**Palestine polytechnic university**

**College of Information Technology and Computer Engineering**

**Software Engineering**


# Customer Focused E-commerce Site With AI Bot

Submitted To,
Ezdehar Jawabreh

**أرسل إلى**:
إزدهار جوابرة

Submitted By,

**Abdlraheem shawer tamimi,211055**
**Yazan Abu khaled,211066**
**Ahmad Asafrah,211033**

**أرسل من:**

عبدالرحيم شاور تميمي ,211055
يزن أبو خالد ,211066
أحمد عصافرة,211033

24.3.2024

> This report explores an e-commerce platform with AI integration, focusing on user and administrator interactions. With users browsing and purchasing, and administrators managing the platform, our analysis aims to outline essential requirements for effective system design.

- **List of Actors:**

Table 1.0 provides descriptions for the two primary actors in the e-commerce platform.

| Actor | Description |
|---|---|
| User | Represents individuals interacting with the e-commerce platform. |
| Admin | Represents administrators managing the e-commerce platform. |

Table 1.0

- **List of functional requirements as user requirements for The user:**

Table 1.1 presents various user requirements for the e-commerce platform, detailing essential functionalities for user interactions.

| User Requirements | Description |
|---|---|
| Register | Allow users to create an account on the platform. |
| Admin | Enable registered users to log in to their accounts. |
| Logout | Provide users with the ability to securely log out of their accounts. |
| View Recommended Products | Present users with personalized product recommendations based on their preferences. |
| View Product Details | Allow users to view detailed information about specific products. |
| Add to Cart | Enable users to add selected products to their shopping cart. |
| Remove from Cart | Allow users to remove items from their shopping cart. |
| Chat with AI | Provide users with the ability to engage in a conversation with the AI bot for assistance. |
| View Query Results | Allow users to view the results of their queries made to the AI bot. |
| Payment | Enable users to securely complete transactions and make payments for their purchases. |
| View History of Purchases | Allow users to access and review their purchase history on the platform. |

Table 1.1

- **List of functional requirements as user requirements for The admin:**

| User Requirements | Description |
|---|---|
| Add Product | Allow admin users to add new products to the platform's inventory. |
| Remove Product | Provide admin users with the ability to remove products from the platform's inventory. |
| Edit Product Info | Enable admin users to modify and update existing product information. |
| View All Order Details | Allow admin users to access comprehensive details of all orders placed on the platform. |
| View Sales of Products | Allow users to access and review their purchase history on the platform. |

Table 1.1

- **List of nonfunctional requirements measurable requirements.**

Table 1.2 This table provides comprehensive criteria and standards for evaluating the e-commerce platform's performance, reliability, security, usability, maintainability, interoperability, and handling of exceptions.

| **Performance** | **Response Time: System responds within 2 seconds. Throughput: Supports 100 transactions/minute at peak hours. Scalability: Handles 1000 concurrent users without performance drop.** |
|---|---|
| Reliability | System Uptime: Achieves 99.9% uptime monthly. MTBF: Average time between failures is 500 hours. Error Rate: Below 0.5% for user interactions. |
| Security | Data Encryption: Utilizes AES-256 encryption for user data. Access Control: Implements robust role-based access control. Compliance: GDPR compliant, ensuring user data privacy. |
| Usability | Interface Responsiveness: Loads within 3 seconds on all devices. Accessibility: Adheres to WCAG 2.1 standards. Error Handling: Provides clear error messages for user guidance. |
| Maintainability | Code Maintainability: Source code maintains a minimum index of 80. Documentation: Regularly updated and accessible documentation for stakeholders. |
| Interoperability | Integration: Supports seamless integration with popular payment gateways and shipping providers. API Compatibility: APIs follow RESTful principles and are well-documented. |
| Exceptions | The user provides invalid information or the chosen username/email is already in use. |

Table 1.2

## 1.3 Use Case Model:

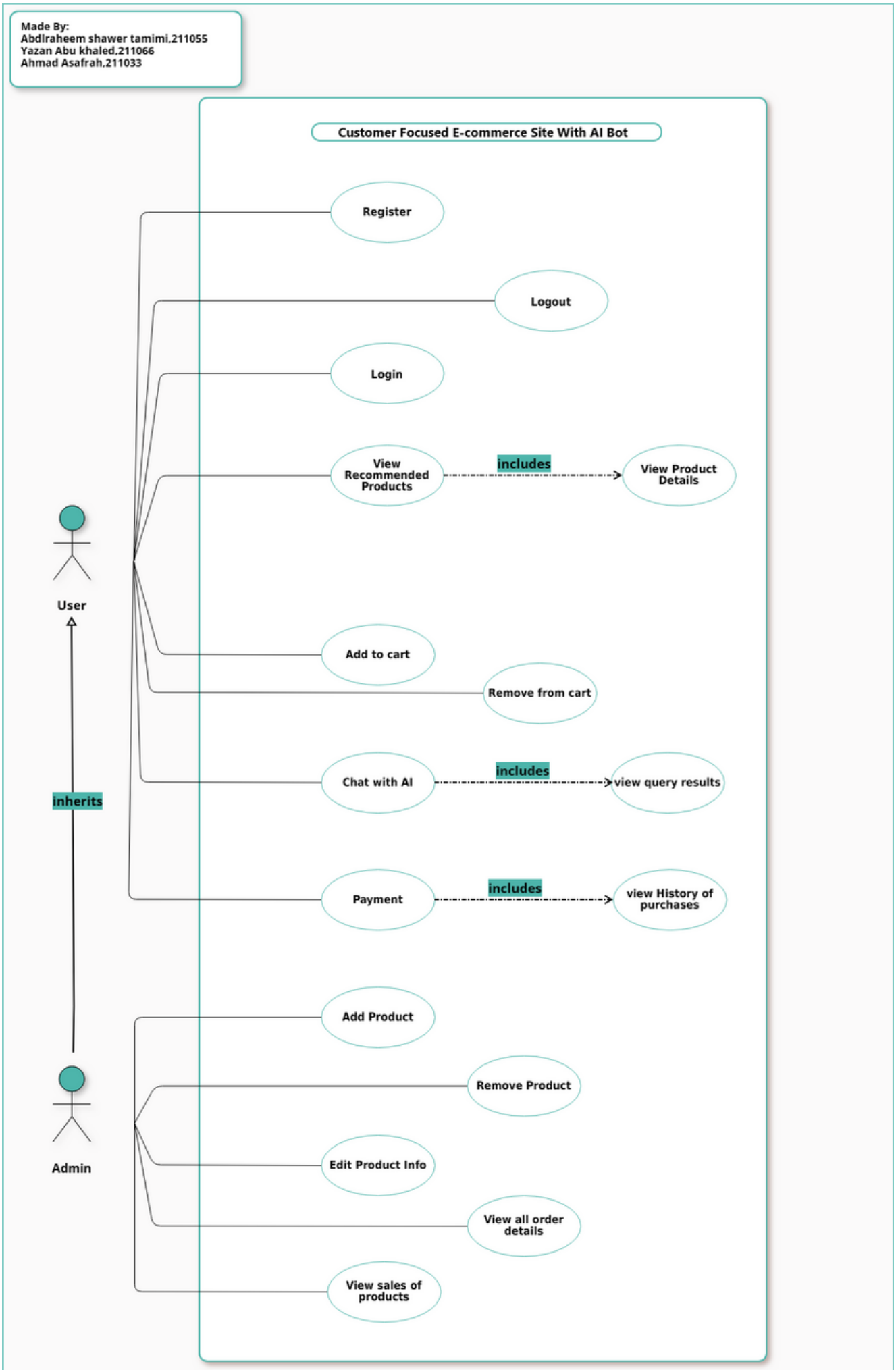



**Figure 1.3 Use Case ModelCustomer Focused E-commerce Site With AI Bot**

- **List of functional requirements as system requirements**

Table 1.4 outlines the Register use case description, illustrating the interaction between a user and the e-commerce platform during the registration process.

| **Use Case** | **Register** |
|---|---|
| primary Actor | User |
| Goal in Context | Provide users with the ability to create an account on the platform. |
| Preconditions | The user is not already registered on the platform. |
| Trigger | The user desires to access personalized features such as saving preferences or making purchases. |
| Scenario | 1) The user navigates to the registration page. 2) The user enters their basic details (name, email, password). 3) The user submits the registration form. 4) The system validates the information and creates the user account. |
| Exceptions | The user provides invalid information or the chosen username/email is already in use. |

Table 1.4 Register In Use Case Description

Table 1.5 details the Login use case description, highlighting the interaction between a user and the e-commerce platform when logging into their account.

| **Use Case** | **Login** |
|---|---|
| primary Actor | User |
| Goal in Context | Provide users with the ability to access their account on the platform. |
| Preconditions | The user is not currently logged in. |
| Trigger | The user desires to access their account to perform actions such as viewing order history or making purchases. |
| Scenario | 1) The user navigates to the login page. 2) The user enters their username/email and password. 3) The user submits the login form. 4) The system validates the credentials and grants access to the user's account. |
| Exceptions | The user provides invalid credentials or the account is inactive/suspended. |

Table 1.5 Log In Use Case Description

- **List of functional requirements as system requirements**

Table 1.6 presents the Logout use case description, demonstrating the interaction between a user and the e-commerce platform when logging out of their account.

| **Use Case** | **Logout** |
|---|---|
| primary Actor | User |
| Goal in Context | Allow users to securely end their session on the platform. |
| Preconditions | The user is currently logged in to their account. |
| Trigger | The user wishes to exit their session or switch to a different account. |
| Scenario | 1) The user selects the logout option from the navigation menu. 2) The system confirms the logout action. 3) The system ends the user's session and redirects them to the login page. |
| Exceptions | None. |

Table 1.6 Logout Use Case Description

Table 1.7 illustrates the View Recommended Products use case description, showcasing the interaction between a user and the e-commerce platform when browsing personalized product recommendations.

| **Use Case** | **View Recommended Products** |
|---|---|
| primary Actor | User |
| Goal in Context | Provide users with personalized product recommendations based on their preferences. |
| Preconditions | The user is logged in to their account. |
| Trigger | The user navigates to the recommended products section or homepage. |
| Scenario | 1) The user visits the recommended products section. 2) The system retrieves and displays recommended products based on the user's browsing history or preferences. |
| Exceptions | None. |

Table 1.7 View Recommended Products Use Case Description

- **List of functional requirements as system requirements**

Table 1.8 provides an overview of the View Product Details use case, depicting the interaction between a user and the e-commerce platform when viewing detailed information about a specific product.

| **Use Case** | **View Product Details** |
|---|---|
| Primary Actor | User |
| Goal in Context | Provide detailed information about a specific product |
| Preconditions | User is viewing recommended products |
| Trigger | User selects a product for more details |
| Scenario | 1) User clicks on a product from the recommended list. 2) System retrieves and displays detailed information about the selected product. |
| Exceptions | Product not found or unavailable. |

Table 1.8 View Product Details Case Description

Table 1.9 offers a comprehensive outline of the Add to Cart use case, delineating the dynamic engagement between a user and the e-commerce platform during the process of adding products to their cart.

| **Use Case** | **Add to Cart** |
|---|---|
| Primary Actor | User |
| Goal in Context | Allow users to add products to their cart |
| Preconditions | User is viewing product details |
| Trigger | User wants to add a product to their cart |
| Scenario | 1) User clicks on "Add to Cart" button for a product. 2) System adds the product to the user's cart. |
| Exceptions | Product not found or unavailable. |

Table 1.9 Add to Cart Use Case Description

- **List of functional requirements as system requirements**

Table 1.10 presents a detailed overview of the Remove from Cart use case, illustrating the interactive process between a user and the e-commerce platform when removing products from their cart.

| **Use Case** | **Remove from Cart** |
|---|---|
| Primary Actor | User |
| Goal in Context | Allow users to remove products from their cart |
| Preconditions | User has items in their cart |
| Trigger | User wants to remove a product from their cart |
| Scenario | 1) User navigates to their cart. 2) User selects the option to remove a product. 3) System removes the product from the user's cart. |
| Exceptions | Product not found in the user's cart. |

Table 1.10 Remove from Cart Use Case Description

Table 1.11 outlines the Chat with AI use case, delineating the interaction between a user and the AI bot to provide automated assistance and information.

| **Use Case** | **Chat with AI** |
|---|---|
| Primary Actor | User |
| Goal in Context | Provide users with automated assistance and information |
| Preconditions | User is logged in |
| Trigger | User wants to chat with the AI bot |
| Scenario | 1) User clicks on the chat option. 2) User enters a query. 3) System processes the query and provides a response. |
| Exceptions | AI bot unable to understand or process the query. |

Table 1.11 Chat with AI Use Case Description

- **List of functional requirements as system requirements**

Table 1.12 provides an overview of the View Query Results use case, illustrating the interaction between a user and the system to allow users to view query results.

| **Use Case** | **View Query Results** |
|---|---|
| Primary Actor | User |
| Goal in Context | Allow users to view query results |
| Preconditions | User is logged in |
| Trigger | User wants to view query results |
| Scenario | 1) User enters a query in the chat with AI interface. 2) System processes the query and retrieves relevant information. 3) System displays the query results to the user. |
| Exceptions | Unable to process the query, no relevant results found. |

Table 1.12 View Query Results Use Case Description

Table 1.13 offers an overview of the Payment use case, delineating the process by which users can make online payments for purchases.

| **Use Case** | **Payment** |
|---|---|
| Primary Actor | User |
| Goal in Context | Enable users to make online payments for purchases |
| Preconditions | User is logged in |
| Trigger | User wants to make a payment |
| Scenario | 1) User proceeds to checkout after adding desired items to cart. 2) User selects payment method and provides necessary details. 3) System processes payment securely. |
| Exceptions | Payment failure, incomplete payment information. |

Table 1.13 Payment Use Case Description

- **List of functional requirements as system requirements**

Table 1.14 presents an overview of the View History of Purchases use case, illustrating the process by which users can access and review their purchase history.

| **Use Case** | **View History of Purchases** |
|---|---|
| Primary Actor | User |
| Goal in Context | Allow users to view their purchase history |
| Preconditions | User is logged in |
| Trigger | User wants to view their purchase history |
| Scenario | 1) User navigates to purchase history section. 2) System retrieves and displays a list of all past purchases. |
| Exceptions | Unable to retrieve purchase history. |

Table 1.14 View History of Purchases Use Case Description

Table 1.15 provides an overview of the Add Product use case, illustrating the process by which admins can add new products to the system.

| **Use Case** | **Add Product** |
|---|---|
| Primary Actor | Admin |
| Goal in Context | Enable admins to add new products |
| Preconditions | Admin is logged in |
| Trigger | Admin wants to add a new product |
| Scenario | 1) Admin navigates to product management section. 2) Admin selects option to add a new product. 3) System prompts admin to enter product details. |
| Exceptions | None |

Table 1.15 Add Product Use Case Description

- **List of functional requirements as system requirements**

Table 1.16 outlines the Remove Product use case, detailing the process by which admins can remove products from the inventory.

| **Use Case** | **Remove Product** |
|---|---|
| Primary Actor | Admin |
| Goal in Context | Allow admins to remove products from inventory |
| Preconditions | Admin is logged in, product exists in inventory |
| Trigger | Admin wants to remove a product from inventory |
| Scenario | 1) Admin navigates to product management section. 2) Admin selects option to remove a product. 3) System prompts admin to confirm removal. |
| Exceptions | Product not found in inventory, unauthorized access. |

Table 1.16 Remove Product Use Case Description

Table 1.17 provides an overview of the Edit Product Info use case, illustrating the process by which admins can modify product information.

| **Use Case** | **Edit Product Info** |
|---|---|
| Primary Actor | Admin |
| Goal in Context | Enable admins to modify product information |
| Preconditions | Admin is logged in, product exists in inventory |
| Trigger | Admin wants to edit product details |
| Scenario | 1) Admin navigates to product management section. 2) Admin selects option to edit a product. 3) System retrieves product details for editing. |
| Exceptions | Product not found in inventory, unauthorized access. |

Table 1.17 Edit Product Info Use Case Description

- **List of functional requirements as system requirements**

Table 1.18 offers an overview of the View All Order Details use case, illustrating the process by which admins can access comprehensive order information for tracking and management.

| **Use Case** | **View All Order Details** |
|---|---|
| Primary Actor | Admin |
| Goal in Context | Provide admins with access to comprehensive order information for tracking and management |
| Preconditions | Admin is logged in |
| Trigger | Admin needs to review all orders |
| Scenario | 1) Admin navigates to order management section. 2) Admin selects option to view all orders. 3) System retrieves and displays a list of all orders with relevant details. |
| Exceptions | No orders found, unauthorized access, or technical issues during order retrieval. |

Table 1.18 View All Order Details Use Case Description

Table 1.19 outlines the View Sales of Product use case, detailing how admins can track the sales performance of products.

| **Use Case** | **View Sales of Product** |
|---|---|
| Primary Actor | Admin |
| Goal in Context | Allow admins to track sales performance of products |
| Preconditions | Admin is logged in |
| Trigger | Admin wants to view sales data for products |
| Scenario | 1) Admin navigates to sales analysis section. 2) Admin selects option to view sales of products. 3) System retrieves and displays sales data for products. |
| Exceptions | No sales data available, unauthorized access. |

Table 1.19 View Sales of Product Use Case Description

Activity diagram for the functional requirment **make payment**:

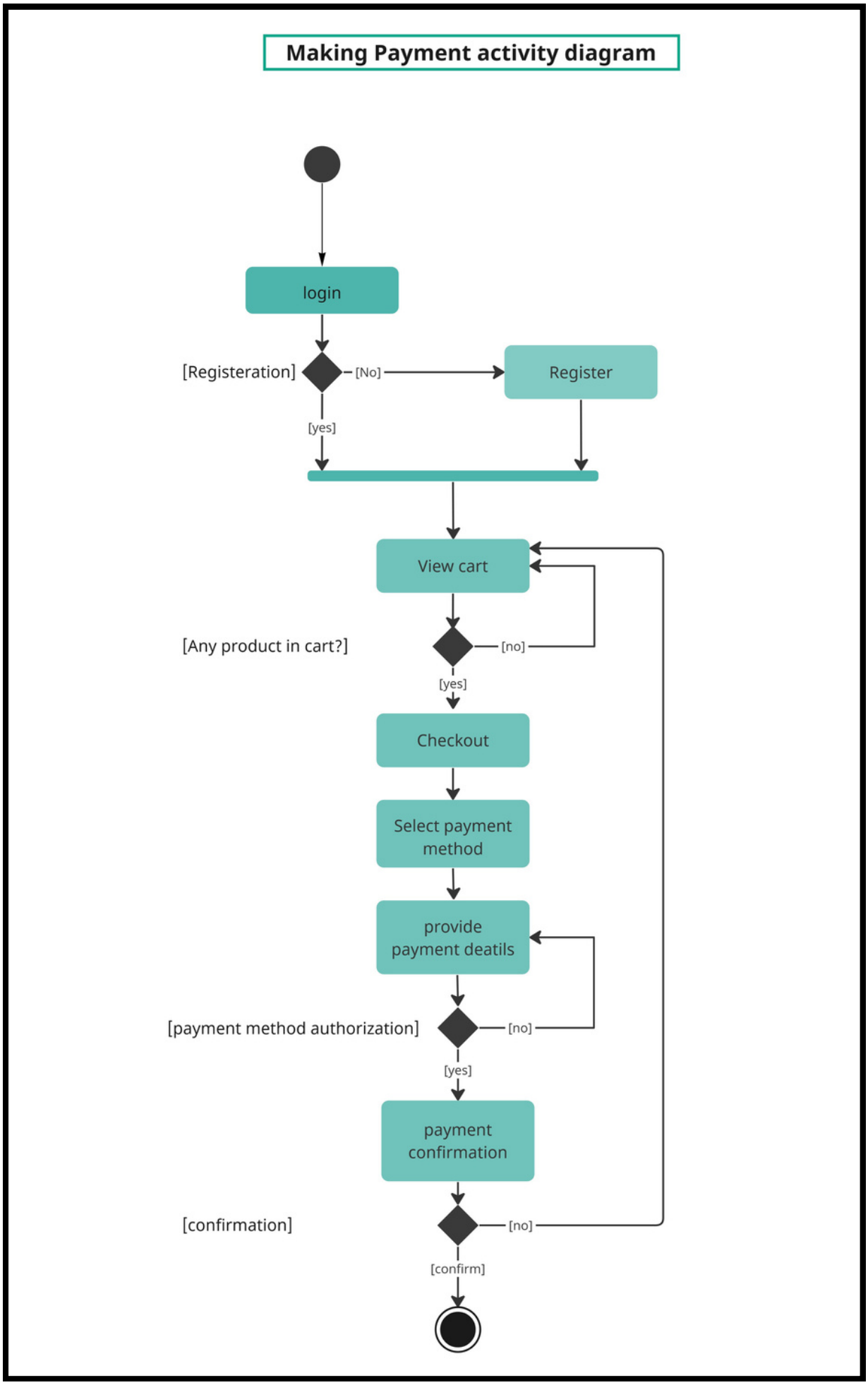



Diagram 1

Activity diagram for the functional requirment **Remove product**:

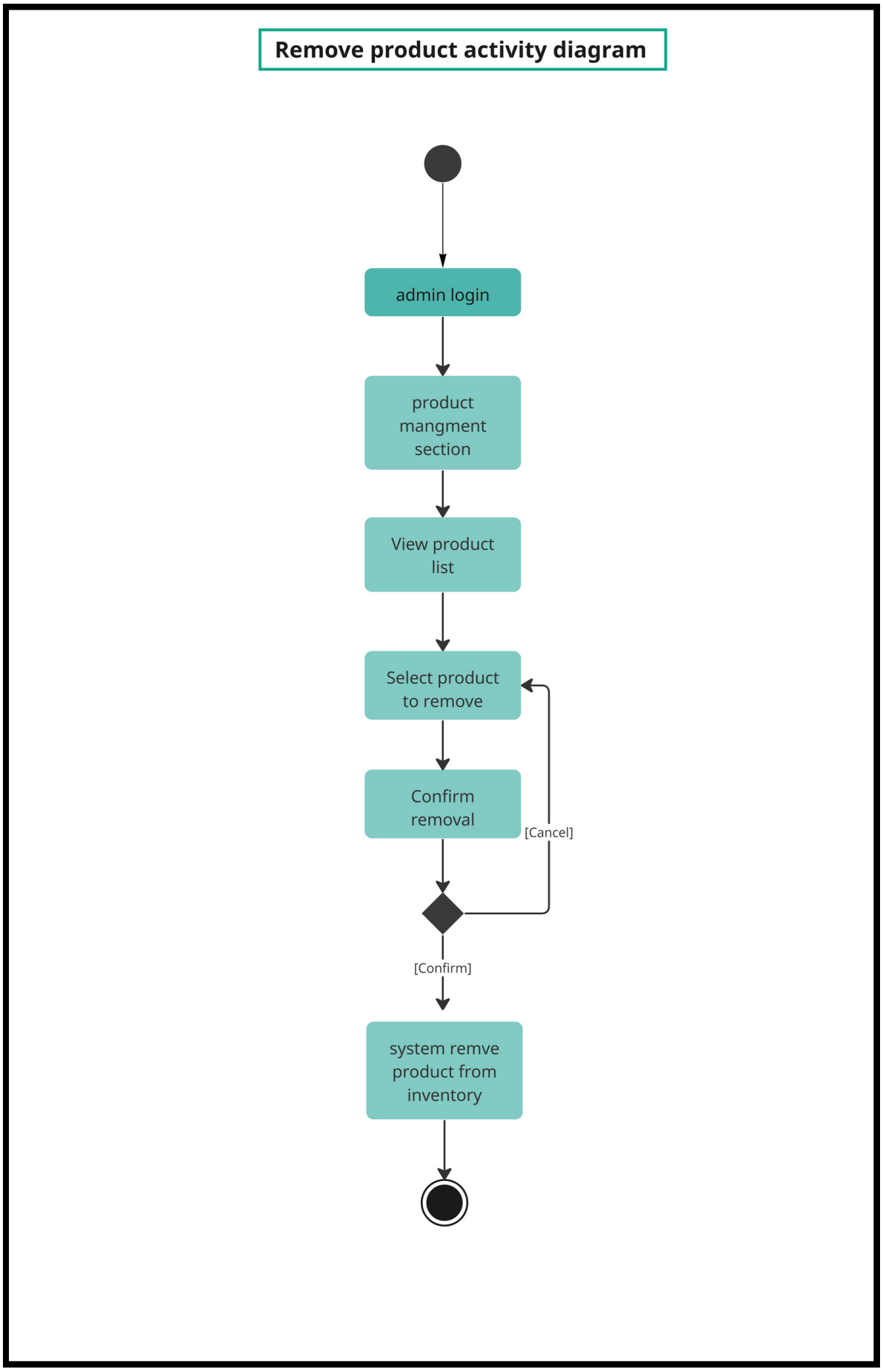



Diagram 2

State diagram for the **Order state:**

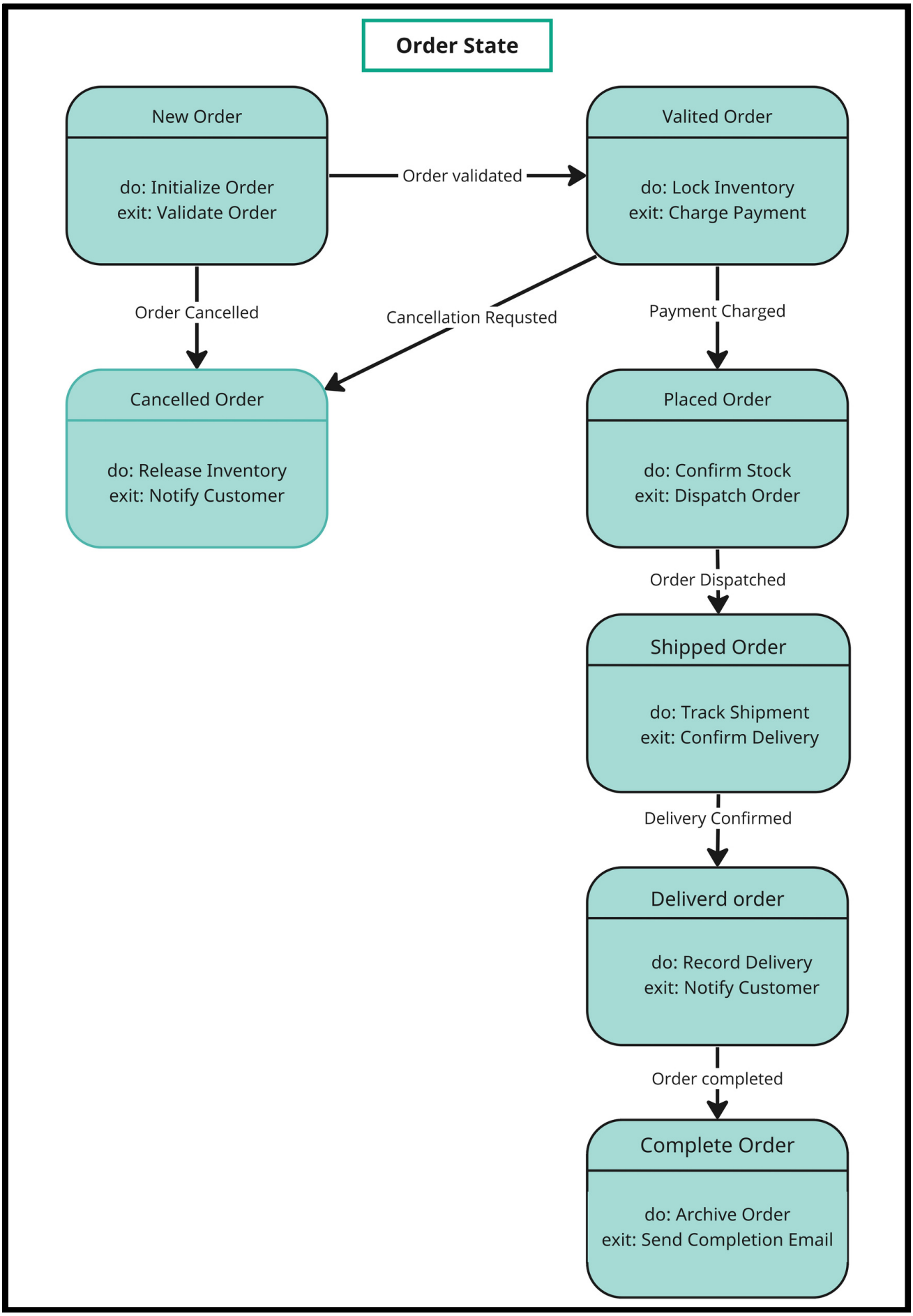



Diagram 3

State diagram for the **Payment state:**

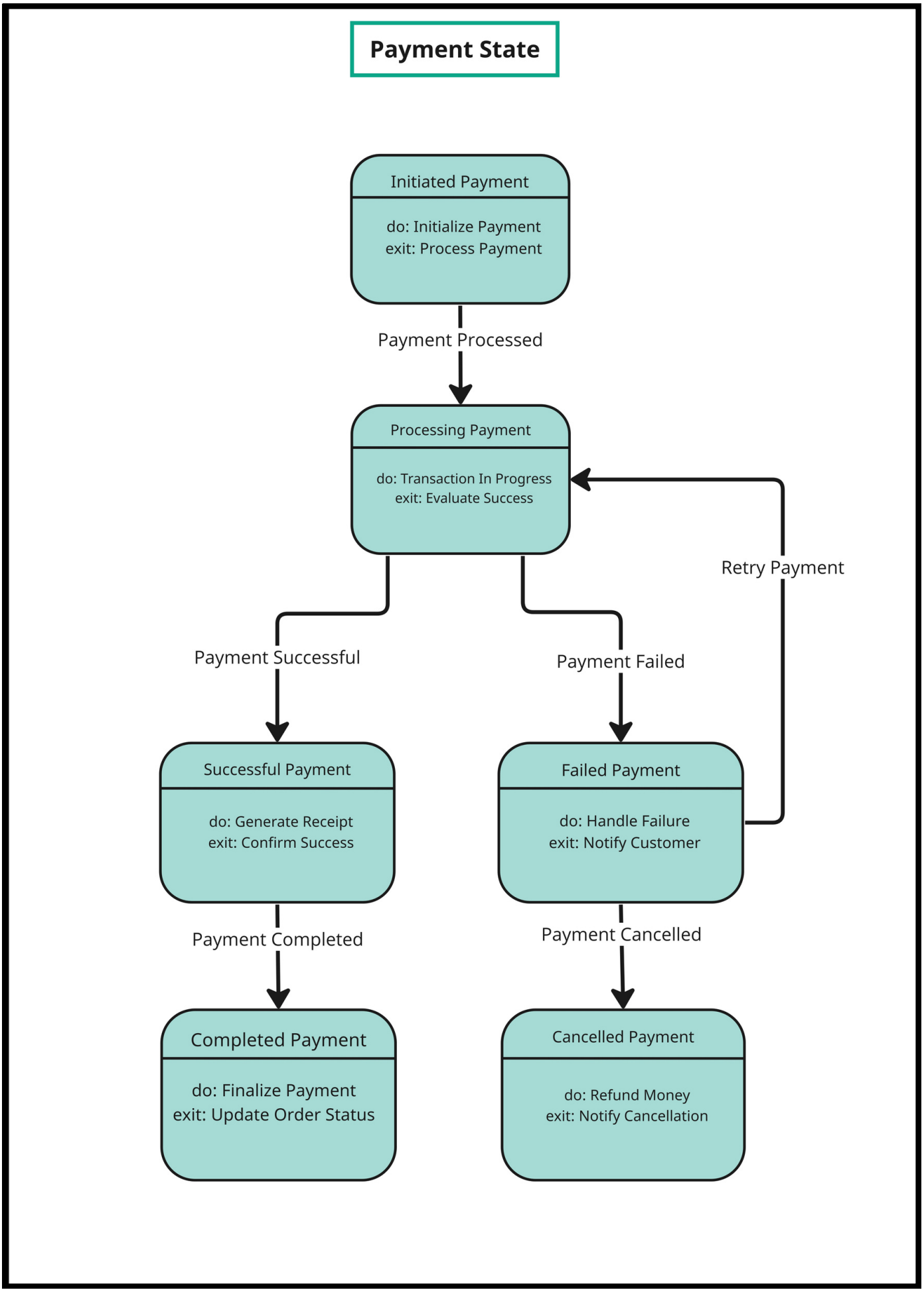



Diagram 4

Sequence diagram for the **make payment:**

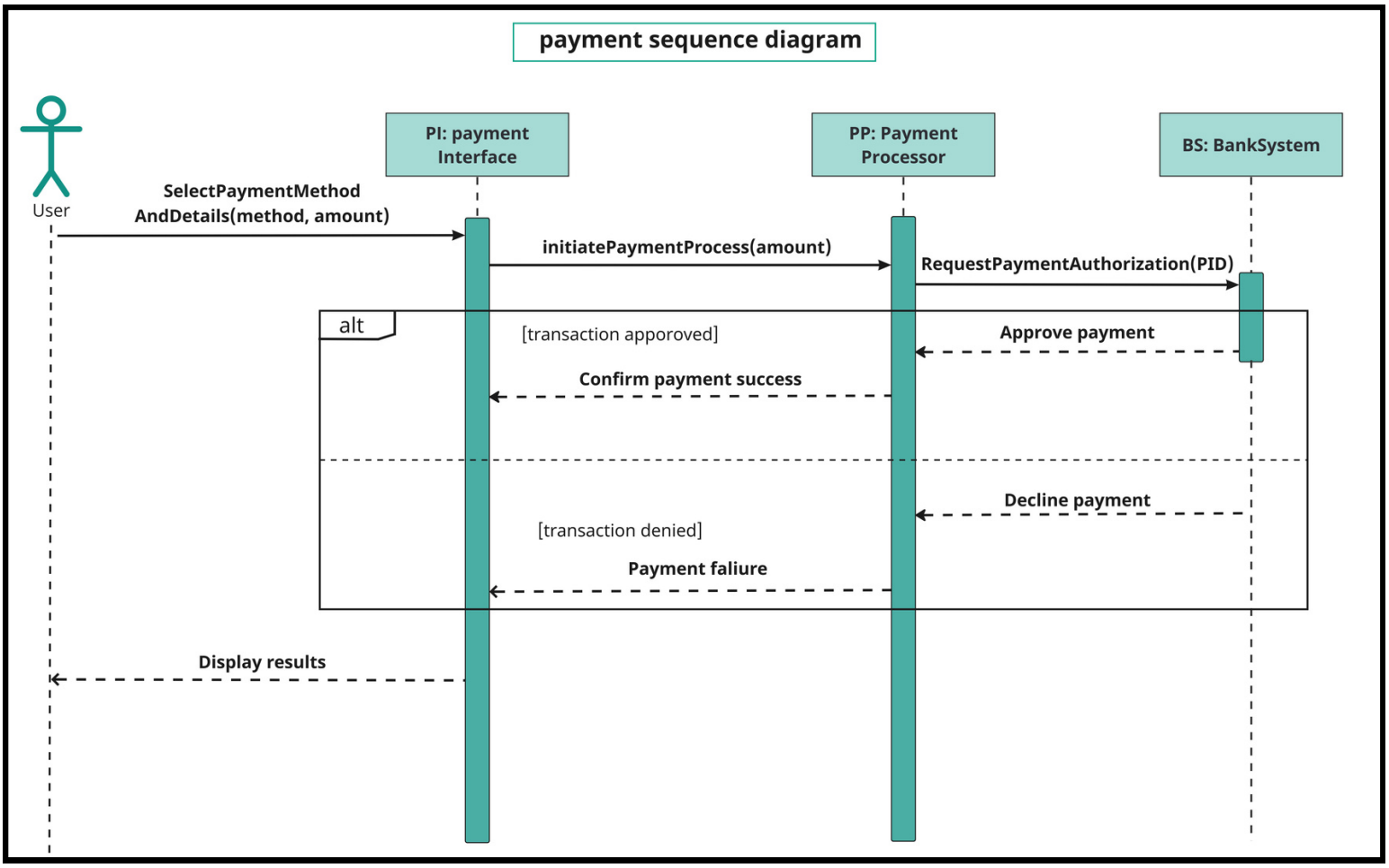



Diagram 5

Sequence diagram for the **view query result:**

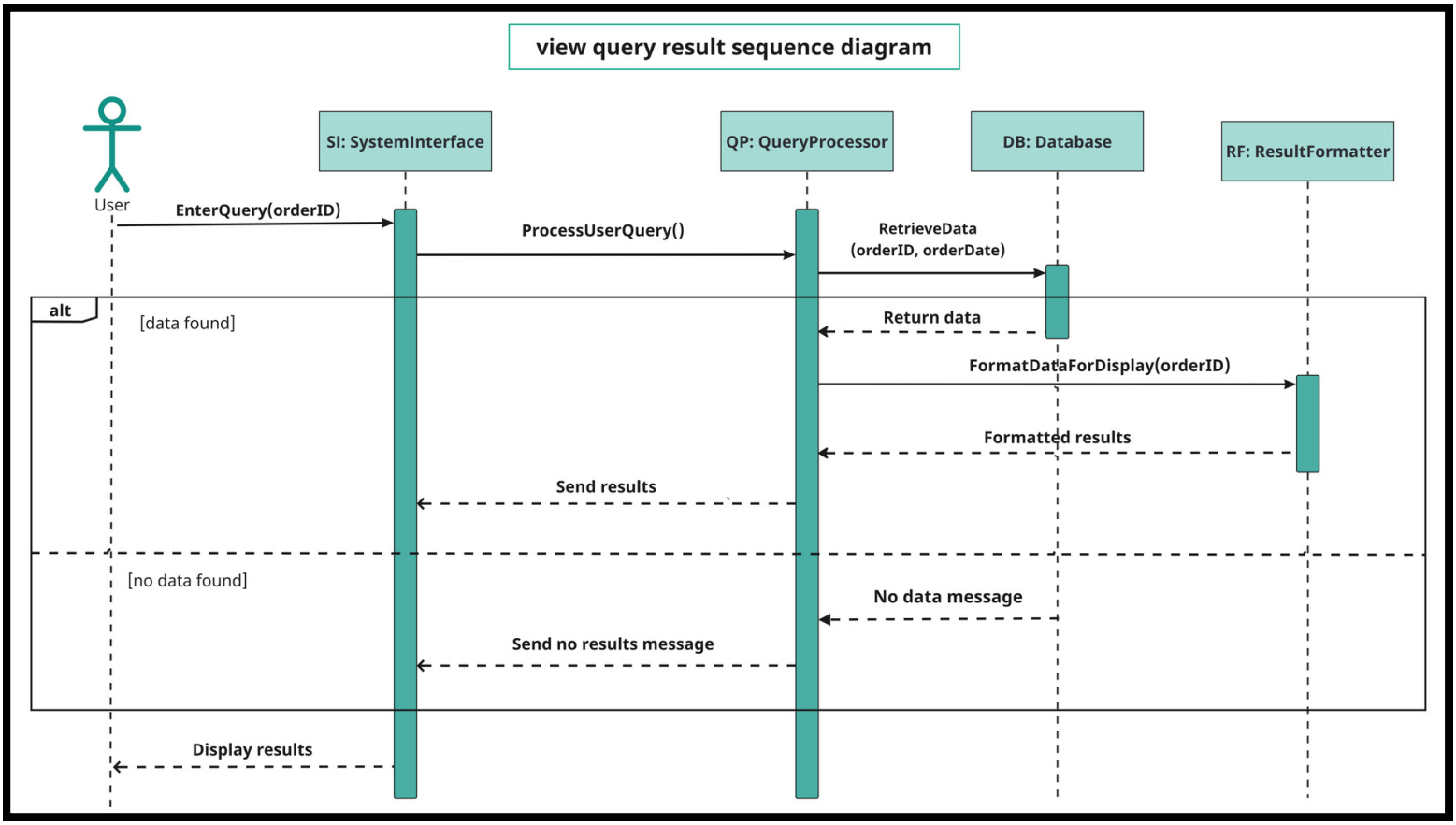



Diagram 6

Class diagram for the system

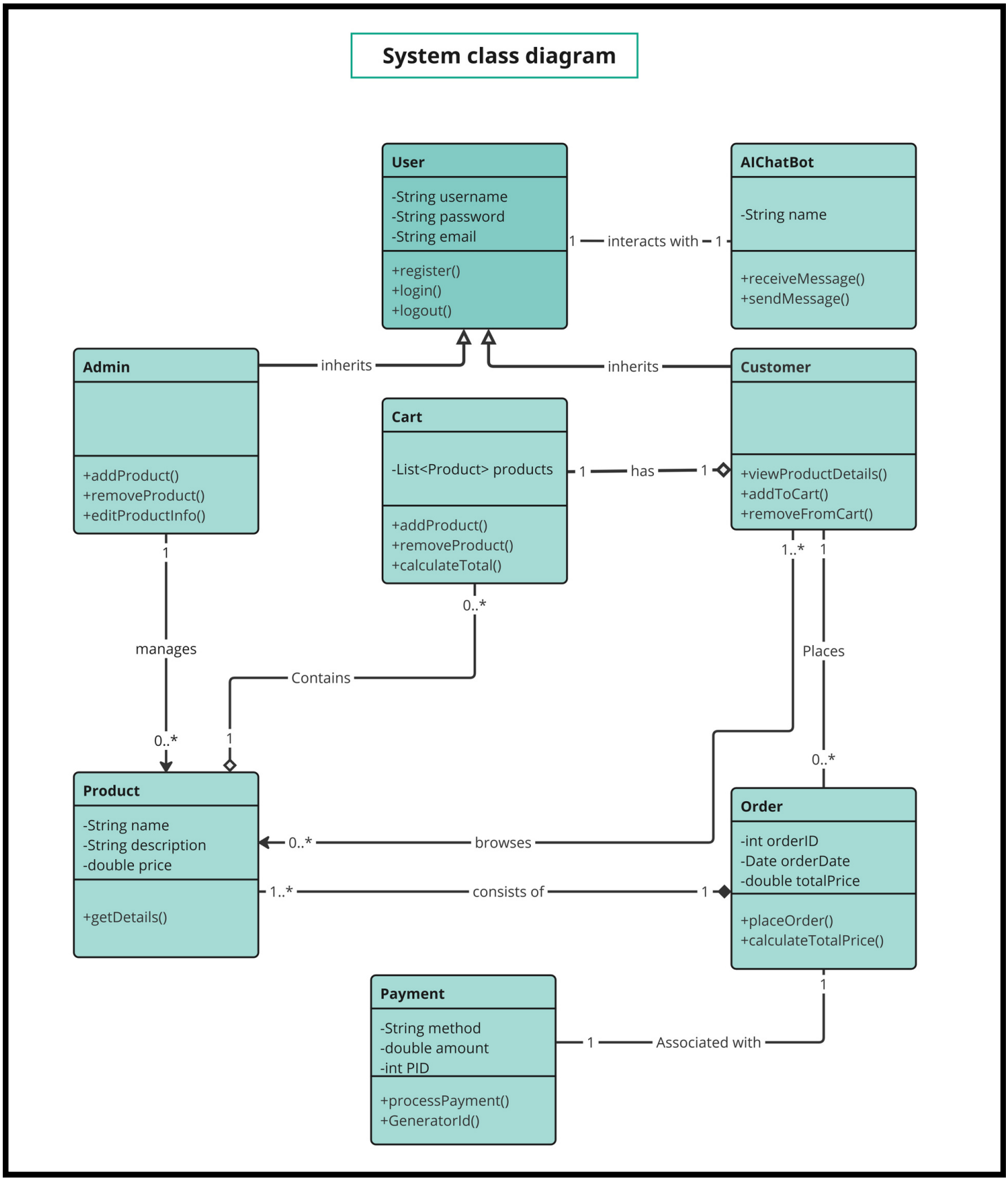
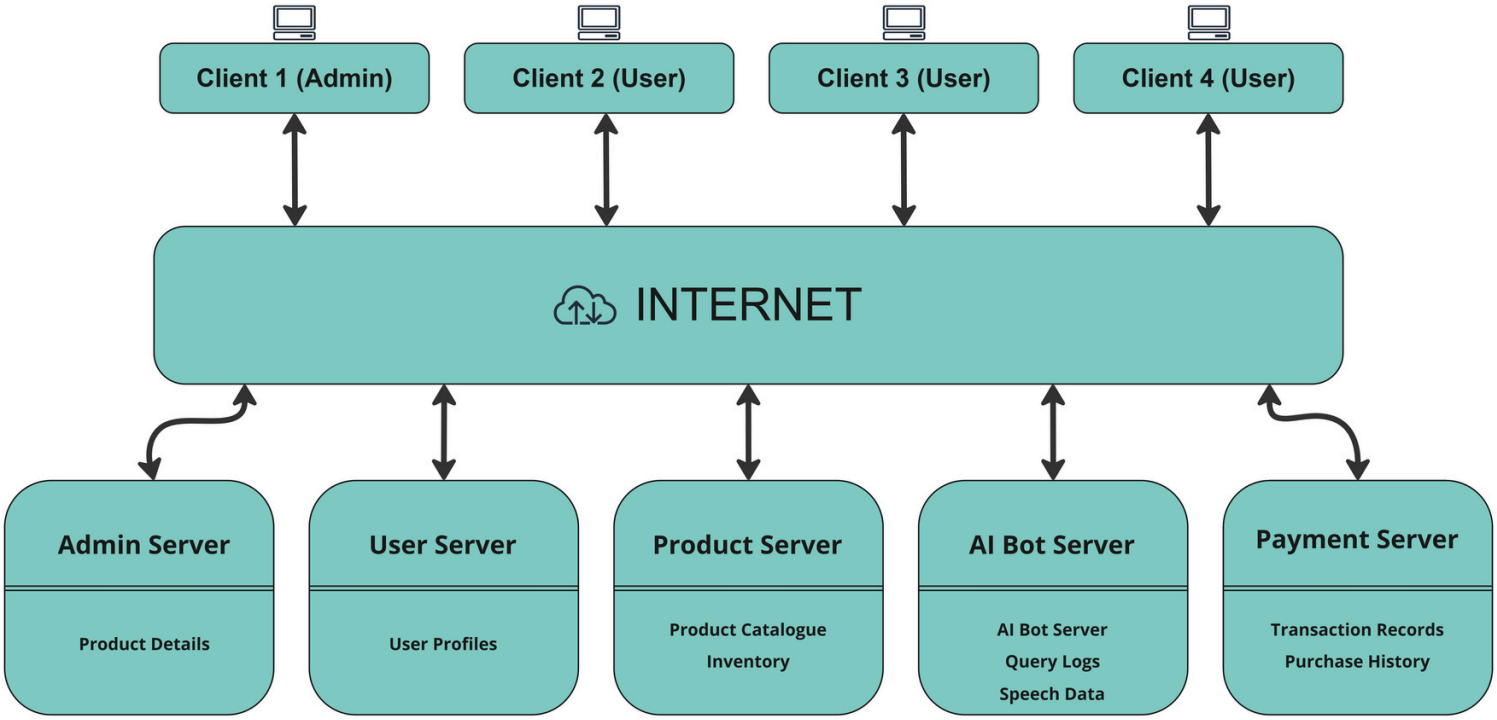



Diagram 7

view our work using **miro** :

https://miro.com/welcomeonboard/VlBUMGNXYXJLTUxobk1kNklVTGVXemF6dFZxN1ZLVkIwOHRKR2FFanpVME1oUU12WGdsQ1FHVmhmaXRxaUJXR3wzNDU4NzY0NTgyMTU1NzYxNzgxfDI=?share_link_id=306419258781

# Architectural Design Client-server




## Justification of the Client-Server Architecture

**Scalability**: This architecture allows the system to handle increasing loads by adding more servers or upgrading existing hardware, making it ideal for an e-commerce platform that might experience variable traffic.

**Maintainability**: Separation of concerns is a core advantage. Each server has a specific role, making it easier to maintain and update the system without affecting other components.

**Security**: Each server can be secured according to its needs, and security protocols can be applied specifically where sensitive data is handled, such as in the Payment and User Servers.

**Performance**: Load balancing can be implemented between servers to ensure that no single server becomes a bottleneck, thereby enhancing the overall performance of the system.

**Flexibility**: New services like an additional AI Bot or a new payment gateway can be integrated without significant changes to the client applications.

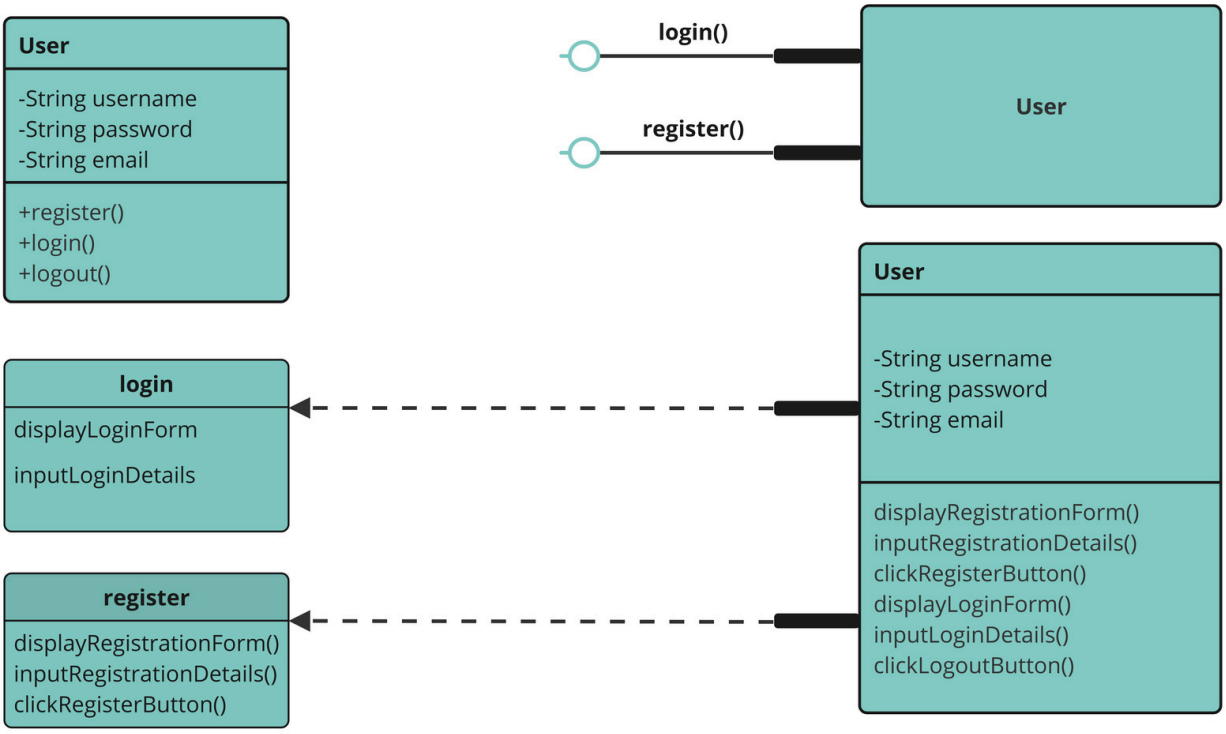
This architecture not only supports current technological needs but also provides a foundation that can evolve with technological advancements and changing business requirements.

## General Description of Components

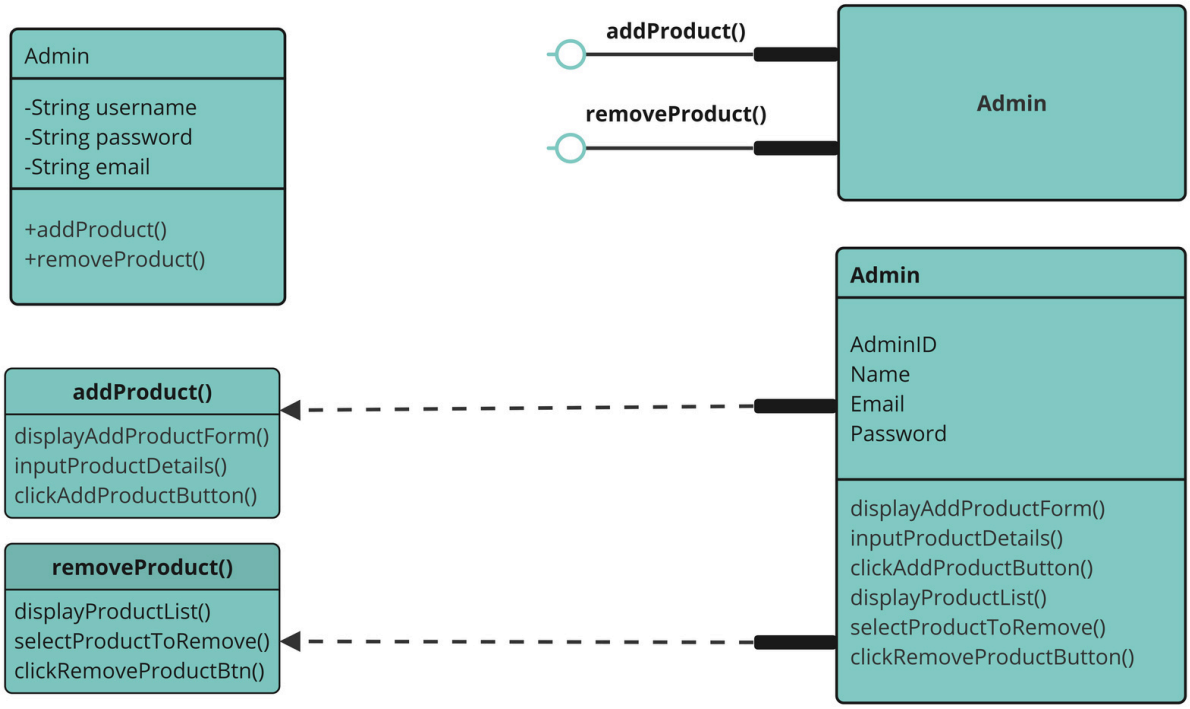
1. **Clients** (Client 1 - Admin, Client 2 - User, Client 3 - User, Client 4 - User):
   - **Function**: Interact with the servers to perform tasks such as managing products (Admin) and browsing or purchasing products (Users).
   - **Data Interaction**: Clients send requests to and receive responses from servers via the Internet, facilitating a dynamic user experience.
2. **Internet**:
   - **Function**: Acts as the communication layer between clients and servers, allowing for the exchange of data across different geographic locations.
   - **Data Interaction**: Transmits data packets between clients and servers securely using protocols like HTTP/HTTPS.
3. **Admin Server**:
   - **Storage**: Holds "Product Details" which include comprehensive information about the products available on the e-commerce platform.
   - **Function**: Allows the Admin to manage product details efficiently.
4. **User Server**:
   - **Storage**: Contains "User Profiles" which store personal and login information of the users, ensuring personalized experiences.
   - **Function**: Manages user authentication and profile settings.
5. **Product Server**:
   - Storage: Stores the "Product Catalogue" and "Inventory" data, critical for maintaining current product offerings and stock levels.
   - Function: Ensures users have up-to-date information on products and availability.
6. **AI Bot Server**:
   - **Storage**: Maintains "Query Logs" and "Speech Data," which help in improving the AI responses based on user interactions.
   - **Function**: Provides automated responses to user queries, enhancing user experience with efficient communication.
7. **Payment Server**:
   - **Storage**: Handles "Transaction Records" and "Purchase History," ensuring transactions are processed securely and efficiently.
   - **Function**: Manages financial transactions and keeps detailed records of all purchases.

# Component level design:

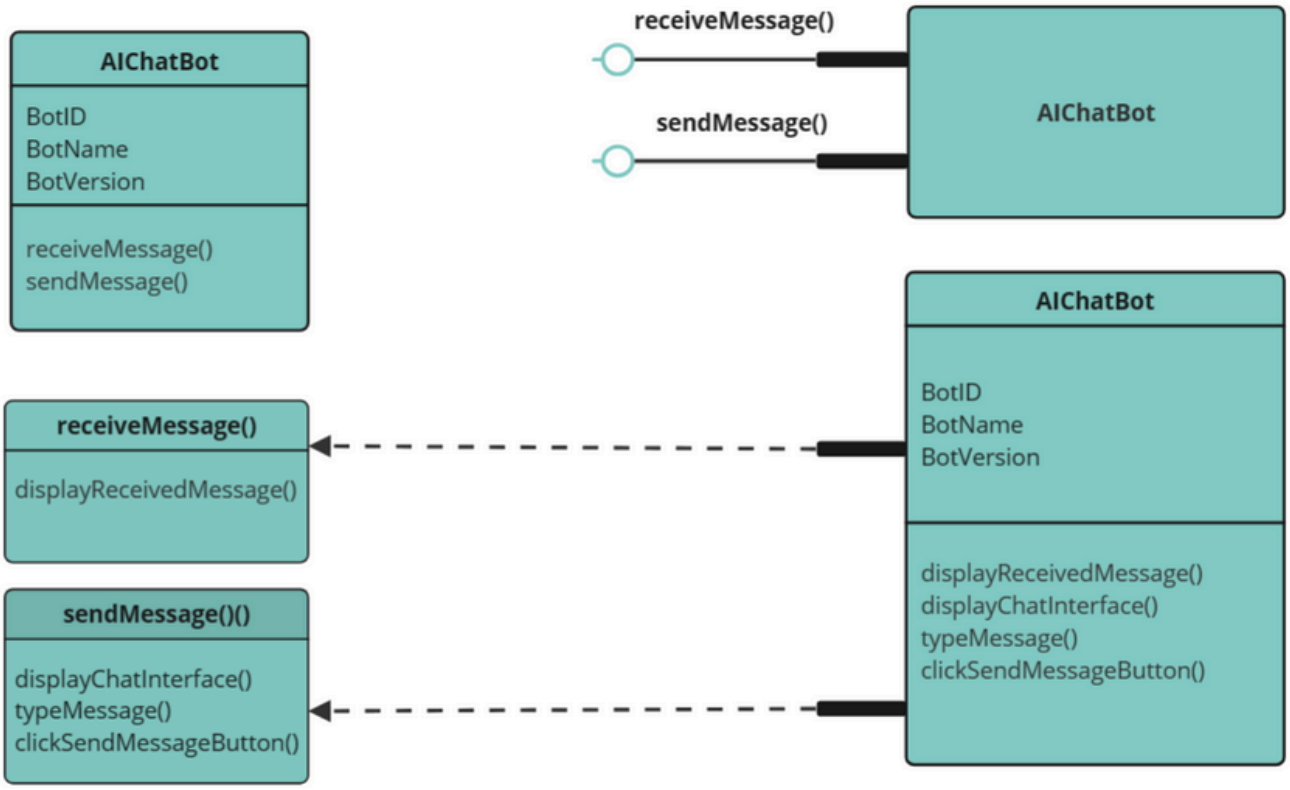
## User:



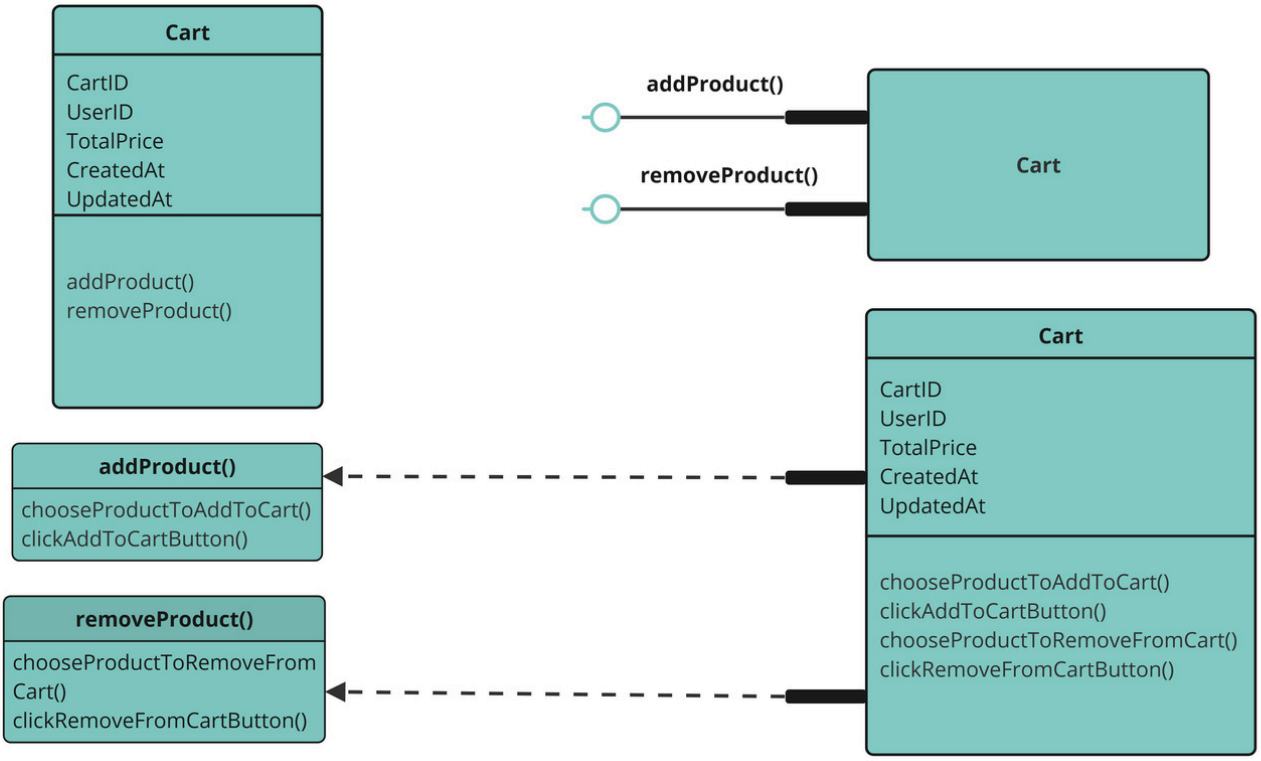

## Admin:



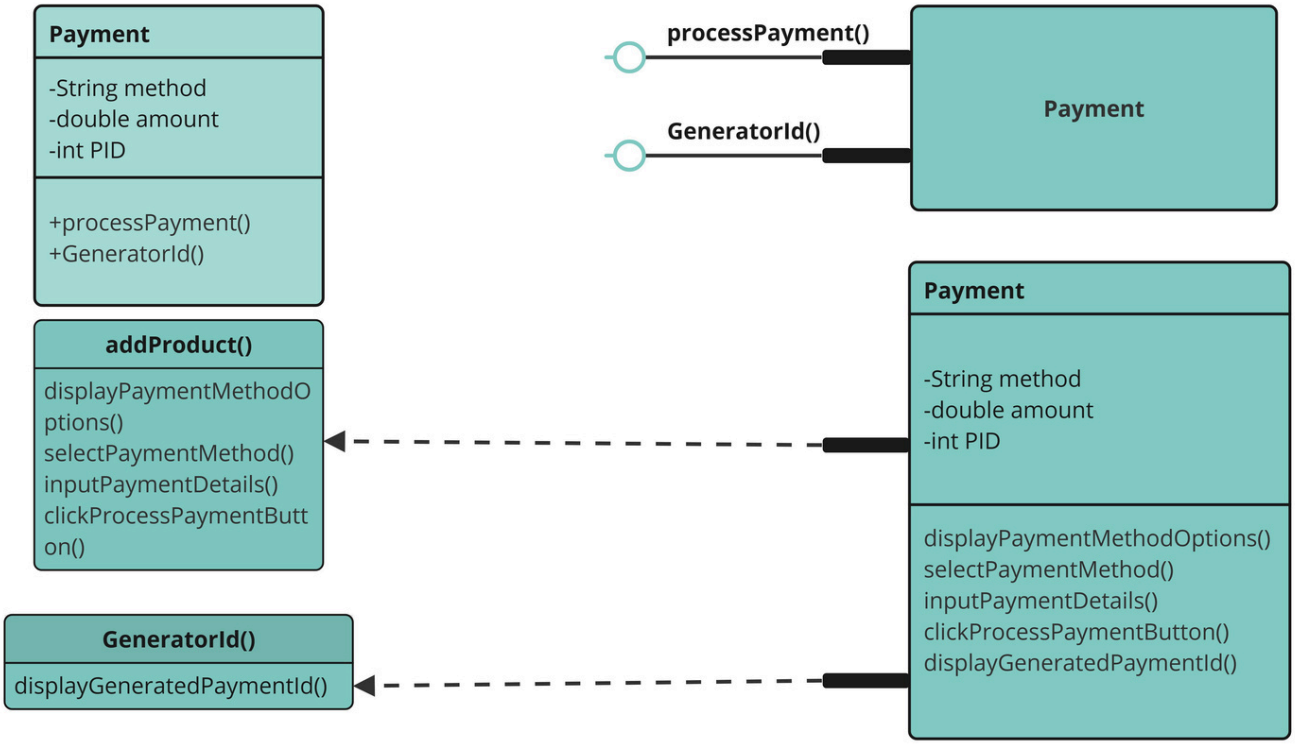
## AiChatBot:



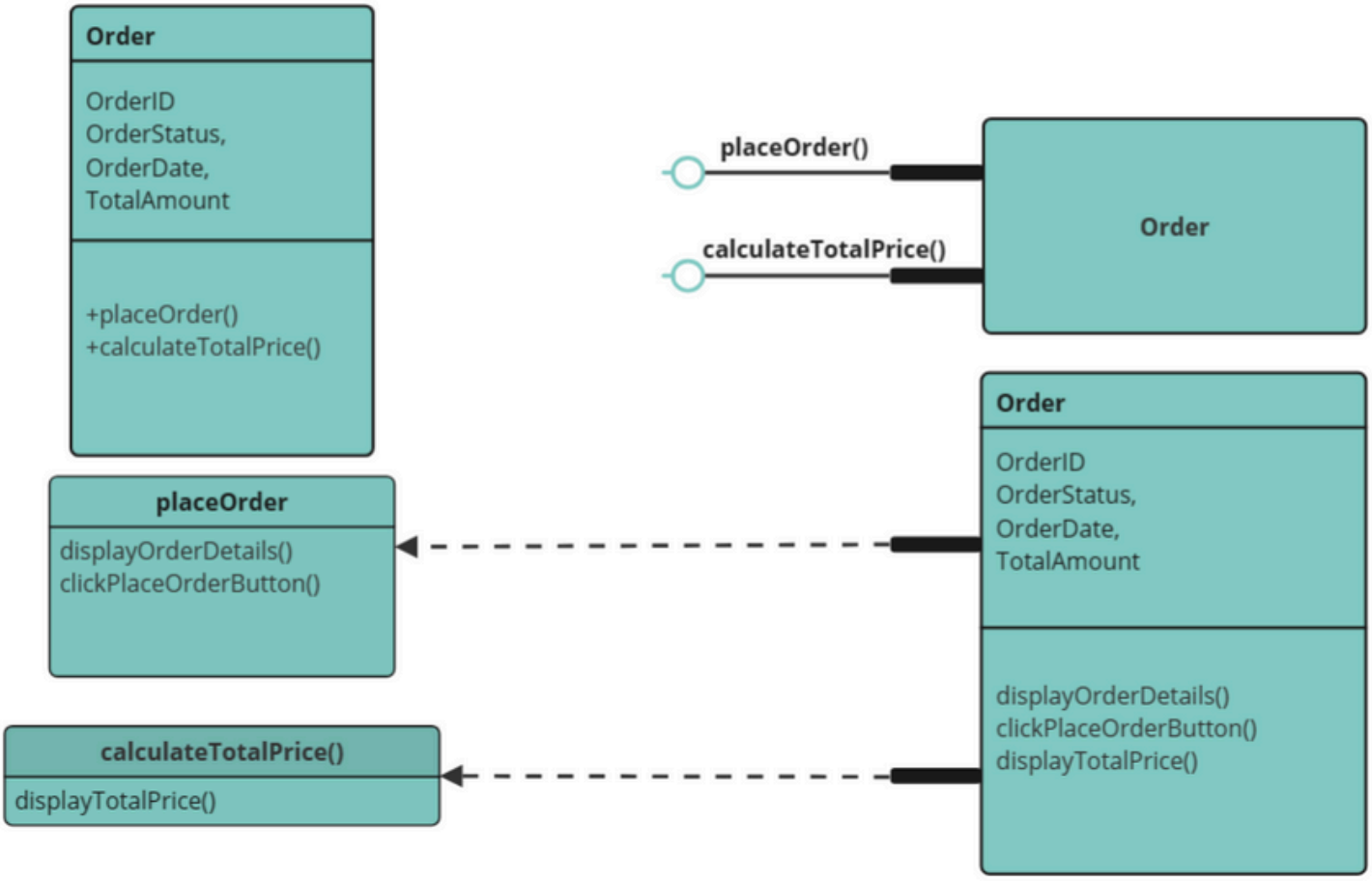

## Cart:



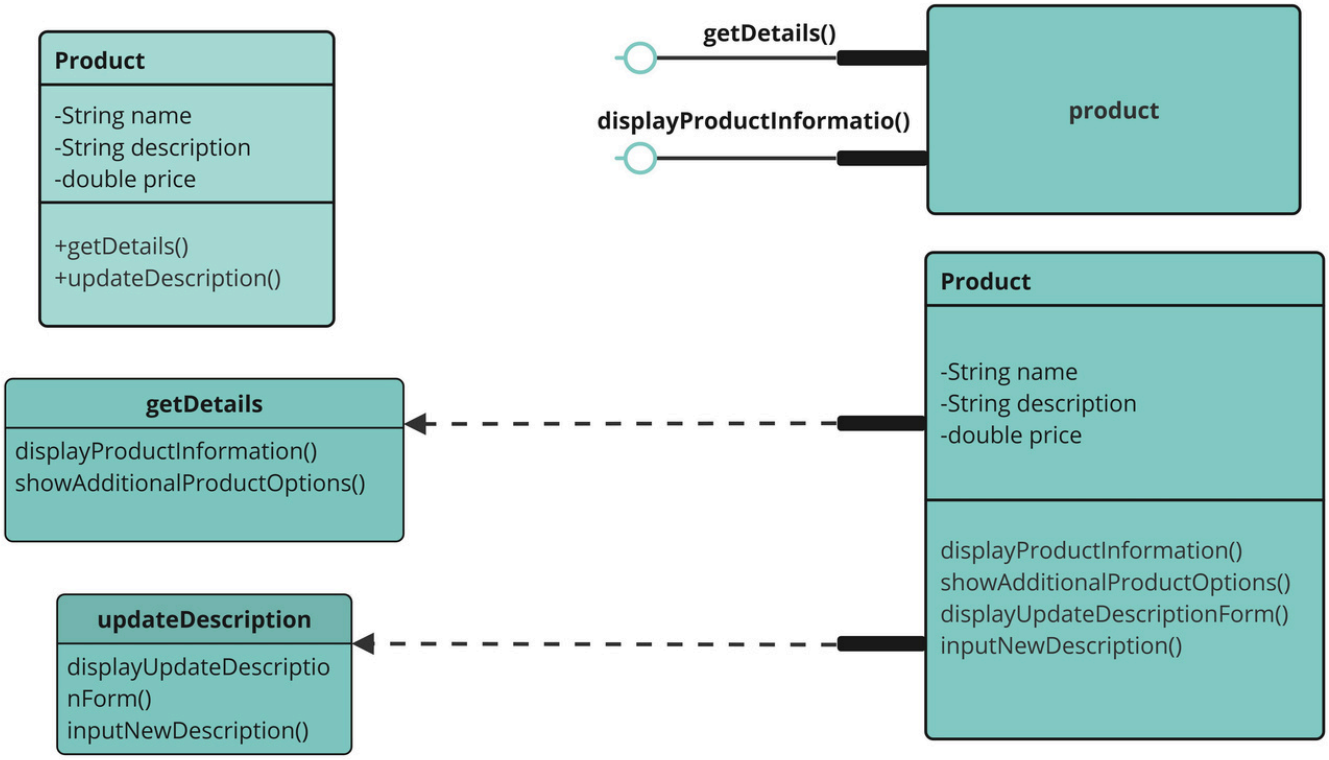
## Payment:



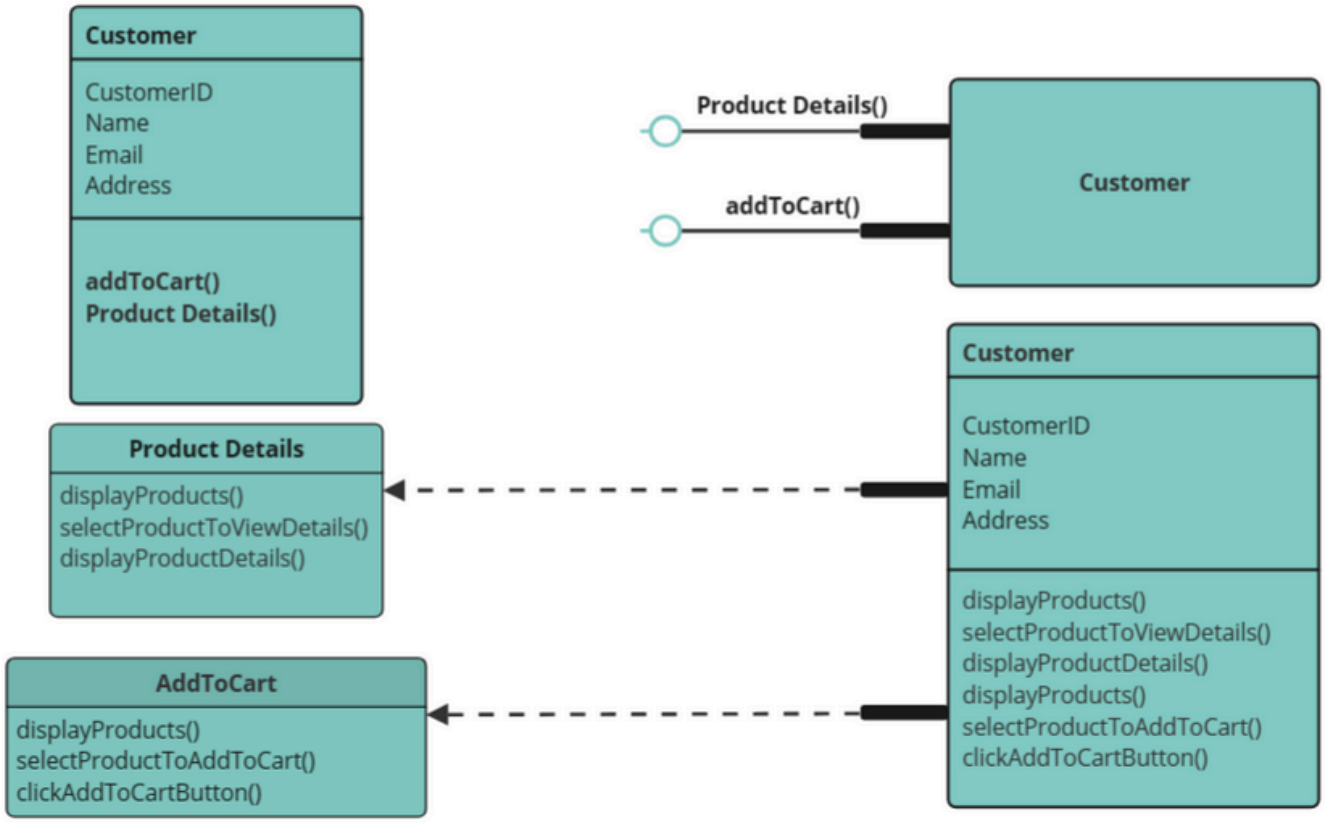

## Order:



# Product:

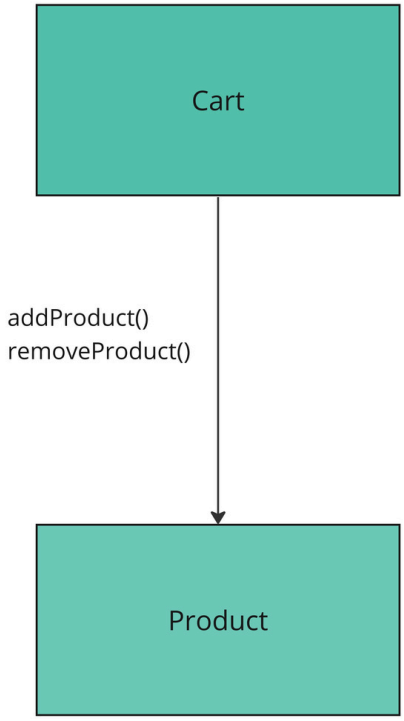



# Customer:

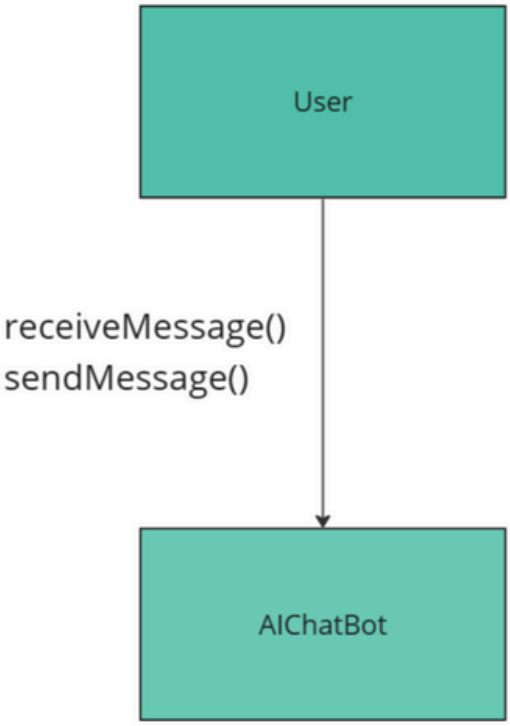


**Specific message details when classes or component collaborate:**

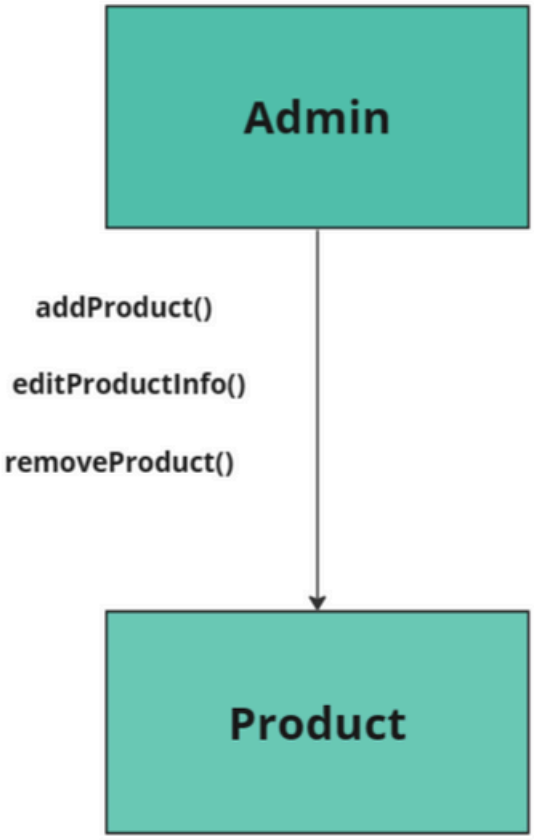
**cart-product**



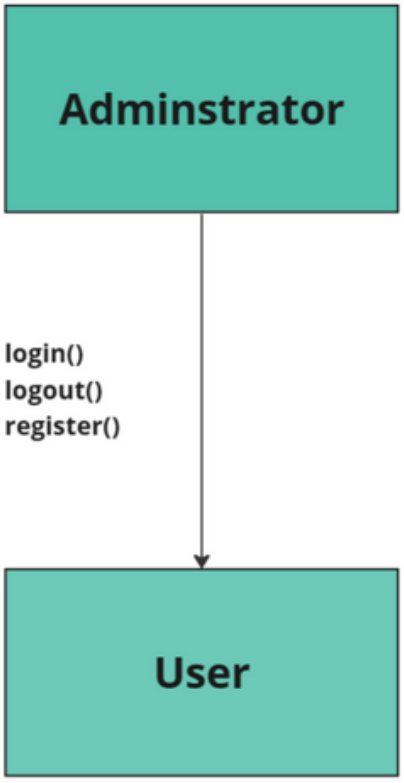

**User-AIChatBot**



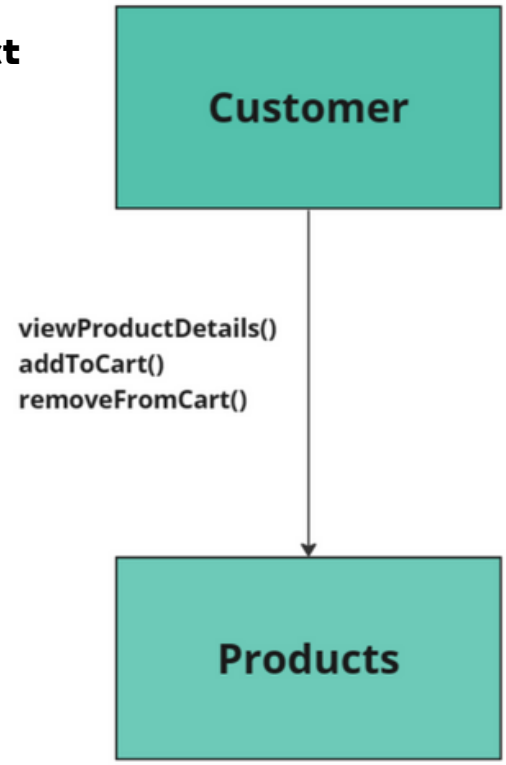
**Admin- Product**



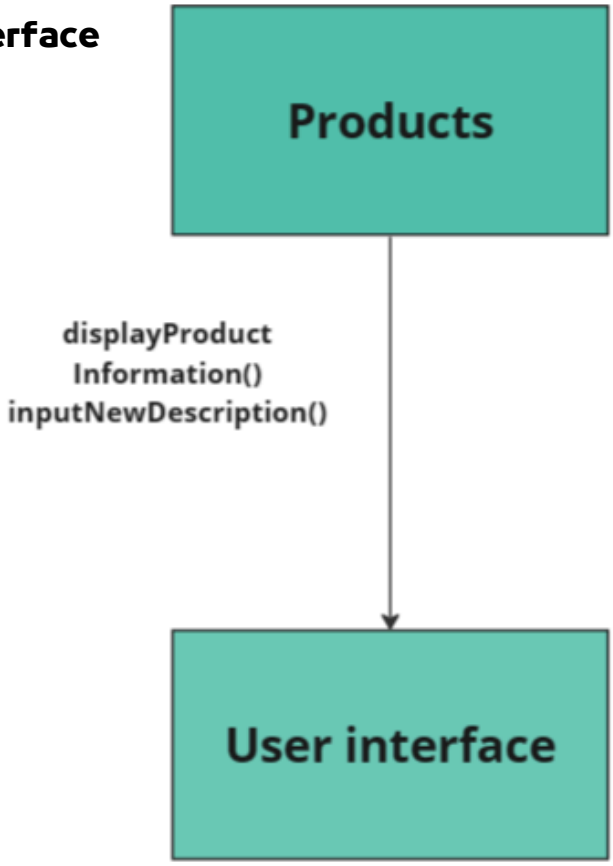

**Adminstartor-User**



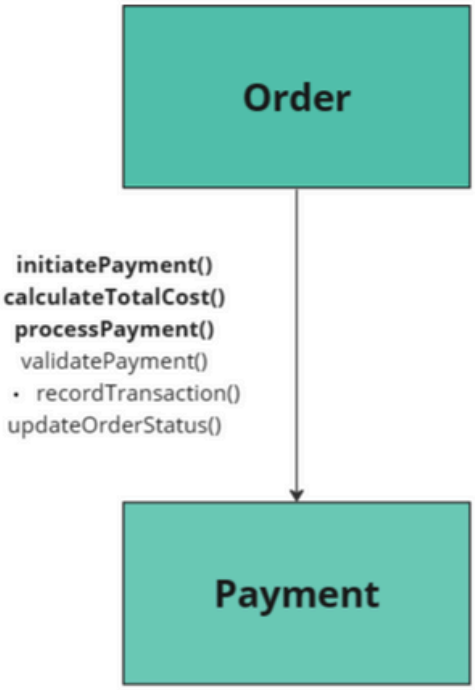
**customer-Product**



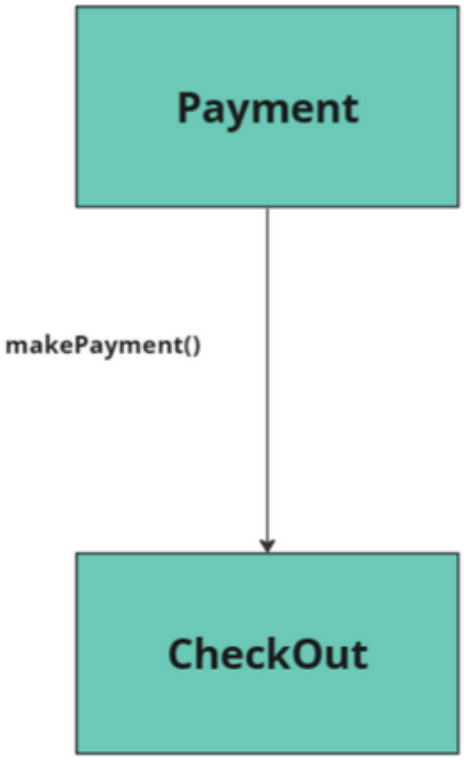

**Products-User interface**



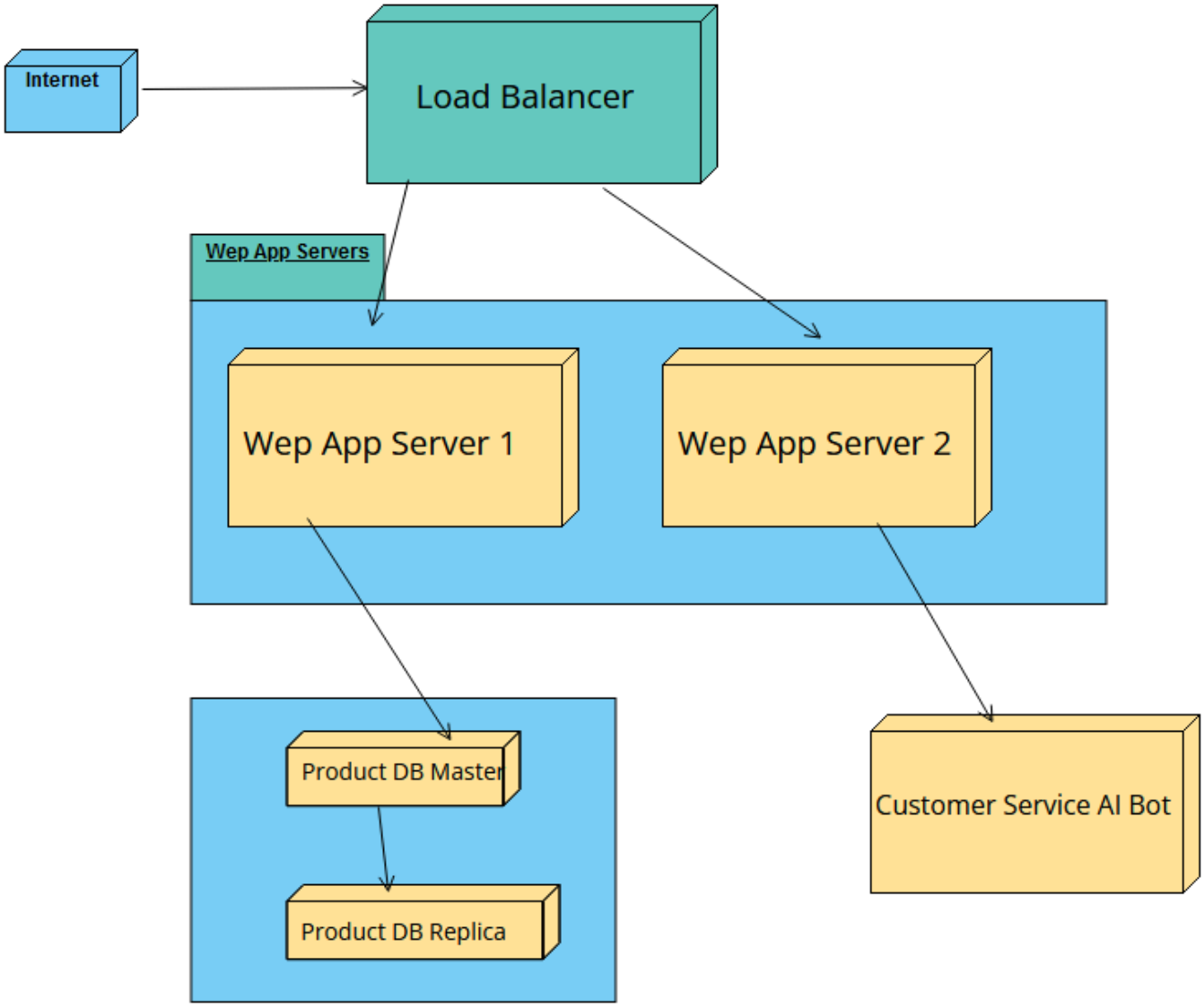
**Order-Payment**




**Payment-Checkout**

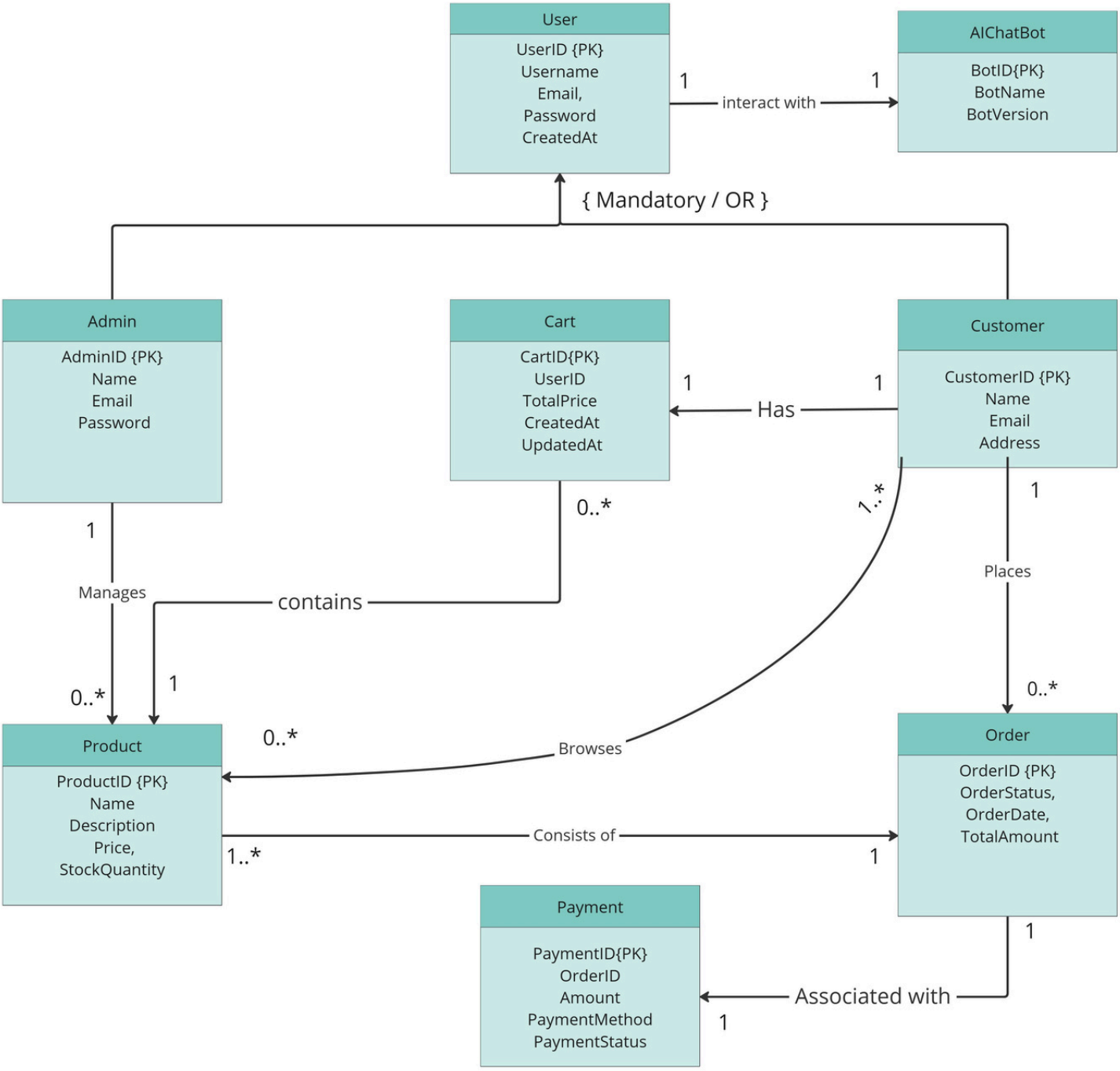


## Elaborate Deployment Diagrams to Provide Additional Implementation Detail

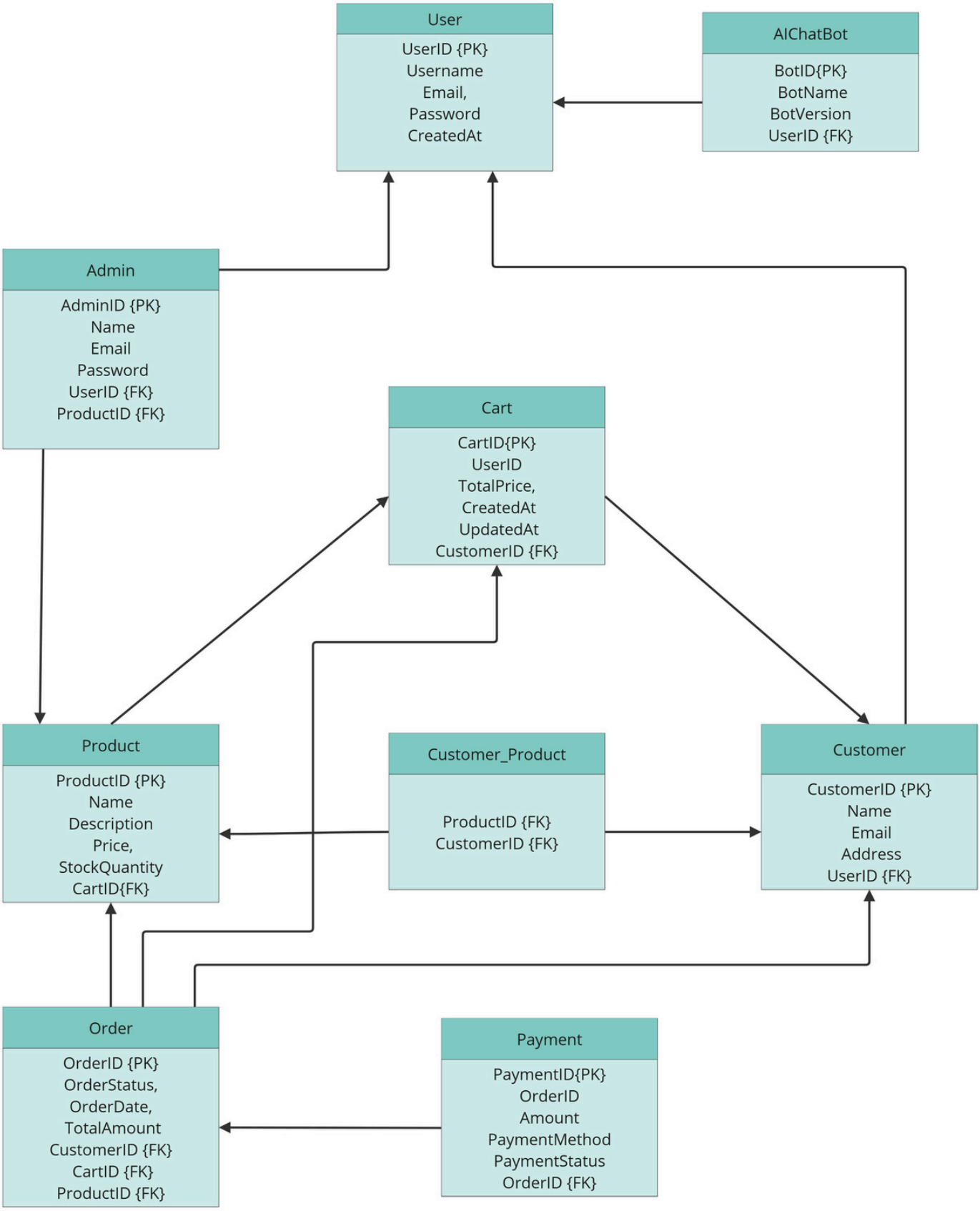


# Data base design:

## Conceptual Model:



# Logical Model




**View our full work using miro:**
**https://miro.com/welcomeonboard/Zk5KWVVsaEIGbDhmOGZ5WUd3cIJnaGNaZHpudXRNRTVYWlNibG5CQzMxTG5nNDZkYTIrYk1QNW9COVpzZHhiV3wzNDU4NzY0NTcwMDIwOTgyOTc1fDI=?share_link_id=297920958908**

## Relations Normalization Phase

**First Normal Form (1NF):**
Ensured atomicity by verifying that each table cell contains indivisible values.
Resolved any instances of composite or multivalued attributes.

**Second Normal Form (2NF):**
Confirmed that every non-prime attribute is fully functionally dependent on the primary key.
Addressed partial dependencies, restructuring tables to eliminate redundancy.

**Third Normal Form (3NF):**
Scrutinized the design to ensure non-transitive dependencies for every non-prime attribute.
Refactored tables with transitive dependencies, promoting data integrity.

Our commitment to normalization enhances the overall robustness and maintainability of the database. The resulting design adheres to best practices, minimizing duplication and providing a solid foundation for the E-commerce database.

## User

| Attribute | Data Type | Descriptio n |
|---|---|---|
| UserID | INT | Unique identifier for each user (Primary Key) |
| Username | VARCHAR | User's chosen username |
| Email | VARCHAR | User's email address |
| Password | VARCHAR | Hashed password for authenticati on |
| CreatedAt | DATETIME | Timestamp when the user account was created |

## Albot

| Attribute | Data Type | Description |
|---|---|---|
| BotID | INT | Unique identifier for each chatbot (Primary Key) |
| BotName | VARCHAR | Name of the chatbot |
| BotVersion | VARCHAR | Version information of the chatbot |

**Admin**

| Attribute | Data Type | Description |
|---|---|---|
| AdminID | INT | Unique identifier for each administrator (Primary Key) |
| Name | VARCHAR | Full name of the administrator |
| Email | VARCHAR | Email address of the administrator |
| Password | VARCHAR | Hashed password for authentication |

**customer**

| Attribute | Data Type | Description |
|---|---|---|
| CustomerID | INT | Unique identifier for each customer (Primary Key) |
| Name | VARCHAR | Customer's full name |
| Email | VARCHAR | Customer's email address |
| Address | VARCHAR | Customer's physical address |

## Cart

| Attribute | Data Type | Description |
|---|---|---|
| CartID | INT | Unique identifier for each cart (Primary Key) |
| UserID | INT | Foreign key linking to the User table |
| TotalPrice | DECIMAL | Total price of all products in the cart |
| CreatedAt | DATETIME | Timestamp when the cart was created |
| UpdatedAt | DATETIME | Timestamp of the last update to the cart |

## Product

| Attribute | Data Type | Description |
|---|---|---|
| ProductID | INT | Unique identifier for each product (Primary Key) |
| Name | VARCHAR | Name of the product |
| Description | VARCHAR | Detailed description of the product |
| Price | DECIMAL | Selling price of the product |
| StockQuantity | INT | Number of units available in stock |

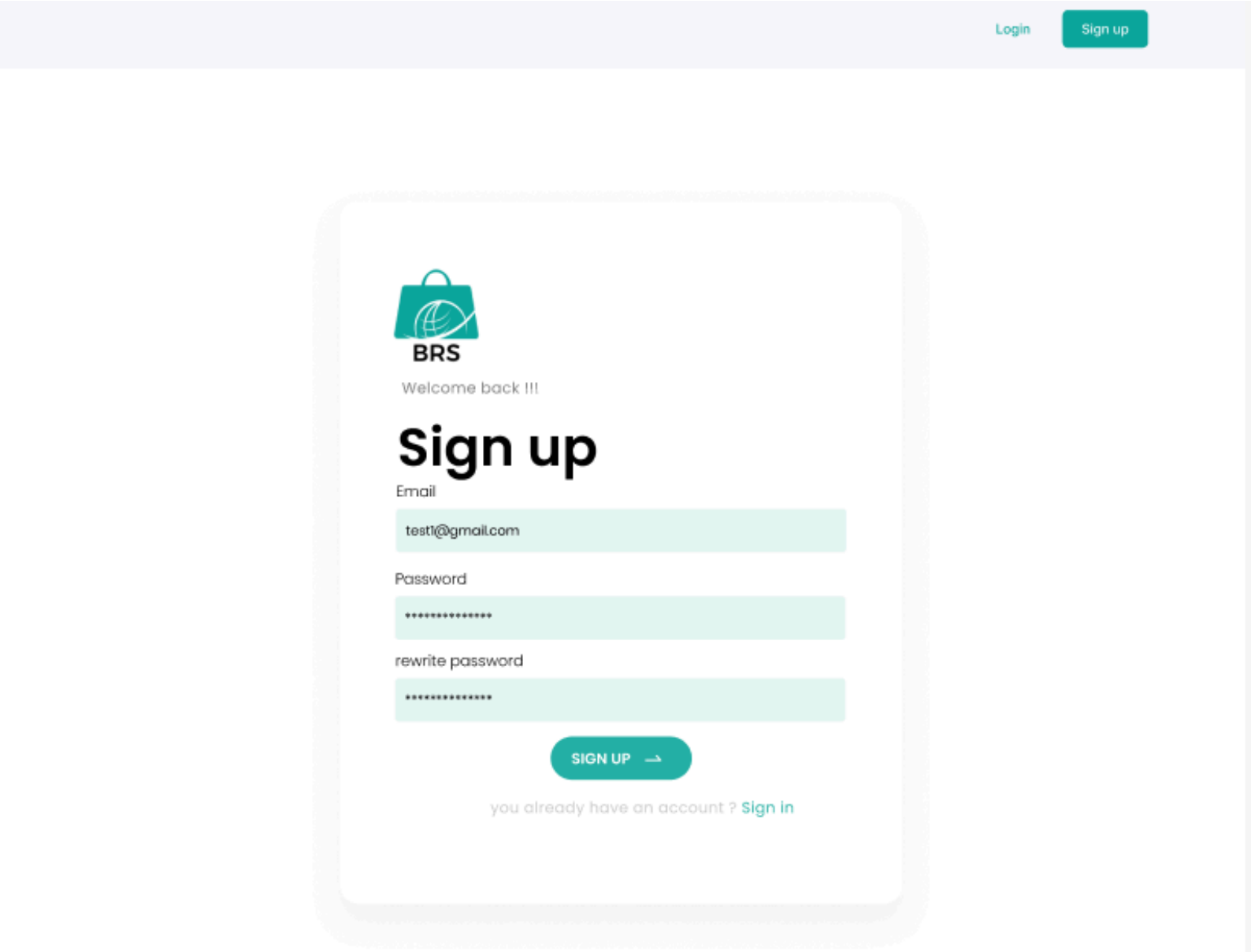
**Order**

| Attribute | Data Type | Description |
|---|---|---|
| OrderID | INT | Unique identifier for each order (Primary Key) |
| OrderStatus | VARCHAR | Current status of the order |
| OrderDate | DATETIME | Date on which the order was placed |
| TotalAmount | DECIMAL | Total amount charged for the order |

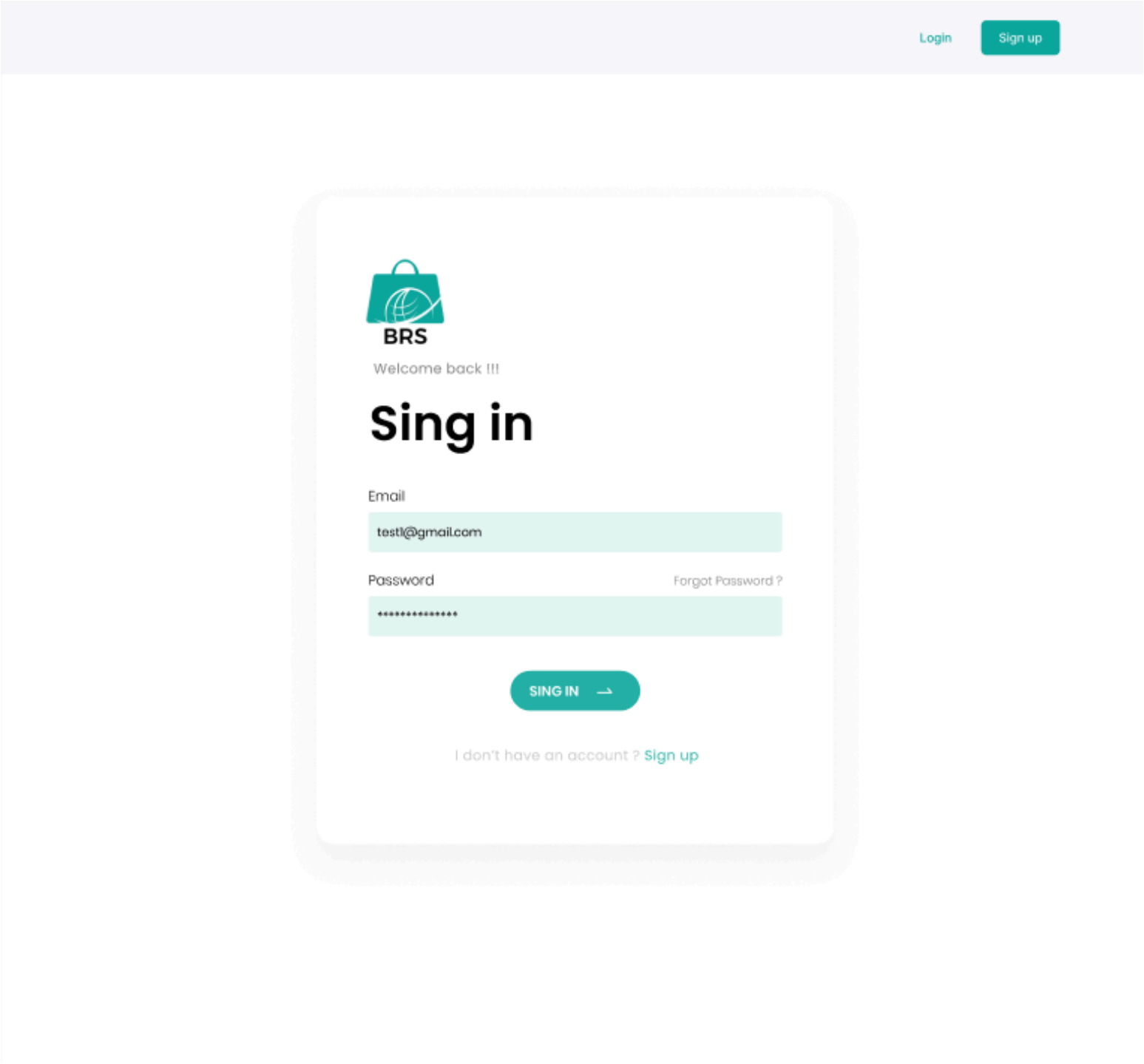
**Payment**

| Attribute | Data Type | Description |
|---|---|---|
| PaymentID | INT | Unique identifier for each payment (Primary Key) |
| OrderID | INT | Foreign key linking to the Order table |
| Amount | DECIMAL | Amount paid |
| PaymentMethod | VARCHAR | Method of payment used |
| PaymentStatus | VARCHAR | Status of the payment |

## sign up page:

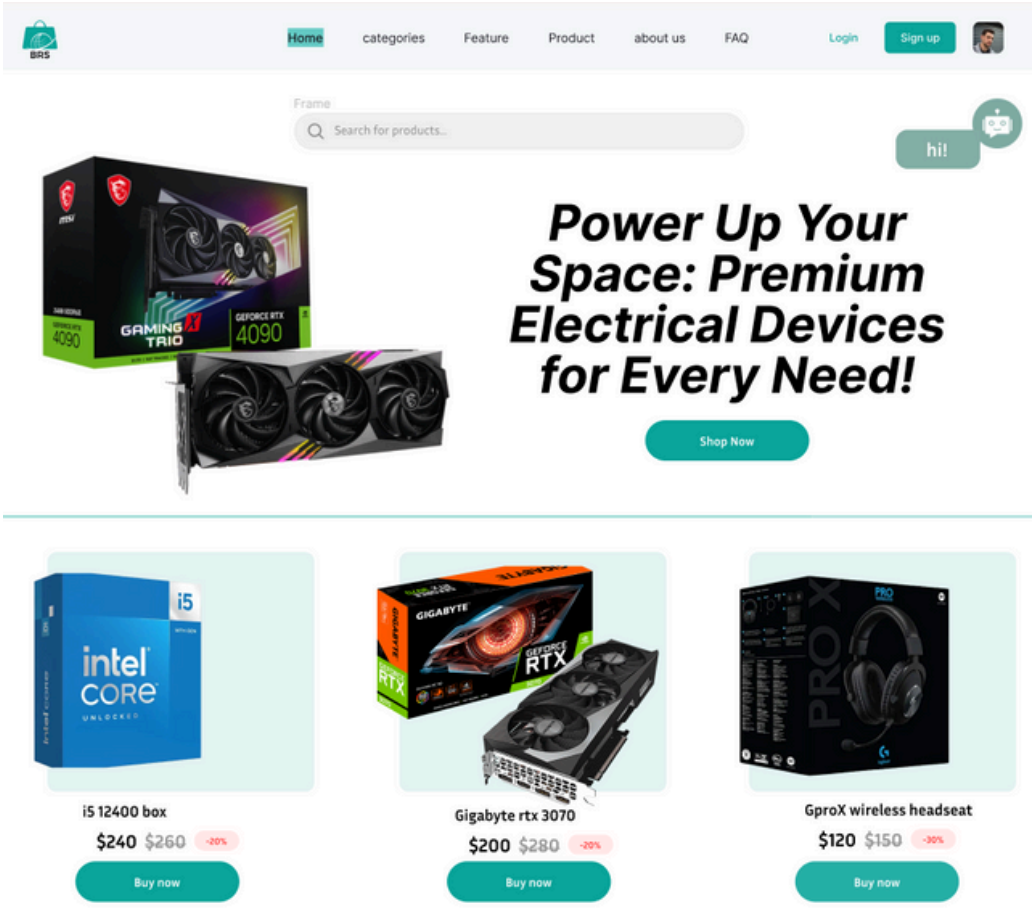

Login
Sign up
BRS
Welcome back !!!
Sign up
Email
test1@gmail.com
Password
rewrite password
SIGN UP
you already have an account ? Sign in

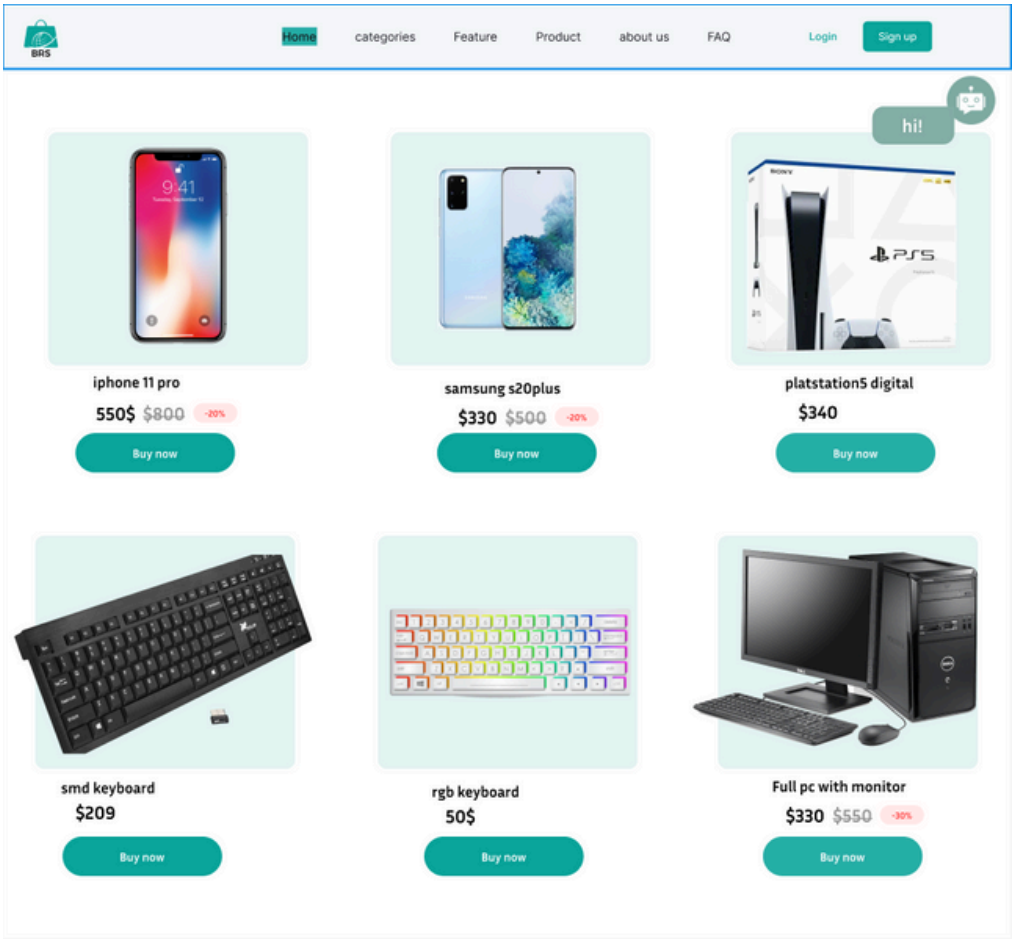
## sign in page:


Login
Sign up
BRS
Welcome back !!!
Sing in
Email
testl@gmail.com
Password
Forgot Password ?
SING IN
I don't have an account ? Sign up

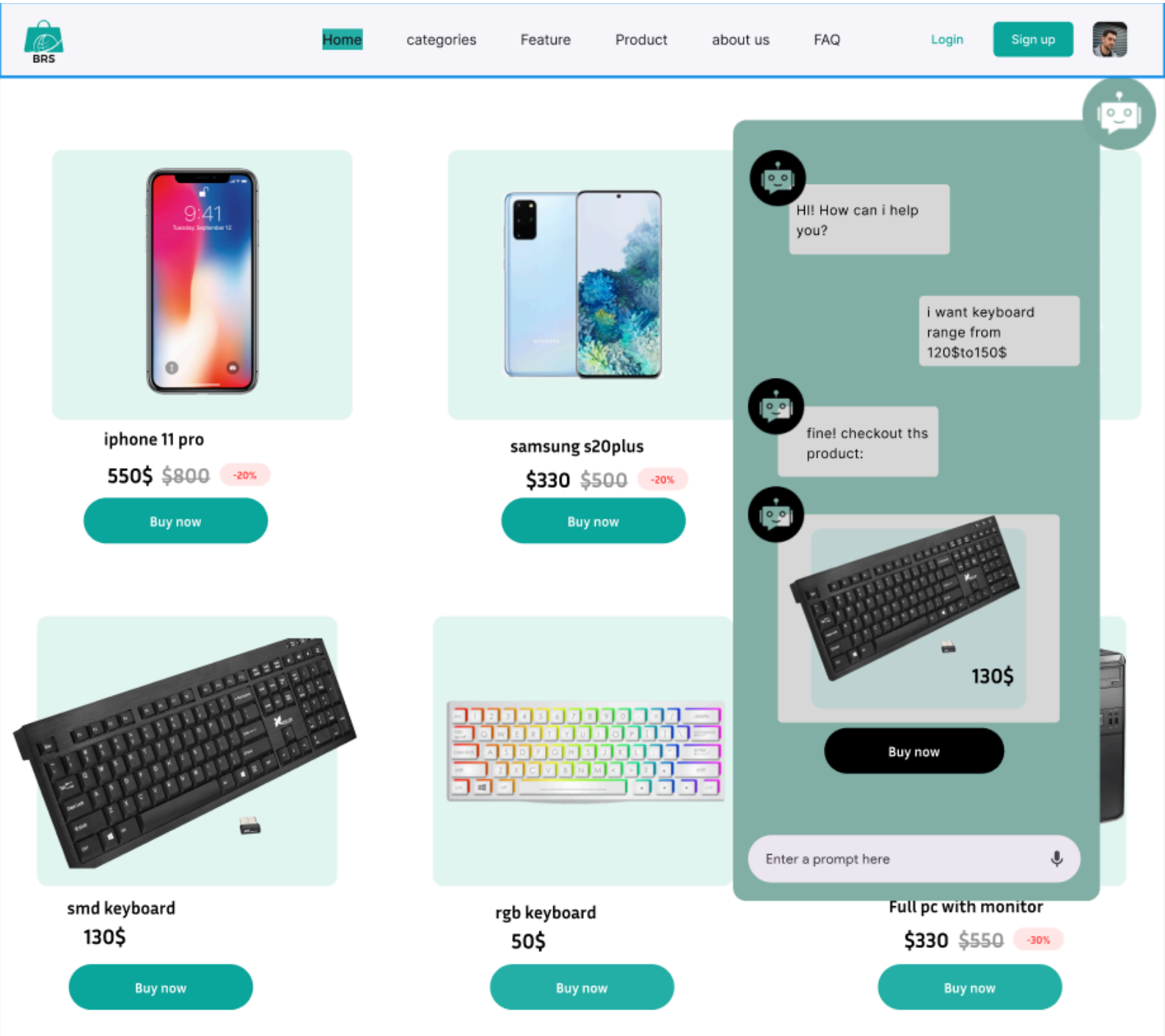
# User interface screens

## Home page:

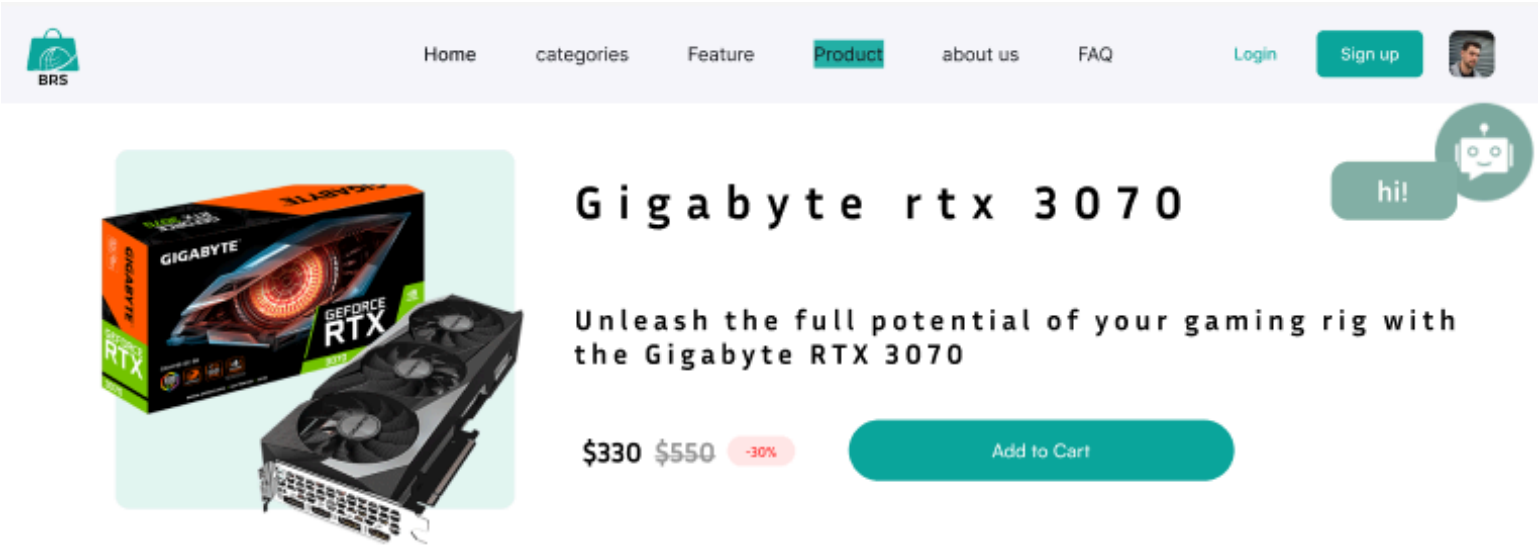


# Ai bot:


BRS
Home
categories
Feature
Product
about us
FAQ
Login
Sign up
iphone 11 pro
550$ $800 -20%
Buy now
samsung s20plus
$330 $500 -20%
Buy now
HI! How can i help you?
i want keyboard range from 120$to150$
fine! checkout ths product:
130$
Buy now
Enter a prompt here
smd keyboard
130$
Buy now
rgb keyboard
50$
Buy now
Full pc with monitor
$330 $550 -30%
Buy now

# product page:




specifications:

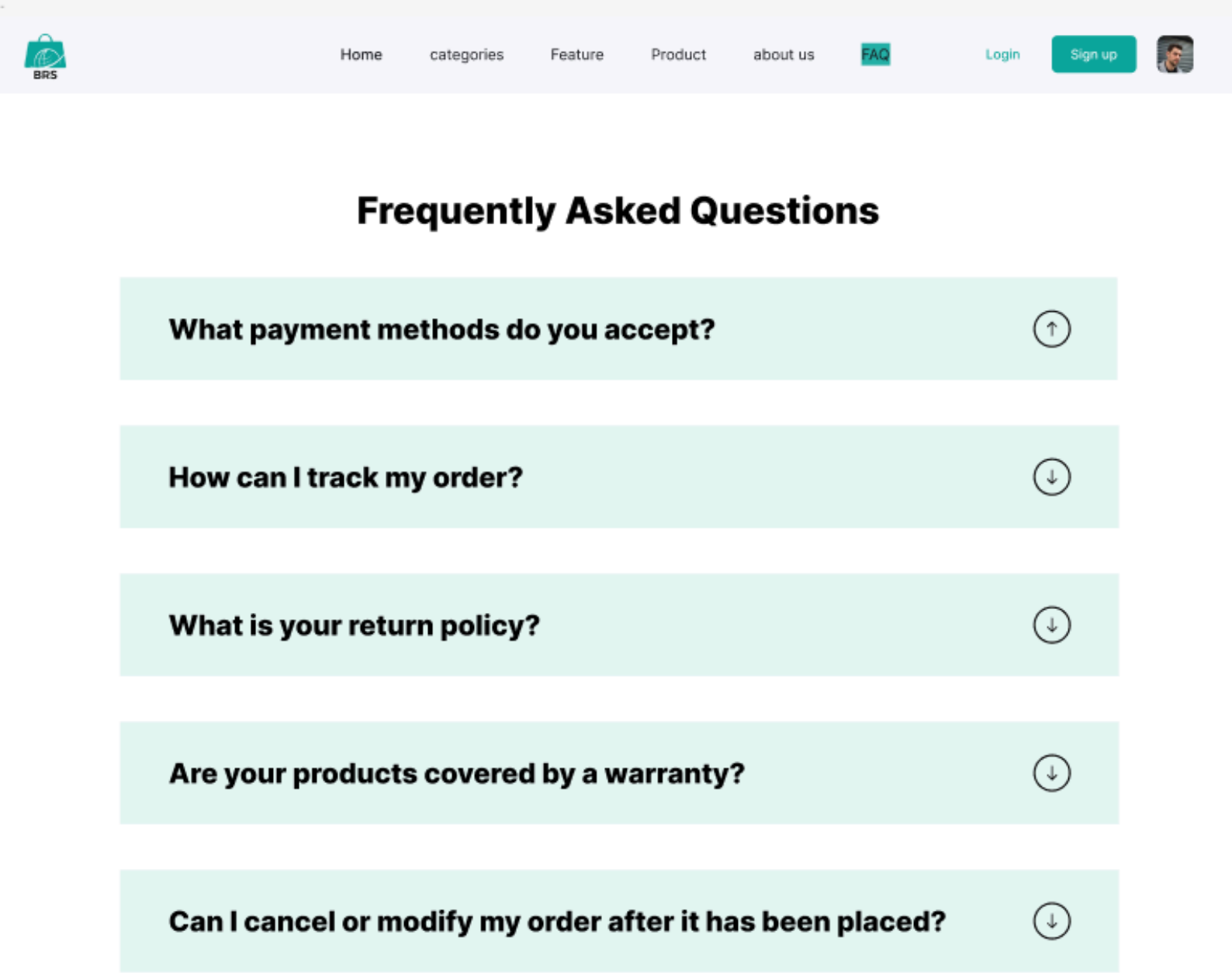
- GPU Architecture: NVIDIA Ampere
- CUDA Cores: 5888
- Boost Clock: Up to [Insert Boost Clock Speed] MHz
- Memory Size: 8GB GDDR6
- Memory Bus: 256-bit
- Memory Speed: 14 Gbps
- Output: 3 x DisplayPort 1.4a, 2 x HDMI 2.1
- Max Resolution: 7680x4320 @60Hz
- DirectX Support: DirectX 12 Ultimate
- Ray Tracing Cores: Yes
- Tensor Cores: Yes
- VR Ready: Yes
- Recommended Power Supply: 650W
- Power Connector: 8-pin x 2
- Dimensions: [Insert Dimensions]
- Cooling: WINDFORCE 3X Cooling System
- RGB Fusion 2.0: Customizable RGB lighting
- Protection: Metal Back Plate

Features:

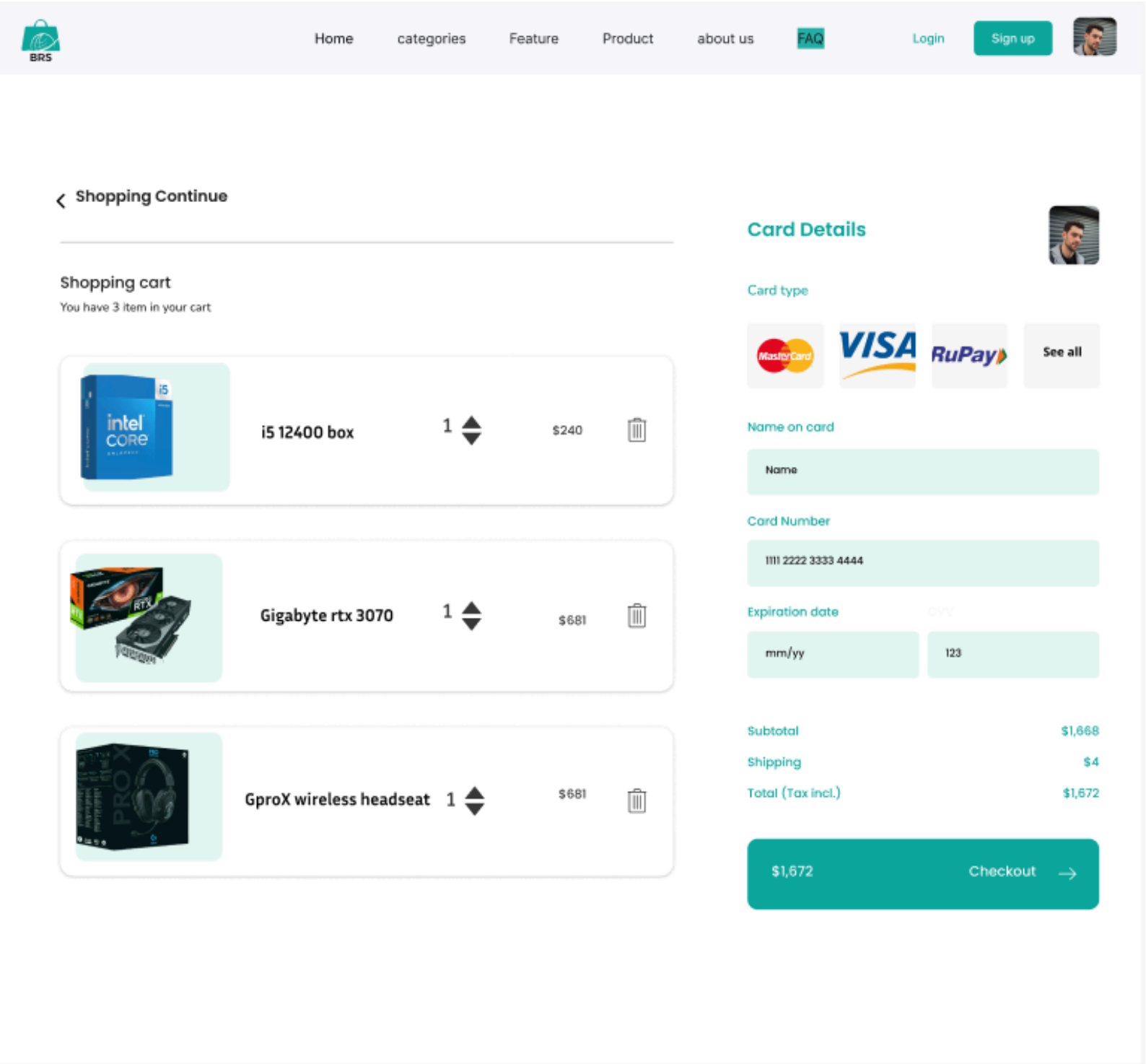
1. NVIDIA Ampere Architecture: Experience next-gen gaming with NVIDIA's Ampere architecture, delivering improved performance and efficiency.
2. Ray Tracing and AI: Real-time ray tracing and AI-enhanced graphics bring lifelike lighting, shadows, and reflections to your favorite games.
3. WINDFORCE 3X Cooling System: Keep your GPU cool and quiet even during intense gaming sessions with Gigabyte's WINDFORCE 3X cooling system.
4. RGB Fusion 2.0: Customize your gaming rig with Gigabyte's RGB Fusion 2.0, allowing you to synchronize lighting effects with other RGB-enabled components.

E-commerce

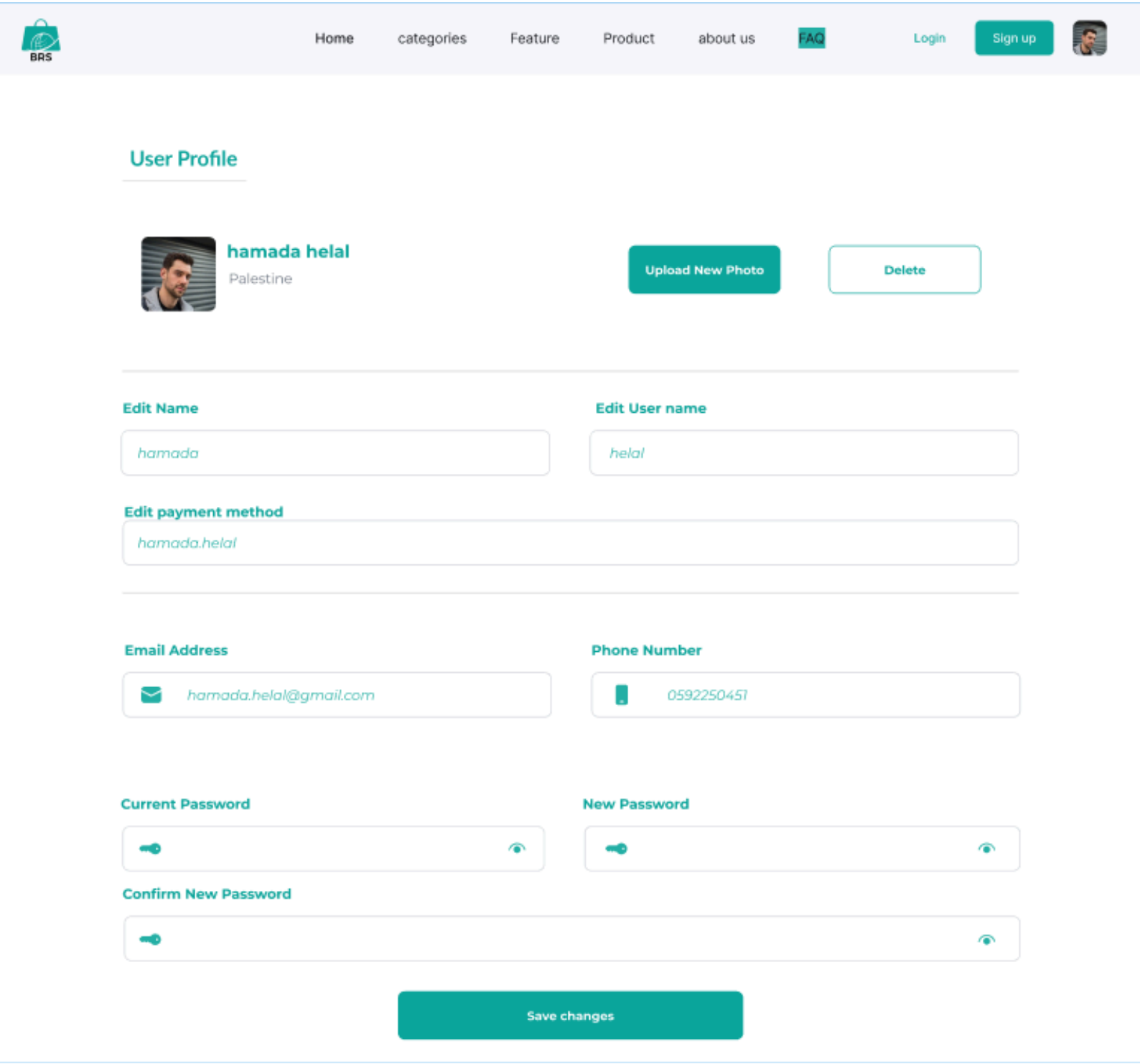
## FAQ page:


BRS
Home
categories
Feature
Product
about us
FAQ
Login
Sign up
Frequently Asked Questions
What payment methods do you accept?
How can I track my order?
What is your return policy?
Are your products covered by a warranty?
Can I cancel or modify my order after it has been placed?

# Cart page:


BRS
Home
categories
Feature
Product
about us
FAQ
Login
Sign up
Shopping Continue
Shopping cart
You have 3 item in your cart
i5 12400 box
1
$240
Gigabyte rtx 3070
1
$681
GproX wireless headseat
1
$681
Card Details
Card type
See all
Name on card
Name
Card Number
1111 2222 3333 4444
Expiration date
mm/yy
123
Subtotal
$1,668
Shipping
$4
Total (Tax incl.)
$1,672
$1,672
Checkout

# Profile page:




**View our full work using Figma:**
**https://www.figma.com/file/sL6QcOLq9F9ngpvTEJveBO/e-commerce-WEBSITE?type=design&node-id=0%3A1&mode=design&t=ZTg32yF6Cd55oDQR-1**