Southern New Hampshire University

CS-255 System Analysis and Design

4-2 Assignment: Evaluate an Object Model

Winnie Kwong

Professor Eppenger

July 22nd, 2023

**Interpret the Object Model**

**Different Functions of the Online Storefront Represented in an Object Model**

The different functions of the online storefront are: addCartItem(), calcPrice(), checkOut(), login(), placeOrder(), register(), updateCatalog(), updateProfile(), updateQuantity(), updateShippingInfo(), verifyLogin(), and viewCartDetails(). These functions, or operations, are represented as private and public functions within the classes and subclasses methods that allow the developer to specify the function's behavior. When a function is public, it can be accessed by any other class or subclass. Private functions restrict the access of any other class or subclasses. The UML diagram represents different processes such as administrative functions, user registration, and order and payment management.

**Different Classes of “Users” Represented in an Object Model**

The different classes of “Users” represented by this object model are “Administrator” and “Customer.” The association between these classes is the aggregation connection meaning “Administrator” and “Customer” can exist without its parent class “Users.” The associated attributes are address string, Name string, and email string. The “Users” are individual customers that can create an account and purchase products. The administrator will handle all administrative tasks that include updating customer information.

**How Objects “Use” Their Respective Variables and Functions**

Objects use their respective variables and functions by using commands to operate in programming. When the objects use their variables, they will capture and store their data to use at a later stage. An example would be requiring the customer's login information to purchase items or requesting administration update the catalog with the customer's new data. The shopping cart is another example that stores the data of all products until the customer is ready to purchase the items. Once the items have been purchased, the information will be held in the customer's history, including the customer's name and shipping ID.

**Capturing All of Hamp Crafts’ Desired functionality**

The object model captures majority functionality, but it does not capture all of Hamp Crafts’ desired functionality. Some functions that could have been included are purchase history, credit card vendor service, and notifications. Purchase history would allow customers to track their order as well as the status of their order. The new online store front is also missing the credit card vendor service that is supposed to receive customer payments. The subclass should include the payment transaction that allows the merchant to accept varieties of payment methods to increase the business’ cash flow. Once the customers have purchased the item, the subclass “Order” should include a function to alert customers of the transaction.

**UML Diagram with Aggregation**

The solid diamond shape represents a form of aggregation called composition. The connection for composition is when a child object cannot exist without its parent object. Composition is the appropriate choice because an “Order Detail” and “Shipping Info” cannot exist unless their parent, an “Order,” is activated. Also, an “Order” and “Shopping Cart” can only exist if their parent class, “Customer,” exists and is shopping and places an order activating “Order Detail” and “Shipping Info.” By using composition in the UML diagram, it helps to provide a layout and structure of how each class and subclasses will be affected by one another and how each object’s association describes the relationship between all existing objects within the workflow.

**Comparing a Process Model and an Object Model**

**Understanding a Process Model**

A process model should describe a workflow with individual steps and overview of the tasks that include arrows, connectors, indicators, etc. The process model’s visualization was easy to analyze and understand since the processes were divided into smaller and manageable steps. The arrows and connectors made it easier to understand because these components reflected the flow of work and how the program will operate. Aspects of the system would be difficult to represent is the data structure of how the customer’s and administrator’s workflows in the real world. The process model would be unable to show if a customer abandons their cart or how the business will handle returns and cancellations. The process model does not include how administrators would receive an alert that a transaction was made or update the company website. Process models would be able to describe the system, however it would not be able to capture all data structures that would occur in the real world.

**Understanding an Object Model**

For an object model to describe a system well, the system must show the structure of the system and the association between different aspects of the system. The UML of Hamp Crafts’ new online storefront clearly shows how all the variables and functions work together. However, the only minor aspect that was not represented was to include hierarchy in the UML diagram. The “Shipping Info” and “Order Details” subclasses should have been at the bottom tier. In contrast, “Shopping Cart,” “Order,” and “Administrator” should have been aligned in the middle tier to show the order of inheritance between classes and subclasses. The object model was organized, and I understood each key variable and function, but it could have been improved to show the hierarchy in the UML diagram.