كلية الحاسبات والذكاء الإصطناعي
Faculty of Computers & Artificial Intelligence

جامعة حلوان
HELWAN UNIVERSITY

Software Engineering-2 Cover Sheet

Project Title: ...Clothing E-commerce Platform.....

Row Number (in PDF): .33 Time: ..9:45..

#	Member Id (printed)	Member Name (printed In Arabic)	Handwritten signature
1	20210676	فريدة عماد الدين السيد	
2	20210397	سارة عبدالمنعم عبدالمنعم	
3	20210392	سارة احمد منازع	
4	20210401	سارة ممدوح عوض	
5	20210466	شهد حسن شحاتة	
6	20210268	حازم عبدالرحمن عبدالجواد	
7			

Project Requirements(grades)

SRS	
SDD	
Validation	
OCL	
AOP	
Microservices	
Cloud	

_____

20

Clothing E-commerce Platform

Description:

1- Online Catalog: platform hosts an extensive catalog of clothing products, including various types of apparel such as shirts, pants, dresses, jackets, shoes, accessories, and more. Each product is accompanied by detailed descriptions, multiple images from different angles, available sizes, colors, materials, and pricing information.

2-User Registration and Authentication: Customers have the option to register for an account on your platform. Registration typically involves providing basic information such as name, email address, and creating a password. Registered users can then log in securely to access their accounts and enjoy personalized features such as order history, saved preferences, and faster checkout.

3-Browsing and Search: Customers can easily navigate through your platform's catalog using intuitive browsing categories such as gender, clothing type, brand, size, color, and price range. A robust search functionality allows users to quickly find specific products by entering keywords or using advanced filters.

4-Product Details and Reviews: Each product page displays comprehensive details about the item, including size charts, care instructions, customer reviews, ratings, and recommendations for related products. Customer reviews and ratings contribute to transparency and help other shoppers make informed purchasing decisions.

5-Shopping Cart and Checkout: Customers can add desired items to their shopping cart with a single click while browsing the catalog. The shopping cart displays a summary of selected items, quantities, and total costs. During checkout, users proceed to securely enter shipping and billing information, select preferred payment methods, and confirm their orders.

6-Order Management: Your platform facilitates order management for both customers and administrators. Customers can track the status of their orders in real-time, receive updates on order processing, shipping, and delivery. Administrators have access to a dashboard where they can manage orders, update order status, process refunds, and handle customer inquiries.

7-Customer Support: platform provides robust customer support channels to assist users throughout their shopping journey. This may include live chat support, email support, a comprehensive FAQ section, and self-service resources.

Prompt and responsive customer service enhances the overall shopping experience and fosters customer satisfaction and loyalty.

8-Marketing and Promotions: Your platform employs various marketing strategies to attract customers and drive sales. This may include targeted email campaigns, social media promotions, seasonal discounts, loyalty programs, referral incentives, and partnerships with influencers or fashion bloggers. Marketing efforts aim to increase brand visibility, engage customers, and encourage repeat purchases.

- In summary, The Clothing E-Commerce Platform is an online marketplace dedicated to offering a wide range of clothing products to customers. It facilitates the purchase of clothing items conveniently and securely over the internet. The platform aims to provide users with an intuitive and seamless shopping experience, allowing them to browse, search for, and purchase clothing products from various categories.

Functional Requirements

Admin:

- LOGIN:

  - Admins should be able to log in securely to access the administrative dashboard.

- CRUD Categories:

  - Admins should be able to create, read, update, and delete (CRUD) product categories.

- CRUD Products:

  - Admins should be able to manage products, including adding new products, updating existing ones, and removing outdated products.

- Retrieve Orders:

  - Admins should have access to view and manage customer orders, including order details and order status.

Customer:

- LOGIN:

  - Customers should be able to log in securely using their credentials to access their accounts.

- Register:

  - New customers should be able to register for an account by providing necessary information such as name, email, and

password.

- Add Product to Cart:

  - Customers should be able to add products to their shopping cart while browsing the catalog.

- Increase/Decrease Quantity of Products in Cart:

  - Customers should be able to adjust the quantity of products in their shopping cart, both increasing and decreasing the quantity as needed.

- Delete Product from Cart:

  - Customers should be able to remove products from their shopping cart if they no longer wish to purchase them.

- Retrieve Orders:

  - Customers should have access to view their order history, including past orders and their status.

- Retrieve All Products:

  - Customers should be able to view all available products on the platform.

- Retrieve Products According to Category:

  - Customers should be able to browse and filter products based on categories such as gender, type of clothing, size, color, etc.

Non-Functional Requirements

- Performance:

  - The platform should respond to user actions within a reasonable time frame to ensure a smooth shopping experience.

  The system should be able to handle concurrent user sessions and transactions efficiently.

- Usability:

  - The user interface should be intuitive and easy to navigate for both admins and customers.

  The platform should support multiple languages and be accessible to users with disabilities.

- Security:

  - User data, including personal information and payment details, should be encrypted to prevent unauthorized access.

  - Secure authentication mechanisms should be implemented to protect user accounts from unauthorized access.

- Reliability:

  - The platform should be available for use with minimal downtime, ensuring users can access it whenever needed.

  - Backup and recovery mechanisms should be in place to prevent data loss in case of system failures.

Architecture Scope

The architecture of the Clothing E-Commerce Platform will consist of:

- **Frontend:**

  User interface for customers and admins.

- **Backend:**

  Server-side logic and database management.

- **Database:**

  Storage for user data, product information, and order details.

- **Authentication and Authorization:**

  Mechanisms to secure user accounts and control access to administrative functions.

- **Scalability:**

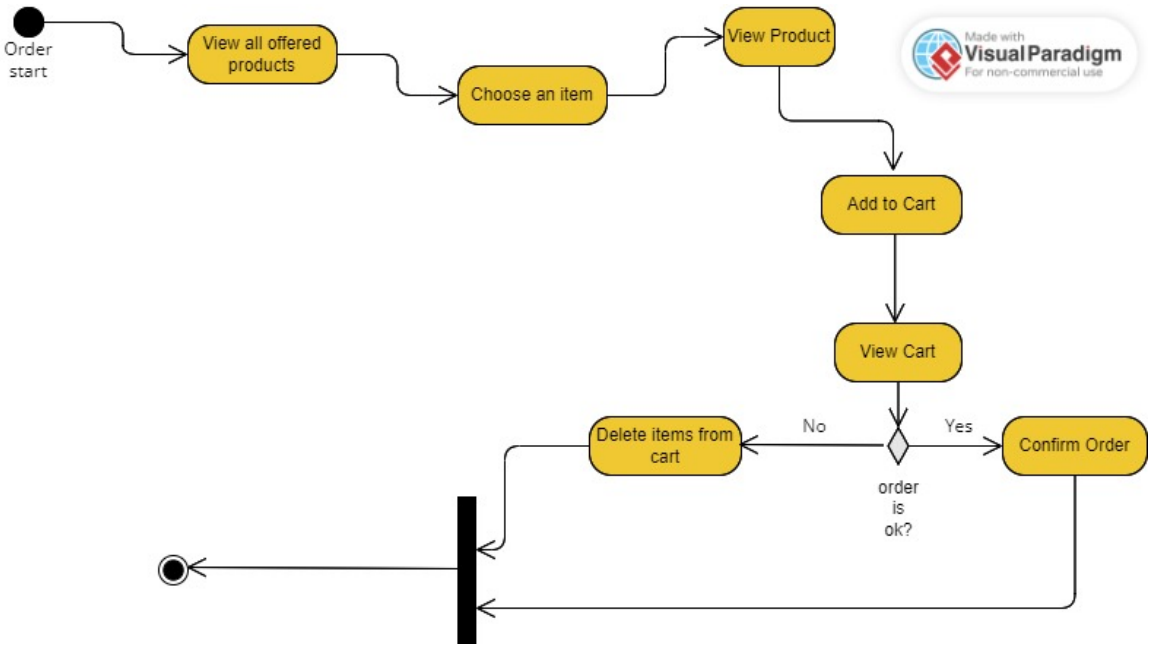
  The platform should be designed to scale horizontally and vertically to accommodate increasing user traffic and data volume.

- **Compliance:**

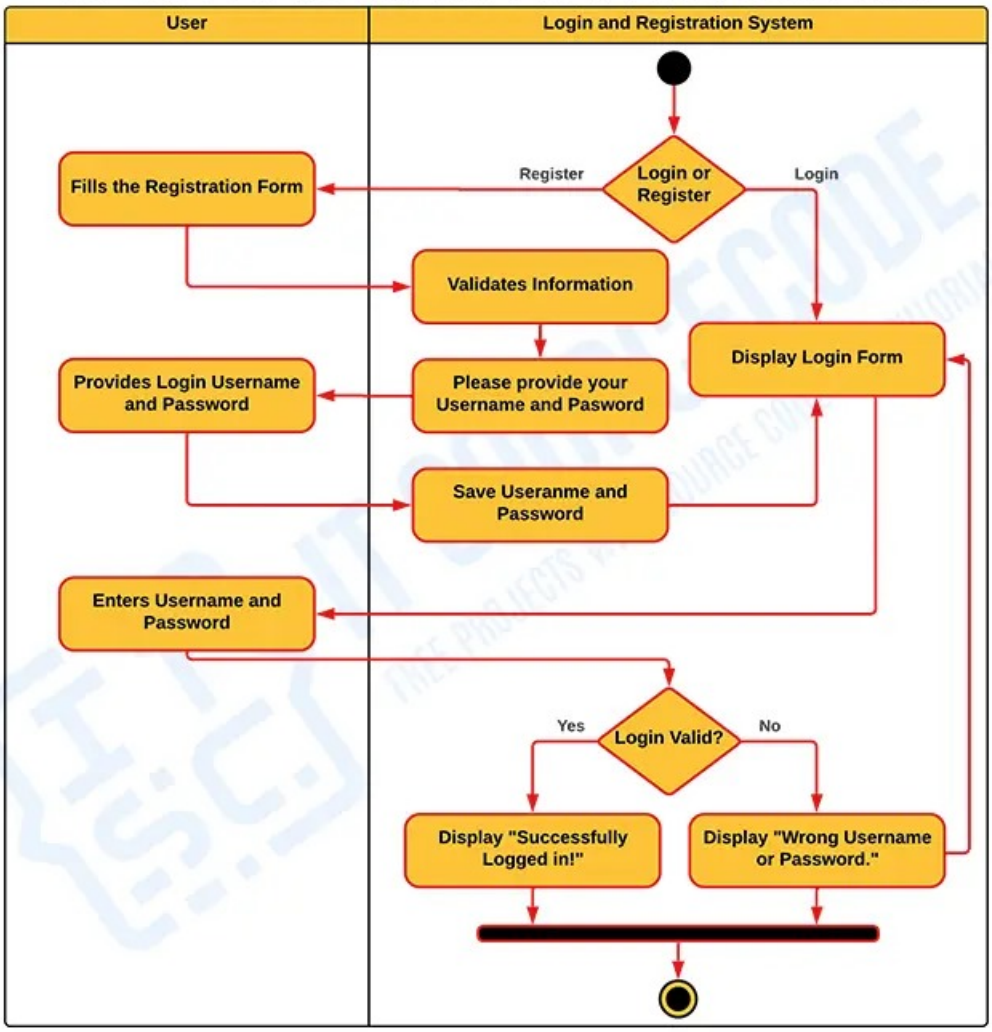
  The architecture should comply with relevant regulations and standards for e-commerce platforms, ensuring data protection and consumer rights.

Use case Diagram

Activity Diagram

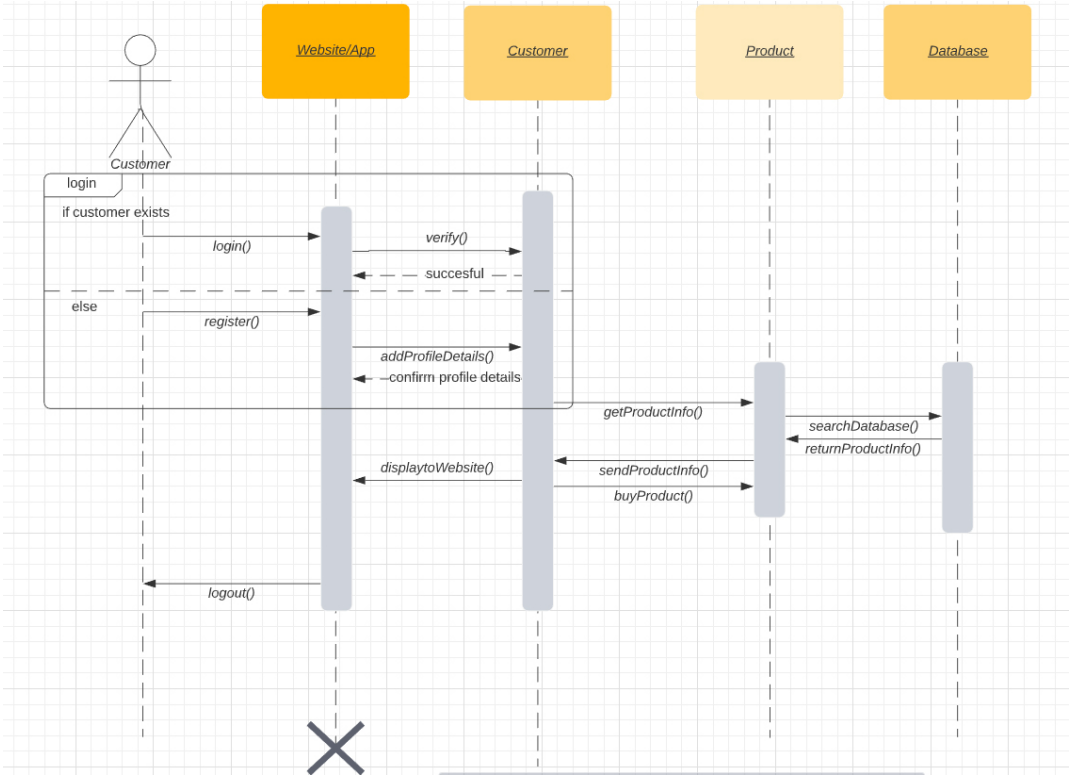


LOGIN AND REGISTRATION SYSTEM

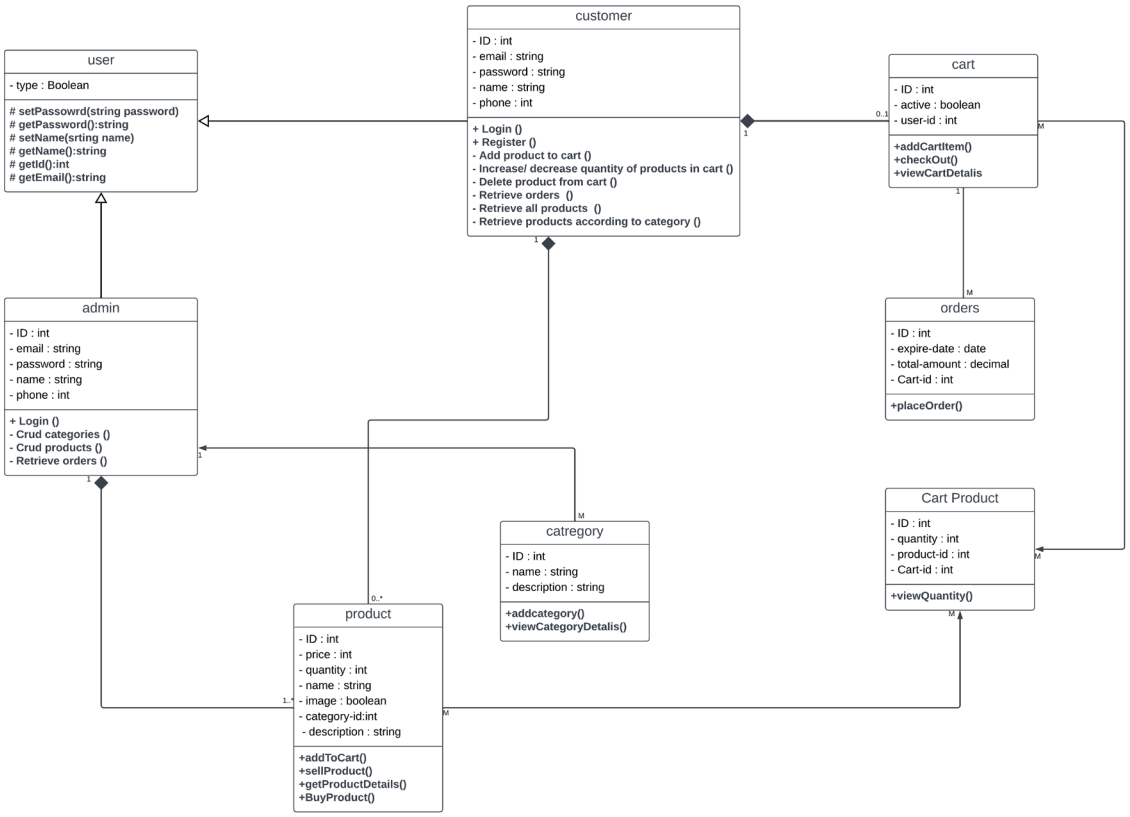


ACTIVITY DIAGRAM

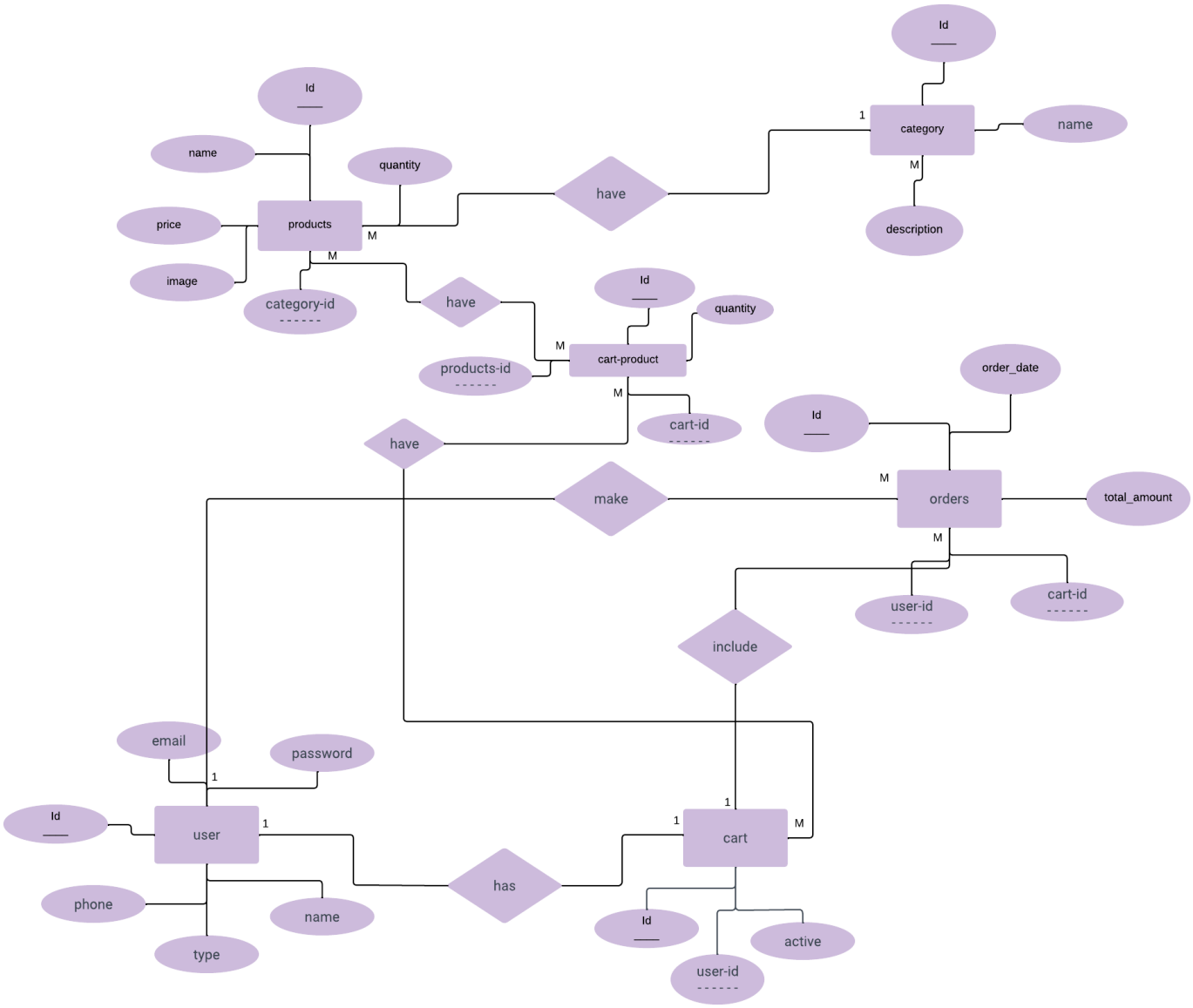
Sequence Diagram



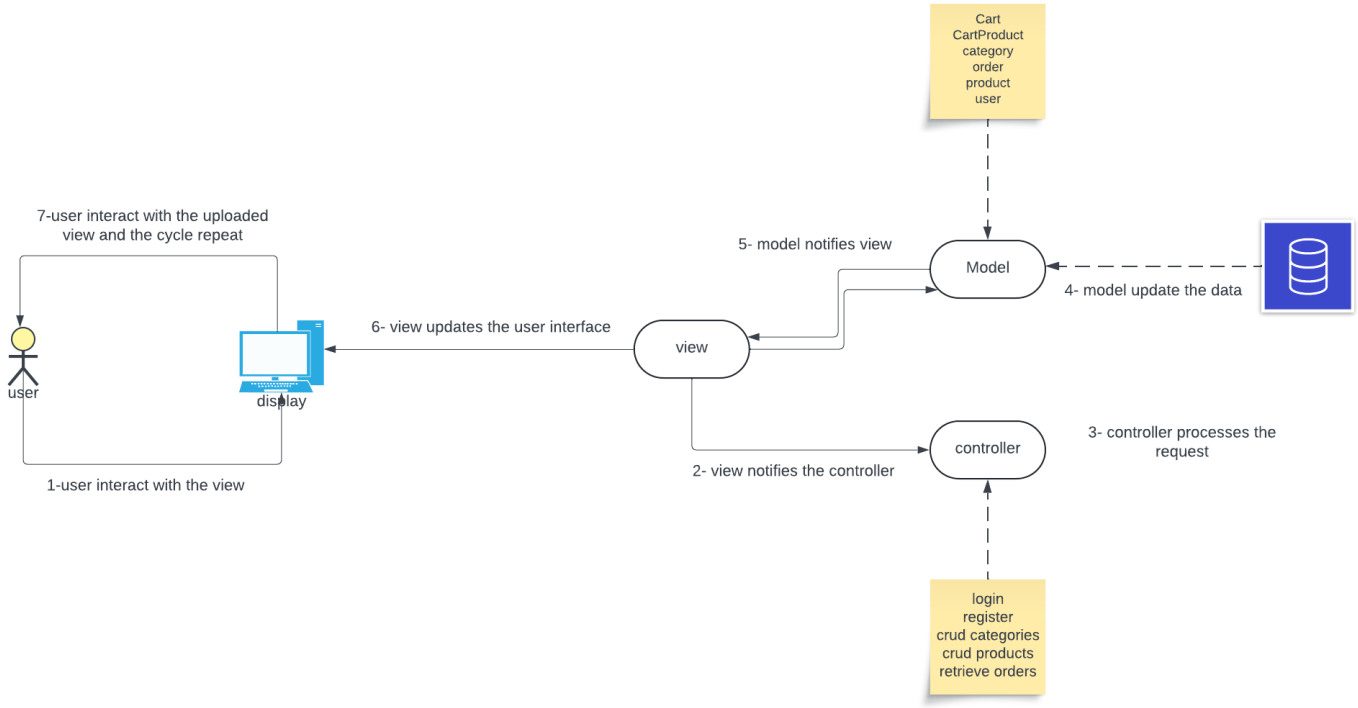
Class Diagram

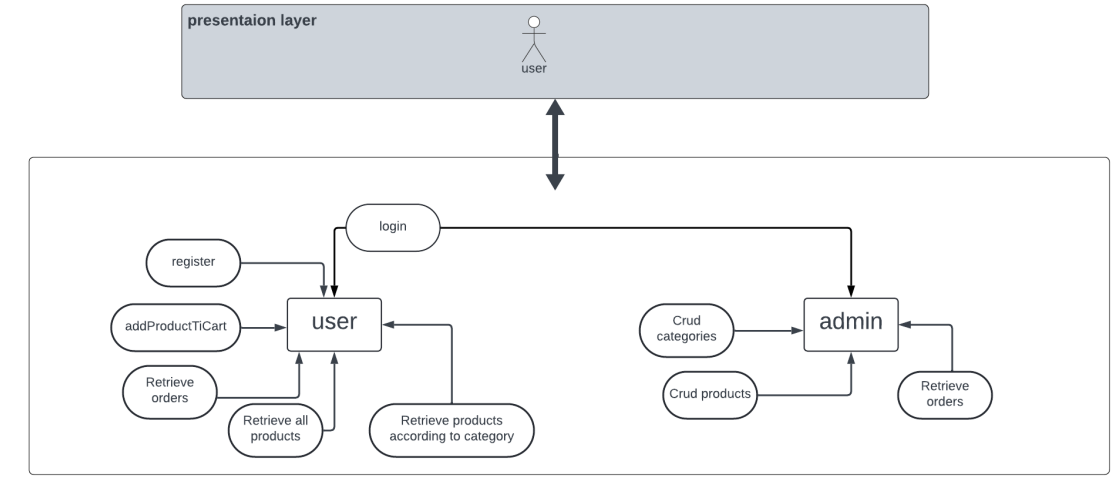


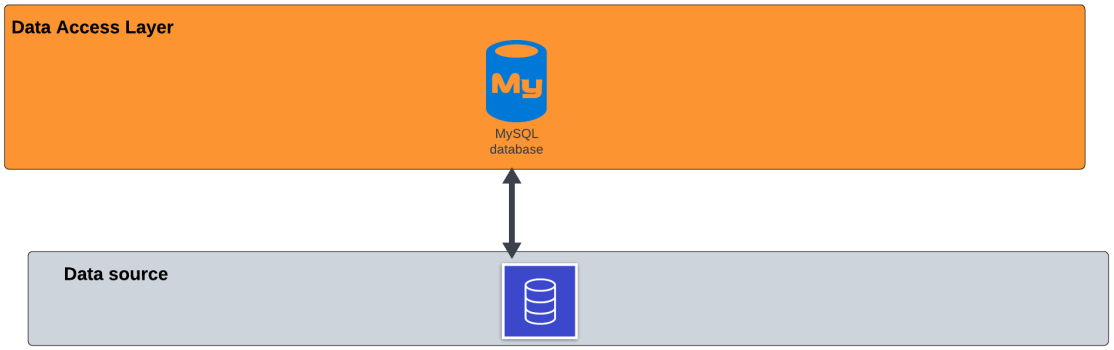
ERD


Id
name
quantity
price
products
image
category-id
have
M
1
category
name
description
Id
quantity
products-id
cart-product
cart-id
have
Id
order_date
make
orders
total_amount
user-id
cart-id
include
email
password
Id
user
phone
name
type
has
cart
Id
active
user-id

System Architecture







OCL

Context User

Inv:

--constraint : Username must be unique

```
User.allInstances()->forAll(u1, u2 | u1 <> u2 implies u1.username <> u2.username)
```

--constraint : Email must be valid

```
Self.email.isValid()
```

--constraint : user has a unique email address

```
User.allInstances()->forAll(u | u <> self implies u.email <> self.email)
```

--constraint : User's password must meet certain criteria

```
self.password.size() >= 8
```

--constraint : Phone number must be in a valid format

```
self.phoneNumber.matches('^[0-9]{10}$')
```

Context Orders

Inv:

--constraint : Total price must be greater than or equal to 0

```
self.totalPrice >= 0
```

--constraint : Order must contain at least one item

```
self.items->notEmpty
```

--constraint : Order must belong to a user

```
self.user <> null
```

--constraint : Items in the order must have positive quantities

self.items->forAll(item | item.quantity > 0)

Context Products

Inv:

--constraint : Product price must be greater than 0:

self.price > 0

--constraint : Product name must be unique

Product.allInstances()->forAll(p1, p2 | p1 <> p2 implies p1.name <> p2.name)

--constraint : Product quantity must be greater than or equal to 0:

self.quantity >= 0

--constraint : Product must belong to a category

self.category <> null

Context Category

Inv:

--constraint : Category must have a name

self.name <> null and not self.name.isEmpty() --constraint :

--constraint : Category must have at least one product

self.products->notEmpty()

--constraint : Category name must be unique

Category.allInstances()->forAll(c1, c2 | c1 <> c2 implies c1.name <> c2.name)

Context Cart

Inv:

--constraint : Items in the cart must have a valid product

self.product <> null

--constraint : Cart must belong to a user:

self.user <> null