CSIT121 Final Project
Sean Overton
SN: 6421490

1)UML Class Diagram:

CafeOrderSystem
-order: Order
-mainFrame: MenuFrame
-printer: Printer
+main(String[] args): void
+createMembersMenu(): void
+createNonMembersMenu(): void
+getMainFrame(): MenuFrame
+getOrder(): Order
+addItems(): void
+reset(): void
+getPrinter(): Printer
+printTextFile(): Printer

Order
-totalPrice: double
-items: ArrayList<Item>
-customerName: String
-tableNumber: int
+Order()
+setCustomerName(String): void
+getCustomerName(): void
+setTableNumber(int): void
+getTableNumber(): int
+getItem(int): Item
+addItem(Item): void
+getTotalPrice(): double

Item
-name: String
-description: String
-price: double
-quantity: int
-dietInfo: String
+Item()
+Item(String, String, double)
+Item(String, String, double, String)
+setName(String): void
+getName(): String
+setDescription(String): void
+getDescription(): String
+setPrice(double): void
+getPrice(): double
+setQuantity(int): void
+getQuantity(): int
+getDietInfo(): String
+setDietInfo(): String
+clone(): Item
+toString(): String

Printer
-output: Formatter
+openTextFile(String, String): void
+saveOrder(): void
+closeTextFile(): void
+getFormatter(): Fomratter

MenuFrame
-layout: GridBagLayout
-membersMenu: Menu
-nonMembersMenu: Menu
-firstPanel: FirstPanel
-secondPanel: SecondPanel
-thirdPanel: ThirdPanel
-fourthPanel: FourthPanel
+MenuFrame()
+getMembersMenu(): Menu
+setMembersMenu(Menu): void
+getNonMembersMenu(): Menu
+setNonMembersMenu(Menu): void
+getFirstPanel(): FirstPanel
+getSecondPanel(): SecondPanel
+getThirdPanel(): OrderPanel
+getFourthPanel(): ButtonPanel
+successfulSubmitHandler(): void
+unsuccessfulSubmitHandler(): void

Menu
-pizzas: ArrayList<Item>
-burgers: ArrayList<Item>
-sides: ArrayList<Item>
+Menu()
+getPizzas(): ArrayList<Item>
+getBurgers(): ArrayList<Item>
+getSides(): ArrayList<Item>
+setPizzas(ArrayList<Item>): void
+setBurgers(ArrayList<Item>): void
+setSides(ArrayList<Item>): void

ItemPanel
-layout: BorderLayout
-title: JLabel
-options: JList
-optionsModel: DefaultListModel
-listener: ItemSelectionListener
+ItemPanel()
-class: ItemSelectionListener
+setTitle(String): void
+changeOptions(ArrayList<Item>): void
+reset(): void
+getValue(): Item
+disableListener(): void

ThirdPanel
-layout: FlowLayout
-title: JLabel
-order: JTextArea
-quantityPanel: InputPanel
-addButton: JButton
+ThirdPanel()
+update(String): void
+reset(): void
+getQuantityPanel(): InputPanel
+getAddButton(): JButton

InputPanel
-layout: FlowLayout
-title: JLabel
-input: JTextField
+InputPanel()
+InputPanel(String)
+setTitle(String): void
+reset(): void
+getValue(): String

FirstPanel
-layout: FlowLayout
-namePanel: InputPanel
-tableNumberPanel: InputPanel
-memberPanel: MemberPanel
+NorthPanel()
+getNamePanel(): InputPanel
+getTableNumberPanel(): InputPanel
+getMemberPanel(): MemberPanel
+validateCustomerName(): void
+validateTableNumber(): void
+reset()

SecondPanel
-layout: FlowLayout
-categoryPanel: CategoryPanel
-itemPanel: ItemPanel
-descriptionPanel: DescriptionPanel
+SecondPanel()
+getCategoryPanel(): CategoryPanel
+getDescriptionPanel(): DescriptionPanel
+getItemPanel(): ItemPanel
+reset(): void

CategoryPanel
-layout: BorderLayout
-title: JLabel
-options: JList
-optionsModel: DefaultListModel
-listener: CategorySelectionListener
+CategoryPanel(String[])
-class: CategorySelectionListener
+reset(): void
+getValue(): String

FourthPanel
-layout: FlowLayout
-submitButton: JButton
-resetButton: JButton
+FourthPanel()
+getSubmitButton(): JButton
+getResetButton(): JButton

DescriptionPanel
-layout: BorderLayout
-title: JLabel
-description: JTextArea
+DescriptionPanel()
+changeValue(String): void
+reset(): void

AddButtonHandler
+actionPerformed(ActionEvent): void

ResetButtonHandler
+actionPerformed(ActionEvent): void

SubmitButtonHandler
+actionPerformed(ActionEvent): void

MemberPanel
-final layout: FlowLayout
-final buttonGroup: ButtonGroup
-buttons: JRadioButton[]
+MemberPanel()
+getValue(): String
-class: ChangeMembership

2)GUI and Explanations:

-Design uses a combination of JFrame, JPanel (with layouts including GridBagLayout and FlowLayout), JTextField, JLabel, JRadioButton, JList and JButton to create the above GUI layout.

-It allows for a number of inputs ie. the customer name, table number and membership status as well as the items selection and quantity from the menu to add to the order

-the program is able to handle input errors

-it is also capable of generating a text file of the order upon "submission" or reset the order

3)Compilation and execution/testing:

Successful compilation and execution

```
C:\Users\Sean\Documents\CSI121 OOP\Final Project\Rev2>javac CafeOrderSystem.java

C:\Users\Sean\Documents\CSI121 OOP\Final Project\Rev2>java CafeOrderSystem
```

1. Enter name, table number and select membership.

2. Select category, item and quantity.

3. Add the first dish.

4. Add several more dishes.

5. Submit order and view text file.

Order was placed successfully!   ✕

(i)   Text file has been generated.

OK

Sean_26 - Notepad

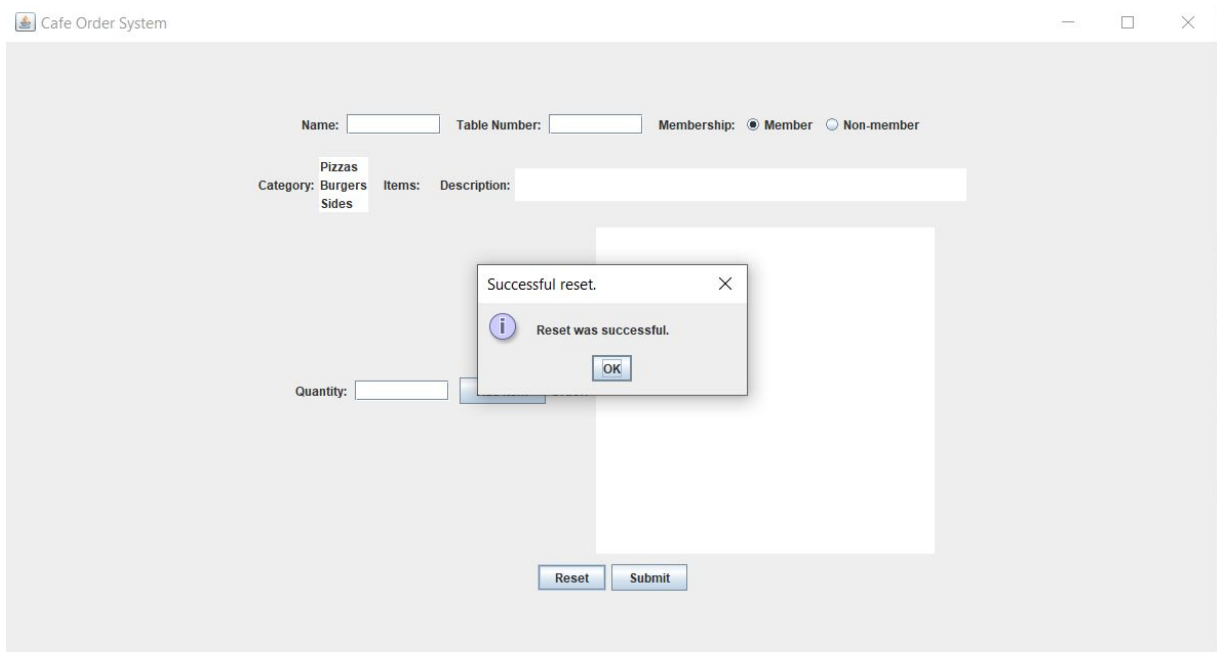File   Edit   Format   View   Help

Customer name: Sean, Table Number: 26

Tropicana Pizza   Quantity: 2 $20.0
BBQ Meat Lovers Pizza   Quantity: 1 $16.0
Steak Sandwich   Quantity: 1 $18.5
Side salad   Quantity: 3 $15.0   Vegetarian
Sweet potato fries (Bowl)   Quantity: 2 $13.0   Vegetarian

Total Price: $82.5

6. Reset the Order -popup notification and all text fields are blanked
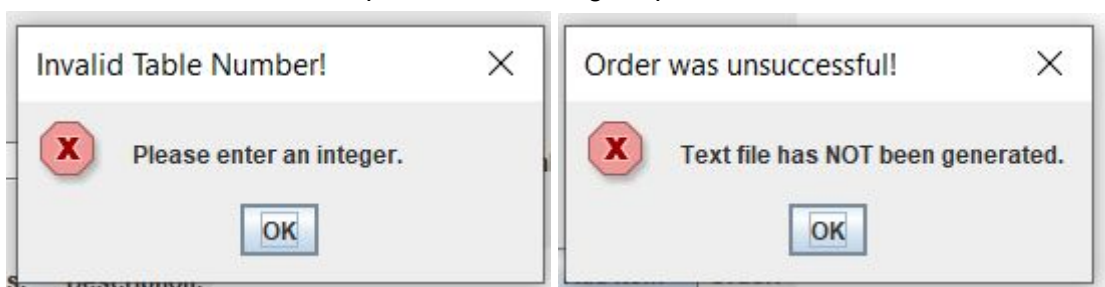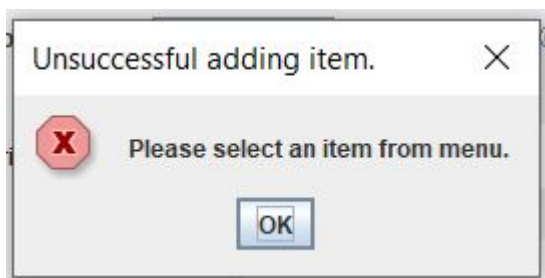

Input Validation:
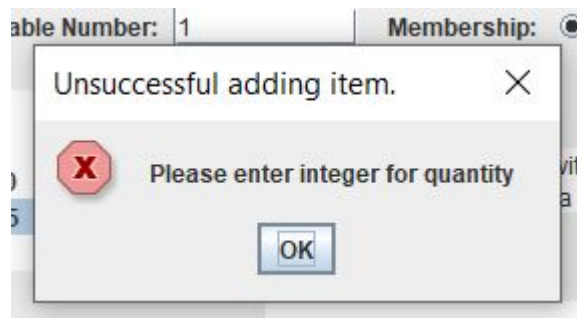1. No "Name" input upon "Submit" button pressed
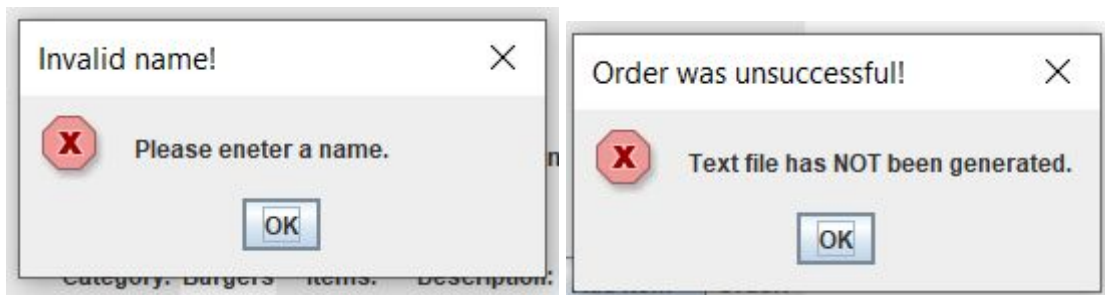
2. No "table number" input or invalid integer upon "Submission"

3. No Item selected when "add" button pressed

4. No "quantity" or not an integer when "add" button pressed

4)Text file generated within same directory: