Mikayla-Joy Botha

CS- 255 System Analysis and Design

Module 4: Evaluating An Object Model

**Interpretation of Object Model for Online Storefront**

* Different Functions
  + The online store has a variety of functions, including user account management, payment processing, order processing, notifications, and administrative support. These applications are represented as classes in the model, each with its characteristics and methods.
* Different User Classes
  + This object model consists of two main categories of users: "Customer" and "Administrator." Customers are the regular buyers who establish accounts, make purchases, and engage with the store. On the other hand, Administrators take charge of controlling the system and offering customer assistance.
* Associations Between User Classes
  + Customers and administrators have dynamic and collaborative relationships with multiple classes. As an illustration, Customers are connected with the class "Shipping Info" when it comes to their order shipments, the class "Order" for placing new orders, and the "Shopping Cart" class to effectively manage their selected items. Administrators are strongly affiliated with the "Order Details" class, which undoubtedly assists them in efficiently handling and fulfilling orders and engaging with the store. On the other hand, Administrators take charge of controlling the system and offering customer assistance.
* Variable and Function Use
  + The objects created from these classes used their transformations and functions to perform different tasks. For example, the Customer object can use its variables to store shipping information and use its parts to place orders. The Administrator object can use its functions to manage order information or provide customer support.
* Object Model Functionality
  + In terms of functionality, the object model captures several key functionalities that Hamp Crafts desires for its online storefront, including user account management, payment processing, order tracking, and administrative support. However, with more details about the specific attributes and methods within each class, it's easier to determine if it covers all desired functionality.
* Aggregation
  + The solid diamond form inside the object model commonly represents "composition" or "strong aggregation." This means that the classes in the diamond are quintessential to the whole; if the whole is destroyed, its components are also removed. In this context, classes like "Shipping Info," "Order," and "Shopping Cart" are carefully related to the "User" or "Customer" magnificence, suggesting that those classes are a part of or owned through the person or customer entity. This way, if the person or customer is removed from the system, those associated classes can also be eliminated.

**Comparison Between Process and Object Models**

* Process Models
  + The process model effectively visualizes business processes and data flows within the system. It focuses on the sequence of activities and the data flow between them. However, the detailed structure of the system remains the same.
* Object Models
  + The object model is valid in representing the hierarchy of the system and the relationships between classes and objects. It helps to understand how data and functions are structured and contained. It may not provide a comprehensive view of business sequences.
* In conclusion, the object model is well suited to capture various aspects of system design, including user classes, their interactions, and the composition of objects. Thus, the object model complements the process model, which focuses more on the flow of activity. Both of these models provide a comprehensive understanding of the structure and operation of the system.

**References:**

Redding, G., Dumas, M., Ter Hofstede, A. H. M., & Iordachescu, A. (2008, January 1). *Transforming Object-Oriented Models to Process-Oriented Models*. Springer eBooks. https://doi.org/10.1007/978-3-540-78238-4\_15

Redding, G., Dumas, M., Hofstede, A. H. T., & Iordachescu, A. (2008, October 9). Generating Business Process Models from Object Behavior Models. *Information Systems Management*, *25*(4), 319–331. https://doi.org/10.1080/10580530802384324