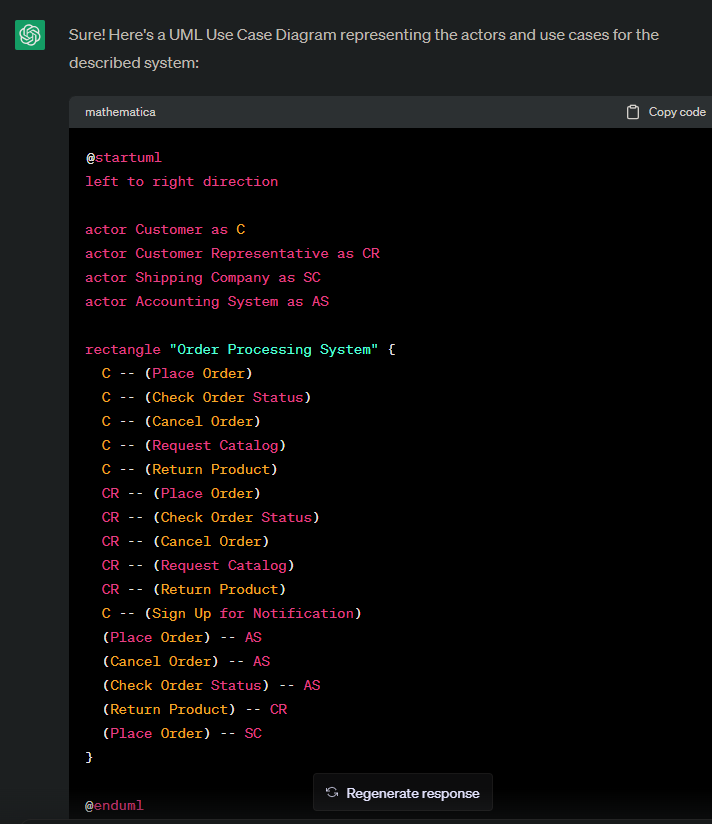
（1）

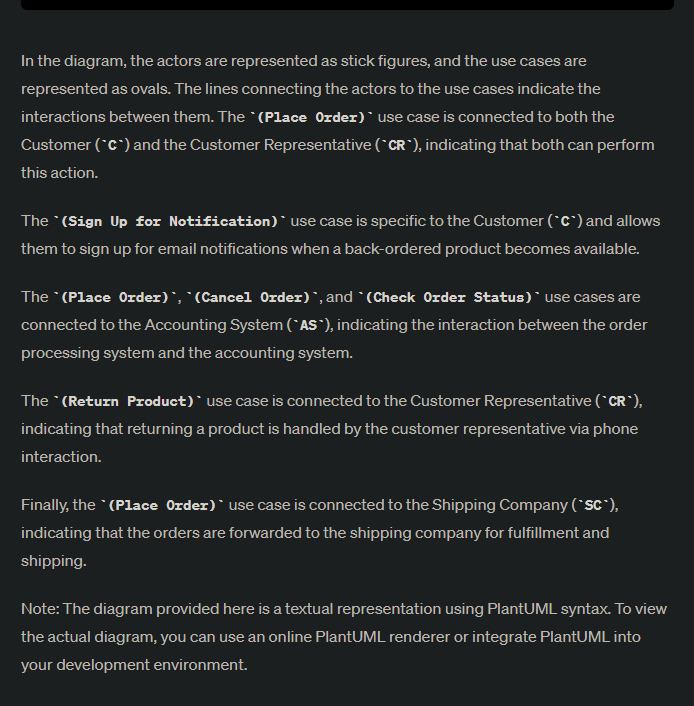
First question:

It takes five minutes to ask a total of one question

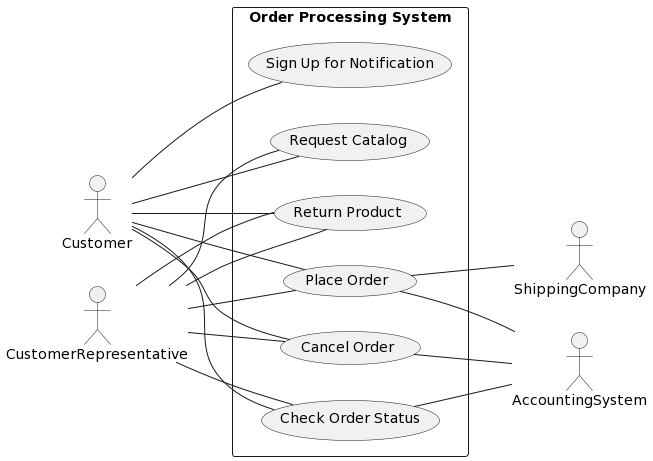
Consider the following problem description: A mail-order company wants to automate its order processing. The initial version of the order processing system should be accessible to customers via the web. Customers can also call the company by phone and interact with the system via a customer representative. It is highly likely that the company will enhance this system in upcoming years with new features. The system allows customers to place orders, check the status of their orders, cancel an existing order and request a catalog. Customers may also return a product but this is only possible through the phone, not available on the web. When placing an order, the customer identifies himself by means of customer number (only for existing registered customers) or by means of his name and address. He then selects a number of products by giving the product number or by selecting products from the online catalogue. For each product, information such as price, a description and a picture (only on demand as they are usually high-resolution images of large size) are presented to the customer. Also, the availability of the product is obtained from the inventory. The customer indicates whether he wants to buy the product and in what quantity. When all desired products have been selected, the customer provides a shipping address and a credit card number and a billing address (if different from the shipping address). Then an overview of the ordered products and the total cost are presented. If the customer approves, the order is submitted. Credit card number, billing address and a specification of the cost of the order are used on the invoice, which is forwarded to the accounting system (an existing software module). Orders are forwarded to the shipping company, where they are filled and shipped. Customers who spent over a certain amount within the past year are promoted to be gold customers. Gold customers have additional rights such as being able to return products in an extended time period as well as earning more bonus points with each purchase. In addition, in cases where a product is on back order, gold customers have the option to sign up for an email notification for when the particular product becomes available.

Identify actors and use cases for the system described above and show them on a UML Use Case Diagram using uml codes.

Original Solution:



UML Use Case Diagram using plantuml:



1. Second question:

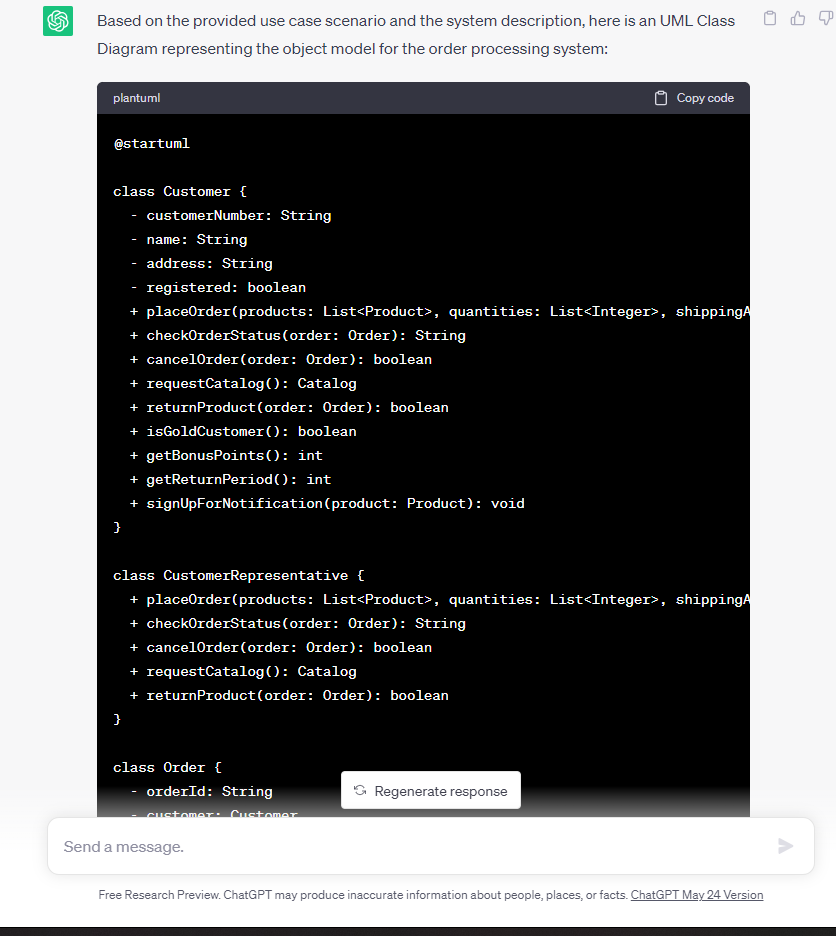
It takes fifteen minutes to ask a total of two questions

Perform a quick application domain analysis to come up with an object model for the above system. Express your findings with a UML Class Diagram using uml codes, making sure to identify any critical operations of classes.

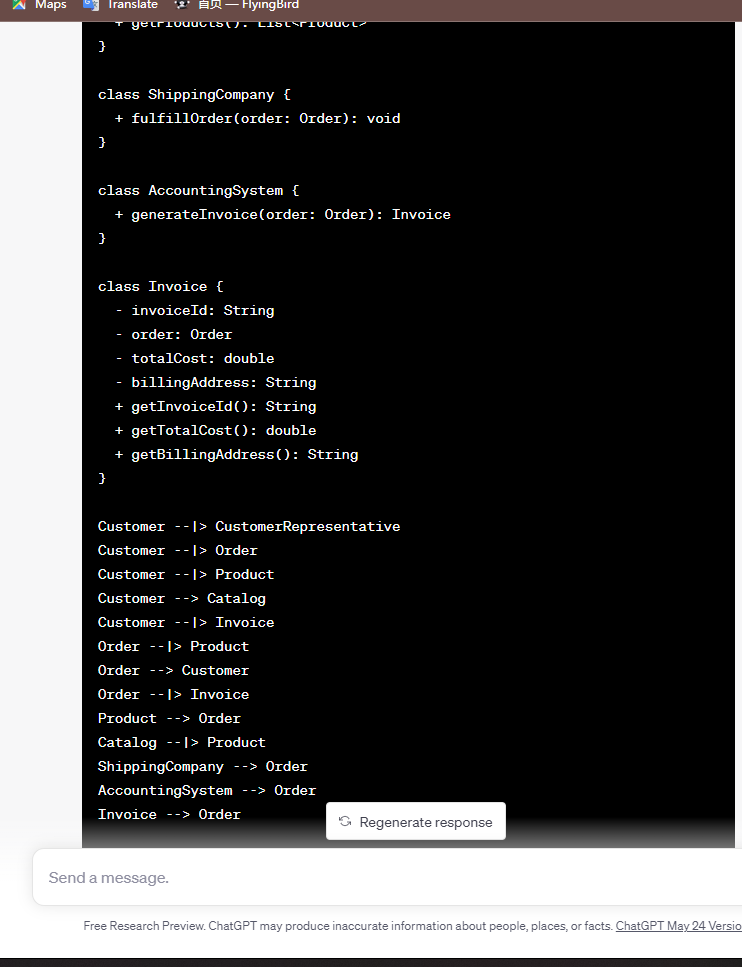
Consider the following use case scenario (for use case “place order”):

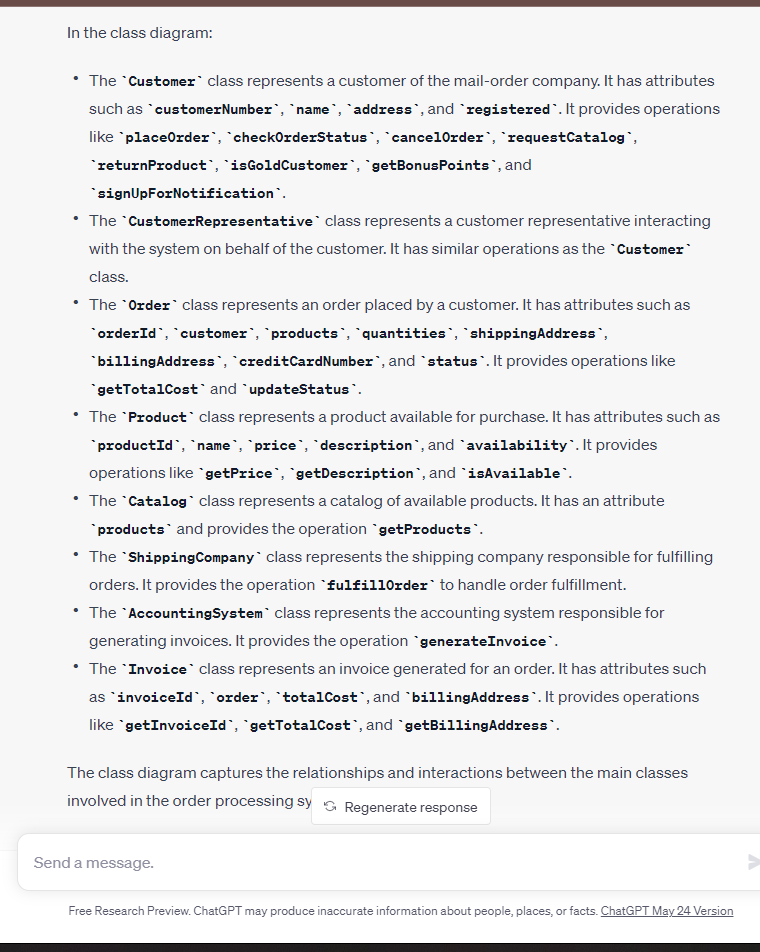
Ali is an existing customer of the order processing company described earlier, registered with their web site. Also assume that having browsed the printed catalogue he has, he already identified the two items (including their prices) he likes to buy from the company’s website using their product numbers (i.e. #2 and #9). First, he tries to buy one of product #2, but it is listed as unavailable in the inventory. Then, he adds two quantities of product #9, which turns out to be available, to his basket. He is then asked to confirm his registered shipping and billing addresses and credit card information from the customer database. He completes the order by clicking the Submit button. You may ignore processing of customer authentication.

Original Solution:

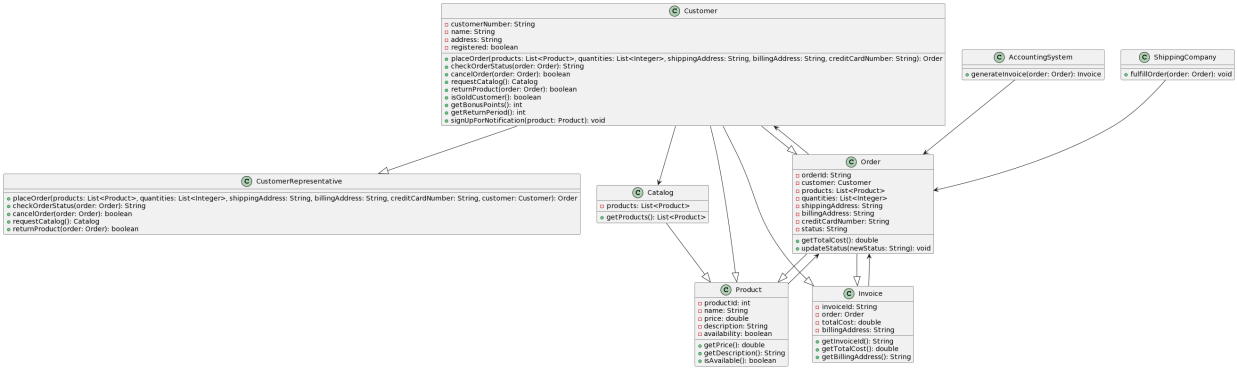








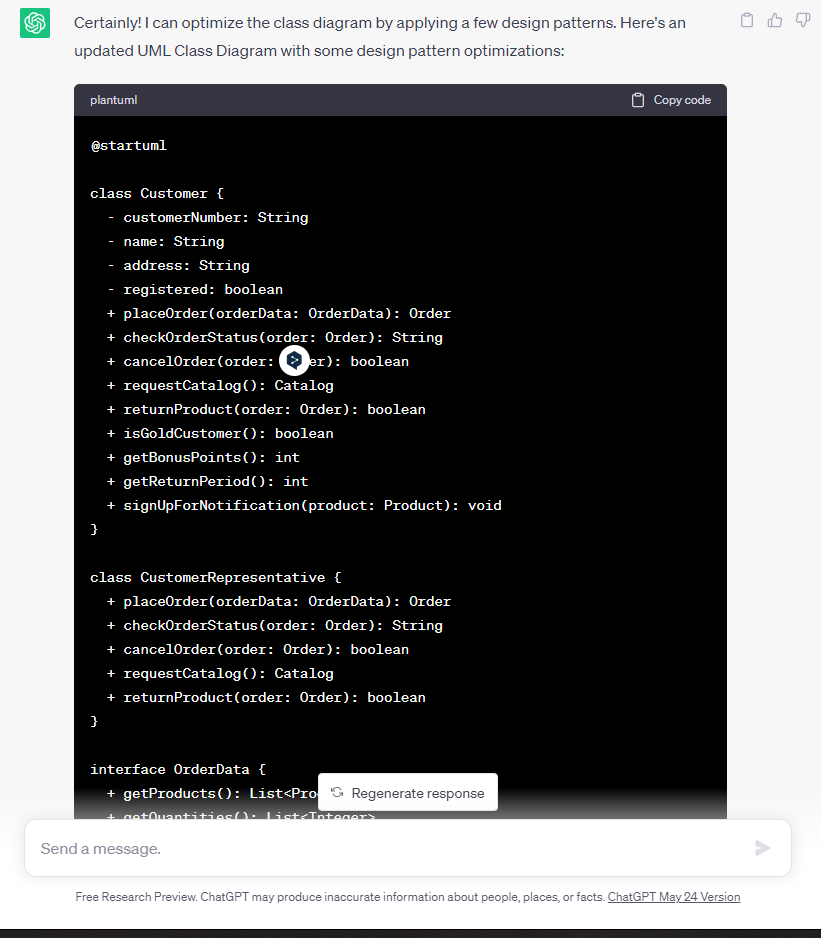
UML Use Case Diagram using plantuml:

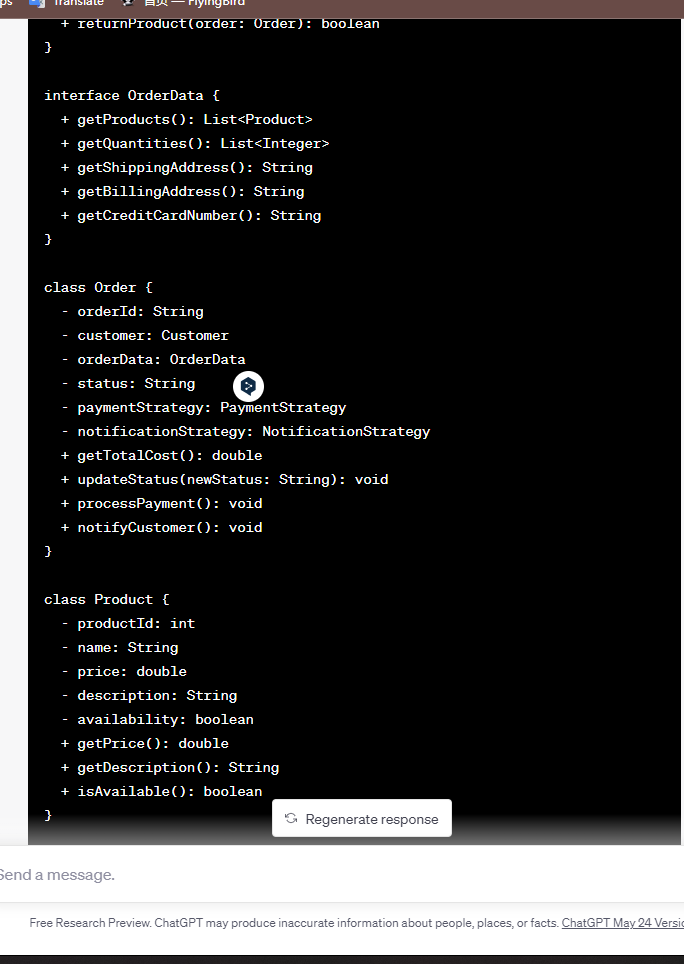


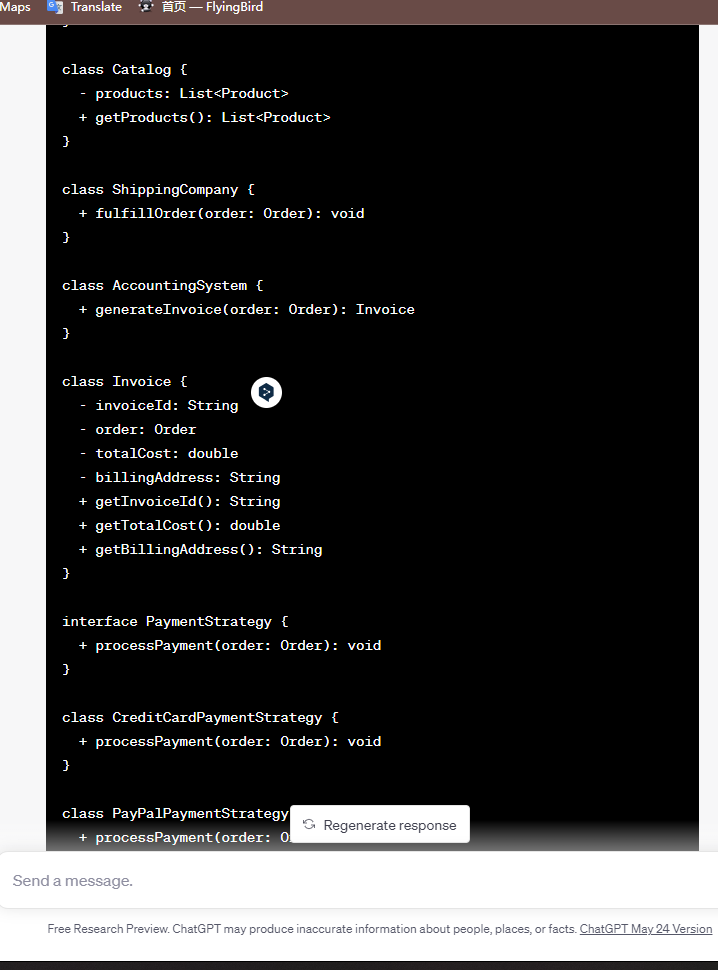
Final solution:

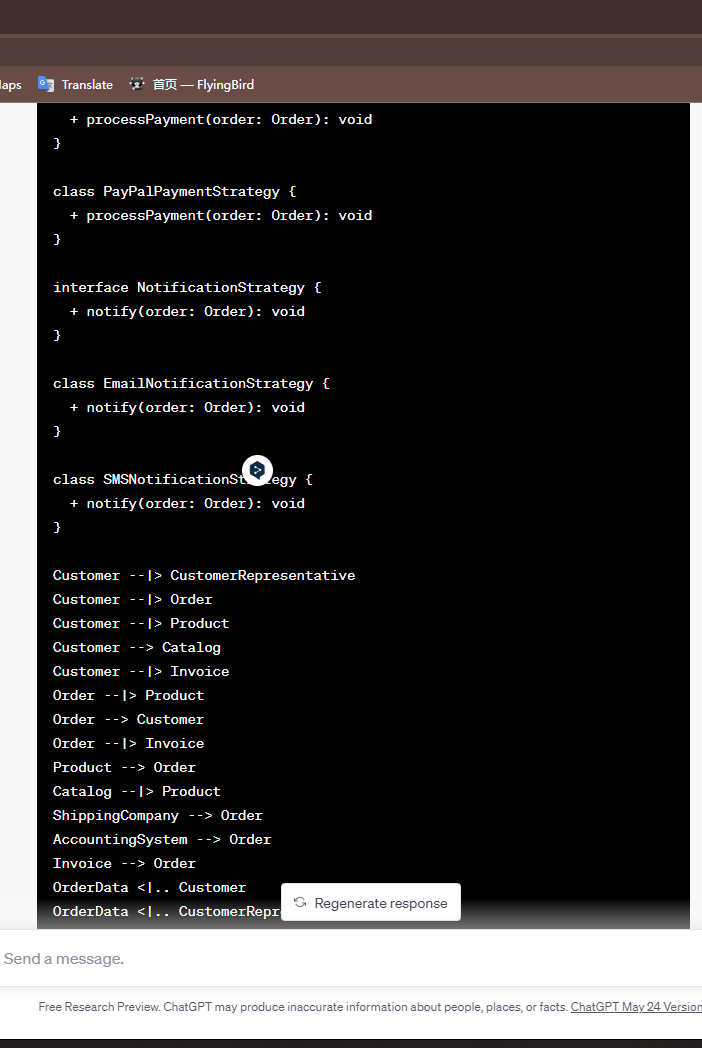
Question:Can you optimize the above class diagram using some design patterns, giving the corresponding uml code

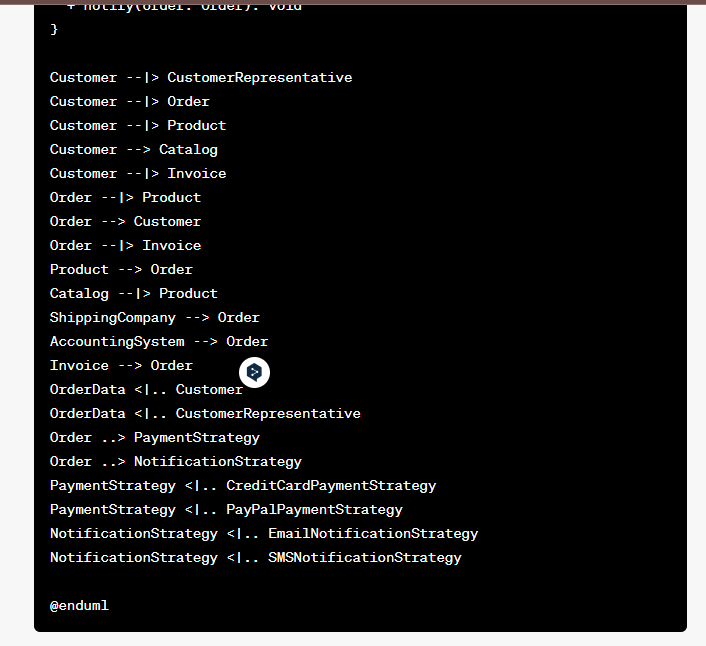
Solution:

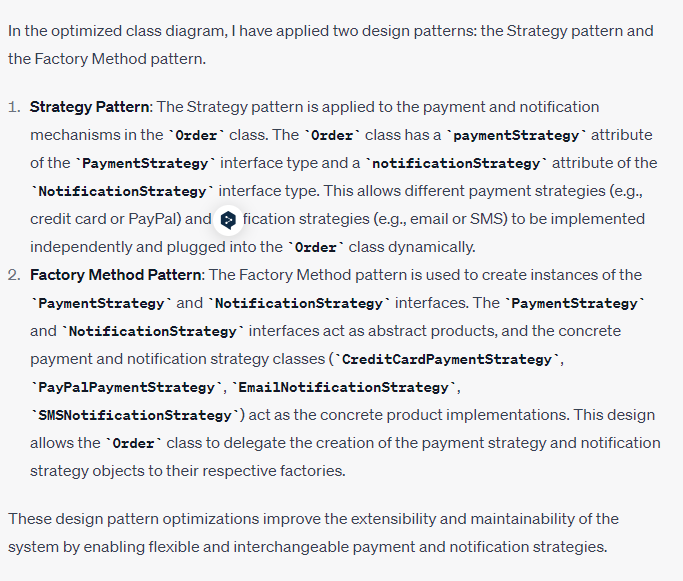




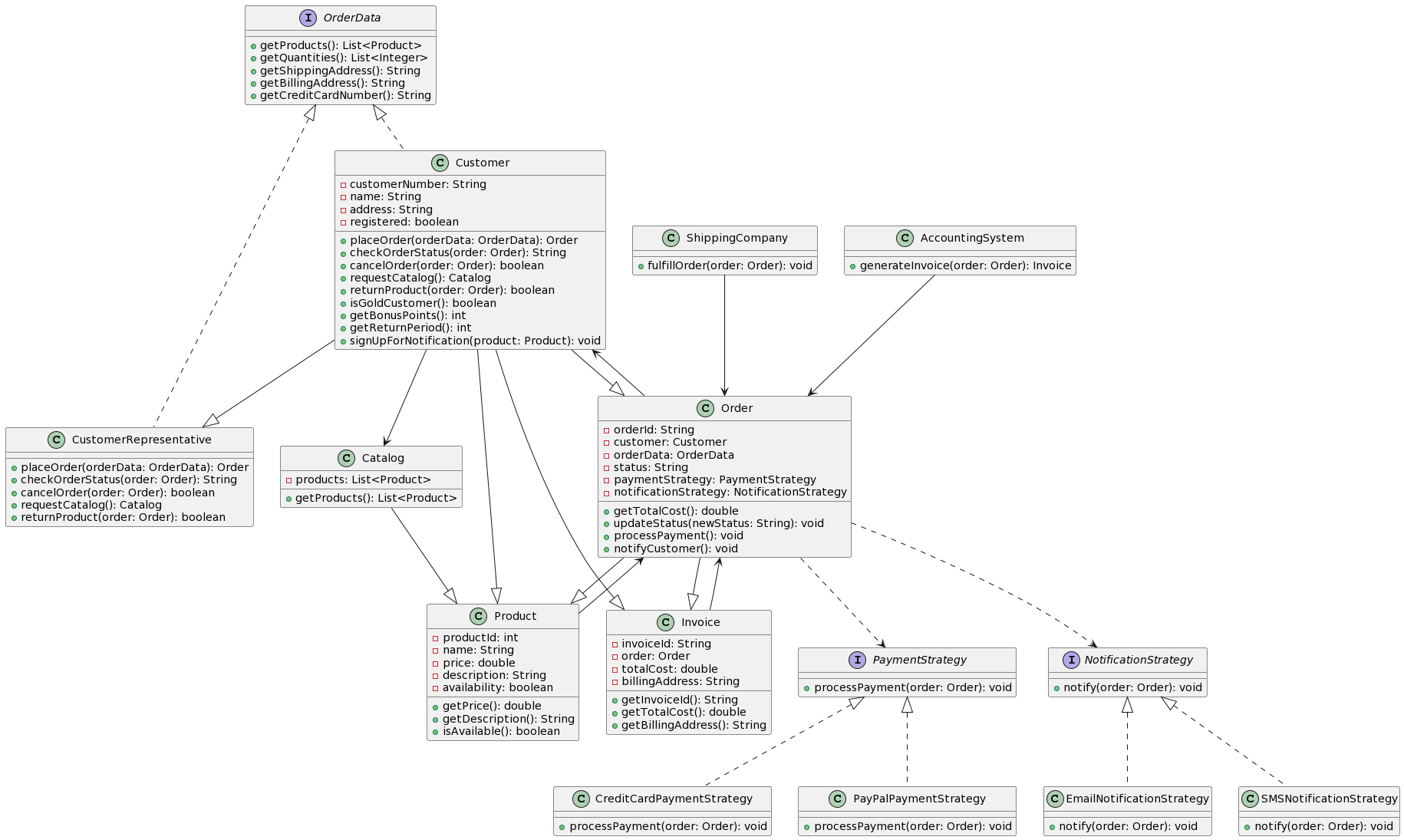








UML class Diagram using plantuml:



The difference between the initial solution suggested by AI and the optimized final solution:

**Abstraction and Flexibility:** The optimized solution with design patterns introduces more abstraction and flexibility in the system. By using the Strategy pattern for payment and notification strategies, the system can easily support different payment methods (such as credit card or PayPal) and notification methods (such as email or SMS) without modifying the core classes. This allows for easier extensibility and maintenance of the system as new payment or notification options can be added by implementing the corresponding strategies.

**Separation of Concerns:** The optimized solution separates the payment and notification logic from the Order class by using separate strategy objects. This adheres to the Single Responsibility Principle, making the Order class responsible for managing the order-related data and behavior, while the payment and notification strategies handle their respective responsibilities. This separation of concerns improves code clarity and maintainability.

**Code Reusability:** The optimized solution promotes code reusability by encapsulating payment and notification behavior into separate strategy classes. These strategy classes can be used by multiple classes within the system, reducing code duplication and improving overall code organization.

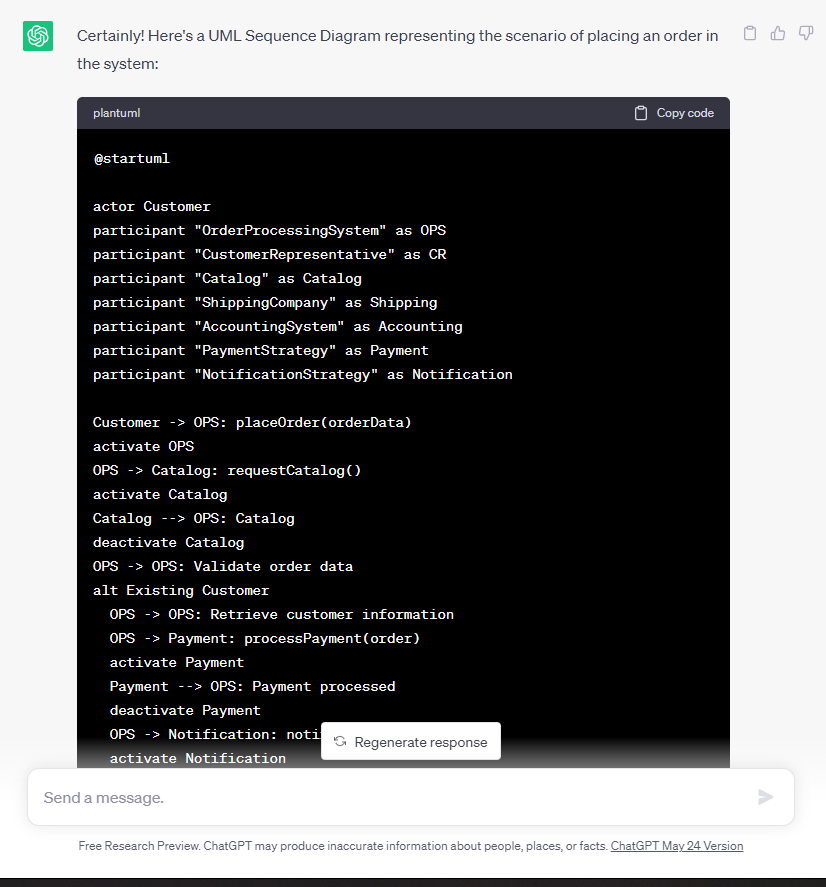
**Dynamic Behavior:** The optimized solution allows for dynamic behavior of payment and notification strategies. The Order class can switch between different strategies at runtime, providing flexibility to adapt to changing requirements or user preferences. This dynamic behavior is achieved through the use of interfaces and the ability to plug in different concrete strategy implementations.

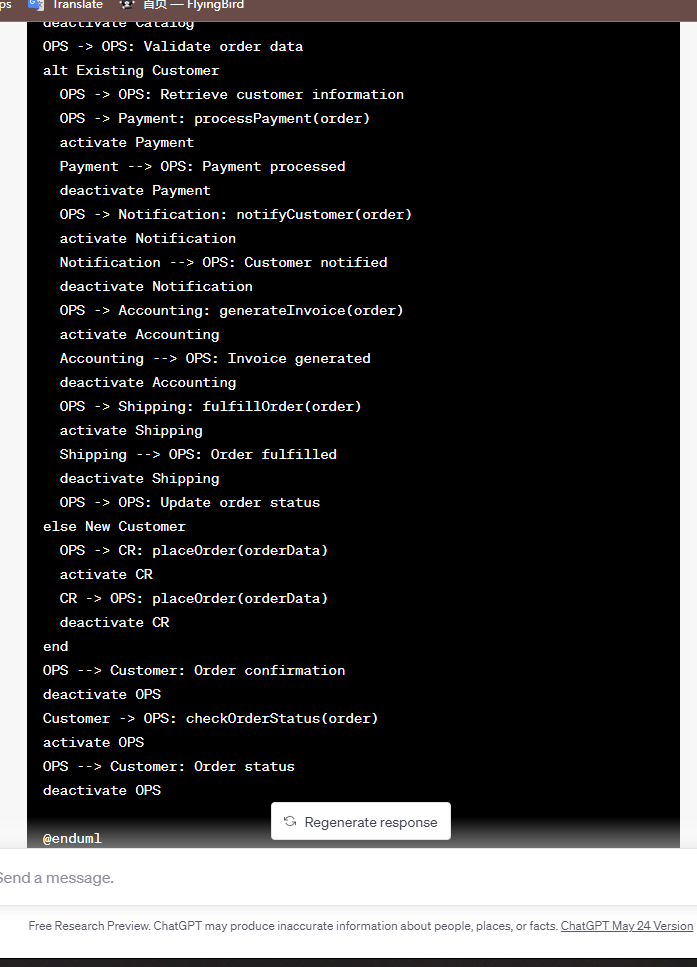
(3)third question:

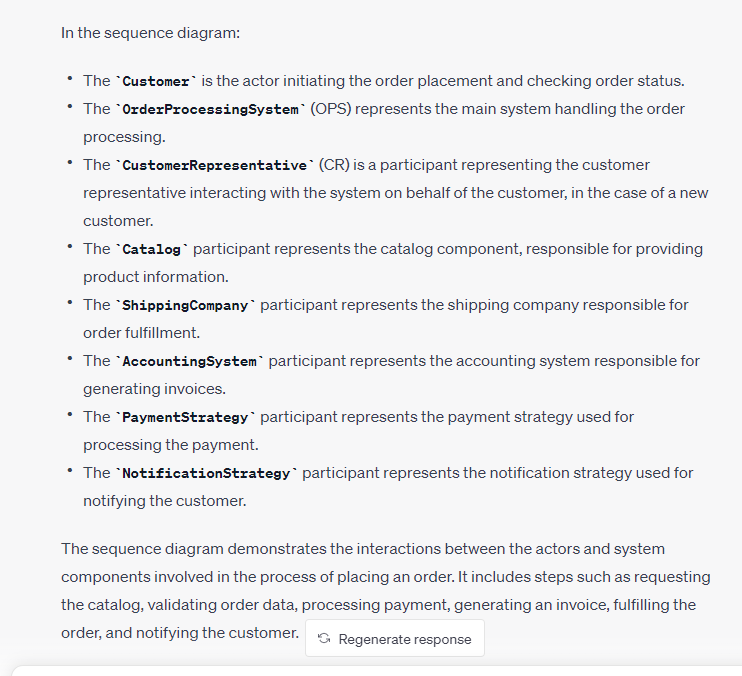
It takes ten minutes to ask a total of one question

Question:draw a UML Sequence Diagram for this particular scenario using uml codes.

Original Solution:







UML Sequence Diagram using plantuml:

