

Daniel Rumfelt  
Grand Canyon University  
CST-239 Programing in Java 2  
Milestone 3  
12/18/2022

---

Assessment Description:

Analyze and model out, using UML class diagrams, an Inventory Manager that supports the following features:

Initialization of the Store inventory (should be invoked when the Store Front starts up).

Removing a Salable Product from Store inventory (should be invoked when a Salable Product is purchased).

Adding a Salable Product to Store inventory (should be invoked when a Salable Product purchase is canceled).

Return the entire inventory.

Integration of this class with the Store Front application.

Analyze and model out, using UML class diagrams, a Shopping Cart that supports the following features:

Initialization of the Shopping Cart (should be invoked when the Store Front starts up).

Adding a Salable Product to the Shopping Cart (should be invoked when a Salable Product is purchased).

Removing a Salable Product from the Shopping Cart (should be invoked when a Salable Product purchase is canceled).

Return the contents of the Shopping Cart.

Empty the contents of the Shopping Cart.

Integration of this class with the Store Front application.

Update the Salable Product UML class diagrams to support the following new features:

Update all Weapon classes so that they implement the comparable interface. Comparison should be based on the name of the item and follow alphabetical ordering rules that ignore case.

Update the Store Front UML class diagrams to support the following new features:

Integration of the Inventory Manager.

Integration of the Shopping Cart.

Update the flow chart of the logic of a Game User interacting with the Store Front and with the internal Inventory Manager and Shopping Cart.

Implement the code for all UML class designs.

Document all code using JavaDoc documentation standards and generate the JavaDoc files.

Create a screencast video that includes a functional demonstration of the application and a code walk-through explaining how the design and code work. The screencast video should be 8–10 minutes long.

Submit all code to an approved GCU GIT private repository.

Link Submissions:

Video: <http://somup.com/c3lqfPwpd4>

GitHub: [https://github.com/GCUStudent-DanielRumfelt/Week\\_5\\_Activity\\_10](https://github.com/GCUStudent-DanielRumfelt/Week_5_Activity_10)