REQUIRMENTS PLAN

PROFESSOR ADKISSON

Table of Contents

Requirements Design

Section 1: GUI

1.1 Main Screen

1.1.1 The main screen will have the following large buttons:
- Order Drink
- Customer List
- Management Tools

1.2 Customer List Screen

1.2.1 Sorted by last name, then first name, then phone number. The *anonymous* Customer should always be at the top of the list.

1.2.2 Includes number of reward points.

1.2.3 Includes a button to add a new Customer.

1.2.4 Each Customer has an “Order Drink” button.

1.2.5 Includes a button to return to the “Main” screen.

1.3 Add Customer Screen

1.3.1 Allows user to enter the Customer’s first name, last name, and phone number.

1.3.2 All fields are required to be filled out.

1.3.3 Verifies phone number is unique, and name is at least 10 characters in total.

1.3.4 Includes an “Add Customer” button and a “Cancel” button.

1.3.5 After adding a new Customer, go to the “Order Drink” screen.

1.3.6 After cancelling, return to the “Main” screen.

1.4 Payment Screen

1.4.1 10 Reward Points are worth $1.

1.4.1.1 Reward Points are rounded down.

1.4.1.2 Reward Points can only be used to redeem order total.

1.4.1.3 If there are enough Reward Points on the account, offer to use Reward Points for the purchase instead of a Credit Card. If yes, go to “Receipt” Screen. If no, pay with Credit Card.

1.4.2 Customer can pay with either Credit Card or Reward Points.

1.4.2.1 Credit Card validation on card number and expiration date.

Commented [HM1]: Should this be capitalized?

Commented [HM2]: First name - their first name

Commented [HM3]: Go instead of goes

Commented [HM4]: Talks about capitalizing and not capitalizing credit card and reward point

1.4.2.2 Fields for Credit Card number and expiration date.

1.4.3 Anonymous Customers do not receive Reward Points.

1.4.4 Go to “Receipt” Screen after successful payment.

1.4.5 Cancelling order goes back to “Main” screen.

1.5 Receipt Screen

1.5.1 Screen includes a list of purchased drinks, their cost, the drink customizations, and the customization costs, and the payment method.

1.5.2 Screen includes the subtotal, tax on the order, and the total.

1.5.3 Screen includes the number of Reward Points earned on this purchase and current amount.

Commented [HM5]: The current amount instead of how many the customer now has

1.5.4 Screen includes the date and time of the order.

1.6 Management Screen

1.6.1 Includes a button to generate a CSV report.

1.6.1.1 This includes all sales for the day and opens the file in Excel.

1.6.2 Includes a button to return to the “Main” screen.

1.7 General Buttons

1.7.1 There will be a “Return to Main Menu” button on every screen.

1.7.2 There will be a “Return to Previous Screen” button on every screen.

1.7.3 After returning to a previous screen or “Main” screen all data inputted past the screen returned to will be cleared.

1.7.4 When returning to a previous screen or cancelling an action, prompt the user with a confirmation button.

1.8 Order Drink Screen

1.8.1 Customer is anonymous if ordering drink from main screen.

Commented [HM6]: Putting in period

1.8.2 Customer can order one or more drinks.

1.8.3 Each drink can be customized; customizations affect the price.

Commented [HM7]: Use semi colon instead

1.8.4 This screen will have two panes.

1.8.4.1 Left side is the drink creator.

1.8.4.2 Right side shows the drinks added to order, plus subtotal, tax, and total.

1.8.5 Once there is one order added, the button to continue to checkout will become available.

1.8.6 Canceling the order returns you to the main screen.

Section 2: Database Management

2.1 Customer List and Sales Data

2.1.1 The Customer List and Sales Data are stored in the same JSON file.

2.1.2 Customer List

2.1.2.1 Customer records may be identified by inputting their phone numbers in the “Customer List”.

2.1.2.2 If this does not return a result, you can add a new record via the “Add New Customer” button.

2.1.2.3 When creating a Customer record, there are three text fields to enter the Customer’s first name, last name, and phone number. A Customer record cannot be created unless all three fields are filled in and valid.

2.1.2.4 Phone number must be a valid length, at least 10 characters. If the phone number entered is already in the “Customer List”, the record will not be created.

2.1.2.5 Each Customer record will have a unique, auto-incremented integer assigned to it.

2.1.2.6 Reward points are stored in the Customer record. When points are accrued or spent, this number will be altered.

2.1.2.7 The *anonymous* Customer record always appears first in the customer list. The first name is “Anonymous”, the last name is “Anonymous”, and the phone number is “000-000-0000”. This record is unable to gain or spend reward points.

2.1.3 Sales Data

2.1.3.1 Each receipt’s information is stored in the Customer List and Sales Data file. This includes all things related to it, such as the Customer’s ID, the cost breakdown of items, and the date/time of purchase.

2.1.3.2 Each drink has a name, all its customizations formatted specifically in CSV, and the total price of the drink.

2.2 Configuration Data

2.2.1 When the application is started, configuration data is read.

2.2.2 The Configuration Data file sets the tax rate and reward points earned per dollar in the application.

2.3 Drink Menu and Customization

2.3.1 Each unique drink base has its own set of customization attributes.

2.3.2 All drinks and their customizations are written in the JSON file, separate from the application. The application will not write to the Drink Menu JSON file.

Section 3: Payment and Order Handling

3.1 Credit Card Payments

3.1.1 Program verifies Credit Card expiration date, and the Credit Card number is valid.

3.2 Reward Point Payments

3.2.1 Verifies there are enough Reward Points on the account to pay for the entire purchase.

3.2.2 Deducts reward points from total.

Section 4: Program Architecture

4.1 The application will be made using C# and WinForms.

4.2 JSON files will be used for data storage.

4.3 The Decorator pattern will be used for drink customization.

4.4 Required NuGet Packages

4.4.1 Newtonsoft JSON will be used for reading and writing JSON data in the application.

4.4.2 CSVHelper will be used to generate CSV files, such as the end-of-day sales reports.

4.4.3 CreditCardValidator will be used to verify credit card is not expired and that it is a valid number.

Section 5: Nonfunctional Requirements

5.1 Compatibility

5.1.1 The application should be compatible with any recent (Windows 7 and beyond) version of Windows.

5.2 Performance

5.2.1 Increasing the Customer List length should not increase response times to any more than 2 seconds.

5.3 Usability

5.3.1 The application should have an uncluttered, straightforward design that is easy to learn. Toggles should be apparent, buttons are clearly marked and informative, and required information is clearly displayed.

Commented [HM8]: Verifies creditcard expiration date

Commented [HM9]: For consistency, capitalize or lowercase rewardpoints and credit card

Commented [HM10]: Verififes there are enough reward points for the entire purchase

Commented [HM11]: Could just use total

Commented [HM12]: NuGet

Commented [HM13]: Could say valid number instead

5.3.2 An Employee should be able to quickly move through the order process. Screen transitions should be quick, and the form layouts should be arranged such that the most used options are closer to the cursor's path.

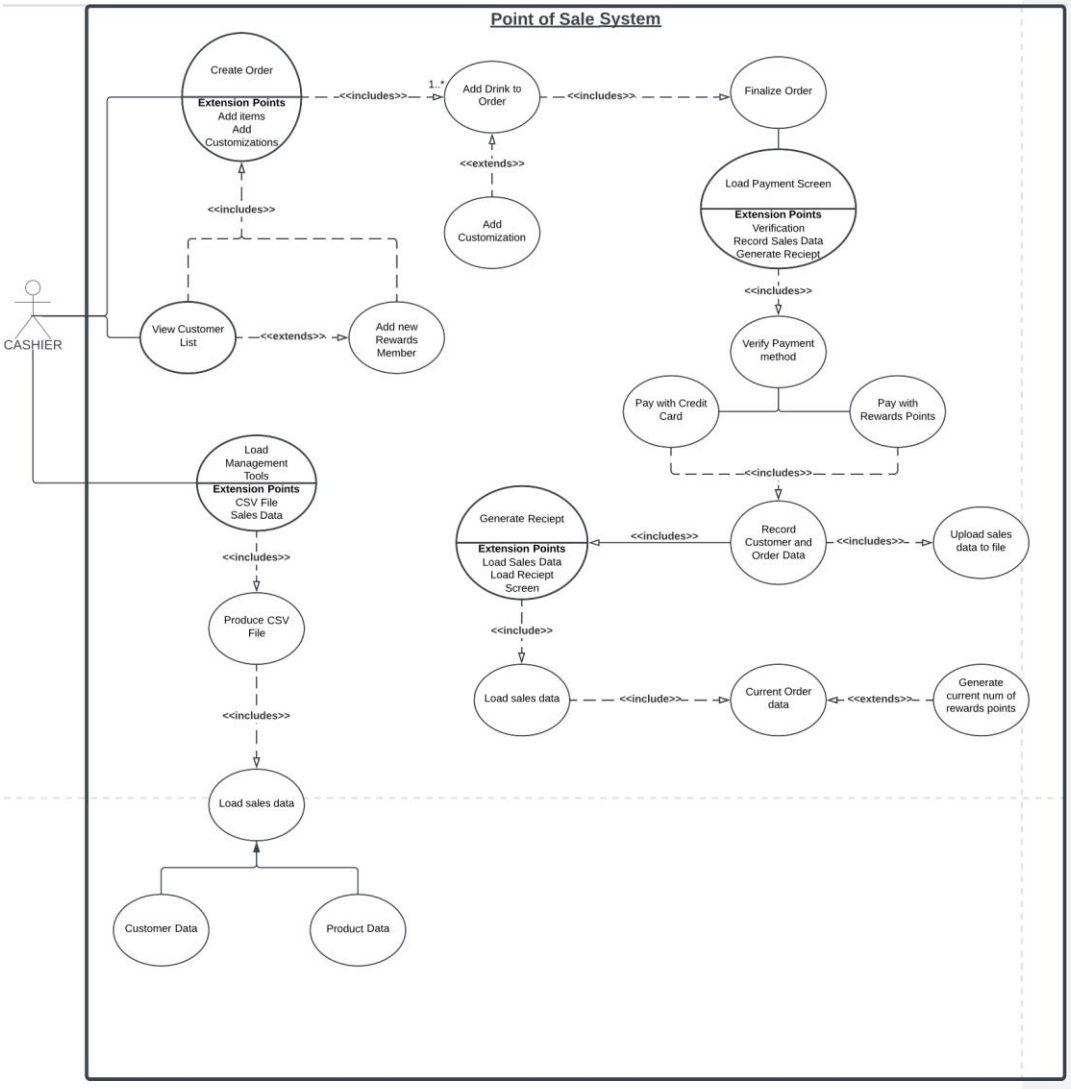
5.3.3 An Employee should have little room to make errors in the order process, and if they do, it should be easily rectifiable. Confirmations should be used for any non-reversable input and back buttons should be available on each form.

Priority Table

Must Have	1.1.1 1.2.1 1.2.2 1.2.3 1.2.4 1.2.5 1.3.1 1.3.2 1.3.3 1.3.4 1.3.5 1.3.6 1.4.1 1.4.1.1 1.4.1.2 1.4.2 1.4.2.1 1.4.2.2 1.4.3 1.4.4 1.4.5 1.5.1 1.5.2 1.5.3 1.5.4 1.6.1 1.6.1.1 1.6.2 1.7.1 1.7.3 1.8.1 1.8.2 1.8.3 1.8.4 1.8.4.1 1.8.4.2 1.8.5 1.8.6 2.1.1 2.1.2.1 2.1.2.2 2.1.2.3

	2.1.2.4 2.1.2.5 2.1.2.6 2.1.2.7 2.1.3.1 2.1.3.2 2.2.1 2.2.2 2.3.1 2.3.2 3.1.1 3.2.1 3.2.2 4.1 4.2 4.3 4.4.1 4.4.2 4.4.3
Need to Have	5.1.1 5.2.1
Nice to Have	1.4.1.3 1.7.2 1.7.4 5.3.1 5.3.2 5.3.3

Use Case Diagram



USE CASE FLOW OF EVENTS

1.0 Flow of events for Anonymous Customer Ordering a Drink

1.1 **Preconditions:** Customer does not have an existing account in the customer database.

1.2 **Main Flow:** The Employee will click the "Create Order" Button from the "Main" screen and continue to create their order.

2.0 Flow of events for Existing Customer Ordering a Drink

2.1 **Preconditions:** Customer has an existing account in the customer database.

2.2 **Main Flow:** The Employee will click on the "Customer List" button. The Employee will then find the Customer in the Customer List and select the "Begin Order" button next to their name. They will continue to create their order.

2.3 **Alternate Flows:**

2.3.1 The Customer decides to not order: The "Return to Main Menu" button is clicked.

3.0 Flow of events for Customer who wants to create a new account

3.1 **Preconditions:** Customer does not have an existing account but wants to create one.

3.2 **Main Flow:** The Employee will click on the "Customer List" button. Then the Employee will click on the create new customer button. The Employee populates the add Customer fields. When finished the Employee clicks "Add Customer"; then, the system will automatically send the user to the "Create Order" screen.

3.3 **Alternate Flows:**

3.3.1 The Customer decides they no longer want to create a new account but still wants to order: The Employee selects the "Begin Order" button for the *anonymous* account.

3.3.2 The Customer decides to not make an account or order: The "Return to Main Menu" button is clicked.

4.0 Flow of events for creating an Order

4.1 **Preconditions:** Customer status has been established and Employee is now on "Create Order" screen.

4.2 **Main Flows:** The Employee adds the drink that the Customer specifies and any customizations they want as well. When the Customer is satisfied with their order the Employee clicks the "Finalize Order" button. This moves them to the "Payment" screen.

4.3 **Sub-Flows:**

4.3.1 Base Drink: The Customer first chooses the base drink.

4.3.2 Customize Drink: The Customer can customize their drink
4.3.3 Add Drink to Order: When the order is completed it gets added to the order. This can be done for multiple drinks.

4.4 **Alternate Flows:**

4.4.1 The Customer decides to cancel the order: All input data so far is cleared, and the screen returns to the main menu.

5.0 Flow of events for Finalizing Order for an Anonymous Customer

5.1 **Preconditions:** Customer has finished creating their order and is *anonymous*

5.2 **Main Flows:** The Employee collects the Customer's Credit Card information and enters it into the system. After successful payment verification a Receipt is Generated. Sales data is collected and uploaded to the database.

5.3 **Alternate Flows:**

5.3.1 The Customer's Credit Card information is invalid: The Customer provides a different Credit Card, or the order is cancelled.

5.3.2 The Customer decides they don't want their order anymore: All input data so far is cleared, and the "Return to Main Menu" button is clicked.

6.0 Flow of events for Finalizing Order for a Customer with an Account

6.1 **Preconditions:** The Customer has a rewards account and has completed their order.

6.2 **Main Flows:** The screen will let the Customer know if they have enough Reward Points to pay for their order. If they do, the Customer can choose to pay with Reward Points or a Credit Card. If they do not have enough points to cover the whole order, the Customer must provide a Credit Card. For Credit Card payments the Employee will collect Credit Card information from the Customer. After successful payment verification a Receipt is Generated. For Credit Card payments the Customer is awarded **ten** rewards points for every dollar spent. Sales data with reward point data is collected and uploaded to the database.

6.3 **Alternate Flows:**

6.3.1 The Customer's Credit Card information is invalid: The Customer can pay with Reward Points if the Customer has enough, or provide a different Credit Card, or the order is cancelled.

6.3.2 The Customer decides they don't want their order anymore: All input data so far is cleared, and the "Return to Main Menu" button is clicked.

7.0 Flow of events for creating a sales report

7.1 **Preconditions:** None

7.2 **Main Flow:** Employee clicks generate sales report. A CSV file is created containing all sales data for the day and is then opened in Excel for the Employee to view.

7.3

7.4 **Alternate Flows:**

7.4.1 Employee decides to return to main menu: Click return button

Decision Table

Conditions	**Customer Registration**		
Existing customer	T	-	-
New customer wants membership	-	T	-
New customer does not want membership	-	-	T

Actions			
Order drink without customer information	-	-	X
Open customer list	X	X	
Add customer list	-	X	-
Order drink with customer information	X	X	-

This table specifies what actions to execute when beginning a customer's order. If a customer record already exists, the employee will open the customer list and begin the drink order. If the customer does not have a record and would like to sign up for membership, the employee will open the customer list, add the new customer record to the list, then begin the drink order. If the customer does not want a membership, the employee will begin using the "Order Drink" button or, alternatively, the *anonymous* customer record. The latter will be used if the customer decides not to create a membership account after moving to the customer list.

Commented [HM14]: The new customer

Commented [HM15]: The employee

Commented [HM16]: Clarify, might have to write new sentence or clarification

Commented [HM17]: Remove comma

Conditions	**Order Payment**		
Reward Points Value	Has Enough	Does Not Have Enough	-
Existing customer	T	F	-
Anonymous customer	-	-	T

Actions			
Credit Card	X	X	X
Reward Points	X	-	-

This table determines what payment options are available for a customer. If they are an existing customer with a reward point balance greater than or equal to the order total, they can pay with credit card or reward points. If they are an existing customer with a reward point balance less than the order total, they can pay with a credit card. If they are an anonymous customer, they can pay with a Credit Card.