Matthew Trembley

11/21/2021

Southern New Hampshire University

CS-250-T2609

4-2 Assignment: Evaluate an Object Model

There are many functions within this model. They are all public functions, with up to 4 under a class. These functions and their corresponding classes are as follows:

* Shopping Cart – addCartItem(), updateQuantity(), viewCartDetails(), checkOut()
* Customer – register(), login(), updateProfile()
* User – verifyLogin()
* Administrator – updateCatalog()
* Order – placeOrder()
* Shipping Info – updateShippingInfo()
* Order Details – calcPrice()

The different classes of “users” are Customer, Administrator, and User. User is the prent class, in which Customer and Administrator both inherit traits from. The objects within this model “use” their variables and functions to essentially create the system. Each object creates data to be stored and kept track of. Whether it is a new Customer using register() or if the system needs to use verifyLogin() to check if the user logging in is an Administrator or a Customer. Overall, this model seems to capture Hamp Crafts’ desired functionality. It seems to be a generic online store tool in which customers can easily place orders and receive their shipping info. If there were to be any improvement, is to create a function of viewShippingInfo() in Shipping Info class, to allow for a customer or administrator to view the data for a variety of possible reasons.

The solid black diamond used to represent a form of aggregation called composition. What this means for example; a Shopping Cart class can not exist without a Customer class. The Order Details class can not exist without an Order. In non-technical speak, it makes sense. Order details do not exist without an order to go with it. A shopping cart needs a customer to push it around and add things to it. The solid black diamond implies that these classes depend on one another. The solid black diamond is the correct choice to use here because of this implication that one class needs another to exist. The relationship is solid and so is the diamond.

I think the process model represents the system “okay”. In my opinion, it just gives the flow of information and how it is to be handled, or which direction it is able to take. It does a wonderful job of showing this information flow, but it doesn’t necessarily describe “how” like an object model would.

On the other hand, I think an object model represents the system terrifically. Although it does now show the flow in which information moves in, it gives the “how” by listing each classes separate functions. The attributes, functions, and variable names all help define how the system will work, which is something that is missing from the process model. The object model certainly makes the creation of each class from idea to written code much, much easier.