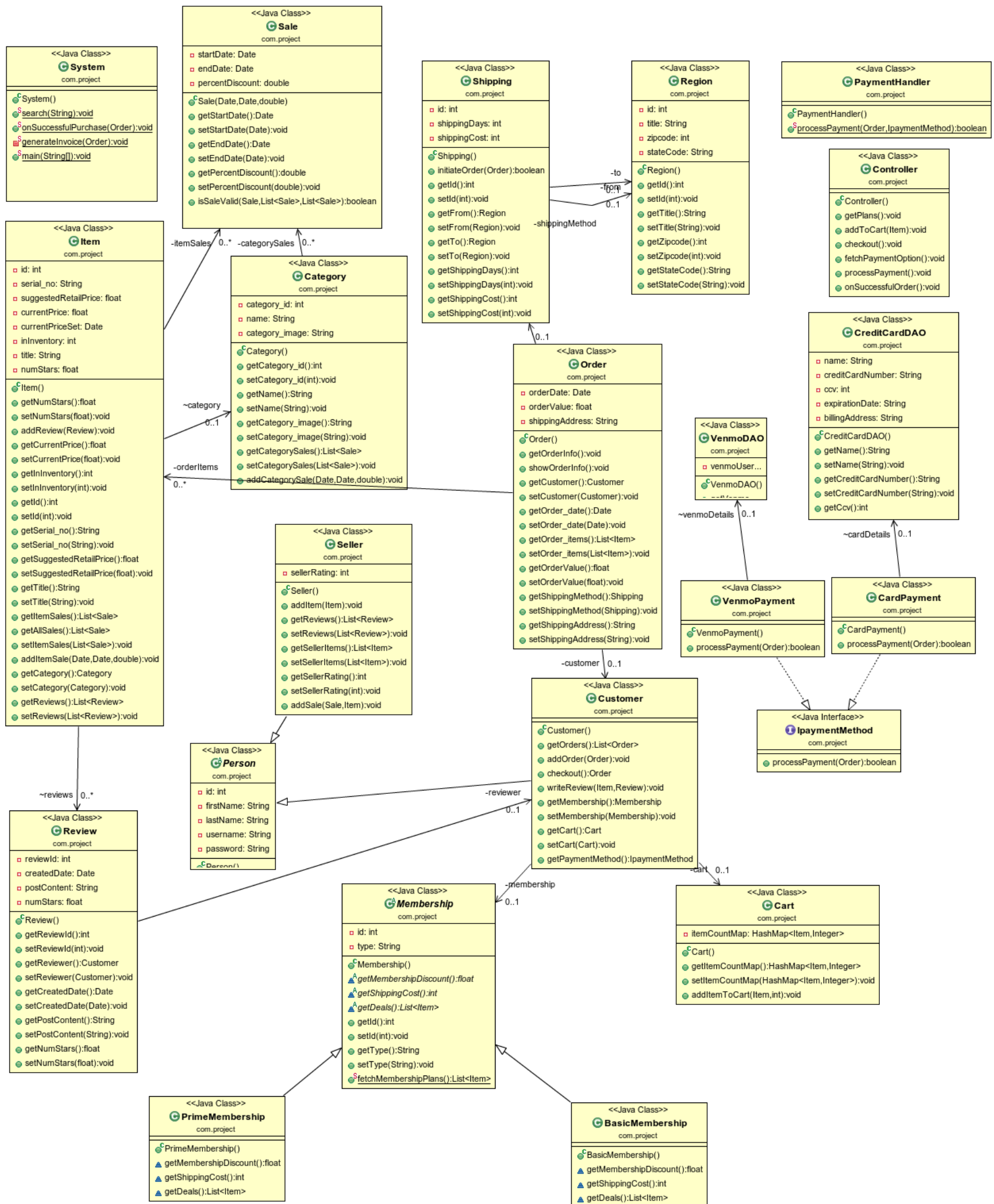


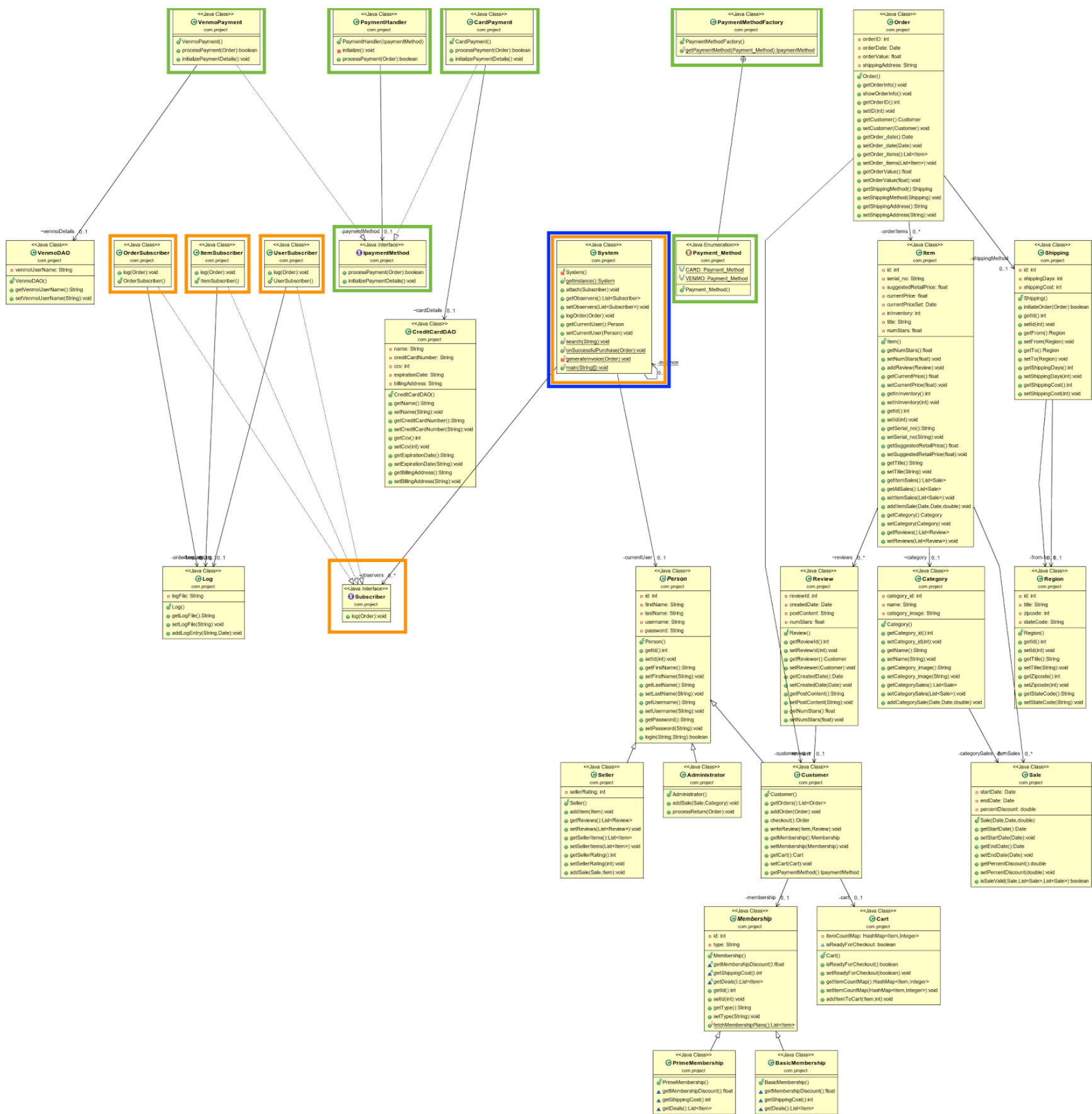
Team:
Rohit Gupta
Corin Sandford
Mike Gore

Title:
Marketplace

Part 2 Class Diagram:



Part 3 New Class Diagram:



New Class Diagram Legend:

- Orange boxes: Denote classes in **Observer** design pattern (used for logging).
- Green boxes: Denote classes in **Strategy** design pattern (used for payment).
- Blue box: Denotes class (System) in **Singleton** design pattern.

Description of Changes:

- Changed payment handling to employ **Strategy** design pattern.
Added **PaymentMethodFactory** class. `initializePaymentDetails()` method added to **IPaymentMethod** interface and its implementing classes, **VenmoPayment** and **CardPayment**. This allows us to use the Strategy pattern in **PaymentHandler**, which has a new `initialize()` method. When **PaymentHandler** calls this `initialize` method, it decides which type of **IPaymentMethod** to create based on the information provided by the user about their payment methods.
- Added logging, which employs **Observer** design pattern, with **System** class as publisher and logging classes as subscribers.
When an **Order** is successfully completed, the **System** publishes the order to observers, currently classes called **OrderSubscriber**, **ItemSubscriber**, and **UserSubscriber**, which implement an abstract Observer class called **Subscriber**. Once **System** publishes, these observers are used to notified to update their own state. In the current implementation, each performs a different type of order logging.
- Changed **System** class to employ **Singleton** design pattern.
System class now includes a `currentUser` (type **Person**).
- Controller** class on original diagram was subsumed into **System** on new diagram, and some extraneous methods were removed.
- Administrator** class was added to diagram.
- Added needed methods to other classes.