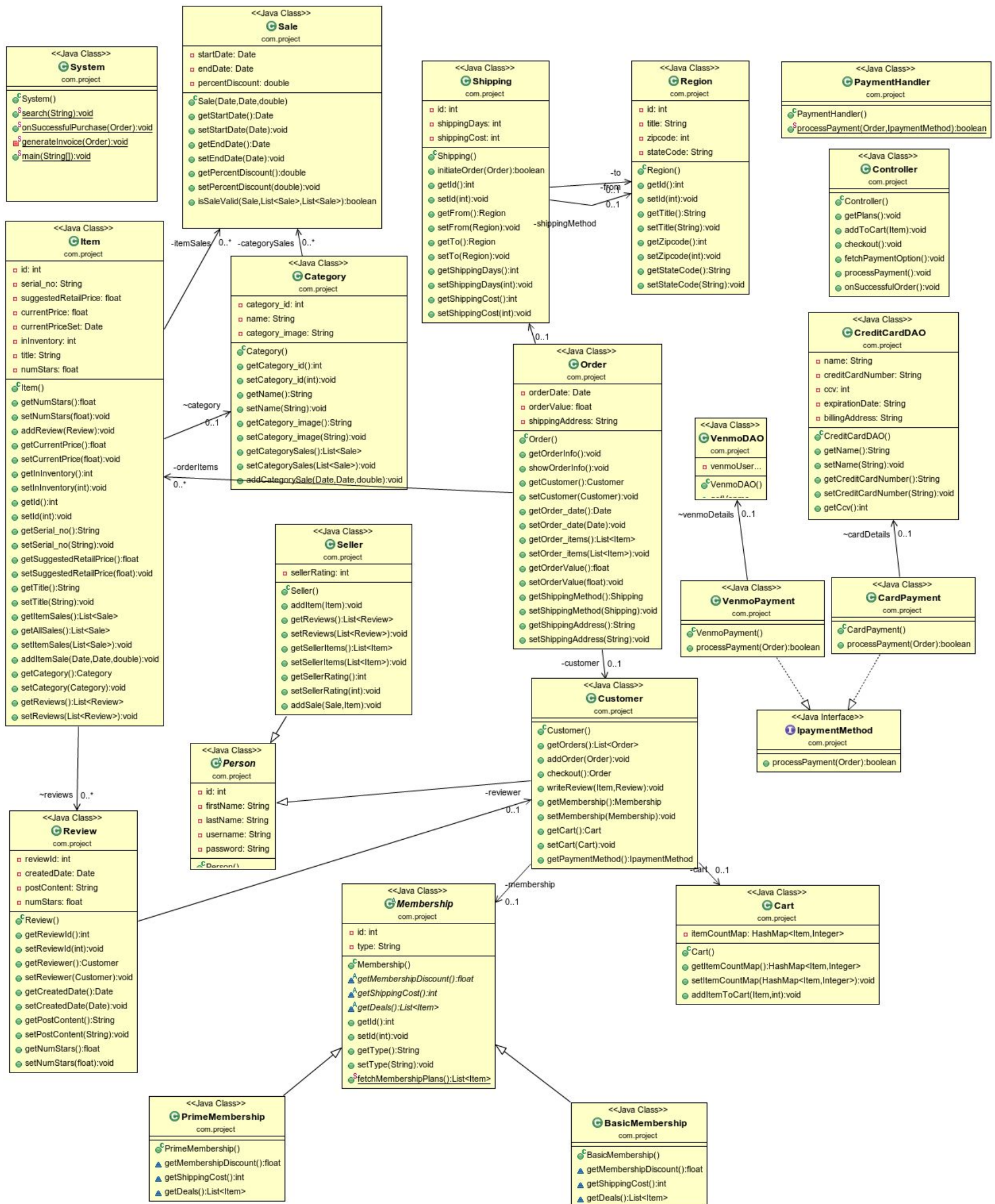
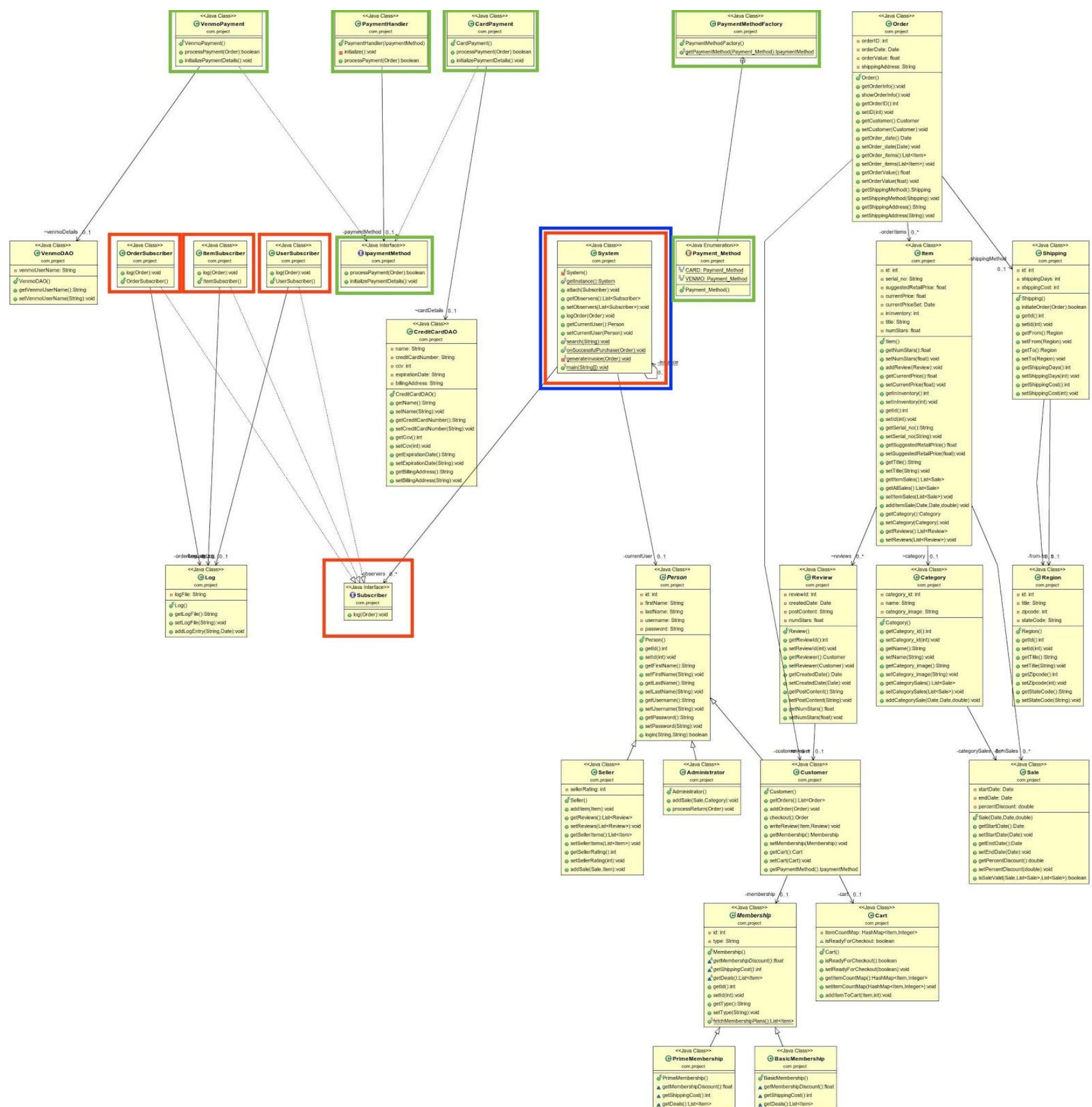


Mike Gore

## Marketplace



### 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:**

1. Changed payment handling to employ **Strategy** design pattern.  
Added **PaymentMethodFactory** class. `initializePaymentDetails()` method added to **PaymentMethod** interface and its implementing classes, **VenmoPayment** and **CardPayment**. **PaymentHandler** receives new `initialize()` method, and Strategy design pattern is implemented in **PaymentHandler**, which can now call all methods on a generic `paymentMethod` member variable without any if statements.
2. 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** pushes this to classes that log for Orders, Items, and Users. These classes (**OrderSubscriber**, **ItemSubscriber**, and **UserSubscriber**) implement from an abstract **Subscriber** class. (These classes also inherit from the **Log** class, and each logs certain aspects of an Order in its own way.)
3. Changed **System** class to employ **Singleton** design pattern.  
**System** class now includes a `currentUser` (type **Person**).
4. **Controller** class on original diagram was subsumed into **System** on new diagram, and some extraneous methods were removed.
5. **Administrator** class was added to diagram.
6. Added needed methods to other classes.