**Sale Module**

**This is a group lab, and all group members *must* participate.**

Sarah and Ted’s online business is expanding, and since they’re looking at loyalty programs, recording customer information is important to them. Recording sales from their website is easy, but recording sales at their kiosks is problematic. Some customers simply do not want to give their personal information, so Sarah and Ted are comfortable with recording these sales to one ‘customer’ by location.

At the end of the day, Sarah and Ted would like the ability to query purchases – minus sales by location so that they can adjust inventory to account for unrecorded sales.

Your task is to create a class diagram and sequence diagrams to support the following user stories and systems use case specifications.

Copy your diagrams to a word file. Analyze your model and outline what the .h files would look like if you generated code from your model. Include pseudo code to indicate what each method would do. Attach this word file and your visual paradigm file to your submission.

Use Case: Maintain Customer

User Story

As the owner of this business, I would like to record contact information for my customers.

Acceptance Criteria:

1. Must be able to flag customers as no longer active.
2. Must be able to query customers.
3. Must be able to query sales by company.

Use Case Descriptions

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case Name | **Create Customer** | | |
| Triggering Event | New customer requests sale. | | |
| Brief Description | Allows the Owner to record a new customer’s information. | | |
| Actors | Owner | | |
| Related Use Cases |  | | |
| Preconditions | Owner has opened the Main Menu. | | |
| Post Conditions | Customer is saved to the database. Sales can be created for the Customer | | |
| Flow of activities | Actor | | System |
|  |  | Requests to add a new Customer | Prompts for Customer name, address, telephone and email address |
|  |  | Enters customer name, address, telephone number and email address | Verifies that name, address, telephone number and email address have been entered.  Creates a unique customer identifier.  Displays the customer’s identifier, name, address, telephone number and email address.  Prompts to save. |
|  | 3. | Request to save | Saves the customer. |
| Exception Conditions | * Owner chooses to cancel adding the customer | | |

User Story

As the owner of this business, I would like to record my sales so that I can quickly see how much money I’ve made.

Acceptance Criteria:

1. Must be able to record sales by selecting products by product type and enter quantity sold.

Use Case Descriptions

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case Name | **Create Sale** | | |
| Triggering Event | Customer has to purchase products | | |
| Brief Description | Allows the Owner to record a new sale | | |
| Actors | Owner | | |
| Related Use Cases |  | | |
| Preconditions | Owner has opened the Main Menu. | | |
| Post Conditions | Sale is created and can now be queried | | |
| Flow of activities | Actor | | System |
|  | 1. | Requests to add a new sale | Displays a list of customers and prompts for selection. Displays a list of locations and prompts for selection. Prompts for sales date |
|  | 2. | Selects a location.  Selects a customer.  Enters sales date | Verifies that customer and location were selected.  Verifies that sale date has been entered and generates a unique sale number.  Displays Sale identifier, date and customer. Display a list of product types and prompts for selection. |
|  | 3. | Selects a product type | Displays a list of products including product name and price. Prompts for selection and quantity sold |
|  | 3. | Selects a product and enters quantity sold | Verifies that quantity sold was entered.  Adds product to the sale and calculates extended price \* quantity.  Calculates HST.  Updates totals.  Display a list of products sold. Prompts for another product from the list and quantity sold.  Prompts for another product type. |
|  | 4. | Repeats above step until all products entered | Prompts to select a product type. Prompts to confirm sale. |
|  | 5. | Repeats above 2 steps until all product types are selected. | Prompts to confirm sale |
|  | 5. | Chooses to save | Saves the sale.  Return to the main menu. |
| Exception Conditions | * Owner chooses to cancel adding the sale. | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case Name | **Query Purchase - Sale** | | |
| Triggering Event | Owner requires a list of remaining inventory (end of day processing) | | |
| Brief Description | Allows the Owner to retrieve purchases – sales so that inventory can be reconciled | | |
| petActors | Owner | | |
| Related Use Cases |  | | |
| Preconditions | Owner has opened the Main Menu. | | |
| Post Conditions | Purchases – sales are displayed. | | |
| Flow of activities | Actor | | System |
|  | 1. | Retrieve locations | Displays a list of locations and prompts for selection |
|  | 2. | Requests to purchases – sales for location | Retrieves Purchases by product assigned to the location. Retrieves Sales by product assigned to the location. Retrieves Inventory adjustments by product assigned to the location. Subtracts purchases – sales and inventory adjustments) and displays quantity in inventory. |
|  |  | Request to exit | returns to the main menu |
| Exception Conditions | Owner requests to cancel query | | |

User Story

As the owner of this business, I would like to record inventory adjustments to accommodate for the cash sales that happen quickly and aren’t recorded.

Acceptance Criteria:

1. Must be able to record inventory adjustments by location and product.

Use Case Descriptions

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case Name | **Create Inventory Adjustment** | | |
| Triggering Event | End of day processing | | |
| Brief Description | Allows the Owner to adjust inventory to accommodate for unrecorded sales. | | |
| Actors | Owner | | |
| Related Use Cases |  | | |
| Preconditions | Owner has opened the Main Menu. | | |
| Post Conditions | Inventory adjustment is recorded and available to query | | |
| Flow of activities | Actor | | System |
|  | 1. | Requests to add an inventory adjustment | Displays a list of product types and prompts for selection. Displays a list of locations and prompts for selection. |
|  | 2. | Selects a location.  Selects a product type. | Verifies that product type and location were selected.  Displays a list of products and prompts for selection. Prompts for adjustment quantity. |
|  | 3. | Selects a product and enters adjusted quantity. | Creates an adjustment identifier and assigns the system date to adjustment date. Displays the inventory adjustment |
|  | 4. | Repeat the above step until all adjustments for the product type are entered. Chooses to confirm. | Saves the inventory adjustments.  Returns to the main menu. |
| Exception Conditions | * Owner chooses to cancel adding the inventory adjustments. | | |

**Your tasks:**

1. Create a class diagram to support the above case study and Systems Use Case Specifications
2. Create sequence diagrams for the above Systems Use Case Specifications.
3. Analyze your model and outline what the “.h” files would look like if you generated code from your model.