Anthony Dushaj

Professor Arias

Software Development I

26 April 2019

Milestone

The main objective of my Java project will be to create a simple virtual model of a shopping experience for the user, while accommodating their specific preferences and to minimize hardships and time wasted for the users shopping experience. My java project will consist of about four primary different classes, and 1 class for user interface.

The first class will be the Customer (the user), the first class will consist of a customer ID (int), customer Name (String), and lastly customer Budget (int) and also dateCreated (date). Those attributes will be given to any customer, they will enter their name and budget (minimum $0) themselves as those are contrary to everybody. The next class will be called product. The product class will consist of a name (string), price (int), product id (int), and a stock (int). Furthermore, the next class will be called products and will consist of an ArrayList of objects, and the role of it is to provide the user with the store products. Moreover, the products class is used to initialize objects for the class and will use a method called store products to create a list of the object product. The shopping cart is the next class which is a main part of this project, and the shopping cart will consist of a various different amount of methods including a method to add an item, remove an item, find a specific item and print its details (if the item is not found, then throw a Product Not Found Exception()), get the total number of items in the cart, print all the items in the cart, sort the products in the list by their price, name, or quantity. To conclude,

the shopping cart along with the user Interface class will do most of the work within the program. The last class will consist of a class and their properties devoted specifically for user interface, this is the class where the user interacts directly with the given java application, and where the user becomes in control of the facilitated shopping experience.

This program is creating a list of objects (ArrayList to make things simpler) (of the product class) that the user creates by adding the items they want, with this list of items the user can view, access and alter their shopping cart, by deciding if they want to remove items, add items, and even tell them how much of their budget is left. The users budget is an attribute of the customer class and the limit of it will not be allowed to be exceeded. The program will create an initialized shopping cart and will allow users to decide if they want the items or not, and the user will be able to add items by their item ID so that they make sure they can get the ideal products that they want. The shopping cart will throw a ItemNotFoundException() if the product that the user searches for is not on their cart, and will tell the user to add a different item.

The program is addressing the problem of people having a hard time shopping and forgetting items that they want (they can search their carts for items), removing unnecessary duplicates of items (remove an item), allowing the user to keep track of their cart and budget while shopping (check price / budget while shopping), counting the number of items (count items) so that their is not too many or too little.

The system should be used by the user entering in their specifics information, and from there allowing the user to have a shopping experience that is facilitated through programming methods, and classes that are described above and in the UML diagrams that I have created for the project.

Lastly, the main goal is to allow users to mimic an actual shopping experience, but excluding some of the hardships that come with shopping, while on the other hand also creating algorithms and methods in my program to allow this facilitated experience for the user to take place.

Customer
,name (string) , budget price (int) , customer id (int) ,date created - (date)
getCustomerName() getBudgetPrice() getCustomerID() setCustomerName() setBudgetPrice() setCustomerID()

Product
name (string), price (int), product id (int), stock (int).
getProductName() getProductPrice() getProductID() getProductStock()

Products
Products (objects)
storeItems() getProducts()

Shopping Cart
Number of Items - (int) Cart Total $ - (int) Date Created - (date)
addItemByID() removeItemByID() getItemNameByID() getItemPrice() searchItemByID() ItemNotFoundException() printCartItems() getNumberOfItems()

findMostExpensiveItem() isThereMoreThanOneItem() sortbyItemPrice() sortbyItemName()

User Interface
displayScreen() itemsMenu() Menu() displayItems() addItems() removeItems() getUserInput()