Xinyuan Pan
XP572
Chang Park
CWP639

Assignment 3

Analysis and Assumptions:

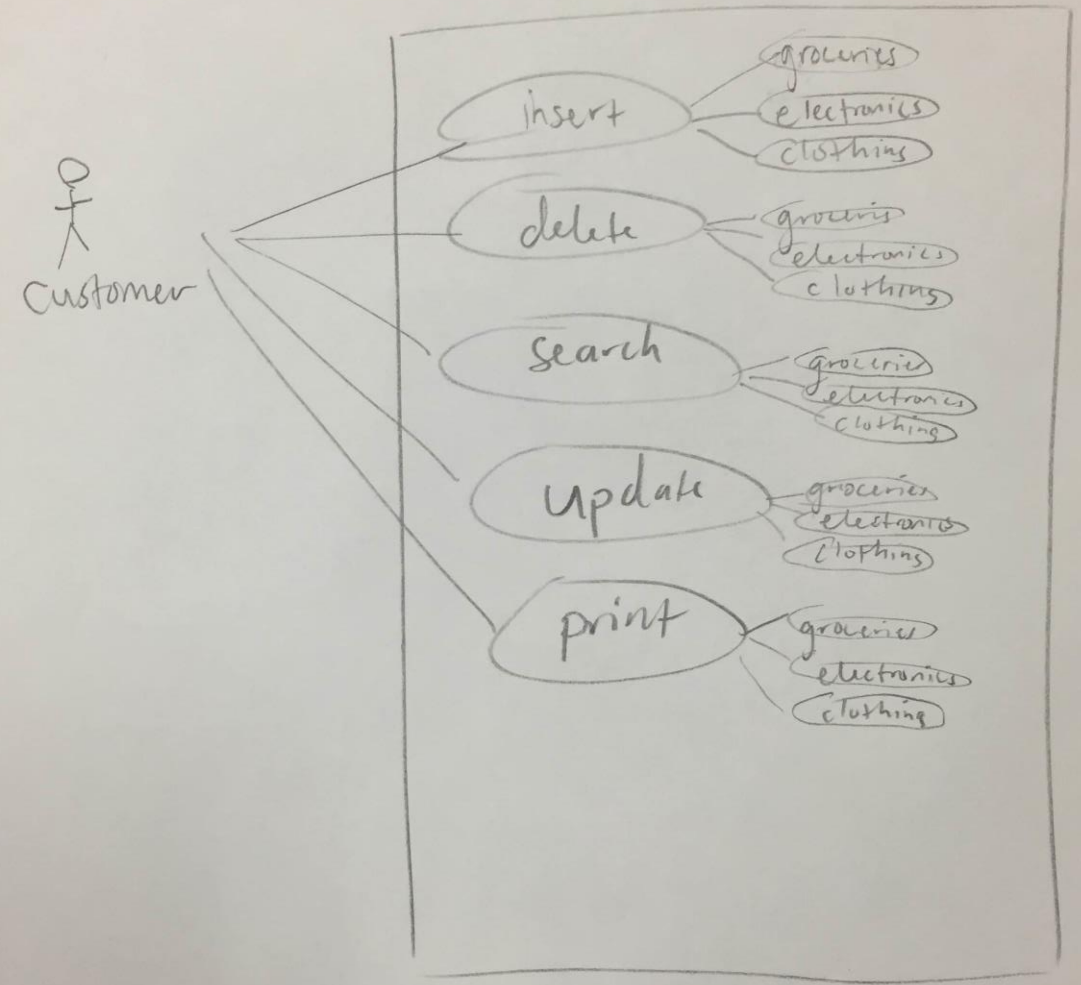
My "plan of action":
1. Analyze the problem and list assumptions (and stay consistent to these assumptions throughout)
2. Plan out my solution in the form of pseudo-code or a block diagram
3. Implement my solution

This program is a model of a shopping cart. Our goal is to design a working program where we can manipulate the items in the shopping cart with certain operations. We need to complete the Item classs and its three subclasses – Electronics, Grocery and Clothing. Then we need to design a method in the Driver class that reads the input and complete the corresponding operation to the shopping cart (which is implemented as an ArrayList).

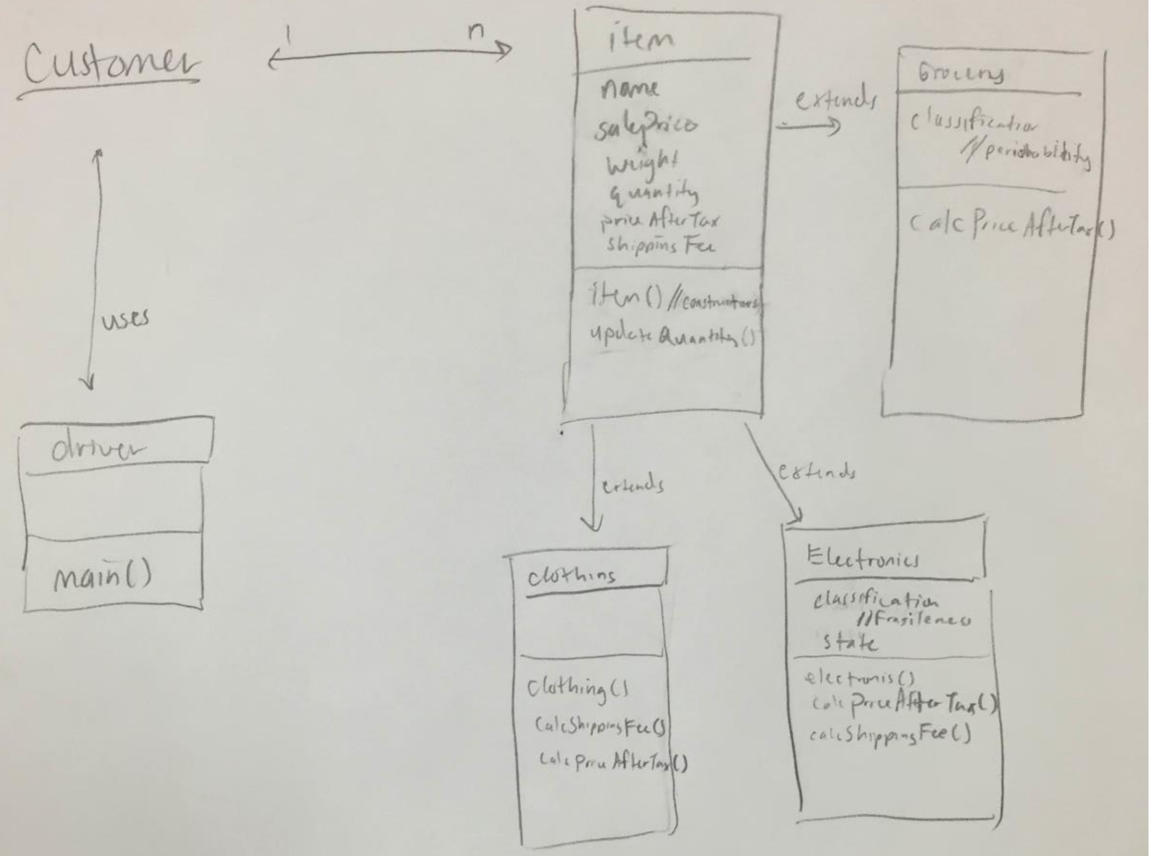
There are 5 different operations that can be performed on the cart:
Insert: creates a new object for a new item and adds to shopping cart array list
Delete: deletes all entries of the given item name
Search: outputs the number of instances of the given item name in the shopping cart
Update: updates the quantity value for a given item name
Print: prints the contents of the shopping cart, listing out each item as well as all of their attributes, and then prints out a total price for all items in the cart

Since this is a partner project, we will be working in pairs, working through git. The URL for our assignment is https://github.com/cwp639/EE422C-Assignment-3-Shopping-Cart
We in general plan to work together in person, but at times when that is not possible, changes will be uploaded to git, with a comment as to where in the program the changes were, and comments inside the program explaining the changes.

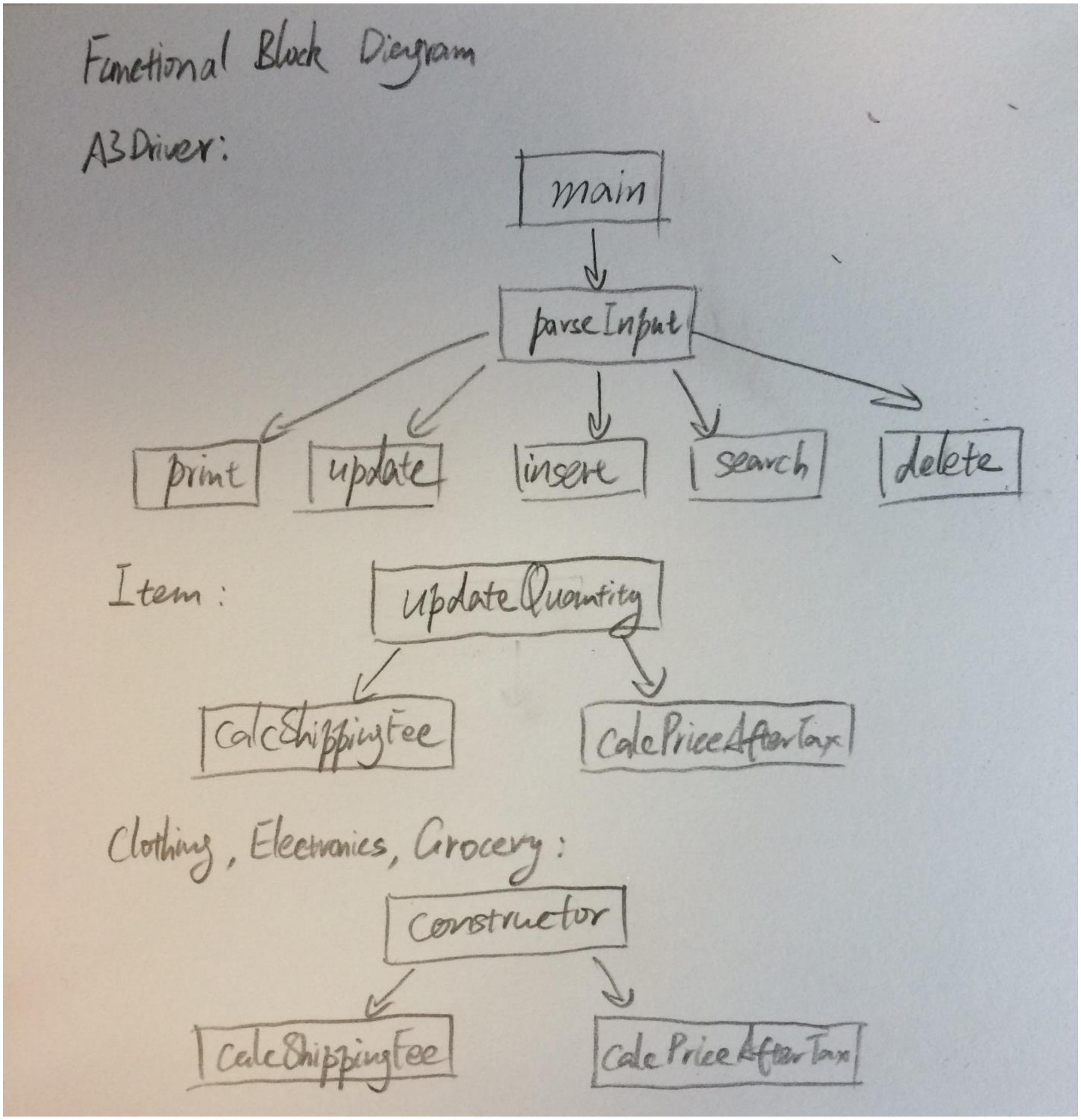
Use-Case Diagram



UML Class diagram



Functional Block Diagram



Pseudocode

```
main
  read the file
  LOOP through each line in the file
    parseInput()

parseInput()
  split input into an array of String
  check the first element in the array, and figure out which operation is it
  then direct to the corresponding method

insert()
  check the second element in the array to identity the category
  read the rest of the input and construct object of that category
  and add the objects into ArrayList shoppingCart

search()
  LOOP through the shoppingCart and check if the object.name is equal to the
  String suggested by the input
    IF yes then count += object.quantity
  print the statement showing how the count

delete()
  make a copy of shoppingCart
  LOOP through the copy and check if the object.name is equal to the
  String suggested by the input
    IF yes then count += object.quantity and remove it from the shoppingCart
  print the statement showing how many were deleted

update()
  read in the name of the item to be updated and the new quantity
  LOOP through shoppingCart and find the Item by name
    call object.updateQuantity(new quantity) to update the quantity and the
    shipping fee and price after tax
    print the statement that quantity has been updated
    break
  print the statement that no matching item in the shoppingCart

print()
  make a copy of ShoppingCart
  use Collections.sort(copy) to sort the ArrayList into the desired order
  the specific sorting rules are specified by compareTo method in class Item
```

```
Loop through copy
   print each attribute of each item
print the total cost in the shopping cart
```