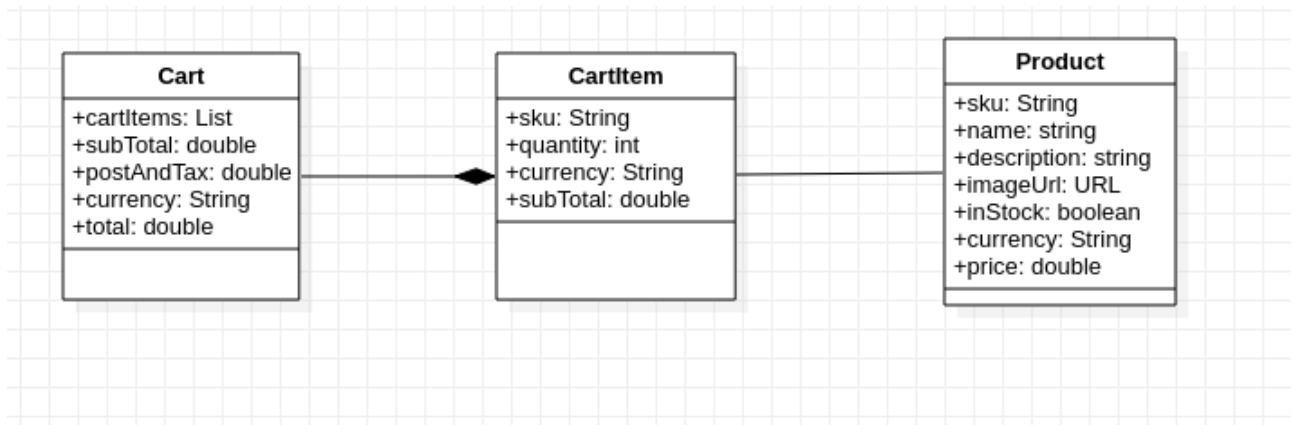


Coding Exercise (2.5h)

Create a "Spring Boot" application that produces a list of REST endpoint to perform basic operations on a shopping cart. The basic entities of the application should be:



The test product database will be provided in JSON format and supplied to you in the email [product_database.json](#)

Guidelines

- You need to produce code that is or was written by you.
- The scope of this exercise is to showcase that you can produced code that is clear and effective code that is implemented to the highest standard
 - docs, comments, logging, unit testing, code coverage.
- You may use any other java libraries or framework to facilitate your tasks.
- You may use any embedded databases such as H2 or Derby
- Minimum Java JDK 8.

What you need to implement:

- Product backlog is loaded in the database when application is started.
- Design a REST API to add a *CartItem* to the *Cart* by passing a SKU (product_id) and optionally the quantity.
 - Adding the same cart item will increase that cart item quantity.
- You can remove a *CartItem* from the *Cart* by passing a product SKU (product_id) and optionally the quantity.
 - Removing multiple cart items will decrease that cart item quantity.
- You can view the *Cart* (with all its cart items).
- You can list all the *Product*.
- Setup a scheduled task (every 2h) which updates the total of the cart with the latest currency exchange using this [currency exchange api](#).

Considerations:

- A sku is a stock keeping unit and it is an unique product identification number.
- `postAndTax` attribute can be calculated as a 10% of `subTotal`.
- The default cart currency is EUR, while products are based in GBP, so the total needs to be translated in EUR.
- Return correct error message and HTTP status code if:
 - Cart item is added to the cart when it is out of stock.
 - Cart item is added to the cart when the sku doesn't exists in the product catalog.
- Validate the API inputs:
 - Valid sku (not empty).
 - Valid quantity (positive integer).

Technical Questions (0.5h)

1. If the shopping cart API (for the same person) is shared between multiple concurrent applications (Mobile, Web, Tablet) at the same time, how would you handle cart state synchronization so that if i add a cart item to the Mobile the Web and Tablet application see the item added?
2. How would you improve performance of REST Service in particular of GET requests?
3. How would you secure this REST service considering that it will be deployed in a cluster environment?