# COMET SALE


Team 5

**Team Members:**

Jiawei Gu

Ankita Ahir

Melvin James

Deepan Verma

Akhilesh Kumar Kagalvadi Chinnaswamy

## Table of Contents

# ITERATION - 1

## 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 1 consists of:

1. Understating the Use Case scenarios of the system.
2. Working on the Domains Model, Class Diagram and sequence diagrams.
3. Designing the system architecture and deciding on how the system interacts with the user.
4. User can register into the system by entering the user details.
5. User can post the product description of new product for sale.

## 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. This application will have following features:

## 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.The system shall allow only UTD Students to register with application and provide details.
3.1.2.The system shall allow student user to post an item.
3.1.3.The system shall student buyer to express his interest in an item for buying.
3.1.4.The system shall student buyer to make offer for an item by providing price, delivery options.
3.1.5. The system shall allow student buyer to search items based on category.
3.1.6. The system shall allow student to rate the seller.
3.1.7. The system shall allow student seller to lock a buyer for item, until the sale is complete.
3.1.8. The system shall allow buyer to post questions via comment on items.
3.1.9. The system shall allow to get the contact information of seller.

### 3.2.Non Functional Requirement:

This sectional 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:

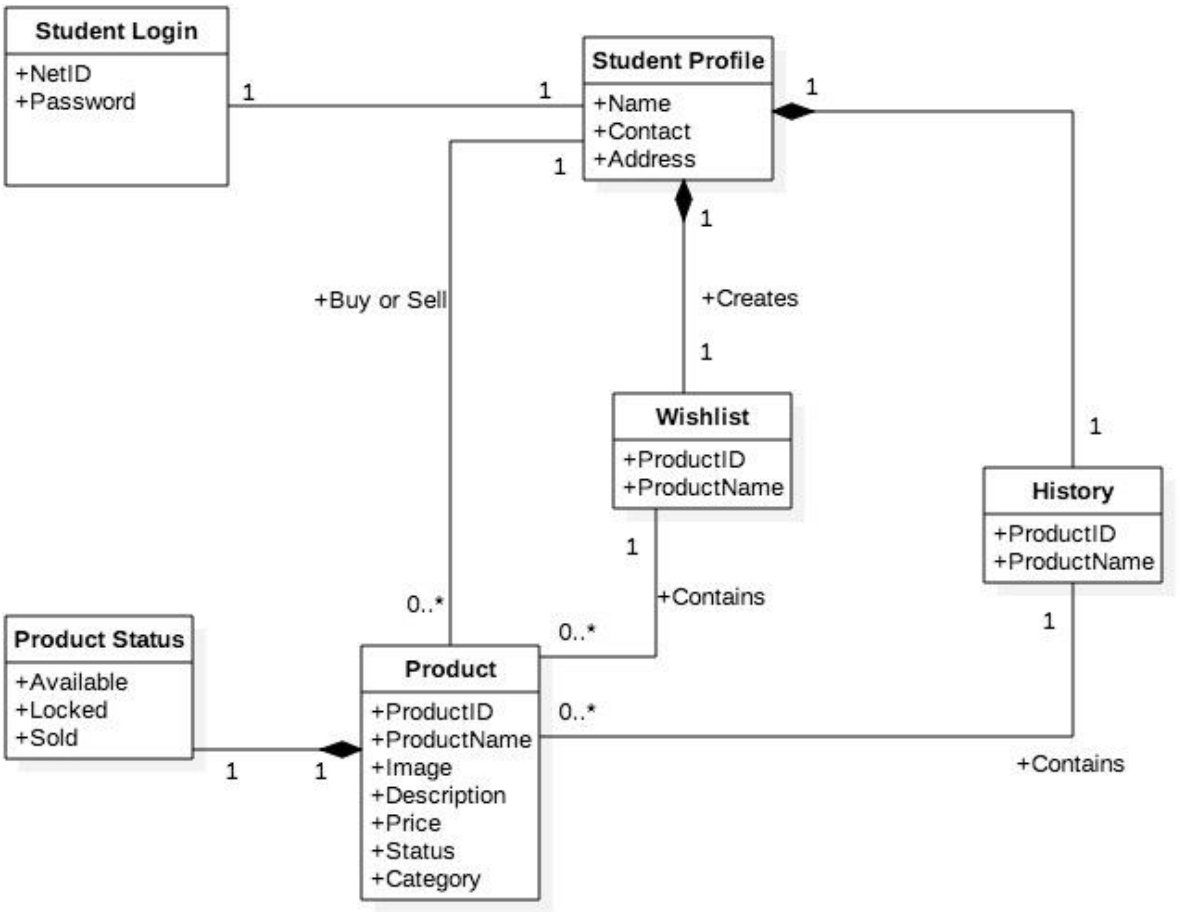


*Figure 4.1:Domain Model*

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 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.

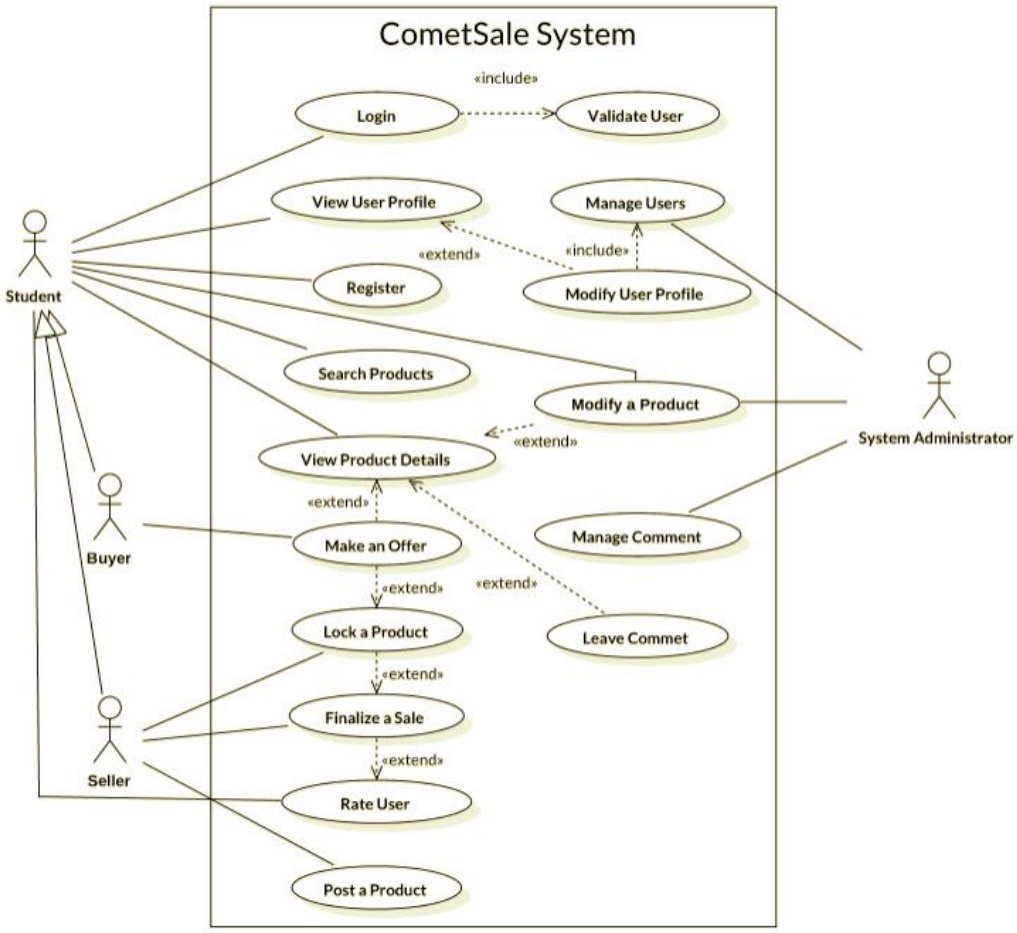


*Figure 5.1: Use Case Diagram*

## 5.2 Use Case UC1: Login

**USE CASE NAME**: Register

**ACTORS**:

- Primary: A student user who can use his UTD netId for login who intend to buy or sell a product.
- The student has not registered yet.

**STAKEHOLDERS**:

- Student who registers to the system using his UTD netId.
- The system which validates the student registration details and sends the confirmation to the user.

**PRECONDITIONS**:

- The Student has click on the register button.

**BASIC FLOW**:

1. The System prompts the user for a username and password or register new account.
2. The user selects registration option.
3. The System prompts user for registration information NetID, password, etc
4. The user enters in their information.
5. System verifies information and creates registers the user.

**ALTERNATE FLOW:**

**Cancel Registration:**

- The user selects the cancel option.
- The system returns the user to the home page without the user being logged in and any information entered has been erased.

**Invalid Information Entered:**

- User clicks submit after entering information system asked for.
- System displays information with appropriate message to correct invalid information.
- User re-enters information.

**POSTCONDITIONS**:

- **Success**: The user entered successful information and is given a success confirmation message.
- **Failure**: User enters some invalid registration details and given an error message.

## 5.3 Use Case UC2: Enter Product Details

**USE CASE NAME**: Product Details

**ACTORS**:

- Primary: A student user who can use his UTD netId for login who intend to sell a product.

**STAKEHOLDERS**:

- Registered User of the system.
- 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 Add Product for Sale button.

**BASIC FLOW**:

- The Student has click on the Add Product for Sale button
- The System prompts user for product information Product name, category,Product Description, etc
- The user enters in the product information.
- System verifies information and posts a product for sale.

**ALTERNATE FLOW:**

**Cancel:**

- The user selects the cancel option.

- The system returns the user to the home page without the product being posted for sale and any information entered has been erased.

**Invalid Information Entered:**

- User clicks submit after entering information system asked for.
- System displays information with appropriate message to correct invalid information.
- User re-enters information.

**POSTCONDITIONS**:

- **Success**: The user entered successful product information and is given a success confirmation message.
- **Failure**: User enters some invalid product details and given an error message.

# 6 CLASS DIAGRAM

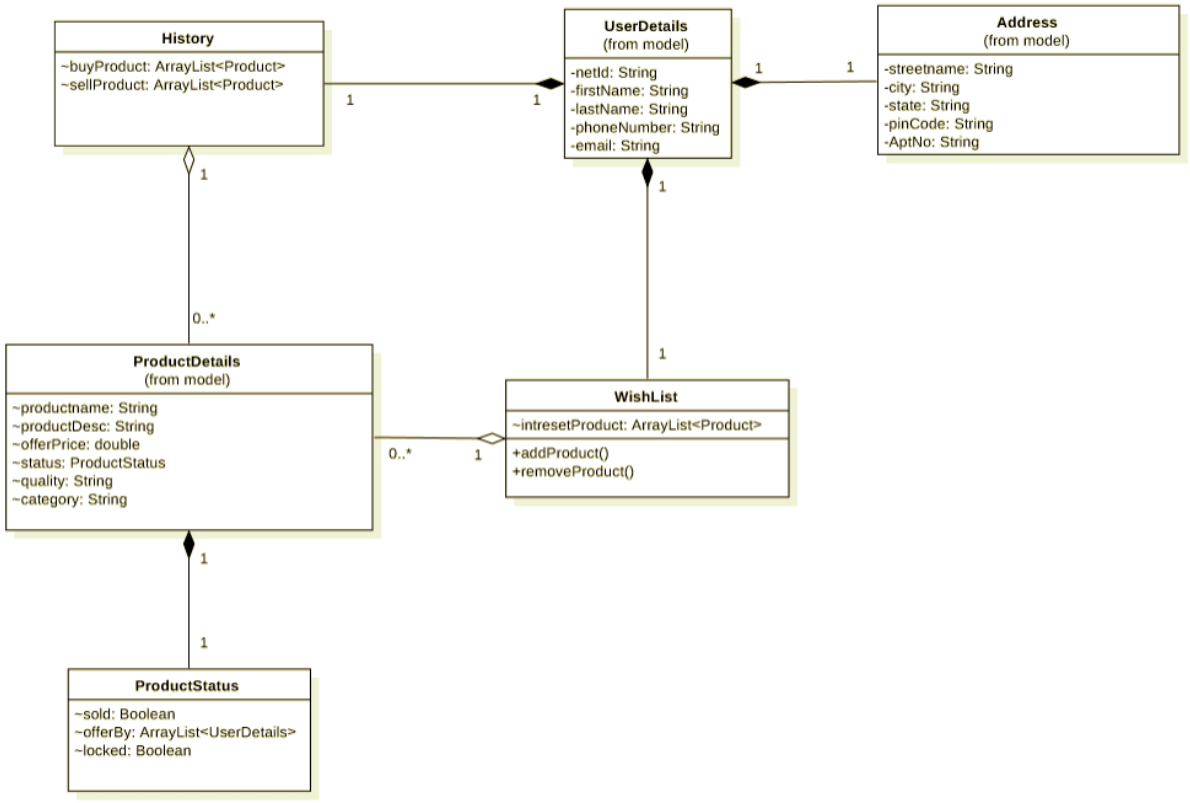


*Figure 6.1: Class Diagram*

# 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.

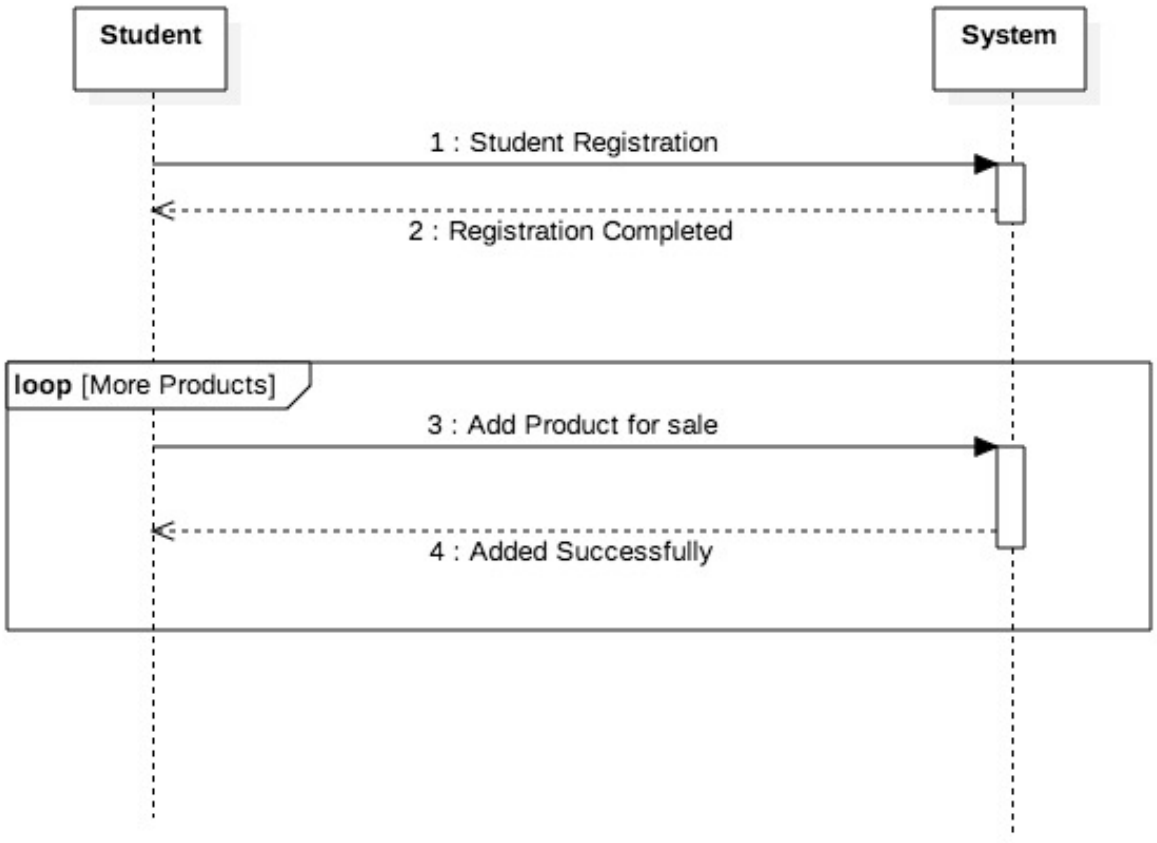


*Figure 7.1: System Sequence Diagram*

# 8 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.

## 8.1TEST 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.

## 8.2TEST 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<br>• 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<br>• 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<br>• 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<br>• 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 |

# 9 ScreenShots



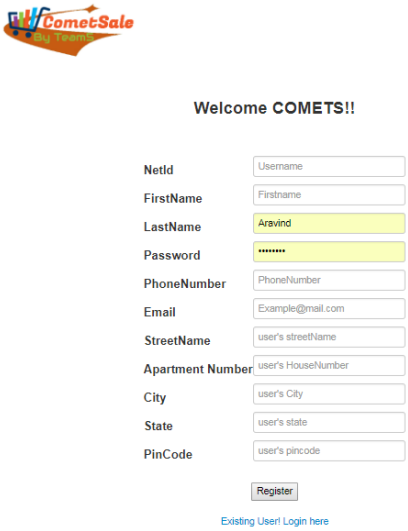


*Figure 9.1: User Registration*

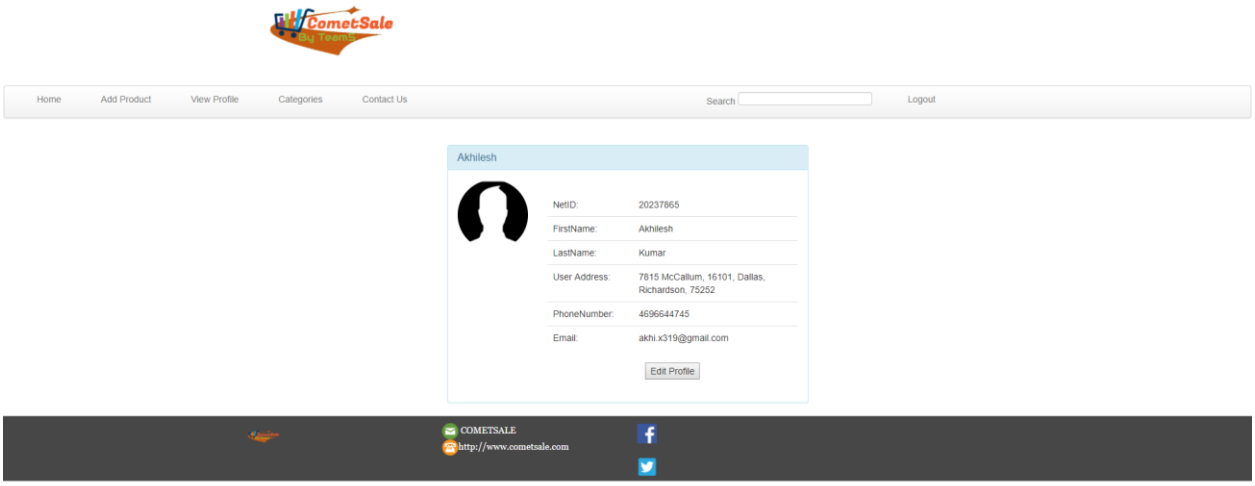


*Figure 9.2: View User Details*

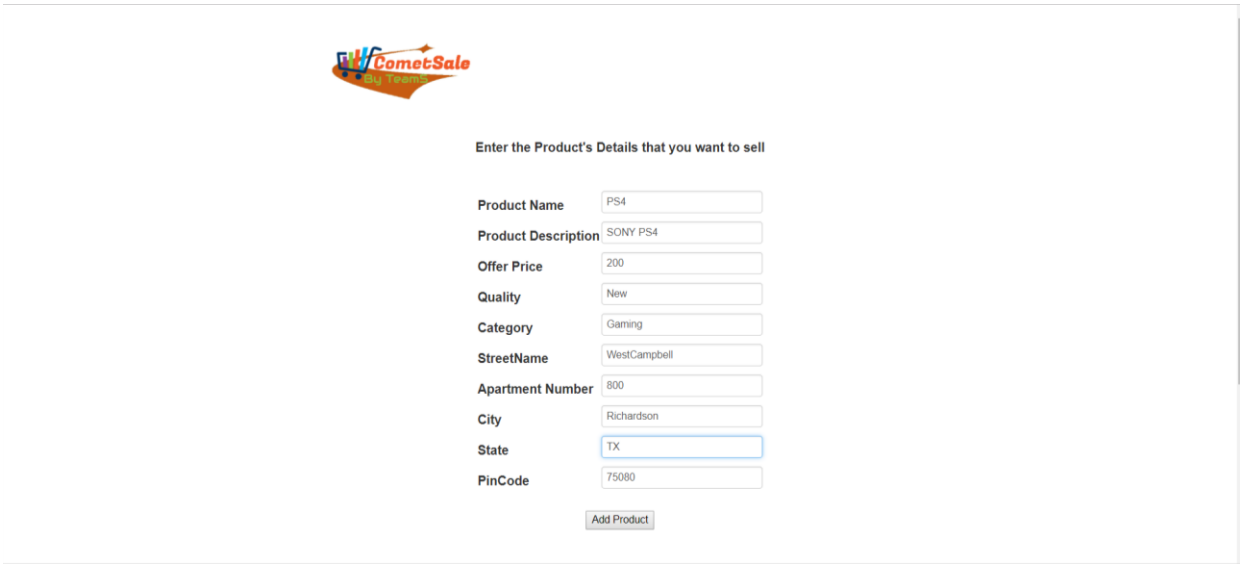


*Figure 9.3: Add Product*

# 10 SOFTWARE AND TOOLS USED:

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

# 11 PLAN FOR NEXT ITERATION

1. Requirement analysis
2. Creating Module in which the buyer can view different products.
3. Search Product by Category Module.
4. Creating a module where the seller can lock a product.
5. Designing User Interface for Buy Screen.
6. Documentation and Report

# 12. 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 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.