COMET SALE

Team 5

Team Members:

Jiawei Gu

Ankita Ahir

Melvin James

Deepan Verma

Akhilesh Kumar Kagalvadi Chinnaswamy

Table of Contents

Table of Figures

ITERATION - 2

1. OVERVIEW

The Comet Goods application is a comprehensive selling and buying portal with a minimalistic design that allows students to buy and sell goods within their community. This application has been created with the intention that only students can buy and sell products thus ensuring convenience and trust in the process.

Iteration 2 consists of:

1. Understating the Use Case scenarios of the system.
2. Working on the Domains Model, Class Diagram, Sequence diagrams and Interaction diagrams.
3. Designing the system architecture and deciding on how the system interacts with the user.
4. User can view different products posted by various sellers on the home screen.
5. A user can search for a particular product according to need.
6. From the display screen the user can add the desired product into wish-list.

2. VISION

We are developing a system which will ease out the buying and selling the goods by incorporating the searching ability of items. Further, the system will be trustworthy as only UTDallas students are allowed in the application. The system will also allow the students to search for any prospective buyers or seller so that they can plan their sales accordingly.

In addition to that, the students can get good deals on the applications and will be able to compare the prices of the interested goods. Further, the system will also notify all the interested users of the item when the item changes. For example, when an item is sold off, the users who are interested in buying those items will be notified via an email.

Comet Sale is a web based application for students of UTD to exclusively buy and sell items. The proposed features of this application have been discussed in the next section

3. REQUIREMENT SPECIFICATION:

Comet Sale application requirement are described below into three categories.

3.1. Functional Requirement:

This section deals with requirement that deals with direct functionality of the system.

3.1.1. User Registration and Login

UR.001 The system shall allow only UTD Students to register with application and provide details.
UR.002 The system shall allow registered students to log-in.

3.1.2. Sell Product

SP.001 The system shall allow student user to post an item.
SP.002 The system shall allow student seller to lock a buyer for item, until the sale is complete.
SP.003 The system shall allow the student seller to view the seller history

3.1.3. Buy Product

BP.001 The system shall allow student buyer to search items based on category.
BP.002 The system shall allow student buyer to add/remove an item to wish-list for buying.
BP.003 The system shall allow student buyer to make offer for an item.
BP.004 The system shall allow student to rate the seller.
BP.005 The system shall allow to get the contact information of seller.
BP.006 The system shall allow student buyer to view the buyer history.

3.2.Non-Functional Requirement:

This section deals with requirement which include performance, quality,

Interface requirements.

3.2.1. Performance requirements

3.2.1.1 The system must be capable of handling registering 4000 students.

3.2.1.2 The system must be capable of running more than 50 users at time.

3.2.1.3 The system must run on a 4 GB RAM on a single machine.

3.2.2. Quality requirements

3.2.2.1 The system must support HTML 5.

3.2.3. Security requirements

3.2.3.1 The system should protect the user information from outside UTD user.

3.2.3.2 The system must be capable of authenticating UTD students with help of net id.

3.2.3 Interface requirements

3.2.4.1 The system will be a web application accessed by a browser in phone or laptop.

4. DOMAIN MODEL

Some of the important conceptual classes in the system are identified and the association and the interactions between the various classes are represented in the above diagram. Some of the key interactions and associations are:

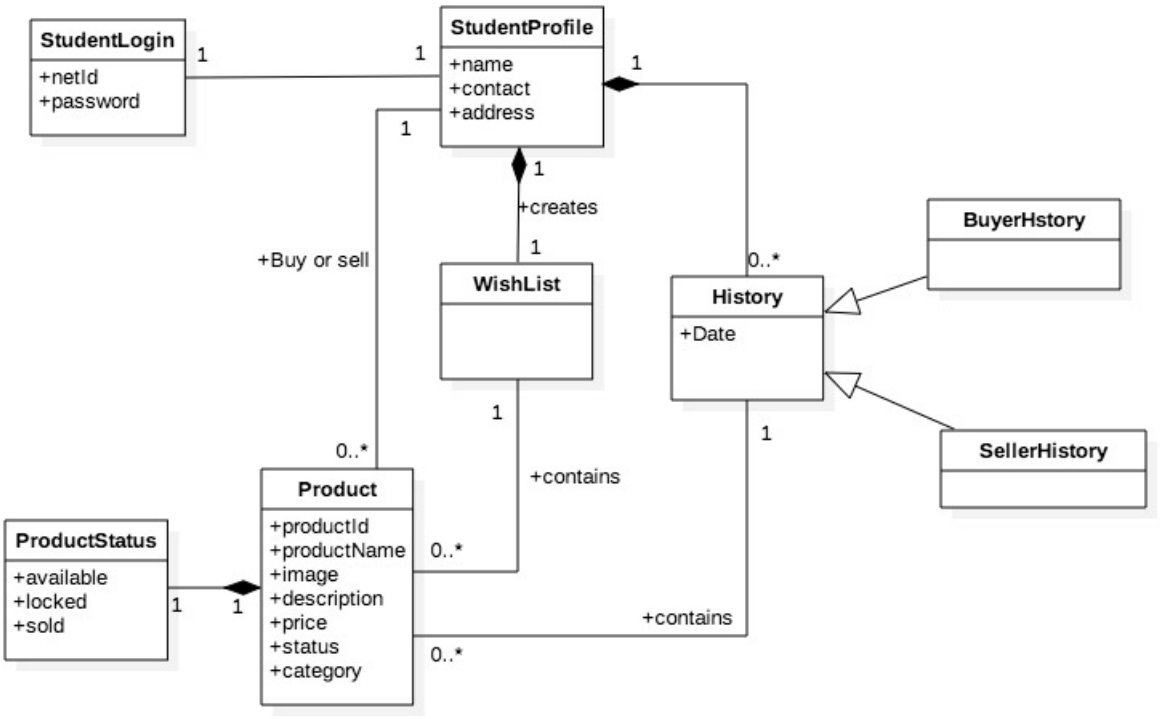

Figure 4.1:Domain Model

1. User can Login.
2. Users can search for products to buy.
3. User can sell a product.
4. Users can create wish lists.
5. Users can have Buyer and Seller history.
6. Seller can lock the product.

5. USE CASE DIAGRAMS

The following diagram is the generalized use case diagram for the entire system. The use case scenarios that follow it correspond to each use case.

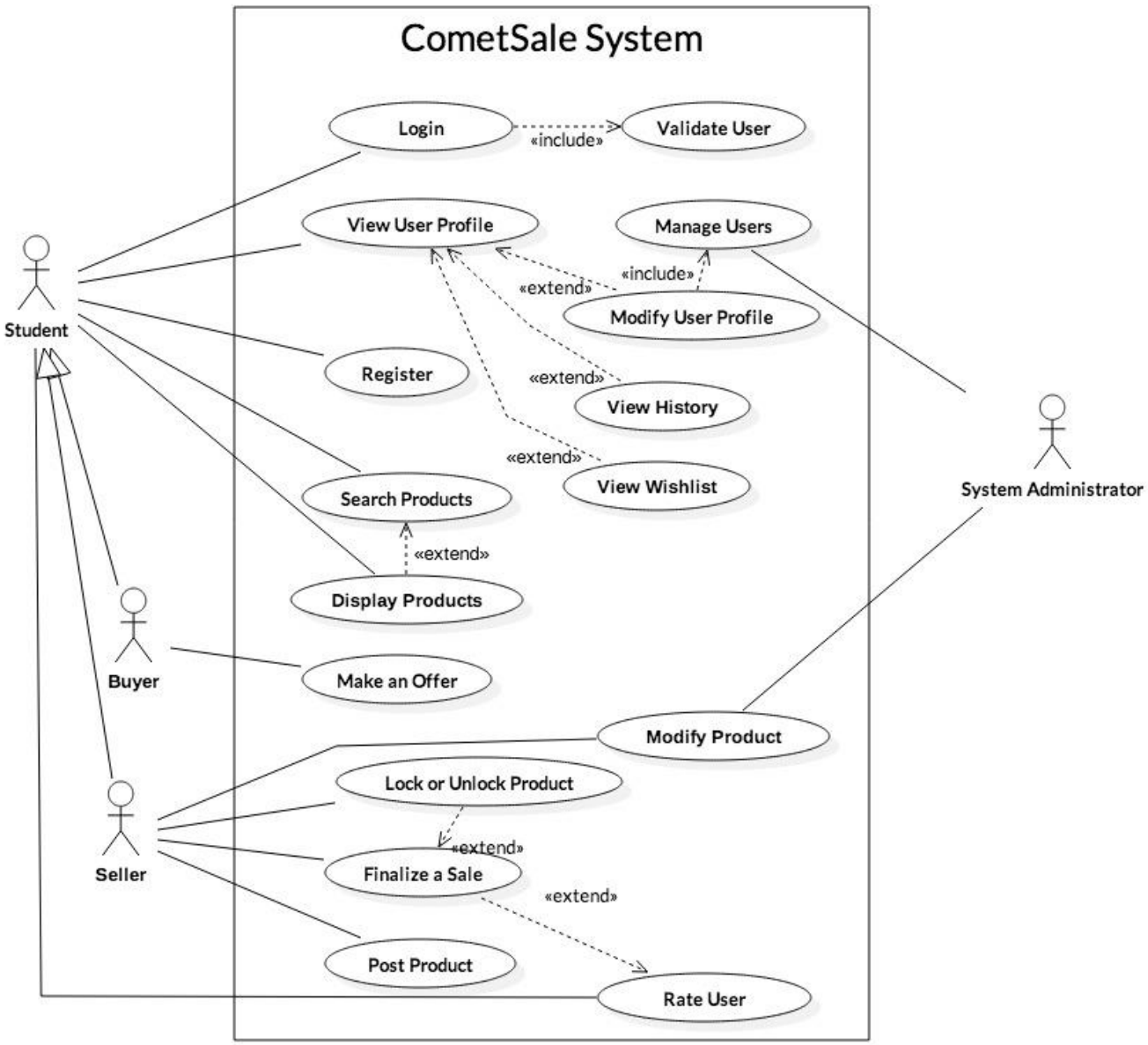

Figure 5.1: Use Case Diagram

5.1.UC-001 REGISTER

USE CASE NAME	Register
SCOPE	Comet Sale
LEVEL	User-goal
ACTORS	A student user who can use his UTD net-Id for login who intend to buy or sell a product and has not registered yet.
STAKEHOLDERS	1. Student who registers to the system using his UTD net-Id. 2. The system which validates the student registration details and sends the confirmation to the user.
PRECONDITIONS	The Student has click on the “register now” button

MAIN SUCCESS FLOW	*1.* The user enters in their information. *2.* User clicks on the Register button. *3.* System verifies information and registers the user.
ALTERNATE FLOW	**1a. Cancel Registration:** 1. The user clicks on the cancel option. 2. The system returns the user to the home page without the user being logged in and any information entered has been erased. **2a. Invalid Information Entered:** 1. User clicks submit after entering information system asked for. 2. System displays information with appropriate message to correct invalid information. 3. User re-enters information.
POSTCONDITIONS	**Successful Condition:** The user entered successful information and is given a success confirmation message. **Failure Condition:** User enters some invalid registration details and given an error message.

5.2.UC-002 POST PRODUCT

USE CASE NAME	Post Product
SCOPE	Comet Sale
LEVEL	User-goal
ACTORS	Registered User
STAKEHOLDERS	1. Registered User of the system. 2. The system which validates the product details entered by the user and sends the confirmation to the user.
PRECONDITIONS	The Student has click on the Sell Product button.
MAIN SUCCESS FLOW	1. The user enters in the product information. 2. User clicks on the 'Add Product button' 3. System verifies information and posts a product for sale.
ALTERNATE FLOW	**2a Invalid Information Entered:** 1. User clicks submit after entering information system asked for. 2. System displays information with appropriate message to correct invalid information. 3. User re-enters information.
POSTCONDITIONS	**Successful Condition:** The user entered successful information and is given a success confirmation message. **Failure Condition:** User enters some invalid registration details and given an error message.

5.3.UC-003 DISPLAY PRODUCTS

USE CASE NAME	Display products

SCOPE	Comet Sale
LEVEL	User-goal
ACTORS	Registered User
PRECONDITIONS	User should be logged in to the system.
MAIN SUCCESS FLOW	1. User is shown the list of all the available products in the system and the use case completes successfully.
ALTERNATE FLOW	**2a** If there are no products available in the system that are available for purchase.
POSTCONDITIONS	**Successful Condition:** 1. System displays the list of all the products available in the system with all the product details for the products. 2. System displays the list of the products whose category matches with the search keyword. **Failure Condition:** 1. If there are no products, user is shown a message: "No products available".

5.4.UC-004 SEARCH PRODUCTS

USE CASE NAME	Search products
SCOPE	Comet Sale
LEVEL	User-goal
ACTORS	Registered User(Primary)
STAKEHOLDERS	User **-**Enters the search keyword for the category he wants to search. System**-** Displays the results according to the search criteria.
PRECONDITIONS	User should be logged in to the system.
MAIN SUCCESS FLOW	1. User enters the search keyword and press the enter button or clicks on the search button. 2. The system searches the results according to the search criteria given by the user. 3. System displays the list of the products to the user and the use case ends successfully.
ALTERNATE FLOW	**1a** User does not enter anything in the search box and clicks on the search button. **1b** User enters invalid input like just white spaces or invalid characters.
POSTCONDITIONS	**Successful Condition:** System displays the list of the products whose category matches with the search keyword. **Failure Condition:** If there are no products that matches the search criteria, user is shown a message: "No products available with this search criteria."

5.5.UC-005 ADD PRODUCT TO WISH LIST

USE CASE NAME	Add a product to wish list
SCOPE	Comet Sale
LEVEL	User-goal
ACTORS	Registered User(Primary)
STAKEHOLDERS	User- Add the product to wish list System- Adds the product to the wish list of the user.
PRECONDITIONS	User should be logged in to the system.
MAIN SUCCESS FLOW	1. User clicks on the 'Add to Wish List' button of a product from the list of the products available to buy. 2. The system adds the product in the wish list of the buyer. 3. User can see the product in his/her wish list and the use case ends successfully.
ALTERNATE FLOW	**1a** If the product is already in the users wish list then the user is not shown "Add to wish list" button.
POSTCONDITIONS	**Successful Condition:** User can see the product in his/her wish list

5.6.UC-006 REMOVE A PRODUCT FROM WISH LIST

USE CASE NAME	Remove a product from wish list
SCOPE	Comet Sale
LEVEL	User-goal
ACTORS	Registered User(Primary)
STAKEHOLDERS	User- Remove a product from wish list System- Removes the product from the wish list of the user.
PRECONDITIONS	1. User should be logged in to the system. 2. The user should have at least one item in the wish list.
MAIN SUCCESS FLOW	1. User is shown list of the products in the wish list he has added previously. 2. User clicks on 'Remove from Wish List' button. 3. The system removes that particular product from the wish list of the user and the use case completes successfully.
ALTERNATE FLOW	**2a** User has no products in the wish list.
POSTCONDITIONS	**Successful Condition:** System removes the product from the users wish list and notifies the user with message: "Removed from wish list successfully." **Failure Condition:** The product is not removed from the wish list if not removed properly.

6. CLASS DIAGRAM

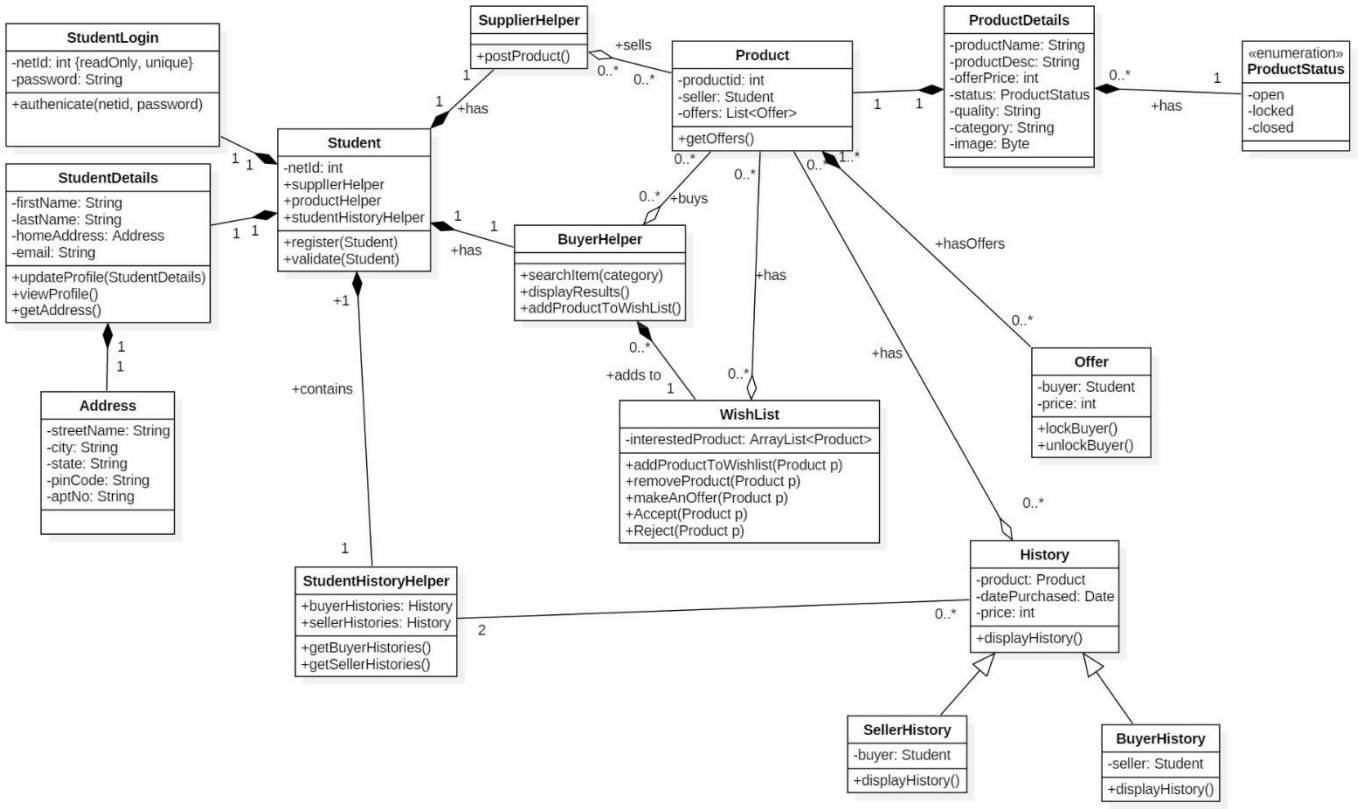

Figure 6.1: Class Diagram

7. SEQUENCE DIAGRAM

A Sequence diagram is an interaction diagram that shows how processes operate with one another and in what order.

UML sequence diagrams model the flow of logic within your system in a visual manner, enabling us to document and validate the system logic, and are commonly used for both analysis and design purposes.

7.1.Registration

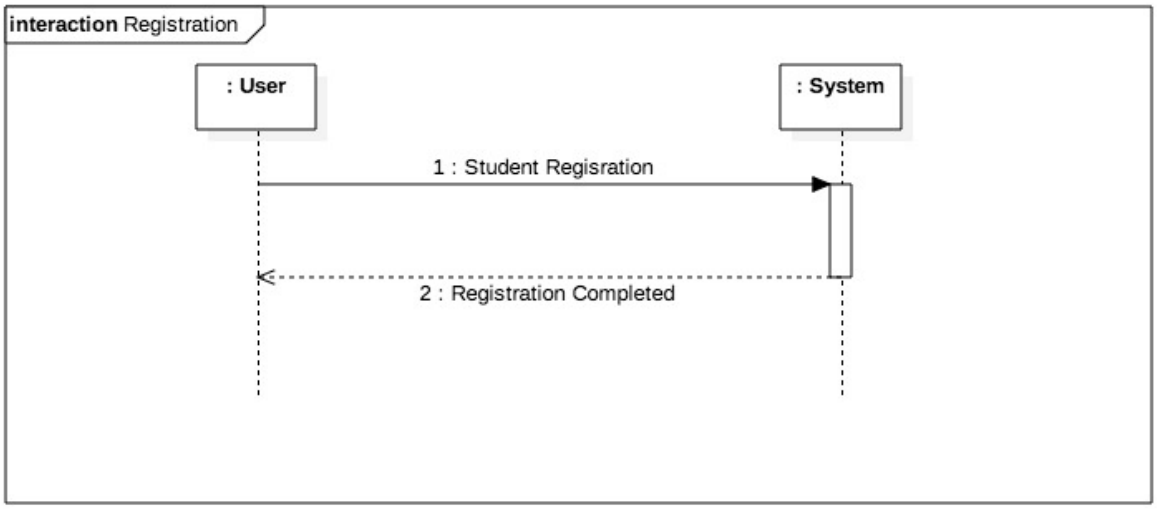

Figure 7.1: Registration

7.2.Add Product

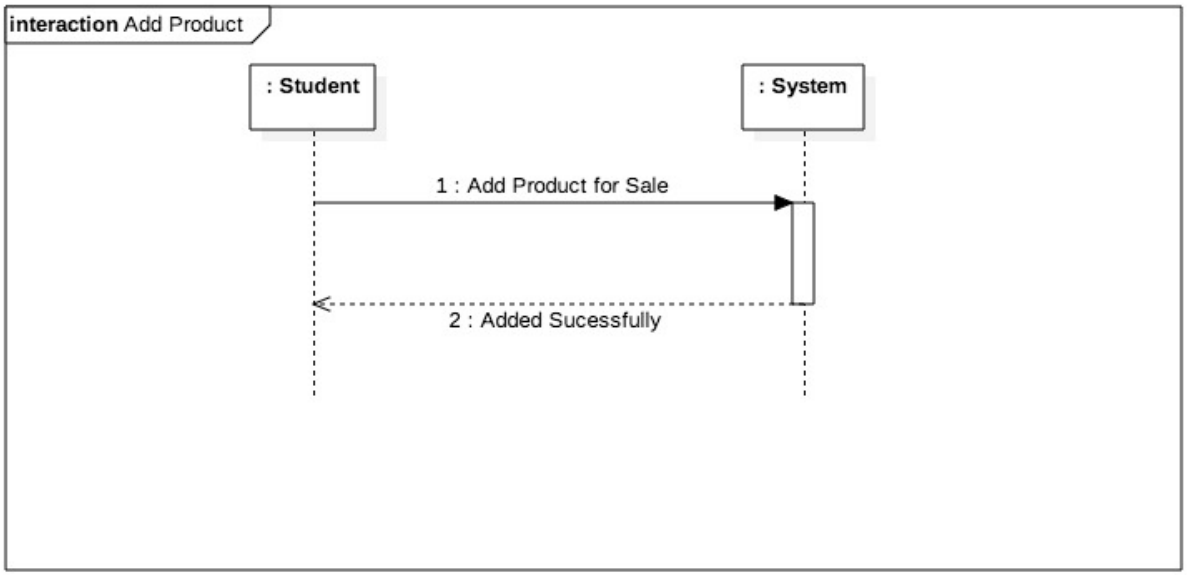

Figure 7.2 Add Product

7.3.Search Product

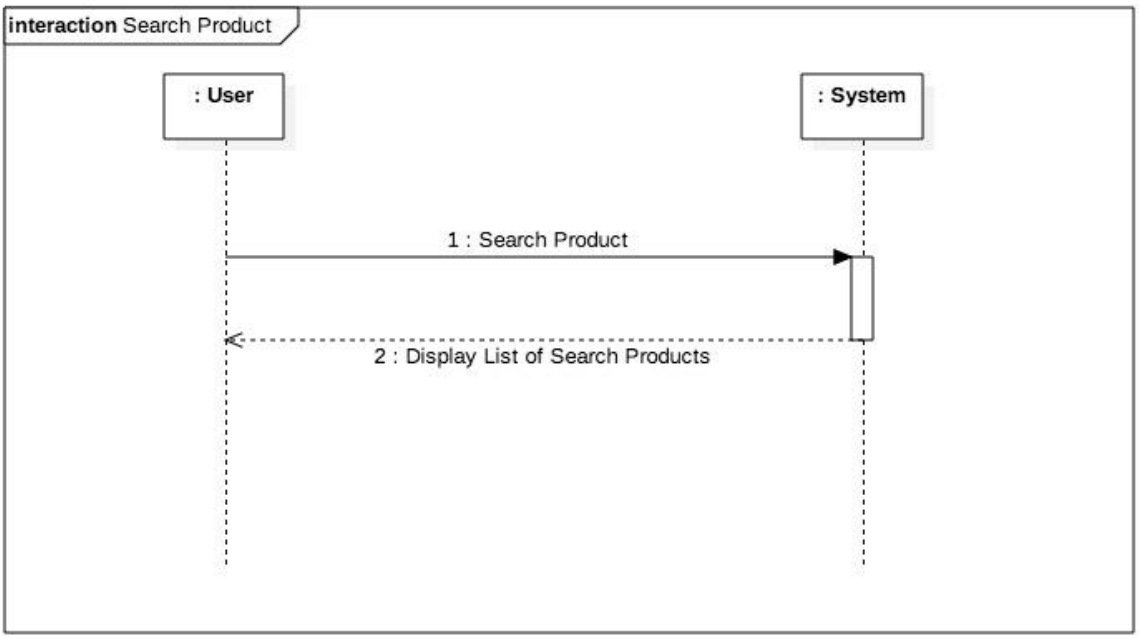

Figure 7.3 Search Product

7.4.Add Product to Wish List

Figure 7.4 Add Product to Wish List

7.5.Remove Product from Wish List

Figure 7.5 Remove Product from Wish List

7.6.Display Products

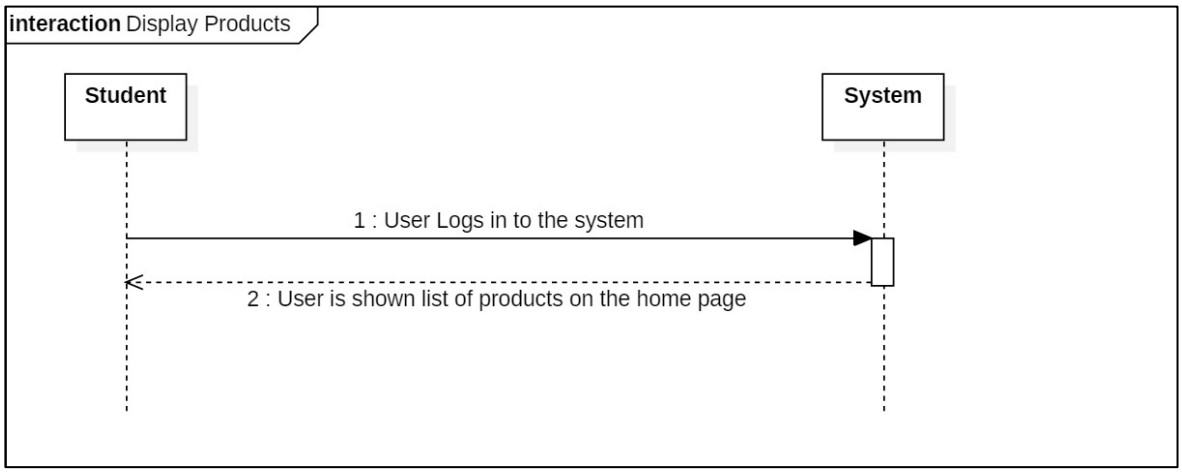

Figure 7.6 Display Products

8. INTERACTION DIAGRAMS

8.1.Registration

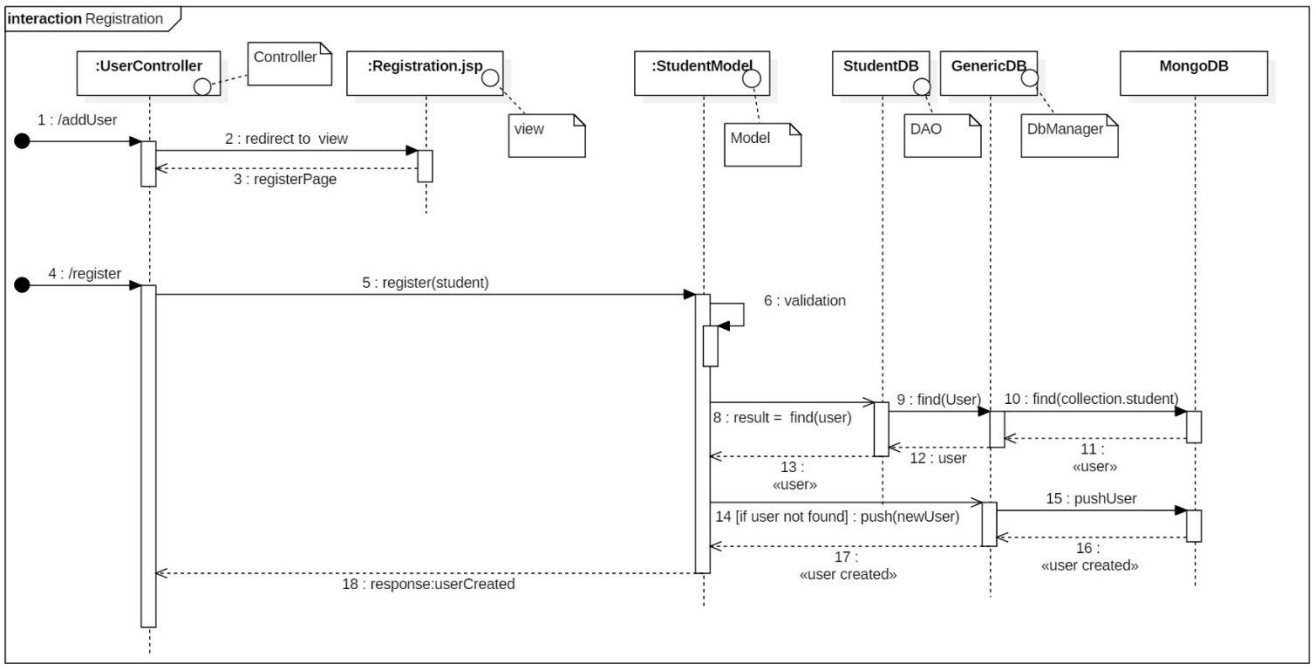

Figure 8.1 Interaction: Registration

8.2.Add Product

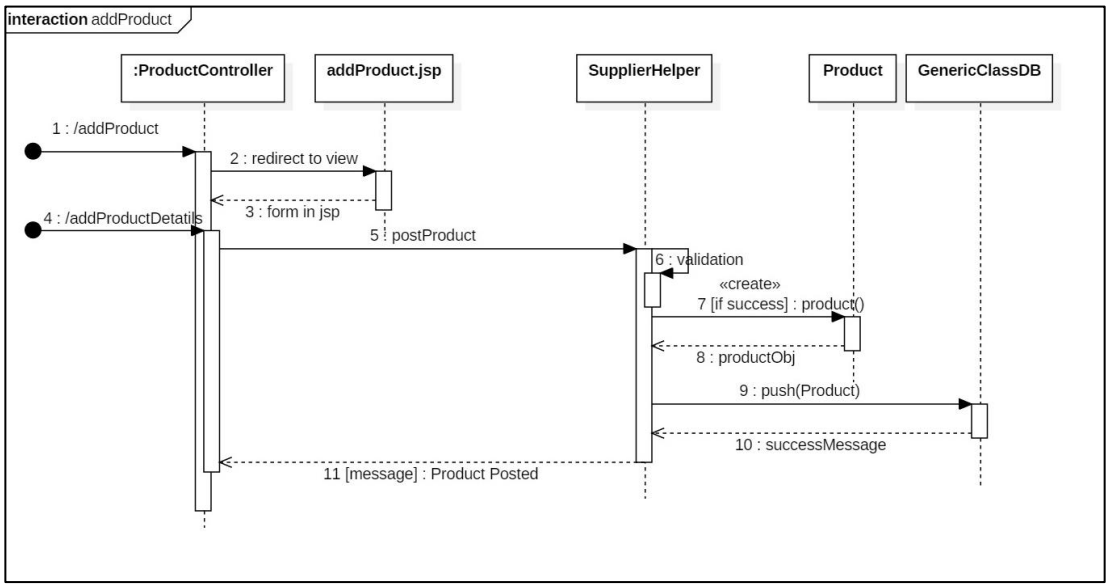

Figure 8.2 Interaction: Add Product

8.3.Search Product

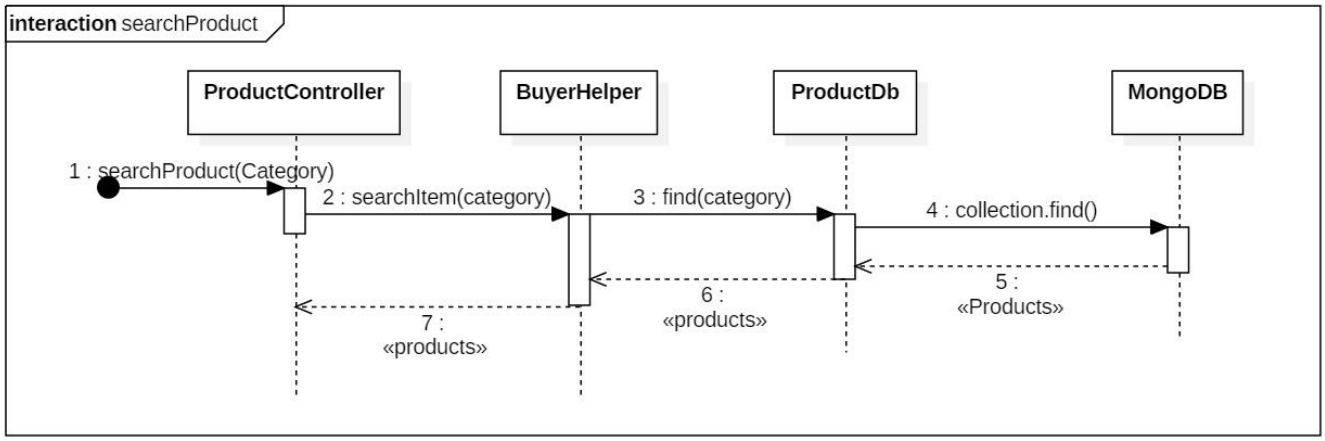

Figure 8.3 Interaction: Search Product

9. TESTING

Testing is done on the software to check if the quality is maintained and implementation is according to requirements stated earlier.

1. Meets the requirements that guided its design and development

2. Works for correct inputs and handles incorrect inputs

3. The system should provide output within an acceptable time limit

4. Is sufficiently usable

5. Achieves the general result expected by Stakeholder.

9.1.TEST PLANS:

- The following activities and test cases are used to check for the quality of code and to map the implementation with the requirements of the system.
- Test the code with various inputs and make sure that the Use cases related to buy and sell methods are working.

- The testing was performed on this iteration by providing inputs to the application and checking if the application worked as per the requirements and the use cases.

9.2.TEST CASES

For User Registration

Test No:	Test Case	Test Case Description	Expected Output	Actual Output
1	User Registration With valid information	• User opens the application • Enter all correct details to register himself to the system	The user is able to register himself to the system	The user is able to register himself to the system
2	User Registration With invalid information	• User opens the application • Enter some incorrect details to register himself to the system	The user is unable to register himself to the system	The user is unable to register himself to the system

For Adding Product Details

Test No:	Test Case	Test Case Description	Expected Output	Actual Output
1	Add Product using With valid information	• User opens the application and clicks on add product for sale • Enter all correct details to add product into system	The user is able to add the product details he wants to sell to the system	The user is able to add the product details he wants to sell to the system
2	Add Product using With invalid information	• User opens the application and clicks on add product for sale • Enter some incorrect details to add product into system	The user is unable to add the product details he wants to sell to the system	The user is unable to add the product details he wants to sell to the system

For Searching Products

Test No:	Test Case	Test Case Description	Expected Output	Actual Output
1	Search Product using With valid keyword	• User opens the application and clicks on search bar • Enter name of the product to be searched	The user is able to view the details of the products from Database	The user is able to view the details of the products from Database
2	Search Product using With invalid keyword	• User opens the application and clicks on search bar • Enter some invalid inputs like special symbol to be searched	The user is unable to view the details of the products from Database	The user is unable to view the details of the products from Database

For Adding Product to Wish List

Test No:	Test Case	Test Case Description	Expected Output	Actual Output

1	Click on the “Add to wish-list button”	• Message is displayed that says that it is successfully added to wish-list.	The user is able to view the details of the products from Database in wish-list	The user is able to view the details of the products from Database in wish-list

For Removing Product to Wish List

Test No:	Test Case	Test Case Description	Expected Output	Actual Output
1	Click on the “Remove from wish list button”	• Message is displayed that says that it is successfully removed from wish list.	The user is able to see that it is no longer present in wish-list	The user is able to see that it is no longer present in wish-list

10. ScreenShots

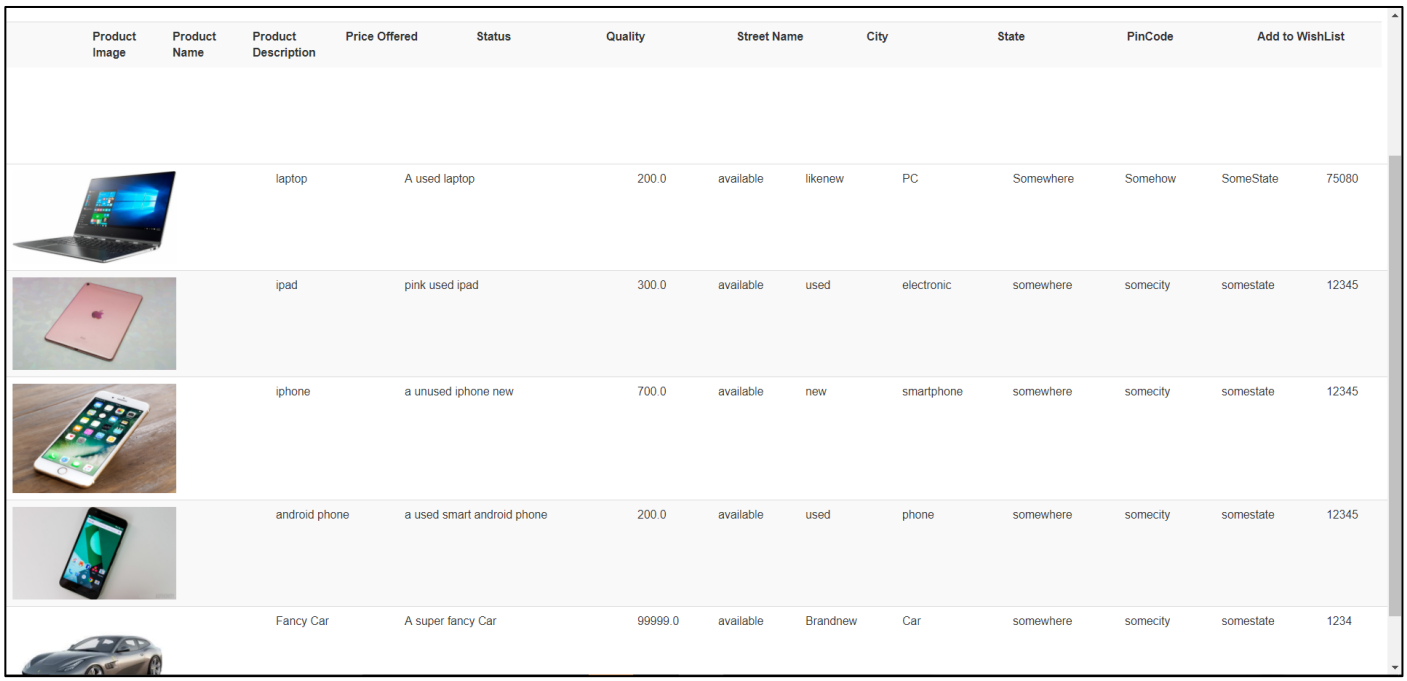

Figure 10.1 Display Products

11. SOFTWARE AND TOOLS USED:

- StarUML for creating class diagrams, Use cases.
- Eclipse as IDE.
- Java Spring MVC Framework
- MongoDB
- Trello

12. PLAN FOR NEXT ITERATION

1. Requirement analysis
2. Creating Module in which the buyer confirms buying products from wish list.
3. Creating a module where the seller can lock a product for the selected buyer.
4. Implementing module to filter Products by Category.

5. Designing User Interface for Buy and Sale History.
6. Documentation and Report

13. Glossary

CometSale: Comet Sale is the name of application being developed, that helps UTD student to buy/sell within the group.

Student: Refers to the UTD student who will be using the system.

Post an item: Student, who is wishing to sell an item can post his description of the item.

Make offer: Student can provide a price to seller by using make offer option.

Delivery: Student can get to know about the delivery information.

Item Category: student can maintain category of item like Phone, Laptop, Clothes for easy identification of items.

Lock buyer: The capability in a system, where a seller can freeze other buyers from offering by locking a buyer.

Comments: The part of system, where user and buyer can interact for buying and selling.

Seller: The role of student in a system, where his functionality is to sell an item.

Buyer: The role of student in a system, where his functionality is to look for an item to buy.

net ID: The Unique id used by UTD student for uniquely identifying a student.