Scheduling Module

Victoria finds herself manually scheduling orders and which of her bakers works on each. Victoria would like the system to help her add orders to her calendar. The order must include the customer information, baker, any baked goods ordered and price. Ideally, Victoria would like to create an invoice from order information.

Task

Create the use case and sequence diagrams as specified at the end of this document. Place the screenshots in a document with proper formatting and descriptions. Then, create an .h file(s) which contain the classes and function declarations necessary, with comments describing the functionalities. Add the screenshots of these .h files to the document. **Your submission should include the .h file(s), the .vpp file and the PDF document with your screenshots.**

User Story

As the owner of this business, I would like to record my scheduled orders and be able to see my scheduled orders. I would like to be able to create a sale from an order, so I'll need the same information on the order, as a sale.

Acceptance Criteria:

1. Must be able to record orders for containing multiple baked goods for a customer.
2. Must be able to turn an order into a sale
3. Must be able to query calendar by day and see the orders due that day.

Use Case Descriptions

Use Case Name	Create Customer Order	
Triggering Event	Customer has placed an order.	
Brief Description	Allows the Owner to record a new order and sales invoice.	
Actors	Owner	
Related Use Cases		
Preconditions	Owner has opened the Main Menu.	
Post Conditions	Order is created and can now be queried	
Flow of activities	Actor	System
	1. Requests to add an order	Displays a list of customers and prompts for selection
	2. Selects a customer	Prompts for order due date

	3.	Enters order date	Displays a list of already scheduled orders for that day.
	4.	Selects a pickup time	Generates and displays an order ID. Displays a list of baked goods and prompts for selection.
	Loop		
	5.	Selects a baked good.	Displays a list of special instructions and quantity.
	6.	Selects instructions that apply and quantity.	Adds baked good and quantity to the invoice. Calculates the detail amount (hours * price). Displays the order details, prompting for any changes to the hours or pricing. Prompts for another baked good. Prompts to save.
	End	Repeats above 2 steps until all desired baked goods have been selected.	Displays the order service and prompts to save.
	7.	Chooses to save.	Saves the invoice. Return to the main menu.
Exception Conditions	• Owner chooses to abort adding the order.		

Diagram Requirements

1. Create a class diagram to support the above case study and Systems Use Case Specifications.
2. Create an object level sequence diagram, detailing the *Create Order* use case specification.