SCHOOL OF
INFORMATION TECHNOLOGY
& COMPUTER SCIENCE

Nile University
جامعة النيل الأهلية

Project CSCI313

Team members:

Sara Onsy_202001706

Ranim Moustafa_202000515

Rana Nasser_202000824

Salma Sameh_202002800

Amira Hikal_202000448

DR: Radwa Mohamed ENG: Hossam Eldin Mostafa
Section :4

Purpose of the project:

The major goal of this project is to create a platform that will allow buyers and sellers to connect online. These initiatives or websites are used to market things and make money from them. Online shopping is a very well-liked web development project. It is mostly employed as a business model to generate revenue. All users can browse the products, but to add any to their shopping carts for purchase, they must first log in and then register. Before purchasing a product, users or consumers can read reviews that other users have left about it. Users are free to add as many items as they like to their shopping cart. The quantity of each item that has been put to the cart can thereafter be set by the users. Finally, consumers can enter their address and preferred payment method at checkout and submit the appropriate order. The administrator can then view the customer details using his/her order details and the address where the order should be delivered.

Features:

- Register and login.
- Session of each user is saved using cookies.
- All Products with their view, key features, and reviews by other users.
- Users can search for any product.
- Cart Functionality.
- Proper Order Summary Before Placing Order.
- Change Password Facility.
- Contact Us.

Scope of the Project:

Customer model: The customer can register to the shop online and can save some information about the customer.

Product model: Admin can add new products very easily in the shop and can save the product.

Model Feature: The admin can select the product and write any features about it. All features of that product will be visible to users when viewing a particular product.

Review form: All customers can write a review about the product which customers can read before purchasing it.

Application form: It stores the customer's order details, especially the order ID.

Item order form: It stores the customer's order ID from the order form and the products in quantity.

Checkout details form: It basically stores the exact address to which the order will be delivered.

Create a super user: After creating the forms, we need to go to the admin panel to access the created forms. Hence, we need a privileged user who can access the forms from the admin panel. The super user can make any changes within the forms.

Technologies Used:

HTML, CSS, JavaScript, Bootstrap, and Python Django are commonly used in the development of AI shopping systems. Here's a breakdown of their usage:

1. **HTML (Hypertext Markup Language):** HTML is used for structuring the content and layout of web pages in an AI shopping system. It defines the elements and their placement, including headings, paragraphs, images, buttons, and forms.

2. **CSS (Cascading Style Sheets):** CSS is used to style and customize the visual appearance of web pages in an AI shopping system. It can be used to

define colors, fonts, spacing, borders, and other visual properties to create an attractive and consistent user interface.

3. **JavaScript:** JavaScript is a programming language that adds interactivity and dynamic functionality to web pages. In an AI shopping system, JavaScript is used for tasks such as client-side form validation, handling user interactions, implementing product filters, and making asynchronous requests to the server for Realtime updates.
4. **Bootstrap**: Bootstrap is a popular front-end framework that provides pre-designed templates, components, and CSS styles. It helps in creating responsive and mobile-friendly user interfaces quickly. In an AI shopping system, Bootstrap can be used to build consistent layouts, responsive navigation menus, and grid systems.
5. **Python Django**: Python Django is a web framework that simplifies the development of complex web applications. In an AI shopping system, Django can be used as the backend framework to handle server-side logic, manage databases, process payments, and provide APIs for communication between the front-end and the server. Django's robustness and scalability make it suitable for building AI shopping systems that handle large amounts of data and user interactions.

Overall, HTML, CSS, JavaScript, Bootstrap, and Python Django are used together to create a seamless and interactive user experience in an AI shopping system. HTML and CSS handle the structure and appearance of web pages, JavaScript adds interactivity, Bootstrap ensures responsiveness, and Python Django manages the backend functionality and data processing.

Intended Audience:

The target audience for AI Procurement System software documentation for AI Procurement Systems typically includes the following: Developers: Developers responsible for installation, configuration, and maintenance Maintaining an AI procurement system will consult documentation for technical guidance, system requirements, and API documentation. System Administrator: The system administrator manages the deployment, security, and overall operations of the AI system. Procurement will use documentation to understand the system architecture, installation step settings, configuration details, and user account management.

Users: AI Shopping System users who want to understand how to navigate the system, browse products, search for items, add products to their cart, complete the checkout process, and track their orders will refer to the user guide. documents section.

Business Stakeholders: Business stakeholders want to understand the features and capabilities of an AI purchasing system, as well as its potential benefits and impact on the business, you can refer to the documentation to better understand the system.

Support Teams: Support teams responsible for responding to user requests and resolving problems can use the documentation to find solutions to common problems, FAQs, and related information. contact for further support. It is important that documentation meets the technical needs of developers and system administrators while providing clear instructions and guidance to users without technical expertise. Documentation must be complete, well-structured, and accessible to all target audiences.

Overview of the document:

The project aims to create an online platform that connects buyers and sellers, allowing users to browse and purchase products. It includes features such as user registration, product browsing, search functionality, cart functionality, order summary, password change facility, and contact options. The scope includes customer and product models, review forms, application forms, item order forms, and checkout details forms. Super user is created for easy access and modification.

The project uses HTML, CSS, JavaScript, Bootstrap, and Python Django for content structuring, styling, interactivity, and backend framework. The documentation is intended for developers, system administrators, users, business stakeholders, and support teams. Developers and administrators will use it for technical guidance, while users will use it to navigate the system, understand features, benefits, and impact on the business. The documentation should be complete, well-structured, and accessible to all target audiences.

Overall Description

2.1 Product Perspective Perspective: In this system, all users can browse a wide range of products. If a user is interested in purchasing any of these products, they are required to first register an account and then log in. Once logged in, they can add desired products to their shopping cart. Furthermore, users or customers have the option to peruse reviews left by other users about a particular product before making a purchasing decision. This way, they can make more informed choices. Users are not limited in the number of products they can add to their shopping cart; they can select as many as they desire. Additionally, they have the flexibility to specify the quantity of each product they've added to their cart. When they are ready to complete their purchase, users proceed to the checkout process. During this step, they provide their shipping address and select a preferred mode of payment. Once these details are submitted, they can place their order. At this point, the administrative team gains access to the customer's information, which includes their order details. This information assists the team in processing and fulfilling the order. The delivery address is also made available to ensure the successful and accurate delivery of the order to the customer. This system ensures a smooth and secure shopping experience for users, with transparency and ease of management for the admin.

2.1.1. Product Function

Customer model: The customer can register to the shop online and can save some information about the customer.

Product model: Admin can add new product very easily in the shop and can save the product Model.

Feature: The admin can select the product and write any features about it. All features of that product will be visible to users when viewing a particular product.

Review form: All customers can write a review about the product which customer.

Application form: It stores the customer's order details, especially the order ID.

Item order form: It stores the customer's order ID from the order form and the products in quantity.

Checkout details form: It basically stores the exact address to which the order will be delivered.

Create a super user: After creating the forms, we need to go to the admin panel to access the created forms. Hence, we need a privileged user who can access the forms from the admin panel. The super user can make any changes within the forms.

2.1. Function:

Functional Requirements for Software Used in Online Shopping:

2.1.1 Functional Requirements

Title: User registration and authentication.

Description: It should be possible for users to securely register and log in.

The system must verify users' credentials and allow access according to their responsibilities (administrator, customer, etc.).

2.1.2 Functional Requirements

Title: Product Catalo

Description: A searchable and well-organized product catalog needs to be provided by the program.

Every product needs a special number, name, description, cost, availability information, and pertinent photos.

2.1.3 Functional Requirements

Title: Product Search and Filtering

Description: Clients must have the ability to look for items by using categories, keywords, or filters like brand, price range, or ratings.

The way the search results are presented should be user-friendly and relevant.

2.1.4 Functional Requirements

Title: Shopping Cart

Description: It should be possible for users to add products to an online shopping basket.

Users should be able to change the quantity or take products out of the cart using the program.

2.1.5 Functional Requirements

Title: Payment and Checkout

Description: Customers should be able to enter their billing and shipping information and continue to the checkout.

Users' personal data should be protected, and the payment procedure should be safe.

2.1.2. No-Function:

Performance: There should be no appreciable lag or degradation in performance when numerous users browse products, add items to their shopping carts, and place orders at the same time on the system.

Scalability: To support expanding product catalogs and rising user traffic, the system should be built to scale horizontally. It ought to be able to accommodate more orders, products, and users without degrading user experience or performance.

Reliability: The system needs to be extremely dependable to guarantee little downtime and a low likelihood of failure. It ought to have systems in place to recover from mistakes, handle errors with grace, and guard against data loss.

Security: The protection of user information and transactions must be the system's top priority. It ought to use safe authorization and authentication.

Usability: Users should be able to browse products, add items to their shopping carts, read reviews, and finish the checkout process with ease thanks to the system's user friendly and intuitive interface. Throughout the purchasing process, users should be guided by clear instructions, error messages, and feedback.

Interoperability: To speed up the checkout and order fulfillment procedures, the system should be able to interface with other systems, such as shipping companies and payment gateways. Standard data formats and APIs should be supported to facilitate seamless data interchange with external systems.

Privacy and Compliance: The system needs to safeguard user data and abide by privacy laws. Users should be able to control their privacy settings with options like deleting their accounts or choosing not to share their data. The system must also abide by all applicable legal requirements for data protection and e-commerce.

Multi-Language Support: To accommodate a varied user base, the system ought to support several languages. To support users from various regions and language preferences, it should offer language options for product information, customer support, and user interface.

Data Backup and Recovery: To guard against data loss or system failures, the system should have strong data backup and recovery procedures in place. Critical data should be regularly backed up, and procedures for restoring data in the event of a failure or disaster should be in place.

Integration with Marketing Tools: To support customer engagement programs, tailored promotions, and targeted marketing campaigns, the system can integrate with marketing tools like email marketing platforms and advertising platforms. It ought to make data sharing and automation between marketing tools and the shopping system easier.

Offline Mode: When a user is not online, the system may offer an offline mode or cache functionality that enables them to browse and use specific features. This can involve accessing saved shopping carts, checking order histories, or viewing previously loaded product information.

Integration with Social Login: Users may choose to register or log in using their Google or Facebook accounts, among other social media accounts, through the system. This integration lowers friction during the sign-up process and streamlines the authentication process for users.

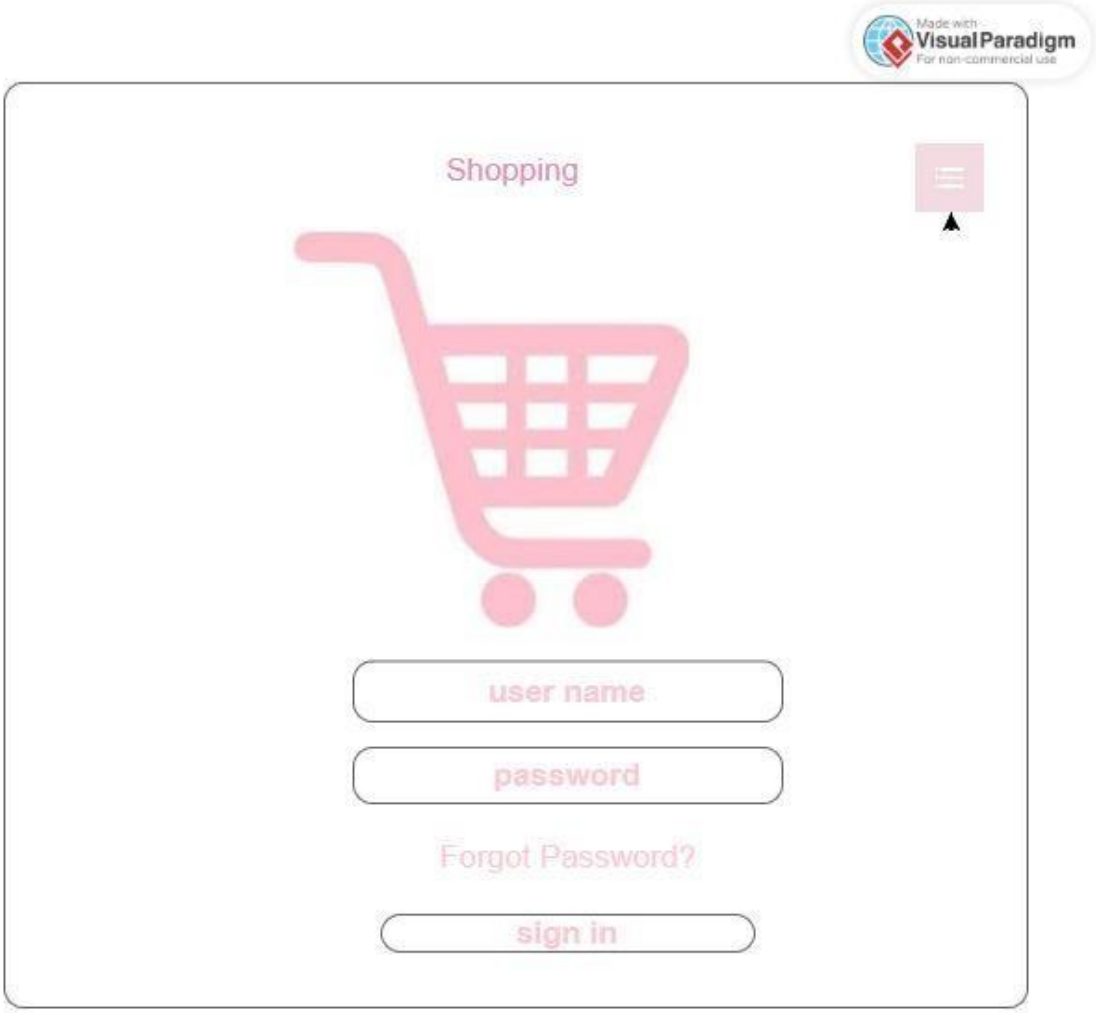
Integration with Chatbots and Virtual Assistants: To offer automated customer service or product recommendations, the system can be integrated with chatbots and virtual assistants. These AI-powered chatbots can offer individualized support, help with product selection, and respond to frequently asked questions from customers.

Mobile Payment Integration: To give users easy and safe ways to make payments, the system can be integrated with mobile payment services like digital wallets or mobile payment apps. This integration can improve user satisfaction by streamlining the checkout procedure.

Interfaces

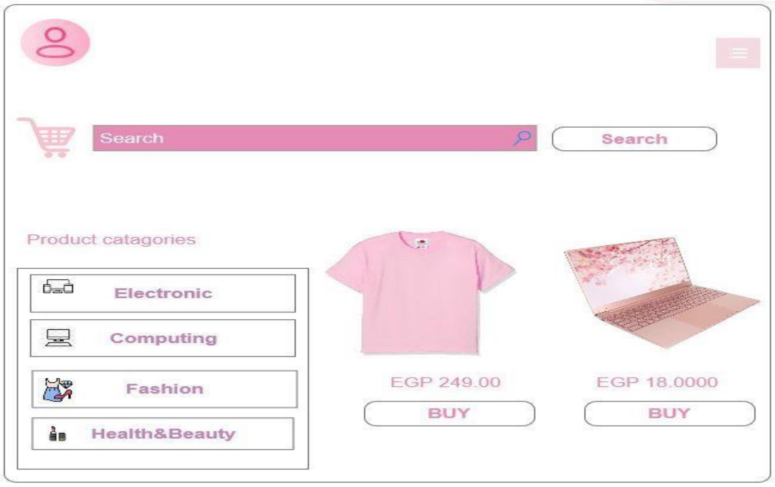
System interface:

Home Page Interface: The home page interface with its login functionality prioritizes simplicity, clarity, and ease of use. It aims to provide a secure and efficient entry point for users to access their accounts and begin utilizing the platform's features.

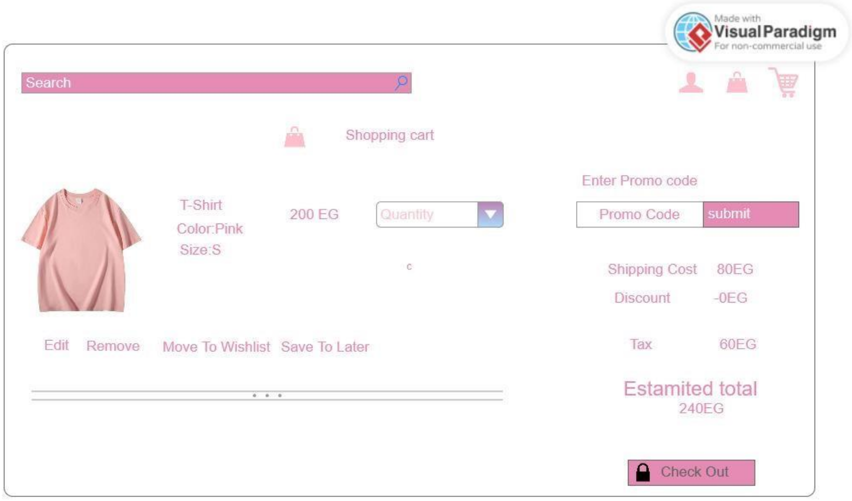


Search Product Interface:

The search product page interface is designed to be intuitive and user-friendly. The combination of a prominent search bar and a product category section allows users to quickly locate products either by searching directly or by browsing through relevant categories, ensuring a smooth and efficient shopping experience.



Shopping cart Interface: The shopping cart interface is designed to be user-friendly and transparent, providing users with a clear overview of their selected items, pricing details, and any applicable taxes, discounts, or shipping costs. The inclusion of a search bar allows for seamless navigation, ensuring a smooth shopping experience from the cart view.

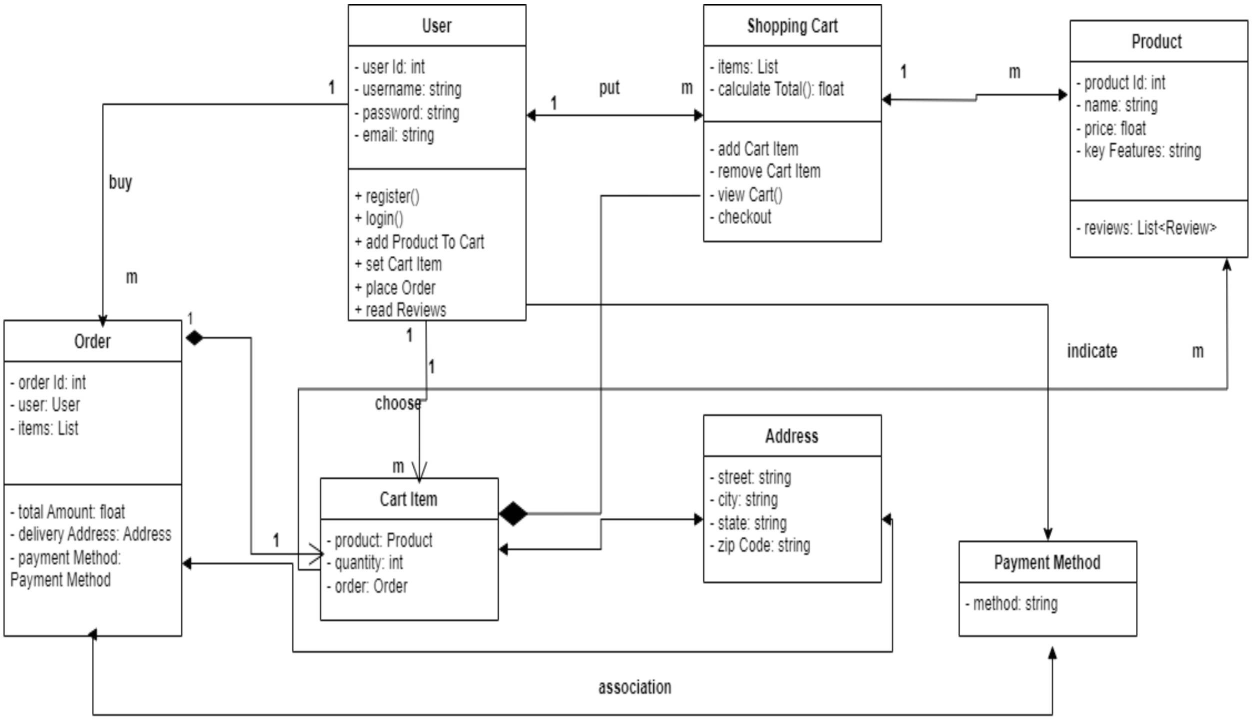


Software interface:

AI shopping system, it is an online system we use tool visual studio code, use HTML to do the interface and use visual studio to build the system. All users have access to browse the goods; but, to add any of them to their basket, they must first register and log in. Before making a purchase, users or consumers may also read other users' reviews regarding a product. Users can add as many items as they choose to their shopping cart. Next, each additional product that is added to the basket can have its amount adjusted by the consumers. Lastly, customers may put in the appropriate purchase and provide their address and preferred payment method during checkout. Subsequently, the administrator has access to the customer's information, including order data and the delivery address.

Diagrams

Class diagram



Explanation:

I. The **(User)** class has a one-to-many relationship with the **(CartItem)** class, indicating that a user can have multiple items in their cart.

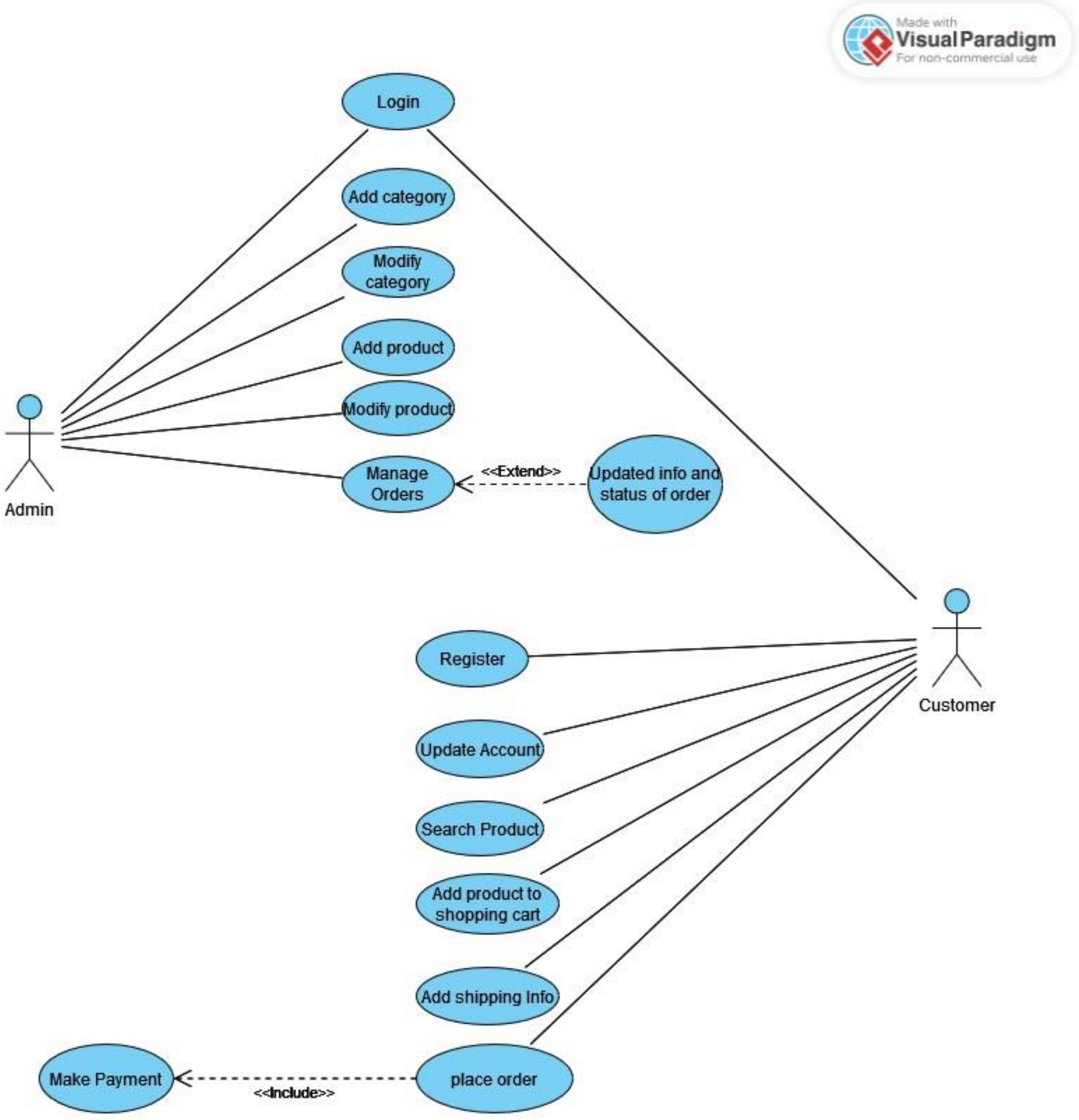
II. The (**ShoppingCart**) class contains a list of **(CartItem)** objects, indicating the composition relationship between **(ShoppingCart)** and **(CartItem).** When a **(ShoppingCart)** is destroyed, its **(CartItem)** instances are also destroyed.

III. The **(CartItem)** class has a many-to-one relationship with the **(Product)** class, indicating that multiple cart items can be associated with one product.

IV. The **(Address)** and (**PaymentMethod)** classes are associated with the **(Order)** class, indicating that an order has a delivery address and a payment method.

V. The **(Order)** class has a composition relationship with the **(CartItem**) class, indicating that when an order is created, it contains specific cart items. When an order is destroyed, its associated cart items are also destroyed.

Use case diagram.

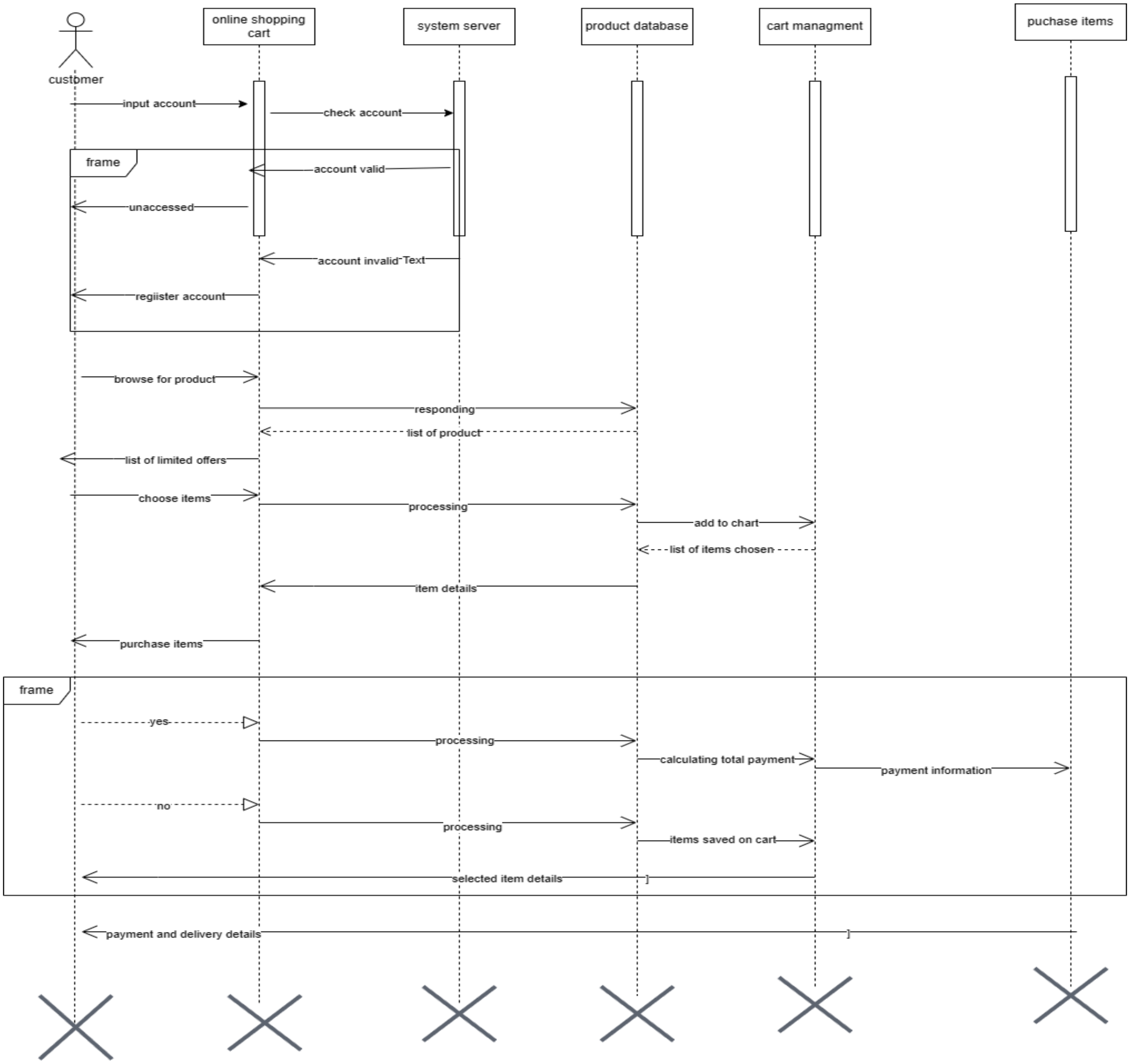


Admin Use Cases:

Use Case	Description
1. Login	The admin logs in using their credentials to access the administrative dashboard.
2. Add Category	The admin adds new categories to the online shopping platform, providing details such as category name and description.
3. Modify Category	The admin makes changes to existing categories, updating attributes like category name, description, or any other associated details.
4. Add Product	The admin adds new products to the platform, entering information such as product name, description, price, images, and other relevant details.
5. Modify Product	The admin modifies details of existing products, such as updating product information, price, availability, or any other attributes.
6. Manage Orders	The admin views and manages customer orders, reviewing details and taking actions such as approving, canceling, or fulfilling orders.
7. Update Order Info/Status	The admin updates and maintains order information and status, ensuring accurate order tracking and facilitating communication with customers.

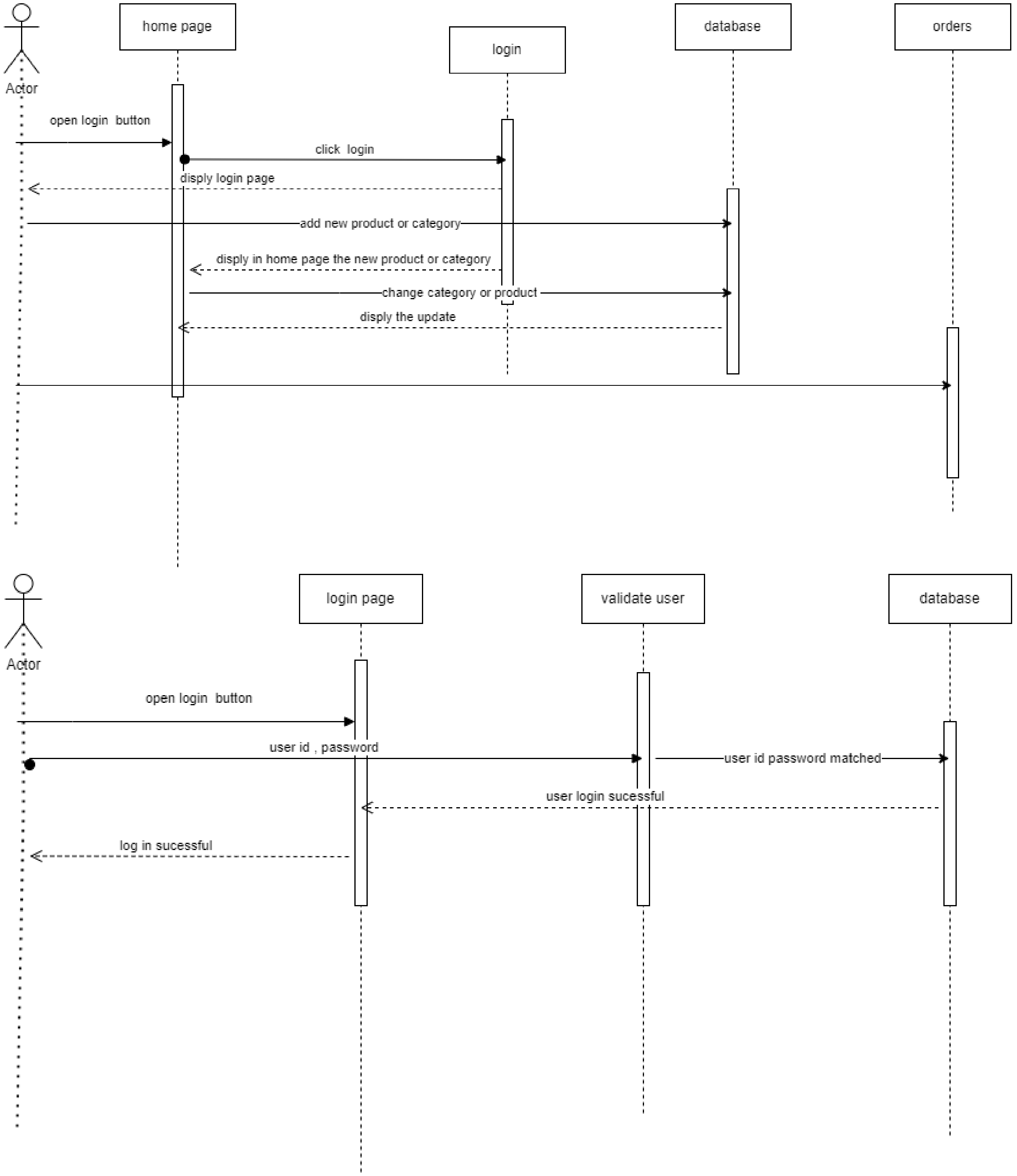
Use Case	Description
1. Search Product	Customers can search for specific products using keywords or product names to find the items they are interested in.
2. Add Product to Shopping Cart	Customers can add products to their shopping cart, allowing them to collect multiple items before proceeding to the checkout process.
3. Add Shipping Info	During the checkout process, customers provide shipping information, including the delivery address and preferred shipping method, to ensure accurate and timely delivery.
4. Place Order	Customers confirm and place their orders after reviewing order details, shipping information, and payment information. This finalizes the purchase process.
5. Make Payment	Customers make payments for their orders using various methods such as credit cards, debit cards, or online payment systems. This completes the transaction and initiates order processing.

Sequence diagram (customer)



Sequence diagram

(admin)



Scrum video

Link:https://nileuniversity.sharepoint.com/sites/software415/Shared%20Documents/General/Recordings/Meeting%20in%20_General_-20231216_203448-Meeting%20Recording.mp4?web=1

GitHub

Link:

References:

https://data-flair.training/blogs/python-onlineshopping-system/