



INF557 – RAPID WEB DEVELOPMENT

DESIGN AND FRONT END

Assessment - 1

Abstract

To design and implement a basic e-commerce store prototype of chosen service or product.

Manoj Mariyappa
N11775637

PART-1:

Beauty and Skincare Online Store

➤ ***Description:***

Welcome to our online store for beauty and skincare, which will be your one stop shop for all things relating to beauty and self-care. We offer a carefully organized variety of high-quality skincare and beauty items in our online store with the goal of making you feel and look your best. We have the necessary items for all types of self-care, whether you are a fan of skincare, makeup, or neither. Explore our wide section of skincare regimens, makeup, haircare, and wellness products. For people who value clean beauty, we take delight in providing a large selection of natural and organic solutions. Our store guarantees that you have access to the newest trends and timeless classics in the world of beauty since all of the goods are properly researched and validated. Find the ideal gifts for your loved ones or treat yourself. Our Beauty and Skincare Online Store is here to enhance your self-care journey.

➤ *User stories:*

01. Title: Appealing Homepage

Story: As a new customer, I want a visually appealing homepage that introduces me to the store's brand and features prominent skincare and beauty goods.

Acceptance Criteria:

- The home page should be designed in attracted way and that aligns with the store's brand.
- The home page should display and introduce the featured beauty and skincare products.
- The home page should include branding elements such as store logo and tagline.

02. Title: Search Bar with Filter

Story: As a new User, I want a search bar to be clearly visible on the site so I can look for certain skincare and cosmetics companies or items.

Acceptance Criteria:

- A search bar must be clearly visible and easy to find on the home page.
- Users should be able to search for products on search bar.

03. Title: Create Multiple Wishlist

Story: I'd want to be able to make and maintain many Wishlists as a user for various events or product categories.

Acceptance Criteria:

- Users Should be able to create multiple wishlists and give each wishlist a name.
- Users should be able to add or remove from whislists.

04. Title: Product reviews and ratings

Story: As a Beauty enthusiast, in order to make wise purchasing selections, I want to check consumer evaluations and ratings for skincare products, so I can buy the product which matches my beauty.

Acceptance Criteria:

- Each beauty and skincare product's page should display other customer reviews and ratings of the product.
- Users should be able to read detailed customer reviews.
- Users should be able to see the overall product rating.

05. Title: Add or remove products from cart

Story: As a user, I want to be able to add or remove the products from the bag or cart and view it.

Acceptance Criteria:

- Users should be able to add products to the cart with a single click.
- Users should be able to remove products from the cart.
- The price should update dynamically as products are added or removed.

06. Title: Description about product

Story: As a user, I want to know all the specifications and information about the product.

Acceptance Criteria:

- Each product should have a detailed description that includes information about ingredients used, usage instructions, and other details.
- Users should be able to access this information on the product page easily.

07. Title: Filter products based on price range

Story: As a consumer, I want to be able to filter things by price range so that I may buy skincare and cosmetic products while staying within my budget.

Acceptance Criteria:

- Users should be able to set a price range filter while browsing products.
- The filter should allow users to specify a minimum and maximum price.

08. Title: Responsive and Device independent

Story: As a Mobile user, I need a flawless purchasing experience on my smartphone or tablet, I need the website to be responsive and mobile-friendly.

Acceptance Criteria:

- The website should be able to adapt to different screen sizes and orientations.
- All website features, including navigation, search, and product display, should work smoothly on mobile devices.

09. Title: Return and refund process

Story: As a consumer, I want the shop to provide a simple returns and refunds procedure, along with directions on how to start returns and exchanges.

Acceptance Criteria:

- The website should provide clear and easily accessible returns and refund policy.
- Instructions for initiating the returns and exchange should be provided.
- Customer should be able to request the request returns and refunds online easily.

10. Title: Customer Support

Story: I want to be able to use customer support tool as a skincare expert to ask questions, receive specialized skincare guidance and resolve all my doubts on the products.

Acceptance Criteria:

- Customer support feature should be available on the website.
- Skincare experts contacts should be displayed in website.
- User's details must be taken when they request for advice and should receive personalized skincare advice when requested.

➤ *Conceptual Model*

Class Name: Customer/user

Attributes: id, Name, Phone, Email, Address

Behaviors: UpdateCust()

Class Name: Product

Attributes: id, Name, price, Description, reviews, ratings

Behaviors: Updateprod(), AddToCart(), AddToWishlist(), GetReviews(), GetRatings()

Class Name: Order

Attributes: id, OrderId, CustomerId, TotalBill

Behaviors: Add(), view(), checkout()

Class Name: Cart

Attributes: Customerid, ProductId, ProductName, Price, TotalBill

Behaviors: Update(), add(), view(), checkout()

Class Name: Wishlist

Attributes: id, WishlistName, Productid, ProductName, Price

Behaviors: Update(), add(), remove(), view(), addToCart()

Class Name: Category

Attributes: id, Name, ProductId

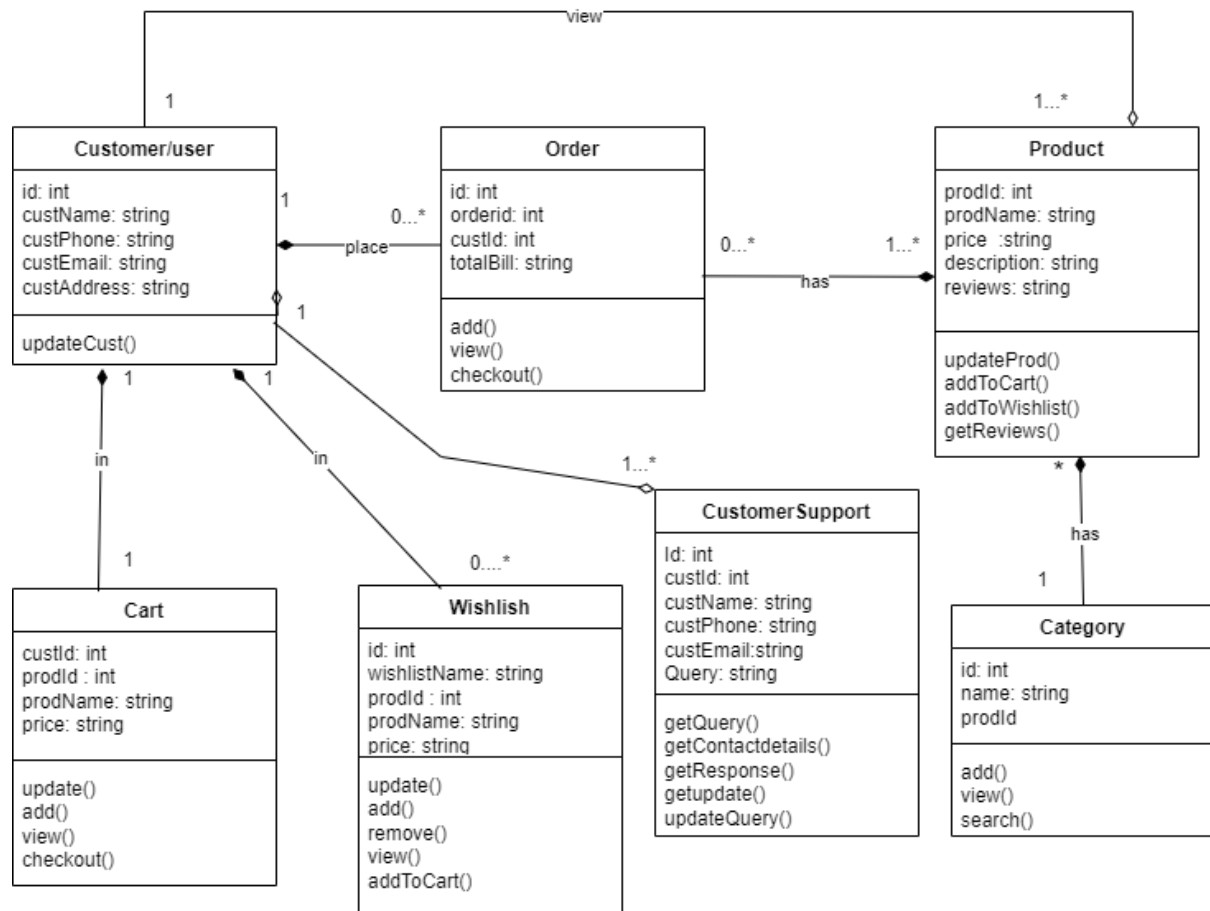
Behaviors: add(), view(), search()

Class Name: CustomerSupport

Attributes: id, Customerid, CustomerName, CustomerPhone, CustomerEmail, CustomerAddress,Query

Behaviors: getquery(),getContactdetails(),getResponse(),updatequery(),getupdate()

UML CLASS DIAGRAM



➤ *Database Design*

Class Name: Customer/user

Attributes: id, Name, Phone, Email, Address

Behaviors: UpdateCust()

Associations:

- ✓ Customers can have Orders (1 to many relationship).
- ✓ Customers can have Carts (1 to 1 relationship).
- ✓ Customers can have Wishlists (1 to many relationship).

Class Name: Product

Attributes: id, Name, price, Description, reviews, ratings

Behaviors: Updateprod(), AddToCart(), AddToWishlist(), GetReviews(), GetRatings()

Associations:

- ✓ Products can be in Carts (many to many relationship).
- ✓ Products can be in Wishlists (many to many relationship).
- ✓ Products can be part of Categories (many to many relationship).

Class Name: Order

Attributes: id, OrderId, CustomerId, TotalBill

Behaviors: Add(), view(), checkout()

Associations:

- ✓ Orders are associated with Customers (many to 1 relationship).
- ✓ Orders can contain Products (many to many relationship).

Class Name: Cart

Attributes: Customerid, ProductId, ProductName, Price, TotalBill

Behaviors: Update(), add(), view(), checkout()

Associations:

- ✓ Carts belong to Customers (1 to 1 relationship).
- ✓ Carts can contain Products (many to many relationship).

Class Name: Wishlist

Attributes: id, WishlistName, Productid, ProductName, Price

Behaviors: Update(), add(), remove(), view(), addToCart()

Associations:

- ✓ Wishlists belong to Customers (many to 1 relationship).
- ✓ Wishlists can contain Products (many to many relationship).

Class Name: Category

Attributes: id, Name, ProductId

Behaviors: add(), view(), search()

Associations:

- ✓ Categories can contain Products (many to many relationship).

Class Name: CustomerSupport

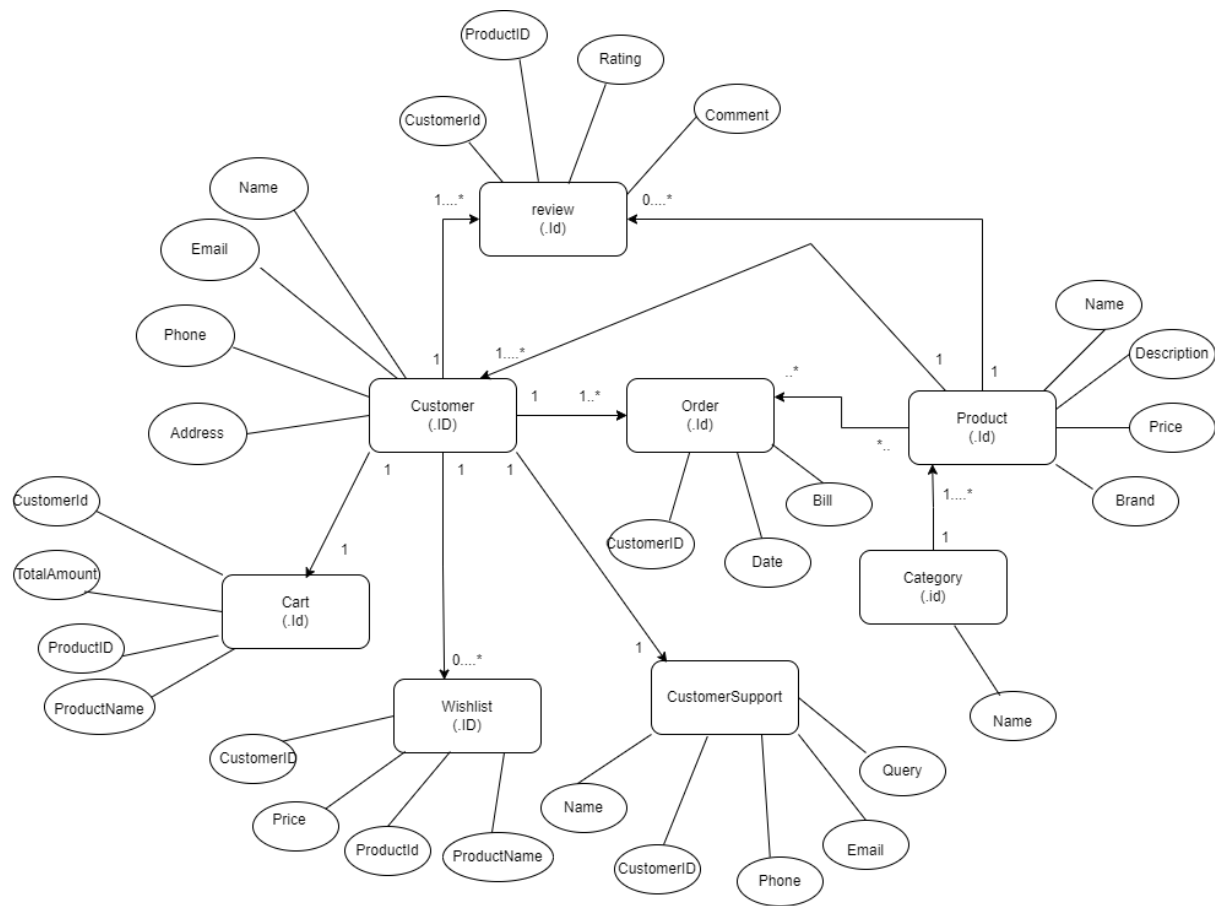
Attributes: id, Customerid, CustomerName, CustomerPhone, CustomerEmail, CustomerAddress, Query

Behaviors: getquery(), getContactdetails(), getResponse(), updatequery(), getupdate()

Associations:

- CustomerSupport can have Customer (many to many relationship).

DATA MODEL:



Relationships:

- ✓ Customers can have multiple Reviews (One-to-Many relationship between Customer and Review).
- ✓ Products can be associated with multiple Reviews (One-to-Many relationship between Product and Review).
- ✓ Products can belong to multiple Shopping Carts (Many-to-Many relationship between Product and ShoppingCart).

- ✓ Products can be added to multiple Wishlists (Many-to-Many relationship between Product and Wishlist).
- ✓ Shopping Carts belong to individual Customers (One-to-One relationship between ShoppingCart and Customer).
- ✓ Wishlists belong to individual Customers (One-to-One relationship between Wishlist and Customer).
- ✓ Orders are placed by Customers (One-to-Many relationship between Customer and Order).
- ✓ Orders can contain multiple Products (Many-to-Many relationship between Order and Product).
- ✓ Products can belong to multiple Categories (Many-to-Many relationship between Product and Category).

This Data Model Provides a Structured representation of the key entities and their attributes and how they are related to each other in the context of my simple online e-commerce store –

Beauty and Skincare Online Store.

DATABASE DESIGN: -

Creating a database design for Beauty and Skincare Online Store using Data Model.

→ Database Tables:

1. Customer Table:

- CustomerID – Primary Key
- Name
- Email
- Phone
- Address

→ Example

CustomerID	Name	Email	Phone	Address
1	Manoj	Manoj@gmail.com	0451234567	ABC
2	Preksha	Preksha@gmail.com	0147852368	xyz

2. Product Table

- ProductID – Primary key
- Name
- Description
- Price
- Brand

→ Example

ProductID	Name	Description	Price	Brand
11	Green Tea Solid Mask	The GreenTea solid mask removes dirt,oil.	\$ 13.45	QWER
22	Beauty Rose Facial Massage set	This contains Rose Quartz Face Roller and a Rose Quartz Guasha	\$29.98	ASDF

3. Review Table

- ReviewID – Primary Key
- CustomerID – Foreign Key
- ProductID – Foreign key
- Rating
- Comment

➔ **Example**

ReviewID	CustomerID	ProductID	Rating	Comment
111	1	11	4.5/5	Super Product.
222	2	22	4.7/5	Worth It.

4. Cart Table

- CartID – Primary Key
- CustomerID – Foreign Key
- ProductID – Foreign Key
- ProductName
- TotalAmount

➔ **Example**

CartID	CustomerId	ProductId	ProductName	TotalAmount
1A	1	11	Green Tea Solid Mask	\$ 13.45
2A	2	22	Beauty Rose Facial Massage set	\$29.98

5. Wishlists Table

- WishlistId - Primary Key
- CustomerID - Foreign Key
- ProductID - Foreign Key
- WishlistName
- ProductName
- Price

➔ **Example**

WishlistId	CustomerID	ProductID	WishlistName	ProductName	Price
123	1	11	FaceMask	Green Tea Solid Mask	\$ 13.45

6. Order Table

- OrderId – Primary Key
- CustomerID – Foreign Key
- Date
- Bill

➔ Example

OrderId	CustomerID	Date	Bill
111	1	11/11/2024	\$13.45
123	2	11/2/2024	\$29.98

7. Categories Table

- CategoryId – Primary Key
- ProductID – Foreign Key
- Name

➔ Example

CategoryId	ProductID	Name
1	11	Mask
2	22	Massage

This is an database design to describe the structure of database for Beauty and Skincare Online Store

Table Relationships

- ❖ Customers (CustomerID) can have multiple Reviews (One-to-Many relationship).
- ❖ Products (ProductID) can have multiple Reviews (One-to-Many relationship).
- ❖ Products (ProductID) can belong to multiple Categories, and Categories can contain multiple Products (Many-to-Many relationship).
- ❖ Products (ProductID) can be in multiple Shopping Carts, and Shopping Carts can contain multiple Products (Many-to-Many relationship).
- ❖ Products (ProductID) can be in multiple Wishlists, and Wishlists can contain multiple Products (Many-to-Many relationship).
- ❖ Customers (CustomerID) can place multiple Orders (One-to-Many relationship).

Database Constraints

- ❖ CustomerID in Reviews and ShoppingCarts is a foreign key referencing the Customers table.
- ❖ ProductID in Reviews, ShoppingCarts, Wishlists, and Categories is a foreign key referencing the Products table.
- ❖ Review ratings are constrained to be between 1 and 5.
- ❖ Quantity in ShoppingCarts is a positive integer value.
- ❖ Date in Orders stores the date when the order was placed.

PART - 2

FRONT END PROTOTYPE