		
Preconditions:	Shop must be logged in.		
Postconditions:	Promotion is updated.		
Normal Flow:	<ol style="list-style-type: none"> 1. Shop navigates to the promotion section. 2. Shop updates the promotion details and saves the changes. 		
Alternative Flows:	None		
Exceptions:	E1: Promotion does not exist. The system displays an error message.		
Priority:	Medium		
Frequency of Use:	Regular		
Business Rules:	BR-5, BR-21		

Other Information:	None
Assumptions:	None

ID and Name:	UC-20: View Promotion		
Created By:	Nguyễn Văn Minh Thoại	Date Created:	22/10/24
Primary Actor:	Shop	Secondary Actors:	Admin
Description:	Shop views details of a promotion.		
Trigger:	Shop wants to view a promotion.		
Preconditions:	Shop must be logged in.		
Postconditions:	Promotion details are displayed.		
Normal Flow:	<ol style="list-style-type: none"> 1. Shop navigates to the promotion section. 2. Shop views the promotion details. 		
Alternative Flows:	None		
Exceptions:	E1: Promotion does not exist. The system displays an error message.		
Priority:	Medium		
Frequency of Use:	Regular		
Business Rules:	BR-5		

Other Information:	None
Assumptions:	None

ID and Name:	UC-21: Delete Promotion		
Created By:	Nguyễn Văn Minh Thoại	Date Created:	22/10/24
Primary Actor:	Shop	Secondary Actors:	None
Description:	Shop deletes an existing promotion.		
Trigger:	Shop wants to delete a promotion.		
Preconditions:	Shop must be logged in.		
Postconditions:	Promotion is deleted.		
Normal Flow:	<ol style="list-style-type: none"> 1. Shop navigates to the promotion section. 2. Shop deletes the promotion. 		
Alternative Flows:	None		
Exceptions:	E1: Promotion does not exist. The system displays an error message.		
Priority:	Medium		
Frequency of Use:	Regular		
Business Rules:	BR-5, BR-21		

Other Information:	None
Assumptions:	None

ID and Name:	UC-22: View Payment		
Created By:	Nguyễn Văn Minh Thoại	Date Created:	22/10/24
Primary Actor:	Shop	Secondary Actors:	Admin
Description:	Shop views details of a payment.		
Trigger:	Shop wants to view a payment.		
Preconditions:	Shop must be logged in.		
Postconditions:	Payment details are displayed.		
Normal Flow:	<ol style="list-style-type: none"> 1. Shop navigates to the payment section. 2. Shop views the payment details. 		
Alternative Flows:	None		
Exceptions:	E1: Payment does not exist. The system displays an error message.		
Priority:	Medium		
Frequency of Use:	Regular		
Business Rules:	BR-24		

Other Information:	None
Assumptions:	None

ID and Name:	UC-23: View Orders		
Created By:	Nguyễn Văn Minh Thoại	Date Created:	22/10/24
Primary Actor:	Shop	Secondary Actors:	Admin
Description:	Shop views a list of orders.		
Trigger:	Shop wants to view orders.		
Preconditions:	Shop must be logged in.		
Postconditions:	List of orders is displayed.		
Normal Flow:	<ol style="list-style-type: none"> 1. Shop navigates to the orders section. 2. Shop views the orders list. 		
Alternative Flows:	None		
Exceptions:	E1: No orders are available. The system displays an error message.		
Priority:	Medium		
Frequency of Use:	Regular		
Business Rules:	BR-6		

Other Information:	None
Assumptions:	None

ID and Name:	UC-24: Update Orders		
Created By:	Nguyễn Văn Minh Thoại	Date Created:	22/10/24
Primary Actor:	Shop	Secondary Actors:	None
Description:	The Shop updates the status or details of an existing order.		
Trigger:	The Shop wants to modify an order, such as updating its status.		
Preconditions:	<p>The Shop must be logged into the system.</p> <p>The order must not be confirmed or shipped, unless authorized by an admin.</p>		
Postconditions:	The order is updated successfully.		
Normal Flow:	<ol style="list-style-type: none"> 1. The Shop navigates to the "Orders" section. 2. The Shop selects the order to be updated. 3. The Shop makes the necessary changes to the order details. 4. The system updates the order and saves the changes. 		
Alternative Flows:	None		
Exceptions:	E1: The order has already been confirmed or shipped, and cannot be updated unless authorized. The system displays an error message.		
Priority:	Medium		
Frequency of Use:	Regular part of performance monitoring.		

Business Rules:	BR-7
Other Information:	None
Assumptions:	None

ID and Name:	UC-25: Create Category		
Created By:	Nguyễn Văn Minh Thoại	Date Created:	02/06/24
Primary Actor:	Shop	Secondary Actors:	None
Description:	The Shop creates a new product category to organize products more effectively.		
Trigger:	The Shop wants to create a new product category.		
Preconditions:	The Shop must be logged into the system.		
Postconditions:	A new product category is successfully created.		
Normal Flow:	<ol style="list-style-type: none"> 1. The Shop navigates to the "Categories" section. 2. The Shop selects the option to create a new category. 3. The Shop enters the details for the new category (e.g., category name, description). 4. The system saves the new category and adds it to the list. 		
Alternative Flows:	None		

Exceptions:	E1: The category name already exists. The system displays an error message.
Priority:	Low
Frequency of Use:	Occasional
Business Rules:	BR-21
Other Information:	None
Assumptions:	None

ID and Name:	UC-26: Update Category		
Created By:	Nguyễn Văn Minh Thoại	Date Created:	22/10/24
Primary Actor:	Shop	Secondary Actors:	None
Description:	The Shop updates the details of an existing product category.		
Trigger:	The Shop wants to modify an existing product category.		
Preconditions:	The Shop must be logged into the system.		
Postconditions:	The product category is updated successfully.		
Normal Flow:	<ol style="list-style-type: none"> 1. The Shop navigates to the "Categories" section. 2. The Shop selects the category to be updated. 3. The Shop updates the category details (e.g., name, description). 4. The system saves the updated category. 		

Alternative Flows:	None
Exceptions:	E1: The category name already exists. The system displays an error message.
Priority:	Medium
Frequency of Use:	Occasional
Business Rules:	BR-21
Other Information:	None
Assumptions:	None

ID and Name:	UC-27: Delete Category		
Created By:	Nguyễn Văn Minh Thoại	Date Created:	22/10/24
Primary Actor:	Shop	Secondary Actors:	None
Description:	The Shop deletes an existing product category.		
Trigger:	The Shop wants to remove an existing product category.		
Preconditions:	<p>The Shop must be logged into the system.</p> <p>The category must not have any associated products or be currently in use.</p>		
Postconditions:	The category is deleted successfully.		

Normal Flow:	<ol style="list-style-type: none"> 1. The Shop navigates to the "Categories" section. 2. The Shop selects the category to be deleted. 3. The system prompts for confirmation before deletion. 4. The category is removed from the system.
Alternative Flows:	None
Exceptions:	E1: The category has associated products. The system displays an error message indicating that the category cannot be deleted.
Priority:	low
Frequency of Use:	Occasional
Business Rules:	BR-21
Other Information:	None
Assumptions:	None

ID and Name:	UC-28: View Category		
Created By:	Nguyễn Văn Minh Thoại	Date Created:	22/10/24
Primary Actor:	Shop	Secondary Actors:	None
Description:	The Shop views a list of existing product categories.		
Trigger:	The Shop wants to view the available product categories.		
Preconditions:	The Shop must be logged into the system.		

Postconditions:	The list of categories is displayed.
Normal Flow:	<ol style="list-style-type: none"> 1. The Shop navigates to the "Categories" section. 2. The system displays the list of available categories.
Alternative Flows:	None
Exceptions:	E1: No categories are available. The system displays an error message.
Priority:	Medium
Frequency of Use:	Regular
Business Rules:	BR -21
Other Information:	None
Assumptions:	None

ID and Name:	UC-29 View Dashboard		
Created By:	Đào Công An Phước	Date Created:	22/10/2024
Primary Actor:	Member	Secondary Actors:	Admin, Shop
Description:	A Member views the system dashboard to check the overall summary, including pond statistics, Koi fish details, and notifications.		
Trigger:	The Member logs into the system and wants to view the dashboard.		
Preconditions:	The Member must be logged into the system.		

	The dashboard must be properly configured with the Member's relevant data.
Postconditions:	The dashboard displays the relevant summary and data for the Member.
Normal Flow:	<ol style="list-style-type: none"> 1. The Member logs into the system. 2. The system loads and displays the dashboard page. 3. The Member reviews. 4. The Member can navigate to other sections (e.g., Koi Fish, Ponds, Reminders) from the dashboard.
Alternative Flows:	None
Exceptions:	E1: Dashboard data fails to load due to a system error. The system shows an error message and prompts the user to try again later.
Priority:	High
Frequency of Use:	Daily
Business Rules:	BR-25
Other Information:	None
Assumptions:	None

ID and Name:	UC-30 Forgot Password		
Created By:	Đào Công An Phước	Date Created:	22/10/2024
Primary Actor:	Member	Secondary Actors:	None

Description:	The Member requests to reset their password if they forget it, triggering the password recovery process.
Trigger:	The Member clicks on "Forgot Password" on the login screen.
Preconditions:	The Member's email must be registered in the system
Postconditions:	The system sends a password reset link to the Member's email.
Normal Flow:	<ol style="list-style-type: none"> 1. The Member clicks "Forgot Password." 2. The system prompts the Member to enter their registered email. 3. The system validates the email and sends a password reset link to the email address. 4. The Member receives the email and follows the instructions to reset their password.
Alternative Flows:	A1: The email is not registered in the system. The system shows an error message
Exceptions:	E1: The email server fails to send the reset link. The system notifies the Member and prompts them to try again later.
Priority:	High
Frequency of Use:	As needed
Business Rules:	BR-25
Other Information:	None
Assumptions:	None

ID and Name:	UC-31 Verify Account
--------------	-----------------------------

Created By:	Đào Công An Phước	Date Created:	22/10/2024
Primary Actor:	Member	Secondary Actors:	Admin
Description:	A Member verifies their account by following a link sent to their email upon registration.		
Trigger:	The Member registers a new account and is prompted to verify their email address.		
Preconditions:	<p>The Member must have successfully registered.</p> <p>The system must have sent the verification link to the Member's email.</p>		
Postconditions:	The Member's account is verified, and their role is updated from "GUEST" to "MEMBER." and status to true.		
Normal Flow:	<ol style="list-style-type: none"> 1. The Member registers and receives an email verification link. 2. The Member clicks the verification link in the email. 3. The system validates the link and verifies the Member's account. 4. The Member's role is updated from "GUEST" to "MEMBER." 5. The Member's status is updated true. 		
Alternative Flows:	A1: The link has expired. The system allows the Member to request a new link.		
Exceptions:	E1: The link is invalid. The system notifies the Member and provides instructions to request a new link.		
Priority:	High		
Frequency of Use:	Once per registration		
Business Rules:	BR-3, BR-7		
Other Information:	None		
Assumptions:	The Member receives the verification email without issue.		

--	--

ID and Name:	UC-32 View Profile		
Created By:	Đào Công An Phước	Date Created:	22/10/2024
Primary Actor:	Member	Secondary Actors:	Shop
Description:	A Member views their profile, which includes their personal information, preferences, and system usage data.		
Trigger:	The Member clicks on "View Profile" in the account settings.		
Preconditions:	The Member must be logged in.		
Postconditions:	The system displays the Member's profile information.		
Normal Flow:	<ol style="list-style-type: none"> 1. The Member navigates to the "Profile" page from the menu. 2. The system fetches and displays the Member's profile details. 3. The Member can view their profile information (e.g., name, email, preferences, etc.). 		
Alternative Flows:	None		
Exceptions:	E1: The profile data fails to load. The system shows an error message and prompts the Member to retry.		
Priority:	Medium		
Frequency of Use:	Occasionally		
Business Rules:	BR-18		
Other Information:	None		
Assumptions:	The Member's profile data is complete and up to date.		

--	--

ID and Name:	UC-33 View Order		
Created By:	Đào Công An Phước	Date Created:	22/10/2024
Primary Actor:	Member	Secondary Actors:	Admin
Description:	The Member views the status and details of past or current orders placed through the system.		
Trigger:	The Member clicks on "View Orders" in their account.		
Preconditions:	The Member must have placed at least one order.		
Postconditions:	The system displays the order history and current order statuses.		
Normal Flow:	<ol style="list-style-type: none"> 1. The Member navigates to the "Orders" section. 2. The system fetches the Member's order history and displays a list of orders. 3. The Member selects an order to view more details. 		
Alternative Flows:	None		
Exceptions:	E1: The system fails to retrieve order data. The system shows an error message.		
Priority:	High		
Frequency of Use:	As needed		
Business Rules:	BR-6, BR-7		
Other Information:	None		
Assumptions:	The system maintains a proper record of all orders.		

--	--

ID and Name:	UC-34 Update Profile		
Created By:	Đào Công An Phước	Date Created:	22/10/2024
Primary Actor:	Member	Secondary Actors:	Admin
Description:	A Member updates their personal information, such as contact details and preferences.		
Trigger:	The Member clicks on "Update Profile" in their account settings.		
Preconditions:	The Member must be logged in. The Member's profile must exist in the system.		
Postconditions:	The system updates the Member's profile with the new information.		
Normal Flow:	<ol style="list-style-type: none"> The Member navigates to the "Update Profile" section. The system displays the current profile information. The Member updates the desired fields (e.g., email, phone number). The Member submits the changes. The system validates and saves the new information. 		
Alternative Flows:	A1: The Member submits invalid data (e.g., incorrect email format). The system shows an error message.		
Exceptions:	E1: The profile update fails due to a system error. The system notifies the Member.		
Priority:	High		
Frequency of Use:	Occasionally		
Business Rules:	BR-18		
Other Information:	None		

Assumptions:	The Member provides valid and accurate information for updates.
--------------	---

ID and Name:	UC-35 Update Koi Fish		
Created By:	Đào Công An Phước	Date Created:	22/10/2024
Primary Actor:	Member	Secondary Actors:	None
Description:	A Member updates the details of an existing Koi fish in their records.		
Trigger:	The Member selects a Koi fish from their list and chooses to update its details.		
Preconditions:	The Member must be logged in. The Koi fish must exist in the Member's records.		
Postconditions:	The system updates the Koi fish details as specified by the Member.		
Normal Flow:	<ol style="list-style-type: none"> 1. The Member navigates to the Koi fish list. 2. The system displays the list of Koi fish. 3. The Member selects a Koi fish to update. 4. The system displays the current details of the Koi fish. 5. The Member updates the necessary fields (e.g., weight, age). 6. The Member submits the changes. 7. The system validates and updates the Koi fish record. 		
Alternative Flows:	A1: The Member tries to update with invalid data. The system shows an error message.		
Exceptions:	E1: The update process fails due to a system error. The system notifies the Member.		
Priority:	High		
Frequency of Use:	Occasionally		
Business Rules:	BR-8		

Other Information:	None
Assumptions:	The Member has valid updates to the Koi fish details.

ID and Name:	UC-36 Create Koi Fish		
Created By:	Đào Công An Phước	Date Created:	22/10/2024
Primary Actor:	Member	Secondary Actors:	None
Description:	A Member creates a new record for a Koi fish they own.		
Trigger:	The Member chooses to create a new Koi fish entry from their account.		
Preconditions:	The Member must be logged in.		
Postconditions:	The system creates a new Koi fish record with the provided details.		
Normal Flow:	<ol style="list-style-type: none"> 1. The Member navigates to the "Create Koi Fish" section. 2. The system presents a form for entering Koi fish details. 3. The Member fills in the details (e.g., name, weight, breed). 4. The Member submits the form. 5. The system validates and creates the Koi fish record. 		
Alternative Flows:	A1: The Member submits invalid data (e.g., missing required fields). The system shows an error message.		
Exceptions:	E1: The creation process fails due to a system error. The system notifies the Member.		
Priority:	Medium		
Frequency of Use:	As needed		
Business Rules:	BR-8		

Other Information:	None
Assumptions:	The Member provides complete and valid information for the Koi fish.

ID and Name:	UC-37 Delete Koi Fish		
Created By:	Đào Công An Phước	Date Created:	22/10/2024
Primary Actor:	Member	Secondary Actors:	None
Description:	A Member deletes an existing Koi fish record from their account.		
Trigger:	The Member selects a Koi fish and chooses to delete it.		
Preconditions:	The Member must be logged in. The Koi fish must exist in the Member's records.		
Postconditions:	The system deletes the specified Koi fish record.		
Normal Flow:	<ol style="list-style-type: none"> 1. The Member navigates to the Koi fish list. 2. The system displays the list of Koi fish. 3. The Member selects a Koi fish to delete. 4. The system asks for confirmation. 5. The Member confirms the deletion. 6. The system deletes the Koi fish record. 		
Alternative Flows:	None.		
Exceptions:	E1: The deletion process fails due to a system error. The system notifies the Member		
Priority:	Medium		
Frequency of Use:	As needed		
Business Rules:	BR-8		

Other Information:	None
Assumptions:	The Member is sure about the deletion action.

ID and Name:	UC-38 View Koi Fish		
Created By:	Đào Công An Phước	Date Created:	22/10/2024
Primary Actor:	Member	Secondary Actors:	None
Description:	A Member views the details of their Koi fish.		
Trigger:	The Member selects "View Koi Fish" from their account menu.		
Preconditions:	The Member must be logged in. The Koi fish must have Koi fish records.		
Postconditions:	The system displays the Koi fish details.		
Normal Flow:	<ol style="list-style-type: none"> 1. The Member navigates to the "Koi Fish" section. 2. The system fetches and displays the list of Koi fish. 3. The Member selects a Koi fish to view details. 4. The system displays the details of the selected Koi fish. 		
Alternative Flows:	None.		
Exceptions:	E1: The system fails to load Koi fish data. The system shows an error message.		
Priority:	Medium		
Frequency of Use:	Occasionally		
Business Rules:	BR-8		
Other Information:	None		

Assumptions:	The system has up-to-date Koi fish records.

ID and Name:	UC-39 Update Growth History		
Created By:	Đào Công An Phước	Date Created:	22/10/2024
Primary Actor:	Member	Secondary Actors:	None
Description:	A Member updates the growth history record of a specific Koi fish		
Trigger:	The Member selects a growth history entry and chooses to update it.		
Preconditions:	<p>The Member must be logged in.</p> <p>The growth history entry must exist for the selected Koi fish.</p>		
Postconditions:	The system updates the growth history record with the new information.		
Normal Flow:	<ol style="list-style-type: none"> 1. The Member navigates to the "Growth History" section. 2. The system displays the list of growth history entries for the selected Koi fish. 3. The Member selects a specific growth history entry to update. 4. The system displays the current details of the entry. 5. The Member updates the necessary fields. 6. The Member submits the changes. 7. The system validates and updates the growth history record. 		
Alternative Flows:	A1: The Member submits invalid data. The system shows an error message.		
Exceptions:	E1: The update process fails due to a system error. The system notifies the Member.		
Priority:	Medium		
Frequency of Use:	Occasionally		

Business Rules:	BR-9
Other Information:	None
Assumptions:	The Member has valid updates for the growth history entry.

ID and Name:	UC-40 Create Growth History		
Created By:	Đào Công An Phước	Date Created:	22/10/2024
Primary Actor:	Member	Secondary Actors:	None
Description:	A Member creates a new growth history record for a specific Koi fish.		
Trigger:	The Member chooses to add a new growth history entry.		
Preconditions:	The Member must be logged in. The Koi fish must exist in the Member's records.		
Postconditions:	The system creates a new growth history record for the selected Koi fish		
Normal Flow:	<ol style="list-style-type: none"> 1. The Member navigates to the "Create Growth History" section. 2. The system presents a form for entering growth details. 3. The Member fills in the details (e.g., date, weight, size). 4. The Member submits the form. 5. The system validates and creates the growth history record. 		
Alternative Flows:	A1: The Member submits invalid data. The system shows an error message.		
Exceptions:	E1: The deletion process fails due to a system error. The system notifies the Member		
Priority:	Medium		

Frequency of Use:	As needed
Business Rules:	BR-8
Other Information:	None
Assumptions:	The Member provides complete and valid information for the growth history entry.

ID and Name:	UC-41 Delete Growth History		
Created By:	Đào Công An Phước	Date Created:	22/10/2024
Primary Actor:	Member	Secondary Actors:	None
Description:	A Member deletes a growth history record of a specific Koi fish.		
Trigger:	The Member selects a growth history entry and chooses to delete it.		
Preconditions:	<p>The Member must be logged in.</p> <p>The growth history entry must exist.</p>		
Postconditions:	The system deletes the specified growth history record.		
Normal Flow:	<ol style="list-style-type: none"> 1. The Member navigates to the "Growth History" section. 2. The system displays the list of growth history entries for the selected Koi fish. 3. The Member selects a specific growth history entry to delete. 4. The system asks for confirmation. 5. The Member confirms the deletion. 6. The system deletes the growth history record. 		
Alternative Flows:	None.		
Exceptions:	E1: The deletion process fails due to a system error. The system notifies the Member		

Priority:	Medium
Frequency of Use:	As needed
Business Rules:	BR-9
Other Information:	None
Assumptions:	The Member is sure about the deletion action.

ID and Name:	UC-42 View Growth History		
Created By:	Đào Công An Phước	Date Created:	22/10/2024
Primary Actor:	Member	Secondary Actors:	None
Description:	A Member views the growth history of a specific Koi fish.		
Trigger:	The Member selects "View Growth History" for a Koi fish.		
Preconditions:	The Member must be logged in. The selected Koi fish must have growth history records.		
Postconditions:	The system displays the growth history entries.		
Normal Flow:	<ol style="list-style-type: none"> 1. The Member navigates to the "Growth History" section. 2. The system fetches and displays the growth history for the selected Koi fish. 3. The Member can view details of each growth entry. 		
Alternative Flows:	None.		
Exceptions:	E1: The system fails to load growth history data. The system shows an error message.		
Priority:	Medium		

Frequency of Use:	Occasionally
Business Rules:	BR-9
Other Information:	None
Assumptions:	The system has up-to-date growth history records.

ID and Name:	UC-43 Update Remark		
Created By:	Đào Công An Phước	Date Created:	22/10/2024
Primary Actor:	Member	Secondary Actors:	None
Description:	A Member updates remarks associated with a specific Koi fish.		
Trigger:	The Member selects a remark to update.		
Preconditions:	<p>The Member must be logged in.</p> <p>The remark must exist for the selected Koi fish.</p>		
Postconditions:	The system updates the remark with the new information.		
Normal Flow:	<ol style="list-style-type: none"> 1. The Member navigates to the "Remarks" section. 2. The system displays the list of remarks for the selected Koi fish. 3. The Member selects a specific remark to update. 4. The system displays the current details of the remark. 5. The Member updates the necessary fields. 6. The Member submits the changes. 7. The system validates and updates the remark. 		
Alternative Flows:	A1: The Member submits invalid data. The system shows an error message.		
Exceptions:	E1: The deletion process fails due to a system error. The system notifies the Member		
Priority:	Medium		

Frequency of Use:	Occasionally
Business Rules:	BR-12
Other Information:	None
Assumptions:	The Member has valid updates for the remark.

ID and Name:	UC-44 Create Remark		
Created By:	Đào Công An Phước	Date Created:	22/10/2024
Primary Actor:	Member	Secondary Actors:	None
Description:	A Member creates a new remark for a specific Koi fish.		
Trigger:	The Member chooses to add a new remark.		
Preconditions:	The Member must be logged in. The selected Koi fish must exist.		
Postconditions:	The system creates a new remark associated with the selected Koi fish.		
Normal Flow:	<ol style="list-style-type: none"> 1. The Member navigates to the "Create Remark" section. 2. The system presents a form for entering remark details. 3. The Member fills in the remark information. 4. The Member submits the form. 5. The system validates and creates the remark. 		
Alternative Flows:	A1: The Member submits invalid data. The system shows an error message.		
Exceptions:	E1: The creation process fails due to a system error. The system notifies the Member.		

Priority:	Medium
Frequency of Use:	As needed
Business Rules:	BR-12
Other Information:	None
Assumptions:	The Member provides complete and valid information for the remark.

ID and Name:	UC-45 Delete Remark		
Created By:	Đào Công An Phước	Date Created:	22/10/2024
Primary Actor:	Member	Secondary Actors:	None
Description:	A Member deletes a remark associated with a specific Koi fish.		
Trigger:	The Member selects a remark and chooses to delete it		
Preconditions:	<p>The Member must be logged in.</p> <p>The remark must exist for the selected Koi fish.</p>		
Postconditions:	The system deletes the specified remark.		
Normal Flow:	<ol style="list-style-type: none"> 1. The Member navigates to the "Remarks" section. 2. The system displays the list of remarks for the selected Koi fish. 3. The Member selects a specific remark to delete. 4. The system asks for confirmation. 5. The Member confirms the deletion. 6. The system deletes the remark. 		
Alternative Flows:	None.		
Exceptions:	E1: The deletion process fails due to a system error. The system notifies the Member.		

Priority:	Medium
Frequency of Use:	As needed
Business Rules:	BR-12
Other Information:	None
Assumptions:	The Member is sure about the deletion action.

ID and Name:	UC-46 View Remark		
Created By:	Đào Công An Phước	Date Created:	22/10/2024
Primary Actor:	Member	Secondary Actors:	None
Description:	A Member views remarks associated with a specific Koi fish.		
Trigger:	The Member selects "View Remarks" for a Koi fish.		
Preconditions:	The Member must be logged in. The selected Koi fish must have remarks associated with it.		
Postconditions:	The system displays the remarks for the selected Koi fish.		
Normal Flow:	<ol style="list-style-type: none"> 1. The Member navigates to the "Remarks" section. 2. The system fetches and displays the remarks for the selected Koi fish. 3. The Member can view details of each remark. 		
Alternative Flows:	None.		
Exceptions:	E1: The system fails to load remarks data. The system shows an error message.		
Priority:	Medium		

Frequency of Use:	Occasionally
Business Rules:	BR-12
Other Information:	None
Assumptions:	The system has up-to-date remark records.

ID and Name:	UC-47 Create Koi Pond		
Created By:	Đào Công An Phước	Date Created:	22/10/2024
Primary Actor:	Member	Secondary Actors:	None
Description:	A Member creates a new Koi pond record.		
Trigger:	The Member chooses to create a new Koi pond entry.		
Preconditions:	The Member must be logged in.		
Postconditions:	The system creates a new Koi pond record with the provided details.		
Normal Flow:	<ol style="list-style-type: none"> 1. The Member navigates to the "Create Koi Pond" section. 2. The system presents a form for entering pond details. 3. The Member fills in the details (e.g., name, size, depth). 4. The Member submits the form. 5. The system validates and creates the Koi pond record. 		
Alternative Flows:	A1: The Member submits invalid data. The system shows an error message.		
Exceptions:	E1: The deletion process fails due to a system error. The system notifies the Member		
Priority:	Medium		
Frequency of Use:	As needed		

Business Rules:	BR-10
Other Information:	None
Assumptions:	The Member provides complete and valid information for the Koi pond.

ID and Name:	UC-48 Delete Koi Pond		
Created By:	Đào Công An Phước	Date Created:	22/10/2024
Primary Actor:	Member	Secondary Actors:	None
Description:	A Member deletes a Koi pond record.		
Trigger:	The Member selects a pond and chooses to delete it.		
Preconditions:	The Member must be logged in. The selected Koi pond must exist.		
Postconditions:	The system deletes the specified Koi pond record.		
Normal Flow:	<ol style="list-style-type: none"> 1. The Member navigates to the "Koi Ponds" section. 2. The system displays the list of Koi ponds associated with the Member. 3. The Member selects a specific pond to delete. 4. The system asks for confirmation. 5. The Member confirms the deletion. 6. The system deletes the Koi pond record. 		
Alternative Flows:	None.		
Exceptions:	E1: The deletion process fails due to a system error. The system notifies the Member		
Priority:	Medium		

Frequency of Use:	As needed
Business Rules:	BR-10
Other Information:	None
Assumptions:	The Member is sure about the deletion action.

ID and Name:	UC-49 Update Water Parameter		
Created By:	Đào Công An Phước	Date Created:	22/10/2024
Primary Actor:	Member	Secondary Actors:	None
Description:	A Member updates the water parameters for a specific Koi pond.		
Trigger:	The Member selects a water parameter to update.		
Preconditions:	<p>The Member must be logged in.</p> <p>The water parameter record must exist.</p>		
Postconditions:	The system updates the specified water parameter with new values.		
Normal Flow:	<ol style="list-style-type: none"> 1. The Member navigates to the "Water Parameters" section. 2. The system displays the list of water parameters for the selected pond. 3. The Member selects a specific water parameter to update. 4. The system displays the current values of the parameter. 5. The Member updates the necessary fields. 6. The Member submits the changes. 7. The system validates and updates the water parameter. 		
Alternative Flows:	A1: The Member submits invalid data. The system shows an error message.		
Exceptions:	E1: The update process fails due to a system error. The system notifies the Member.		

Priority:	Medium
Frequency of Use:	As needed
Business Rules:	BR-10
Other Information:	None
Assumptions:	The Member provides valid updates for the water parameters.

ID and Name:	UC-50 View Water Parameter		
Created By:	Đào Công An Phước	Date Created:	22/10/2024
Primary Actor:	Member	Secondary Actors:	None
Description:	A Member views the current water parameters for a specific Koi pond.		
Trigger:	The Member selects "View Water Parameters" for a Koi pond.		
Preconditions:	<p>The Member must be logged in.</p> <p>The selected Koi pond must have water parameter records.</p>		
Postconditions:	The system displays the current water parameters for the selected Koi pond.		
Normal Flow:	<ol style="list-style-type: none"> 1. The Member navigates to the "Water Parameters" section. 2. The system fetches and displays the water parameters for the selected Koi pond. 3. The Member can review the details of each water parameter. 		
Alternative Flows:	None.		
Exceptions:	E1: The system fails to load water parameter data. The system shows an error message.		
Priority:	Medium		

Frequency of Use:	Occasionally
Business Rules:	BR-10
Other Information:	None
Assumptions:	The system has up-to-date water parameter records.

ID and Name:	UC-51 Create Water Parameter		
Created By:	Đào Công An Phước	Date Created:	22/10/2024
Primary Actor:	Member	Secondary Actors:	None
Description:	A Member creates a new water parameter record for a specific Koi pond.		
Trigger:	The Member chooses to create a new water parameter entry.		
Preconditions:	The Member must be logged in. The selected Koi pond must exist.		
Postconditions:	The system creates a new water parameter record with the provided details.		
Normal Flow:	<ol style="list-style-type: none"> 1. The Member navigates to the "Create Water Parameter" section. 2. The system presents a form for entering water parameter details. 3. The Member fills in the necessary fields (e.g., temperature, pH). 4. The Member submits the form. 5. The system validates and creates the water parameter record. 		
Alternative Flows:	A1: The Member submits invalid data. The system shows an error message.		

Exceptions:	E1: The creation process fails due to a system error. The system notifies the Member.
Priority:	Medium
Frequency of Use:	As needed
Business Rules:	BR-10
Other Information:	None
Assumptions:	The Member provides complete and valid information for the water parameter.

ID and Name:	UC-52 Delete Water Parameter		
Created By:	Đào Công An Phước	Date Created:	22/10/2024
Primary Actor:	Member	Secondary Actors:	None
Description:	A Member deletes a water parameter record		
Trigger:	The Member selects a water parameter and chooses to delete it.		
Preconditions:	The Member must be logged in. The selected water parameter must exist.		
Postconditions:	The system deletes the specified water parameter record.		
Normal Flow:	<ol style="list-style-type: none"> 1. The Member navigates to the "Water Parameters" section. 2. The system displays the list of water parameters for the selected Koi pond. 3. The Member selects a specific water parameter to delete. 4. The system asks for confirmation. 5. The Member confirms the deletion. 6. The system deletes the water parameter record. 		

Alternative Flows:	None.
Exceptions:	E1: The deletion process fails due to a system error. The system notifies the Member.
Priority:	Medium
Frequency of Use:	As needed
Business Rules:	BR-10
Other Information:	None
Assumptions:	The Member is sure about the deletion action.

ID and Name:	UC-53 Update Reminder		
Created By:	Đào Công An Phước	Date Created:	22/10/2024
Primary Actor:	Member	Secondary Actors:	None
Description:	A Member updates an existing reminder.		
Trigger:	The Member selects a reminder to update.		
Preconditions:	The Member must be logged in. The reminder must exist.		
Postconditions:	The system updates the specified reminder.		
Normal Flow:	<ol style="list-style-type: none"> 1. The Member navigates to the "Reminders" section. 2. The system displays the list of reminders. 3. The Member selects a specific reminder to update. 4. The system displays the current details of the reminder. 5. The Member updates the necessary fields. 		

	<p>6. The Member submits the changes.</p> <p>7. The system validates and updates the reminder.</p>
Alternative Flows:	A1: The Member submits invalid data. The system shows an error message.
Exceptions:	E1: The system fails to load water parameter data. The system shows an error message.
Priority:	Medium
Frequency of Use:	As needed
Business Rules:	BR-10
Other Information:	None
Assumptions:	The Member provides valid updates for the reminder.

ID and Name:	UC-54 Create Reminder		
Created By:	Đào Công An Phước	Date Created:	22/10/2024
Primary Actor:	Member	Secondary Actors:	None
Description:	A Member creates a new reminder for a specific task.		
Trigger:	The Member chooses to add a new reminder.		
Preconditions:	The Member must be logged in.		
Postconditions:	The system creates a new reminder with the provided details.		
Normal Flow:	<ol style="list-style-type: none"> 1. The Member navigates to the "Create Reminder" section. 2. The system presents a form for entering reminder details. 3. The Member fills in the reminder information. 4. The Member submits the form. 		

	5. The system validates and creates the reminder.
Alternative Flows:	A1: The Member submits invalid data. The system shows an error message
Exceptions:	E1: The system fails to load water parameter data. The system shows an error message.
Priority:	Medium
Frequency of Use:	As needed
Business Rules:	BR-10
Other Information:	None
Assumptions:	The Member provides complete and valid information for the reminder.

ID and Name:	UC-55 Delete Reminder		
Created By:	Đào Công An Phước	Date Created:	22/10/2024
Primary Actor:	Member	Secondary Actors:	None
Description:	A Member deletes a reminder.		
Trigger:	The Member selects a reminder and chooses to delete it.		
Preconditions:	The Member must be logged in. The selected reminder must exist.		
Postconditions:	The system deletes the specified reminder.		
Normal Flow:	<ol style="list-style-type: none"> 1. The Member navigates to the "Reminders" section. 2. The system displays the list of reminders. 3. The Member selects a specific reminder to delete. 		

	4. The system asks for confirmation. 5. The Member confirms the deletion. 6. The system deletes the reminder.
Alternative Flows:	None.
Exceptions:	E1: The deletion process fails due to a system error. The system notifies the Member.
Priority:	Medium
Frequency of Use:	As needed
Business Rules:	BR-10
Other Information:	None
Assumptions:	The Member is sure about the deletion action.

ID and Name:	UC-56 View Reminder		
Created By:	Đào Công An Phước	Date Created:	22/10/2024
Primary Actor:	Member	Secondary Actors:	None
Description:	A Member views all reminders.		
Trigger:	The Member selects "View Reminders."		
Preconditions:	The Member must be logged in.		
Postconditions:	The system displays all reminders associated with the Member.		
Normal Flow:	1. The Member navigates to the "Reminders" section. 2. The system displays the list of reminders for the Member. 3. The Member reviews the details of each reminder.		

Alternative Flows:	None.
Exceptions:	E1: The system fails to load reminder data. The system shows an error message.
Priority:	Medium
Frequency of Use:	Occasionally
Business Rules:	BR-10
Other Information:	None
Assumptions:	The system has up-to-date reminder records.

ID and Name:	UC-57 Search Product		
Created By:	Đào Công An Phước	Date Created:	22/10/2024
Primary Actor:	Member	Secondary Actors:	None
Description:	A Member searches for products using keywords.		
Trigger:	The Member enters keywords into the search bar.		
Preconditions:	The Member must be logged in.		
Postconditions:	The system displays a list of products matching the search criteria.		
Normal Flow:	<ol style="list-style-type: none"> 1. The Member navigates to the "Products" section. 2. The Member enters keywords in the search bar. 3. The Member submits the search request. 4. The system retrieves and displays the matching products. 		
Alternative Flows:	A1: No products match the search criteria. The system informs the Member.		

Exceptions:	E1: The search process fails due to a system error. The system shows an error message.
Priority:	Medium
Frequency of Use:	As needed
Business Rules:	BR-20
Other Information:	None
Assumptions:	The system has a valid product catalog.

ID and Name:	UC-58 View Product Details		
Created By:	Đào Công An Phước	Date Created:	22/10/2024
Primary Actor:	Member	Secondary Actors:	None
Description:	A Member views detailed information about a specific product.		
Trigger:	The Member selects a product from the list.		
Preconditions:	The Member must be logged in. The selected product must exist.		
Postconditions:	The system displays detailed information about the selected product.		
Normal Flow:	<ol style="list-style-type: none"> 1. The Member navigates to the "Products" section. 2. The system displays a list of products. 3. The Member selects a specific product to view. 4. The system displays detailed product information. 		
Alternative Flows:	None.		

Exceptions:	E1: The system fails to load product details. The system shows an error message.
Priority:	Medium
Frequency of Use:	As needed
Business Rules:	BR-20
Other Information:	None
Assumptions:	The system has up-to-date product information.

ID and Name:	UC-59 Update Cart		
Created By:	Đào Công An Phước	Date Created:	22/10/2024
Primary Actor:	Member	Secondary Actors:	None
Description:	A Member updates the contents of their shopping cart.		
Trigger:	The Member adds or removes items from the cart.		
Preconditions:	The Member must be logged in. The cart must exist.		
Postconditions:	The system updates the cart with the new contents.		
Normal Flow:	<ol style="list-style-type: none"> 1. The Member navigates to the "Cart" section. 2. The system displays the current contents of the cart. 3. The Member adds or removes items. 4. The Member submits the update. 5. The system validates and updates the cart. 		
Alternative Flows:	A1: The cart is empty. The system informs the Member.		

Exceptions:	E1: The update process fails due to a system error. The system notifies the Member.
Priority:	Medium
Frequency of Use:	As needed
Business Rules:	BR-20
Other Information:	None
Assumptions:	The Member is aware of the current contents of their cart.

ID and Name:	UC-60 View Cart		
Created By:	Đào Công An Phước	Date Created:	22/10/2024
Primary Actor:	Member	Secondary Actors:	None
Description:	A Member views the current contents of their shopping cart.		
Trigger:	The Member selects "View Cart."		
Preconditions:	The Member must be logged in. The cart must exist.		
Postconditions:	The system displays the current items in the cart.		
Normal Flow:	<ol style="list-style-type: none"> 1. The Member navigates to the "Cart" section. 2. The system displays the current contents of the cart. 		
Alternative Flows:	None.		
Exceptions:	E1: The system fails to load product details. The system shows an error message.		

Priority:	Medium
Frequency of Use:	As needed
Business Rules:	BR-20
Other Information:	None
Assumptions:	The cart contains products.

ID and Name:	UC-61 Add to Cart		
Created By:	Đào Công An Phước	Date Created:	22/10/2024
Primary Actor:	Member	Secondary Actors:	None
Description:	A Member adds a product to their shopping cart.		
Trigger:	The Member selects a product and chooses to add it to the cart.		
Preconditions:	<p>The Member must be logged in.</p> <p>The selected product must be available.</p>		
Postconditions:	The system adds the selected product to the cart.		
Normal Flow:	<ol style="list-style-type: none"> 1. The Member views a product's details. 2. The Member selects the "Add to Cart" option. 3. The system adds the product to the cart and confirms the action. 		
Alternative Flows:	None.		
Exceptions:	E1: The product is out of stock. The system notifies the Member.		
Priority:	Medium		

Frequency of Use:	As needed
Business Rules:	BR-20
Other Information:	None
Assumptions:	The product is available for purchase.

ID and Name:	UC-62 Place Order for Checkout		
Created By:	Đào Công An Phước	Date Created:	22/10/2024
Primary Actor:	Member	Secondary Actors:	None
Description:	A Member places an order for the items in their cart.		
Trigger:	The Member chooses to proceed to checkout.		
Preconditions:	The Member must be logged in. The cart must contain items.		
Postconditions:	The system processes the order and prepares it for payment.		
Normal Flow:	<ol style="list-style-type: none"> 1. The Member navigates to the cart. 2. The system displays the cart contents. 3. The Member selects "Checkout." 4. The system displays the checkout page. 5. The Member confirms their shipping address and payment method. 6. The Member submits the order. 7. The system processes the order. 		
Alternative Flows:	A1: The cart is empty. The system informs the Member		
Exceptions:	E1: The order process fails due to a system error. The system notifies the Member.		

Priority:	High
Frequency of Use:	As needed
Business Rules:	BR-20
Other Information:	None
Assumptions:	The Member confirms the order details.

ID and Name:	UC-63 Select Payment Method		
Created By:	Đào Công An Phước	Date Created:	22/10/2024
Primary Actor:	Member	Secondary Actors:	None
Description:	A Member selects a payment method during the checkout process.		
Trigger:	The Member proceeds to the payment section.		
Preconditions:	<p>The Member must be logged in.</p> <p>The cart must contain items.</p>		
Postconditions:	The system records the selected payment method for the order.		
Normal Flow:	<ol style="list-style-type: none"> 1. The Member navigates to the payment section during checkout. 2. The system displays available payment options. 3. The Member selects a preferred payment method. 4. The system records the selection. 		
Alternative Flows:	None		
Exceptions:	E1: The system fails to load payment options. The system shows an error message.		

Priority:	Medium
Frequency of Use:	As needed
Business Rules:	BR-20
Other Information:	None
Assumptions:	The payment options are correctly configured.

ID and Name:	UC-64 Select Shipping Address		
Created By:	Đào Công An Phước	Date Created:	22/10/2024
Primary Actor:	Member	Secondary Actors:	None
Description:	A Member selects a shipping address for their order.		
Trigger:	The Member proceeds to the shipping address section during checkout.		
Preconditions:	<p>The Member must be logged in.</p> <p>The cart must contain items.</p>		
Postconditions:	The system records the selected shipping address for the order.		
Normal Flow:	<ol style="list-style-type: none"> 1. The Member navigates to the shipping address section during checkout. 2. The system displays available shipping addresses. 3. The Member selects a preferred shipping address. 4. The system records the selection. 		
Alternative Flows:	A1: No addresses are saved. The system prompts the Member to enter a new address.		
Exceptions:	E1: The system fails to load shipping addresses. The system shows an error message.		

Priority:	Medium
Frequency of Use:	As needed
Business Rules:	BR-20
Other Information:	None
Assumptions:	The Member has previously saved addresses.

ID and Name:	UC-65 View Food Salt Calculator		
Created By:	Đào Công An Phước	Date Created:	22/10/2024
Primary Actor:	Member	Secondary Actors:	None
Description:	A Member views the food salt calculator tool for Koi fish care.		
Trigger:	The Member selects the "Food Salt Calculator" option.		
Preconditions:	The Member must be logged in.		
Postconditions:	The system displays the food salt calculator.		
Normal Flow:	<ol style="list-style-type: none"> 1. The Member navigates to the "Tools" section. 2. The Member selects the "Food Salt Calculator." 3. The system displays the calculator tool. 		
Alternative Flows:	None		
Exceptions:	E1: The system fails to load the calculator. The system shows an error message.		
Priority:	Low		

Frequency of Use:	Occasionally
Business Rules:	BR-20
Other Information:	None
Assumptions:	The calculator is functioning correctly.

ID and Name:	UC-66 View Statistic		
Created By:	Đào Công An Phước	Date Created:	22/10/2024
Primary Actor:	Member	Secondary Actors:	None
Description:	A Member views statistics related to their Koi fish and ponds.		
Trigger:	The Member selects the "Statistics" option.		
Preconditions:	The Member must be logged in.		
Postconditions:	The system displays relevant statistics.		
Normal Flow:	<ol style="list-style-type: none"> 1. The Member navigates to the "Statistics" section. 2. The system retrieves and displays statistics for the Member's ponds and Koi fish. 		
Alternative Flows:	None		
Exceptions:	E1: The system fails to load statistics. The system shows an error message.		
Priority:	Medium		
Frequency of Use:	Occasionally		

Business Rules:	BR-20
Other Information:	None
Assumptions:	The Member confirms the order details.

ID and Name:	UC-67: Update Promotion		
Created By:	Nguyễn Văn Minh Thoại	Date Created:	22/10/2024
Primary Actor:	Admin	Secondary Actors:	None
Description:	Admin updates a status promotion.		
Trigger:	Admin wants to modify a promotion.		
Preconditions:	Admin is logged into the system.		
Postconditions:	The promotion details are updated successfully.		
Normal Flow:	<ol style="list-style-type: none"> Admin navigates to the list of promotions. Admin selects a promotion to update. Admin modifies the promotion status. Admin saves the changes. The system confirms the successful update. 		
Alternative Flows:	None		
Exceptions:	E1: Promotion not found. The system displays an error message.		
Priority:	Medium		

Frequency of Use:	Regularly used for marketing activities.
Business Rules:	BR -5
Other Information:	None
Assumptions:	None

ID and Name:	UC-68 Update Member		
Created By:	Nguyễn Văn Minh Thoại	Date Created:	22/10/24
Primary Actor:	Admin	Secondary Actors:	None
Description:	Admin updates an existing member's status.		
Trigger:	Admin wants to modify a member's information.		
Preconditions:	Admin is logged into the system.		
Postconditions:	The member's information is updated successfully.		
Normal Flow:	<ol style="list-style-type: none"> 1. Admin navigates to the list of members. 2. Admin selects a member to update. 3. Admin modifies the member details. 4. Admin saves the changes. 5. The system confirms the successful update. 		
Alternative Flows:	None		
Exceptions:	E1: Member not found. The system displays an error message.		

Priority:	Medium
Frequency of Use:	Regularly used for member management.
Business Rules:	BR-18, BR-19
Other Information:	None
Assumptions:	None

ID and Name:	UC-69 View Staff		
Created By:	Nguyễn Văn Minh Thoại	Date Created:	22/10/24
Primary Actor:	Admin	Secondary Actors:	None
Description:	Admin views the list of staff members.		
Trigger:	Admin wants to view all registered staff.		
Preconditions:	Admin is logged into the system.		
Postconditions:	Admin can see the list of staff members.		
Normal Flow:	<ol style="list-style-type: none"> 1. Admin navigates to the staff section. 2. Admin selects "View Staff." 3. The system displays the list of staff members. 		
Alternative Flows:	None		
Exceptions:	E1: No staff found. The system displays a message indicating this.		

Priority:	Medium
Frequency of Use:	Regularly used for staff management.
Business Rules:	BR-19
Other Information:	None
Assumptions:	None

ID and Name:	UC-70 Create Staff		
Created By:	Nguyễn Văn Minh Thoại	Date Created:	22/10/24
Primary Actor:	Admin		
Description:	Admin creates a new staff member.		
Trigger:	Admin wants to add a new staff member to the system.		
Preconditions:	Admin is logged into the system.		
Postconditions:	A new staff member is created and listed successfully.		
Normal Flow:	<ol style="list-style-type: none"> 1. Admin navigates to the staff creation page. 2. Admin enters staff details (name, role, contact information, etc.). 3. Admin saves the new staff member. 4. The system confirms the successful creation. 		
Alternative Flows:	None		

Exceptions:	E1: Required staff information is missing. The system displays an error message.
Priority:	Medium
Frequency of Use:	Regularly used for staff management.
Business Rules:	BR-19
Other Information:	None
Assumptions:	None

ID and Name:	UC-71 Update Staff		
Created By:	Nguyễn Văn Minh Thoại		
Primary Actor:	Admin		
Description:	Admin wants to modify a staff member's details.		
Trigger:	Admin wants to view all registered staff.		
Preconditions:	Admin is logged into the system.		
Postconditions:	The staff member's information is updated successfully.		
Normal Flow:	<ol style="list-style-type: none"> 1. Admin navigates to the list of staff members. 2. Admin selects a staff member to update. 3. Admin modifies the staff details. 4. Admin saves the changes. 5. The system confirms the successful update. 		

Alternative Flows:	None
Exceptions:	E1: Staff member not found. The system displays an error message..
Priority:	Medium
Frequency of Use:	Regularly used for staff management.
Business Rules:	BR-19
Other Information:	None
Assumptions:	None

ID and Name:	UC-72 Delete Staff		
Created By:	Nguyễn Văn Minh Thoại	Date Created:	22/10/24
Primary Actor:	Admin	Secondary Actors:	None
Description:	Admin deletes an existing staff member.		
Trigger:	Admin wants to remove a staff member from the system.		
Preconditions:	Admin is logged into the system.		
Postconditions:	The staff member is deleted successfully.		

Normal Flow:	<ol style="list-style-type: none"> 1. Admin navigates to the list of staff members. 2. Admin selects a staff member to delete. 3. Admin confirms the deletion. 4. The system confirms the successful deletion.
Alternative Flows:	None
Exceptions:	E1: Staff member not found. The system displays an error message
Priority:	Medium
Frequency of Use:	Regularly used for member management.
Business Rules:	BR-19
Other Information:	None
Assumptions:	None

3. Business Rules

ID	Rule Definition	Type of Rule
BR-1	<p>The price of products (e.g., treatments, food) is calculated using the formula: Sale Price = Product Cost * Markup Rate - Promotion.</p> <p>Product Cost includes the cost of raw materials, manufacturing, and logistics.</p>	Computation
BR-2	A product can be added to the cart via product code or search functionality.	Constraint
BR-3	Members can view available products and add them to the cart.	Constraint
BR-4	Discounts applied to products must follow promotional policies.	Constraint
BR-5	Promotions must be applied according to system rules and can only be added during a valid promotion period.	Constraint

BR-6	The system must allow members to place orders and track order statuses.	Constraint
BR-7	Orders cannot be updated after confirmation unless authorized by an admin.	Constraint
BR-8	Members can create, update, and delete Koi fish records (name, age, weight, pond association).	Constraint
BR-9	Growth tracking for Koi fish must include weight and size updates over time.	Constraint
BR-10	Pond water parameters must be updated at least weekly to maintain water quality.	Constraint
BR-11	The system must calculate recommended salt and food amounts based on pond volume and Koi fish growth stage.	Computation
BR-12	Members can set reminders for regular pond maintenance tasks like feeding or cleaning.	Constraint
BR-13	Reminder notifications are sent via WebSocket 5 minutes before the scheduled time.	Constraint
BR-14	Reminders not viewed within the set time must be stored as unread notifications.	Constraint
BR-15	Only Shops can create and manage blog posts related to Koi care	Constraint
BR-16	Blogs can include tips, promotions, and product features for treating Koi health and improving water quality.	Constraint
BR-17	Shops can categorize blog content using tags (create, update, delete).	Constraint
BR-18	Members can manage their own profile, including updating contact information and preferences.	Constraint
BR-19	Only Admins can manage other users, including creating, updating, or deleting user profiles.	Constraint
BR-20	Admins have full access to system settings and can update water parameter rules and product information.	Constraint
BR-21	Shops can manage product listings and promotions but cannot modify other users' information.	Constraint
BR-22	When water parameters (e.g., pH, temperature) exceed safe ranges, the system automatically generates an issue and suggests products to resolve it.	Computation
BR-23	The system must allow members to track historical water parameter data for each pond	Constraint
BR-24	Payment options for orders include credit card and PayPal, with secure verification required.	Constraint
BR-25	Members can only place orders once their cart is verified, and payment has been authorized.	Constraint
BR-26	The system must recommend health treatments based on common Koi health issues.	Computation
BR-27	Shops must provide product details, including price, stock availability, and associated health treatments.	Constraint

BR-28

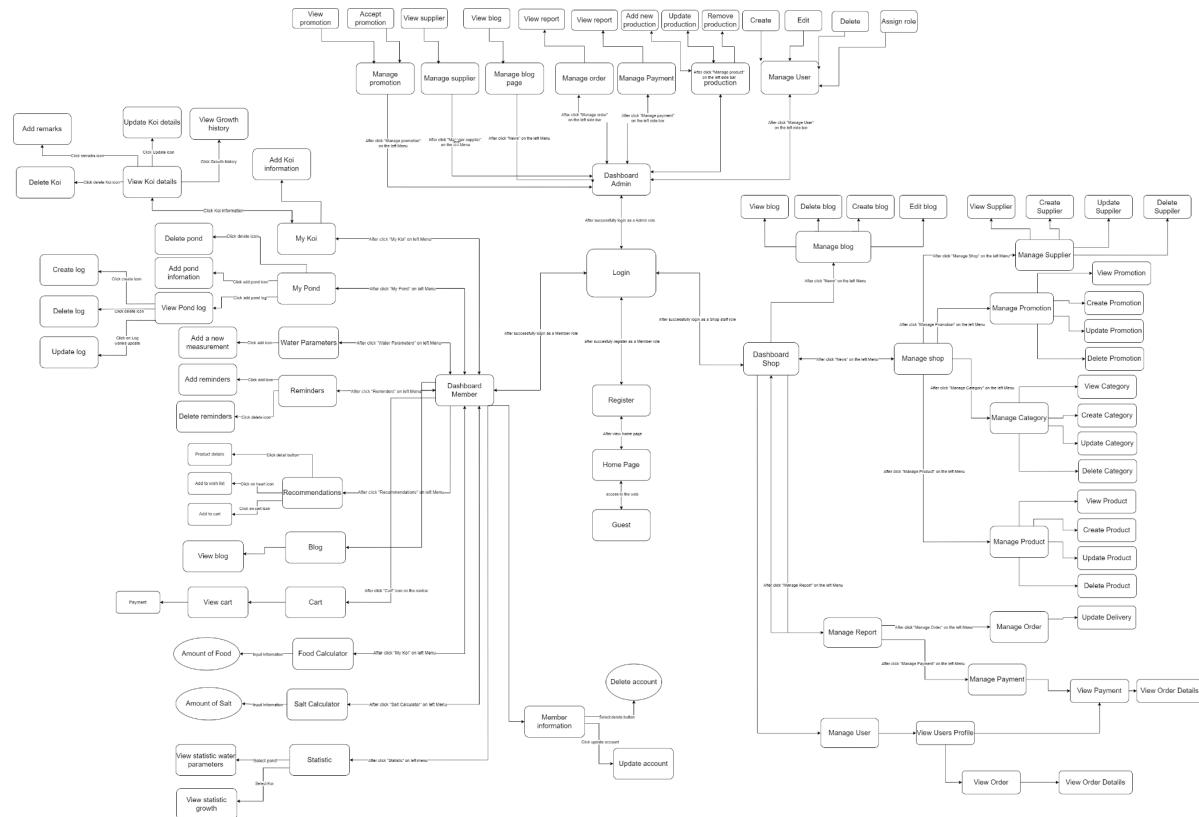
Products that are out of stock must be updated daily by the Shop.

Constraint

IV. Functional Requirements

1. System Functional Overview

1.1 Screen Flow



1.2 Screen Descriptions

#	Screen	Feature	Description
FR-01	Homepage	Introduction about web app	Introduction about the web app
FR-02	Login	Account Access	Authenticate a user's identity before they access application or system security functions or data.
FR-03	Register	Account register	Register a new account

FR-04	Dashboard Member	Dashboard	Member's dashboard overview (pond, order, koi fish, ...)
FR-05	My Koi	List Koi	Display list of koi
FR-06	Create Koi	Koi Management	Manage koi creation
FR-07	Koi details	View Koi, View Growth history, View remark	View details, growth history, and remarks for koi
FR-08	Update Koi	Update Koi details	Update koi information
FR-09	Create growth history	Create growth history	Create new growth history
FR-10	Update growth history	Update existing growth history	Update existing growth history
FR-11	Create remark	Create remark	Create new remark
FR-12	Update remark	Update existing remark	Update existing remark
FR-13	My Pond	List pond	Display list of ponds
FR-14	Create pond	Pond Management	Manage pond creation
FR-15	Update pond	Update pond details	Update pond details

FR-16	Pond Log	List log	Display a list of pond logs
FR-17	Create pond log	Pond log management	Manage creation of pond logs
FR-18	Update pond log	Update pond log detail	Update pond log details
FR-19	Water parameters	List water parameters	Display a list of water parameters
FR-20	Create water parameter	Water parameter management	Manage water parameter creation
FR-21	Edit water parameter	Edit water parameter	Edit water parameter details
FR-22	Reminders	List reminders	Display list of reminders
FR-23	Add reminders	Reminder management	Manage reminders
FR-24	Recommendations	List products	List recommended products
FR-25	Product details	View product	View detailed information about products
FR-26	Blog	List blog	Display list of blog posts
FR-27	Blog details	View blog details	View blog post details

FR-28	Cart	View all item in cart	View items in the shopping cart
FR-29	Checkout	Information about order	Order details and checkout process
FR-30	Food Calculator	Calculate the amount of food	Calculate required food for koi
FR-31	Salt Calculator	Calculate the amount of salt	Calculate the amount of salt needed
FR-32	Statistic	View statistic about koi pond and koi fish	View statistics on koi ponds and koi fish
FR-33	Member information	View member profile, Edit profile, Change password	View and update member profile, including changing passwords
FR-34	Dashboard Shop	Dashboard	Shop staff dashboard overview
FR-35	Manage User	View list users	Display a list of users profile
FR-36	View user profile	View user profile	View a user's profile
FR-37	View user's payment details	View payment details	View payment details for users
FR-38	View user's order details	View order details	View users' order details

FR-39	Manager Order	View list orders, Update Delivered status	List and manage orders, including updating delivery status
FR-40	Manager Payment	View list payments	List and manage payment details
FR-41	Manage blog tag	List blog tag	Display a list of blog tags
FR-42	Create tag	Create blog tag	Create a new blog tag
FR-43	Update tag	Update blog tag	Update existing blog tags
FR-44	Manage blog	List blog tag	List and manage blog tags
FR-45	Create blog	Create new blog	Create a new blog post
FR-46	Update blog	Update blog	Update an existing blog post
FR-47	View blog	View blog	View a blog post
FR-48	Manage supplier	List supplier	Display a list of suppliers
FR-49	Create supplier	Create new supplier	Create a new supplier
FR-50	Edit supplier	Edit supplier	Edit supplier details

FR-51	Manage promotion	List promotion	List and manage promotions
FR-52	Create promotion	Create new promotion	Create a new promotion
FR-53	Edit promotion	Edit promotion	Edit existing promotions
FR-54	Manage category	List product category	List product categories
FR-55	Create category	Create new category	Create a new product category
FR-56	Edit category	Edit category	Edit an existing category
FR-57	Manage image	List product images	Display list of product images
FR-58	Create image	Create new images for product	Add new product images
FR-59	Edit image	Edit image	Edit product image details
FR-60	Manage Product	List product	List and manage products
FR-61	Create product	Create new product	Create a new product
FR-62	Edit product	Edit product	Edit existing product details

FR-63	Dashboard Admin	Dashboard	Admin dashboard overview	
FR-64	Promotion Management	View list promotion, Accept/Reject promotion	View, accept, or reject promotions	
FR-65	Manage Shop Staff	List staff accounts	List and manage shop staff accounts	
FR-66	Create Shop staff account	Create Shop staff account	Create a new shop staff account	
FR-67	Change Shop account password	Change shop account password	Update passwords for shop accounts	
FR-68	Upgrade Premium for member	Update Premium for member	Upgrade member accounts to premium	
FR-69	About	Information about the development team	Information about the development team	

1.3 Screen Authorization

Screen	Guest	Shop Staff	Users	Admin
Homepage	X	X	X	X

Login	X	X	X	X
Register	X			
Dashboard Member			X	
My Koi			X	
Create Koi			X	
Koi details			X	
Update Koi			X	
Create growth history			X	
Update growth history			X	
Create remark			X	
Update remark			X	
My Pond			X	
Create pond			X	
Update pond			X	
Pond Log			X	
Create pond log			X	
Update pond log			X	

Water parameters			X	
Create water parameter			X	
Edit water parameter			X	
Reminders			X	
Add reminders			X	
Recommendations			X	
Product details			X	
Blog			X	
Blog details			X	
Cart			X	
Checkout			X	
Food Calculator			X	
Salt Calculator			X	
Statistic			X	
Member information			X	
Dashboard Shop		X		
Manage User	X			X

View user profile	X		X
View user's payment details	X		X
View user's order details	X		X
Manager Order	X		X
Manager Payment	X		X
Manage blog tag	X		
Create tag	X		
Update tag	X		
Manage blog	X		X
Create blog	X		
Update blog	X		
View blog	X		
Manage supplier	X		X
Create supplier	X		
Edit supplier	X		
Manage promotion	X		X
Create promotion	X		

Edit promotion	X			
Manage category	X			
Create category	X			
Edit category	X			
Manage image	X			
Create image	X			
Edit image	X			
Manage Product	X			X
Create product	X			
Edit product	X			
Dashboard Admin				X
Promotion Management				X
Manage Shop Staff				X
Create Shop staff account				X
Change Shop account password				X
Upgrade Premium for member				X

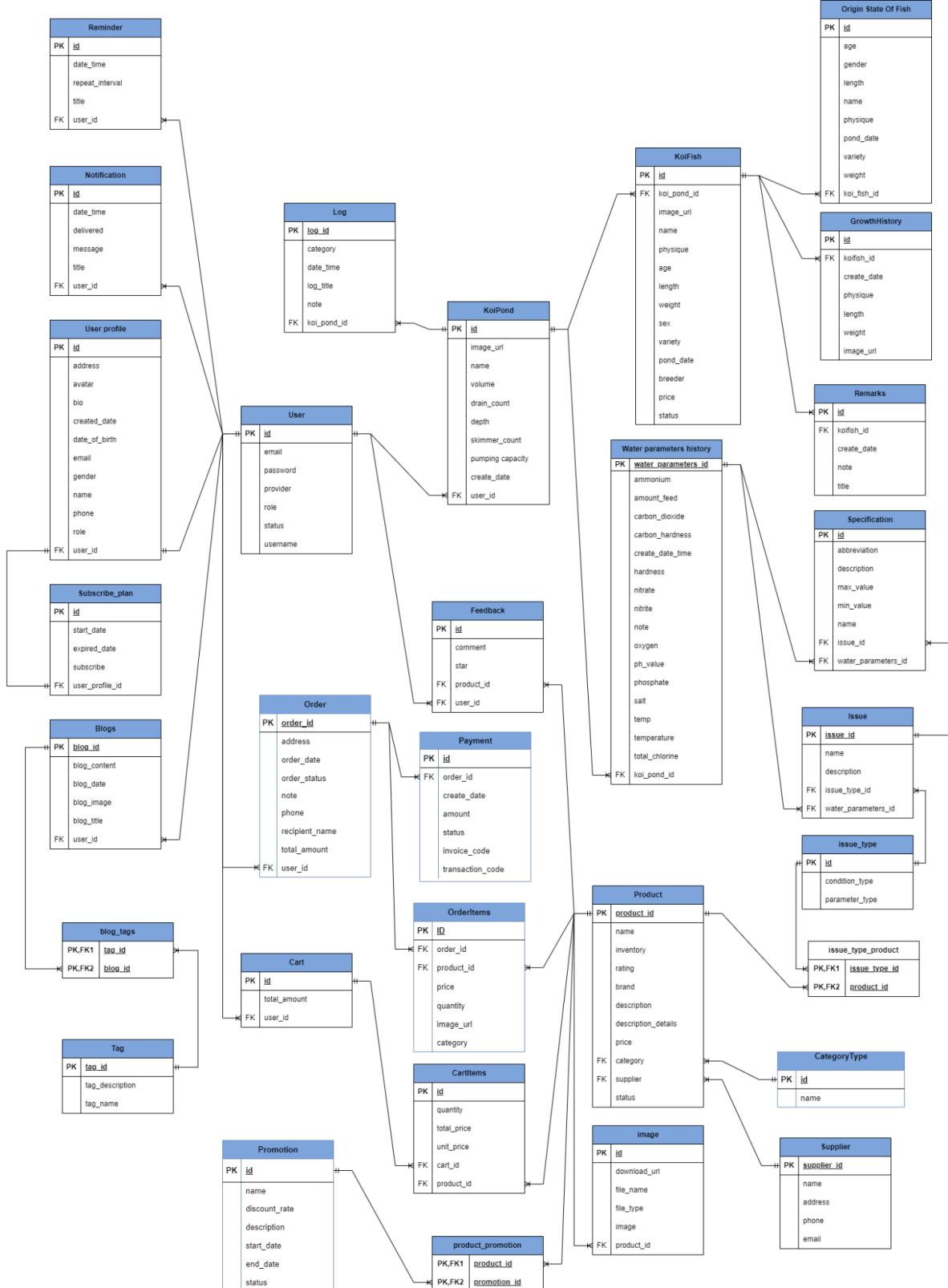
About			X	
-------	--	--	---	--

1.4 Non-Screen Functions

#	System function	Feature	Description
FR-70	Auto Update Promotion Status	Auto Update Promotion Status	Automatically Update Promotion status base on company policies

2. Data Requirements

2.1 Logical Data Model



2.2 Data Dictionary

Data Element	Description	Composition or Data Type	Length	Value
users.id	Unique identifier for each user	bigint		PK
users.email	User's email address	varchar	255	
users.password	Encrypted user password	varchar	255	
users.provider	The provider used for authentication	varchar	255	
users.role	User's role or permission level	varchar	255	
users.status	Active status of the user account	bit		
users.username	User's chosen display name	varchar	255	
user_profile.id	Unique identifier for each profile	bigint		PK
user_profile.address	User's physical address	varchar	255	
user_profile.avatar	URL or path to user's profile picture	varchar	255	
user_profile.bio	Short biography or description of the user	varchar	255	
user_profile.created_date	The date the profile was created	date		
user_profile.date_of_birth	User's date of birth	date		

user_profile.email	User's email address	varchar	255	
user_profile.gender	User's gender	varchar	255	
user_profile.name	Full name of the user	varchar	255	
user_profile.phone	User's contact number	varchar	255	
user_profile.role	Role or status assigned to the user	varchar	255	
user_profile.user_id	Foreign key linking to the users table	bigint		FK
water_parameters.id	Unique identifier for each water parameter record	bigint		PK
water_parameters.ammonium	Ammonium level in the water	float		
water_parameters.amount_fed	Amount of food fed to fish	float		
water_parameters.carbon_dioxide	Carbon dioxide level in the water	float		
water_parameters.carbon_hardness	Carbonate hardness level	float		
water_parameters.create_date_time	Date and time the parameters were recorded	datetime2(6)		
water_parameters.hardness	General hardness of the water	float		
water_parameters.nitrate	Nitrate concentration in the water	float		

water_parameters.nitrite	Nitrite concentration in the water	float		
water_parameters.note	Additional notes or comments	varchar	255	
water_parameters.oxygen	Oxygen concentration in the water	float		
water_parameters.ph_value	pH value of the water	float		
water_parameters.phosphate	Phosphate concentration in the water	float		
water_parameters.salt	Salinity of the water	float		
water_parameters.temp	Temperature of the water	float		
water_parameters.temperature	Duplicate or alternative temperature field	float		
water_parameters.total_chlorine	Total chlorine concentration in the water	float		
water_parameters.koi_pond_id	Foreign key linking to a koi pond	bigint		FK
supplier.id	Unique identifier for each supplier	bigint		PK
supplier.address	Supplier's physical address	varchar	255	
supplier.name	Name of the supplier	varchar	255	
supplier.phone	Supplier's contact phone number	varchar	255	

subscribe_plan.id	Unique identifier for each subscription plan	bigint		PK
Subscribe_plan.expired_date	Expiration date of the subscription plan	Date		
Subscribe_plan.start_date	Start date of the subscription plan	Date		
Subscribe_plan.subscribe	Details of the subscription plan	Varchar	255	
Subscribe_plan.user_profile_id	Foreign key linking to the user profile	Bigint		FK
Specification.id	Unique identifier for each specification	Bigint		PK
Specification.abbreviation	Abbreviation or shorthand for the specification	Varchar	255	
Specification.description	Detailed description of the specification	Varchar	255	
Specification.min_value	Minimum acceptable value for the specification	Float		
Specification.max_value	Maximum acceptable value for the specification	Float		
Specification.name	Name of the specification	Varchar	255	
Specification.issue_id	Foreign key linking to an issue record	Bigint		FK
Specification.water_parameter_id	Foreign key linking to water parameters	Bigint		FK

Reminder.id	Unique identifier for each reminder	Bigint		PK
Reminder.date_time	Date and time of the reminder	datetime		
Reminder.repeat_interval	Interval at which the reminder repeats	Varchar	255	
Reminder.title	Title of the reminder	Varchar	255	
Reminder.user_id	Foreign key linking to the user who set the reminder	Bigint		FK
Remark.id	Unique identifier for each remark	Bigint		PK
Remark.create_date	Date when the remark was created	Datetime		
Remark.note	Notes or additional details about the remark	Varchar	255	
Remark.title	Title or subject of the remark	Varchar	255	
Remark.koifish_id	Foreign key linking to a koi fish record	Bigint		FK
Promotion.id	Unique identifier for each promotion	Bigint		PK
Promotion.description	Description of the promotion	Varchar	255	
Promotion.discount_rate	Discount rate applied in the promotion	Float		

Promotion.start_date	Start date of the promotion	Datetime		
Promotion.end_date	End date of the promotion	Datetime		
Promotion.name	Name of the promotion	Varchar	255	
Promotion.status	Current status of the promotion	Varchar	255	
Product.id	Unique identifier for each product	Bigint		PK
Product.brand	Brand name of the product	Varchar	255	
Product.description	Short description of the product	Text		
Product.description_detail	Detailed description of the product	Text		
Product.inventory	Number of units available in stock	Int		
Product.name	Name of the product	Varchar	255	
Product.price	Price of the product	Numeric(38,2)		
Product.rating	User rating of the product	Float		
Product.status	Availability status of the product	Bit		
Product.category_id	Foreign key linking to the product category	Bigint		FK

Product.supplier_id	Foreign key linking to the supplier	Bigint		FK
Payment.id	Foreign key linking to the supplier	Bigint		PK
Payment.amount	Total amount of the payment	Float		
Payment.create_date	Date when the payment was created	Datetime		
Payment.invoice_code	Invoice code associated with the payment	Varchar	255	
Payment.status	Status of the payment	Varchar	255	
Payment.transaction_code	Code for tracking the transaction	Varchar	255	
Payment.order_id	Foreign key linking to the order associated with the payment	Bigint		FK
Origin_state_of_fish.id	Unique identifier for the state of fish record	Bigint		PK
Origin_state_of_fish.age	Age of the fish	Int		
Origin_state_of_fish.gender	Gender of the fish	Varchar	255	
Origin_state_of_fish.length	Length of the fish	Float		
Origin_state_of_fish.name	Name of the fish variety	Varchar	255	
Origin_state_of_fish.physique	Description of the fish's physique	Varchar	255	

Origin_state_of_fish.pond_date	Date when the fish was introduced into the pond	Date		
Origin_state_of_fish.variety	Variety of the koi fish	Varchar	255	
Origin_state_of_fish.weight	Weight of the fish	Float		
Origin_state_of_fish.koi_fish_id	Foreign key linking to a specific koi fish	Bigint		FK
Orders.id	Unique identifier for each order	Bigint		PK
Orders.address	Shipping address for the order	Varchar	255	
Orders.note	Additional notes or instructions for the order	Varchar	255	
Orders.order_date	Date when the order was placed	Datetime		
Orders.order_status	Current status of the order	Varchar	255	
Orders.phone	Contact phone number for the order recipient	Varchar	255	
Orders.recipient_name	Name of the person receiving the order	Varchar	255	
Orders.total_amount	Total monetary amount of the order	Numeric(38,2)		
Orders.user_id	Foreign key linking to the user who placed the order	Bigint		FK

Order_item.id	Unique identifier for each order item	Bigint		PK
Order_item.category	Category of the product ordered	Varchar	255	
Order_item.image_url	URL of the product image	Varchar	255	
Order_item.price	Price of the product ordered	Numeric(38,2)		
Order_item.quantity	Quantity of the product ordered	Int		
Order_item.order_id	Foreign key linking to the order	Bigint		FK
Order_item.product_id	Foreign key linking to the product	Bigint		FK
Notification.id	Unique identifier for each notification	Bigint		PK
Notification.date_time	Date and time when the notification was sent	Datetime		
Notification.delivered	Indicator of whether the notification was successfully delivered	Bit		
Notification.message	Message content of the notification	Varchar	255	
Notification.title	Title or subject of the notification	Varchar	255	

Notification.user_id	Foreign key linking to the user receiving the notification	Bigint		FK
Log.id	Unique identifier for each log entry	Bigint		PK
Log.category	Category or type of log entry	Varchar	255	
Log.date_time	Date and time when the log entry was created	Datetime		
Log.log_title	Title or subject of the log entry	Varchar	255	
Log.note	Notes or details associated with the log entry	Varchar	255	
Log.koi_pond_id	Foreign key linking to a koi pond record	Bigint		FK
Koi_pond.id	Unique identifier for each koi pond	Bigint		PK
Koi_pond.create_date	Date when the koi pond was created	Date		
Koi_pond.depth	Depth of the koi pond in meters	Float		
Koi_pond.drain_count	Number of times the pond has been drained	Int		
Koi_pond.image_url	URL to the image of the koi pond	Varchar	255	

Koi_pond.name	Name assigned to the koi pond	Varchar	255	
Koi_pond.pump_capacity	Water pump capacity in the koi pond	Float		
Koi_pond.skimmer	Number of skimmers in the koi pond	Int		
Koi_pond.volume	Volume of water in the pond (liters)	Int		
Koi_pond.user_id	Foreign key linking to the user who owns the pond	Bigint		FK
Koi_fish.id	Unique identifier for each koi fish	Bigint		PK
Koi_fish.age	Age of the koi fish	Int		
Koi_fish.breeder	Name of the koi fish breeder	Varchar	255	
Koi_fish.gender	Gender of the koi fish	Varchar	255	
Koi_fish.image_url	URL to the image of the koi fish	Varchar	255	
Koi_fish.length	Length of the koi fish (cm)	Float		
Koi_fish.name	Name of the koi fish	Varchar	255	
Koi_fish.physique	Description of the koi fish's physique	Varchar	255	
Koi_fish.pond_date	Date the koi fish was placed in the pond	Date		

Koi_fish.price	Price of the koi fish	Float		
Koi_fish.status	Current health or availability status of the koi fish	Varchar	255	
Koi_fish.variety	Variety or breed of the koi fish	Varchar	255	
Koi_fish.weight	Weight of the koi fish (kg)	Float		
Koi_fish.koi_pond_id	Foreign key linking to the koi pond where the fish is housed	Bigint		FK
Issue_type.id	Unique identifier for each issue type	Bigint		PK
Issue_type.condition_type	Type of condition associated with the issue	Varchar	255	
Issue_type.parameter_type	Type of parameter linked to the issue	Varchar	255	
Issue.id	Unique identifier for each issue	Bigint		PK
Issue.description	Description of the issue	Varchar	255	
Issue.name	Name assigned to the issue	Varchar	255	
Issue.issue_type_id	Foreign key linking to the issue type	Bigint		FK
Issue.water_parameters_id	Foreign key linking to the water parameters associated with the issue	Bigint	255	FK

Image.id	Unique identifier for each image	Bigint		PK
Image.download_url	URL to download the image	Varchar	255	
Image.file_name	Name of the image file	Varchar	255	
Image.file_type	Type of image file	Varchar	255	
Image.image	Binary data for the image	Varbinary		
Image.product_id	Foreign key linking to the associated product	Bigint		FK
Growth_history.id	Unique identifier for each growth history record	Bigint		PK
Growth_history.create_date	Date when the growth history record was created	Datetime		
Growth_history.image_url	URL to the image documenting growth	Varchar	255	
Growth_history.length	Length of the koi fish recorded at the time	Float		
Growth_history.physique	Description of the koi fish's physique at the time	Varchar	255	
Growth_history.weight	Weight of the koi fish at the time	Float		
Growth_history.koifish_id	Foreign key linking to the koi fish	Bigint		FK

Feedback.id	Unique identifier for each feedback record	Bigint		PK
Feedback.comment	Customer's comment or review of the product	Text		
Feedback.star	Star rating given by the customer	Int		
Feedback.product_id	Foreign key linking to the product	Bigint		FK
Feedback.user_id	Foreign key linking to the user who gave the feedback	Bigint		FK
Category.id	Unique identifier for each product category	Bigint		PK
Category.name	Name of the product category	Varchar	255	
Cart_item.id	Unique identifier for each cart item	Bigint		PK
Cart_item.quantity	Quantity of the product in the cart	Int		
Cart_item.total_price	Total price of the cart item	Numeric(38,2)		
Cart_item.unit_price	Unit price of the product	Numeric(38,2)		
Cart_item.cart_id	Foreign key linking to the cart	Bigint		FK
Cart_item.product_id	Foreign key linking to the product	Bigint		FK

Cart.id	Unique identifier for each cart	Bigint		PK
Cart.total_amount	Total monetary amount of the cart	Numeric(38,2)		
Cart.user_id	Foreign key linking to the user who owns the cart	Bigint		FK
Blogs.blog_id	Unique identifier for each blog post	Bigint		PK
Blogs.blog_content	Content or body of the blog post	Text		
Blogs.blog_date	Date when the blog post was published	Date		
Blogs.blog_image	URL to the image associated with the blog post	Varchar	255	
Blogs.blog_title	Title of the blog post	Varchar	255	
Blogs.user_id	Foreign key linking to the user who wrote the blog	Bigint		FK

2.3 Reports

Dashboard Member		
#	Name	Description

1	Koi Overview	<p>Koi overview helps evaluate the number of member's koi, pond, water parameters, reminders and orders . It includes data on:</p> <ul style="list-style-type: none"> + Ponds + Koi + Water parameters + Reminders + Orders
2	Recent Transaction	<p>Recent transactions help evaluate the member's transaction. It includes data on:</p> <ul style="list-style-type: none"> + Transaction code + Time + Total amount
3	Payment Details	<p>Payment details help evaluate the list of product in payment, including:</p> <ul style="list-style-type: none"> + Name of product + Quantity + Price

Report ID	RPT-01
Report Title	Koi Overview
Report Purpose	Koi overview helps evaluate the number of member's koi, pond, water parameters, reminders and orders .

Priority	High
Report Users:	Member
Data Sources	Database from koi fish, koi pond, water parameter, reminder, order
Visual Layout	Five block
Report Body	<ul style="list-style-type: none"> + Ponds + Koi + Water parameters + Reminders + Orders

Report ID	RPT-02
Report Title	Recent Transaction
Report Purpose	Recent transactions help evaluate the member's transaction.
Priority	Medium
Report Users:	Member
Data Sources	Database from payment
Visual Layout	List of transaction

Report Body	<ul style="list-style-type: none"> + Transaction code + Time + Total amount
-------------	--

Report ID	RPT-03
Report Title	Payment Details
Report Purpose	Payment details help evaluate the list of product in payment
Priority	Medium
Report Users:	Member
Data Sources	Database from payment
Visual Layout	Table
Report Body	<ul style="list-style-type: none"> + Name of product + Quantity + Price

Dashboard shop		
#	Name	Description

1	Top Products	<p>Top products help evaluate the percent by quantity of product that has been bought. It includes data on:</p> <ul style="list-style-type: none"> + Product image + Product name + Percent + Total price
2	Order and Payment Analytics	<p>Order and Payment Analytics help evaluate the number of order and number of payment have been paid, including:</p> <ul style="list-style-type: none"> + Number of order + Number of payment + Time
3	Earning by Category	<p>Earning by category help evaluate the percent of category of product has been bought, including:</p> <ul style="list-style-type: none"> + Total price of category product
4	User Report	<p>User report present the number of users and premium users, including:</p> <ul style="list-style-type: none"> + Number of users + Number of premium
5	New Orders	<p>New orders help evaluate the new orders in a day, including :</p> <ul style="list-style-type: none"> + Order Id + Date + Recipient name + Status + Total price

Dashboard Admin		
#	Name	Description
1	System Overview	<p>System overview helps evaluate the number of orders in a day, total premium user, premium subscribers in a day, revenue in a month. It includes data on:</p> <ul style="list-style-type: none"> + Order today + Premium user + Revenue + Subscribers today
2	Top Products	<p>Top products help evaluate the percent by quantity of product that has been bought. It includes data on:</p> <ul style="list-style-type: none"> + Product image + Product name + Percent + Total price
3	Order and Payment Analytics	<p>Order and Payment Analytics help evaluate the number of order and number of payment have been paid, including:</p> <ul style="list-style-type: none"> + Number of order + Number of payment + Time
4	Earning by Category	<p>Earning by category help evaluate the percent of category of product has been bought, including:</p> <ul style="list-style-type: none"> + Total price of category product

5	User Report	<p>User report present the number of users and premium users, including:</p> <ul style="list-style-type: none"> + Number of users + Number of premium
6	New Orders	<p>New orders help evaluate the new orders in a day, including :</p> <ul style="list-style-type: none"> + Order Id + Date + Recipient name + Status + Total price

Report ID	RPT-04
Report Title	System Overview
Report Purpose	System overview helps evaluate the number of orders in a day, total premium user, premium subscribers in a day, revenue in a month
Priority	High
Report Users:	Admin
Data Sources	Database from order, subscribe plan, payment
Visual Layout	Four block

Report Body	<ul style="list-style-type: none"> + Order today + Premium user + Revenue + Subscribers today
-------------	---

Report ID	RPT-05
Report Title	Top Products
Report Purpose	Top products help evaluate the percent by quantity of product that has been bought
Priority	High
Report Users:	Admin, Shop staff
Data Sources	Database from order
Visual Layout	Table
Report Body	<ul style="list-style-type: none"> + Product image + Product name + Percent + Total price

Report ID	RPT-06
Report Title	Order and Payment Analytics

Report Purpose	Order and Payment Analytics help evaluate the number of order and number of payment have been paid
Priority	High
Report Users:	Admin, Shop staff
Data Sources	Database from order, payment
Visual Layout	Bar chart
Report Body	<ul style="list-style-type: none"> + Number of order + Number of payment + Time

Report ID	RPT-07
Report Title	Earning by Category
Report Purpose	Earning by category help evaluate the percent of category of product has been bought
Priority	High
Report Users:	Admin, Shop staff
Data Sources	Database from order

Visual Layout	Pie chart
Report Body	+ Total price of category product

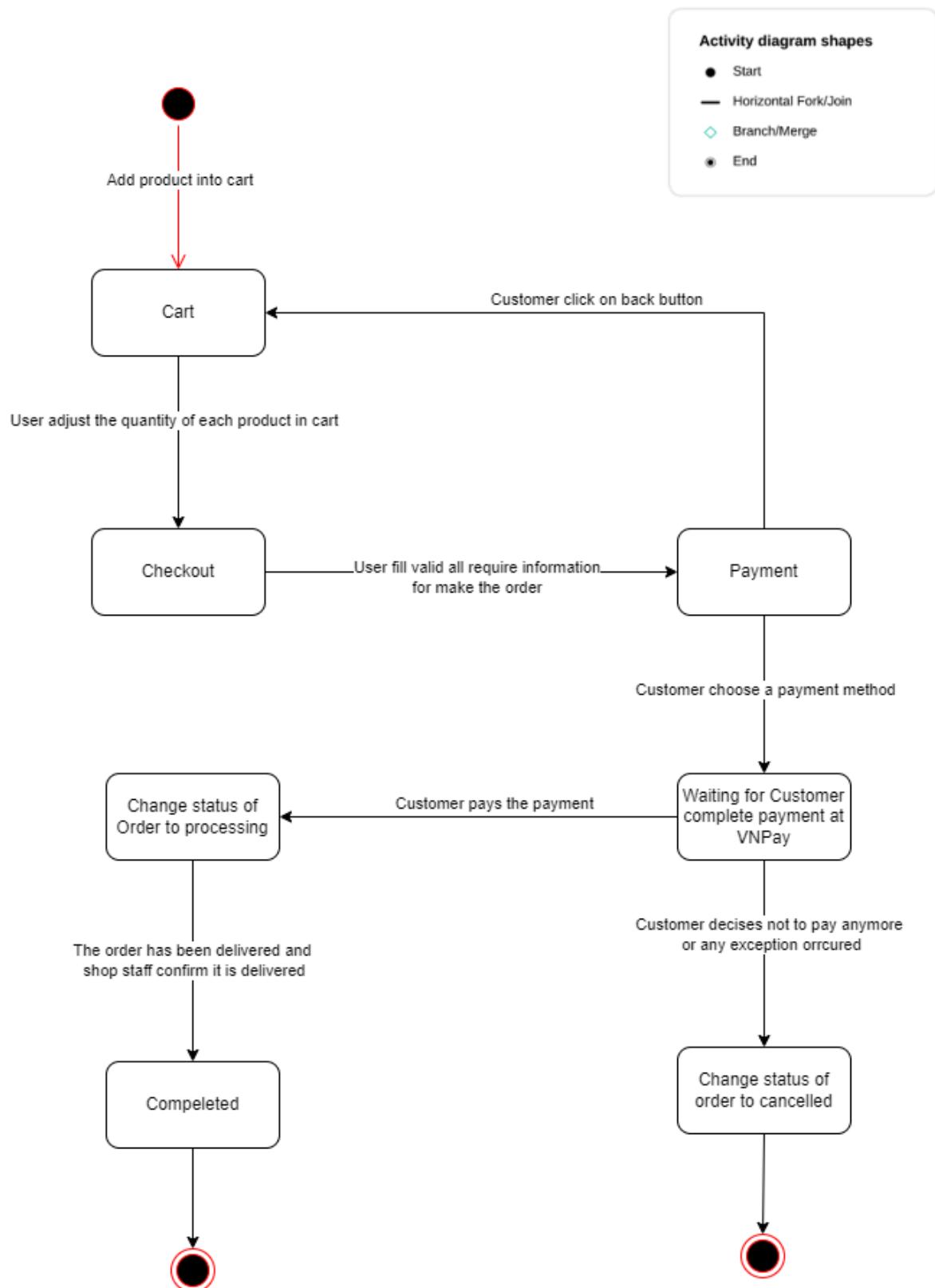
Report ID	RPT-08
Report Title	User Report
Report Purpose	User report present the number of users and premium users
Priority	High
Report Users:	Admin, Shop staff
Data Sources	Database from users, subscribe plan
Visual Layout	Bar chart
Report Body	+ Number of users + Number of premium

Report ID	RPT-09
-----------	--------

Report Title	New Orders
Report Purpose	New orders help evaluate the new orders in a day
Priority	High
Report Users:	Admin, Shop staff
Data Sources	Database from order
Visual Layout	Table
Report Body	<ul style="list-style-type: none"> + Order Id + Date + Recipient name + Status + Total price

2.4 State transition diagram

2.4.1 Buy recommendations product



3. Home Page

3.1 FR-01: HomePage

Function Trigger:

The function is triggered when any user or visitor accesses the Koi Care System via a web browser by entering the system's URL.

Function Description:

- Actors/Roles: Guest , Member, Shop, Admin
- Purpose: To provide users with an introduction to the Koi Care System, an overview of key features, and access to different sections of the system based on their role and preferences.
- Interface: Main HomePage

Data Processing:

Inputs:

- None required for anonymous users or visitors.
- Role-based navigation and content (Admin, Shop, Member) if logged in.

Outputs:

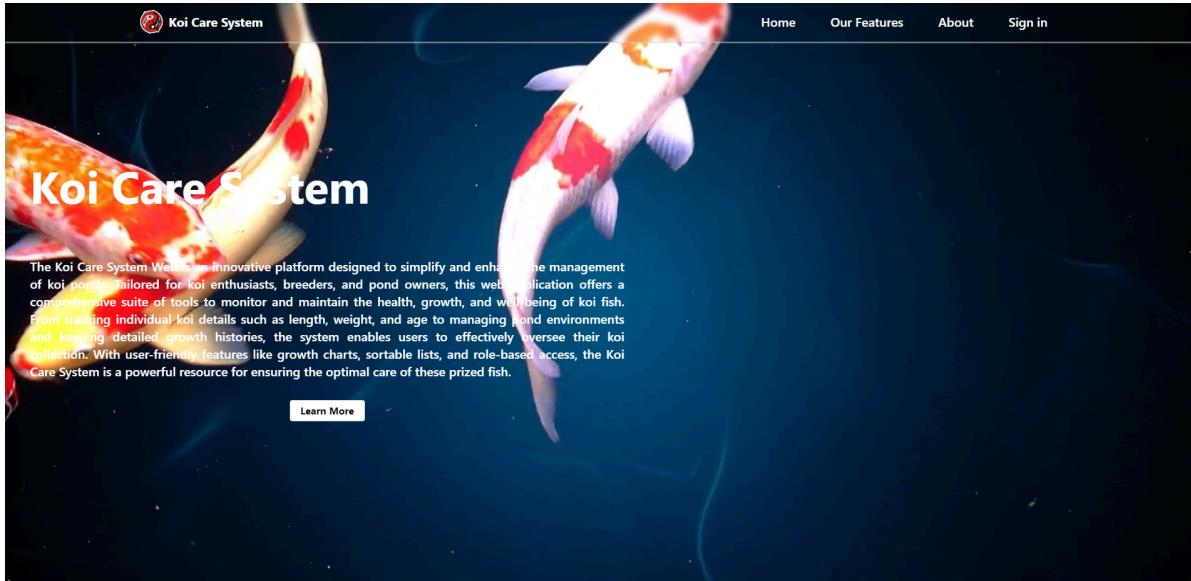
- General information about koi care, system features, and services.
- Quick links to user-specific features (for logged-in users) or call-to-action buttons like "Login," "Register," or "Learn More" .

Process:

1. The system presents a public homepage to all users.
2. The homepage is dynamic, adjusting content based on whether the user is logged in or a guest.
3. Provide an overview of koi care practices, system features, and benefits of using the platform.
4. Present call-to-action buttons like “Login”, “Register” to invite new users.
5. The homepage includes featured koi, tips for pond care, and a section for membership or loyalty programs.

Screen Layout:

System logo, navigation links (Home, Features, About, Sign in) for guest



Function Details:

- None for guest.

Business Logic:

- **Normal Cases:** The home page loads, providing relevant information and options based on the user's role.
- **Abnormal Cases:** If there's an issue loading content (e.g., missing data), an error message is displayed.

4. Login Page

4.1 FR-02: Login

Function Trigger:

The function is triggered when a user attempts to access their account or any restricted area of the system by selecting the "Sign in" button from the home page or any other interface requiring authentication.

Function Description:

- **Actors/Roles:** Admin, Shop, Member, Guest
- **Purpose:** To authenticate a user's identity and ensure that only authorized users can access secure application functions, such as koi management, shop features, or reporting tools.
- **Interface:** Login Screen

Data Processing:

Inputs:

- Username
- Password

- Login by Google

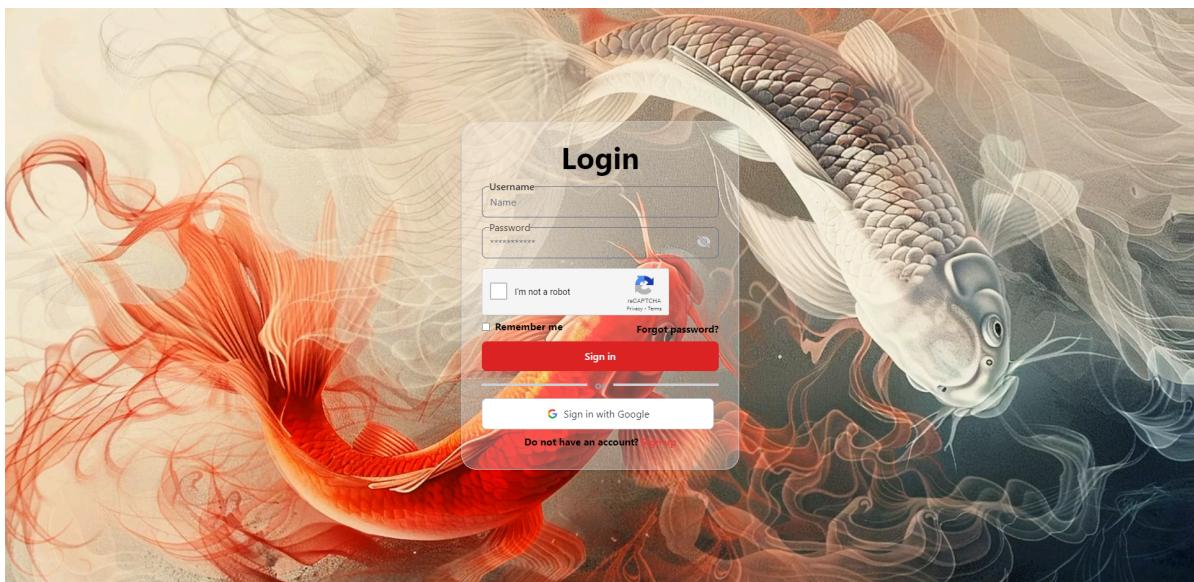
Outputs:

- Authentication Token upon successful login
- Error message in case of failed login attempt

Process:

1. The user enters their Username and Password on the Login screen.
2. The system validates the credentials by checking the inputted data against stored user information in the database.
3. If the credentials are valid: The system generates an authentication token and stores it in the user's session and the user is redirected to the appropriate dashboard or homepage based on their role (Admin, Shop , Member).
4. If the credentials are invalid: An error message is displayed to the user ("Invalid username or password. Please try again.").

Screen Layout:



Function Details:

Data:

- User credentials (username/email, password)

Validation:

- The Username and Password fields must not be empty.
- The Password must meet the system's security requirements (minimum length: 8 characters, with at least 1 special character).

Security:

- Passwords are encrypted and stored securely in the database.
- An authentication token is generated after a successful login, allowing role-based access to the system's features.

Business Logic:

- **Normal Cases:** Upon successful login, the user is redirected to the appropriate role-specific dashboard (e.g., Admin Dashboard, Shop Dashboard, Member Dashboard).
- **Abnormal Cases:** For invalid login attempts, an error message is displayed and the attempt is logged.

4.2 FR-03: Register

Function Trigger:

The function is triggered when a user selects the "Register" button from the homepage or the login page to create a new account.

Function Description:

- **Actors/Roles:** Guest
- **Purpose:** To allow new users to register for an account and access the Koi Care System's features according to their roles.
- **Interface:** Registration Screen

Data Processing:

Inputs:

- Full Name
- Email Address
- Password
- Confirm Password

Outputs:

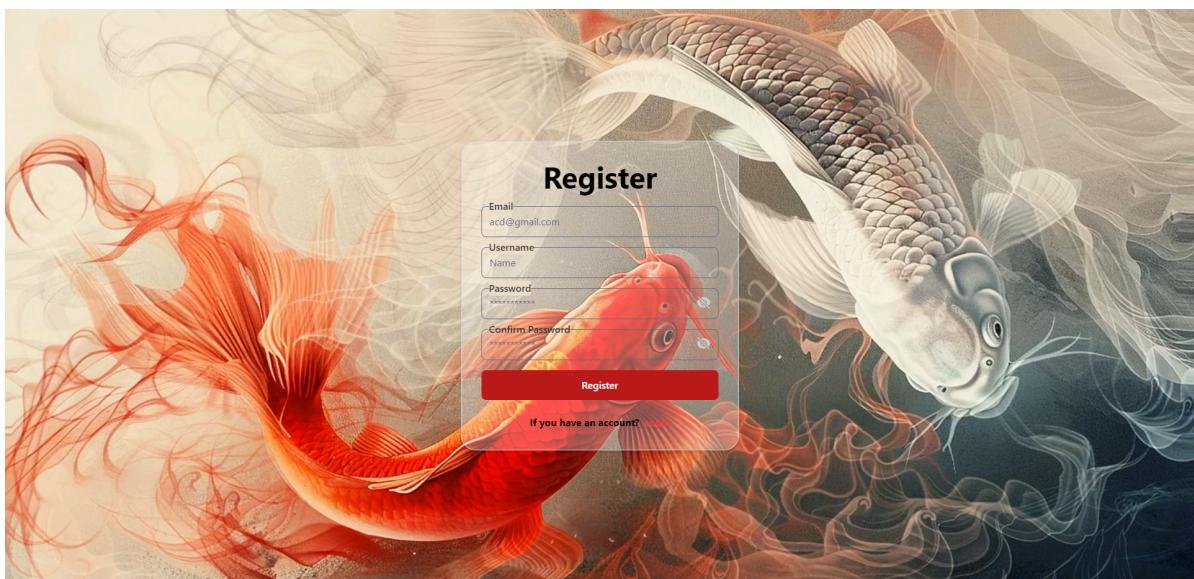
- Success message and confirmation email upon successful registration
- Error message in case of invalid data or failed registration

Process:

1. The user accesses the Registration screen.
2. The user inputs their details, including Full Name, Email, Password, and Confirm Password.
3. The system validates the entered data:
 - a. The Email must be in the correct format (e.g., user@example.com).
 - b. The Password must meet security requirements (minimum length: 8 characters and match the confirm password).
 - c. The Email must not already be registered in the system.
4. If the input is valid, the system:

- a. Stores the new user's information in the database.
 - b. Sends a confirmation email to the user's provided email address.
 - c. Displays a success message and redirects the user to the login page.
5. If the input is invalid, the system:
- a. The system displays an error message indicating the specific issues (e.g., invalid email format, passwords don't match, email already registered).
6. After successful registration, the user can log in with their new credentials.

Screen Layout:



Function Details:

Data:

- User registration data (name, email, password, role)

Validation:

- The Email must follow a valid email format.
- The Password and Confirm Password must match and meet the system's security requirements (minimum 8 characters).
- The Email must not be already in use..

Security:

- Passwords are encrypted before being stored in the database.
- The system uses CAPTCHA or reCAPTCHA (if applicable) to prevent bots from registering.

Business Logic:

● **Normal Cases:**

- The user completes the registration form with valid data and receives a confirmation email.
- The user is redirected to the login page after successful registration.

● **Abnormal Cases:**

- If the email is already registered, the system displays an error message.
- If the password validation fails, the system prompts the user to correct the input.

5. Dashboard Member

5.1 FR-04: Dashboard Member

Function Trigger:

- The function is triggered when a Member logs into the Koi Care System and is successfully authenticated. The member is redirected to their Dashboard.

Function Description:

- **Actors/Roles:** Member
- **Purpose:** To provide the Member with an overview of key data and insights related to their koi care activities, including pond details, koi fish, orders, and other relevant information.
- **Interface:** Member's Dashboard

Data Processing:

● **Inputs:**

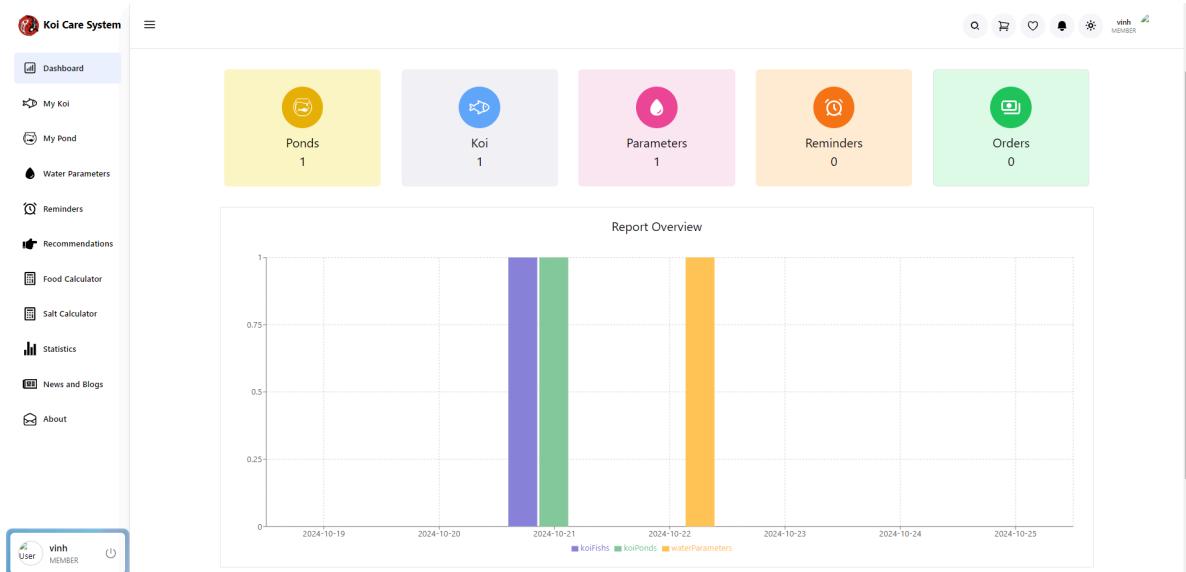
- User Authentication Token

- Member's data (pond information, koi fish, orders, etc.)
- **Outputs:** Display of the Member's key details, including :
 - **Pond Overview:** Total number of ponds, pond status
 - **Koi Fish Overview:** Number of koi
 - **Order and Payment Summary:** Recent orders, total orders and payments

Process:

1. Once the member logs in, the system fetches relevant data for the member's ponds, koi fish, orders and payments from the database.
2. The dashboard is rendered with key metrics and summaries

Screen Layout:



Function Details:

- **Data:** Member's ponds, koi fish, and orders are fetched from the database.
- **Validation:**
 - The member must be authenticated, and only authorized members can access the dashboard.

Business Logic:

- **Normal Cases:** A member successfully logs in and is presented with a personalized dashboard showing an overview of their ponds, koi fish, and orders.
- **Abnormal Cases:** If the system is unable to retrieve the necessary data (e.g., due to a network error), a fallback message is displayed, and the dashboard shows partial or no data.

6. My Koi

6.1 FR-05: My Koi

Function Trigger:

The function is triggered when a Member navigates to the "My Koi" section from the dashboard or main menu.

Function Description:

- **Actors/Roles:** Member
- **Purpose:** To display a list of koi that belong to the member.
- **Interface:** My Koi Page

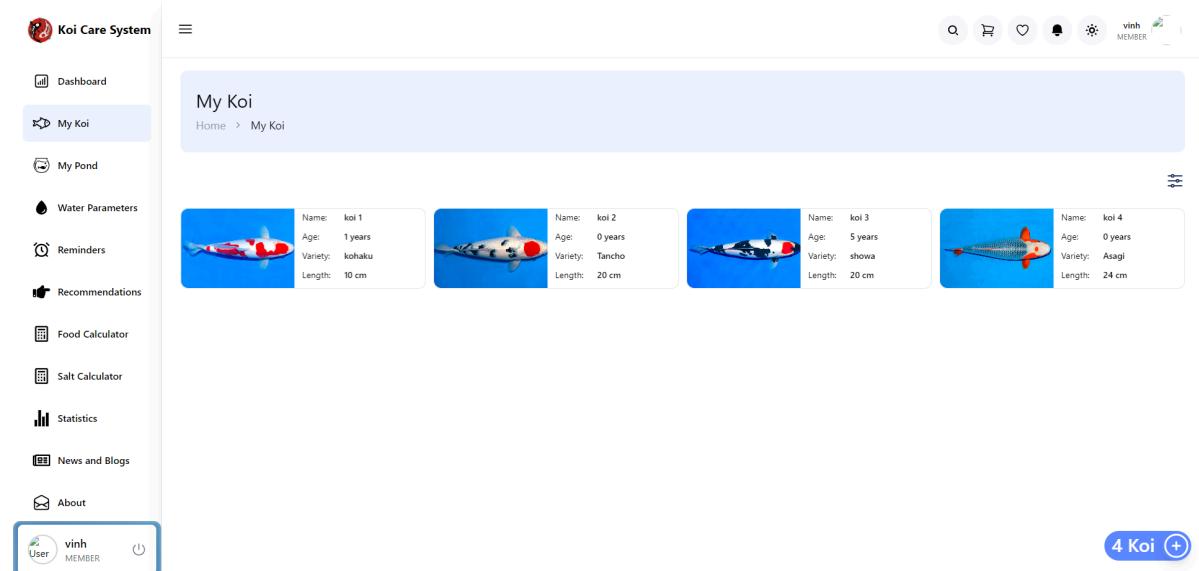
Data Processing:

- **Inputs:** User Authentication Token
- **Outputs:** A paginated or scrollable list of koi, displaying key information:
 - Koi Name
 - Variety
 - Length, Weight
 - Age

Process:

1. The system retrieves the koi list from the database based on the member's account.
2. The koi are displayed in a grid or list format, showing key details like name, length, weight.
3. Members can filter, sort, or search the koi list by name, age, or length.

Screen Layout:



Function Details:

- **Data:** Member's koi information including name, variety, length, weight, age, pond location, and status.
- **Validation:**
 - Not applicable; data is retrieved directly from the database.

Business Logic:

- **Normal Cases:** The list of koi is displayed.
- **Abnormal Cases:** If there are no koi records, a message indicating no koi available is shown.

6.2 FR-06: Create Koi

Function Trigger:

The function is triggered when a Member clicks the "Add Koi" icon button from the "My Koi" page.

Function Description:

- **Actors/Roles:** Member
- **Purpose:** To allow members to create a new koi record in the system.
- **Interface:** Koi Creation Form

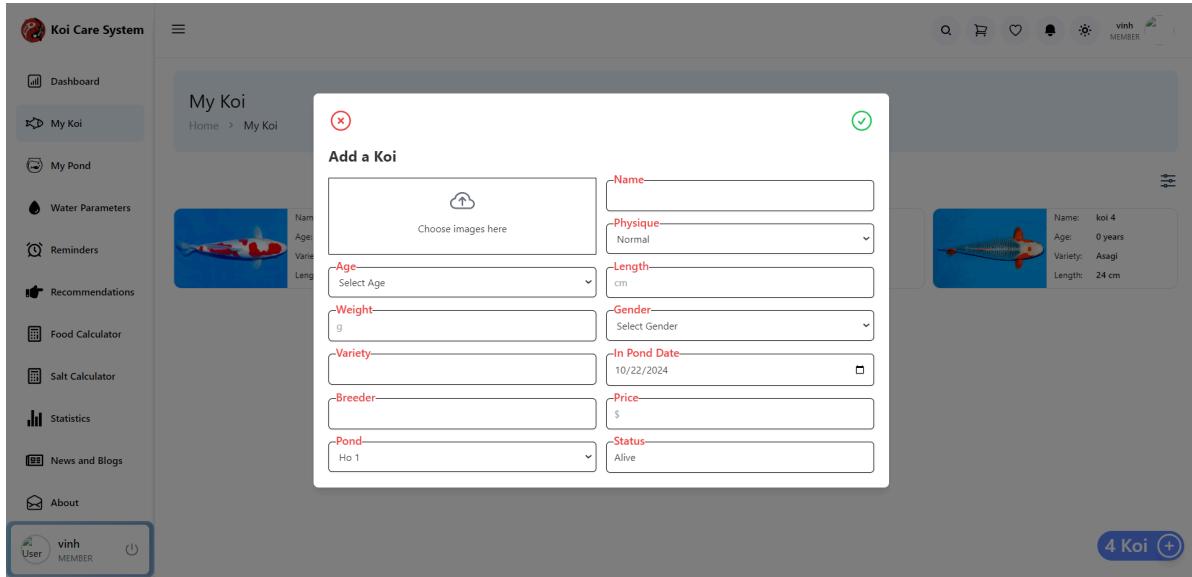
Data Processing:

- **Inputs:** Koi details entered by the member:
 - Name (required)
 - Length (required)
 - Weight (required)
 - Age (required)
 - Pond (required)
 - Variety, Gender, Breeder, Price (optional)
- **Outputs:** A new koi record is created in the database.

Process:

1. The member fills in the necessary koi details.
2. The system validates the input fields (mandatory fields like name, length, weight, age, and pond must be filled).
3. If valid, the koi is created, and the user is redirected to the "My Koi" list.
4. If invalid, an error message is displayed, and the member is asked to correct the input.

Screen Layout:



Function Details:

- **Data:** Koi details (name, length, weight, age, pond, variety, gender, breeder, price)
- **Validation:**
 - Required fields (name, length, weight, age, pond) must not be empty.
 - Length and weight must be positive numbers.
 - Age must be a positive integer.

Business Logic:

- **Normal Cases:** When valid koi details are provided, a new koi record is created in the database.
- **Abnormal Cases:** If required fields are missing or invalid, an error message is shown, and the user is prompted to correct the input.

6.3 FR-07: Koi Details

Function Trigger:

The function is triggered when a Member clicks on a specific koi in the "My Koi" list.

Function Description:

- **Actors/Roles:** Member
- **Purpose:** To view detailed information about a specific koi, including growth history and any remarks.
- **Interface:** Koi Details Page

Data Processing:

- **Inputs:** None
- **Outputs:**

- Koi Details (name, variety, pond, etc.)
- Growth History (length, weight over time)
- Remarks (any notes or special care instructions)

Process:

1. The system retrieves all details related to the selected koi from the database.
2. Growth history data is fetched and displayed .
3. The remarks section shows any notes made by the member or system about the koi's care.

Screen Layout:

The screenshot shows the Koi Care System interface. On the left is a sidebar with icons for Dashboard, My Koi (selected), My Pond, Water Parameters, Reminders, Recommendations, Food Calculator, Salt Calculator, Statistics, News and Blogs, and About. The main area shows a large image of a koi fish named 'koi 1' with the variety 'Kohaku' and pond 'Ho 1'. Below the image are statistics: Age 1 years, Length 20 cm, and Weight 160 g. To the right is a 'Koi Description' box containing text about the koi's history and purchase. Below that is a 'Remarks' section with a note about growth. At the bottom left is a user profile for 'vinh MEMBER'.

Function Details:

- **Data:**
 - Koi basic information at the top.
 - Growth history graph (showing changes in length and weight over time).
 - Remarks section below the growth history.
- **Validation:**
 - Not applicable.

Business Logic:

- **Normal Cases:** When a koi is selected, all related details are displayed correctly.
- **Abnormal Cases:** If the selected koi does not exist, an error message is shown.

6.4 FR-08: Update Koi

Function Trigger:

The function is triggered when a Member clicks the "Edit" button on a specific koi's details page.

Function Description:

- **Actors/Roles:** Member
- **Purpose:** To update information for an existing koi.
- **Interface:** Koi Update Form

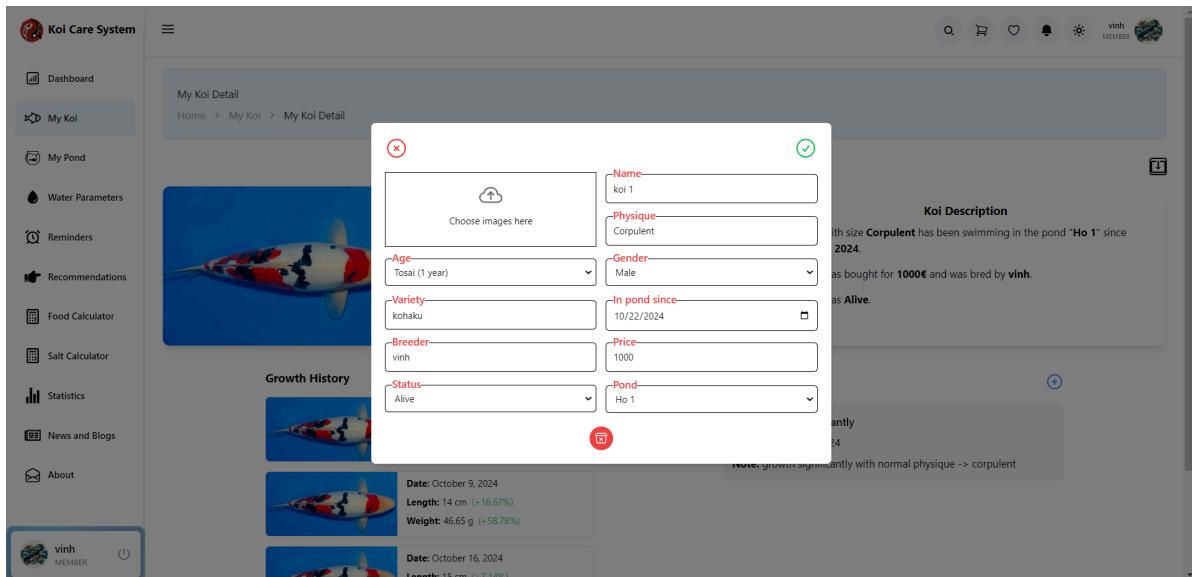
Data Processing:

- **Inputs:** Updated koi details (name, length, weight, pond, etc.)
- **Outputs:** Updated koi record in the database.

Process:

1. The system pre-fills the form with the current data of the selected koi.
2. The member updates the fields as needed (e.g., adjusting the length, weight, or changing the pond).
3. The system validates the input fields.
4. If valid, the koi information is updated in the database, and the member is redirected to the koi details page.
5. If invalid, an error message is displayed.

Screen Layout:



Function Details:

- **Data:** Updated koi information (name, length, weight, pond, etc.)
- **Validation:**
 - Required fields must not be empty.

- Length and weight must be positive numbers.
- Age must be a positive integer.

Business Logic:

- **Normal Cases:** When valid updates are provided, the koi information is successfully updated in the database.
- **Abnormal Cases:** If required fields are missing or invalid, an error message is displayed, prompting the member to correct the input.

6.5 FR-09: Create Growth History

Function Trigger:

The function is triggered when a Member clicks the "Add Growth History" button on the Koi Details page.

Function Description:

- **Actors/Roles:** Member
- **Purpose:** To create a new growth history record for a specific koi.
- **Interface:** Growth History Creation Form

Data Processing:

Inputs:

- Growth history details entered by the member:
 - Date (required)
 - Length (required)
 - Weight (required)
 - Image (optional)

Outputs:

- A new growth history record is created in the database.

Process:

1. The member fills in the necessary growth history details.
2. The system validates the input fields (mandatory fields like date, length, and weight must be filled).
3. If valid, the growth history is created, and the member is redirected to the Koi Details page.
4. If invalid, an error message is displayed, and the member is asked to correct the input.

Screen Layout:

The screenshot shows the Koi Care System dashboard. On the left sidebar, there are various menu items: Dashboard, My Koi, My Pond, Water Parameters, Reminders, Recommendations, Food Calculator, Salt Calculator, Statistics, News and Blogs, and About. A user profile for 'vinh MEMBER' is at the bottom left. The main area displays a koi named 'koi 1' with variety 'kohaku' and pond 'Ho 1'. To the right is a 'Koi Description' section with notes about the koi's size and purchase. Below this is a 'Growth History' section. A modal window is open, titled 'Edit Growth History', allowing the user to input growth details for 'koi 1'. The modal includes fields for 'Date' (mm/dd/yyyy), 'Length (cm)', 'Weight (g)', and 'Physique' (Corpulent). It also has a file upload field 'Choose images here' with a preview area showing two images of the koi. At the bottom of the modal are 'Save' and 'Cancel' buttons.

Function Details:

- **Data:** Growth history details (date, length, weight, image)
- **Validation:**
 - Required fields (date, length, weight) must not be empty.
 - Length and weight must be positive numbers.

Business Logic:

- **Normal Cases:** When valid growth history details are provided, a new record is created in the database.
- **Abnormal Cases:** If required fields are missing or invalid, an error message is shown, prompting the member to correct the input.

6.6 FR-10: Update Growth History

Function Trigger:

The function is triggered when a Member clicks the "Edit" button on an existing growth history record.

Function Description:

- **Actors/Roles:** Member
- **Purpose:** To update existing growth history information for a specific koi.
- **Interface:** Growth History Update Form

Data Processing:

Inputs:

- Growth history details entered by the member:
 - Date (required)
 - Length (required)
 - Weight (required)
 - Image (optional)

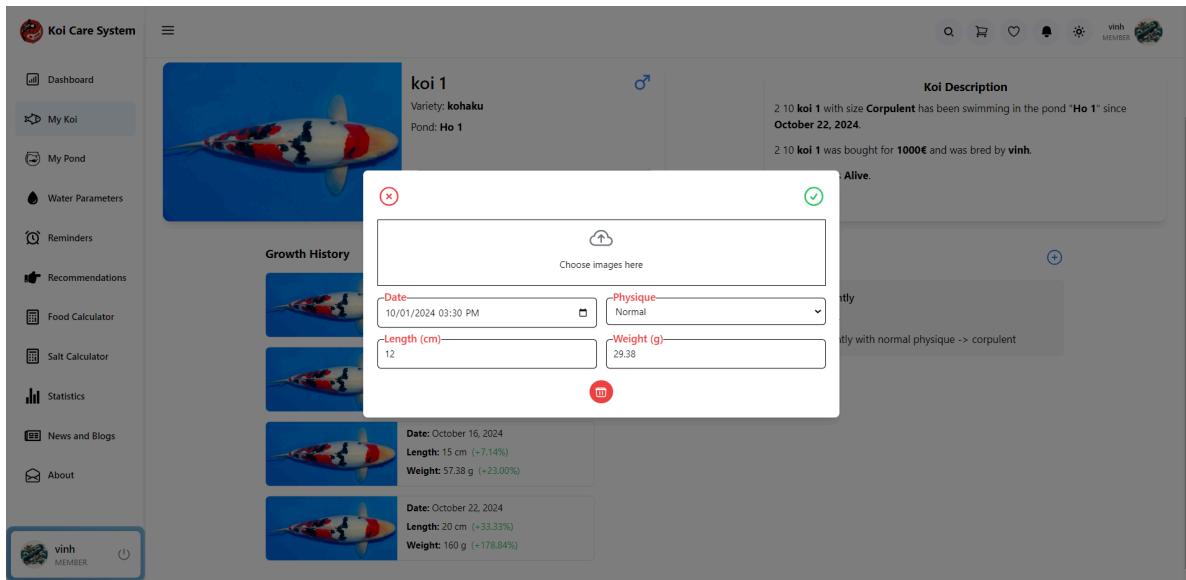
Outputs:

- Updated growth history record in the database.

Process:

5. The system pre-fills the form with the current data of the selected growth history record.
6. The member updates the fields as needed (e.g., adjusting the length, weight, or date).
7. The system validates the input fields.
8. If valid, the growth history information is updated in the database, and the member is redirected to the Koi Details page.
9. If invalid, an error message is displayed.

Screen Layout:



Function Details:

- **Data:** Growth history details (date, length, weight, image)
- **Validation:**
 - Required fields (date, length, weight) must not be empty.
 - Length and weight must be positive numbers.

Business Logic:

- **Normal Cases:** When valid updates are provided, the growth history information is successfully updated in the database.
- **Abnormal Cases:** If required fields are missing or invalid, an error message is displayed, prompting the member to correct the input.

6.7 FR-11: Create Remark

Function Trigger:

The function is triggered when a Member clicks the "Add Remark" button on the Koi Details page.

Function Description:

- **Actors/Roles:** Member
- **Purpose:** To create a new remark for a specific koi.
- **Interface:** Remark Creation Form

Data Processing:

Inputs:

- Remark details entered by the member:
 - Title (required)
 - Note (required)
 - Date (required)

Outputs:

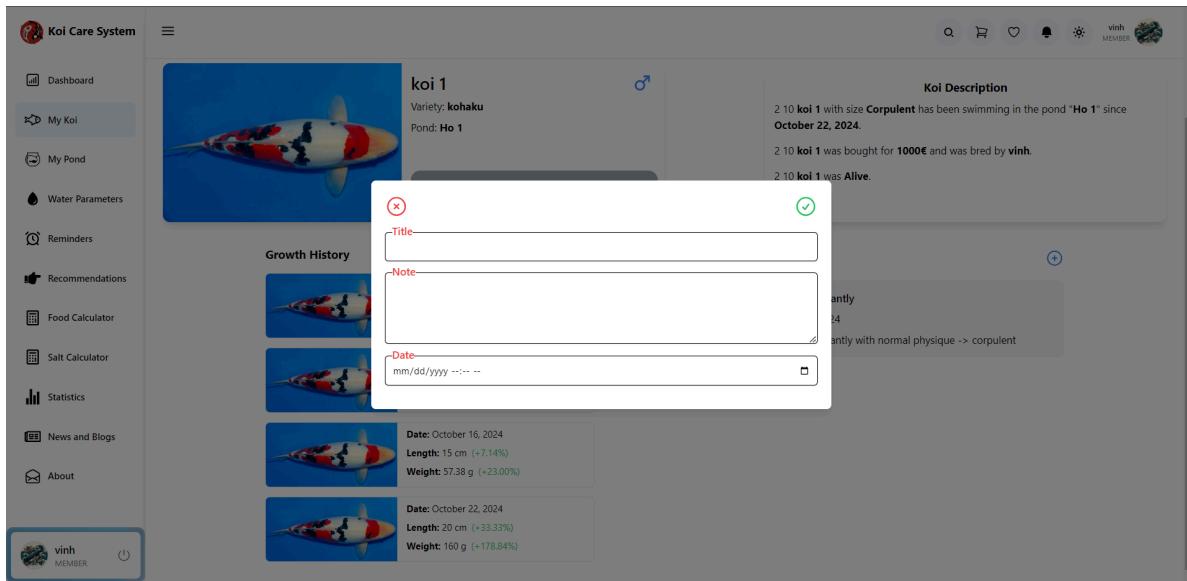
- A new remark record is created in the database.

Process:

10. The member fills in the necessary remark details.
11. The system validates the input fields (both date and title must be filled).
12. If valid, the remark is created, and the member is redirected to the Koi Details page.
13. If invalid, an error message is displayed, and the member is asked to correct the input.

Screen Layout:

Update remark



Function Details:

- **Data:** Remark details (title, date, note)
- **Validation:**
 - Required fields (title, date, note) must not be empty.

Business Logic:

- **Normal Cases:** When valid remark details are provided, a new record is created in the database.
- **Abnormal Cases:** If required fields are missing or invalid, an error message is shown, prompting the member to correct the input.

6.8 FR-12: Update Remark

Function Trigger:

The function is triggered when a Member clicks the "Edit" button on an existing remark.

Function Description:

- **Actors/Roles:** Member
- **Purpose:** To update an existing remark for a specific koi.
- **Interface:** Remark Update Form

Data Processing:

Inputs:

- Remark details entered by the member:
 - Title (required)

- Note (required)
- Date (required)

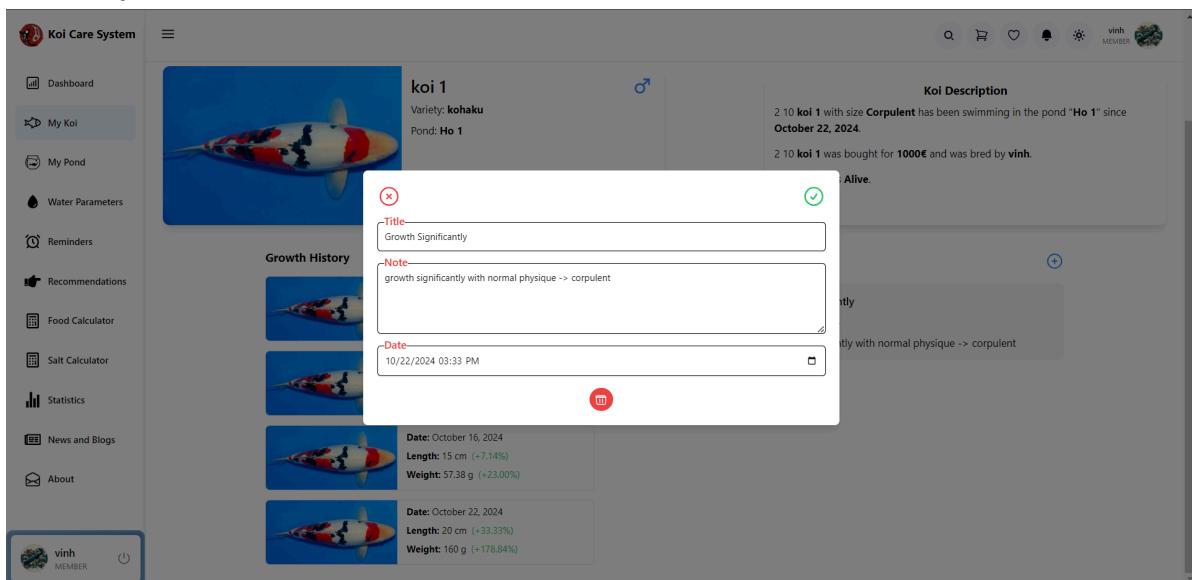
Outputs:

- Updated remark record in the database.

Process:

14. The system pre-fills the form with the current data of the selected remark.
15. The member updates the fields as needed (e.g., adjusting the date or content).
16. The system validates the input fields.
17. If valid, the remark is updated , and the member is redirected to the Koi Details page.
18. If invalid, an error message is displayed.

Screen Layout:



Function Details:

- **Data:** Remark details (title, date, note)
- **Validation:**
 - Required fields (title, date, note) must not be empty.

Business Logic:

- **Normal Cases:** When valid updates are provided, the remark information is successfully updated in the database.
- **Abnormal Cases:** If required fields are missing or invalid, an error message is displayed, prompting the member to correct the input.

7. My Pond

7.1 FR-13: My Pond

Function Trigger:

The function is triggered when a Member navigates to the "My Pond" section from the dashboard or main menu.

Function Description:

- **Actors/Roles:** Member
- **Purpose:** To display a list of ponds that belong to the member.
- **Interface:** My Pond Page

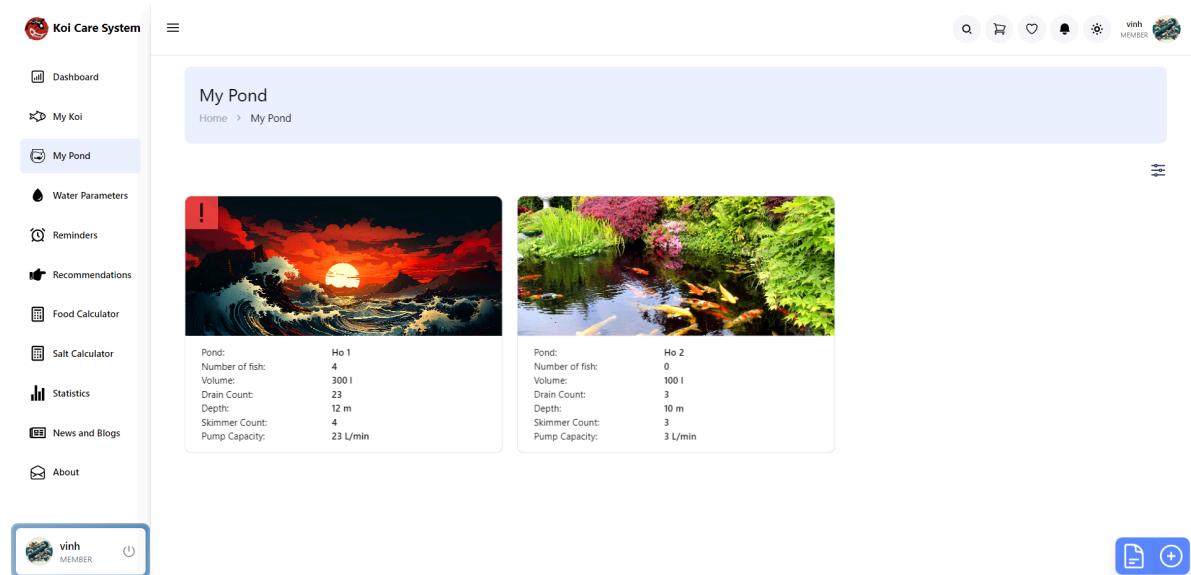
Data Processing:

- **Inputs:** User Authentication Token
- **Outputs:** A paginated or scrollable list of ponds, displaying key information

Process:

1. The system retrieves the pond list from the database based on the member's account.
2. The ponds are displayed in a grid or list format, showing key details like name, volume, depth, drain count,...
3. Members can filter, sort, or search the pond list by name or volume.

Screen Layout:



Function Details:

- **Data:** Pond information (name, volume, number of fishes, depth, ...)
- **Validation:**

- Not applicable.

Business Logic:

- **Normal Cases:** The list of ponds is successfully displayed.
- **Abnormal Cases:** If there are no ponds, a message indicating no ponds available is shown.

7.2 FR-14: Create Pond

Function Trigger:

The function is triggered when a Member clicks the "Add Pond" button from the "My Pond" page.

Function Description:

- **Actors/Roles:** Member
- **Purpose:** To allow members to create a new pond record in the system.
- **Interface:** Pond Creation Form

Data Processing:

- **Inputs:**

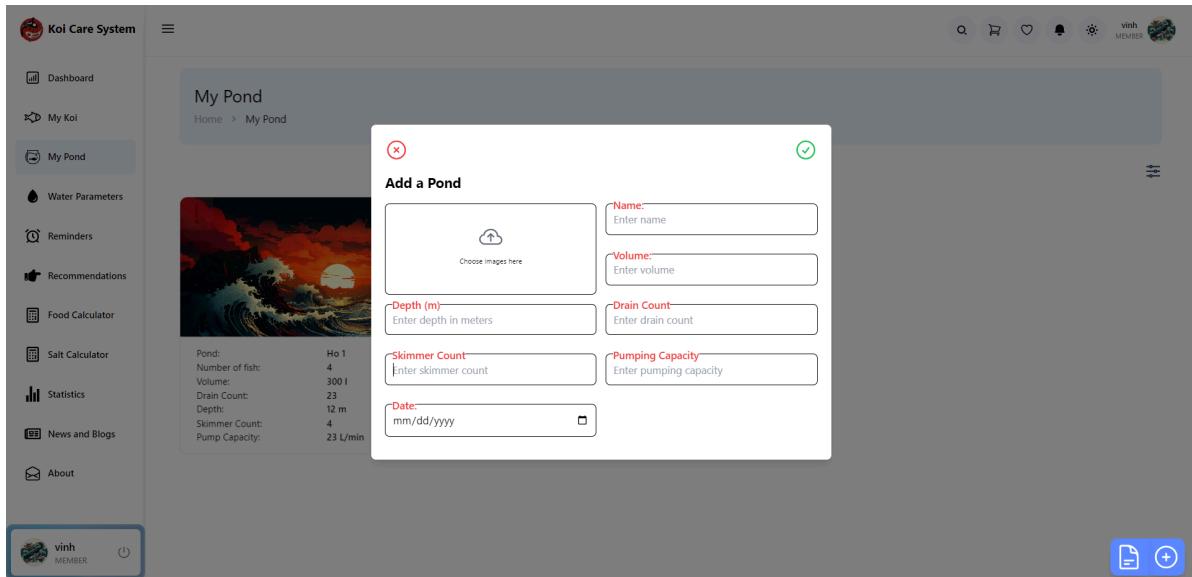
- Image (optional)
- Name (required)
- Volume (required)
- Depth (required)
- Drain Count (required)
- SkimmerCount (required)
- Pumping Capacity (required)
- Date (required)

- **Outputs:** A new pond record is created in the database.

Process:

1. The member fills in the necessary pond details.
2. The system validates the input fields .
3. If valid, the pond is created, and the member is redirected to the "My Pond" list.
4. If invalid, an error message is displayed, and the member is asked to correct the input.

Screen Layout:



Function Details:

- **Data:** Pond details
- **Validation:**
 - Required fields must not be empty.
 - Volume, depth, drain count, skimmer count must be a positive number.

Business Logic:

- **Normal Cases:** When valid pond details are provided, the pond is successfully created in the database.
- **Abnormal Cases:** If required fields are missing or invalid, an error message is displayed, prompting the member to correct the input.

7.3 FR-15: Update Pond

Function Trigger:

The function is triggered when a Member clicks the "Edit" button on a specific pond's details page.

Function Description:

- **Actors/Roles:** Member
- **Purpose:** To update information for an existing pond.
- **Interface:** Pond Update Form

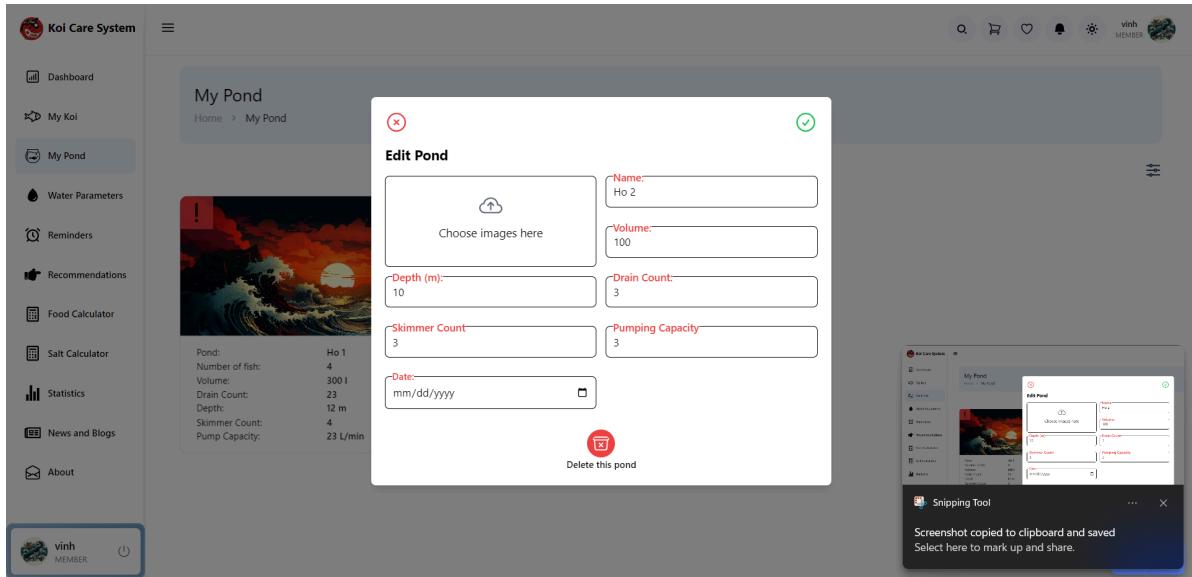
Data Processing:

- **Inputs:** Updated pond details
- **Outputs:** Updated pond record in the database.

Process:

1. The system pre-fills the form with the current data of the selected pond.
2. The member updates the fields as needed.
3. The system validates the input fields.
4. If valid, the pond information is updated in the database, and the member is redirected to the pond details page.
5. If invalid, an error message is displayed.

Screen Layout:



Function Details:

- **Data:** Updated pond information
 - Required fields must not be empty.
 - Volume must be a positive number.

Business Logic:

- **Normal Cases:** When valid updates are provided, the pond information is successfully updated in the database.
- **Abnormal Cases:** If required fields are missing or invalid, an error message is displayed, prompting the member to correct the input.

8. Pond Log

8.1 FR-16: Pond Log

Function Trigger:

The function is triggered when a Member navigates to the "Pond Log" section from the dashboard or main menu.

Function Description:

- **Actors/Roles:** Member

- **Purpose:** To display a list of logs associated with the member's ponds.
- **Interface:** Pond Log Page

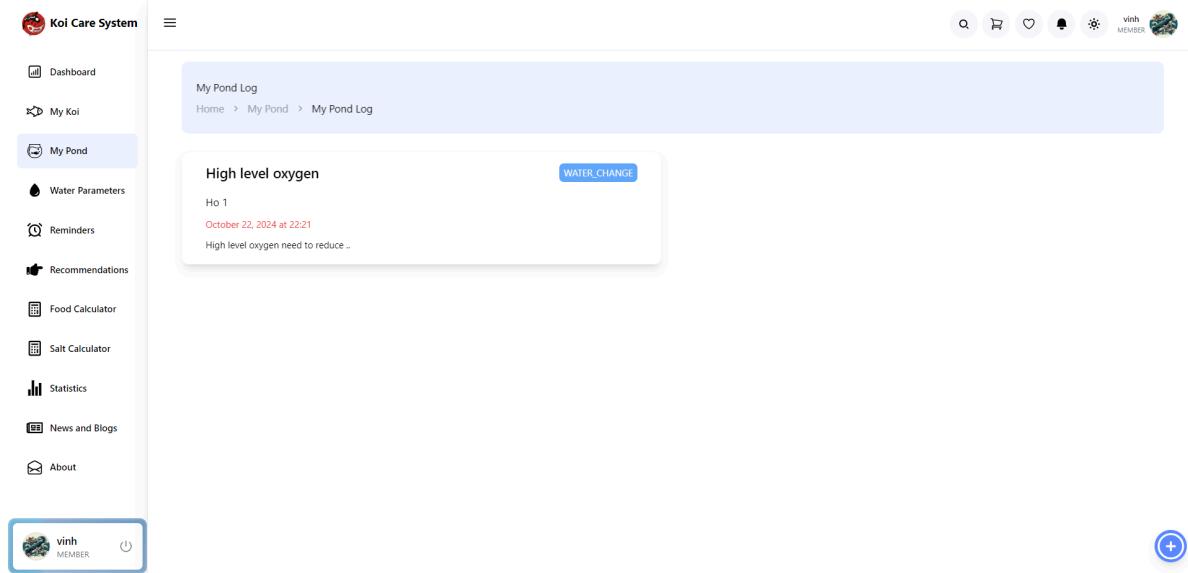
Data Processing:

- **Inputs:** User Authentication Token
- **Outputs:** A paginated or scrollable list of pond logs, displaying key information:
 - Title
 - Date & Time
 - Category
 - Pond
 - Note

Process:

1. The system retrieves the pond log list from the database based on the member's account.
2. The logs are displayed in a grid or list format, showing key details like date, pond name, category and note.
3. Members can filter, sort, or search the pond log list by date, pond name.

Screen Layout:



Function Details:

- **Data:** Pond log information
- **Validation:**
 - Not applicable.

Business Logic:

- **Normal Cases:** The list of pond logs is successfully displayed.
- **Abnormal Cases:** If there are no pond logs, a message indicating no logs available is shown.

8.2 FR-17: Create Pond Log

Function Trigger:

- The function is triggered when a Member clicks the "Add Log" button from the "Pond Log" page.

Function Description:

- **Actors/Roles:** Member
- **Purpose:** To allow members to create a new log entry for a pond.
- **Interface:** Pond Log Creation Form

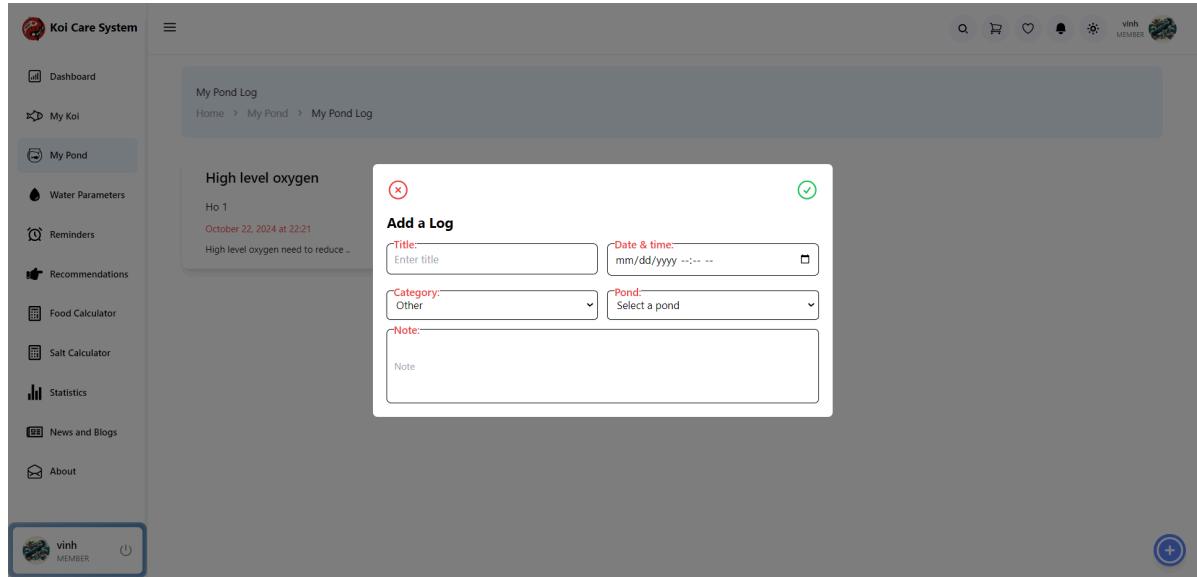
Data Processing:

- **Inputs:**
 - Pond log details entered by the member:
 - Title (required)
 - Date & Time (required)
 - Category (required)
 - Pond (required)
 - Note (optional)
- **Outputs:**
 - A new pond log record is created in the database.

Process:

- The member fills in the necessary pond log details.
- The system validates the input fields (mandatory fields like title, date & time, category, and pond must be filled).
- If valid, the pond log is created, and the member is redirected to the "Pond Log" list.
- If invalid, an error message is displayed, and the member is asked to correct the input.

Screen Layout:



Function Details:

- **Data:** Pond log details (title, date & time, category, pond, note)
- **Validation:**
 - Required fields must not be empty.
- **Business Logic:**
 - **Normal Cases:** When valid pond log details are provided, the log is successfully created in the database.
 - **Abnormal Cases:** If required fields are missing or invalid, an error message is displayed, prompting the member to correct the input.

8.3 FR-18: Update Pond Log

Function Trigger:

- The function is triggered when a Member clicks the "Edit" button on a specific pond log entry's details page.

Function Description:

- **Actors/Roles:** Member
- **Purpose:** To update information for an existing pond log entry.
- **Interface:** Pond Log Update Form

Data Processing:

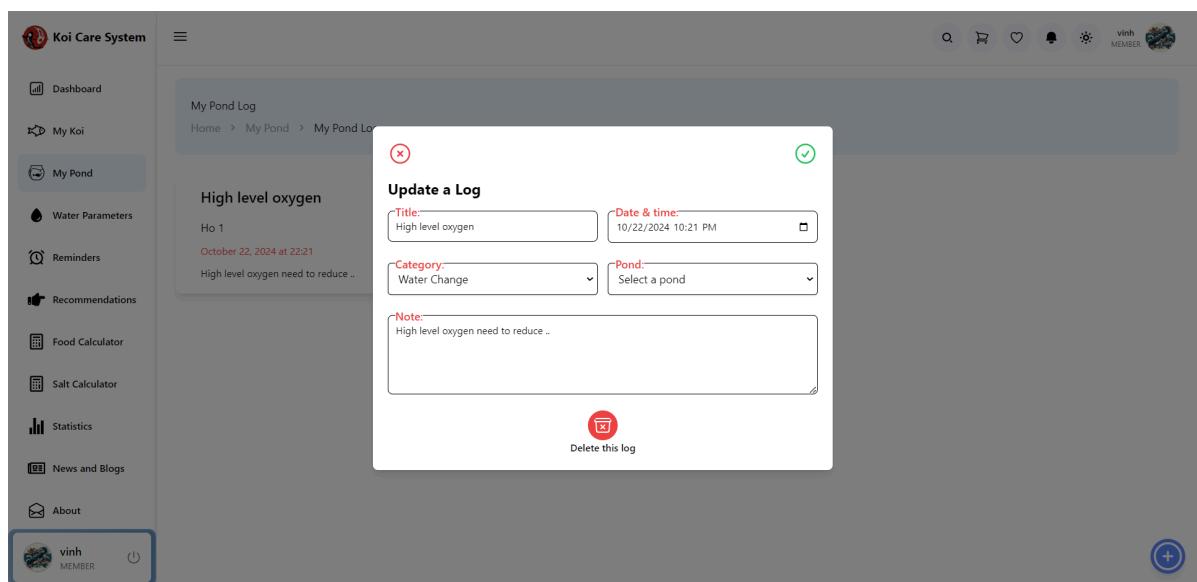
- **Inputs:**
 - Updated pond log details:
 - Title (required)
 - Date & Time (required)
 - Category (required)
 - Pond (required)
 - Note (optional)

- **Outputs:**
 - Updated pond log record in the database.

Process:

- The system pre-fills the form with the current data of the selected pond log.
- The member updates the fields as needed (e.g., changing the title, date & time, category, or note).
- The system validates the input fields.
- If valid, the pond log information is updated in the database, and the member is redirected to the pond log details page.
- If invalid, an error message is displayed.

Screen Layout:



Function Details:

- **Data:** Updated pond log information (title, date & time, category, pond, note)
- **Validation:**
 - Required fields must not be empty.
- **Business Logic:**
 - **Normal Cases:** When valid updates are provided, the pond log information is successfully updated in the database.
 - **Abnormal Cases:** If required fields are missing or invalid, an error message is displayed, prompting the member to correct the input.
 -

9. Water Parameters

9.1 FR-19: Water Parameters

Function Trigger:

- The function is triggered when a Member navigates to the "Water Parameters" section from the dashboard or main menu.

Function Description:

- **Actors/Roles:** Member
- **Purpose:** To display a list of water parameters associated with the member's ponds.
- **Interface:** Water Parameters Page

Data Processing:

- **Inputs:** User Authentication Token
- **Outputs:** A paginated or scrollable list of water parameters, displaying key information:
 - Pond Name
 - Date
 - Nitrite(NO2)
 - Nitrate(NO3)
 - Phosphate(PO4)
 - Ammonium(NH4)
 - Hardness(GH)
 - Oxygen(O2)
 - Temperature
 - PH Value
 - KH
 - CO2
 - Salt
 - Total chlorines
 - Outdoor temp
 - Amount fed

Process:

- The system retrieves the water parameter list from the database based on the member's account.
- The parameters are displayed in a grid or list format.
- Members can filter, sort, or search the water parameters list by name or date.

Screen Layout:

The screenshot shows the 'Water Parameters' section of the Koi Care System. On the left is a sidebar with various icons and links: Dashboard, My Koi, My Pond, Water Parameters (which is selected and highlighted in blue), Reminders, Recommendations, Food Calculator, Salt Calculator, Statistics, News and Blogs, and About. At the top right are icons for search, cart, heart, user profile, and settings. The user profile 'vinh MEMBER' is also visible.

Water Parameters

Home > Water Parameters

Date	Parameter	Value	Date	Parameter	Value
2024-10-22 13:31:00	Nitrite(NO ₂)	1 mg/l	2024-10-22 16:59:00	Nitrite(NO ₂)	0.1 mg/l
	Nitrate(NO ₃)	1 mg/l		Nitrate(NO ₃)	20 mg/l
	Phosphate(PO ₄)	11 mg/l		Ammonium(NH ₄)	0.1 mg/l
	Hardness(GH)	1 °dH		Oxygen (O ₂)	1 mg/l
	Temperature	1 °C		pHValue	1
	KH	1 mg/l		CO ₂	11 mg/l
	Salt	1%		Total chlorines	1 mg/l
	Outdoor temp.	24 °C		Amount fed	200 g
	note			note	

Function Details:

- **Data:** Water parameter information
- **Validation:**
 - Not applicable.
- **Business Logic:**
 - **Normal Cases:** The list of water parameters is successfully displayed.
 - **Abnormal Cases:** If there are no water parameters, a message indicating no parameters available is shown.

9.2 FR-20: Create Water Parameter

Function Trigger:

The function is triggered when a Member clicks the "Add Water Parameter" button from the "Water Parameters" page.

Function Description:

- **Actors/Roles:** Member
- **Purpose:** To allow members to create a new water parameter record in the system.
- **Interface:** Water Parameter Creation Form

Data Processing:

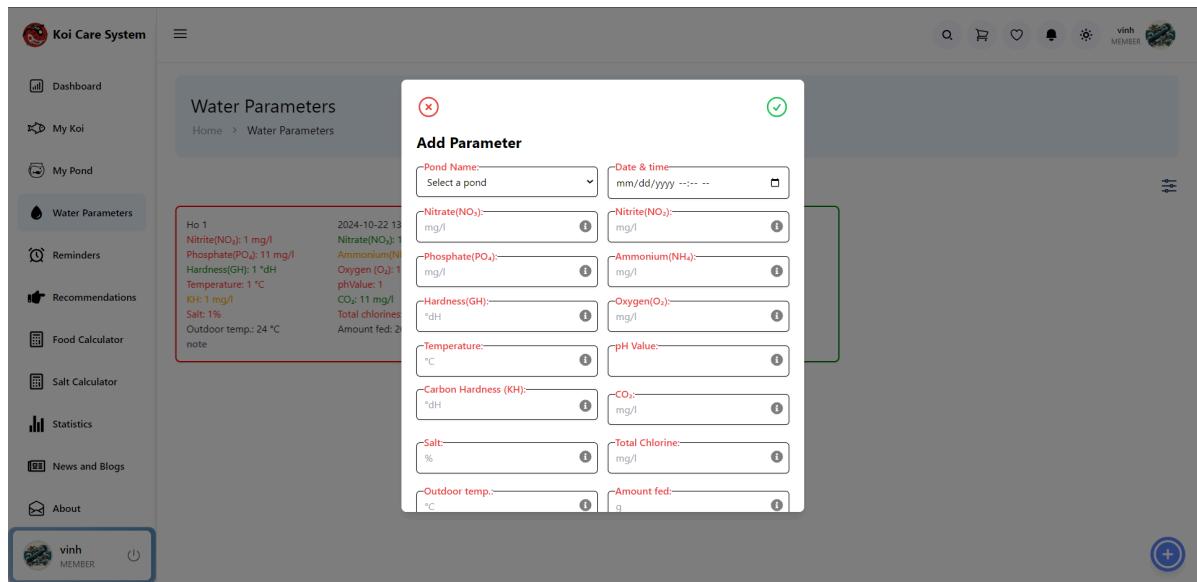
- **Inputs:** Water parameter details entered by the member:
 - Pond Name (required)
 - Date (required)
 - Nitrite (NO₂) (required)
 - Nitrate (NO₃) (required)
 - Phosphate (PO₄) (required)
 - Ammonium (NH₄) (required)
 - Hardness (GH) (required)
 - Oxygen (O₂) (required)

- Temperature (required)
 - pH Value (required)
 - KH (required)
 - CO2 (required)
 - Salt (required)
 - Total Chlorines (required)
 - Outdoor Temp (required)
 - Amount Fed (required)
- **Outputs:**
 - A new water parameter record is created in the database.

Process:

- The member fills in the necessary water parameter details.
- The system validates the input fields (all mandatory fields must be filled).
- If valid, the water parameter is created, and the member is redirected to the "Water Parameters" list.
- If invalid, an error message is displayed, and the member is asked to correct the input.

Screen Layout:



Function Details:

- **Data:** Water parameter details (pond name, date, nitrite, nitrate, phosphate, ammonium, hardness, oxygen, temperature, pH value, KH, CO2, salt, total chlorines, outdoor temp, amount fed)
- **Validation:**
 - Required fields must not be empty.
- **Business Logic:**
 - **Normal Cases:** When valid water parameter details are provided, the parameter is successfully created in the database.

- **Abnormal Cases:** If required fields are missing or invalid, an error message is displayed, prompting the member to correct the input.

9.3 FR-21: Edit Water Parameter

Function Trigger:

The function is triggered when a Member clicks the "Add Water Parameter" button from the "Water Parameters" page.

Function Description:

- **Actors/Roles:** Member
- **Purpose:** To update information for an existing water parameter.
- **Interface:** Water Parameter Edit Form

Data Processing:

- **Inputs:**
 - Updated water parameter details:
 - Pond Name (required)
 - Date (required)
 - Nitrite (NO2) (required)
 - Nitrate (NO3) (required)
 - Phosphate (PO4) (required)
 - Ammonium (NH4) (required)
 - Hardness (GH) (required)
 - Oxygen (O2) (required)
 - Temperature (required)
 - pH Value (required)
 - KH (required)
 - CO2 (required)
 - Salt (required)
 - Total Chlorines (required)
 - Outdoor Temp (required)
 - Amount Fed (required)
- **Outputs:**
 - Updated water parameter record in the database.

Process:

- The system pre-fills the form with the current data of the selected water parameter.
- The member updates the fields as needed.
- The system validates the input fields.
- If valid, the water parameter information is updated in the database, and the member is redirected to the water parameters list or details page.
- If invalid, an error message is displayed.

Screen Layout:

The screenshot shows the 'Edit Parameter' dialog box over a list of water parameters. The parameters listed are:

- Pond Name: Ho 1
- Date & time: 10/22/2024 04:59 PM
- Nitrite(NO_2^-): 20
- Nitrate(NO_3^-): 0.1
- Phosphate(PO_4^{3-}): 0.35
- Ammonium(NH_4^+): 0.1
- Hardness(GH): 20
- Oxygen (O_2): 7
- Temperature: 25
- pH Value: 7
- Carbon Hardness (KH): 41
- CO_2 : 32
- Salt: 0.1
- Total Chlorine: 0 (highlighted in red)
- Outdoor temp.: 20
- Amount fed: 10

Function Details:

- Data:** Updated water parameter information (pond name, date, nitrite, nitrate, phosphate, ammonium, hardness, oxygen, temperature, pH value, KH, CO₂, salt, total chlorines, outdoor temp, amount fed)
- Validation:**
 - Required fields must not be empty.
- Business Logic:**
 - Normal Cases:** When valid updates are provided, the water parameter information is successfully updated in the database.
 - Abnormal Cases:** If required fields are missing or invalid, an error message is displayed, prompting the member to correct the input.

10. Reminders

10.1 FR-22: List Reminders

Function Trigger:

- The function is triggered when a Member navigates to the "Reminders" section from the dashboard or main menu.

Function Description:

- Actors/Roles:** Member
- Purpose:** To display a list of reminders.
- Interface:** Reminders Page

Data Processing:

- Inputs:** User Authentication Token
- Outputs:** A paginated or scrollable list of pond logs, displaying key information:

- Title
- Start
- End
- Repeat
- Description

Process:

- The system retrieves the reminders list from the database based on the member's account.
- The reminders are displayed in a grid or list format, showing key details like title, start, end, repeat, description.

Screen Layout:

The screenshot shows the Koi Care System interface. On the left is a vertical sidebar with various menu items. The 'Reminders' item is currently selected and highlighted. The main content area displays a weekly calendar for the week of February 11-17, 2018. A specific event, 'Meeting' from 10:00 AM to 12:00 PM on Thursday, February 15, is shown as a blue rectangular block on the calendar grid.

Function Details:

- **Data:** List reminders

Business Logic:

- **Normal Cases:** The system successfully retrieves and displays the list of reminders from the database based on the member's account.
- **Abnormal Cases:** If the system fails to retrieve reminders (e.g., due to database issues), an error message is displayed, indicating the issue and prompting the member to try again later.

10.2 FR-23: Add Reminders

Function Trigger: The function is triggered when a member navigates to the "Reminders" section from the dashboard or main menu and selects the "Create Reminder" option.

Function Description:

- **Actors/Roles:** Member
- **Purpose:** To allow members to add new reminders related to their koi pond activities, such as feeding schedules, water parameter checks, or maintenance tasks.
- **Interface:** Create Reminder Form

Data Processing:

- **Inputs:**
 - **User Authentication Token:** Used to verify the member's identity and grant access to the reminder creation process.
 - **Reminder Details:**
 - **Title** (required): The name of the reminder.
 - **Start Date** (required): The date when the reminder starts.
 - **End Date** (optional): The date when the reminder ends (must be after the start date, if provided).
 - **Repeat** (optional): The frequency of the reminder (e.g., daily, weekly, monthly).
 - **Description** (optional): Additional notes or details about the reminder.

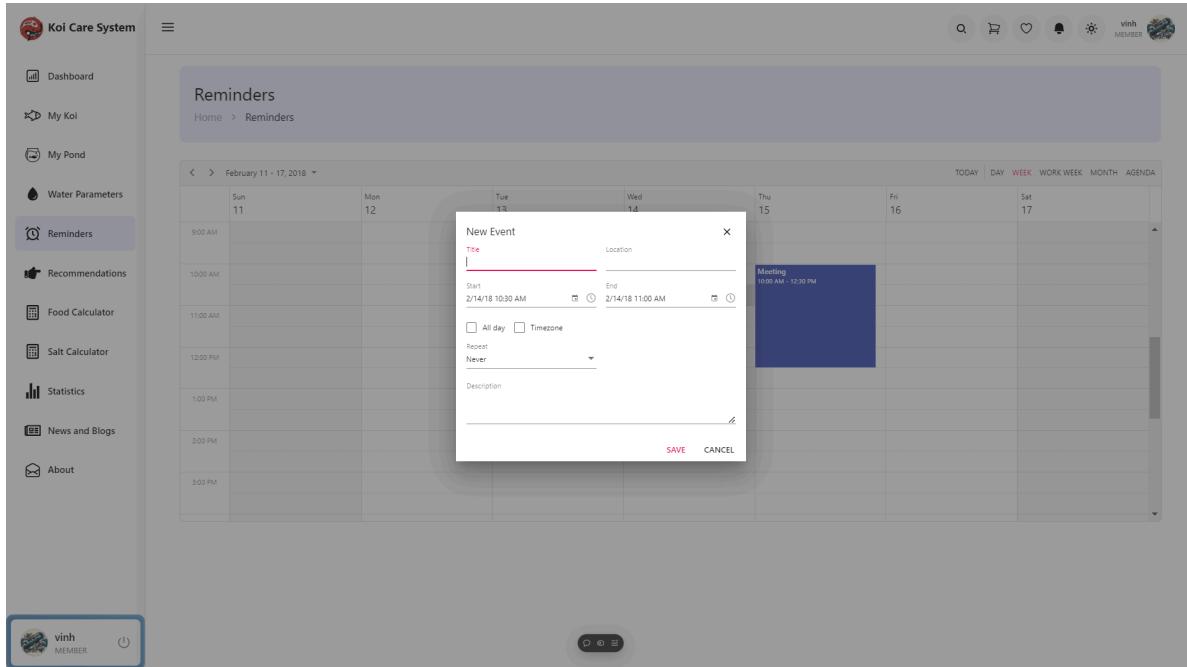
Outputs:

- Confirmation message upon successful creation of the reminder.
- The newly created reminder appears in the reminders list.
- Error messages in case of invalid input or missing required fields.

Process:

- The system retrieves the **User Authentication Token** to verify the member's access rights.
- The member inputs the reminder details (title, start date, etc.).
- The system validates the input fields:
 - Required fields (**Title**, **Start Date**) must not be empty.
 - If an **End Date** is provided, it must be later than the **Start Date**.
- **If valid:**
 - The new reminder is saved in the database.
 - A confirmation message is displayed, and the member is redirected to the reminders list.
- **If invalid:**
 - The system displays an error message, prompting the member to correct the input.

Screen Layout:



Business Logic:

- **Normal Cases:**
 - When all required fields are valid, the reminder is created successfully, saved in the database, and the member is notified with a success message.
- **Abnormal Cases:**
 - If required fields are missing or invalid (e.g., an empty title or invalid date), the system prevents the reminder creation and displays an error message asking the member to fix the issues.

This logic ensures smooth creation of reminders while managing input errors and member authentication.

11. Recommendations

11.1 FR-23: Recommendations

Function Trigger:

- The function is triggered when a Member navigates to the "Recommendations" section from left side bar menu.

Function Description:

- **Actors/Roles:** Member
- **Purpose:** To display a list of recommended products for the member.
- **Interface:** Recommendations Page

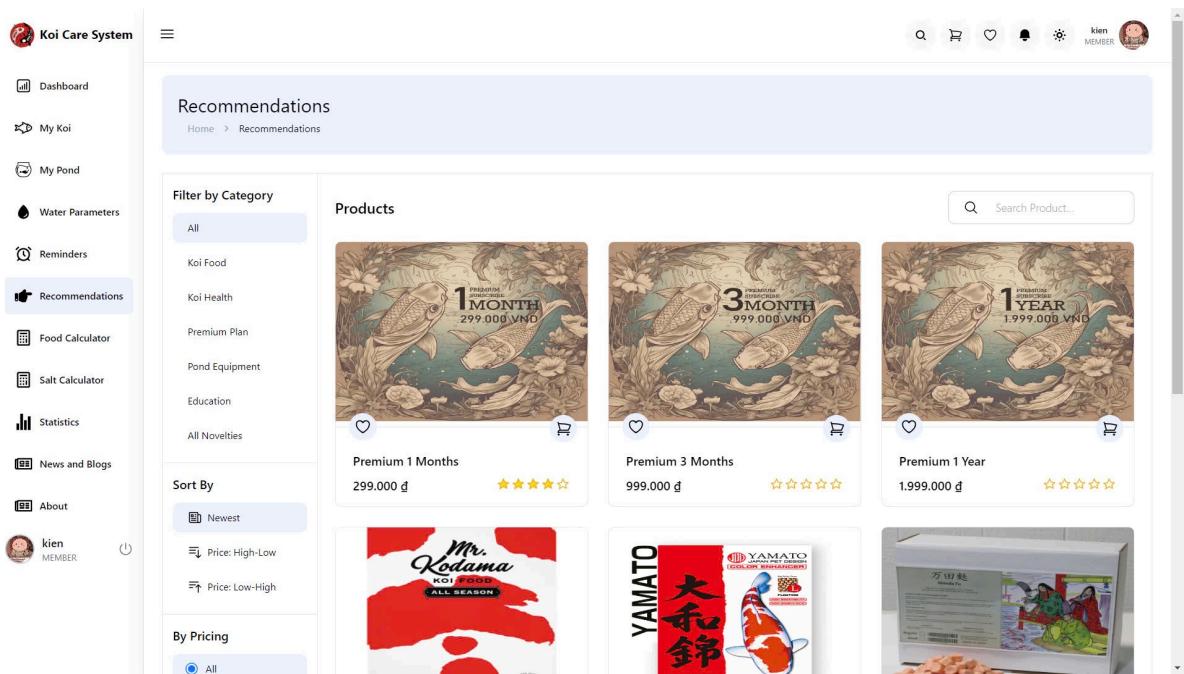
Data Processing:

- **Inputs:** User Authentication Token
- **Outputs:** A paginated or scrollable list of recommended products, displaying key information:
 - Product Name
 - Price
 - Rating

Process:

- The system retrieves the recommended products from the database based on the member's preferences and activity.
- The recommended products are displayed in a grid or list format, showing key details like name, price, and rating.
- Members can filter, sort, or search the recommended products list.

Screen Layout:



Function Details:

- **Data:** Recommended product information (name, price, rating)
- **Validation:** Not applicable.
- **Business Logic:**
 - **Normal Cases:** The list of recommended products is successfully displayed.
 - **Abnormal Cases:** If there are no recommended products, a message indicating no recommendations available is shown.

11.2 FR-25: Recommendation Details

Function Trigger:

- The function is triggered when a Member clicks on a specific product from the "Recommendations" page or any product list.

Function Description:

- **Actors/Roles:** Member
- **Purpose:** To view detailed information about a specific product.
- **Interface:** Product Details Page

Data Processing:

- **Inputs:**
 - Product ID (selected product)
- **Outputs:**
 - Display of the following product details:
 - Name
 - Price
 - Description
 - Images
 - Ratings and Reviews
 - Availability
 - Related Products

Process:

- The system retrieves all details related to the selected product from the database.
- The product details are displayed on the product details page.
- Members can view related products and leave reviews or ratings.

Screen Layout:

In Stock Koi Food

Mr. Kodama Koi Food – Small Pack 50/50 for Growth

We are now offering Mr. Kodama 50/50 to those who may not have a big pond or many Koi, but still want the benefit of both foods to help your Koi grow healthy and beautiful.

2.000.000 ₫

★★★★★ (1 reviews)

Quantity:

Buy Now **Add to cart**

Post Reviews

★★★★★

Post Review

Description	Reviews
Average Rating 5.0/5 ★★★★★	Comment vinhshop ★★★★★ good food

Function Details:

- **Data:** Product information (name, price, description, images, ratings)
- **Validation:** Not applicable.
- **Business Logic:**
 - **Normal Cases:** The product details are successfully displayed.
 - **Abnormal Cases:** If the product is not found, a message indicating the product is unavailable is shown.

12. News and Blogs

12.1 FR-26: Blog

Function Trigger:

The function is triggered when a Member navigates to the "Blog" section from the dashboard or main menu.

Function Description:

- Actors/Roles: Member
- Purpose: To display a list of blog posts related to koi care and pond management.
- Interface: Blog Page

Data Processing:

- Inputs: User Authentication Token
- Outputs: A paginated or scrollable list of blog posts, displaying key information:
 - Blog Title
 - Date
 - Tag

Process:

- The system retrieves the list of blog posts from the database based on the member's account.
- The blog posts are displayed in a grid or list format, showing key details like title and date published.
- Members can filter, sort, or search the blog posts list.

Screen Layout:

The screenshot shows the Koi Care System interface. On the left is a sidebar with various icons and links: Dashboard, My Koi, My Pond, Water Parameters, Reminders, Recommendations, Food Calculator, Salt Calculator, Statistics, News and Blogs (which is selected), and About. At the bottom of the sidebar is a user profile for 'vinh MEMBER'. The main content area is titled 'News' and shows a breadcrumb path 'Home > News'. There is a search bar labeled 'Search Blog...'. Below it is a large image of a koi fish. Underneath the image is a card with the title 'Koi Health' and the date '2024-10-22'. The card contains a short description: 'A koi farm is a specialized facility focused on the breeding and nurturing of koi fish. A koi farm is a specialized facility focused on the breeding and nurturing of koi fish, celebrated for their striking colors and serene movements.' There are also back and forward navigation buttons.

This screenshot shows a detailed view of a blog post. The sidebar on the left remains the same. The main content area has a header 'Koi Health'. The post content reads: 'A koi farm is a specialized facility focused on the breeding and nurturing of koi fish. A koi farm is a specialized facility focused on the breeding and nurturing of koi fish, celebrated for their striking colors and serene movements.' Below this is the date '2024-10-22'. The full content of the post is visible: 'A koi farm is a specialized facility focused on the breeding and nurturing of koi fish. A koi farm is a specialized facility focused on the breeding and nurturing of koi fish, celebrated for their striking colors and serene movements. The Art of Koi Breeding. At the heart of every koi farm is the breeding process, a meticulous and delicate operation that requires deep knowledge and attention to detail. Breeding koi is not simply about increasing numbers; it is a craft focused on producing fish with the most striking patterns, vibrant colors, and ideal body shapes. Selecting Parent Koi: The breeding process begins with selecting the right parent fish, or broodstock. These are koi that exhibit exceptional qualities—bright colors, well-defined patterns, and a strong, healthy form. Breeders aim to pass these desirable traits on to the next generation. Creating Optimal Conditions: Koi farms provide highly controlled environments, including the right water temperature, pH balance, and oxygen levels. Koi farmers often simulate natural seasonal changes to encourage spawning, which typically occurs in spring. Nurturing Fry: Once the eggs are fertilized, they hatch into fry—tiny, delicate koi fish. These fry require special care, as they are highly susceptible to changes in their environment. Farmers focus on feeding them high-nutrient food and keeping water conditions optimal to support healthy growth.' At the bottom right is a 'Share this post:' button with icons for Facebook, Twitter, LinkedIn, and Email.

Function Details:

- Data: Blog post information (title, date, description)
- Validation: Not applicable.
- Business Logic:
 - Normal Cases: The list of blog posts is successfully displayed.
 - Abnormal Cases: If there are no blog posts, a message indicating no blog posts available is shown.

12.2 FR-27: Blog Details

Function Trigger:

- The function is triggered when a Member clicks on a specific blog post from the "Blog" page.

Function Description:

- **Actors/Roles:** Member
- **Purpose:** To view detailed information about a specific blog post, including the content.
- **Interface:** Blog Details Page

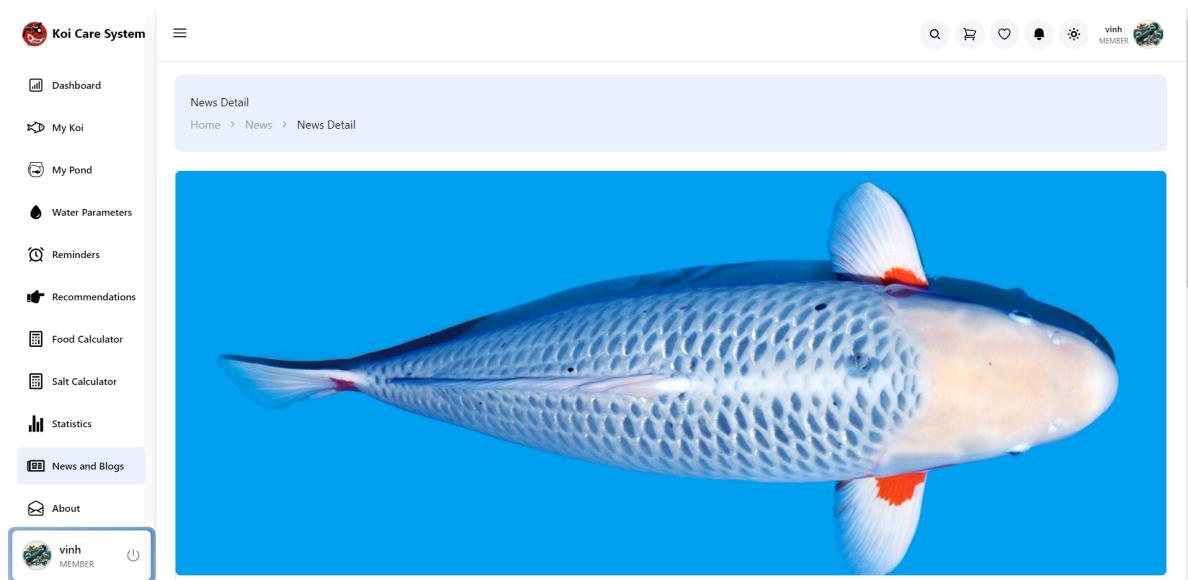
Data Processing:

- **Inputs:**
 - Blog Post ID (selected blog post)
- **Outputs:**
 - Display of the following blog post details:
 - Title
 - Author
 - Date Published
 - Full Content
 - Comments Section

Process:

- The system retrieves all details related to the selected blog post from the database.
- The blog post details are displayed on the blog details page.
- Members can read the full content and view or leave comments.

Screen Layout:



Function Details:

- **Data:** Blog post information (title, author, date, content, comments)
- **Validation:** Not applicable.
- **Business Logic:**
 - **Normal Cases:** The blog post details are successfully displayed.
 - **Abnormal Cases:** If the blog post is not found, a message indicating the post is unavailable is shown.

13. Cart and Checkout

FR-28: Cart

Function Trigger: The function is triggered when a member navigates to the "Cart" section from the dashboard, main menu, or after adding items to the cart.

Function Description:

- **Actors/Roles:** Member
- **Purpose:** To display all items currently in the member's shopping cart, allowing them to view and manage their selected items before proceeding to checkout.
- **Interface:** Shopping Cart Page

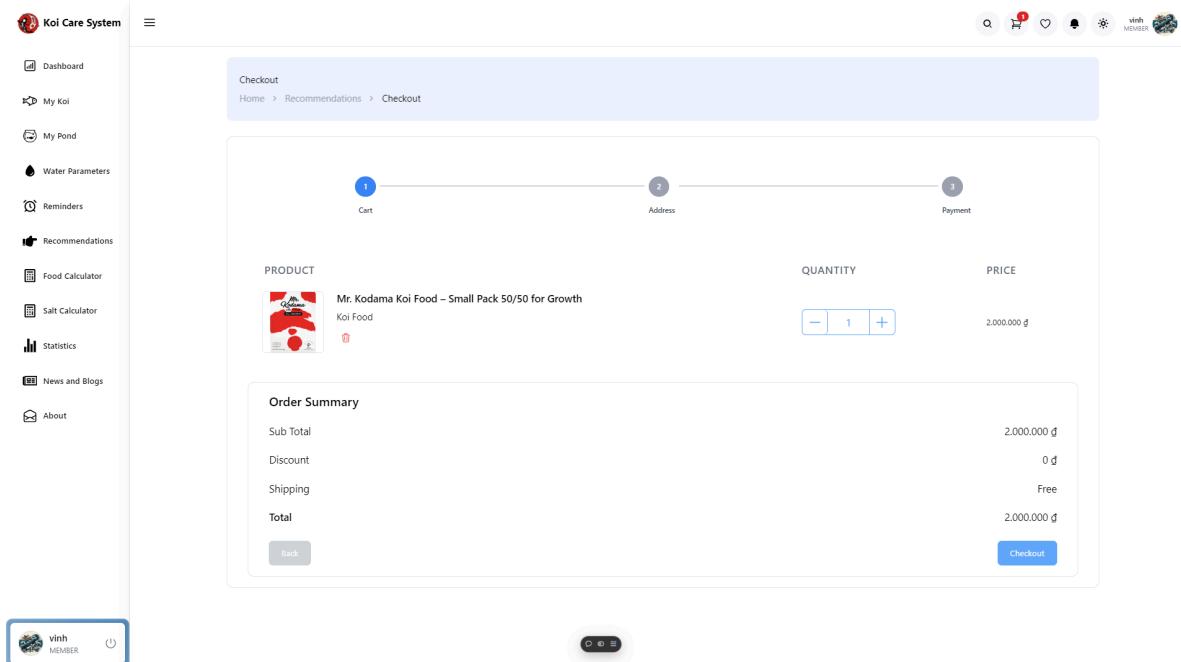
Data Processing:

- **Inputs:**
 - **User Authentication Token:** Verifies the member's identity and retrieves the cart data specific to their account.
 - **Cart Item Details:**
 - **Name:** The name of the product in the cart.
 - **Category:** The category to which the product belongs (e.g., food, equipment).
 - **Price:** The price per unit of the product.
 - **Quantity:** The number of units the member has added to the cart.
 - **Total:** The total cost for that product, calculated as price × quantity.
 - **Discount (if applicable):** Any discount applied to the product.
- **Outputs:**
 - A list of items in the cart, displaying the details mentioned above.
 - A summary of the total cost, including discounts and overall cart value.
 - Error messages in case of invalid data retrieval or issues with the cart.

Process:

- The system retrieves the **User Authentication Token** and fetches the cart data from the database.
- The member's cart items are displayed in a list or grid format, showing key details:
 - **Name, Category, Price, Quantity, Total, and Discount (if applicable).**
- The system calculates the **Total** for each product as well as the overall cart value, taking any **Discounts** into account.
- The member can view the cart items, adjust quantities, or remove items as needed.
- If any issues occur (e.g., items missing or invalid data), an error message is displayed, prompting the member to try again.

Screen Layout:



Function Details:

- **Data:**
 - **Cart Items:** A collection of items added to the cart, including all relevant attributes (name, category, price, quantity, total, and discount).
 - **Cart Summary:** Total price calculations, including any discounts applied.

Validation:

- Not applicable to the cart view since the primary function is to display existing data.

Business Logic:

- **Normal Cases:**
 - The system successfully retrieves the cart data and displays all items, allowing the member to view their selections and manage the cart.
 - Cart totals, discounts, and quantities are updated in real-time based on member actions (e.g., increasing quantity, removing an item).
- **Abnormal Cases:**
 - If the system is unable to retrieve the cart data (e.g., due to network issues or invalid token), an error message is displayed, prompting the member to refresh the page or log in again.
 - If cart items contain invalid data (e.g., incorrect prices or quantities), the system shows an error and prevents proceeding to checkout until the issues are resolved.

FR-29: Checkout

Function Trigger: The function is triggered when a member navigates to the "Checkout" section from the cart page after adding items to their cart.

Function Description:

- **Actors/Roles:** Member
- **Purpose:** To allow members to review their order details, provide necessary shipping information, and complete the checkout process.
- **Interface:** Checkout Page

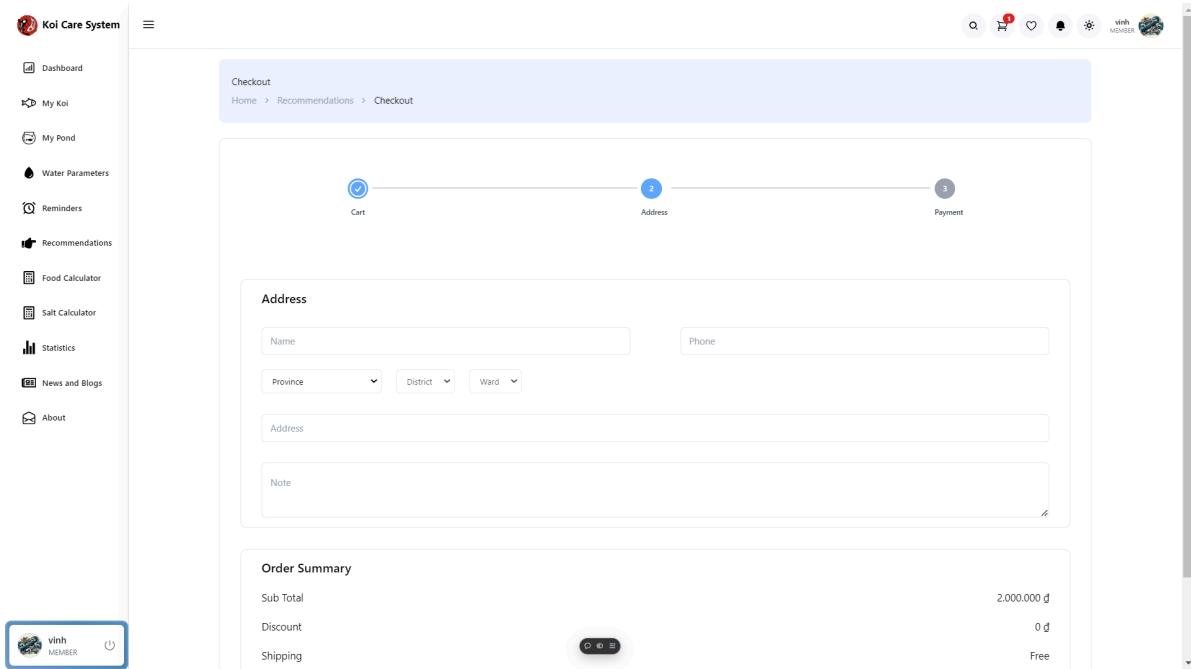
Data Processing:

- **Inputs:**
 - **User Authentication Token:** Verifies the member's identity and retrieves their cart and previous order information.
 - **Order Details:**
 - **Name:** The member's full name (required).
 - **Phone:** Contact phone number (required).
 - **Province:** The province or state for shipping (required).
 - **District:** The district for shipping (required).
 - **Ward:** The ward or locality for shipping (required).
 - **Note:** Any additional notes or instructions for the order (optional).
- **Outputs:**
 - A summary of the order, including items purchased, total cost, shipping details, and any applicable taxes or discounts.
 - Confirmation messages upon successful checkout or error messages in case of invalid input.

Process:

- The system retrieves the **User Authentication Token** and fetches the cart data and any previous order information.
- The member reviews their order details displayed on the checkout page, which includes:
 - Item list with name, price, quantity, and total for each item.
 - Order summary with the overall total cost.
- The member inputs their shipping information, including name, phone number, province, district, ward, and any notes.
- The system validates the input fields:
 - Ensures that required fields (name, phone, province, district, ward) are filled.
 - Checks for valid phone number format.
- Upon successful validation, the system processes the order and saves the details to the database.
- The member receives a confirmation message, and the system displays an order summary, including an order number and estimated delivery time.

Screen Layout:



Function Details:

- **Data:**

- **Order Summary:** Includes all items in the cart, total cost, applicable discounts, and shipping information.
- **Shipping Information:** Name, phone, province, district, ward, and note provided by the member.

Validation:

- Required fields must not be empty.
- Phone number must be in a valid format.

Business Logic:

- **Normal Cases:**

- The system successfully retrieves the order details and shipping information, allowing the member to review and confirm their order.
- The order is processed successfully, and a confirmation message is displayed, along with the order details.

- **Abnormal Cases:**

- If required fields are missing or invalid, an error message is displayed, prompting the member to correct the input.
- If there are issues processing the order (e.g., payment failure or inventory issues), the system provides an appropriate error message and suggests next steps.

14. Food Calculator

14.1 FR-30: Food Calculator

Function Trigger:

- The function is triggered when a Member navigates to the "Food Calculator" section from the dashboard or main menu.

Function Description:

- **Actors/Roles:** Member
- **Purpose:** To calculate the appropriate amount of food required for koi based on specific information such as pond and temperature.
- **Interface:** Food Calculator Page

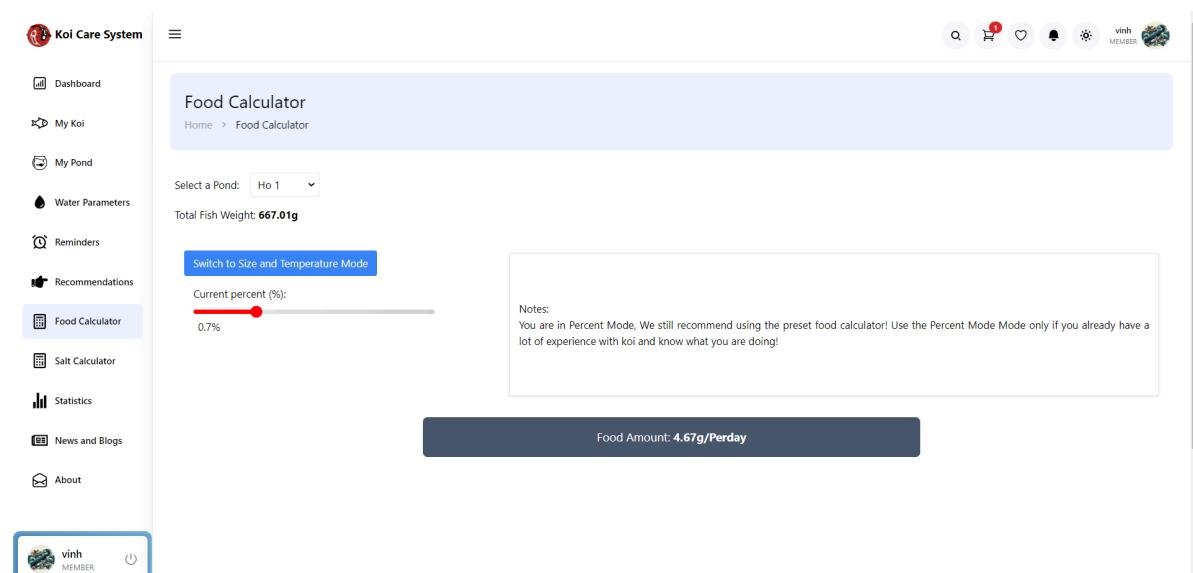
Data Processing:

- **Inputs:**
 - Pond Name
 - Current percent
 - Fish size
 - Temperature
- **Outputs:**
 - Recommended amount of food (in grams or ounces)
 - Feeding schedule (if applicable)

Process:

- The member inputs the koi's weight and feeding frequency into the calculator.
- The system uses predefined feeding guidelines and calculations to determine the appropriate amount of food.
- The recommended food amount is displayed, along with any additional feeding advice.

Screen Layout:



Function Details:

- **Data:** Koi weight and feeding guidelines
- **Validation:**
 - Not applicable.
- **Business Logic:**
 - **Normal Cases:** The recommended amount of food is successfully calculated and displayed.
 - **Abnormal Cases:** an error message is shown prompting the user to correct the amount of foods.

15. Salt Calculator

15.1 FR-31: Salt Calculator

Function Trigger:

The function is triggered when a Member navigates to the "Salt Calculator" section from the dashboard or main menu.

Function Description:

- **Actors/Roles:** Member
- **Purpose:** To calculate the appropriate amount of salt needed for pond treatment based on specific information such as pond volume and desired salt concentration.
- **Interface:** Salt Calculator Page

Data Processing:

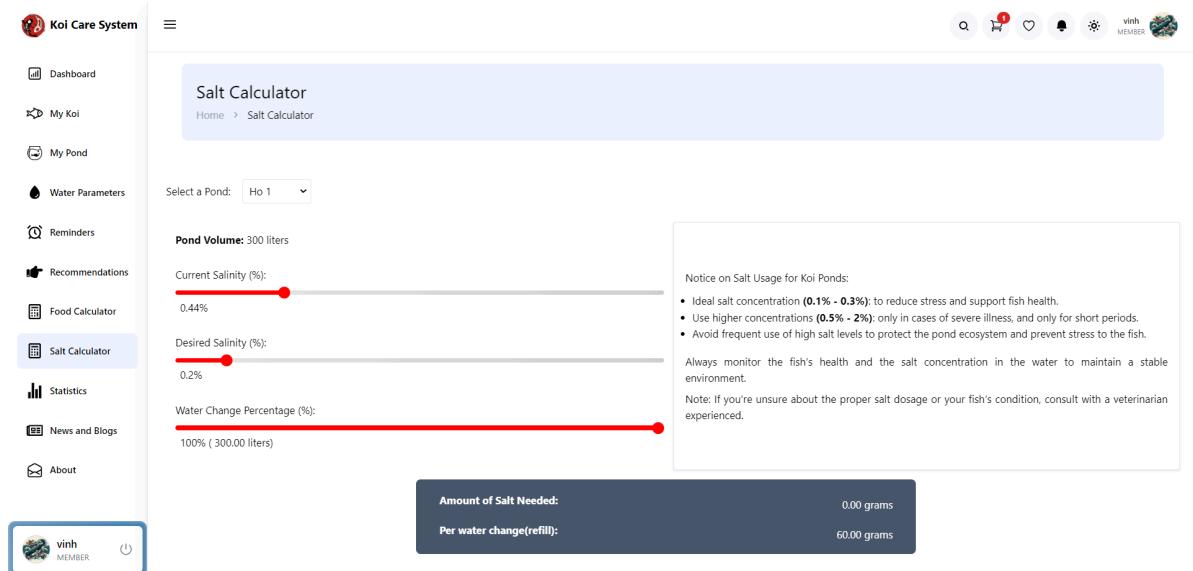
- **Inputs:**
 - Pond Name
 - Current Salinity

- Desired Salinity
 - Water Change Percentage
- **Outputs:**
 - Recommended amount of salt (in grams)
 - Per water change

Process:

- The member inputs the pond name and desired salt concentration into the calculator.
- The system uses predefined formulas to determine the appropriate amount of salt needed.
- The recommended salt amount is displayed along with any additional application instructions.

Screen Layout:



Function Details:

- **Data:** Pond volume, desired salt concentration
- **Validation:**
 - Pond volume must be a positive number.
 - Desired salt concentration must be a valid value (within acceptable range).
- **Business Logic:**
 - **Normal Cases:** The recommended amount of salt is successfully calculated and displayed.
 - **Abnormal Cases:** If invalid inputs are provided (e.g., negative volume), an error message is shown prompting the user to correct the inputs.

16. Statistics

FR-32: Statistics - View Statistics About Koi Ponds and Koi Fish

Function Trigger: The function is triggered when a member navigates to the "Statistics" section from the dashboard or main menu.

Function Description:

- **Actors/Roles:** Member
- **Purpose:** To allow members to view statistics related to their koi ponds and koi fish, providing insights into their care and management.
- **Interface:** Statistics Page

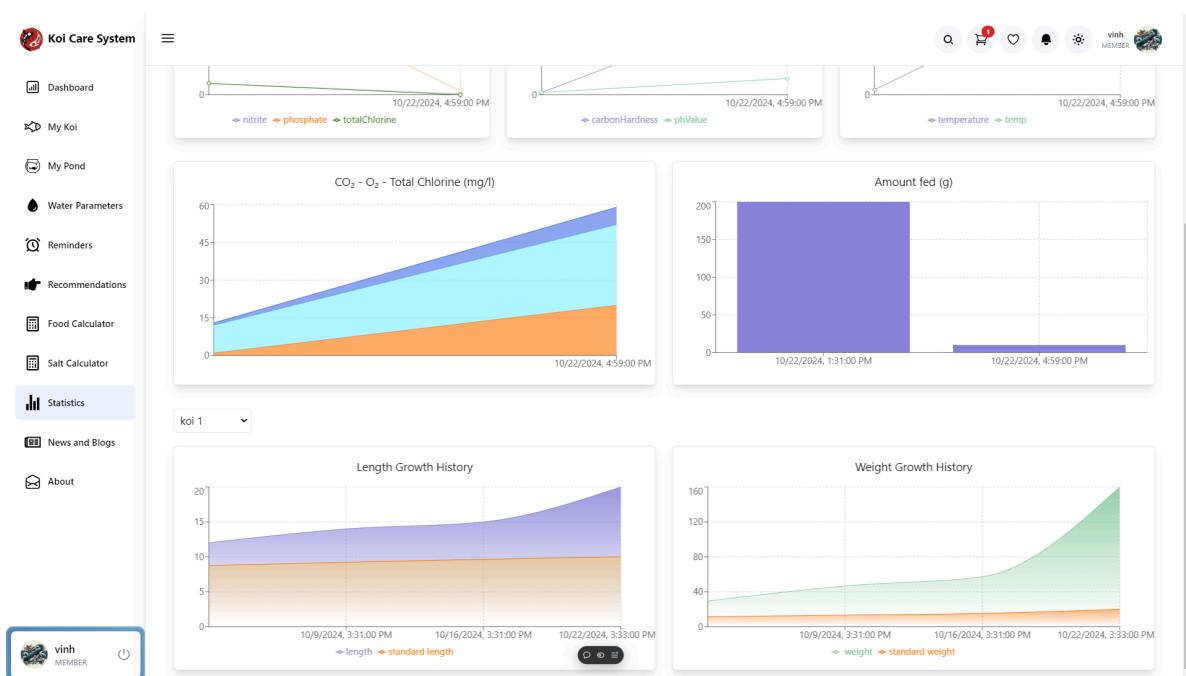
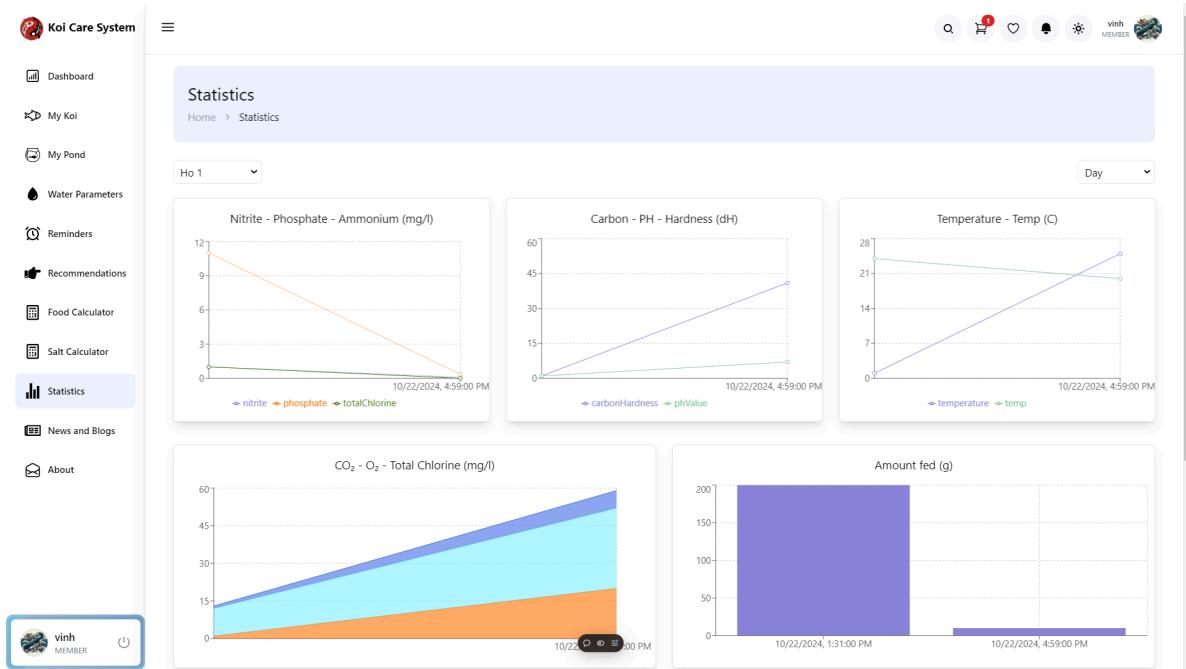
Data Processing:

- **Inputs:**
 - **User Authentication Token:** Verifies the member's identity to retrieve relevant statistics.
- **Outputs:**
 - A summary of statistics including:
 - Nitrite, Phosphate, Ammonium
 - Carbon, PH, Hardness
 - Temperature, Temperature
 - CO₂, O₂, Total Chlorine
 - Amount fed
 - Length, Weight

Process:

- The system retrieves the **User Authentication Token** and fetches statistics from the database related to the member's koi ponds and fish.
- The statistics are displayed in a visually appealing format, such as charts or tables, for easy interpretation.
- Members can view insights into their koi care, pond health, and overall statistics.

Screen Layout:



Function Details:

- **Data:** Statistics related to koi ponds and koi fish, including counts, metrics, and historical data.

Validation: Not applicable, as this function focuses on displaying data.

Business Logic:

- **Normal Cases:** The system successfully retrieves and displays the relevant statistics for the member's ponds and fish.

- **Abnormal Cases:** If the system fails to retrieve statistics (e.g., due to database issues), an error message is displayed, prompting the member to try again later.

This concise specification for the statistics function ensures that members can easily access important information about their koi ponds and fish, facilitating better care and management.

17. Member information

FR-33: Member Information - View and Update Member Profile

Function Trigger: The function is triggered when a member navigates to the "Member Information" section from the dashboard or account settings.

Function Description:

- **Actors/Roles:** Member
- **Purpose:** To allow members to view their profile details, update their information, and change their password.
- **Interface:** Member Profile Page

Data Processing:

- **Inputs:**
 - **User Authentication Token:** Verifies the member's identity to retrieve and update profile information.
 - **Profile Information:**
 - **Name:** Full name of the member (required).
 - **Email:** Member's email address (required).
 - **Gender**
 - **Date of birth**
 - **Created date**
 - **Bio**
 - **Phone:** Contact number (optional).
 - **Address:** Shipping or billing address (optional).
 - **New Password:** For changing the current password (optional).
- **Outputs:**
 - Updated member profile information in the database.
 - Confirmation messages upon successful updates or error messages for invalid inputs.

Process:

- The system retrieves the **User Authentication Token** and fetches the member's current profile information.
- The member can view their profile details and make changes as needed.
- The system validates the input fields:
 - Required fields must not be empty.
 - Password change must follow the required security criteria (e.g., minimum length).

- Upon successful validation, the system updates the profile information in the database.
- The member receives a confirmation message upon successful updates.

Screen Layout:

Profile Information

Name: vinh Email: sandilove59@gmail.com

Phone: 01234567822 Gender: Male

Date of Birth: 10/22/2024 Created Date: 10/20/2024

Address: TPHCM

Bio: zzzz

Change Password

Current Password:

New Password:

Confirm New Password:

Change Password

Function Details:

- Data:** Member profile information, including name, email, phone, address, and password.

Validation:

- Required fields must not be empty.
- Passwords must meet security requirements.

Business Logic:

- Normal Cases:** The system successfully retrieves and displays the member's profile information, allowing for updates and password changes.
- Abnormal Cases:** If required fields are missing or invalid, an error message is displayed, prompting the member to correct the input. If the password change fails (e.g., due to non-matching fields), an appropriate error message is shown.

18. Dashboard Shop

18.1 FR-32: Dashboard Shop

Function Trigger:

The function is triggered when a Shop logs in and navigates to the "Shop Dashboard".

Function Description:

- **Actors/Roles:** Shop
- **Purpose:** To provide an overview of key metrics and operations relevant to the shop staff, including users and product data, total information with earnings displayed by chart, and new orders.
- **Interface:** Shop Dashboard Page

Data Processing:

- **Inputs:**
 - User Authentication Token
- **Outputs:**
 - Overview metrics displayed on the dashboard, including:
 - Total Users
 - Total Orders
 - Total Payments
 - Total Suppliers
 - Total Products
 - Total Blogs
 - Top products
 - Order and Payments Analytics
 - Earnings by category
 - User reports
 - New Orders

Process:

- The system retrieves real-time data relevant to the shop staff from the database based on the user's account.
- Key metrics are calculated and displayed visually, possibly using charts or graphs for easier interpretation.
- The dashboard may also include links to detailed reports or other sections for further action.

Screen Layout:

The screenshot shows the Koi Care System dashboard. On the left is a sidebar with navigation links: Dashboard, Manage User, Manage News, Manage Shop, and Manage Report. The main area has a header "Dashboard" and a breadcrumb "Home > Dashboard". Below the header are six cards: "Total Users" (6), "Total Orders" (0), "Total Payments" (0), "Total Supplier" (1), "Total Products" (1), and "Total Blogs" (1). To the right is a section titled "Top Products" with a table showing one item: "Mr. Kodama Koi Food - Small Pack 50/50 for Growth" with a green progress bar indicating 0% and a total price of 0d. At the bottom are two sections: "Orders and Payments Analytics" and "Earning by Category".

Function Details:

- **Data:** Sales data, order data, inventory levels, customer ratings
- **Validation:**
 - Not applicable; data is fetched based on user authentication.
- **Business Logic:**
 - **Normal Cases:** The dashboard displays accurate and up-to-date metrics for the shop staff.
 - **Abnormal Cases:** If there are issues retrieving data (e.g., network errors), an error message is shown indicating that the dashboard is unable to load data at this time.

19. Manage User

FR-35: Manage User - View List of Users

Function Trigger: The function is triggered when an Admin and Shop navigates to the "Manage Users" section from the dashboard or main menu.

Function Description:

- **Actors/Roles:** Shop, Admin
- **Purpose:** To allow Shop, Admin to view a comprehensive list of user profiles within the system.
- **Interface:** User Management Page

Data Processing:

- **Inputs:**
 - **Authentication Token:** Verifies identity and access privileges to view user profiles.
- **Outputs:**
 - A paginated or scrollable list of user profiles, displaying key information:
 - **Image**

- **Name**
- **Email**
- **Phone**
- **Address**
- **Create Date**
- **Role**
- **Action**

Process:

- The system retrieves the **Authentication Token** and verifies access level.
- The system fetches the list of user profiles from the database.
- The user profiles are displayed in a grid or list format, allowing to view key details.
- Shop and admin can filter, sort, or search the user list based on various criteria (e.g., name, role, status).

Screen Layout:

Image	Name	Email	Phone	Address	Create Date	Status	Action
	An Phuoc Dao	anphuocda2410@gmail.com			2024-10-20	★ PREMIUM	
	vinh	sandilove59@gmail.com	01234567822	TPHCM	2024-10-20	★ PREMIUM	
	thoai	nvmthoai12@gmail.com			2024-10-20	★ PREMIUM	
	thoai12	nvmthoai1473837@gmail.com			2024-10-21	★ PREMIUM	
	An Hoang Q...	anhoang800140@gmail.com			2024-10-21	★ PREMIUM	
	D	Dao Cong An...	phuocdcase180581@fpt.edu.vn		2024-10-22	NORMAL	

Rows per page: 100 < 1-6 of 6 >

Function Details:

- **Data:** User profile information, including user ID, name, email, phone, role, and status.

Validation: Not applicable, as this function primarily focuses on data retrieval and display.

Business Logic:

- **Normal Cases:** The system successfully retrieves and displays the list of user profiles, allowing the shop and admin to manage users effectively.

- **Abnormal Cases:** If the system fails to retrieve user profiles (e.g., due to database issues), an error message is displayed, prompting the shop to try again later.

FR-36: View User Profile

Function Trigger: The function is triggered when an Admin and Shop selects a user from the user list on the "Manage Users" page to view detailed information about that user.

Function Description:

- **Actors/Roles:** Admin and Shop
- **Purpose:** To allow Admin and Shops to view detailed profile information for a specific user within the system.
- **Interface:** User Profile Page

Data Processing:

- **Inputs:**
 - **Admin Authentication Token:** Verifies the Admin and Shop's identity and access privileges to view user profiles.
 - **User ID:** The unique identifier for the user whose profile is being viewed.
- **Outputs:**
 - Detailed information about the user, including:
 - **User ID**
 - **Name**
 - **Email**
 - **Phone**
 - **Role** (e.g., Member, Admin)
 - **Status** (e.g., Active, Inactive)
 - **Created Date**

Process:

- The system retrieves the **Admin Authentication Token** and verifies the Admin and Shop's access level.
- The Admin and Shop selects a user from the list, triggering the retrieval of the user's profile data.
- The system fetches the user profile information from the database based on the provided **User ID**.
- The user profile details are displayed on the profile page for the Admin and Shop to review.

Screen Layout:

- The user profile page presents the user information in a clear and organized format.

Function Details:

- **Data:** User profile information, including user ID, name, email, phone, role, status, creation date, last login date, and additional notes.

Validation: Not applicable.

Business Logic:

- **Normal Cases:** The system successfully retrieves and displays the detailed profile information for the selected user.
- **Abnormal Cases:** If the system fails to retrieve user profile information, an error message is displayed.

FR-37: View User's Payment Details

Function Trigger: The function is triggered when an Admin and Shop selects a user from the user list to view their payment details.

Function Description:

- **Actors/Roles:** Admin and Shop
- **Purpose:** To allow Admin and Shops to view detailed payment information for a specific user within the system.
- **Interface:** User Payment Details Page

Data Processing:

- **Inputs:**
 - **Admin Authentication Token**
 - **User ID**
- **Outputs:**
 - Detailed payment information for the user, including:
 - **Create date**
 - **Invoice code**
 - **Transaction code**
 - **Amount**
 - **Status**
 - **Action**

Process:

- The system verifies the **Admin Authentication Token**.
- The Admin and Shop selects a user and requests their payment details.
- The system fetches the payment information from the database based on the **User ID**.
- The payment details are displayed for the Admin and Shop to review.

Screen Layout:

Koi Care System

Users

Home > Users > User Details

NO	CREATE DATE	INVOICE CODE	TRANSACTION CODE	AMOUNT	STATUS	ACTION
1	18:16:58 22/10/2024	88580143	14626920	2.000.000đ	COMPLETED	

vinh

Email: sandilove59@gmail.com
Phone: 01234567822
Address: TPHCM
Gender: male
Bio: zzzz
Date of Birth: 2024-10-22
Create Date: 2024-10-20
Status: PREMIUM

Koi Care System

Users

Home > Users > User Details

ORDER DETAILS

Order ID: 1
Order Date: 18:16:10 22/10/2024
Address: 123 AA, Xã Võ Điện, Huyhn B?c Quang, T?nh Hà Giang
Phone: 0123456789
Note: aasd
Recipient Name: Kien
Status: PROCESSING
Total Amount: 2.000.000đ

Item Details

NO	PRODUCT NAME	QUANTITY	TOTAL PRICE
1	Mr. Kodama Koi Food – Small Pack 50/50 for Growth	1	2.000.000đ

vinh

Email: sandilove59@gmail.com
Phone: 01234567822
Address: TPHCM
Gender: male
Bio: zzzz
Date of Birth: 2024-10-22
Create Date: 2024-10-20
Status: PREMIUM

Function Details:

- Data:** Payment details for the user, including payment ID, user ID, amount paid, payment method, payment date, and status.

Validation: Not applicable.

Business Logic:

- Normal Cases:** The system successfully retrieves and displays the payment details for the selected user.

- **Abnormal Cases:** If the system fails to retrieve payment details, an error message is displayed.

FR-38: View User's Order Details

Function Trigger: The function is triggered when an Admin and Shop selects a user from the user list to view their order history.

Function Description:

- **Actors/Roles:** Admin and Shop
- **Purpose:** To allow Admin and Shops to view detailed order information for a specific user within the system.
- **Interface:** User Order Details Page

Data Processing:

- **Inputs:**
 - **Admin Authentication Token**
 - **User ID**
- **Outputs:**
 - Detailed order information for the user, including:
 - **Order ID**
 - **Order Date**
 - **Total Amount**
 - **Order Status** (e.g., Completed, Pending, Cancelled)
 - **Detail**

Process:

- The system verifies the **Admin Authentication Token**.
- The Admin and Shop selects a user and requests their order details.
- The system fetches the order information from the database based on the **User ID**.
- The order details are displayed for the Admin and Shop to review.

Screen Layout:

Users

Home > Users > User Details

NO	ORDER DATE	ORDER ID	TOTAL AMOUNT	STATUS	DETAIL
1	18:16:10 22/10/2024	1	2.000.000đ	PROCESSING	

vinh

Email: sandilove59@gmail.com
Phone: 01234567822
Address: TPHCM
Gender: male
Bio: zzzz
Date of Birth: 2024-10-22
Create Date: 2024-10-20
Status: PREMIUM

ORDER DETAILS

Order ID: 1
Order Date: 18:16:10 22/10/2024
Address: 123 AA, Xã Võ Đèm, Huy?n B?c Quang, T?nh Hà Giang
Phone: 0123456789
Note: aasd
RecipientName: Kien
Status: PROCESSING
Total Amount: 2.000.000đ

Item Details

NO	PRODUCT NAME	QUANTITY	TOTAL PRICE
1	Mr. Kodama Koi Food – Small Pack 50/50 for Growth	1	2.000.000đ

vinh

Email: sandilove59@gmail.com
Phone: 01234567822
Address: TPHCM
Gender: male
Bio: zzzz
Date of Birth: 2024-10-22
Create Date: 2024-10-20
Status: PREMIUM

Function Details:

- Data:** Order details for the user, including order ID, order date, total amount, order status, and items ordered.

Validation: Not applicable.

Business Logic:

- Normal Cases:** The system successfully retrieves and displays the order details for the selected user.
- Abnormal Cases:** If the system fails to retrieve order details, an error message is displayed.

20. Manage Order

FR-39: Manage Order

Function Trigger: The function is triggered when an Admin and Shop navigates to the "Manage Orders" section from the admin dashboard.

Function Description:

- **Actors/Roles:** Admin and Shop
- **Purpose:** To allow Admin and Shops to view and manage a list of orders, including updating the delivery status of each order.
- **Interface:** Manage Orders Page

Data Processing:

- **Inputs:**
 - **Admin Authentication Token:** Verifies the Admin and Shop's identity and access privileges to manage orders.
- **Outputs:**
 - A paginated or scrollable list of orders displaying key information, including:
 - **Order ID**
 - **Recipient Name**
 - **Phone**
 - **Address**
 - **Total Amount**
 - **Order Status** (e.g., Pending, Delivered)
 - **Action Buttons** (e.g., Update Status)

Process:

- The system retrieves the **Admin Authentication Token** and verifies the Admin and Shop's access level.
- The Admin and Shop requests to view the list of orders.
- The system fetches the order details from the database and displays them in a structured format.
- The Admin and Shop can select an order to update its delivery status.
- Upon updating the delivery status, the system validates the input and updates the order in the database.
- The system confirms the update and refreshes the order list.

Screen Layout:

Koi Care System

- Dashboard
- Manage User
- Manage News
- Manage Shop
- Manage Report
- Manage Order**
- Manage Payment

vinhshop SHOP

Orders

Home > Orders

Download Excel

	Order ID	Order Date	Recipient Name	Phone	Address	Total Amount	Order Status	Action
<input type="checkbox"/>	1	18:16:10 22/10/2024	Kien	0123456789	123 AA, Xã Võ Đèm, Huy?n B?c Quang, T?nh Hà Giang	2.000.000đ	PROCESSING	
<input type="checkbox"/>	2	18:22:03 22/10/2024	Lê L?c	0343514996	Thu Duc City, Ho Chi Minh City, Viet Nam, ..	2.000.000đ	PENDING	

User vinhshop SHOP

Koi Care System

- Dashboard
- Manage User
- Manage News
- Manage Shop
- Manage Report
- Manage Order**
- Manage Payment

vinhshop SHOP

Orders

Home > Orders

ORDER DETAILS

Order ID: 1
Order Date: 18:16:10 22/10/2024
Address: 123 AA, Xã Võ Đèm, Huy?n B?c Quang, T?nh Hà Giang
Phone: 0123456789
Note: aaasd
RecipientName: Kien
Status: PROCESSING
Total Amount: 2.000.000đ

Item Details

NO	PRODUCT NAME	QUANTITY	TOTAL PRICE
1	Mr. Kodama Koi Food – Small Pack 50/50 for Growth	1	2.000.000đ

Close

Download Excel

Total Amount	Order Status	Action
2.000.000đ	PROCESSING	
2.000.000đ	PENDING	

User vinhshop SHOP

The screenshot shows the 'Orders' section of the Koi Care System. A modal window titled 'koi-care-system.vercel.app says' asks 'Are you sure to update status delivery?' with 'OK' and 'Cancel' buttons. Below the modal is a table with two rows of order data:

Order ID	Order Date	Recipient Name	Phone	Address	Total Amount	Order Status	Action
1	18:16:10 22/10/2024	Kien	0123456789	123 AA, Xã Võ Đèm, Huyện Bến Cát, Tỉnh Bình Phước	2.000.00đ	PROCESSING	
2	18:22:03 22/10/2024	Lê Lợi	0343514996	Thủ Đức City, Hồ Chí Minh City, Vietnam, ..	2.000.00đ	PENDING	

At the bottom left is a user profile for 'User vinhshop SHOP'. The top right shows search, filter, and account icons.

Function Details:

- **Data:** Order information including order ID, user ID, order date, total amount, order status, delivery status, and action buttons for updates.

Validation:

- The system verifies that the delivery status update is valid before applying changes.

Business Logic:

- **Normal Cases:**
 - The system successfully retrieves and displays the list of orders.
 - When the delivery status is updated, the change is reflected in the database, and a success message is displayed to the Admin and Shop.
- **Abnormal Cases:**
 - If the system fails to retrieve orders (e.g., due to database issues), an error message is displayed.
 - If the delivery status update fails due to invalid input, an error message prompts the Admin and Shop to check the input and try again.

21. Manage Payment

FR-40: Manage Payment - View and Manage Payment Details

Function Trigger: The function is triggered when an Admin and Shop navigates to the "Manage Payments" section from the admin dashboard.

Function Description:

- **Actors/Roles:** Admin and Shop
- **Purpose:** To allow Admin and Shops to view and manage a list of payment details associated with user transactions.
- **Interface:** Manage Payments Page

Data Processing:

- **Inputs:**
 - **Admin Authentication Token:** Verifies the Admin and Shop's identity and access privileges to manage payments.
- **Outputs:**
 - A paginated or scrollable list of payments displaying key information, including:
 - **Invoice Code**
 - **Create Date**
 - **Transaction Code**
 - **Amount**
 - **Payment Status** (e.g., Completed, Pending, Failed)
 - **Action Buttons** (e.g., View Details, Update Status)

Process:

- The system retrieves the **Admin Authentication Token** and verifies the Admin and Shop's access level.
- The Admin and Shop requests to view the list of payments.
- The system fetches the payment details from the database and displays them in a structured format.
- The Admin and Shop can select a payment to view more details or update its status.
- Upon updating the payment status, the system validates the input and updates the payment record in the database.
- The system confirms the update and refreshes the payment list.

Screen Layout:

The screenshot shows the 'Payment' section of the Koi Care System. On the left is a sidebar with navigation links: Dashboard, Manage User, Manage News, Manage Shop, Manage Report, Manage Order, and Manage Payment (which is highlighted). The main area has a header 'Payment' and a breadcrumb 'Home > Payment'. A 'Download Excel' button is at the top right. Below is a table with one row:

	Invoice Code	Create Date	Transaction Code	Amount	Status	Action
<input type="checkbox"/>	88580143	18:16:58 22/10/2024	14626920	2.000.000đ	COMPLETED	

User information at the bottom left: vinhshop SHOP.

This screenshot shows the same 'Payment' page as above, but with a modal window open over it. The modal is titled 'ORDER DETAILS' and contains the following information:

Order ID:	1
Order Date:	18:16:10 22/10/2024
Address:	123 AA, Xã Võ Đèo, Huy?n B?c Quang, T?nh Hà Giang
Phone:	0123456789
Note:	aaasd
RecipientName:	Kien
Status:	DELIVERED
Total Amount:	2.000.000đ

Below the modal is a 'Item Details' table:

NO	PRODUCT NAME	QUANTITY	TOTAL PRICE
1	Mr. Kodama Koi Food – Small Pack 50/50 for Growth	1	2.000.000đ

At the bottom of the modal are 'Close' and 'OK' buttons.

User information at the bottom left: vinhshop SHOP.

Function Details:

- Data:** Payment information including payment ID, user ID, order ID, payment date, amount paid, payment status, and action buttons for updates.

Validation:

- The system verifies that the payment status update is valid before applying changes.

Business Logic:

- Normal Cases:**

- The system successfully retrieves and displays the list of payments.
 - When the payment status is updated, the change is reflected in the database, and a success message is displayed to the Admin and Shop.
- **Abnormal Cases:**
 - If the system fails to retrieve payments (e.g., due to database issues), an error message is displayed.
 - If the payment status update fails due to invalid input, an error message prompts the Admin and Shop to check the input and try again.

22. Manage Blog Tag

FR-41: Manage Blog Tag - List and Manage Blog Tags

Function Trigger: The function is triggered when an Admin and Shop navigates to the "Manage Blog Tags" section from the admin dashboard.

Function Description:

- **Actors/Roles:** Admin and Shop
- **Purpose:** To allow Admin and Shops to view and manage a list of tags used for categorizing blog posts.
- **Interface:** Manage Blog Tags Page

Data Processing:

- **Inputs:**
 - **Admin Authentication Token:** Verifies the Admin and Shop's identity and access privileges to manage blog tags.
- **Outputs:**
 - A paginated or scrollable list of blog tags displaying key information, including:
 - **Tag Name**
 - **Description**
 - **Action Buttons** (e.g., View Details, Edit, Delete)

Process:

- The system retrieves the **Admin Authentication Token** and verifies the Admin and Shop's access level.
- The Admin and Shop requests to view the list of blog tags.
- The system fetches the blog tag details from the database and displays them in a structured format.
- The Admin and Shop can select a tag to view more details, edit, or delete it.
- If the Admin and Shop chooses to edit or delete a tag, the system validates the action and updates the database accordingly.
- The system confirms any updates or deletions and refreshes the blog tags list.

Screen Layout:

Function Details:

- **Data:** Blog tag information including tag ID, tag name, number of associated posts, and action buttons for management.

Validation:

- The system verifies that any edits or deletions are valid before applying changes.

Business Logic:

- **Normal Cases:**
 - The system successfully retrieves and displays the list of blog tags.
 - Any updates or deletions are reflected in the database, and a success message is displayed to the Admin and Shop.
- **Abnormal Cases:**
 - If the system fails to retrieve blog tags (e.g., due to database issues), an error message is displayed.
 - If an attempt to delete or edit a tag fails due to invalid input or constraints, an error message prompts the Admin and Shop to check the input and try again.

FR-42: Create Tag

Function Trigger: The function is triggered when an Admin and Shop initiates the process to create a new blog tag via the "Manage Blog Tags" page or through a dedicated "Create Tag" button.

Function Description:

- **Actors/Roles:** Admin and Shop
- **Purpose:** To allow Admin and Shops to create a new tag for categorizing blog posts.
- **Interface:** Create Blog Tag Form

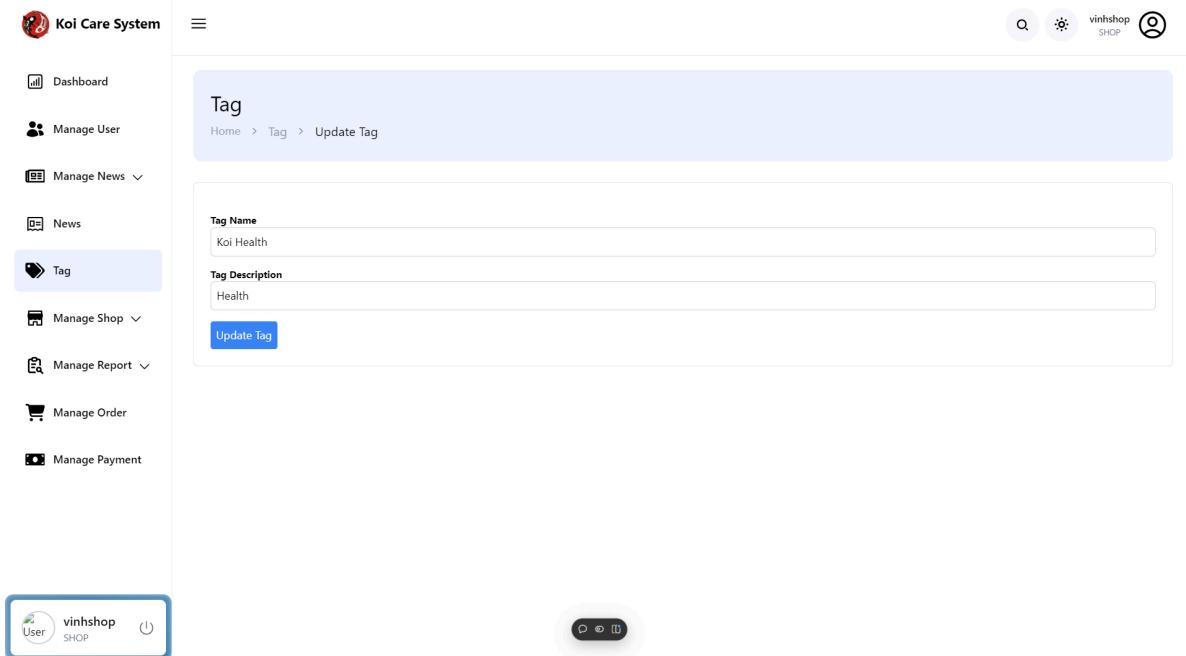
Data Processing:

- **Inputs:**
 - **Admin Authentication Token:** Verifies the Admin and Shop's identity and access privileges to create blog tags.
 - **Tag Name:** The name of the new tag (required).
 - **Tag Description**
- **Outputs:**
 - A confirmation message for successful tag creation.
 - The newly created tag appears in the list of blog tags.

Process:

- The system verifies the **Admin Authentication Token** and retrieves the Admin and Shop's information.
- The Admin and Shop inputs the details for the new tag.
- The system validates the input fields, ensuring that the tag name is not empty and unique.
- Upon successful validation, the new tag is saved to the database.
- The system displays a success message and updates the blog tags list.

Screen Layout:



Function Details:

- **Data:** New blog tag information including tag name.

Validation:

- Required fields must not be empty, and the tag name must be unique.

Business Logic:

- **Normal Cases:**
 - The system successfully creates the new tag and saves it to the database, displaying a success message to the Admin and Shop.
- **Abnormal Cases:**
 - If the tag name is empty or already exists, an error message is displayed, prompting the Admin and Shop to correct the input.

FR-43: Update Tag - Update Existing Blog Tags

Function Trigger: The function is triggered when an Admin and Shop selects an existing blog tag from the "Manage Blog Tags" page to edit it.

Function Description:

- **Actors/Roles:** Admin and Shop
- **Purpose:** To allow Admin and Shops to update the details of an existing blog tag.
- **Interface:** Update Blog Tag Form

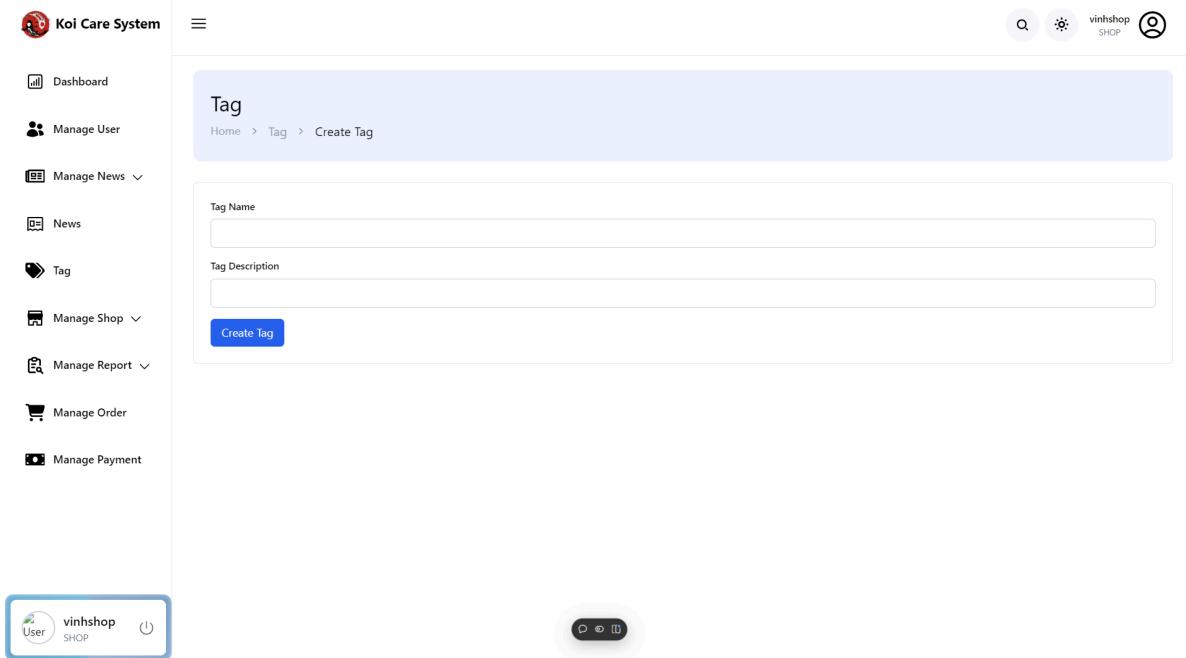
Data Processing:

- **Inputs:**
 - **Admin Authentication Token:** Verifies the Admin and Shop's identity and access privileges to update blog tags.
 - **Tag Name:** The new name for the tag (required).
 - **Tag Description**
- **Outputs:**
 - A confirmation message for successful tag update.
 - The updated tag appears in the list of blog tags.

Process:

- The system retrieves the **Admin Authentication Token** and verifies the Admin and Shop's access level.
- The Admin and Shop selects the tag to update and inputs the new tag name.
- The system validates the input fields, ensuring that the tag name is not empty and unique.
- Upon successful validation, the tag information is updated in the database.
- The system displays a success message and updates the blog tags list.

Screen Layout:



Function Details:

- **Data:** Updated blog tag information including tag ID and new tag name.

Validation:

- Required fields must not be empty, and the new tag name must be unique.

Business Logic:

- **Normal Cases:**
 - The system successfully updates the tag and saves it to the database, displaying a success message to the Admin and Shop.
- **Abnormal Cases:**
 - If the tag name is empty or already exists, an error message is displayed, prompting the Admin and Shop to correct the input.

23. Manage Blog

FR-44: Manage Blog

Function Trigger: The function is triggered when an Admin and Shop navigates to the "Manage Blog" section from the dashboard or main menu.

Function Description:

- **Actors/Roles:** Admin and Shop
- **Purpose:** To display a list of blog tags and manage them (e.g., view, edit, delete).
- **Interface:** Manage Blog Tags Page

Data Processing:

- **Inputs:**
 - **Admin Authentication Token:** Verifies the Admin and Shop's identity and access privileges to manage blog tags.
- **Outputs:**
 - A list of blog tags, including options to edit or delete each tag.

Process:

- The system verifies the **Admin Authentication Token** and retrieves the Admin and Shop's information.
- The system retrieves the list of blog tags from the database.
- The blog tags are displayed in a grid or list format, with actions available for each tag (e.g., edit, delete).

Screen Layout:

The screenshot shows the Koi Care System interface. On the left is a vertical sidebar with a navigation menu. The menu items include: Dashboard, Manage User, Manage News (selected), News, Tag, Manage Shop, Manage Report, Manage Order, and Manage Payment. At the bottom of the sidebar is a user profile section labeled 'User vinhshop SHOP' with a power button icon. The main content area has a header 'News' and a breadcrumb 'Home > News'. Below the header is a 'New Blog' button and a 'Download Excel' button. The main area displays a table of blog posts. The first post in the table has a preview image of a fish, the title 'A koi farm is a specialized...', the content 'A koi farm is a specialized facility focused on the breeding and ...', the date '2024-10-22', the tag 'Koi Health', and three action icons (edit, view, delete). The table has columns for Image, Blog Title, Blog Content, Date, Tag, and Action.

Function Details:

- **Data:** List of blog tags with options for managing each tag.

Validation:

- Not applicable.

Business Logic:

- **Normal Cases:**
 - The system successfully retrieves and displays the list of blog tags from the database based on the Admin and Shop's account.

- **Abnormal Cases:**

- If the system fails to retrieve the blog tags (e.g., due to database issues), an error message is displayed, indicating the issue and prompting the Admin and Shop to try again later.

FR-45: Create Blog

Function Trigger: The function is triggered when an Admin and Shop initiates the process to create a new blog post via the "Manage Blog" page or through a dedicated "Create Blog" button.

Function Description:

- **Actors/Roles:** Admin and Shop
- **Purpose:** To allow Admin and Shops to create a new blog post for publication.
- **Interface:** Create Blog Form

Data Processing:

- **Inputs:**

- **Admin Authentication Token:** Verifies the Admin and Shop's identity and access privileges to create blog posts.
- **Title:** The title of the blog post (required).
- **Image**
- **Content:** The main content/body of the blog post (required).
- **Tags:** Any associated tags for the blog post (optional).

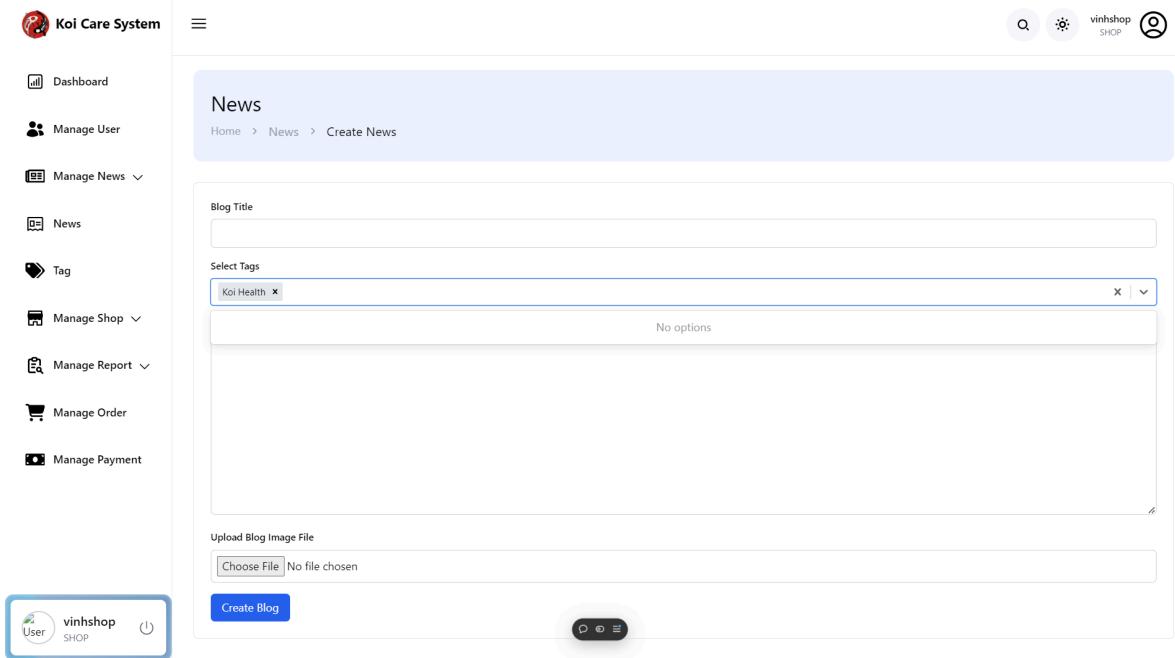
- **Outputs:**

- A confirmation message for successful blog creation.
- The newly created blog post appears in the list of blog posts.

Process:

- The system verifies the **Admin Authentication Token** and retrieves the Admin and Shop's information.
- The Admin and Shop inputs details for the new blog post.
- The system validates the input fields, ensuring that the title and content are not empty.
- Upon successful validation, the new blog post is saved to the database.
- The system displays a success message and updates the blog posts list.

Screen Layout:



Function Details:

- **Data:** New blog post information including title, content, and tags.

Validation:

- Required fields must not be empty.

Business Logic:

- **Normal Cases:**
 - The system successfully creates the new blog post and saves it to the database, displaying a success message to the Admin and Shop.
- **Abnormal Cases:**
 - If required fields are missing or invalid, an error message is displayed, prompting the Admin and Shop to correct the input.

FR-46: Update Blog

Function Trigger: The function is triggered when an Admin and Shop selects an existing blog post from the "Manage Blog" page to edit it.

Function Description:

- **Actors/Roles:** Admin and Shop
- **Purpose:** To allow Admin and Shops to update the details of an existing blog post.
- **Interface:** Update Blog Form

Data Processing:

- **Inputs:**

- **Admin Authentication Token:** Verifies the Admin and Shop's identity and access privileges to update blog posts.
- **Blog ID:** Identifier for the blog post to be updated (required).
- **Image**
- **Title:** The new title for the blog post (required).
- **Content:** The updated content/body of the blog post (required).
- **Tags:** Updated associated tags for the blog post (optional).
- **Outputs:**
 - A confirmation message for successful blog update.
 - The updated blog post appears in the list of blog posts.

Process:

- The system retrieves the **Admin Authentication Token** and verifies the Admin and Shop's access level.
- The Admin and Shop selects the blog post to update and inputs the new title and content.
- The system validates the input fields, ensuring that the title and content are not empty.
- Upon successful validation, the blog post information is updated in the database.
- The system displays a success message and updates the blog posts list.

Screen Layout:

Function Details:

- **Data:** Updated blog post information including blog ID, new title, content, and tags.

Validation:

- Required fields must not be empty.

Business Logic:

- **Normal Cases:**
 - The system successfully updates the blog post and saves it to the database, displaying a success message to the Admin and Shop.
- **Abnormal Cases:**
 - If required fields are missing or invalid, an error message is displayed, prompting the Admin and Shop to correct the input.

FR-47: View Blog

Function Trigger: The function is triggered when a user clicks on a blog post title from the blog listing page.

Function Description:

- **Actors/Roles:** Admin and Shop
- **Purpose:** To display the details of a specific blog post.
- **Interface:** Blog Post Detail Page

Data Processing:

- **Inputs:**
 - **Blog ID:** Identifier for the blog post to be viewed (required).
- **Outputs:**
 - The full content of the selected blog post, including title, content, tags, and publication date.

Process:

- The system retrieves the blog post details using the provided Blog ID.
- The blog post details are displayed on the blog post detail page, including the title, content, tags, and publication date.

Screen Layout:

The screenshot shows the Koi Care System dashboard. On the left is a sidebar with navigation links: Dashboard, Manage User, Manage News (selected), News, Tag, Manage Shop, Manage Report, Manage Order, and Manage Payment. The main content area displays a blog post titled "A koi farm is a specialized facility focused on the breeding and nurturing of koi fish". The post content discusses the breeding process, selecting parent koi, optimal conditions, and the life cycle of koi fry. It includes a timestamp "2024-10-22" and a "Share this post" button with icons for Facebook, Twitter, and LinkedIn.

Function Details:

- **Data:** Blog post information including blog ID, title, content, and tags.

Validation:

- Not applicable.

Business Logic:

- **Normal Cases:**
 - The system successfully retrieves and displays the blog post details based on the blog ID.
- **Abnormal Cases:**
 - If the blog post cannot be found (e.g., invalid blog ID), an error message is displayed, indicating that the blog post does not exist.

24. Manage Supplier

FR-48: Manage Supplier - List Suppliers

Function Trigger: The function is triggered when an Admin and Shop navigates to the "Manage Supplier" section from the dashboard or main menu.

Function Description:

- **Actors/Roles:** Admin and Shop
- **Purpose:** To display a list of suppliers registered in the system.
- **Interface:** Supplier Management Page

Data Processing:

- **Inputs:**
 - **Admin Authentication Token:** Verifies the Admin and Shop's identity and access privileges to manage suppliers.
- **Outputs:**
 - A list of suppliers, including key information such as name, contact details, and actions for editing or deleting.

Process:

- The system verifies the **Admin Authentication Token** and retrieves the Admin and Shop's information.
- The system retrieves the list of suppliers from the database.
- The suppliers are displayed in a grid or list format, with actions available for each supplier (e.g., edit, delete).

Screen Layout:

The screenshot shows the Koi Care System interface. On the left is a vertical sidebar with a dark blue header containing the 'Koi Care System' logo. Below the header, there are several menu items with icons: Dashboard (home icon), Manage User (user icon), Manage News (newspaper icon), Manage Shop (shop icon), **Supplier** (box icon, highlighted with a blue box), Promotion (promotion icon), Category (grid icon), Images (camera icon), Product (product icon), Manage Report (document icon), Manage Order (cart icon), and Manage Payment (credit card icon). At the bottom of the sidebar is a user profile section with a circular profile picture, the text 'User vinhshop SHOP', and a power button icon. The main content area has a light gray header with the title 'Supplier' and a breadcrumb trail 'Home > Supplier'. Below the header is a 'New Supplier' button. The main area displays a table with one row of data. The table columns are 'Name' (containing 'Koidama'), 'Phone' (containing '0972816453'), 'Address' (containing 'Thu Duc City, Ho Chi Minh City, Viet Nam'), and 'Action' (containing a green checkmark icon and a red trash can icon). At the bottom right of the main area are three small circular icons: a magnifying glass, a gear, and a list.

Function Details:

- **Data:** List of suppliers with options for managing each supplier.

Validation:

- Not applicable.

Business Logic:

- **Normal Cases:**

- The system successfully retrieves and displays the list of suppliers from the database based on the Admin and Shop's account.
- **Abnormal Cases:**
 - If the system fails to retrieve suppliers (e.g., due to database issues), an error message is displayed, indicating the issue and prompting the Admin and Shop to try again later.

FR-49: Create Supplier

Function Trigger: The function is triggered when an Admin and Shop initiates the process to create a new supplier via the "Manage Supplier" page or through a dedicated "Create Supplier" button.

Function Description:

- **Actors/Roles:** Admin and Shop
- **Purpose:** To allow Admin and Shops to create a new supplier profile for the system.
- **Interface:** Create Supplier Form

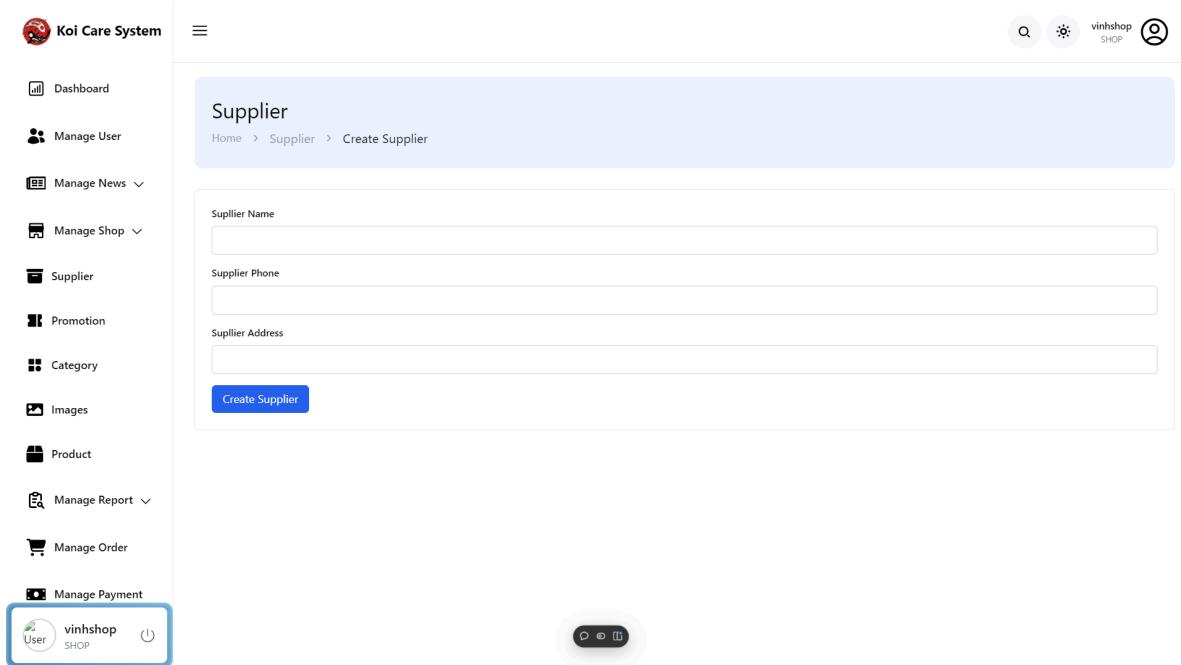
Data Processing:

- **Inputs:**
 - **Admin Authentication Token:** Verifies the Admin and Shop's identity and access privileges to create suppliers.
 - **Supplier Name:** The name of the supplier (required).
 - **Supplier Phone:** The contact details for the supplier (required).
 - **Address:** The physical address of the supplier (optional).
- **Outputs:**
 - A confirmation message for successful supplier creation.
 - The newly created supplier appears in the list of suppliers.

Process:

- The system verifies the **Admin Authentication Token** and retrieves the Admin and Shop's information.
- The Admin and Shop inputs details for the new supplier.
- The system validates the input fields, ensuring that the supplier name and contact information are not empty.
- Upon successful validation, the new supplier is saved to the database.
- The system displays a success message and updates the suppliers list.

Screen Layout:



Function Details:

- **Data:** New supplier information including name, contact information, address, and notes.

Validation:

- Required fields must not be empty.

Business Logic:

- **Normal Cases:**
 - The system successfully creates the new supplier and saves it to the database, displaying a success message to the Admin and Shop.
- **Abnormal Cases:**
 - If required fields are missing or invalid, an error message is displayed, prompting the Admin and Shop to correct the input.

FR-50: Edit Supplier

Function Trigger: The function is triggered when an Admin and Shop selects an existing supplier from the "Manage Supplier" page to edit its details.

Function Description:

- **Actors/Roles:** Admin and Shop
- **Purpose:** To allow Admin and Shops to update the details of an existing supplier.
- **Interface:** Edit Supplier Form

Data Processing:

- **Inputs:**

- **Admin Authentication Token:** Verifies the Admin and Shop's identity and access privileges to edit suppliers.
- **Supplier Name:** The new name for the supplier (required).
- **Supplier Phone:** Updated contact details for the supplier (required).
- **Address:** Updated physical address of the supplier (optional).
- **Outputs:**
 - A confirmation message for successful supplier update.
 - The updated supplier appears in the list of suppliers.

Process:

- The system retrieves the **Admin Authentication Token** and verifies the Admin and Shop's access level.
- The Admin and Shop selects the supplier to update and inputs the new details.
- The system validates the input fields, ensuring that the supplier name and contact information are not empty.
- Upon successful validation, the supplier information is updated in the database.
- The system displays a success message and updates the suppliers list.

Screen Layout:

Function Details:

- **Data:** Updated supplier information including supplier ID, new name, contact information, address, and notes.

Validation:

- Required fields must not be empty.

Business Logic:

- **Normal Cases:**
 - The system successfully updates the supplier details and saves them to the database, displaying a success message to the Admin and Shop.
- **Abnormal Cases:**
 - If required fields are missing or invalid, an error message is displayed, prompting the Admin and Shop to correct the input.

25. Manage Promotion

FR-51: Manage Promotion - List Promotions

Function Trigger: The function is triggered when an Admin and Shop navigates to the "Manage Promotions" section from the dashboard or main menu.

Function Description:

- **Actors/Roles:** Admin and Shop
- **Purpose:** To display a list of promotions currently active or available in the system.
- **Interface:** Promotion Management Page

Data Processing:

- **Inputs:**
 - **Admin Authentication Token:** Verifies the Admin and Shop's identity and access privileges to manage promotions.
- **Outputs:**
 - A list of promotions, including key information such as title, start date, end date, and actions for editing or deleting.

Process:

- The system verifies the **Admin Authentication Token** and retrieves the Admin and Shop's information.
- The system retrieves the list of promotions from the database.
- The promotions are displayed in a grid or list format, with actions available for each promotion (e.g., edit, delete).

Screen Layout:

Koi Care System

- Dashboard
- Manage User
- Manage News
- Manage Shop
- Supplier
- Promotion
- Category
- Images
- Product
- Manage Report
- Manage Order
- Manage Payment

vinhshop SHOP

Promotions

Home > Promotions

New Promotion

Download Excel

Name	Start Date	End Date	Discount Rate	Description	Status	Products	Action
Promotion 15%	2020-00-23/10/2024	2020-00-24/10/2024	15	Promotion 15%	PENDING		

Koi Care System

- Dashboard
- Manage User
- Manage News
- Manage Shop
- Supplier
- Promotion
- Category
- Images
- Product
- Manage Report
- Manage Order
- Manage Payment

vinhshop SHOP

Promotions

Home > Promotions

New Promotion

PROMOTION DETAILS

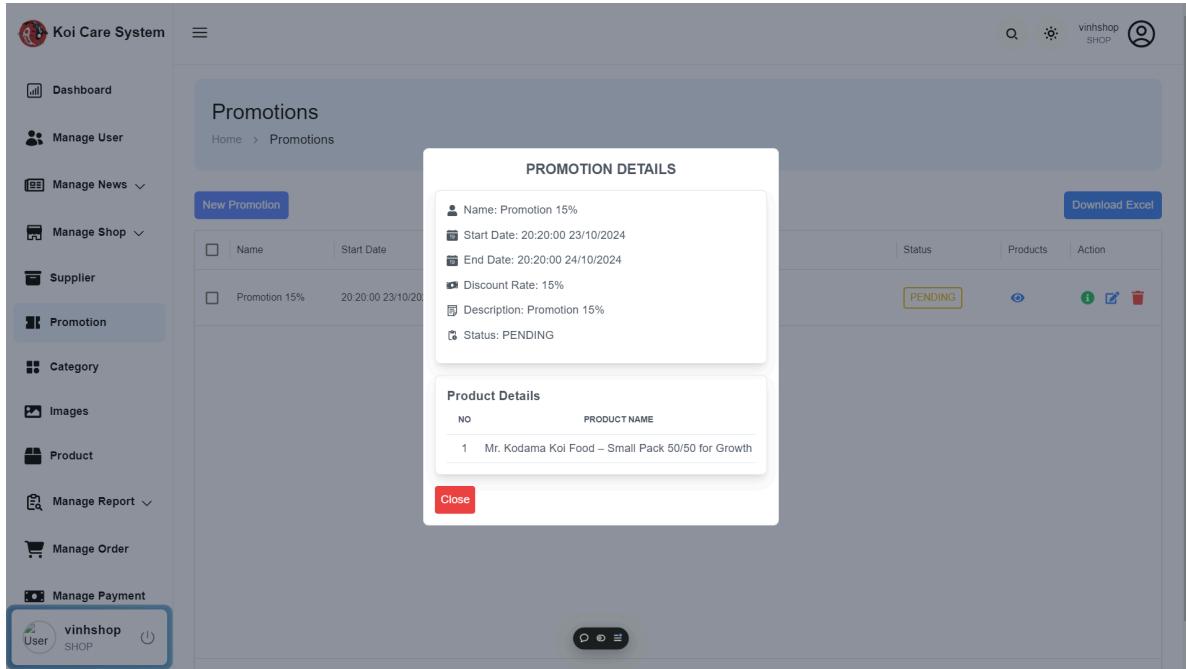
<input checked="" type="checkbox"/> Name	Start Date
<input checked="" type="checkbox"/> Promotion 15%	2020-00-23/10/2024
<input checked="" type="checkbox"/> End Date: 2020-00-24/10/2024	
<input checked="" type="checkbox"/> Discount Rate: 15%	
<input checked="" type="checkbox"/> Description: Promotion 15%	
<input checked="" type="checkbox"/> Status: PENDING	

Product Details

NO	PRODUCT NAME
1	Mr. Kodama Koi Food – Small Pack 50/50 for Growth

Close

Download Excel



Function Details:

- **Data:** List of promotions with options for managing each promotion.

Validation:

- Not applicable.

Business Logic:

- **Normal Cases:**
 - The system successfully retrieves and displays the list of promotions from the database based on the Admin and Shop's account.
- **Abnormal Cases:**
 - If the system fails to retrieve promotions (e.g., due to database issues), an error message is displayed, indicating the issue and prompting the Admin and Shop to try again later.

FR-52: Create Promotion

Function Trigger: The function is triggered when an Admin and Shop initiates the process to create a new promotion via the "Manage Promotions" page or through a dedicated "Create Promotion" button.

Function Description:

- **Actors/Roles:** Admin and Shop
- **Purpose:** To allow Admin and Shops to create a new promotion for products or services offered.

- **Interface:** Create Promotion Form

Data Processing:

- **Inputs:**
 - **Admin Authentication Token:** Verifies the Admin and Shop's identity and access privileges to create promotions.
 - **Promotion Name:** The title of the promotion (required).
 - **Select Product**
 - **Start Date:** The date when the promotion starts (required).
 - **End Date:** The date when the promotion ends (required).
 - **Discount Rate:** The percentage of discount offered (required).
 - **Description:** Additional details or notes about the promotion (optional).
- **Outputs:**
 - A confirmation message for successful promotion creation.
 - The newly created promotion appears in the list of promotions.

Process:

- The system verifies the **Admin Authentication Token** and retrieves the Admin and Shop's information.
- The Admin and Shop inputs details for the new promotion.
- The system validates the input fields, ensuring that the title, start date, end date, and discount percentage are not empty.
- Upon successful validation, the new promotion is saved to the database.
- The system displays a success message and updates the promotions list.

Screen Layout:

The screenshot shows the 'Create Promotion' form within the Koi Care System. The left sidebar includes links for Dashboard, Manage User, Manage News, Manage Shop, Supplier, Promotion, Category, Images, Product, Manage Report, and Manage Payment. The main content area has a header 'Promotion' and a breadcrumb trail 'Home > Promotion > Create Promotion'. The form fields include 'Promotion Name', 'Select Products' (with a note 'At least one product is required'), 'Start Date' (mm/dd/yyyy), 'End Date' (mm/dd/yyyy), 'Discount Rate' (percentage), and 'Description'. At the bottom are 'Create Promotion' and 'Cancel' buttons.

Function Details:

- **Data:** New promotion information including title, start date, end date, discount percentage, and description.

Validation:

- Required fields must not be empty.

Business Logic:

- **Normal Cases:**
 - The system successfully creates the new promotion and saves it to the database, displaying a success message to the Admin and Shop.
- **Abnormal Cases:**
 - If required fields are missing or invalid, an error message is displayed, prompting the Admin and Shop to correct the input.

FR-53: Edit Promotion

Function Trigger: The function is triggered when an Admin and Shop selects an existing promotion from the "Manage Promotions" page to edit its details.

Function Description:

- **Actors/Roles:** Admin and Shop
- **Purpose:** To allow Admin and Shops to update the details of an existing promotion.
- **Interface:** Edit Promotion Form

Data Processing:

- **Inputs:**
 - **Admin Authentication Token:** Verifies the Admin and Shop's identity and access privileges to edit promotions.
 - **Promotion Name:** The new title for the promotion (required).
 - **Product**
 - **Start Date:** Updated start date of the promotion (required).
 - **End Date:** Updated end date of the promotion (required).
 - **Discount Rate:** Updated percentage of discount offered (required).
 - **Description:** Updated additional details about the promotion (optional).
- **Outputs:**
 - A confirmation message for successful promotion update.
 - The updated promotion appears in the list of promotions.

Process:

- The system retrieves the **Admin Authentication Token** and verifies the Admin and Shop's access level.
- The Admin and Shop selects the promotion to update and inputs the new details.
- The system validates the input fields, ensuring that the title, start date, end date, and discount percentage are not empty.
- Upon successful validation, the promotion information is updated in the database.

- The system displays a success message and updates the promotions list.

Screen Layout:

Function Details:

- Data:** Updated promotion information including promotion ID, new title, start date, end date, discount percentage, and description.

Validation:

- Required fields must not be empty.

Business Logic:

- Normal Cases:**
 - The system successfully updates the promotion details and saves them to the database, displaying a success message to the Admin and Shop.
- Abnormal Cases:**
 - If required fields are missing or invalid, an error message is displayed, prompting the Admin and Shop to correct the input.

26. Manage Category

FR-54: Manage Category

Function Trigger: The function is triggered when an Admin and Shop navigates to the "Manage Categories" section from the dashboard or main menu.

Function Description:

- **Actors/Roles:** Admin and Shop
- **Purpose:** To display a list of product categories currently available in the system.
- **Interface:** Category Management Page

Data Processing:

- **Inputs:**
 - **Admin Authentication Token:** Verifies the Admin and Shop's identity and access privileges to manage categories.
- **Outputs:**
 - A list of product categories, including key information such as category name, description, and actions for editing or deleting.

Process:

- The system verifies the **Admin Authentication Token** and retrieves the Admin and Shop's information.
- The system retrieves the list of product categories from the database.
- The categories are displayed in a grid or list format, with actions available for each category (e.g., edit, delete).

Screen Layout:

Name	Action
Koi Food	<input checked="" type="checkbox"/> ✓ ✖

Rows per page: 100 ▾ 1–1 of 1 < >

Function Details:

- **Data:** List of product categories with options for managing each category.

Validation:

- Not applicable.

Business Logic:

- **Normal Cases:**
 - The system successfully retrieves and displays the list of product categories from the database based on the Admin and Shop's account.
- **Abnormal Cases:**
 - If the system fails to retrieve categories (e.g., due to database issues), an error message is displayed, indicating the issue and prompting the Admin and Shop to try again later.

FR-55: Create Category

Function Trigger: The function is triggered when an Admin and Shop initiates the process to create a new product category via the "Manage Categories" page or through a dedicated "Create Category" button.

Function Description:

- **Actors/Roles:** Admin and Shop
- **Purpose:** To allow Admin and Shops to create a new product category for organizing products.
- **Interface:** Create Category Form

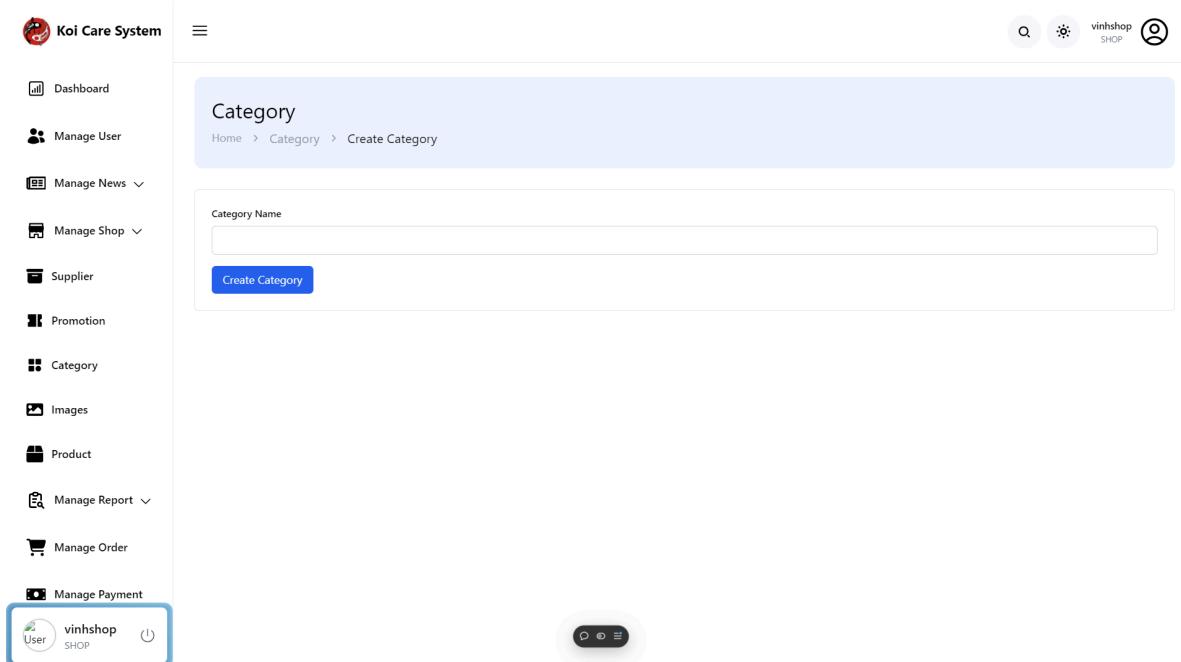
Data Processing:

- **Inputs:**
 - **Admin Authentication Token:** Verifies the Admin and Shop's identity and access privileges to create categories.
 - **Category Name:** The name of the new category (required).
- **Outputs:**
 - A confirmation message for successful category creation.
 - The newly created category appears in the list of product categories.

Process:

- The system verifies the **Admin Authentication Token** and retrieves the Admin and Shop's information.
- The Admin and Shop inputs details for the new category.
- The system validates the input fields, ensuring that the category name is not empty.
- Upon successful validation, the new category is saved to the database.
- The system displays a success message and updates the categories list.

Screen Layout:



Function Details:

- **Data:** New category information including name and description.

Validation:

- Required fields must not be empty.

Business Logic:

- **Normal Cases:**
 - The system successfully creates the new category and saves it to the database, displaying a success message to the Admin and Shop.
- **Abnormal Cases:**
 - If required fields are missing or invalid, an error message is displayed, prompting the Admin and Shop to correct the input.

FR-56: Edit Category

Function Trigger: The function is triggered when an Admin and Shop selects an existing category from the "Manage Categories" page to edit its details.

Function Description:

- **Actors/Roles:** Admin and Shop
- **Purpose:** To allow Admin and Shops to update the details of an existing product category.
- **Interface:** Edit Category Form

Data Processing:

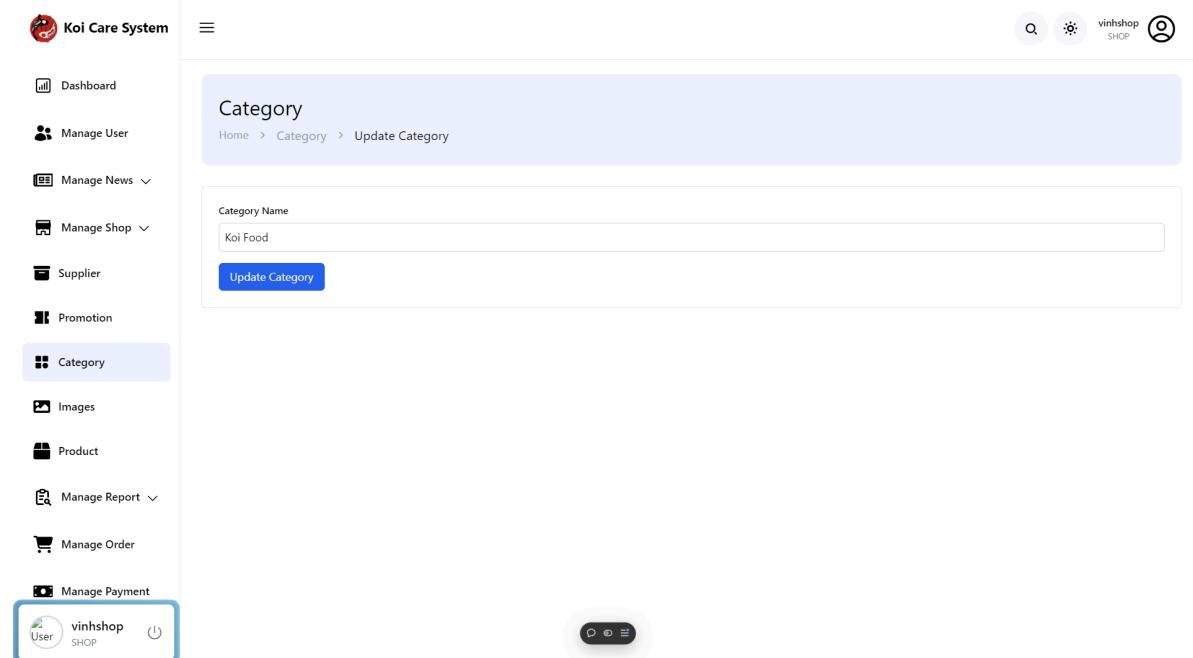
- **Inputs:**

- **Admin Authentication Token:** Verifies the Admin and Shop's identity and access privileges to edit categories.
- **Category Name:** The new name for the category (required).
- **Outputs:**
 - A confirmation message for successful category update.
 - The updated category appears in the list of product categories.

Process:

1. The system retrieves the **Admin Authentication Token** and verifies the Admin and Shop's access level.
2. The Admin and Shop selects the category to update and inputs the new details.
3. The system validates the input fields, ensuring that the category name is not empty.
4. Upon successful validation, the category information is updated in the database.
5. The system displays a success message and updates the categories list.

Screen Layout:



Function Details:

- **Data:** Updated category information including category ID, new name, and description.

Validation:

- Required fields must not be empty.

Business Logic:

- **Normal Cases:**
 - The system successfully updates the category details and saves them to the database, displaying a success message to the Admin and Shop.
- **Abnormal Cases:**

- If required fields are missing or invalid, an error message is displayed, prompting the Admin and Shop to correct the input.

27. Manage Image

FR-57: Manage Image

Function Trigger: The function is triggered when an Admin and Shop navigates to the "Manage Images" section from the dashboard or main menu.

Function Description:

- **Actors/Roles:** Admin and Shop
- **Purpose:** To display a list of product images currently available in the system.
- **Interface:** Image Management Page

Data Processing:

- **Inputs:**
 - **Admin Authentication Token:** Verifies the Admin and Shop's identity and access privileges to manage images.
- **Outputs:**
 - A list of product images, including key information such as image name, associated product, and actions for editing or deleting.

Process:

- The system verifies the **Admin Authentication Token** and retrieves the Admin and Shop's information.
- The system retrieves the list of product images from the database.
- The images are displayed in a grid or list format, with actions available for each image (e.g., edit, delete).

Screen Layout:

Function Details:

- **Data:** List of product images with options for managing each image.

Validation:

- Not applicable.

Business Logic:

- **Normal Cases:**
 - The system successfully retrieves and displays the list of product images from the database based on the Admin and Shop's account.
- **Abnormal Cases:**
 - If the system fails to retrieve images (e.g., due to database issues), an error message is displayed, indicating the issue and prompting the Admin and Shop to try again later.

FR-58: Create Image

Function Trigger: The function is triggered when an Admin and Shop initiates the process to add new images for a product via the "Manage Images" page or through a dedicated "Add Image" button.

Function Description:

- **Actors/Roles:** Admin and Shop
- **Purpose:** To allow Admin and Shops to add new images for products in the system.
- **Interface:** Create Image Form

Data Processing:

- **Inputs:**
 - **Admin Authentication Token:** Verifies the Admin and Shop's identity and access privileges to add images.
 - **Product:** The identifier of the product for which the image is being added (required).
 - **Image File:** The new image file to be uploaded (required).
- **Outputs:**
 - A confirmation message for successful image upload.
 - The newly added image appears in the list of product images.

Process:

- The system verifies the **Admin Authentication Token** and retrieves the Admin and Shop's information.
- The Admin and Shop selects the product and uploads the new image file along with any optional description.
- The system validates the input fields, ensuring that the image file is provided and is of an acceptable format.
- Upon successful validation, the new image is saved to the database.
- The system displays a success message and updates the images list.

Screen Layout:

The screenshot shows the Koi Care System dashboard. On the left, there is a sidebar with various menu items: Dashboard, Manage User, Manage News, Manage Shop, Supplier, Promotion, Category, Images, Product, Manage Report, Manage Order, and Manage Payment. The 'Images' item is highlighted with a blue border. At the top right, there is a search icon, a gear icon for settings, and a user profile icon labeled 'vinhshop SHOP'. The main content area is titled 'Product Images' and shows a sub-path 'Home > Product Images > Create Images'. It contains two input fields: 'Product' (with a dropdown placeholder 'Select Product') and 'Upload Images' (with a file input field showing 'Choose Files | No file chosen'). Below these is a large blue button labeled 'Create Images'.

Function Details:

- **Data:** New image information including product ID, image file, and description.

Validation:

- Required fields must not be empty, and the uploaded file must be in an acceptable format.

Business Logic:

- **Normal Cases:**
 - The system successfully uploads the new image and saves it to the database, displaying a success message to the Admin and Shop.
- **Abnormal Cases:**
 - If required fields are missing or invalid, or if the uploaded file is in an unacceptable format, an error message is displayed, prompting the Admin and Shop to correct the input.

FR-59: Edit Image - Edit Product Image Details

Function Trigger: The function is triggered when an Admin and Shop selects an existing image from the "Manage Images" page to edit its details.

Function Description:

- **Actors/Roles:** Admin and Shop
- **Purpose:** To allow Admin and Shops to update the details of an existing product image.
- **Interface:** Edit Image Form

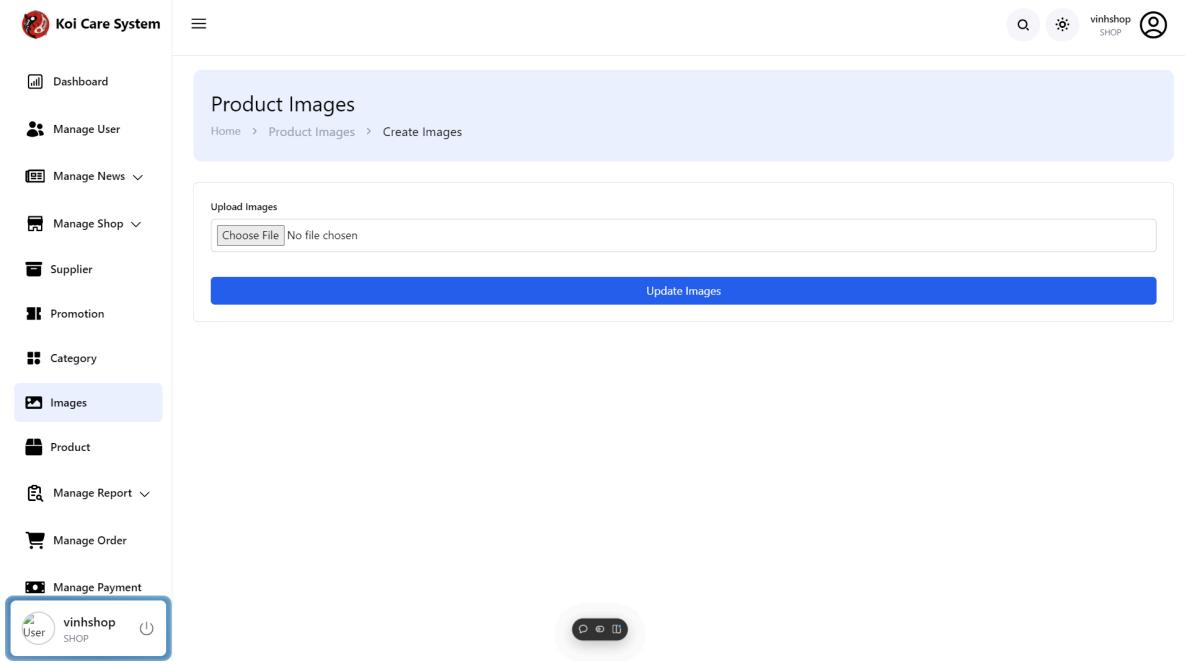
Data Processing:

- **Inputs:**
 - **Admin Authentication Token:** Verifies the Admin and Shop's identity and access privileges to edit images.
 - **Image File:** The new image file to be uploaded (optional).
- **Outputs:**
 - A confirmation message for successful image update.
 - The updated image appears in the list of product images.

Process:

- The system retrieves the **Admin Authentication Token** and verifies the Admin and Shop's access level.
- The Admin and Shop selects the image to update and inputs the new details, including uploading a new image file if desired.
- The system validates the input fields, ensuring that if an image file is provided, it is of an acceptable format.
- Upon successful validation, the image information is updated in the database.
- The system displays a success message and updates the images list.

Screen Layout:



Function Details:

- **Data:** Updated image information including image ID, new image file, and description.

Validation:

- If provided, the uploaded file must be in an acceptable format.

Business Logic:

- **Normal Cases:**
 - The system successfully updates the image details and saves them to the database, displaying a success message to the Admin and Shop.
- **Abnormal Cases:**
 - If required fields are missing or invalid, or if the uploaded file is in an unacceptable format, an error message is displayed, prompting the Admin and Shop to correct the input.

27. Manage Product

FR-60: Manage Product

Function Trigger: The function is triggered when an Admin and Shop navigates to the "Manage Products" section from the dashboard or main menu.

Function Description:

- **Actors/Roles:** Admin and Shop
- **Purpose:** To display a list of products currently available in the system.

- **Interface:** Product Management Page

Data Processing:

- **Inputs:**
 - **Admin Authentication Token:** Verifies the Admin and Shop's identity and access privileges to manage products.
- **Outputs:**
 - A list of products, including key information such as product name, category, price, stock status, and actions for editing or deleting.

Process:

- The system verifies the **Admin Authentication Token** and retrieves the Admin and Shop's information.
- The system retrieves the list of products from the database.
- The products are displayed in a grid or list format, with actions available for each product (e.g., edit, delete).

Screen Layout:

	Image	Product Name	Brand	Price	Inventory	Category	Supplier	Action
<input type="checkbox"/>		Mr. Kodama Koi Food – Small ...	Koimada	2.000.000đ	97	Koi Food	Koidama	

The screenshot shows the Koi Care System's product detail page. On the left is a sidebar with navigation links for Dashboard, Manage User, Manage News, Manage Shop, Supplier, Promotion, Category, Images, Product (selected), Manage Report, Manage Order, and Manage Payment. At the bottom of the sidebar is a user profile icon for 'User vinhshop' and a power button icon.

The main content area has a header 'Products' and a breadcrumb 'Home > Products > Detail'. It displays a large image of 'Mr. Kodama KOI FOOD ALL SEASON' koi food, which is red and white. Below the image are product details: 'In Stock' and 'Koi Food'. The product name is 'Mr. Kodama Koi Food – Small Pack 50/50 for Growth'. A description states: 'We are now offering Mr. Kodama 50/50 to those who may not have a big pond or many Koi, but still want the benefit of both foods to help your Koi grow healthy and beautiful.' The price is '2.000.000 ₫'. A rating of '★★★★★ (1 reviews)' is shown. A quantity selector with buttons for minus, one, and plus is present. Below the selector are two buttons: 'Buy Now' (blue) and 'Add to cart' (orange). Smaller images of other product variants are shown at the bottom left.

Function Details:

- **Data:** List of products with options for managing each product.

Validation:

- Not applicable.

Business Logic:

- **Normal Cases:**
 - The system successfully retrieves and displays the list of products from the database based on the Admin and Shop's account.
- **Abnormal Cases:**
 - If the system fails to retrieve products (e.g., due to database issues), an error message is displayed, indicating the issue and prompting the Admin and Shop to try again later.

FR-61: Create Product

Function Trigger: The function is triggered when an Admin and Shop initiates the process to create a new product via the "Manage Products" page or through a dedicated "Add Product" button.

Function Description:

- **Actors/Roles:** Admin and Shop
- **Purpose:** To allow Admin and Shops to create new products in the system.
- **Interface:** Create Product Form

Data Processing:

- **Inputs:**

- **Admin Authentication Token:** Verifies the Admin and Shop's identity and access privileges to create products.
- **Product Name:** The name of the new product (required).
- **Brand**
- **Issue Type**
- **Category:** The category to which the product belongs (required).
- **Price:** The price of the product (required).
- **Inventory:** The quantity of the product in stock (required).
- **Supplier:** The new name for the supplier (required).
- **Product Description:** Additional details about the product (optional).
- **Product Image:** An image file representing the product (optional).

- **Outputs:**

- A confirmation message for successful product creation.
- The newly created product appears in the product list.

Process:

- The system verifies the **Admin Authentication Token** and retrieves the Admin and Shop's information.
- The Admin and Shop inputs details for the new product.
- The system validates the input fields, ensuring that required fields are not empty and the price is valid.
- Upon successful validation, the new product is saved to the database.
- The system displays a success message and updates the products list.

Screen Layout:

The screenshot shows the 'Create Product' page of the Koi Care System. The left sidebar contains navigation links for Dashboard, Manage User, Manage News, Manage Shop, Supplier, Promotion, Category, Images, Product, Manage Report, Manage Order, and Manage Payment. The main area has a header 'Product' and a breadcrumb 'Home > Product > Create Product'. The form fields include: Product Name, Brand, Price, Inventory, Description, Detailed Description, Category (dropdown menu 'Select Category'), Supplier (dropdown menu 'Select Supplier'), Issue Type, and Upload Images (file input field 'Choose Files | No file chosen'). At the bottom is a blue 'Create Product' button.

Function Details:

- **Data:** New product information including name, category, price, stock quantity, description, and image.

Validation:

- Required fields must not be empty, and the price must be a valid number.

Business Logic:

- **Normal Cases:**
 - The system successfully creates the new product and saves it to the database, displaying a success message to the Admin and Shop.
- **Abnormal Cases:**
 - If required fields are missing or invalid, an error message is displayed, prompting the Admin and Shop to correct the input.

FR-62: Edit Product

Function Trigger: The function is triggered when an Admin and Shop selects an existing product from the "Manage Products" page to edit its details.

Function Description:

- **Actors/Roles:** Admin and Shop
- **Purpose:** To allow Admin and Shops to update the details of an existing product.
- **Interface:** Edit Product Form

Data Processing:

- **Inputs:**
 - **Admin Authentication Token:** Verifies the Admin and Shop's identity and access privileges to edit products.
 - **Product Name:** Updated name for the product (optional).
 - **Brand.**
 - **Supplier**
 - **Issue Type**
 - **Category:** Updated category for the product (optional).
 - **Price:** Updated price for the product (optional).
 - **Stock Quantity:** Updated stock quantity (optional).
 - **Product Description:** Updated additional details about the product (optional).
 - **Product Image:** A new image file for the product (optional).
- **Outputs:**
 - A confirmation message for successful product update.
 - The updated product appears in the product list.

Process:

- The system retrieves the **Admin Authentication Token** and verifies the Admin and Shop's access level.
- The Admin and Shop selects the product to update and inputs the new details.
- The system validates the input fields, ensuring that if provided, the price is valid.
- Upon successful validation, the product information is updated in the database.
- The system displays a success message and updates the products list.

Screen Layout:

The screenshot shows the 'Product' update screen in the Koi Care System. The left sidebar contains navigation links for Dashboard, Manage User, Manage News, Manage Shop, Supplier, Promotion, Category, Images, Product, and Manage Report. The main content area is titled 'Product' and shows a form for updating a product. The form includes fields for Product Name (Mr. Kodama Koi Food - Small Pack 50/50 for Growth), Brand (Koimada), Price (2000000), Inventory (97), Description (We are now offering Mr. Kodama 50/50 to those who may not have a big pond or many Koi, but still want the benefit of both foods to help your Koi grow healthy and beautiful.), and Detailed Description (Mixing different type of Koi food is what we usually recommend in order to maximize your Koi's potential and to meet your koi goals. Our goal is to help you grow your Koi healthy and enhance the color at the same time, and by doing so we mixed Mr. Kodama Koi Food All season and Mr. Kodama Koi Food Color Up by 50% each and created the Mr. Kodama 50/50 Koi Food Package.). Below these are dropdowns for Category (Koi Food), Supplier (Koldama), and Issue Type. At the bottom left is a user profile section labeled 'User' with 'User Name' and 'User Role' fields.

Function Details:

- **Data:** Updated product information including product ID, name, category, price, stock quantity, description, and image.

Validation:

- If provided, the price must be a valid number.

Business Logic:

- **Normal Cases:**
 - The system successfully updates the product details and saves them to the database, displaying a success message to the Admin and Shop.
- **Abnormal Cases:**
 - If required fields are missing or invalid, an error message is displayed, prompting the Admin and Shop to correct the input.

28. Dashboard Admin

FR-63: Dashboard Admin

Function Trigger: Activated when an Admin logs into the system and navigates to the "Dashboard" section.

Actors/Roles: Admin

Purpose: To provide an overview of the system's key metrics and statistics for the Admin.

Interface: Admin Dashboard Page

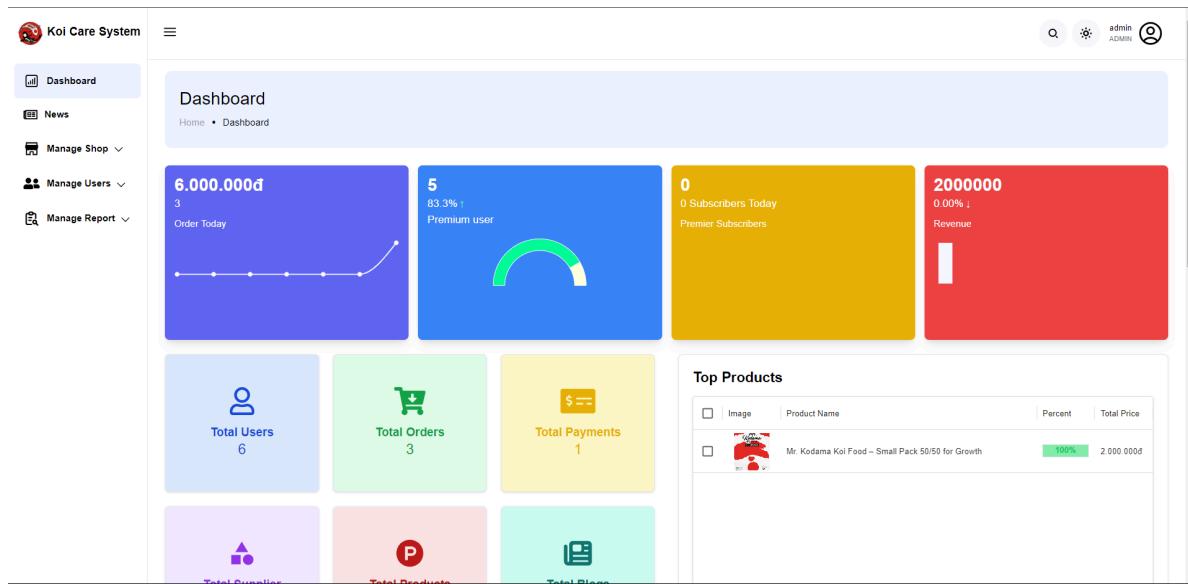
Data Processing:

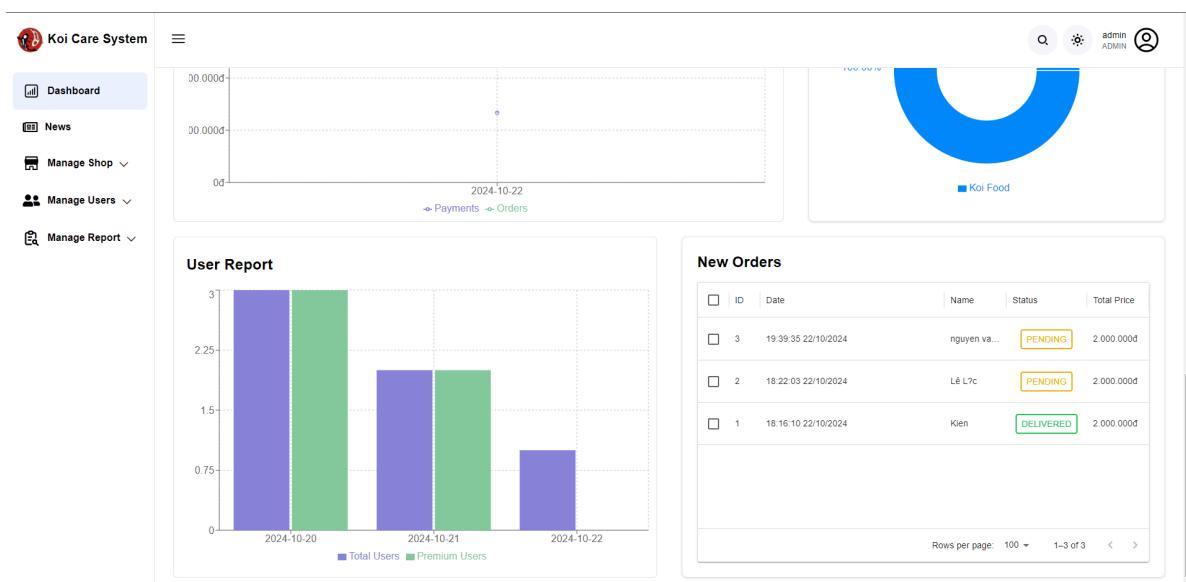
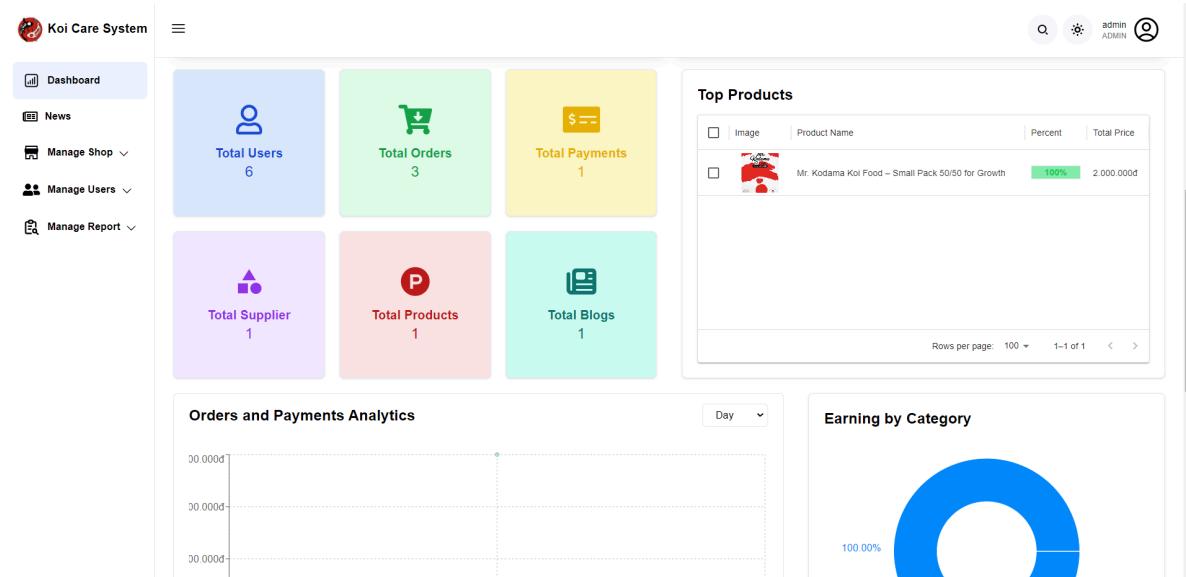
- **Inputs:** Admin authentication token
- **Outputs:**
 - Key system metrics (e.g., total users, sales, promotions, and orders)
 - Recent activity summary (e.g., recent promotions, new members)

Process:

- Fetch and display data related to system activity and key metrics in real-time.

Screen Layouts:





Function Details:

- Data:** System metrics, user counts, sales data
- Validation:** None required.
- Business Logic:**
 - Normal Cases:** The dashboard displays up-to-date data.
 - Abnormal Cases:** In case of data fetch errors, display an error message.

FR-64: Promotion Management

- Function Trigger:** Triggered when an Admin navigates to the "Promotions" section.
- Actors/Roles:** Admin
- Purpose:** To allow the admin to view, accept, or reject submitted promotions from shops.

- **Interface:** Promotion Management Page
- **Data Processing:**
 - **Inputs:** Admin authentication token
 - **Outputs:**
 - A list of promotions with status (Pending, Accepted, Processing, Rejected)
 - Action buttons to accept or reject promotions.
- **Process:**
 - Admin views the promotions and selects actions (accept/reject).
- **Screen Layouts:**

Name	Start Date	End Date	Discount Rate	Description	Products	Status
Promotion 15%	2020-09-23/10/2024	2020-09-24/10/2024	15	Promotion 15%		PENDING

- **Function Details:**
 - **Data:** Promotion details (name, description, shop, status)
 - **Validation:** None required.
 - **Business Logic:**
 - **Normal Cases:** Promotions are accepted/rejected successfully.
 - **Abnormal Cases:** An error message is displayed if the update fails.

FR-65: Manage Shop Staff

- **Function Trigger:** Triggered when Admin navigates to the "Manage Staff" section.
- **Actors/Roles:** Admin
- **Purpose:** To display a list of shop staff accounts and manage them.
- **Interface:** Shop Staff Management Page
- **Data Processing:**
 - **Inputs:** Admin authentication token
 - **Outputs:**
 - List of shop staff accounts with details (name, role, status).
 - Options to update or delete accounts.
- **Process:**
 - Fetch staff accounts associated with the shop.

- **Screen Layouts:**

	Username	Email	Role	Actions
<input type="checkbox"/>	vinhshop	shopvinh	SHOP	
<input type="checkbox"/>	thoishop	thoishop@kolicare.comany.vn	SHOP	
<input type="checkbox"/>	Minion	Minion@kolicare.comany.vn	SHOP	

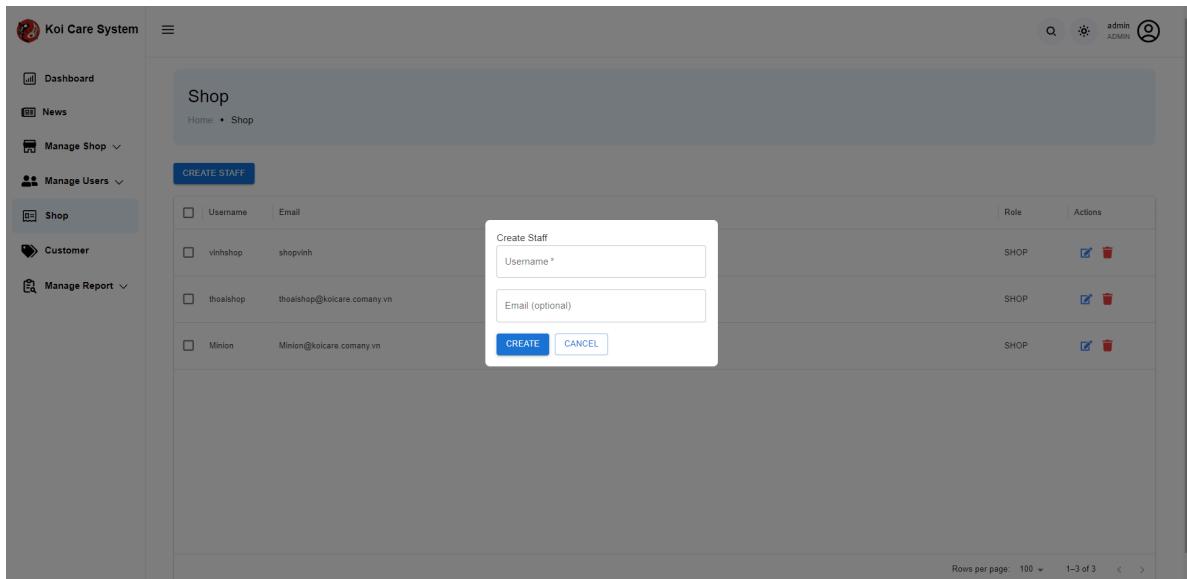
Rows per page: 100 ▾ 1-3 of 3 < >

- **Function Details:**

- **Data:** Staff details (name, role, email)
- **Validation:** None required.
- **Business Logic:**
 - **Normal Cases:** Staff accounts are successfully listed and managed.
 - **Abnormal Cases:** Error messages in case of failure.

FR-66: Create Shop Staff Account

- **Function Trigger:** Triggered when an Admin clicks on the "Create Shop Staff" button.
- **Actors/Roles:** Admin
- **Purpose:** To allow the creation of new shop staff accounts.
- **Interface:** Create Shop Staff Form
- **Data Processing:**
 - **Inputs:** Admin authentication token, staff details (name, email, role)
 - **Outputs:** A new shop staff account is created in the system.
- **Process:**
 - Admin fills in the form to create a new shop staff account.
- **Screen Layouts:**

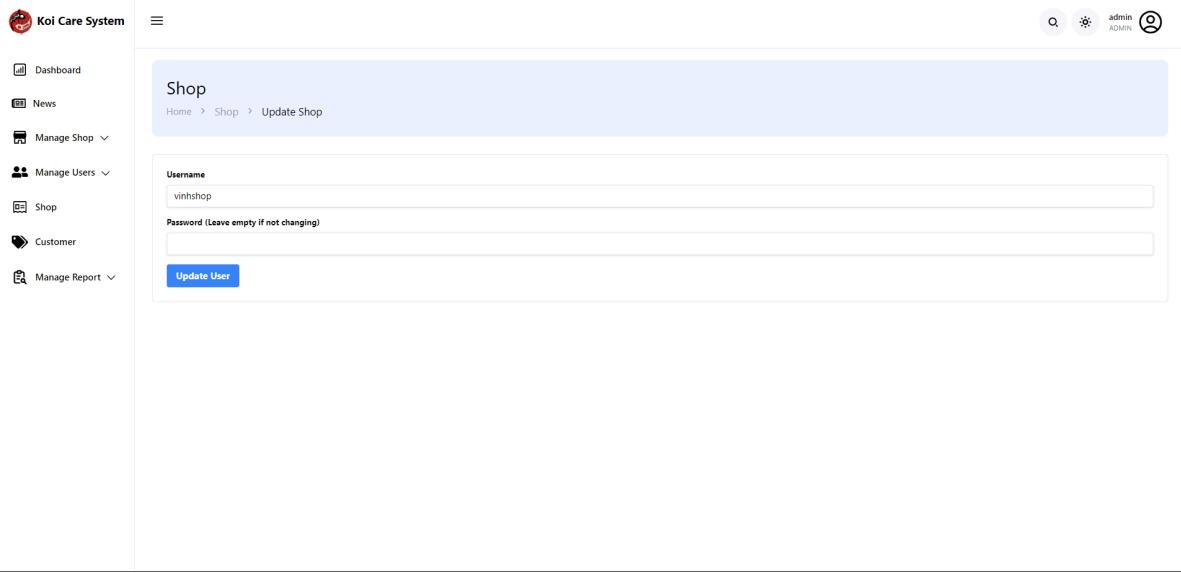


- **Function Details:**

- **Data:** Shop staff details (name, role, email)
- **Validation:** Mandatory fields must be filled.
- **Business Logic:**
 - **Normal Cases:** Staff account is successfully created.
 - **Abnormal Cases:** Error message is displayed for invalid inputs or creation failure.

FR-67: Change Shop Account Password

- **Function Trigger:** Triggered when an Admin navigates to the "Change Password" section for a shop staff account.
- **Actors/Roles:** Admin
- **Purpose:** To allow Admin to update shop account passwords.
- **Interface:** Change Password Form
- **Data Processing:**
 - **Inputs:** Admin authentication token, new password, staff ID
 - **Outputs:** The shop staff's password is updated in the database.
- **Process:**
 - Admin enters the new password and confirms it.
- **Screen Layouts:**



- **Function Details:**

- **Data:** New password
- **Validation:** Ensure the password meets the security requirements.
- **Business Logic:**
 - **Normal Cases:** Password is successfully updated.
 - **Abnormal Cases:** Error message if password update fails.

FR-68: Upgrade Premium for Member

- **Function Trigger:** Triggered when an Admin navigates to the "Upgrade to Premium" section for a member account.
- **Actors/Roles:** Admin
- **Purpose:** To upgrade member accounts to premium status.
- **Interface:** Premium Upgrade Page
- **Data Processing:**
 - **Inputs:** Admin authentication token, member ID
 - **Outputs:** The member's account is updated to premium status.
- **Process:**
 - Admin selects the member account and upgrades their status to premium.
- **Screen Layouts:**

Image	Name	Email	Phone	Address	Create Date	Date of Birth	bio	Role	Status	Action
	An Phuoc Dao	anphuocda2410@gmail.com			2024-10-20			MEMBER		<input type="button" value="Select Duration"/>
	vinh	sandilove59@gmail.com	01234567822	TPHCM	2024-10-20	2024-10-22	zzzz	MEMBER		<input type="button" value="Select Duration"/>
	User	vinhshop	shopvinh		2024-10-20			SHOP		<input type="button" value="Select Duration"/>
	User	thoai	nvmthoai12@gmail.com		2024-10-20			MEMBER		<input type="button" value="Select Duration"/>
	User	thoai12	nvmthoai1473837@gmail.com		2024-10-21			MEMBER		<input type="button" value="Select Duration"/>
	An	An Hoang Q.	anhoang800140@gmail.com		2024-10-21			MEMBER		<input type="button" value="Select Duration"/>
	User	thoaishop	thoaishop@kincare.com.vn		2024-10-21			SHOP		<input type="button" value="Select Duration"/>

2 rows selected

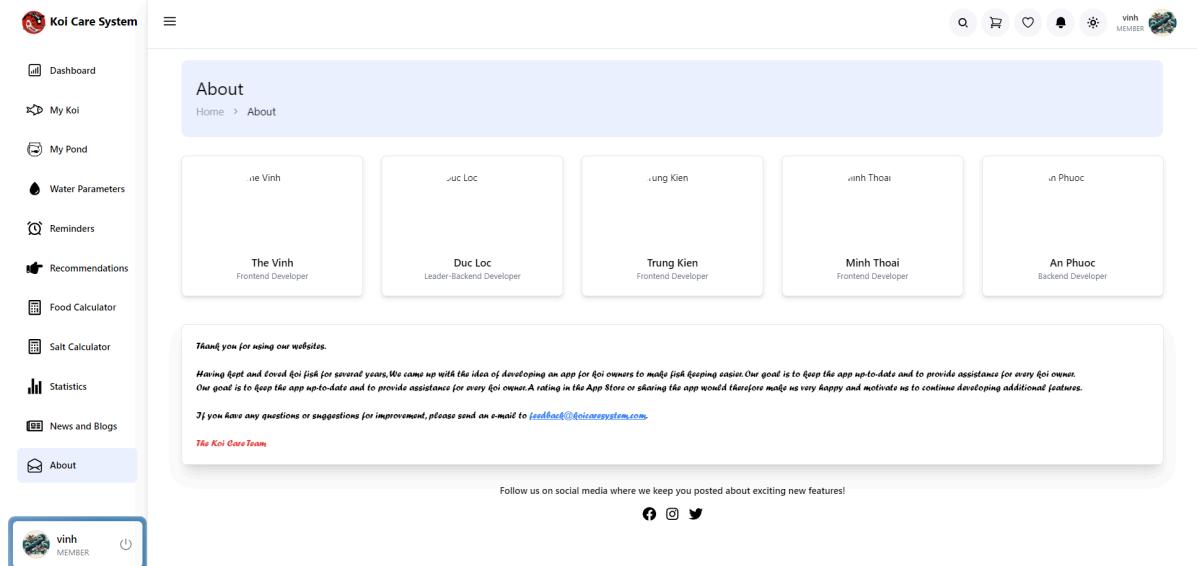
Rows per page: 100 < 1-9 of 9 >

- **Function Details:**

- **Data:** Member ID, premium status
- **Validation:** Ensure member ID exists.
- **Business Logic:**
 - **Normal Cases:** The account is upgraded to premium.
 - **Abnormal Cases:** Error message in case of failure.

FR-69: About

- **Function Trigger:** Triggered when a user navigates to the "About" section.
- **Actors/Roles:** Member
- **Purpose:** To provide information about the development team and the system.
- **Interface:** About Page
- **Data Processing:**
 - **Inputs:** None
 - **Outputs:** Information about the development team is displayed.
- **Process:**
 - The system fetches and displays details about the development team.
- **Screen Layouts:**



- **Function Details:**

- **Data:** Development team details (names, roles)
- **Validation:** None required.
- **Business Logic:**
 - **Normal Cases:** The development team information is displayed successfully.
 - **Abnormal Cases:** None expected.

V. Non-Functional Requirements

1. External Interface Requirements

1.1 User Interfaces

UI-1: The Koi Care System screens shall align with the company's existing GUI standards to maintain uniformity across all software products.

UI-2: Web Pages shall be fully navigable using the keyboard alone, as well as mouse and keyboard combinations.

UI-3: Webpages of the system shall have a homepage that provides guidance and an introduction on using its specific features.

UI-4: The system shall display an error message when a user inputs invalid koi data, pond data (e.g., age, length, weight, volume etc.).

UI-5: Common buttons or icons like “Create”, “Cancel”, “Delete,” and “Update” shall be present on applicable screens.

UI-6: The system shall have a user-friendly interface for managing koi growths, remarks, ponds, water parameters.

UI-7: The system shall display koi statistics including age, weight, and length.

UI-8: The system shall display koi pond statistics with environmental data (e.g., water temperature, pH levels).

UI-9: The system shall have a user-friendly interface for a food calculator and a salt calculator to calculate the recommended amount of food and salt.

UI-10: The system shall allow for product recommendations, including suggesting products for koi or koi ponds that users may need to purchase.

UI-11: The system shall have a dashboard for generating general information and statistics on kois, water parameters, and performances.

UI-12: The system shall provide interfaces for shops to manage all users, shops, news, orders and payments.

UI-13: The system shall provide interfaces for admins to manage all users, shops, news, orders and payments.

1.2 Software Interfaces

SI-1: Koi Data Management System

SI-1.1: The system shall store and manage data about each koi, including growth history, remarks.

SI-1.2: The system shall allow the addition, modification, and removal of koi data.

SI-2: Pond Data Management System

SI-2.1: The system shall store and manage data about each pond, including pond issues, pond assignments and pond log.

SI-2.2: The system shall allow the addition, modification, and removal of pond data.

SI-3: Water Parameters Monitoring System

SI-2.1: The system shall allow users to input water quality data (pH, temperature, etc.).

SI-2.2: The system shall validate the sensor data against predefined safe thresholds and alert users if they are exceeded.

SI-4: Food Calculator System

SI-4.1: The system shall connect to internal APIs to retrieve the amount and total weight of koi to calculate the amount of food needed.

SI-4.2: The system shall allow users to input desired growth and water parameters to recommend the amount of food and the number of feedings per day.

SI-5: Salt Calculator System

SI-5.1: The system shall integrate with koi pond databases to provide guidelines for salt intake.

SI-5.2: The system shall allow users to input the current concentration, desired concentration, and water change to customize salt recommendations.

SI-6: Product Recommendation System

SI-6.1: The system shall retrieve product details from the shop's database for recommendations.

SI-6.2: The system shall track user interactions to offer dynamic product recommendations based on pond issues.

SI-7: Statistic Koi Growth and Pond Water System

SI-7.1: The system shall track and manage koi growth statistics, including length, weight by time with standard length and weight displayed in area charts.

SI-7.2: The system shall monitor and display pond water parameters (e.g., pH levels, temperature, oxygen levels) with trend data over time.

SI-7.3: The system shall allow users to view koi growth and pond water statistics side-by-side for better analysis of environmental impacts.

SI-8: News & Blogs System

SI-8.1: The system shall allow authorized shop to create, modify, and delete news and blogs related to koi care.

SI-8.2: The system shall display news and blog content for users based on the newest and tags.

SI-9: Dashboard for Members

SI-9.1: The system shall provide members with a personalized dashboard to view koi care schedules, pond statistics, and payment details.

SI-9.2: The system shall allow members to track amounts of koi, pond, water parameter by date.

SI-10: Reporting and Analytics System for Shop and Admin

SI-9.1: The system shall allow shops to generate reports on product sales, including trends and performance by product categories.

SI-9.2: The system shall provide admins with analytics on user details, blogs, products, order and payments.

1.3 Hardware Interfaces

HI-1: Server Configuration

Application Server

CPU: Intel Xeon 8 Core 2.40 GHz

Memory: 128 GB RAM 1333 MHz

Storage: 2 TB

OS: MS Windows Server 2012 R2

Software: Microsoft IIS 8.x, .NET Framework 4.5

Database Server

CPU: Intel Xeon 12 Core 2.40 GHz

Memory: 256 GB RAM 1333 MHz

Storage: 3 TB

OS: MS Windows Server 2012 R2

Software: Microsoft SQL Server 2000 ////

1.4 Communications Interfaces

CI-1: The system shall send email or text message notifications to users for login users, forgot password.

CI-2: The system shall communicate secure data via encrypted channels (TLS) to maintain confidentiality of customer and koi data.

CI-3: The system shall support the electronic transmission of product transaction records via email.

2. Quality Attributes

2.1 Usability

USE-1: Users shall be able to retrieve koi data or transactions within three clicks.

USE-2: 90% of new users shall complete koi care tasks (e.g., adding koi, assigning ponds) on their first attempt without errors.

2.2 Performance

PER-1: The system shall support 100 concurrent users during peak hours (12:00 PM to 1:00 PM local time), with average session durations of 10 minutes.

PER-2: 95% of web pages shall load within 2 seconds on a 20 Mbps connection.

PER-3: The system shall process koi updates within 2 seconds.

2.3 Security

SEC-1: The system shall implement role-based access control, restricting access to sensitive operations like health data updates and transaction approvals.

SEC-2: All koi health and transaction data shall be encrypted both at rest and in transit using industry-standard encryption methods (e.g., AES-256, TLS).

2.4 Safety

SAF-1: The system shall include robust error-handling and logging mechanisms to capture and report koi health issues or transaction failures.

SAF-2: Regular security audits shall be conducted to protect the system from potential vulnerabilities.

2.5 Availability

AVL-1: The Koi Care System shall maintain a 98% uptime during business hours (5:00 AM to 12:00 AM) and 90% availability during non-business hours, excluding maintenance.

AVL-2: Downtime excluded from availability calculations shall consist of maintenance scheduled during the hours from 6:00 P.M. Sunday Pacific Time through 3:00 A.M. Monday Pacific Time.

2.6 Reliability

REL-1: No more than 5 experimental runs out of 1,000 shall be lost due to software errors.

REL-2: Critical system components like the koi health module shall have a mean time between failures (MTBF) of at least 90 days.

2.7 Design Constraints

DES-01: Programming Languages: Java, Javascript

DES-02: Programming Languages Version: Java 21

DES-03: Programming Tools: VS Code, IntelliJ IDEA, Postman, Microsoft SQL Server Management Studio 19

DES-04: Database: Microsoft SQL Server 2018 Express Edition

DES-05: Web Browser: Chrome

DES-06: Report Tools: Microsoft Word, Google Docs, Excel

DES-07: Diagram Design Tools: Draw.IO, Visual Paradigm

DES-08: Design Tools: Figma

DES-09: Web Server: Apache Tomcat (embedded with Spring Boot)

DES-10: Backend Framework: Spring Boot 3.x

DES-11: Frontend Framework: React.js 18.x

DES-12: CSS Framework: Tailwind CSS