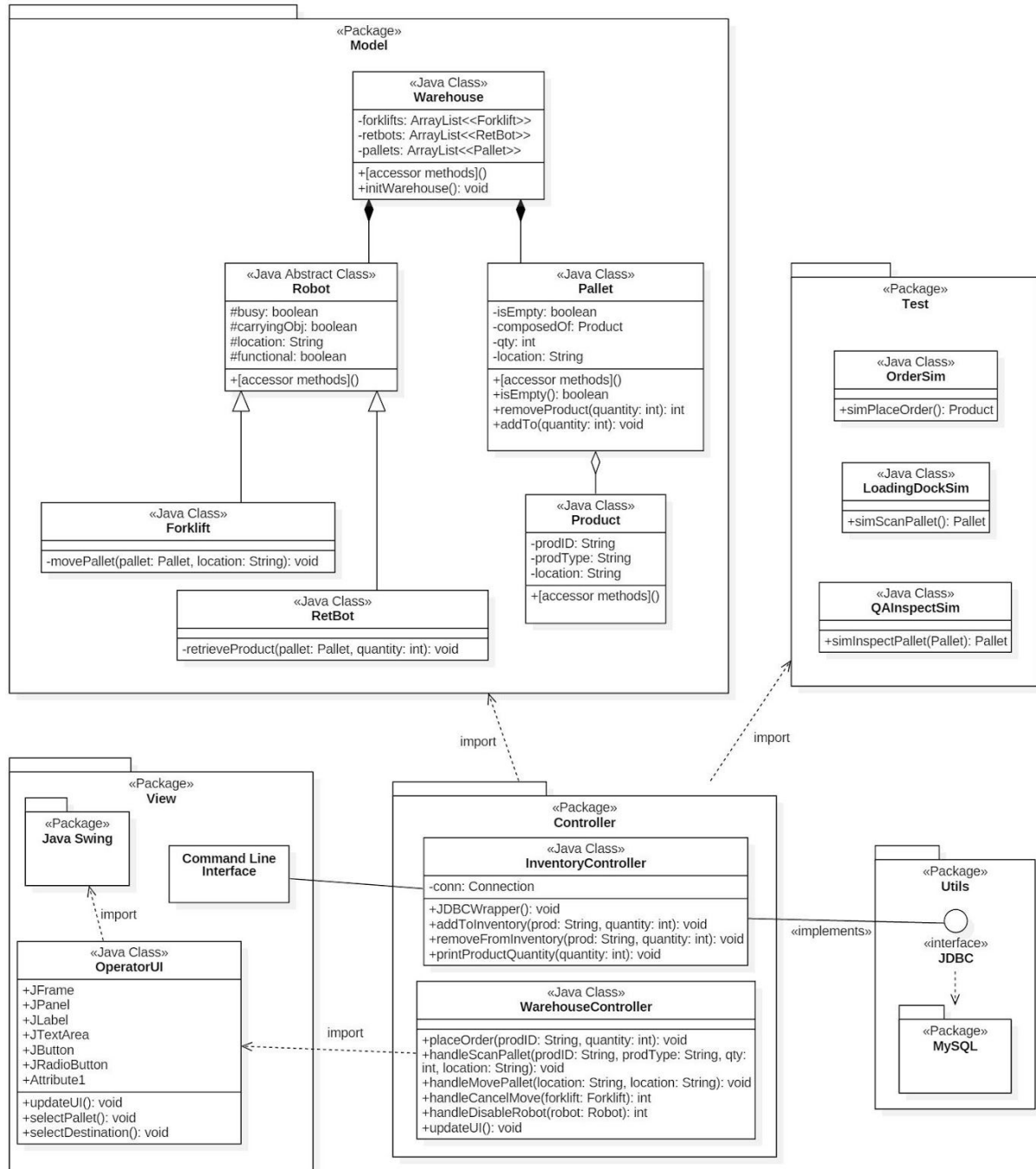


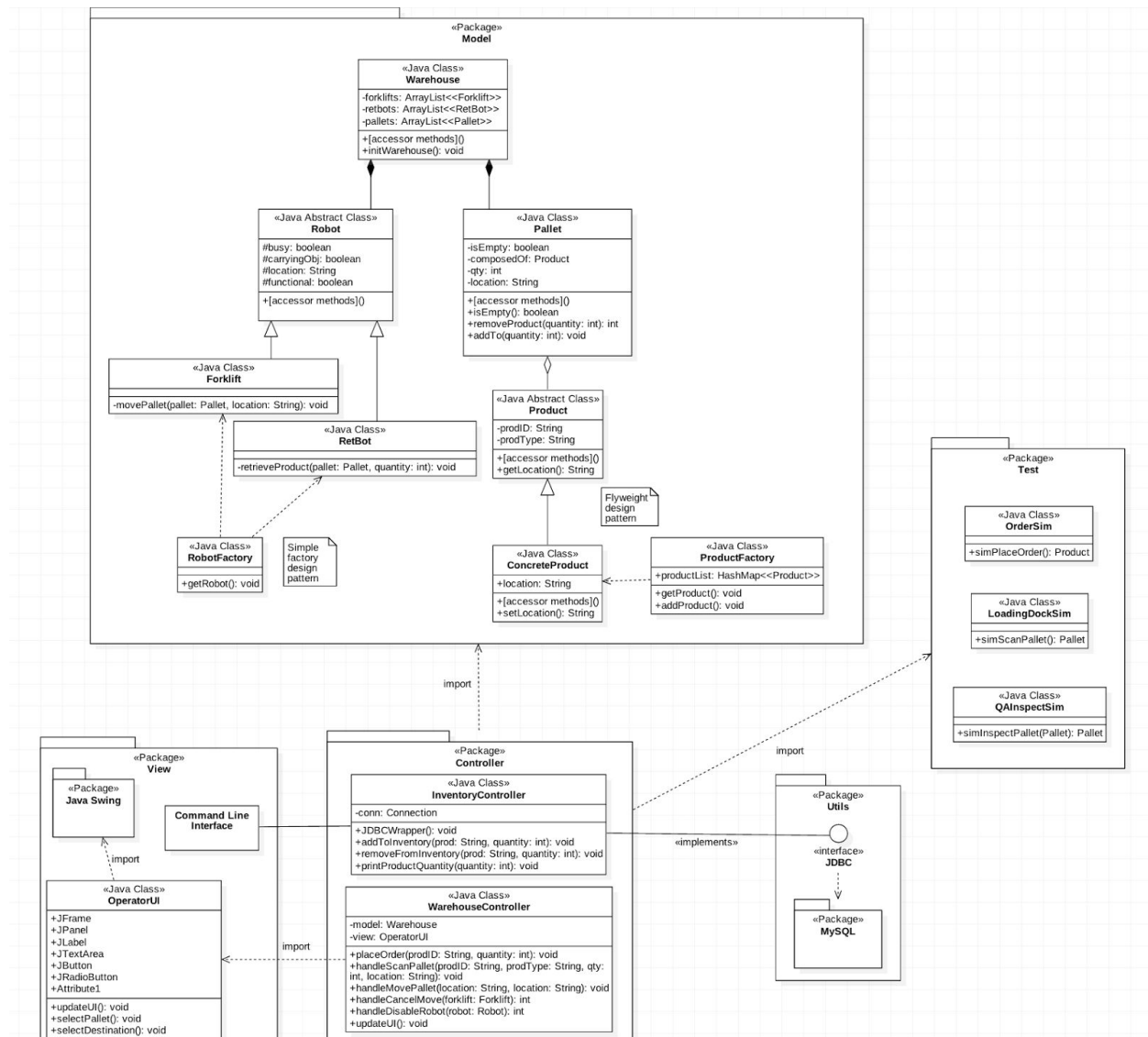
# Warehouse Management System

## CSCI 5448 Project: Part 3

Old class Diagram:



New class diagram:



We can implement two design patterns well for our project. First, we added a simple factory method that instantiates objects of the Robot or Forklift classes for the clients that will be accessing them. Next, we use the flyweight pattern for our Products. This is because many different products will share the prodID and prodType attributes, so there is no need to create numerous objects that will each hold these same values. We will hold the product types in a hash map stored in the ProductFactory class, and we will only create a new product if there is no existing product of that type in the hash map. Otherwise, the client that uses these Products will set each Product's location, which is the extrinsic value.

As for other refactoring, we have added a model and view attribute to the Warehouse Controller class, as the controller will hold both of those instances so it can communicate with each.

