System Level Use Case Diagram:

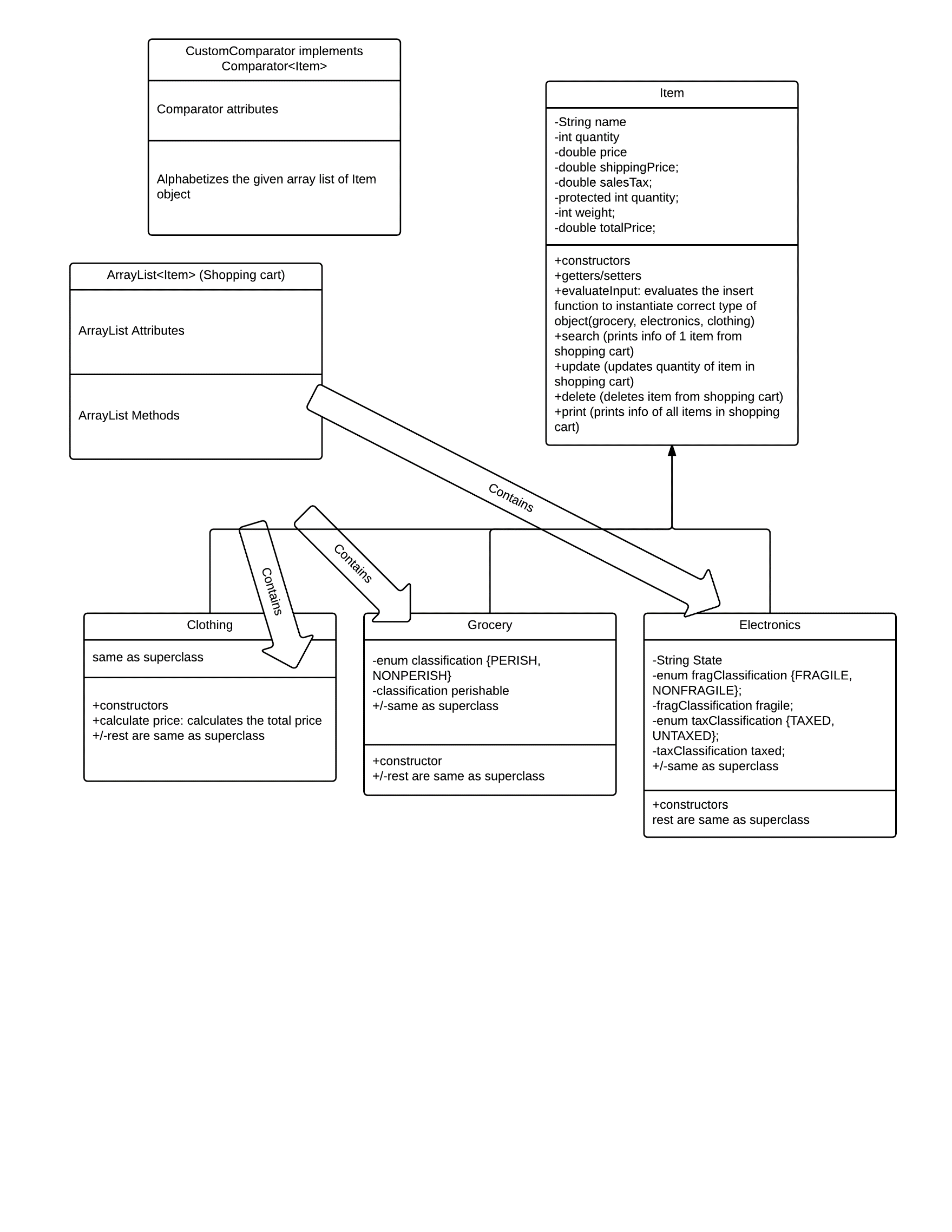
UML Model of needed classes and relationships

ADT Level Description of Each Class:

Block Diagram:

Driver Logic Algorithm

UML Class Diagram:



High level Algorithm for my Driver (main method):

1. Check if there is a command line input (.txt file with transaction list)
2. If it doesn’t:
   1. Print Incorrect number of command arguments
   2. End program
3. Create an ArrayList of Strings to store each line from the file with the transaction list (this ArrayList is called input)
4. Fill ArrayList with each line from the transaction list file (by calling processLinesInFile method, with input and file name as parameters)
5. Create a new ArrayList of Items called shoppingCart, to store various Items that will be added
6. Make an iterator out of the input ArrayList
7. Iterate through the input ArrayList doing this until there are no more elements:
   1. Split each element of input by the space (“ “) character, and store it in a new static size array; call it splitString
   2. Check splitString for any errors; i.e. make sure the input is valid under the rules described by the model details and problem statement (from the assignment 3 documentation). [Do this by calling the check errors method, with the splitString array as an input]
   3. If there is an error:
      1. Print “Invalid Input”
      2. Go to next element in iterator
   4. If the first element is an insert:
      1. Call evaluateInput method in item class with splitString and shoppingCart as parameters; this will add the proper object to the shoppingCart.
   5. Else If the first element is a search:
      1. Call search method in item class with splitString and shoppingCart as parameters; this will search for the item with the name specified in the splitString that is in the shoppingCart, and it will print the item’s info.
   6. Else If the first element is an delete:
      1. Call the delete method in the Item class; this method will search for all items matching the name in splitString and delete all values.
   7. Else if the first element is an update:
      1. Call the update method in the Item class; this will search for the first occurrence of the item with the name specified in splitString and it will change its quantity to the one specified in splitString
   8. Else if the first element is a print:
      1. Call the print method in the Item class; this will print information of items in shoppingCart and then print the final price