Reginald Cooper CS-255

Instructor Trajkovski 7/26/2024

Hamp Crafts' online storefront

Functions of the Online Storefront

1. Shopping Cart:

+ addCartItem(),

+updateQuantity(),

+viewCartDetails(),

+checkOut()

The **ShoppingCar**t class represents the shopping functionalities available to the customers. Each shopping cart object can add items, update quantities, view cart details, and checkout.

2. Customer or User:

Functions:

+register(),

+login(),

+updateProfile()

The **Customer or User** class represents customers' personal and account details. This includes functions for registering, logging in, and updating profile information.

3. User Authentication:

+verifyLogin(): bool

The **User** class handles user credentials and login status, providing a method to verify login details.

4. Order Processing:

+placeOrder()

The **Order** class takes order information, including creation date, shipping details, and customer information.

5. Shipping Information Management:

+updateShippingInfo()`

The **ShippingInfo** class contains shipping details, including type, cost, and region.

6. Order Details:

+calcPrice()

The **OrderDetails** class provides detailed information about each order, including product information, quantity, and price calculation.

7. Administrative Functions:

+updateCatalog()

The **Administrator** class allows administrative tasks such as catalog updates.

Users and Associations Classes

1. Classes of Users:

User: user-related attributes like

-userId,

-password,

-loginStatus

Customer: from User includes attributes such as

-customerName,

-address

-creditCardInfo.

Administrator: Also from User, with attributes like

-adminName

-email.

2. Associations:

A customer has one or more ShoppingCart objects (0..\* to 1 relationship).

A customer can place one or more Order objects (1 to 0..\* relationship).

Each Order has associated OrderDetails (1 to 0..\* relationship).

An Order has one associated ShippingInfo (1 to 1 relationship).

The Usage of Object Variables and Functions

Customer objects use variables like customerName and email to store user information.

ShoppingCart objects manage cart-related data like productID and quantity, using methods like addCartItem() to manipulate these data.

Does this Capture Hamp Crafts' Desired Functionality

The object model captures the essential functionality for Hamp Crafts' online storefront:

Customer account management and profile updates.

Shopping cart operations and checkout.

Order and shipping management.

Administrative tasks like updating the catalog.

But some things are hard to design, such as the integration of external payment systems and the methods for customer notifications, because they don't seem to be in the diagram.

The Aggregation Type and Relationship

The solid diamond shape represents arrangement, a strong form of aggregation where the contained objects (ShoppingCart or OrderDetails) are dependent on the lifecycle of the containing object (Customer or Order). The solid diamond is appropriate here because the existence of these parts (shopping cart items, order details) depends overall (customer account, order).

Process Model and Object Model

1. Process Model:

Strengths: Describes the sequence of operations and the flow of data within the system, making it easier to understand the procedural steps involved.

Weakness: May not provide a clear view of data relationships or encapsulate state and behavior in an intuitive way.

2. Object Model:

Strengths: Offers a clear representation of the system's structure by encapsulating data and behavior within objects. It makes relationships between different data entities and functionalities clearer.

Weakness: Does not fully capture great interactions and processes between objects, requiring more diagrams or descriptions for a complete understanding.