GROUP 6

ANWAR KAMIL (H00315787)
ROHIT SINGH JAMWAL (H00139801)
VISHNU SUBRAMANIAN (H00307008)
GABI MANSOUR (H00313528)
VIMAL VIDYADHARAN (H00316629)

COURSEWORK STAGE 1

GROUP REPORT

SCHOOL OF MATHEMATICAL AND COMPUTER SCIENCES

ADVANCED SOFTWARE ENGINEERING

DR MOHAMMAD HAMDAN

21st February 2019

1. Project Team - Name of Group Members and Summary

Project team responsibilities were defined and distributed as shown below:

Role	Team members
Analysis and Design	All members
Core libraries	Gabi Mansour, Anwar Kamil
GUI	Vimal
Quality Assurance	Vishnu Subramanian, Rohit Singh Jamwal
Documentation	Vishnu Subramanian, Rohit Singh Jamwal

The work was evenly distributed and carried out by all the members of the group. The project was planned and the development report was formed. The Core libraries were formed by Gabi Mansour and Anwar Kamil. The core libraries include ItemList.java, Item.java, OrderList.java and order.java. Vimal was completely in charge of the Graphical User Interface (GUI). He designed the GUI and wrote the code for it as well. Vishnu and Rohit were in charge of both the testing and forming the group report by first understanding each and every class and code.

2. Link to Repository

We have used GitHub as our repository for the project (https://github.com/foodcart)

The reasons for choosing it were pretty simple. Github is the most popular version control software available free of cost to the public. It made it very easy for us to contribute and work on the project simultaneously. It does so by allowing users to download a copy of the code to their machines, work on it, make changes and upload the final version back to the central repository. It interfaces well with the Terminal in Windows/Mac, and also links well with the Eclipse IDE we used to compile the code. Lastly, GitHub also helps with the documentation and allows you to easily track changes in code across versions.

3. Specification/Requirements

All the requirements and specifications described in the coursework have been met completely. All the steps have been taken to complete the coursework.

4. UML Diagrams

ACTIVITY DIAGRAM - Describing the Process from start to finish (Discount or no Discount)

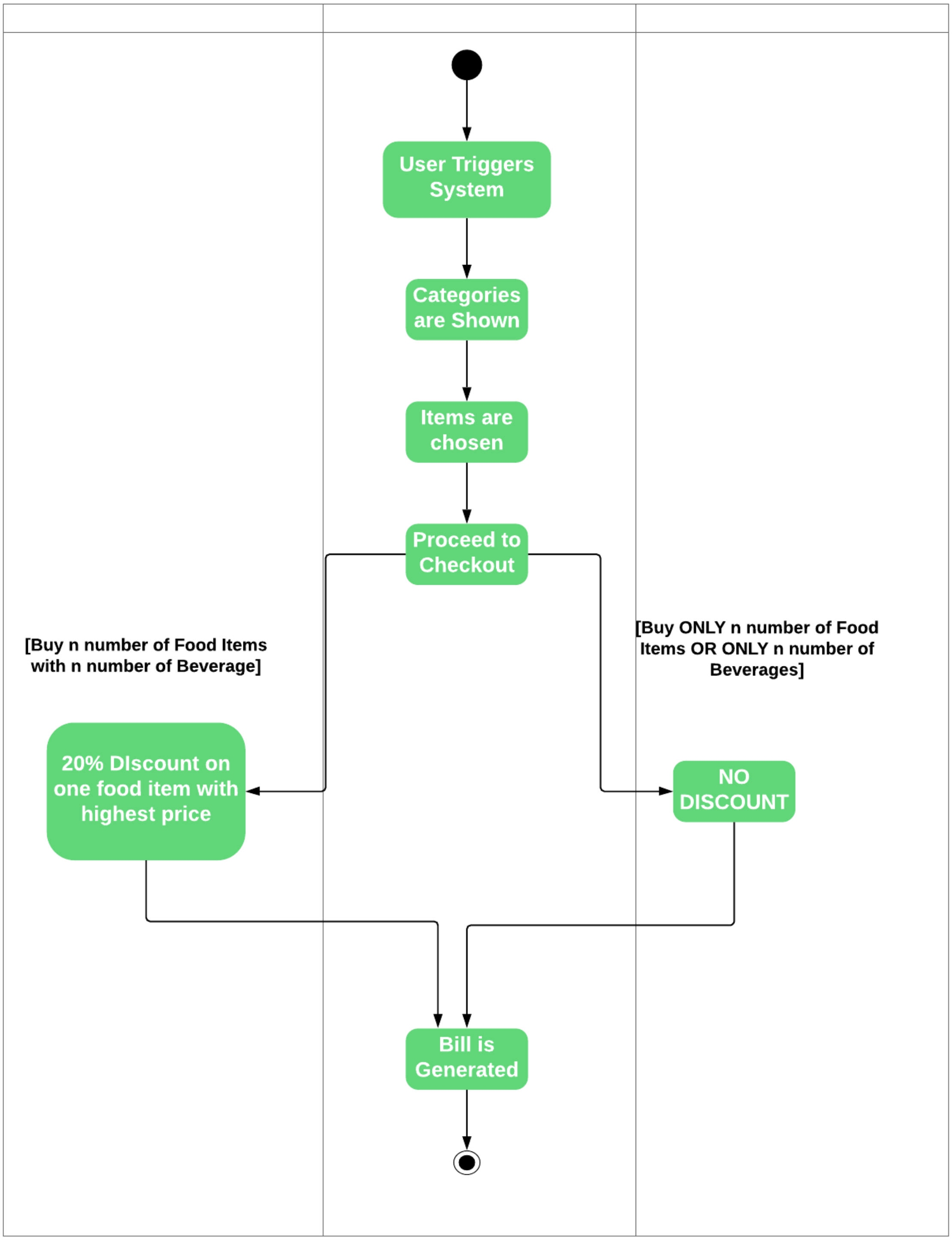

User Triggers System
Categories are Shown
Items are chosen
Proceed to Checkout
[Buy n number of Food Items with n number of Beverage]
[Buy ONLY n number of Food Items OR ONLY n number of Beverages]
20% DIscount on one food item with highest price
NO DISCOUNT
Bill is Generated

Figure: Activity Diagram

CLASS DIAGRAM - Describing the relationship between the classes (ShopGUI and the core classes)

To view the magnified image in more detail:

https://drive.google.com/file/d/1xdvBxM91BAkUJBd9stc0XWXbz8C5HKEr/view

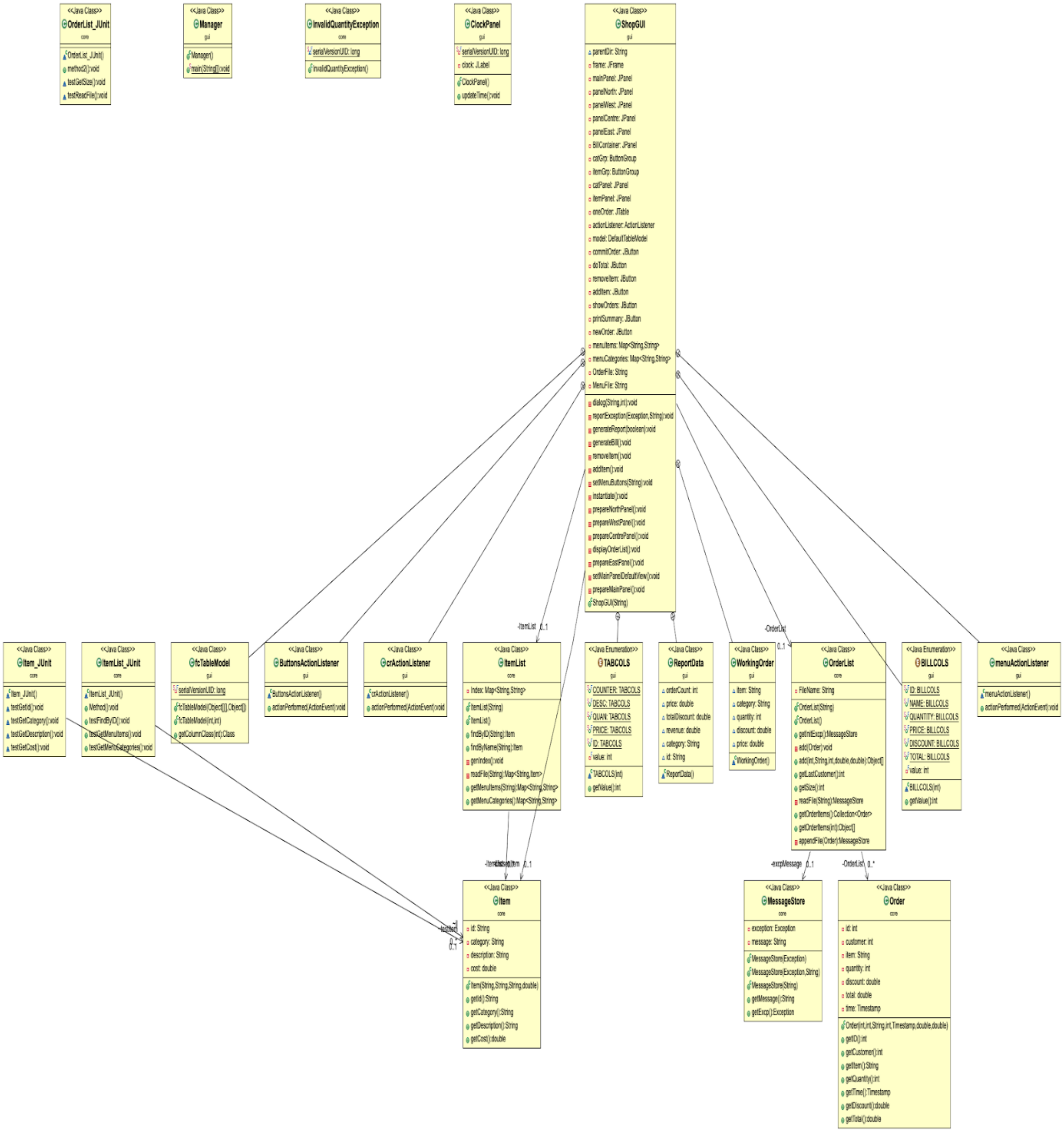


Figure: Class Diagram

5. Explanation of Data Structures Used

Data Structures used -

Component	Class	Data Structure
Item	ItemList	HashMap
Order	OrderList	TreeMap

We chose the above data structures for the following reasons:

Item: In case of Items, **HashMap** is used to store the data where Id (unique identifier) acts as the key to uniquely identify each item so as to facilitate the billing process because each time the bill is generated, this item list (Menu) is accessed thus requiring faster and efficient retrieval which is a property of HashMap data structure. It is implemented in the itemlist class where the key is the item id which is a concatenated string of the automatically generated item ID and the category. Value is the whole object of the itemlist class.

For Orders, we had decided to use HashMap at first to store the data where orderId acted as the key. Also, if and when the same customer orders again, it is treated as a new order thus we needed to uniquely identify orders which could be efficiently done using a HashMap. But we changed our plan to as we wanted a sorted list of orders and therefore we used **TreeMap**.

6. Functionality Of The Program

TAKING ORDER AND GENERATING BILL:

Whenever order is generated. It prints the order and it also goes directly to the file. At the end of transaction, a bill is generated. Order is stored to order.db file. This insures transactional integrity.

REPORT GENERATION:

This functionality shows the total income from all the orders and bills that have been made by using the application. This report is automatically generated when the application is closed.

SHOW ORDER:

Show order button shows the history of all the previous orders placed through the application. It shows the timestamp, order number, customer number, item, unit price , discount and total price.

DISCOUNTS:
Discount Rule:

Coffee Shop has a Menu with items from three categories. The categories are - Hot Beverage, Cold Beverage and Food Items. To delight the customers, Coffee Shop management has has decided to give a 20% discount on a food item if it is purchased with a beverage. If multiple food items are purchased with multiple beverages, the system gives the 20% discount on just one food item with the highest price value.

7. Testing Method and Exceptions

EXCEPTIONS -

Class name: ItemList

Following types of exceptions are being handled in this class:

1) **FileNotFoundException:**

   This exception is handled if no such file with the name mentioned in the parameter for menulist file passed is found in project directory.

2) **NumberFormatException:**

   This is used if a string is passed in place of an integer.

3) **NullPointerException:**

   This exception is thrown if we try to make an itemlist object and it is not able to make an object due to a missing parameter.

Class name: OrderList

1. **InvalidQuantityException**

   It throws error when the quantity of the same item's order is less than 1.

2. **FileNotFoundException**

   This exception is handled if no such file with the name mentioned in the parameter for orderlist file passed is found in project directory.

3. **NumberFormatException**

   This is used if a string is passed in place of an integer.

4. **DateTimeParseException**

   We are keeping a record the order time here. If we are not able to parse it to a string, this exception will be thrown.

5. **ArrayIndexOutOfBoundsException**

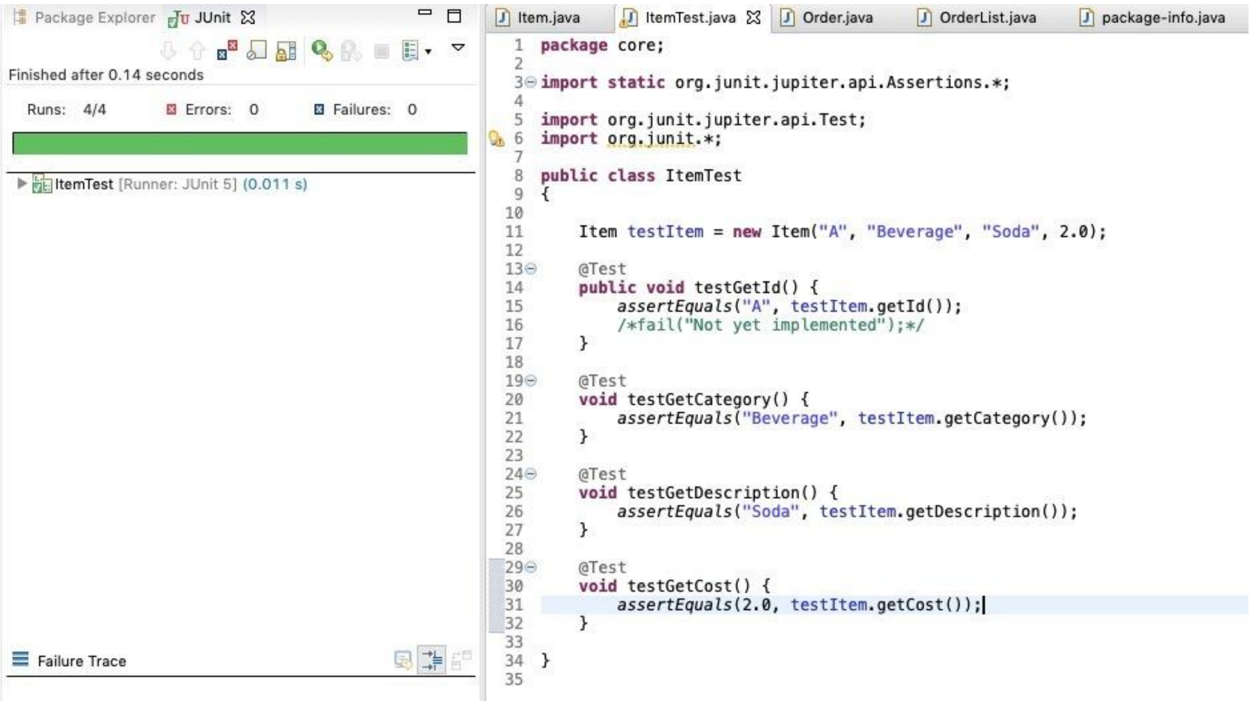
   This exception takes care of an error when the number of elements that is read is actually bigger than the number of elements defined in the array used for reading the values.

6. **NullPointerException**

   This exception is thrown if we try to make an orderlist object and it is not able to make an object due to a missing parameter.

THE **TESTING METHOD** USED FOR THIS PROGRAM IS **JUNIT TESTING.**

Class	Method	Test Condition	Response
Item	getId()	String	Passed
Item	getCost()	Double	Passed
ItemList	findByID()	String	Passed
ItemList	GetMenuItems()	Map <String, String>	Not Passed



```
Finished after 0.14 seconds

Runs: 4/4    Errors: 0    Failures: 0

ItemTest [Runner: JUnit 5] (0.011 s)
```

```java
package core;

import static org.junit.jupiter.api.Assertions.*;

import org.junit.jupiter.api.Test;
import org.junit.*;

public class ItemTest
{

    Item testItem = new Item("A", "Beverage", "Soda", 2.0);

    @Test
    public void testGetId() {
        assertEquals("A", testItem.getId());
        /*fail("Not yet implemented");*/
    }

    @Test
    void testGetCategory() {
        assertEquals("Beverage", testItem.getCategory());
    }

    @Test
    void testGetDescription() {
        assertEquals("Soda", testItem.getDescription());
    }

    @Test
    void testGetCost() {
        assertEquals(2.0, testItem.getCost());
    }

}
```

Figure: The green bar on the left shows that these JUnit Tests passed successfully.

GUI USER MANUAL

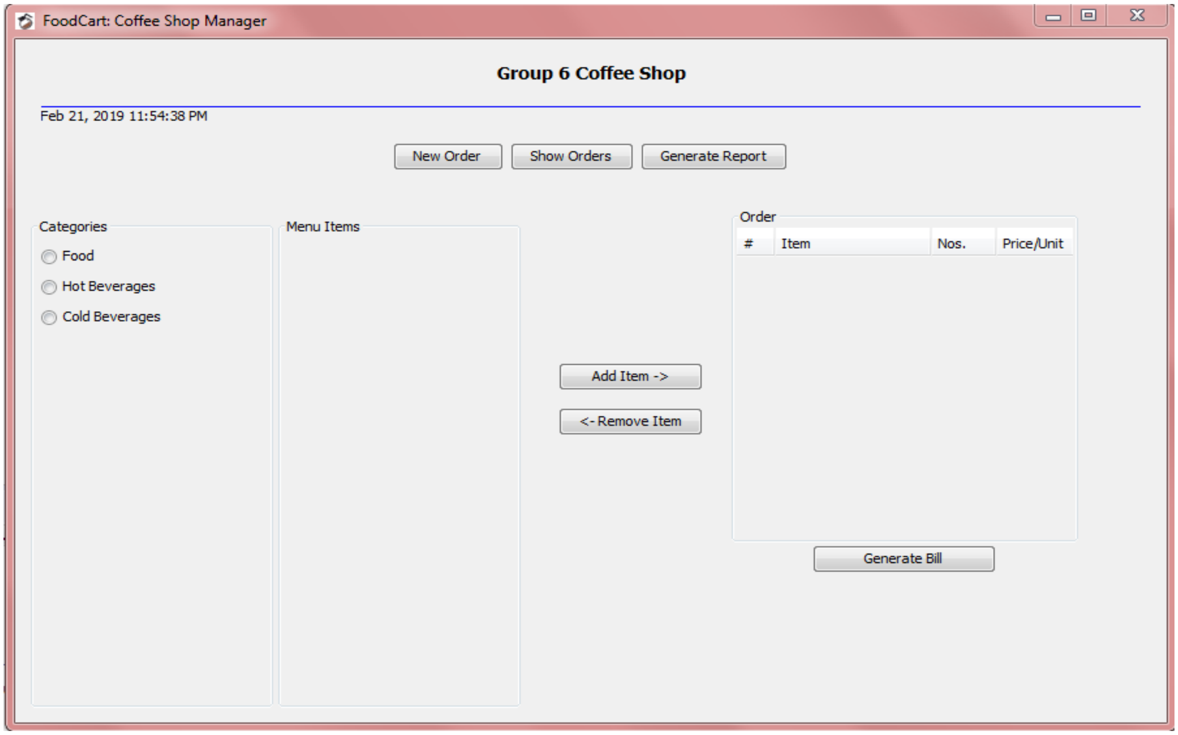
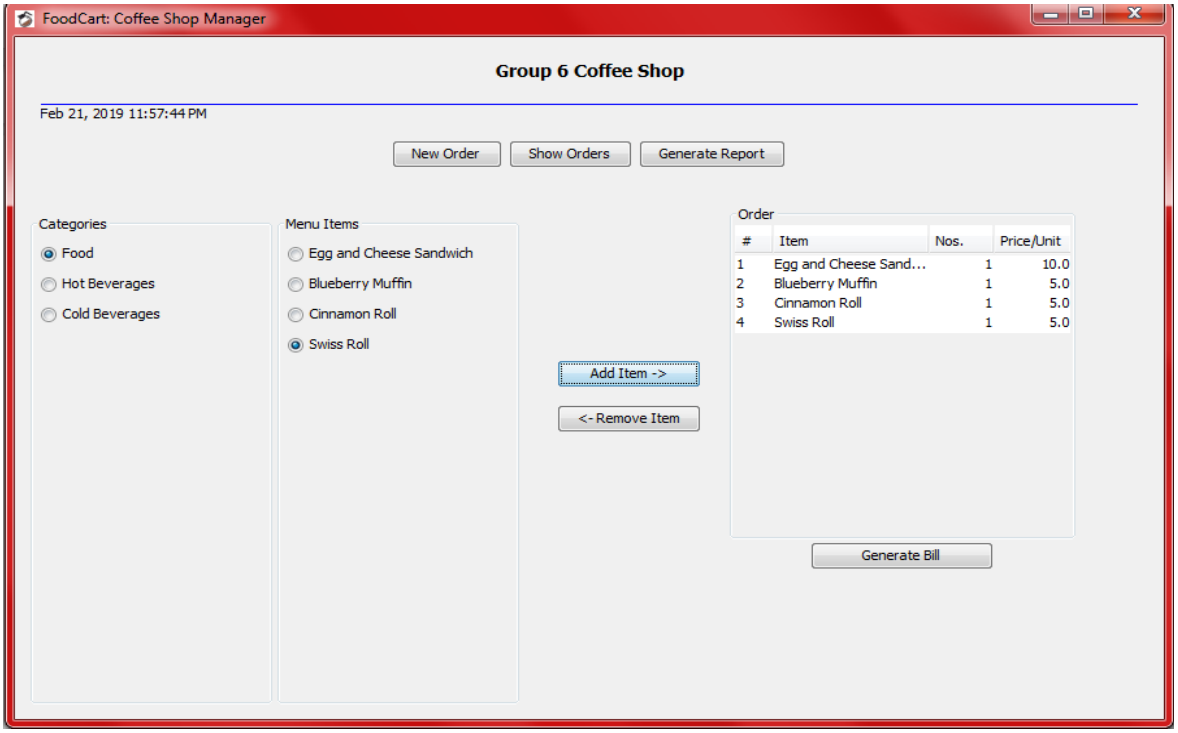
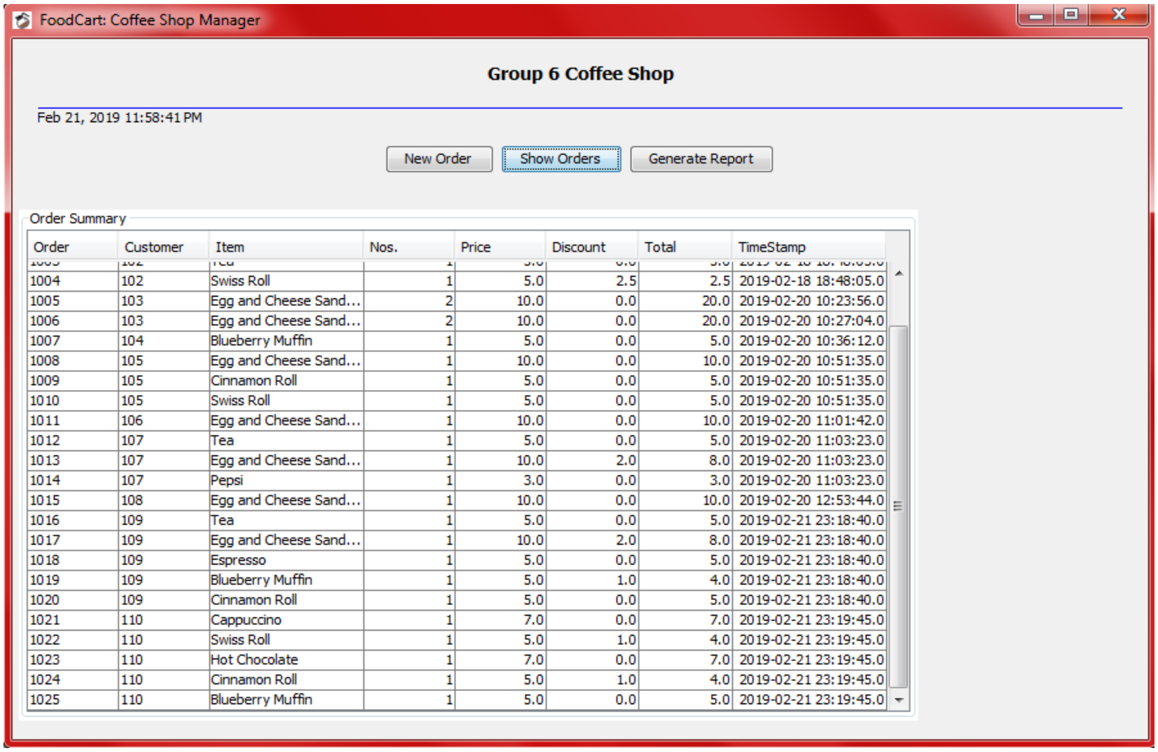


Figure: The figure above shows the FoodCart application



#	Item	Nos.	Price/Unit
1	Egg and Cheese Sand...	1	10.0
2	Blueberry Muffin	1	5.0
3	Cinnamon Roll	1	5.0
4	Swiss Roll	1	5.0

Figure: On selecting items and Pressing Add Item, the items are shown in the list of orders on the right hand side. There is also a functionality to remove items.



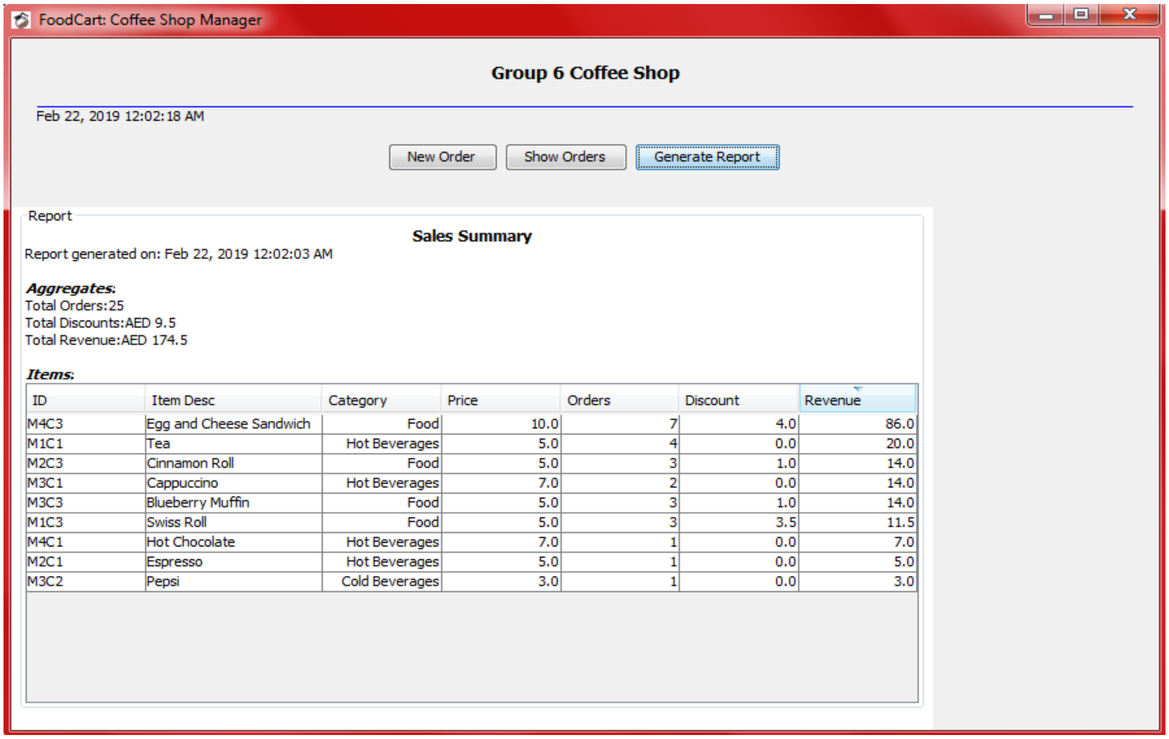
Group 6 Coffee Shop

Feb 21, 2019 11:58:41 PM

Order Summary

Order	Customer	Item	Nos.	Price	Discount	Total	TimeStamp
1004	102	Swiss Roll	1	5.0	2.5	2.5	2019-02-18 18:48:05.0
1005	103	Egg and Cheese Sand...	2	10.0	0.0	20.0	2019-02-20 10:23:56.0
1006	103	Egg and Cheese Sand...	2	10.0	0.0	20.0	2019-02-20 10:27:04.0
1007	104	Blueberry Muffin	1	5.0	0.0	5.0	2019-02-20 10:36:12.0
1008	105	Egg and Cheese Sand...	1	10.0	0.0	10.0	2019-02-20 10:51:35.0
1009	105	Cinnamon Roll	1	5.0	0.0	5.0	2019-02-20 10:51:35.0
1010	105	Swiss Roll	1	5.0	0.0	5.0	2019-02-20 10:51:35.0
1011	106	Egg and Cheese Sand...	1	10.0	0.0	10.0	2019-02-20 11:01:42.0
1012	107	Tea	1	5.0	0.0	5.0	2019-02-20 11:03:23.0
1013	107	Egg and Cheese Sand...	1	10.0	2.0	8.0	2019-02-20 11:03:23.0
1014	107	Pepsi	1	3.0	0.0	3.0	2019-02-20 11:03:23.0
1015	108	Egg and Cheese Sand...	1	10.0	0.0	10.0	2019-02-20 12:53:44.0
1016	109	Tea	1	5.0	0.0	5.0	2019-02-21 23:18:40.0
1017	109	Egg and Cheese Sand...	1	10.0	2.0	8.0	2019-02-21 23:18:40.0
1018	109	Espresso	1	5.0	0.0	5.0	2019-02-21 23:18:40.0
1019	109	Blueberry Muffin	1	5.0	1.0	4.0	2019-02-21 23:18:40.0
1020	109	Cinnamon Roll	1	5.0	0.0	5.0	2019-02-21 23:18:40.0
1021	110	Cappuccino	1	7.0	0.0	7.0	2019-02-21 23:19:45.0
1022	110	Swiss Roll	1	5.0	1.0	4.0	2019-02-21 23:19:45.0
1023	110	Hot Chocolate	1	7.0	0.0	7.0	2019-02-21 23:19:45.0
1024	110	Cinnamon Roll	1	5.0	1.0	4.0	2019-02-21 23:19:45.0
1025	110	Blueberry Muffin	1	5.0	0.0	5.0	2019-02-21 23:19:45.0

Figure: On Clicking the Show Orders button, a text Box is generated with all the orders placed till now.



Group 6 Coffee Shop

Feb 22, 2019 12:02:18 AM

Report

Sales Summary

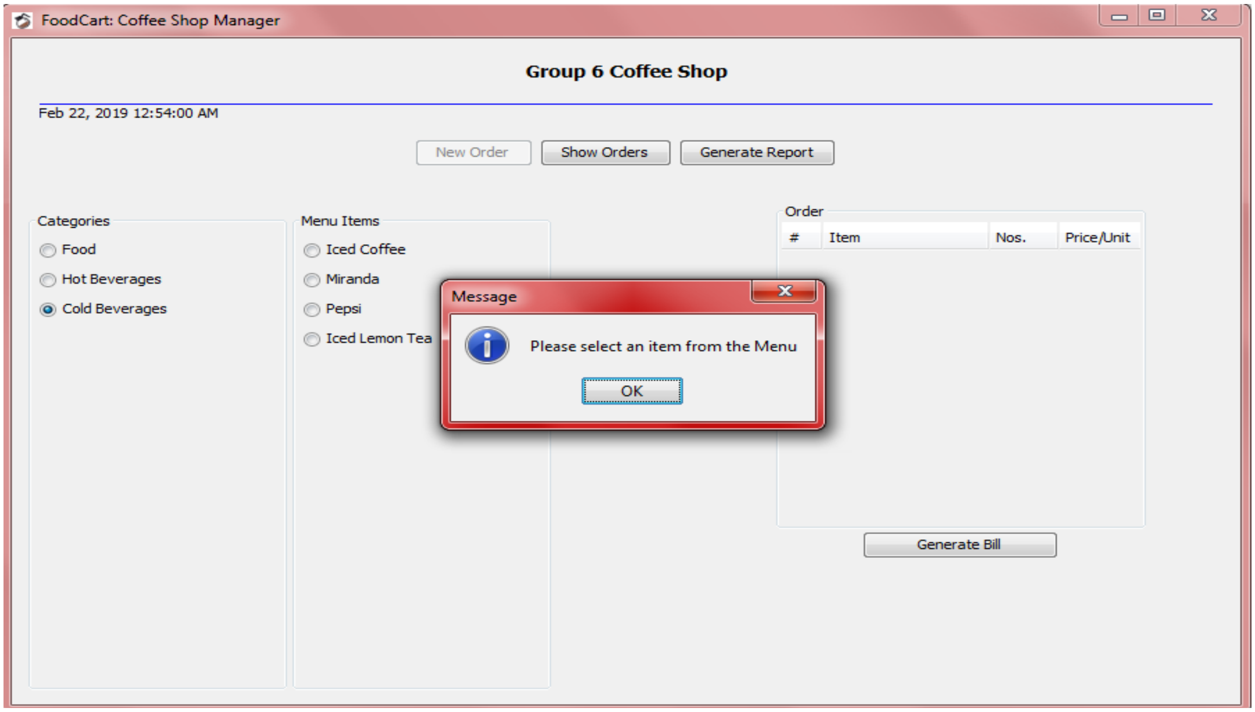
Report generated on: Feb 22, 2019 12:02:03 AM

Aggregates:
Total Orders:25
Total Discounts:AED 9.5
Total Revenue:AED 174.5

Items:

ID	Item Desc	Category	Price	Orders	Discount	Revenue
M4C3	Egg and Cheese Sandwich	Food	10.0	7	4.0	86.0
M1C1	Tea	Hot Beverages	5.0	4	0.0	20.0
M2C3	Cinnamon Roll	Food	5.0	3	1.0	14.0
M3C1	Cappuccino	Hot Beverages	7.0	2	0.0	14.0
M3C3	Blueberry Muffin	Food	5.0	3	1.0	14.0
M1C3	Swiss Roll	Food	5.0	3	3.5	11.5
M4C1	Hot Chocolate	Hot Beverages	7.0	1	0.0	7.0
M2C1	Espresso	Hot Beverages	5.0	1	0.0	5.0
M3C2	Pepsi	Cold Beverages	3.0	1	0.0	3.0

Figure: On Clicking the Generate Report button, a Final Report is generated with all the orders placed till now along with the total revenue and total discounts made.



Group 6 Coffee Shop

Feb 22, 2019 12:54:00 AM

Categories

○ Food
○ Hot Beverages
● Cold Beverages

Menu Items

○ Iced Coffee
○ Miranda
○ Pepsi
○ Iced Lemon Tea

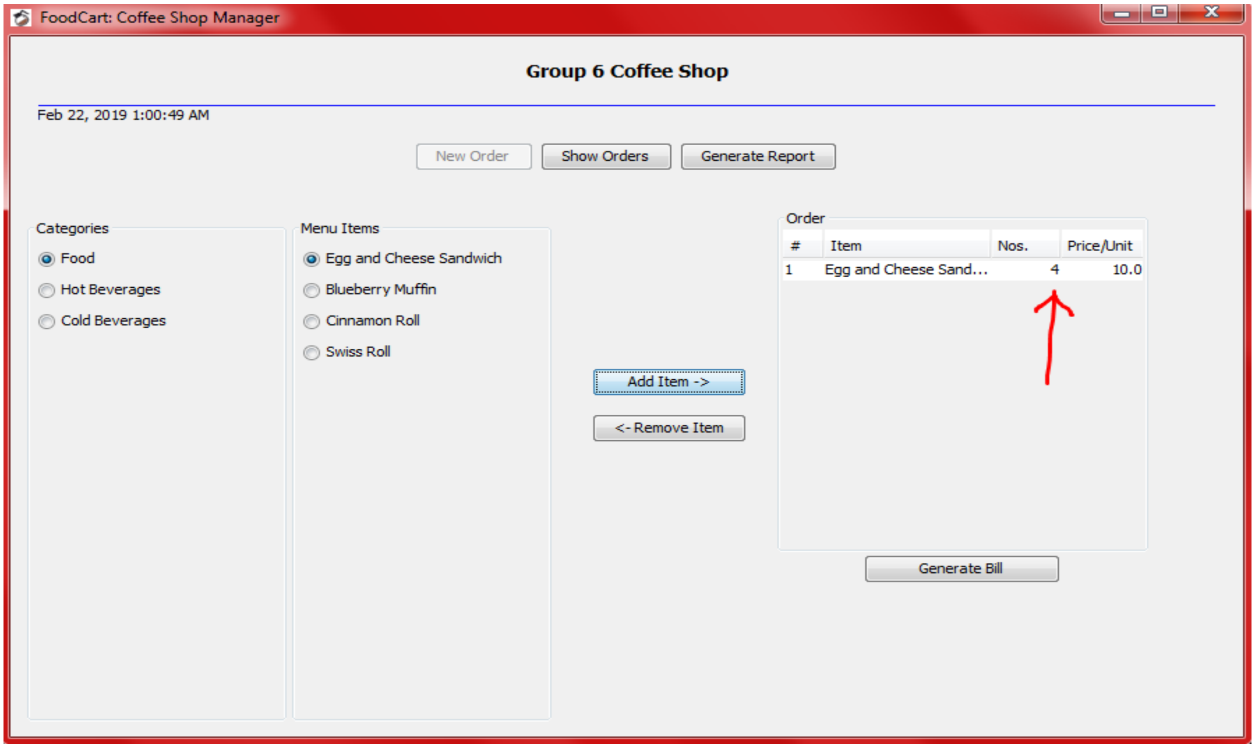
Message

Please select an item from the Menu

Order

#	Item	Nos.	Price/Unit

Figure: If the button “Add Item” is pressed without selecting any item from the list, a pop up box appears asking the user to select an item from the menu



Group 6 Coffee Shop

Feb 22, 2019 1:00:49 AM

Categories

● Food
○ Hot Beverages
○ Cold Beverages

Menu Items

● Egg and Cheese Sandwich
○ Blueberry Muffin
○ Cinnamon Roll
○ Swiss Roll

Order

#	Item	Nos.	Price/Unit
1	Egg and Cheese Sand...	4	10.0

Figure: When the same item is added more than once, Its quantity in the order list is increased rather than adding another row below.