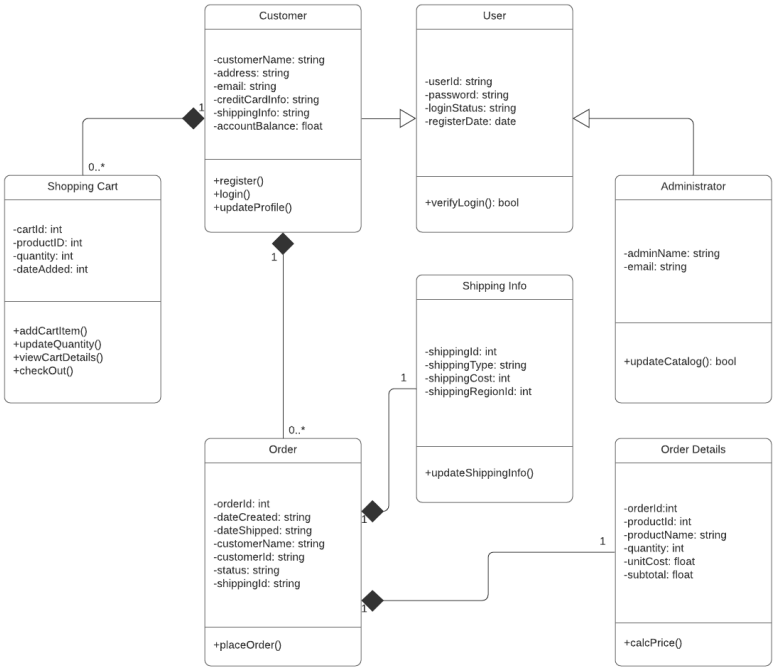
Southern New Hampshire University

CS 255 Module Four: Hamp Crafts' Online Storefront System

Object Model (UML Class Diagram)



PROMPT: *What are the different functions of the online storefront? How are they represented in this type of model?*

Several functions need to be implemented for Hamp Crafts' Online Storefront appliation. In an object model, these functions are represented by methods of the responsible class. Each class represents object types that exist within the system. These include:

- Functions represented by the User class:
    - Login verification: represented by the verifyLogin() method
- Functions represented by the Administrator class:
    - Update Catalog: represented by the updateCatalog() method
- Functions encapsulated by the Customer class
    - Customer registration: represented by the register() method
    - Customer login: represented by the login() method
    - Update customer profile: represented by the updateProfile() method
- Functions encapsulated by the Shopping Cart class
    - Add item to the cart: represented by the addCartItem() method

    - Update cart item's quantity: represented by the updateQuanity() method
    - View cart details: represented by the viewCartDetails() method
    - Checkout: represented by the checkOut() method
- Functions represented by the Order class:
    - Place order: represented by the placeOrder() method
- Functions represented by the Order Details class
    - Calculate the price of the line item: represented by the calcPrice() method
- Functions represented by the Shipping Info class
    - Update shipping information: represented by the updateShippingInfo() method

PROMPT: *What are the different classes of "users" represented by this object model? What are the associations between these classes?*
Users within the Hamp Crafts' Online Storefront application are divided into two types, Customers and Administrators. These are associated with the user class by Inheritance/Generalization, which is a child-parent relationship (Valacich, J. S., & George, J. F., 2019) where the Customer and Administrator classes inherit all the functionality of the User class.

PROMPT: *How would the objects "use" their respective variables and functions?*
Objects are single instances of classes with properties and attributes with values known as state. This state cannot be changed from outside the object but can only be accessed through the object's behaviors which describe how an object acts and reacts, expressed through its methods. An operation occurs when one object acts on another to get a response.

PROMPT: *Does this object model capture all of Hamp Crafts' desired functionality? Why or why not?*
This object model represents the functionality of the system well. However, there are a few classes that are needed to implement Hamp Crafts' Online Storefront system, including:

- Product to represent individual available products for sale
- Catalog to represent the collection of products that are available for sale
- Shipper to represent each shipping company used to ship orders and inventory
- Shipping Method to represent each shipping method available to ship customer orders
- Supplier to represent each inventory supplier
- Supplier Contract to represent each contract made with a supplier

PROMPT: *The above diagram uses a solid diamond shape to represent a form of aggregation. What type of aggregation does this represent? What does it imply about the relationship between the classes? Why is a solid diamond the appropriate choice here?*
The solid diamond shape represents object composition. Composition implies that one object belongs to another and cannot exist without the other object (Valacich, J. S., & George, J. F.,

2019). In the model of the online storefront, there are several classes with composite relationships:

- Shopping Cart and Customer
- Order and Customer
- Shipping Info and Order
- Order Details and Order

A composite relationship is appropriate for each of the classes indicated above. An order cannot exist without a customer, shipping info without an order, or order details without an order.

PROMPT: *How well do you think a process model describes the system? What information does it make easier to understand? What aspects of the system are more difficult to understand or are not represented?*

The process model describes many properties of Hamp Crafts' Online Store system well. It does a great job of describing the flow of data through the system and the discrete processes that exist within the system. Process modeling describes the processes and data flow of a system. Valacich, J. and George, J. (2019) say that "Process modeling involves graphically representing the functions, or processes that capture, manipulate, store, and distribute data between a system and its environment and between components within a system" (p. 179).

This provides a high-level view but is more difficult to design and implement concrete system components. Some questions that are not well defined in a process model include:

- What component of the system is responsible for a particular process?
- What is the desired state of the system before, during, and after a process is executed?
- Does a process have a local or global state?

PROMPT: *How well do you think an object model describes the system? What information does it make easier to understand? What aspects of the system are more difficult to understand or are not represented?*

The object model also describes aspects of the system well. It models what objects exist in the system and how they interact. The model defines the attributes and behaviors of each object and provides a clear understanding of which component is responsible for each function and how each object can change the system's state (Valacich, J. S., & George, J. F., 2019).

The process model simplifies the development of the business logic by using object instances to manage state and functionality. However, object models have a more limited view of how each process flows through the system, and the core functionality is less visible. Each class must be reviewed to determine what functionalities exist, and the order that they occur cannot be immediately discerned.

REFERENCES

Valacich, J. S., & George, J. F. (2019). Modern Systems Analysis and Design (9th ed.). Pearson Education (US). https://mbsdirect.vitalsource.com/books/9780135172827