

**Koi Farm Shop**

**Software Requirement Specification**

Project Code: <Code of the project>

Document Code: <Code of the document >– v<x.x>

Class Code: SE1878

Group Code: 6

**<FPTU HCM, 30/10/2024>**

**SIGNATURE PAGE**

**ORIGINATOR:** <Name> <Date>

<Position>

**REVIEWERS:** <Name> <Date>

<Position>

<Name, if it’s needed> <Date>

<Position>

**APPROVAL:** <Name> <Date>

<Position>

**TABLE OF CONTENTS**

[**1**](#_1fob9te) **Introduction 5**

[1.1](#_3znysh7) Purpose 5

[1.2](#_2et92p0) Scope 5

[1.3](#_tyjcwt) Definitions, Acronyms, and Abbreviations 5

[1.4](#_3dy6vkm) References 5

[1.5](#_1t3h5sf) Overview 6

[**2**](#_4d34og8) **Overall Description 7**

[2.1](#_2s8eyo1) Product Perspective 7

[2.2](#_17dp8vu) User classes and characteristics 7

[2.3](#_3rdcrjn) Design and implementation constraints 7

[**3**](#_26in1rg) **FUNCTIONAL Requirements 8**

[3.1](#_lnxbz9) Swimlane Diagrams 8

[3.2](#_35nkun2) Use Case Diagrams 8

[3.3.1.](#_1ksv4uv) < Use Case Name 1> 8

[3.3.2.](#_44sinio) < Use Case Name 2> 9

[3.3](#_2jxsxqh) State Diagrams 9

[3.4](#_z337ya) Logical Data Model 9

[3.5](#_3j2qqm3) Wire Flows 9

[**4**](#_1y810tw) **NON-FUNCTIONAL Requirements 10**

[4.1](#_4i7ojhp) Usability 10

[4.2](#_2xcytpi) Reliability 10

[4.3](#_1ci93xb) Performance 11

[4.4](#_3whwml4) Supportability 11

[4.5](#_2bn6wsx) Licensing Requirements 11

[**5**](#_qsh70q) **Supporting Information 12**

# Introduction

## Purpose

The purpose of this SRS document is to fully define the functionality and requirements for the **Koi Farm Management System**, a web-based platform supporting the following key objectives:

1. **Koi Sales and Information Access**: Allow customers to browse detailed information on available Koi fish, including origin, size, breed, and other specifics, to make informed purchasing decisions.
2. **Facilitating Consignment Services**: Enable customers to consign their Koi fish to the farm for care or business purposes, including an efficient consignment workflow with options for both offline and online consignments.
3. **Update Customer Service**: Provide tools for managing promotions, gathering feedback, and maintaining customer profiles to improve customer loyalty and satisfaction.

## Scope

The **Koi Farm Management System** is designed to be a centralized platform for managing various business aspects of a Koi farm, including:

* **Customer Browsing and Fish Selection**: Offering detailed fish listings and search filters for finding specific types of Koi fish (e.g., imported purebreds, F1 crossbreeds, native Vietnamese breeds).
* **Ordering and Purchasing**: Allowing customers to place orders, with options for buying fish in bulk or individually, with appropriate certification included for individual fish purchases.
* **Consignment Services**: Providing a structured workflow for customers to consign their Koi fish to the farm, covering initial requests, inspections (if required), pricing agreements, and eventual sales, with support for both offline and online consignment options.
* **Comparative Analysis**: Enabling customers to compare different Koi fish to aid in purchasing decisions.
* **Order and Promotion Management**: Tracking customer orders from placement to fulfillment, applying relevant promotions and loyalty points.
* **Customer Profiles and History**: Maintaining detailed customer profiles and purchase history to support targeted marketing and loyalty programs.
* **Feedback and reviews**: Allow customers to contribute feedback and reviews to improve services.
* **Administrative Dashboards and Reporting**: Equipping farm management with insights into sales, inventory, customer engagement, and system performance.

### Definitions, Acronyms, and Abbreviations

* **Koi Fish**: A type of ornamental fish that is specifically bred for its unique colors and patterns, often considered valuable in Asian culture.
* **F1 Crossbreed**: The first-generation offspring of a crossbred Koi fish, often bred for specific traits.

## References

[OnKoi.vn](https://onkoi.vn) – Reference website for similar Koi farm management functionalities.

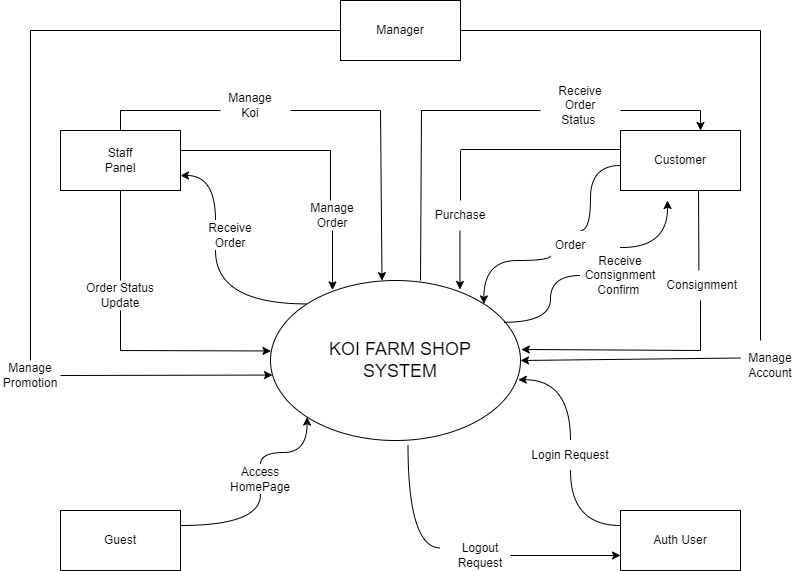
## Overview

This Software Requirements Specification (SRS) document is organized to present a complete, detailed breakdown of the **Koi Farm Management System**, including all functional and non-functional requirements. The document is structured to guide the development and validation processes, ensuring that the system meets both the needs of farm management and customer expectations.

Each section of the SRS serves a specific purpose:

1. **Introduction**: Provides a high-level overview of the system's goals, purpose, and scope. This section also includes key definitions, acronyms, and references that will aid in understanding the system requirements.
2. **Overall Description**: Discusses the general factors that impact the system, including a summary of the product’s purpose, the types of users it serves, and design or operational constraints. It provides a background for the detailed requirements in later sections.
3. **System Requirements**: Divided into **Functional Requirements** and **Non-Functional Requirements**:
   * **Functional Requirements**: Define the specific features and operations of the Koi Farm Management System, such as customer browsing, purchasing, consignment, and feedback.
   * **Non-Functional Requirements**: Outline additional constraints such as system performance, security, scalability, and usability, which the system must fulfill.
4. **Supporting Information**: Contains supplementary details like diagrams, mockups, data flow models, and an appendix that may include future considerations or additional resources.

# 2.Overall Description

**

## 2.1 Product Perspective

*The Koi fish farm management system is a new, web-based platform designed to streamline and automate various processes involved in Koi fish sales, inventory management, customer interactions, and care services. Unlike traditional in-person or fragmented online sales processes, this system centralizes all farm operations related to Koi fish.*

*The platform serves as a digital hub where customers can browse Koi fish listings, compare fish based on specific attributes (e.g., origin, size, breed), and make purchases directly from the farm. Additionally, it supports Koi care services, allowing customers to consign fish for care or resale.*

*The* ***system’s context*** *involves integration with external components:*

* ***Customers*** *who can view, purchase, and consign Koi fish.*
* ***Farm Administrators*** *who manage fish listings, track inventory, and oversee customer orders and care services.*
* ***Customer Support Representatives*** *who assist users with inquiries.*
* ***Payment Gateway*** *for handling online transactions securely.*
* ***Messaging/Email Service*** *for customer notifications about order status, care updates, and promotional offers.*
* ***Authentication Services*** *(JWT or OAuth) for secure login and user management.*

*This platform transforms the farm’s operations by centralizing information, improving customer accessibility, and automating workflows, making the management of Koi fish sales and care efficient and user-friendly.*

## 2.2 User classes and characteristics

The main users of this system are:

1. **Farm Administrators**:
   * Handle inventory, product listings, customer orders, and consignment requests.
   * Need full access to manage all aspects of the system, including dashboards and detailed reporting tools.
2. **Customers**:
   * Access the site to explore Koi fish, compare breeds, make purchases, or engage with the farm’s consignment services.
   * Require an intuitive interface to easily search, filter, and manage orders, with options for account creation to track purchase history and consignment status.
3. **Customer Support Representatives**:
   * Address customer questions, resolve order issues, and provide assistance with Koi care inquiries.
   * Need limited access to order details and customer information to provide effective support without access to full administrative tools.

## 2.3 Design and implementation constraints

1. **Data Privacy and Security**:

* The system must comply with relevant data protection standards (e.g., GDPR) to secure customer information, including secure storage and transmission of sensitive data.

1. **Scalability and Reliability**:

* The platform should be scalable to handle growth in user activity, inventory, and data, ensuring smooth performance even with high user traffic or an expanding database.

1. **Cross-Platform Compatibility**:

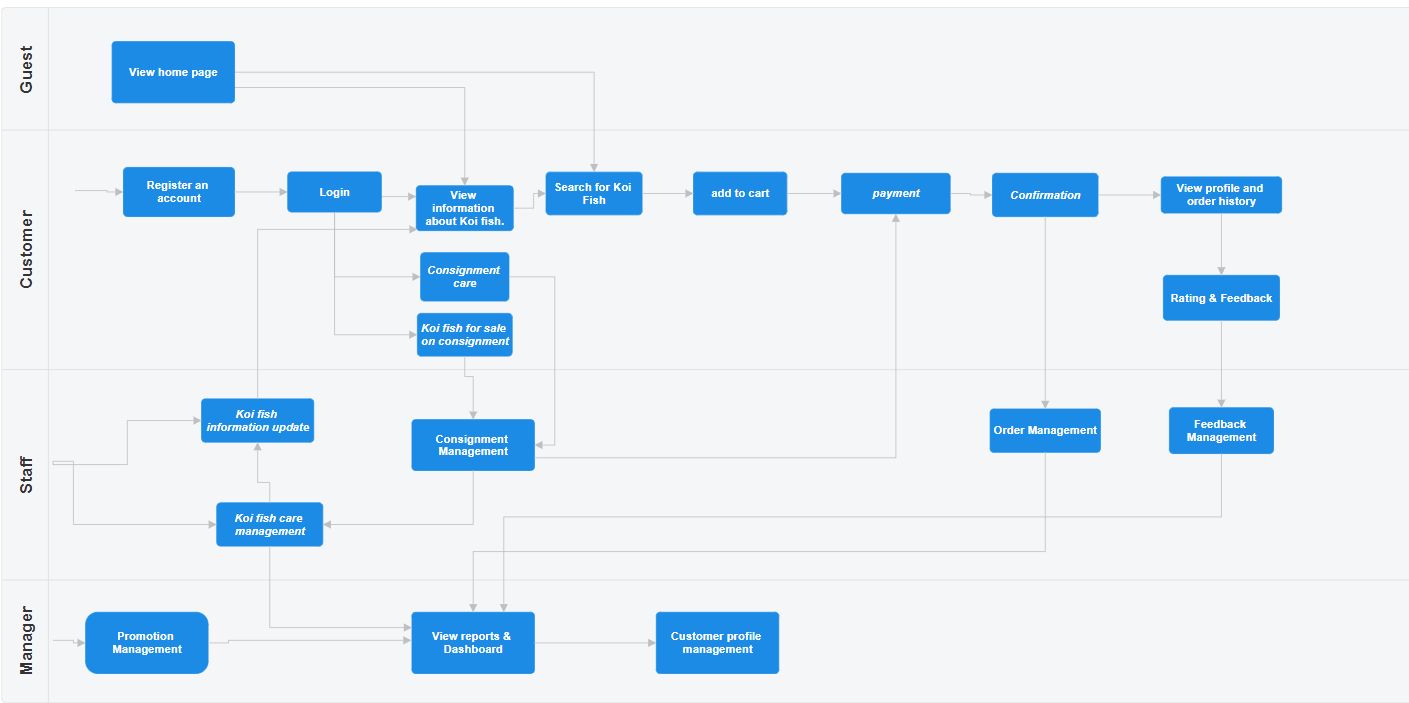
* Designed to be fully compatible across various devices (desktop, mobile, tablet) and web browsers to accommodate a broad user base.

1. **User-Friendly Interface**:

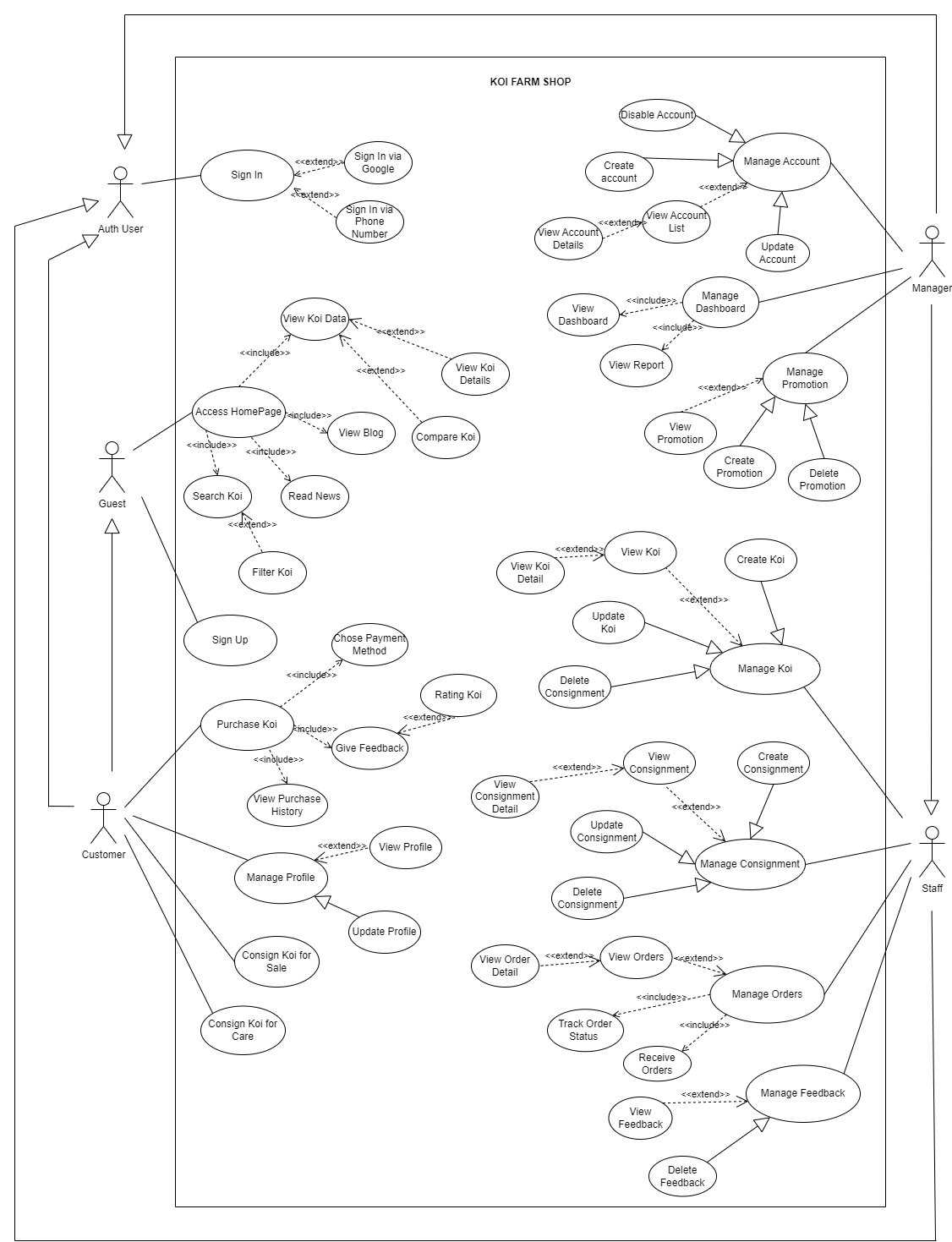
* To meet diverse customer needs, the interface should prioritize usability, with a simple and responsive design.

# FUNCTIONAL Requirements

## Swimlane Diagrams

**

## Use Case Diagrams

**

## View Koi Details

| USE CASE-n SPECIFICATION | | | | |
| --- | --- | --- | --- | --- |
| Use-case No. | <UC001> | Use-case Version | | <1.0> |
| Use-case Name | View Koi Details | | | |
| Author | Tiến | | | |
| Date | 25/10/2024 | Priority | High | |
| Actor:  Guest, Customer  Summary:  This use case allows the actor to view detailed information about a selected Koi fish, including attributes such as size, color, breed, and price, along with available images or videos.  Goal:  To enable the actor to make informed decisions by providing them with comprehensive details about the Koi fish.  Triggers  The actor selects a specific Koi fish from the search results or filtered list to view more information.  Preconditions:   * The actor has access to the Koi Farm Shop application. * The actor is on the page displaying the list of available Koi fish. * The selected Koi fish is available in the system's database.   Post Conditions:   * The actor successfully views the detailed information of the selected Koi fish. * The system logs the actor’s view action for potential recommendations or analysis.   Main Success Scenario:   1. The actor navigates to the Koi fish listing page. 2. The actor selects a specific Koi fish they are interested in. 3. The system retrieves and displays the detailed information for the selected Koi fish. 4. The actor reviews the information, including images, videos, breed, size, color, and price. 5. The actor decides whether to proceed with further actions, like comparing the fish or adding it to the cart.   Alternative Scenario:   * **Step 3a**: If the actor is not signed in, the system prompts them to log in to view additional details or interact with the product. * **Step 4a**: If the Koi fish details are temporarily unavailable, the system displays a message and suggests similar options.   Exceptions:   * **E1**: The selected Koi fish has been recently removed or is unavailable. The system notifies the actor and suggests alternatives. * **E2**: System error while loading details. The actor receives an error message and is encouraged to try again later.   Relationships:   * Related to "Search Koi" use case. * Related to "Compare Koi" use case.   Business Rules:   * The information displayed should comply with accuracy and quality standards as per the business's policies. * Only authorized customers can view pricing information if the business policy restricts it for guests. | | | | |

## Purchase Koi

| USE CASE-n SPECIFICATION | | | | |
| --- | --- | --- | --- | --- |
| Use-case No. | <UC002> | Use-case Version | | <1.0> |
| Use-case Name | Purchase Koi | | | |
| Author | Tiến | | | |
| Date | 25/10/2024 | Priority | High | |
| Actor:  Customer  Summary:  This use case allows the customer to purchase a selected Koi fish by adding it to their cart, selecting a payment method, and confirming the purchase.  Goal:  To enable the customer to successfully purchase a Koi fish and complete the transaction.  Triggers  The customer selects a Koi fish they wish to purchase.  Preconditions:   * The customer is signed in to the Koi Farm Shop system. * The selected Koi fish is available for sale. * The customer has a valid payment method set up.   Post Conditions:   * The system records the completed transaction. * The customer receives a purchase confirmation. * The Koi fish stock is updated.   Main Success Scenario:   1. The customer selects a Koi fish to purchase. 2. The system displays the selected Koi fish details and adds it to the shopping cart. 3. The customer proceeds to the checkout page and selects a payment method. 4. The customer confirms the purchase. 5. The system processes the payment and completes the transaction. 6. The customer receives a purchase confirmation message, and the order appears in their purchase history.   Alternative Scenario:   * **Step 3a**: If the customer has no saved payment methods, they are prompted to add one before proceeding. * **Step 5a**: If the payment fails, the system notifies the customer and allows them to try again with a different method.   Exceptions:   * **E1**: Selected Koi fish is no longer available; the system notifies the customer and removes the fish from their cart. * **E2**: System error during payment processing; the customer is notified and can retry the transaction.   Relationships:   * Related to "View Koi Details" use case. * Related to "Manage Profile" use case (for payment information).   Business Rules:   * Only customers who are signed in can make purchases. * Payment processing must comply with secure transaction standards. | | | | |

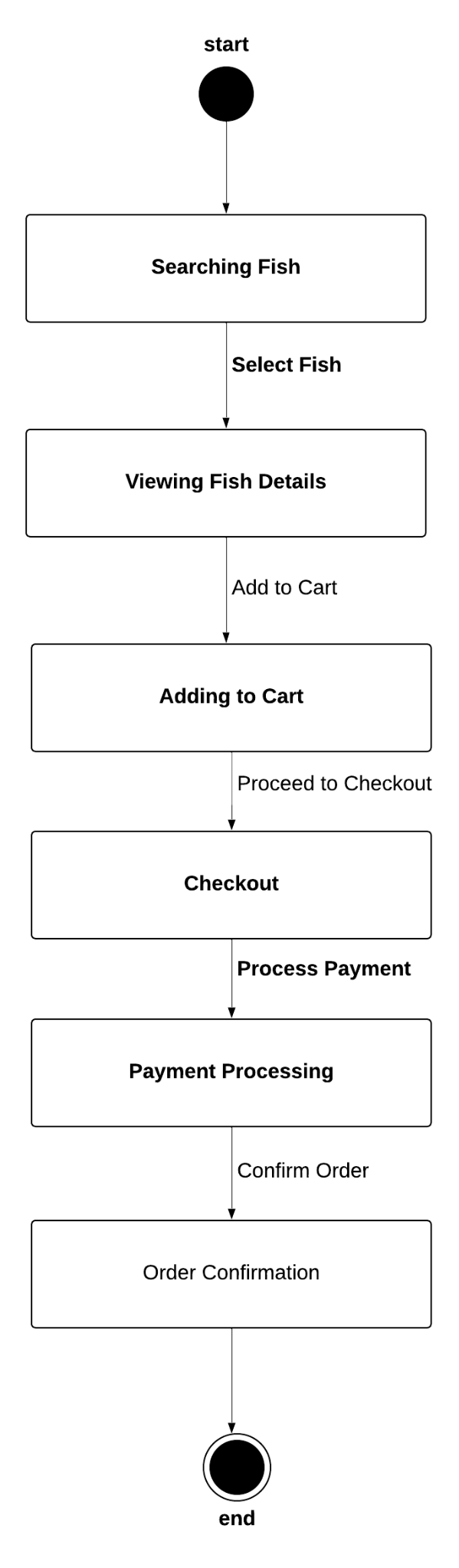
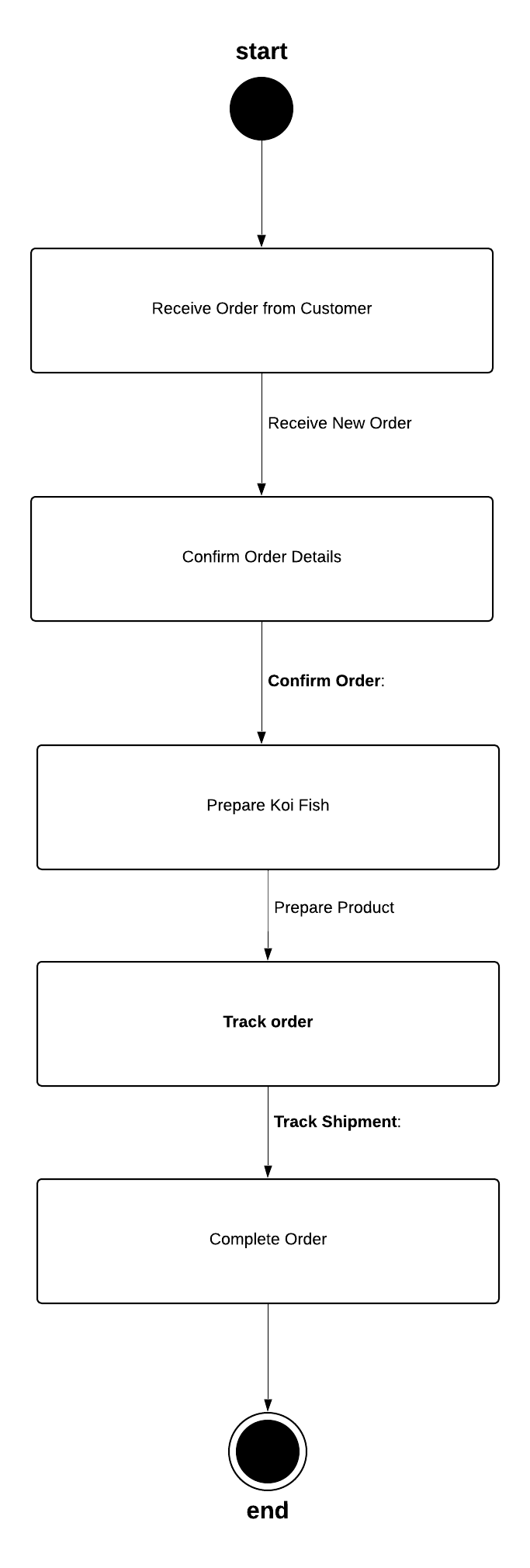
## Consign Koi for Sale

| USE CASE-n SPECIFICATION | | | | |
| --- | --- | --- | --- | --- |
| Use-case No. | <UC003> | Use-case Version | | <1.0> |
| Use-case Name | Consign Koi for Sale | | | |
| Author | Tiến | | | |
| Date | 25/10/2024 | Priority | Normal | |
| Actor:  Customer  Summary:  This use case allows a customer to consign their Koi fish for sale on the Koi Farm Shop platform, enabling other customers to purchase it.  Goal:  To facilitate customers in selling their Koi fish through the platform.  Triggers  The customer decides to list their Koi fish for sale.  Preconditions:   * The customer is signed in to the Koi Farm Shop system. * The customer has Koi fish that meet the requirements for consignment (e.g., health status).   Post Conditions:   * The Koi fish is successfully listed for sale. * The consignment details are visible to other customers.   Main Success Scenario:   1. The customer selects the option to consign their Koi for sale. 2. The system prompts the customer to enter details about the Koi fish (e.g., breed, size, age). 3. The customer submits the consignment information. 4. The system verifies the details and lists the Koi fish for sale on the platform.   Alternative Scenario:   * **Step 3a:** If the consignment details do not meet platform requirements, the system notifies the customer to adjust the information.   Exceptions:   * **E1**: If the system cannot verify the Koi's health status, the listing is temporarily halted, and the customer is notified.   Relationships:   * Related to "Manage Consignment" use case. * Related to "View Koi Details" use case.   Business Rules:   * All consigned Koi must meet health and quality standards set by the platform. * Customers are responsible for accurate information regarding the consignment. | | | | |

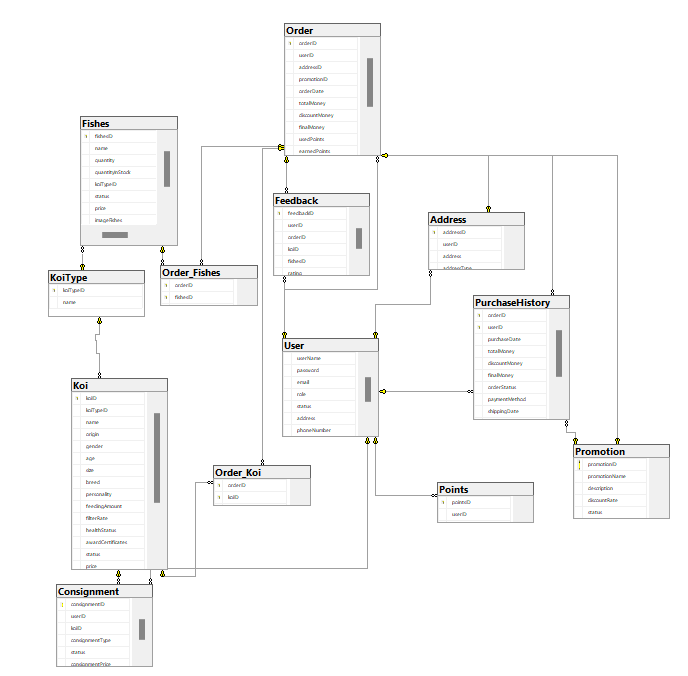
## Manage Consignment

| USE CASE-n SPECIFICATION | | | | |
| --- | --- | --- | --- | --- |
| Use-case No. | <UC004> | Use-case Version | | <1.0> |
| Use-case Name | Manage Consignment | | | |
| Author | Tiến | | | |
| Date | 25/10/2024 | Priority | High | |
| Actor:   * **Staff**: Responsible for creating, updating, and managing consignment records for customers who want to sell or care for their Koi fish at the farm.   Summary:   * The Manage Consignment use case allows Staff to handle consignments of Koi fish brought in by customers either for sale or care. Staff can create new consignment records, update details, and delete consignments as needed. This ensures a smooth process for managing Koi fish that are placed under the farm's care or listed for sale.   Goal:   * To efficiently handle consignment requests by allowing Staff to manage all necessary details and provide a seamless experience for customers who wish to consign their Koi fish.   Triggers  The use case is triggered when a customer submits a consignment request, either for selling their Koi fish through the farm or for temporary care services.  Preconditions:   * **Staff** is logged in to the system with appropriate permissions. * The Koi fish details and the customer’s profile are registered in the system. * The consignment form (specifying sale or care options) has been submitted by the customer.   Post Conditions:   * A new consignment record is created, updated, or deleted. * The consignment status is reflected in the customer’s profile. * Inventory and status information for Koi fish consignments are updated in the system.  Main Success Scenario:  1. **Initiate Consignment Management**: Staff accesses the consignment management section from the system dashboard. 2. **Create New Consignment**:    * Staff clicks on "Create Consignment."    * The system presents a form where Staff enters details (e.g., customer information, Koi fish details, consignment type: sale or care).    * Staff confirms the consignment details, and the system saves them. 3. **Update Existing Consignment**:    * Staff selects an existing consignment from the list and clicks "Edit."    * Staff modifies consignment details (e.g., price, care duration, etc.).    * Staff saves the changes, and the system updates the consignment record. 4. **Delete Consignment**:    * Staff selects a consignment and clicks "Delete."    * The system prompts for confirmation, and Staff confirms.    * The consignment record is deleted from the system. 5. **View Consignment Details**: Staff can view all consignment records, filtering them by consignment type (sale or care) or customer.   Alternative Scenario:  **Step 2a (Incomplete Consignment Form):**   * Condition: Staff submits a consignment form with missing required fields. * System Response: The system displays an error message prompting Staff to complete all mandatory fields. * User Action: Staff fills in the missing information and resubmits.   **Step 3a (Edit Restrictions):**   * Condition: Staff attempts to update a consignment that is marked as "Completed" or "Sold." * System Response: The system prevents editing and shows a notification that completed consignments cannot be modified. * User Action: Staff acknowledges and exits the edit screen.   Exceptions:  **E1 (Insufficient Permissions)**:   * **Condition**: A user without the required permissions tries to access the consignment management feature. * **System Response**: The system restricts access and displays an error message indicating insufficient permissions. * **User Action**: The user can log in with a higher-privilege account or request access.   **E2 (Network Error)**:   * **Condition**: Network connectivity is lost during consignment creation or update. * **System Response**: The system saves data locally (if possible) and displays a message to retry once connected. * **User Action**: Staff waits for reconnection and retries the operation.   **E3 (Data Conflict)**:   * **Condition**: Staff attempts to delete or edit a consignment currently being updated by another Staff member. * **System Response**: The system displays a conflict message and reloads the latest data. * **User Action**: Staff reviews the updated consignment details before attempting any further changes.   Relationships:   * **"View Consignment"**: Staff can access the consignment list and view details. * **"Update Profile"**: This may be linked if customer profiles are updated based on consignment interactions.   Business Rules:   * **Consignment Completion**: Consignments marked as "Completed" cannot be edited. Only active consignments can be modified. * **Authorization Requirements**: Only authorized Staff members can manage consignments. Consignment access is restricted based on role permissions. * **Consignment Expiry**: Consignment records for "Care" type are automatically marked as "Completed" after a predefined duration. * **Duplicate Prevention**: The system should prevent duplicate consignments for the same Koi fish by the same customer within a specific period. | | | | |

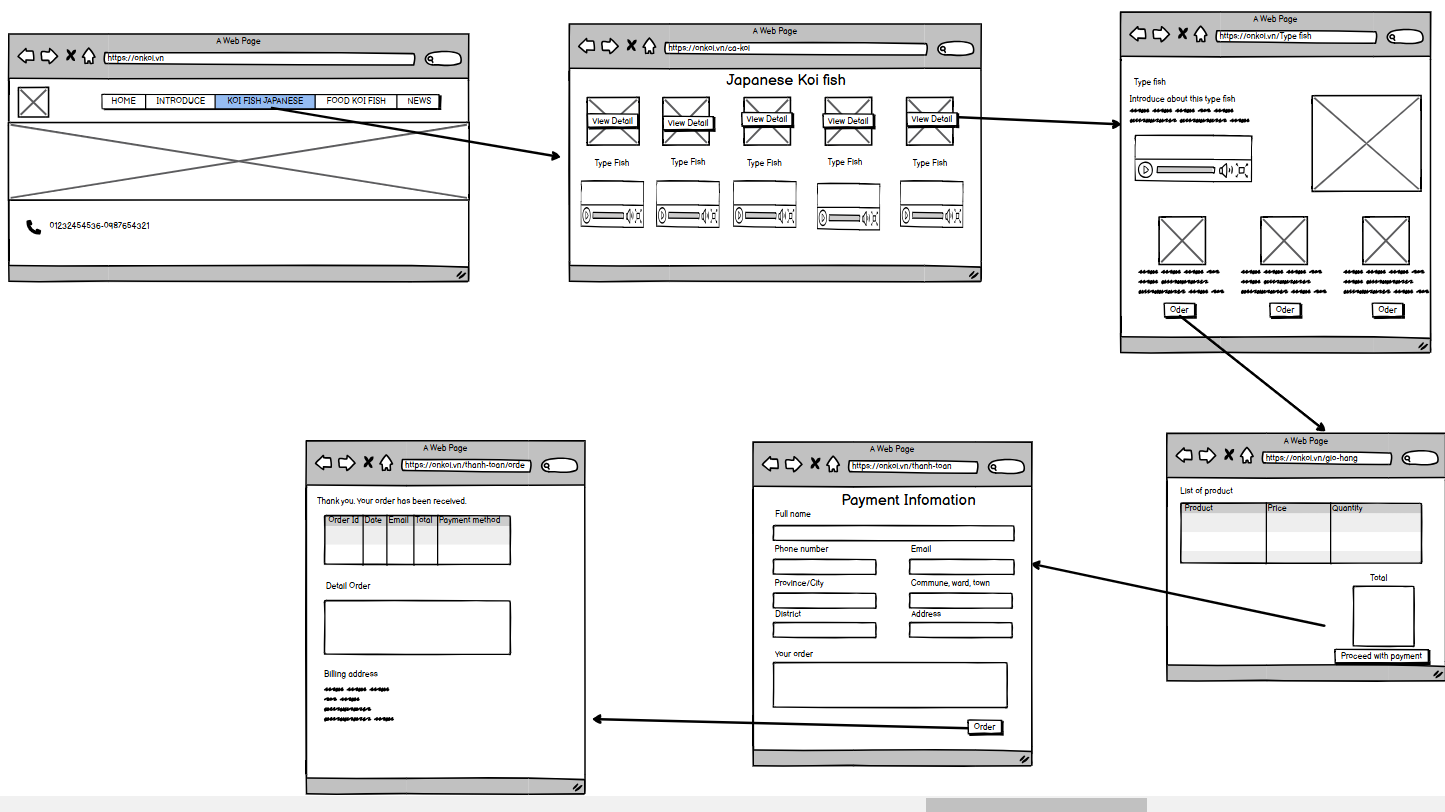
## State Diagrams

* *

## Logical Data Model



## Wire Flows

**

# NON-FUNCTIONAL Requirements

## Usability

* ***Training duration****: The system needs to ensure that regular users can get familiar with and use the basic functions (searching for and purchasing Koi fish) within 2 hours and professional users within 1 hour.*
* ***Usability****: The system interface must be user-friendly, easy to use, and suitable for non-tech-savvy users. Operations like searching, comparing Koi fish, or placing orders must be easily accomplished with only 3-4 mouse clicks.*
* ***UI/UX standards****: The system must comply with common UI/UX standards, such as Google's Material Design or Apple's Human Interface Guidelines, ensuring a visually appealing and consistent interface across devices (PC, laptop, tablet, mobile phone).*

### <Usability Requirement One>

*Easy navigation with clear labeling and tooltips for first-time and regular users.*

## Reliability

* ***Availability****: The system needs to ensure an availability of up to 99.9% throughout the operating period, except for periodic maintenance (no more than 4 hours per month).*
* ***The mean time between failures (MTBF)****: Must be at least 1 year to maintain continuous operation and avoid disrupting the buying, selling, and consignment of Koi fish.*
* ***Mean Time To Repair (MTTR)****: When the system encounters an issue, the recovery and resolution time must be within 2 hours to avoid interrupting the business operations and Koi fish care.*
* ***The software defect rate:*** *must not exceed 0.5 defects per 1,000 lines of code (KLOC) for critical errors that can lead to data loss or impact the system's core functions (such as buying, selling, and order management).*

### <Reliability Requirement One>

*The system must guarantee 99.9% availability and automatically back up data every 24 hours.*

## Performance

* ***Response time****: The core transactions (searching for Koi fish, viewing details, placing orders) must have a response time of no more than 2 seconds for each operation.*
* ***Throughput****: The system must be able to handle at least 1000 transactions per hour without reducing performance. This includes customer searches and orders, as well as updates to the fish consignment process.*
* ***Scalability****: The system must be scalable to handle a maximum of 5000 concurrent customers without compromising performance, especially during promotional periods or special events.*
* ***Resource utilization****: The system must optimize the use of memory and CPU, ensuring that processing large requests (e.g., comparing multiple Koi) does not consume excessive resources, keeping CPU usage below 70% and memory below 80%.*

### <Performance Requirement One>

*The system must process transactions within 2 seconds and handle a minimum of 5,000 requests per hour.*

## Supportability

* ***Source Code Standards****: The system must adhere to standard coding conventions as outlined in documents like PSR-12 (PHP Standard Recommendations) or similar coding standards to ensure maintainability and easy upgrades.*
* ***Maintainability****: The codebase should be designed to be clear and easy to understand, allowing the development team to easily maintain and expand the system. Maintenance tools like Git, Jira, and SonarQube will be used to track and manage any issues that arise.*
* ***Handling and Addressing Errors****: The system needs to integrate a feature to track and record errors, with the ability to instantly notify the development team via email or SMS when issues arise.*

### <Supportability Requirement One>

*The system must be easy to maintain and upgrade and have the capability to integrate performance monitoring tools.*

## Licensing Requirements

* ***Usage license****: The system must comply with software copyright requirements, including the open-source libraries used, the development tools, and third-party software. All these tools and software must have licenses that allow commercial use.*
* ***Usage restrictions****: The system only allows users to log in and use the number of accounts and permissions that have been granted. Accounts that violate or are invalid will be automatically locked after 5 failed login attempts.*

# Supporting Information

***Please refer to the following documents:***

1. Functional Requirements Document.

2. Use case analysis.

3. System Architecture Document.

4. User Interface (UI) mockups.

5. Testing Plan.

6. Implementation Plan.