



SYSTEM

REQUIREMENT SPECIFICATION

Prepared By :

- Ahmed Mohamed Ghareb
- Ziad Nagib El Ghazaly
- Abdelrahman Shrief
- Moamen Mohamed
- Nour Eldin Wael

Contents

Section 1.Overview

- 1.1 Purpose
- 1.2 Audience
- 1.2 Project Scope

Section 2.Requirements

- 2.1 System Req
- 2.2 Functional Req
- 2.3 Non Functional Req

Section 3.Diagrams

- 3.1 Use Case
- 3.2 Context
- 3.3 Activity
- 3.4 Class
- 3.5 Component
- 3.6 Sequence

Section 1.Overview

1.1 Purpose

The purpose of this project is to design and develop a comprehensive e-commerce platform specifically tailored for Sports clothing sales. The platform will provide a seamless shopping experience for users, enabling them to browse, search, and purchase clothing items with ease. It will also offer administrative tools for managing products, orders, inventory, and customer interactions. The goal is to create an efficient, scalable, and user-friendly website that enhances customer satisfaction and increases sales.

1.2 Audience

Individual shoppers looking to purchase Sports clothing online, including men, women, and children of various age groups and styles.

1.3 Project Scope

The scope of the project covers the following key functionalities and features:

Frontend:

- User-friendly interface for browsing and filtering clothing items by category, brand, price, and size.
- Product pages with high-quality images, detailed descriptions, and reviews.
- Shopping cart functionality with real-time updates on item quantities and totals.
- User accounts for managing profiles, order history, wishlists, and saved addresses.

Backend:

- Inventory management to track stock levels and update them after sales.
- Order management, including order placement, status tracking, and cancellation.
- Payment processing integration for secure online transactions.
- Admin panel for managing products, discounts, categories, and user interactions.

Section 2. Requirements

2.1 System Requirements

1) Front-End Requirements:

Technology Stack:

- **HTML**
- **Styling:** CSS, Bootstrap 5.
- **Interactive:** Vanilla Java Script, Its libraries like: (Swiper JS)

Design:

- **Design System:** Define colors, typography, icons, buttons, and UI elements.
- **Responsive Design:** Ensure compatibility across devices(mobile, tablet, desktop). User
- **Experience(UX):** Easy to find information and use little clicks for navigate between pages.
- **Modern:** Colorful and Modern design compatible with the kind of system.

Performance:

- Lazy Loading: only load elements when needed to reduce load time.
- Minification: Minify CSS, JS and HTML to reduce file size.
- Caching: Implement browser caching and use optimized images to enhance load speed.

2) Back-end Requirements:

Architecture:

- **Frameworks Selection:** choose a backend framework (.NET) that best fits the project needs.
- **Database Design:** plan the database schema, including tables, relationships, and normalization for SQL database.

Authentication and Authorization:

- **User Authentication:** Implement secure user authentication to manage user access.
- **Password Management:** Encrypt the User Password to be Secure.
- **Role-Based Access Control:** Define users roles and permissions to restrict access to sensitive endpoints and features.

Database Management:

- **Migration:** Implement a migration system to manage changes in the database schema.
- **Backup&Recovery:** Establish regular backup and recovery plans for data security.

2.2 Functional Requirement

Users Requirements	Client	Admin
F u n c t i o n a l	<ul style="list-style-type: none">• Client should be able to register for an account and log in securely• Client should be able to search for products and browse through categories easily• Client should be able to view detailed information about products, including descriptions, images, prices, and availability• Client should be able to add products to their cart, adjust quantities, and remove items as needed• Client should be guided through a simple and intuitive checkout process, providing necessary information for order placement	<ul style="list-style-type: none">• Access to a centralized dashboard providing an overview of system activities, including sales analytics, order statuses, and inventory levels• Add, edit, and delete products from the catalog, including product descriptions, images, prices, and inventory levels• View and manage orders, including order processing, fulfillment, and tracking. Monitor and manage product inventory, including stock levels, restocking, and inventory adjustments• Manage user accounts, including creating new accounts, editing user profiles, and handling account permissions

F u n c t i o n a l

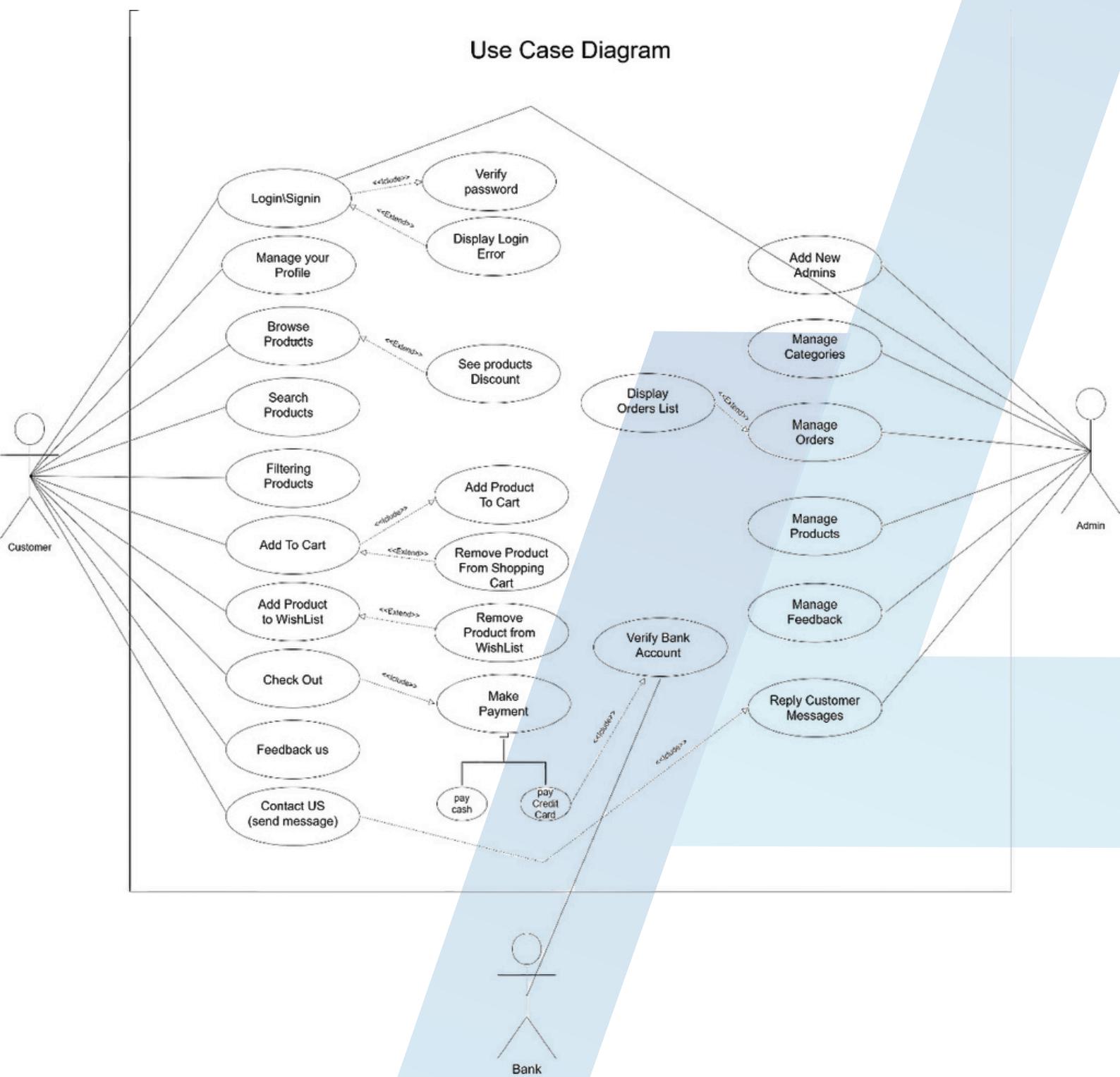
- -Client should be able to track the status of their orders, including order confirmation and shipment tracking
- Client should have access to their order history, including past purchases, order status, and tracking information for shipped orders
- Client should be able to manage their account settings, including profile information, addresses, and payment methods
- Client should have the option to create and manage wishlists or favorite items for future reference
- Client should be able to view reviews and ratings submitted by other users and submit their own reviews
- Client should have access to customer support channels for assistance with inquiries, issues, or returns
- -Create and manage promotional campaigns, discounts, and coupon codes to attract customers and drive sales
- Generate reports and analyze data related to sales performance, customer behavior, and product trends
- Manage website content, including product categories, featured products, and informational pages Provide support to customers via various channels, including email, live chat, and support tickets

2.3 Non Functional Req

Users	Client/Admin
<p>n o n F u n c t i o n a l</p>	<ul style="list-style-type: none">Security and encryption -Ensure that the client's all over data can only be altered by the admin. -Ensure that all delivered system secured and trustedIntegration with other systems -Integrate with popular payment gateways such as PayPal, Stripe, or Square to facilitate secure online payments. This allows users to pay using various methods, including credit/debit cards, digital wallets, and bank transfers. -Integrate with inventory management systems to synchronize product inventory levels, update stock availability in real-time, and automate inventory replenishment processes. This helps prevent Sockouts and ensures accurate product availability information.Performance -The system should be responsive and capable of handling high volumes of concurrent user interactions, ensuring fast loading times for product listings, search results, and checkout processes -Response times for all actions Especially critical actions, such as adding items to the cart or processing payments, should be minimal to provide a seamless user experienceMaintability -the system's ability to be modified or updated efficiently without excessive effort or cost. It focuses on how well the system can accommodate changes, corrections, and enhancements over its lifecycleReliability -the ability of a system to perform consistently and accurately over time, under specified conditions. It focuses on the system's dependability and its capacity to function without failure when required

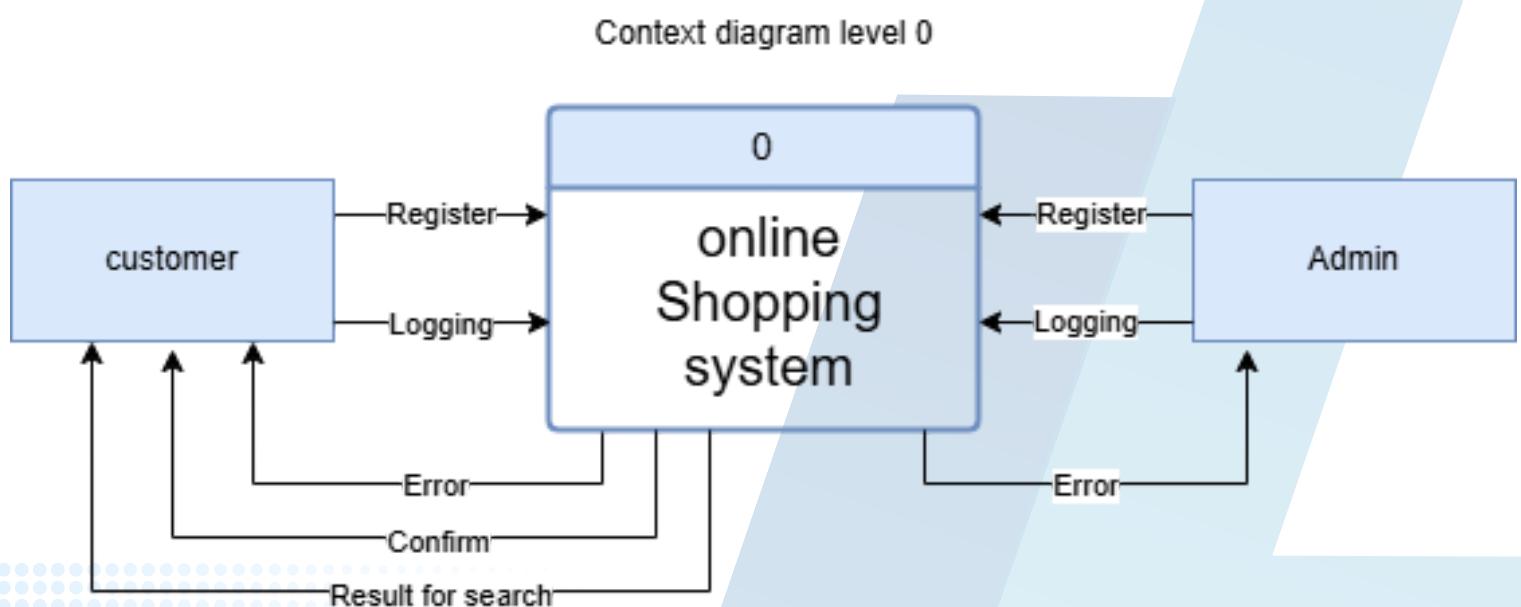
Section 3. Diagrams

3.1 Use Case Diagram

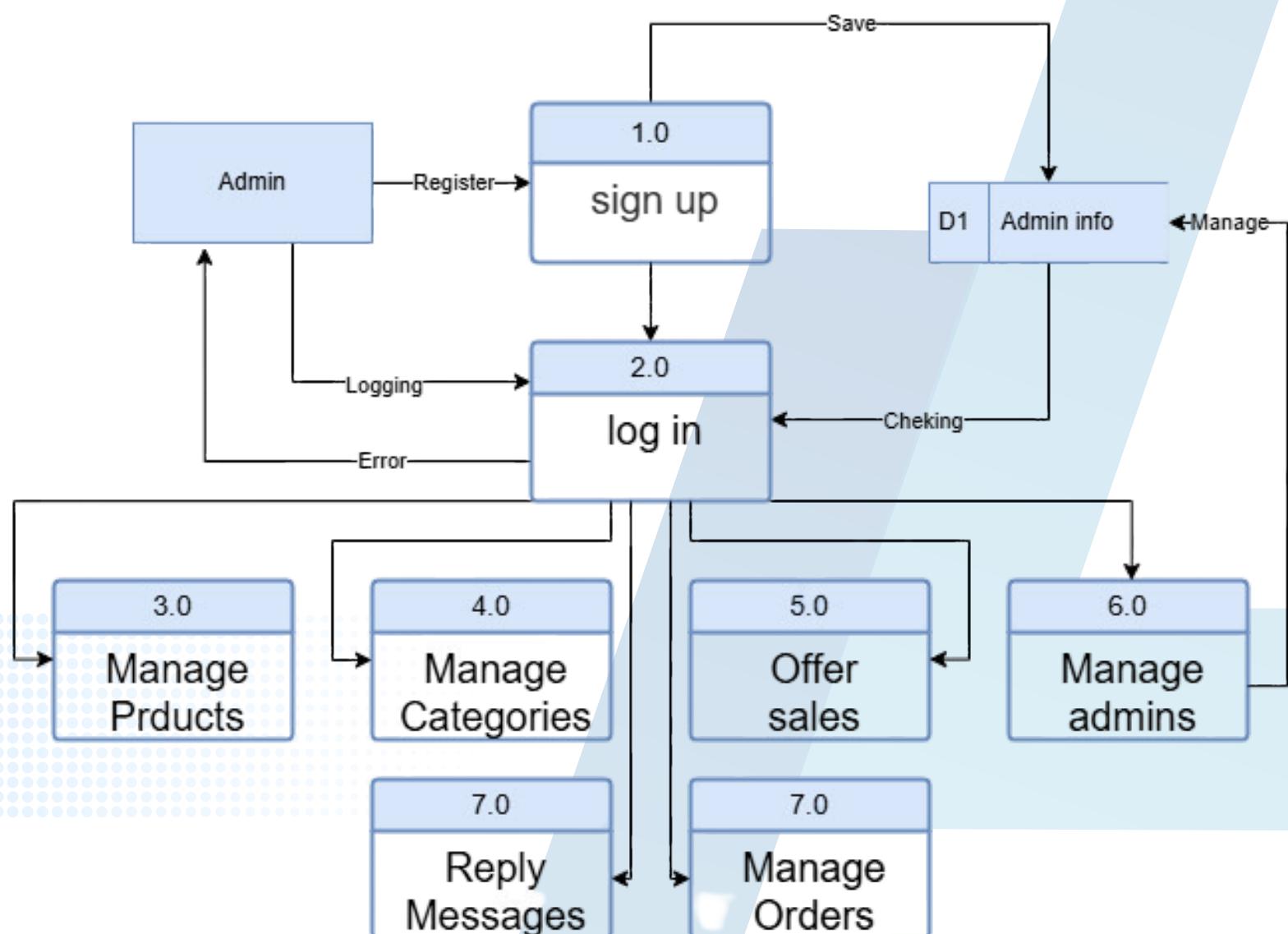


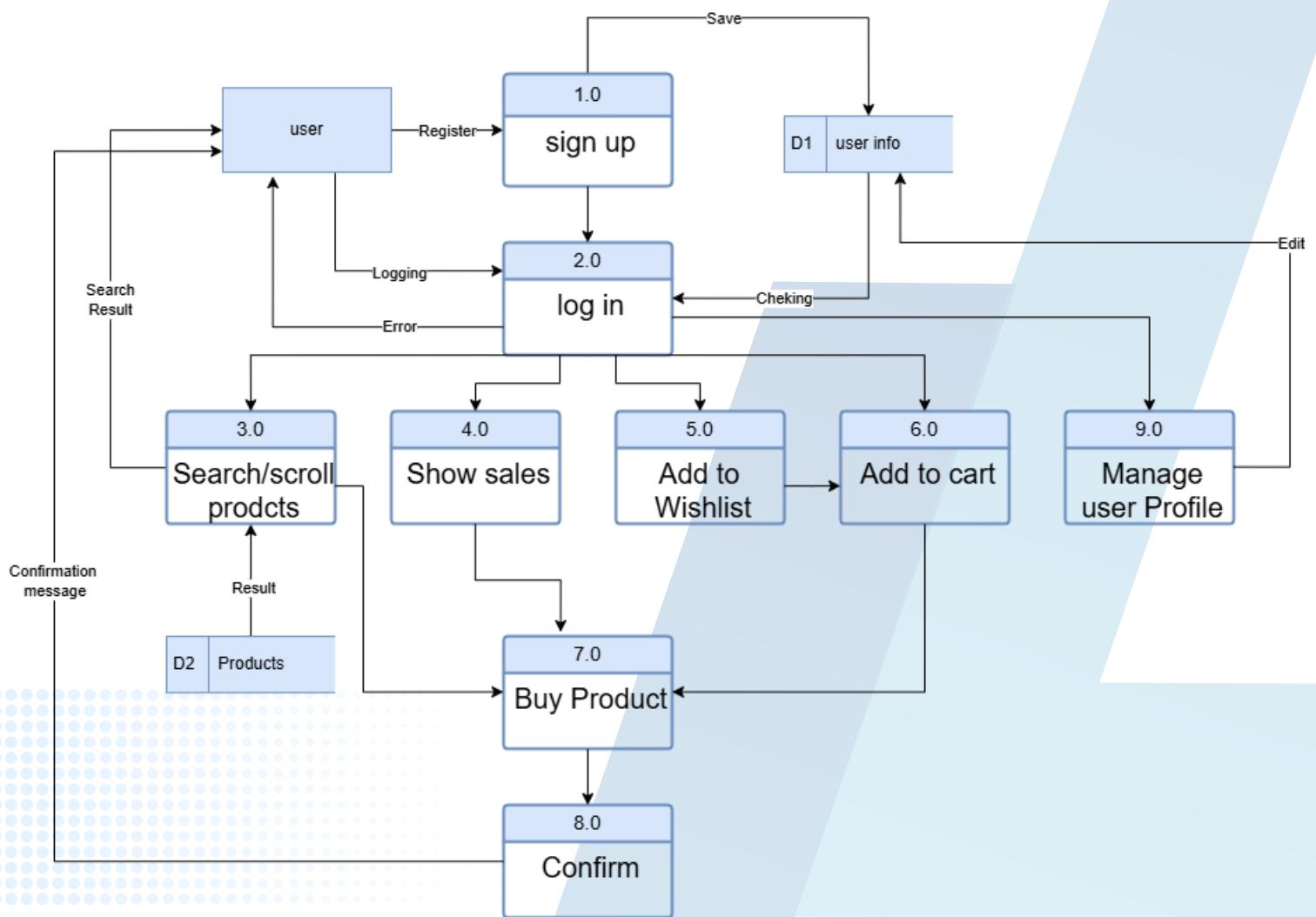
3.2 Context Diagram

Level 0

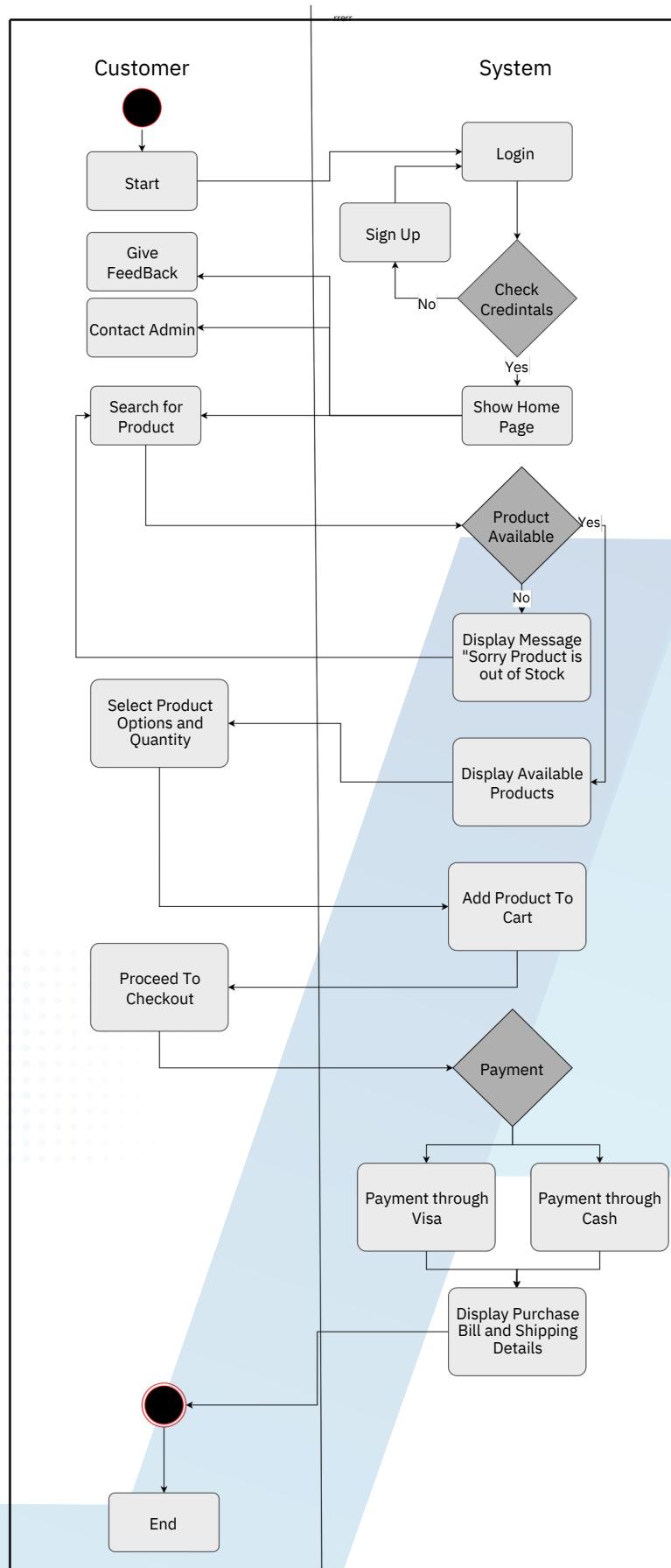


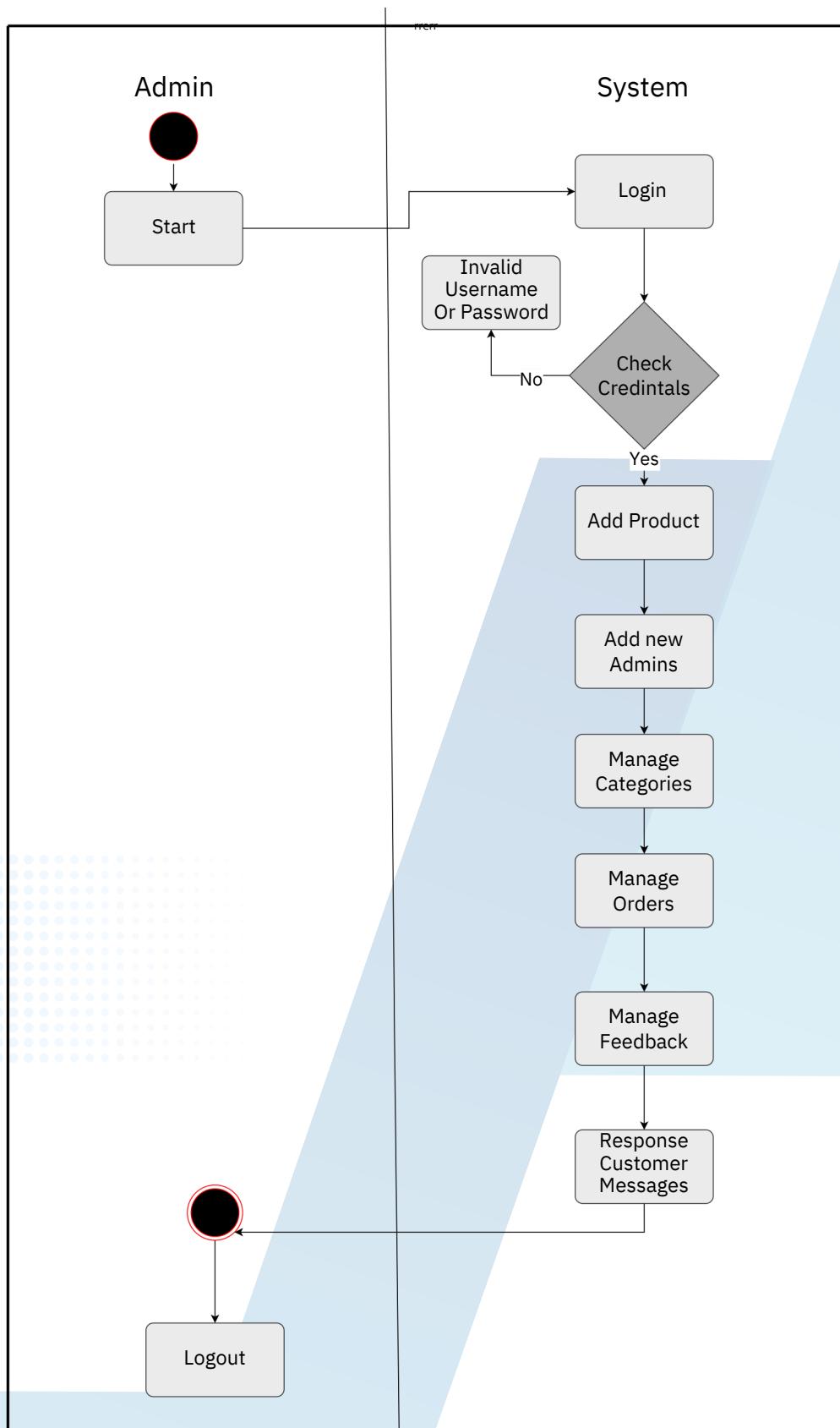
Level 1 (Admin)



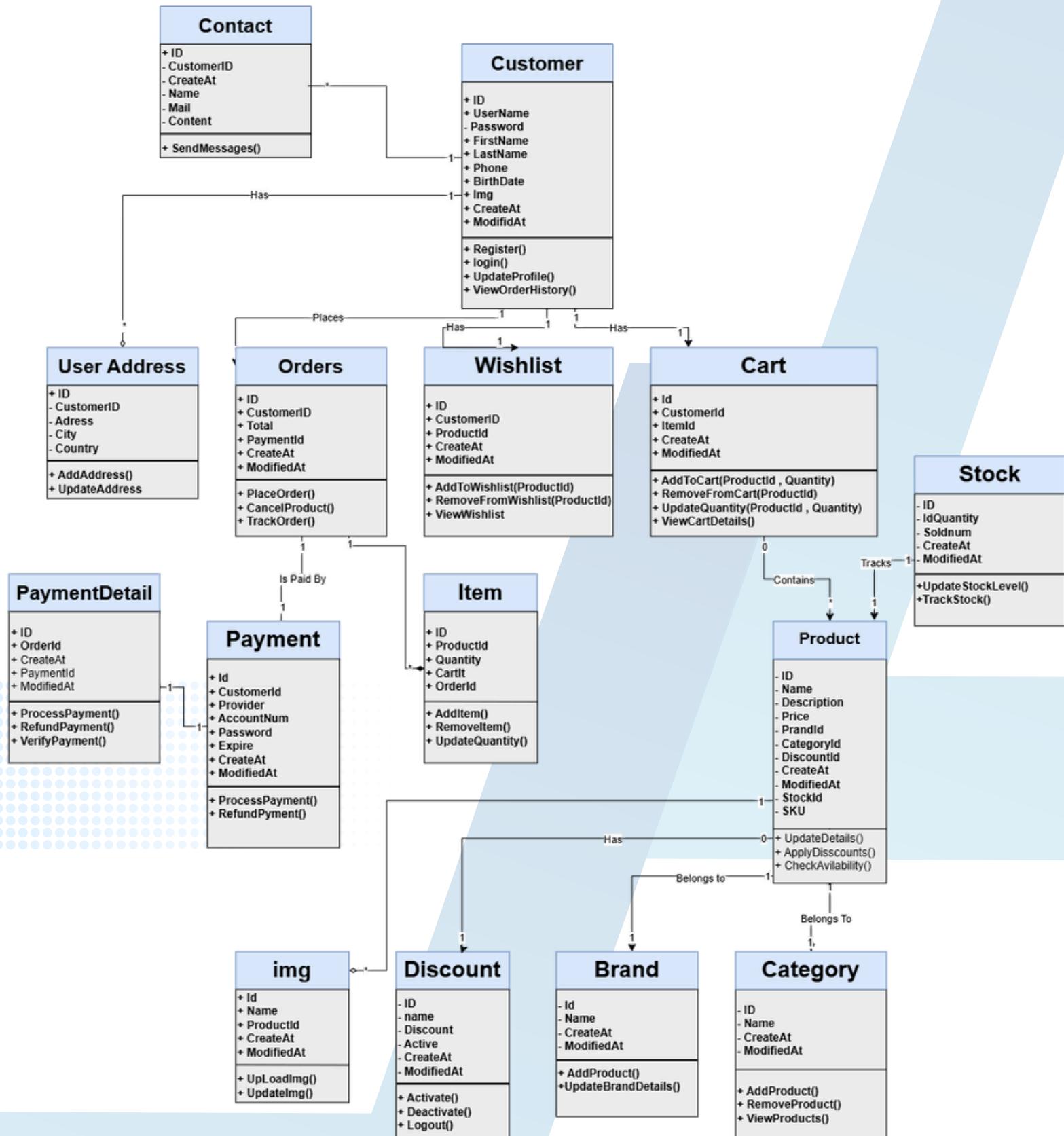
Level 1 (User)

3.3 Activity Diagram (User)

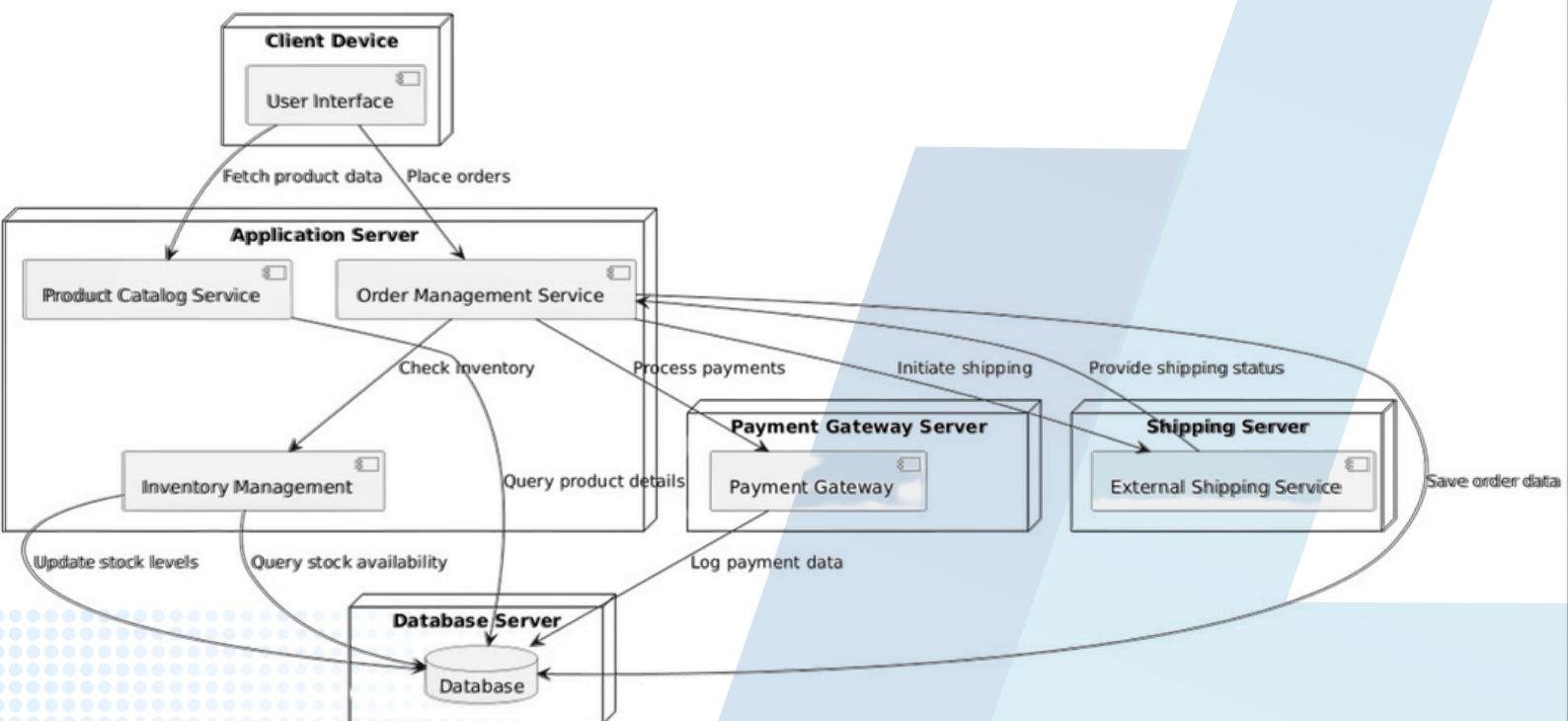


(Admin)

3.4 Class Diagram



3.5 Component Diagram



3.6 Sequence Diagram

