COM 528 AE2 Report

Alastair Cox

**Domain Model and use cases**

When designing my model, I started with the template created and started by adding a catalogue page to add new items and created a stock level system which could be used to show how many items were available. I then added a basket page to purchase goods and then an orders page to view previous orders. I kept the design very basic and linked the new pages through the model, view and controller.

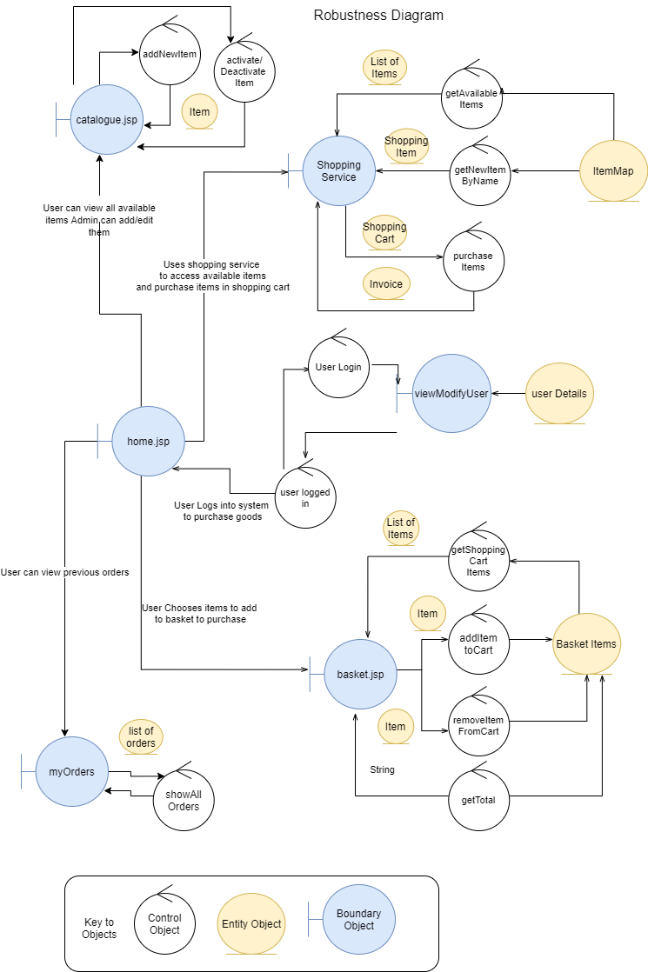
I created the use cases to match these added features including:

* The user/admin login
* User saving card details
* Adding item to stock
* Choosing items for the basket
* Purchasing items
* Viewing orders
* View a failed transaction

I chose these features as they show the system working in a way that which most reflects a real goods purchasing page and links most closely to the features required for the system. I believe if I had more ability, I would have added more use cases and additional features which would have improved the system, such as item searching or the ability to re order a previous order.

**Diagram creation**

When creating my diagrams I used the UML addin for the class diagrams as these show all the different parts of the system, from the bank, DAO, model, service and web. I took the template robustness diagram and added some of the new features which I added to the system, such as the catalogue and basket area where the user could purchase goods and the admin add new items.



**Code and design differences**

I had created buttons for the items which were supposed to activate and deactivate the items so that the users could not choose them, but I was unable to make this work properly. I started this early in the project as it was taking to much time to try and solve, so I moved onto other areas.

I would have given more time made it possible to change the stock levels of each product individually instead of just being able to add a new product.

Currently I have not finished the my orders page as I am unable to save the order once it is completed.

Presently the stock level does not decrease when the order completes, nor does the basket clear.

**Critical evaluation**

I found this module very challenging as it was the first time I had used java as a programming language. I know that some of the design and creation of my code could be better, however I found the design and implementation very hard due to all the various structures and elements of a java program. Improvements could have been made in the layout by making the products easier to search for. The safety of the card details is in question as the Shop owners card details are saved in the code ready to be used. It took me too long to try and understand some of the basic parts of the program and java as a whole which meant I was unable to complete any of the advanced parts of the module.