
# ON-LINE SALES PORTAL

## STRUCTURED ANALYSIS / STRUCTURED DESIGN DOCUMENT (SA/SD)

PRADEEP DOGGA [14CS10013]
SOHAN PATRO [14CS30044]

# CONTENTS


**1 Introduction** **4**
- 1.1 Purpose . . . 4
- 1.2 Project Scope . . . 4
- 1.3 Overview Of Document . . . 5
- 1.4 Glossary . . . 6
- 1.5 References . . . 7

**2 Feasibility Study** **7**
- 2.1 Understanding the Problem . . . 7
- 2.2 Scoping The Problem . . . 7
- 2.3 Analysing Stakeholders . . . 8
  - 2.3.1 *USE CASE : ACCESS TO BUYERS* . . . 8
  - 2.3.2 *USE CASE : ACCESS TO SELLERS* . . . 10
  - 2.3.3 *USE CASE : ACCESS TO MANAGER* . . . 11
- 2.4 Use Case Descriptions . . . 12
  - 2.4.1 Primary Use Cases . . . 12
  - 2.4.2 Included Use Cases . . . 22

**3 Analysis Model** **25**
- 3.1 Structural Model- The Class Diagram . . . 25
- 3.2 Behavioural Model . . . 27
  - 3.2.1 Sequence Diagrams . . . 28
  - 3.2.2 Collaboration Diagrams . . . 32
  - 3.2.3 Statechart Diagram . . . 37
  - 3.2.4 Activity Diagrams . . . 38

**4 Detailed Design** **44**
- 4.1 Global System Architecture . . . 44
- 4.2 Platform . . . 44
- 4.3 Software Architecture . . . 45
- 4.4 Report . . . 45

# List of Figures


1 Buyer Use Cases . . . . . 9
2 Seller Use Cases . . . . . 10
3 Manager Use Cases . . . . . 11
4 Refined Use Case Diagram . . . . . 13
5 Refined Class Diagram . . . . . 25
6 Sequence Diagram For Registration . . . . . 28
7 Sequence Diagram For Buying an Item . . . . . 29
8 Sequence Diagram For Selling an Item . . . . . 31
9 Sequence Diagram For Reviewing an Item . . . . . 32
10 Collaboration Diagram For Registration . . . . . 33
11 Collaboration Diagram For Buying an Item . . . . . 34
12 Collaboration Diagram For Selling an Item . . . . . 35
13 Collaboration Diagram For Reviewing an Item . . . . . 36
14 Statechart Diagram Of An Item . . . . . 37
15 Activity Diagram For Buying An Item . . . . . 41
16 Activity Diagram For Selling An Item . . . . . 42
17 Activity Diagram For Reviewing An Item . . . . . 43

# 1 Introduction

The following subsections of the Software Analysis / Software Design (SA/SD) document provide an overview of the entire SA/SD.

## 1.1 Purpose

The puropse of this SA/SD is to provide a detailed description of the various analysis and design models of the Online Sales Portal software product. It gives a deep insight into the functional features of this software with the various interfaces available for the online customers and the manager. This products aims to a provide a one stop platform for buyers to buy products and sellers to sell products online.

This document comprises of all the models required in the various analysis and the design processes involved in the development of the software product.

## 1.2 Project Scope

This online portal system will serve as a one stop store for people who can buy and sell on the same platform. The system can be divided into the following three subsystems :

1. Selling Item.
2. Buying Item.
3. Managing the system.

The first and second sub-system comprise the transactions between sellers and buyers. The third sub-system is the main one; it helps the system to remain updated, maintained and also manage the customers. There are three types of end users in the system :

- Seller : For the first sub-system.
- Buyer : For the second sub-system.
- Manager : For the third sub-system.

More specifically, this system will be designed to allow the manager to manage and communicate with customers to help better transactions through E-mail. The system uses MySQL database to maintain the records of the users and their transactions.

## 1.3 Overview Of Document

This section describes the organisation of the contents of the remaining sections of the document as listed below :

- Section 2 is about the Feasibility Study which helps to understand the problem in depth and discuss its' scope. In this section, we analyse the various stake holders and define possible alternatives. We also consider the assessment of unusual circumstances and evaluate alternatives. We also look into it for deeper information. Also in this section we gather functional and non-functional requirements. Here we discuss the various accessibilities of the stake holders with the help of use case diagram.

- Section 3 is the final and the most important section : Analysis Model. It in turn contains sub sections which are : Structural model and Behavioural Model. The structural model comprises the class diagram and it's description. The behavioural model describes the behaviour and interaction of the software using sequence diagram , collaboration diagram, state chart diagram and activity diagram.

## 1.4 Glossary

| Term | Definition |
|---|---|
| SA/SD | Software Analysis / Software Design : A document that provides a detailed description of the various analysis and design model of the software product. |
| OSP | Online Sales Portal : The proposed sytem to be designed to provide a one-stop store for the users to buy and sell at the same platform. |
| Buyer | A customer who has an intention to use the system to buy an item |
| Seller | A customer who has an intention to use the system to sell an item |
| Manager | Person who takes the responsibility to manage the customers and transactions |
| Database | Collection of all the information monitored by the system. |
| IEEE | Institute of Electrical and Electronic Engineers |
| End user | All the buyers , sellers and manager |

## 1.5 References

[IEEE]The applicable IEEE standards were approved by IEEE-SA Standards Board in its' 1998 edition. [IEEE Std 830-1998]

# 2 Feasibility Study

## 2.1 Understanding the Problem

In this section we try to understand the purpose of this software i.e., the expected functions from the project to cope with the requirements of the user.
The OSP is intended to provide better platform for buyers and sellers to do transactions online for mutual benefit without much effort. The software helps the sellers to comfortably upload details of item which they wish to sell.
The software helps the buyers to browse through the items of interest and also categories of interest to search for a best match with the best price.
The software also helps the buyers and sellers to have negotiations for a better business transaction over email through the portal.
The portal also facilitates the buyers to pay bills online for the item they purchase after negotiations are done and accepted by the seller.
This product facilitates the manager to manage (block/help) customers. He can also manage (add/remove/change) the categories of items present in the portal. He can also help in the negotiations between the buyer and the seller for a good satisfactory transaction. He can also perform audit of the matched-buy sell of items. He can also review the quality of items put up by the sellers and take necessary actions if the quality is observed poor.

## 2.2 Scoping The Problem

In this section, we specify the scope of the problem and the various functions the software is expected to perform :

- Upload an item for sale by the seller.

- Search for an item by the buyer to get the list of sellers willing to sell it.
- Raise a request by the buyer to a seller to buy an item.
- Optional offering of a price and negotiations on the selling price.
- Online bill payment for the item purchased after the offer is accepted by seller.
- Management of the customers by the manager.
- Helping negotiations between customers by the Manager.
- Management of categories of items to be uploaded in the portal.
- Reviewing quality of items by the manager and rejection of items depending on quality.
- Perform audit of the matched buy-sell of items.

## 2.3 Analysing Stakeholders

The various stakeholders include :

1. Customers
    - Buyers.
    - Sellers.
2. Manager.

### 2.3.1 *USE CASE : ACCESS TO BUYERS*

#### Brief Description :

The buyer can register into the system buy entering all the required details. The portal generates a unique id and a password for the buyer. The buyer uses these as the credentials to login to the system to make use of it. The buyer searches for an item of interest to find the list of sellers willing to sell it. The buyer then selects a seller out of the list of sellers. The buyer then raises a request to the selected seller.

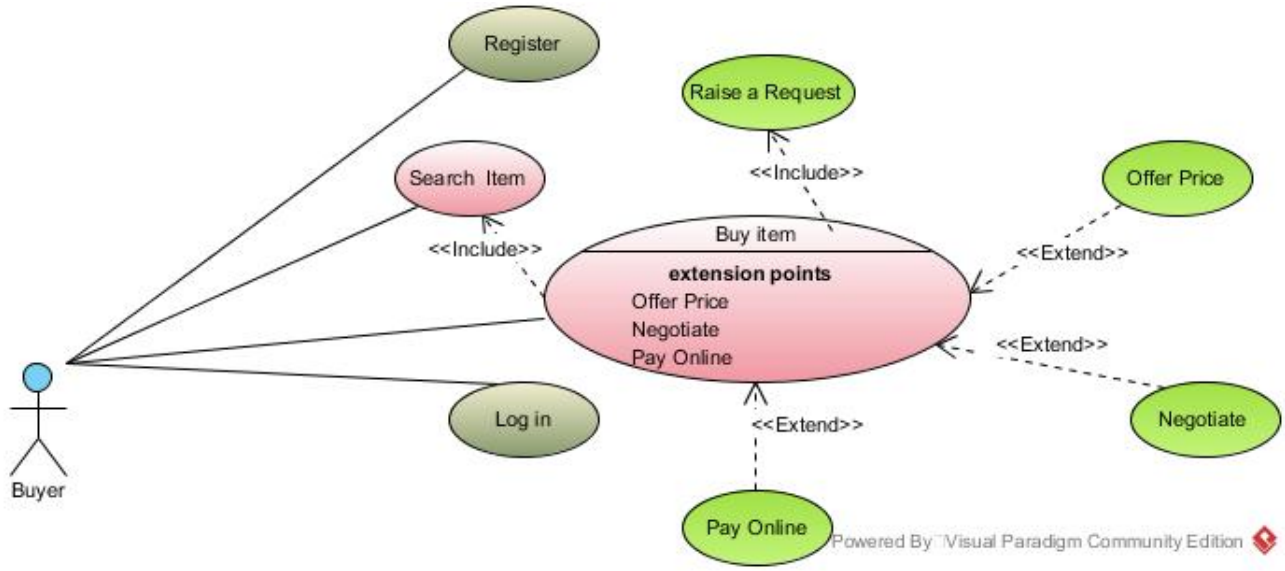


Figure 1: **Buyer Use Cases**

The buyer may offer a price to the item. The buyer can negotiate with the seller and finally comes to a conclusion and if the seller accepts the offer by the buyer , then he proceeds to pay the bill online for the item.

## Interface Details for Buyer :

The interface for the buying process is specified in the SRS document. All the graphical interfaces for the buyer are specified in detail there.

### 2.3.2 *USE CASE : ACCESS TO SELLERS*

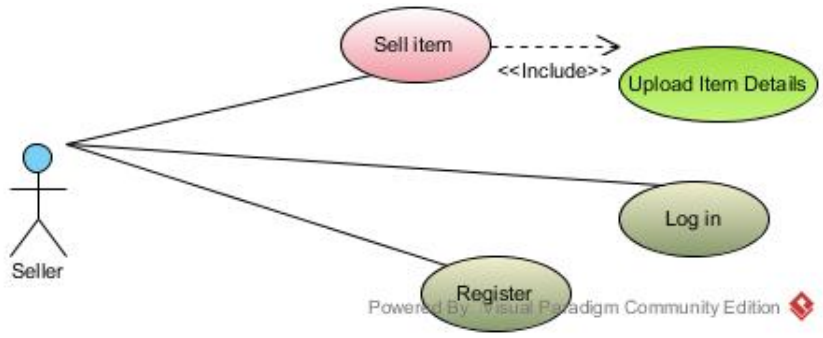


Figure 2: **Seller Use Cases**

## Brief Description :

The seller can register into the system buy entering all the required details. The portal generates a unique id and a password for the seller. The seller uses these as the credentials to login to the system to make use of it. The seller can upload the item details by providing all the required information of the item interested. The item is then reviewed by the manager and then is made available for buyers to purchase.

## Interface Details for Seller :

The interface for the selling process is specified in the SRS document. All the graphical interfaces for the seller are specified in detail there.

### 2.3.3 *USE CASE : ACCESS TO MANAGER*

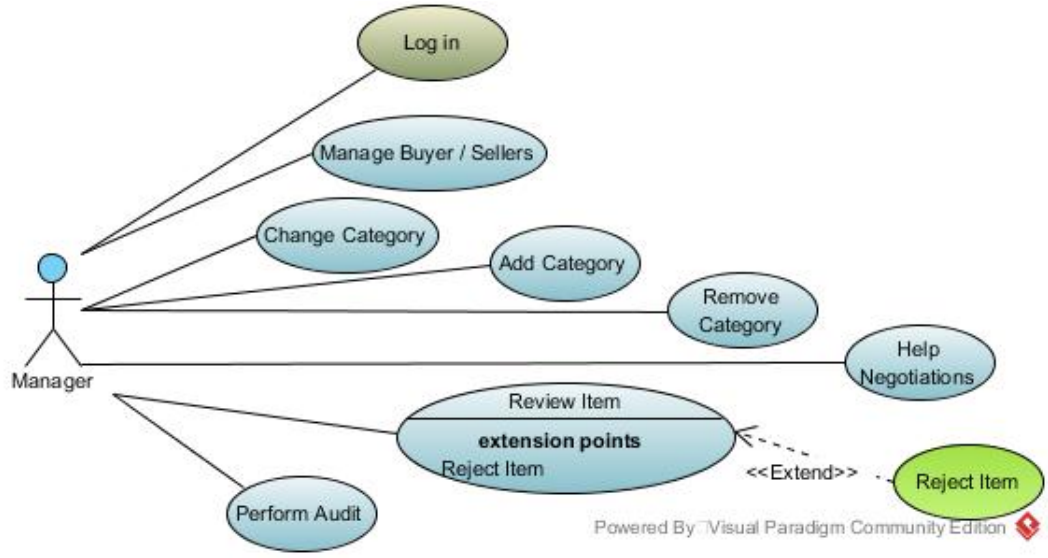


Figure 3: **Manager Use Cases**

## Brief Description :

The Manager uses his/her unique id and password as the credentials to login to the system to manage it. The manager can manage customers. The manager can block a customer upon finding unacceptable usage of system. The manager can also help the customers by resetting the password in case of any trouble logging in. The manager can control the categories of the items that can be uploaded into the portal. The manager also has the access to change the category of any item uploaded by a seller. The manager also has access into the negotiation between the buyers and sellers.

The manager has access to review the quality of an item uploaded by a seller. The item is made available to sale after its quality is reviewed by the manager. If the quality of the item is found to be poor by the manager then it is rejected by the manager. The manager can perform audit of matched buy-sell of items to check if there are any discrepancies in the system for its safe functioning.

## Interface Details for Manager :

The interface for the manager to perform all the functions described above is a graphical interface. The manager can login by filling the login form described in SRS document. The manager can look at the item and click on a button called "review item" to review it's quality. His feedback can be submitted by clicking a button named "approved". If he finds the item to be of poor quality then he can reject the item by clicking a button called "reject item". The item is removed from the portal and a notification is sent to the corresponding seller.

The manager can select an item to change its category. On clicking a button called "change category" , a drop down list appears which shows all the categories present in the portal and he can choose one of them and click "change category" button to save the changes made.

The manager can add a category by clicking a button called "add category" . This subsequently opens a page which contains a text box to enter the category name and a submit button to add it to the portal.

The manager can remove a category by clicking a button called "remove category" . This subsequently opens a page which contains a drop down box which contains all the categories present. The manager can select one of them and click on a button called "remove" to permanently that category from the portal. Any items of that category if are present are displayed to the manager to change their category.

The interface for performing audits contains a button called "perform audit". Clicking the button displays the number of items sold and number of items purchased to the manager.

## 2.4 Use Case Descriptions

### 2.4.1 Primary Use Cases

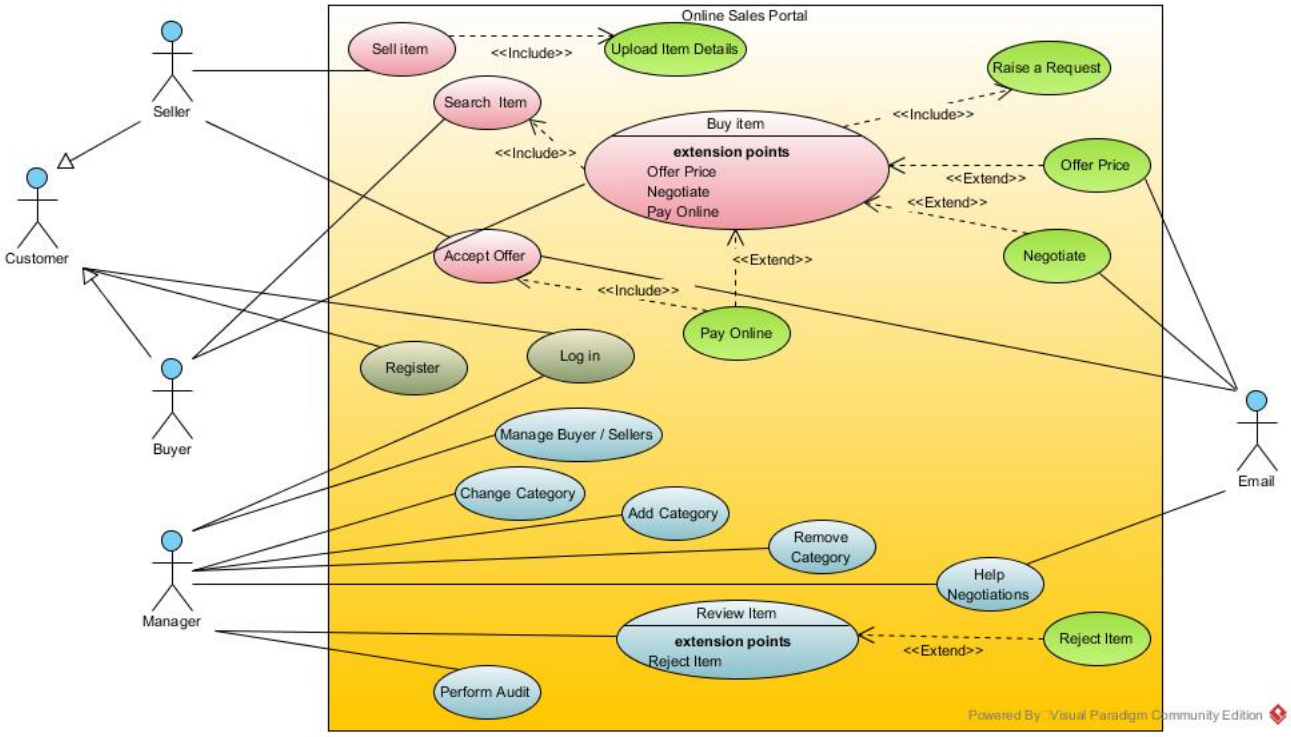


Figure 4: **Refined Use Case Diagram**

| **Use Case** | **Sell Item** |
|---|---|
| Function | Make an item available to purchase through system |
| Precondition | The seller must be logged in. |
| Successful end condition | An item is added into the item record of the system for purchase |
| Failed end condition | No item is uploaded to the item record of the system |
| Primary Actor | Seller |
| Secondary Actor | None |
| Included Cases | Upload Item Details. |
| Main Flow | 1.The seller uploads all the details of item.<br>2. Clicks on the upload button. |
| Extensions | 1. Seller cancels upload.<br>2. Incomplete details are filled, hence uploading fails. |

| **Use Case** | **Buy Item** |
|---|---|
| Function | Purchase an item through system |
| Precondition | The buyer must be logged in. |
| Successful end condition | The item is purchased by paying bill online and it is removed from the system. |
| Failed end condition | The item is not purchased. |
| Primary Actor | Buyer |
| Secondary Actor | Seller and Manager |
| Included Cases | Search Item, Raise a Request |
| Extended Cases | Offer Price, Negotiate, Pay Online. |
| Main Flow | 1.The buyer searches for an item.<br>2.Buyer raises a request to the seller.<br>3.Buyer has optional negotiation with the seller.<br>4. Buyer offer accepted by the seller.<br>5. Buyer pays bill online. |
| Extensions | 1. No seller found for an item.<br>2. Request raised by the buyer but offer not accepted by the seller. |

| **Use Case** | **Register** |
|---|---|
| Function | Creates an account in the system for a customer. |
| Successful end condition | An account is created for the customer as a seller or a buyer. |
| Failed end condition | No account is created. |
| Primary Actor | Customer(Buyer/Seller) |
| Secondary Actor | None |
| Main Flow | 1.The customer uploads the required details.<br>2. Clicks on the register button. |
| Extensions | 1. Customer cancels registration.<br>2. Incomplete details are filled, hence registration fails. |

| **Use Case** | **Login** |
|---|---|
| Function | Login to the system. |
| Precondition | The customer must be registered. |
| Successful end condition | logs in to the system and ready to use the system. |
| Failed end condition | The customer is not logged in. |
| Primary Actor | Customer , Manager |
| Secondary Actor | None |
| Main Flow | 1.The customer or manager enters the unique id and password.<br>2.Clicks login button. |
| Extensions | Invalid credentials are entered and notification is displayed. |

| Use Case | Add Category |
|---|---|
| Function | Adds a category for items in the system. |
| Precondition | Manager must be logged in. |
| Successful end condition | A new category is added to the system. |
| Failed end condition | No category is added to the system.. |
| Primary Actor | Manager |
| Secondary Actor | None |
| Main Flow | 1.The manager enters a category name.<br>2.Clicks Add Category button. |
| Extensions | None |

| Use Case | Remove Category |
|---|---|
| Function | Removes a category for items in the system. |
| Precondition | Manager must be logged in. |
| Successful end condition | The category is removed from the system. |
| Failed end condition | No category is removed from the system.. |
| Primary Actor | Manager |
| Secondary Actor | None |
| Main Flow | 1.The manager selects a category.<br>2.Clicks Remove Category button. |
| Extensions | None |

| Use Case | Change Category |
|---|---|
| Function | Change category of an item in the system. |
| Precondition | Manager must be logged in. |
| Successful end condition | The category of the item is changed. |
| Failed end condition | Item retains the category. |
| Primary Actor | Manager |
| Secondary Actor | None |
| Main Flow | 1.The manager enters a category name.<br>2.Clicks Change Category button. |
| Extensions | Entered category is not present in the system and manager is notified. |

| Use Case | Review Item |
|---|---|
| Function | Checks the quality of an item. |
| Precondition | Manager must be logged in. |
| Successful end condition | Item is reviewed. |
| Primary Actor | Manager |
| Secondary Actor | None |
| Main Flow | 1.The manager checks the quality of item.<br>2.If the item is of poor quality it is rejected and removed. |
| Extensions | Item is not of poor quality and it is made available for purchase. |

### 2.4.2 Included Use Cases

| Use Case | Upload Item Details |
|---|---|
| Function | To enter the information of an item for sale. |
| Precondition | The seller must be logged in. |
| Successful end condition | An item is added into the item record of the system for purchase |
| Failed end condition | No item is uploaded to the item record of the system |
| Primary Actor | Seller |
| Secondary Actor | None |
| Base Use Case | Sell Item. |
| Main Flow | 1.The seller uploads all the details of item.<br>2. Clicks on the upload button. |
| Extensions | 1. Seller cancels upload.<br>2. Incomplete details are filled, hence uploading fails. |

| Use Case | Search Item |
|---|---|
| Function | To search for an item and get the list of sellers in the portal |
| Precondition | The buyer must be logged in. |
| Successful end condition | Item is found and list of sellers is displayed. |
| Failed end condition | No seller for the item is found |
| Primary Actor | Buyer |
| Secondary Actor | None |
| Base Use Case | Buy Item. |
| Main Flow | 1.The buyer enters category name of item.<br>2. Clicks on the search button. |
| Extensions | 1. No seller is found. |

| Use Case | Raise a Request |
|---|---|
| Function | To raise a request to the seller to buy the item. |
| Precondition | The buyer must be logged in and selected the seller. |
| Successful end condition | The seller receives the request of the buyer. |
| Failed end condition | The seller does not receive any request from the buyer. |
| Primary Actor | Buyer |
| Secondary Actor | E-mail |
| Base Use Case | Buy Item. |
| Main Flow | 1. A request is made to the seller of the item through email<br>2. Clicks on the send request button. |
| Extensions | None. |

# 3 Analysis Model

## 3.1 Structural Model- The Class Diagram

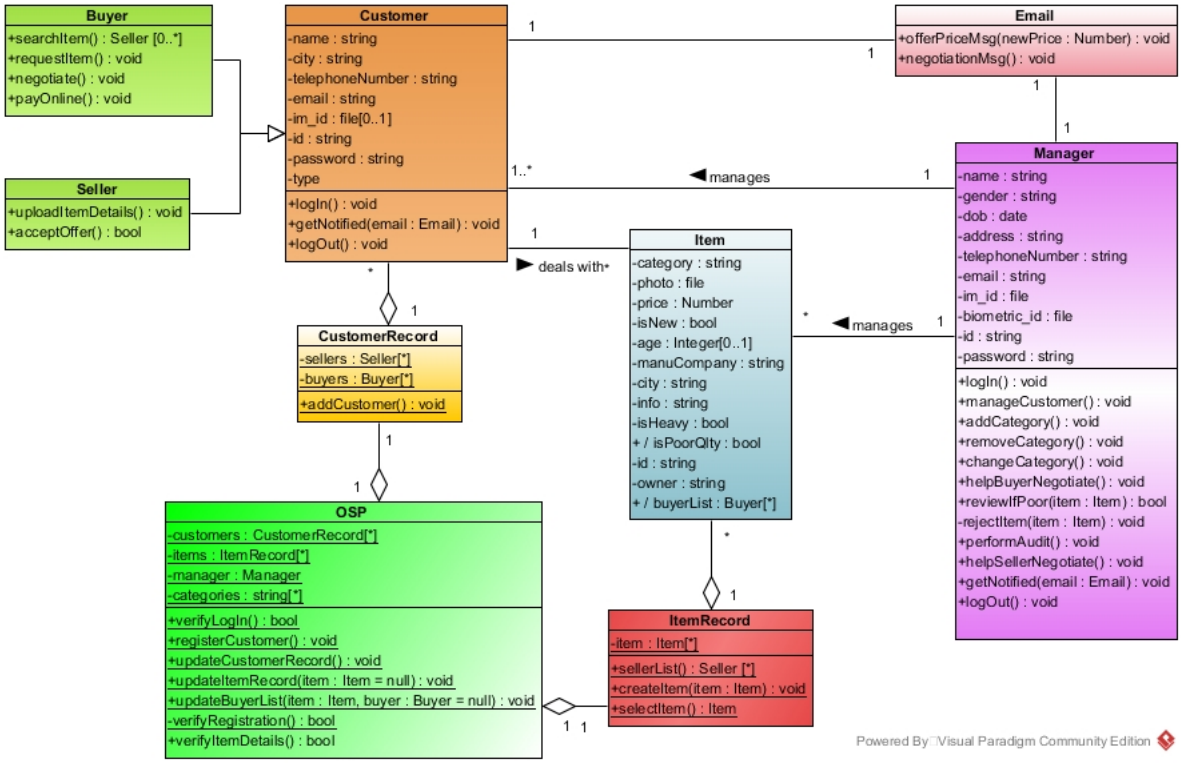


Figure 5: **Refined Class Diagram**

The above Class Diagram consists of the following classes :

- **Customer** : This class represents the buyers and sellers as a whole and contains the common attributes and operations of buyers and sellers. In the inheritance hierarchy it acts as a parent class for the following two sub-classes :
  - **Buyer** : This sub-class of customer can do the operations specific to it such as searching an item, Raising a request , negotiating with the seller and paying bill online.

  - **Seller** : This sub-class of customer can do the operations specific to it such as uploading item details for sale, accepting an offer from a buyer and negotiating with the buyer.

- **Manager** : This class represents the manager of the Online Sales Portal who is responsible for management of the portal. It contains the information of the manager and operations such as reviewing an item, managing customers and categories, performing audit and helping the negotiations.

- **Item** : This class represents the item uploaded by the seller for sale and contains the information required. It also has the information about its owner.

- **CustomerRecord** : This is a container class to hold records of buyers and sellers seperately. This class can perform operations such as adding a customer(buyer/seller) to the respective record.

- **ItemRecord** : This is a container class to hold records of items uploaded by the sellers. This class can perform operations such as adding an item to the record, selecting an item asked by the user and providing the list of sellers of a particular item.

- **OSP** : This is a container class to contain the classes CustomerRecord, ItemRecord, manager and the categories of items. This is the main class which contains all the information required for the functioning of portal. It can perform main operations such as registering a customer, verifying the registration details, verifying the login, verifying item details during upload procedure. It can also update the customer record , item record and buyer list of a particular item.

- **Email** : This class is used by the system for the delivering messages among the customers and also the manager. It is used to notify the customer and also for negotiations.

## 3.2 Behavioural Model

The goal of the current sub section is to specify the inter-object behviour and intra-object behaviour.

This can be achieved through the various UML diagrams mentioned below :

- Interaction Diagrams
  - Sequence Diagrams
  - Collaboration Diagrams
- State chart Diagrams
- Activity Diagrams

## Interaction Diagrams :

These diagrams illustrate the communication between the objects. They specify the realisation of an operation as well as a use case. These can be :

### 3.2.1 Sequence Diagrams

A sequence diagram is a "temporally"-oriented diagram which shows the order of the messages graphically with respect to time.

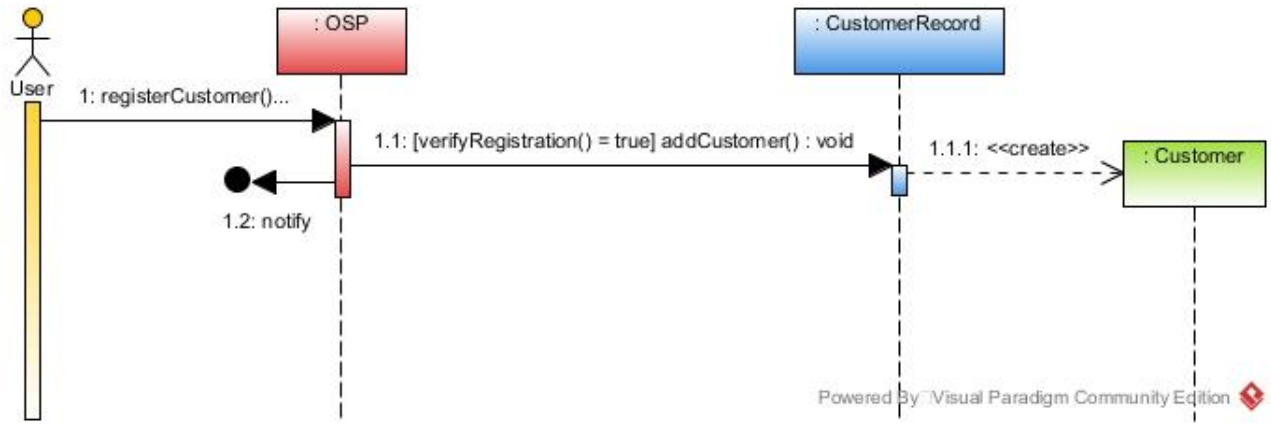


Figure 6: **Sequence Diagram For Registration**

The message sequence between the user, OSP, CustomerRecord and Customer is as follows :

- The user enters his/her details to register.
- The OSP verifies the registration. If the verification is succesful, OSP sends a message to CustomerRecord to add the customer. Customer is registered.
- If the verification fails a notification is sent to the user by the OSP.

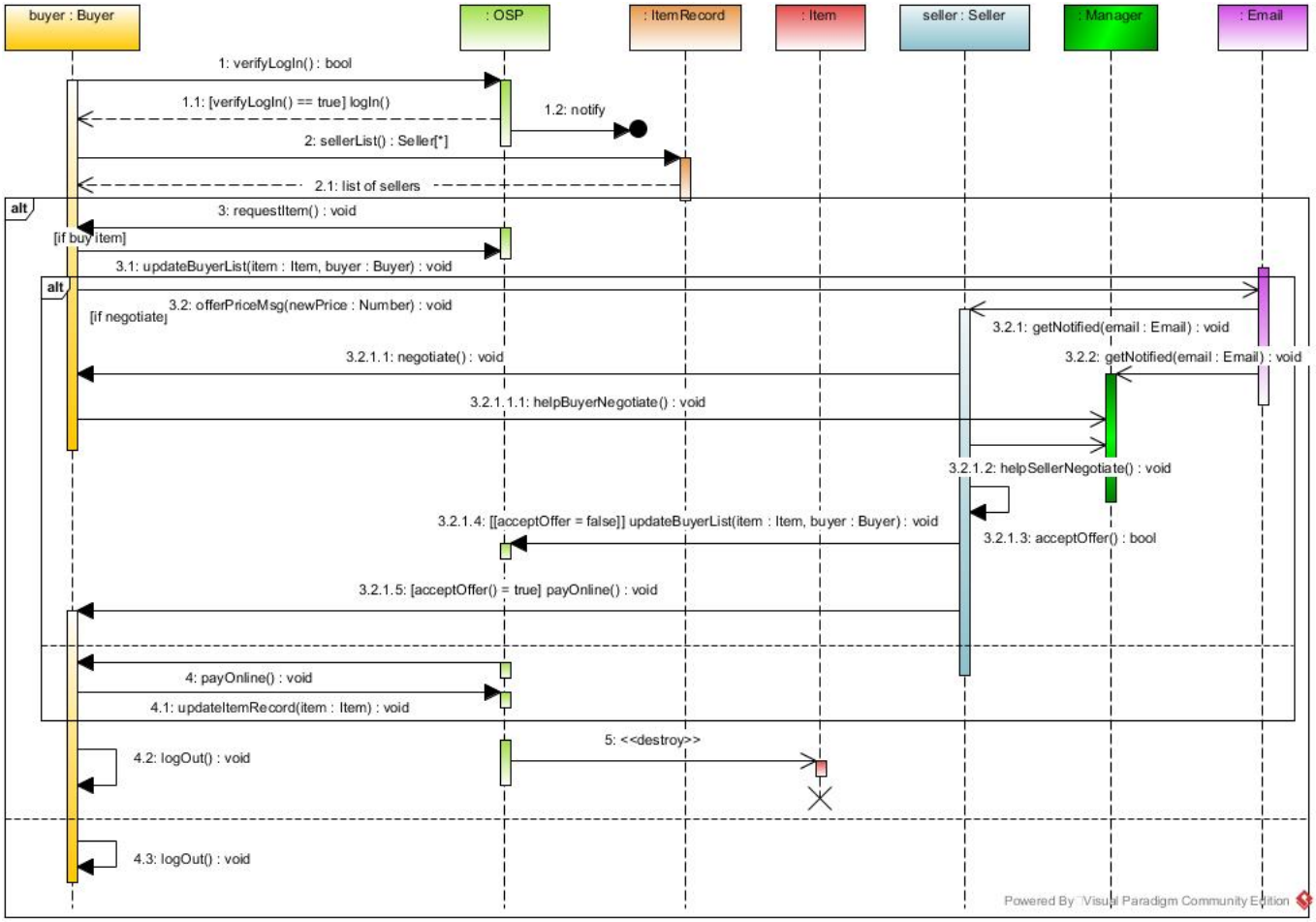


Figure 7: **Sequence Diagram For Buying an Item**

- If verification fails, the OSP sends a notification.
- The buyer sends a message to the Item Record by clicking search button to return the list of sellers for that item.
- The Buyer receives the list of sellers(if any) for that item from the Item Record.
- If the buyer does not choose to buy an item :
  - he/she can simply log out by clicking the sign out button.
- If the buyer chooses to buy an item

- Then raises a request to the selected seller of choice.
- The buyer sends a message to the OSP to update buyer list of that item.
- If the buyer does not choose to negotiate :
  * The OSP sends a message to the buyer to pay the bill online .
  * The user can log out by clicking the Sign out button.
- If the buyer chooses to negotiate :
  * The buyer sends a message to the email to send a message offering a price.
  * The seller receives a message from the email containing the offer.
  * The buyer receives a message from the seller about to negotiate.
  * The manager may also help in the negotiation if needed.
  * After the negotiations are done, if the seller accepts the offer, the buyer receives a message from OSP to pay bill online.
  * If the offer is not accepted, the user can log out of the system by clicking the Sign Out button.

The message sequence between the seller, OSP, ItemRecord and Item is as follows :

- The seller enters the login details and sends a message to OSP to verify it.
- If the verification is successful the OSP notifies the seller that he/she is logged in.
- If verification fails, the OSP sends a notification.
- The OSP sends a message to the seller to upload the details of item.
- After entering the details the seller sends a message to the OSP to verify the details of item provided.
- If the verification is successful the OSP sends a message to the Item Record to create an item.
- An item object is created by the Item Record.

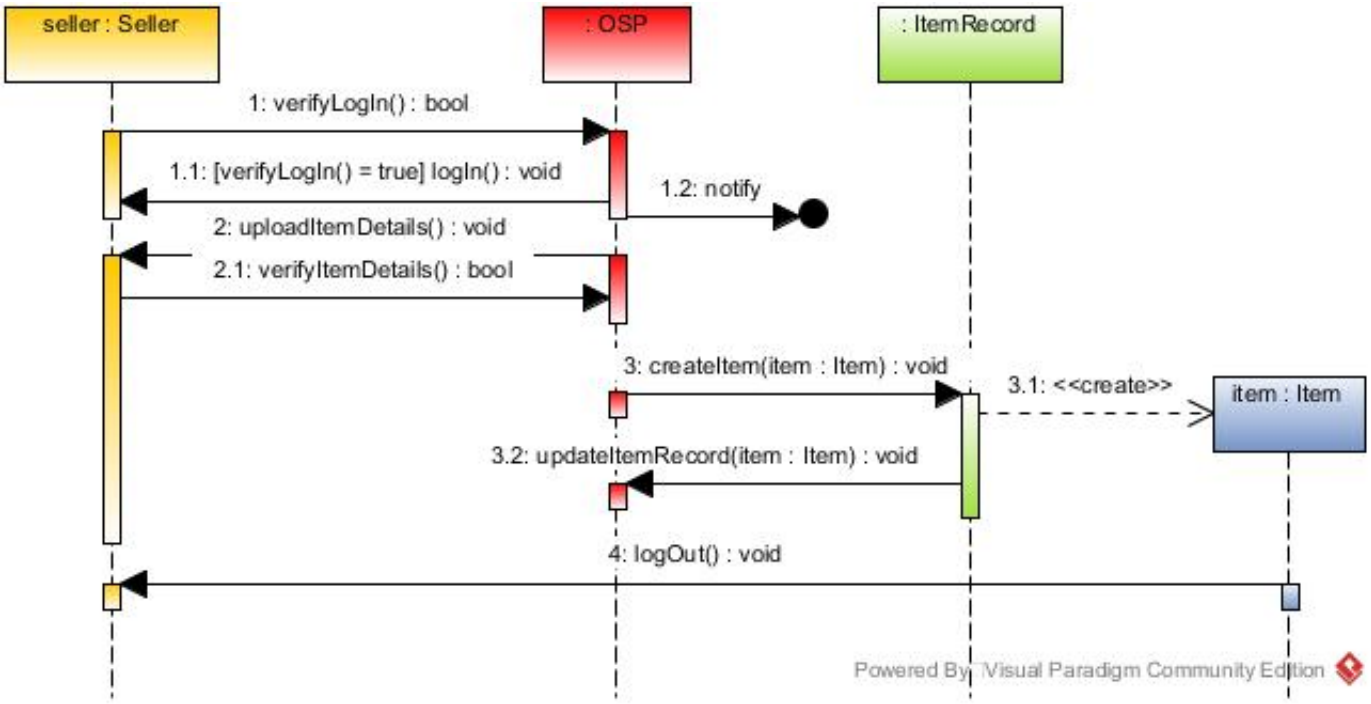


Figure 8: **Sequence Diagram For Selling an Item**

- The Item Record sends a message to the OSP to update the Item Record.
- The user can log out of the portal by clicking the Sign Out button.

The message sequence between the Manager, OSP, Item Record is as follows :

- The manager enters the login details and sends a message to OSP to verify it.
- If the verification is successful the OSP notifies the seller that he/she is logged in.
- If verification fails, the OSP sends a notification.
- The manager sends a message to the Item Record to select an item.
- The Item Record then sends message to the Manager to review that item if it is poor.
- If the manager finds the quality of the item poor he sends a message to the OSP to update the item record by deleting the item as it is of poor quality.

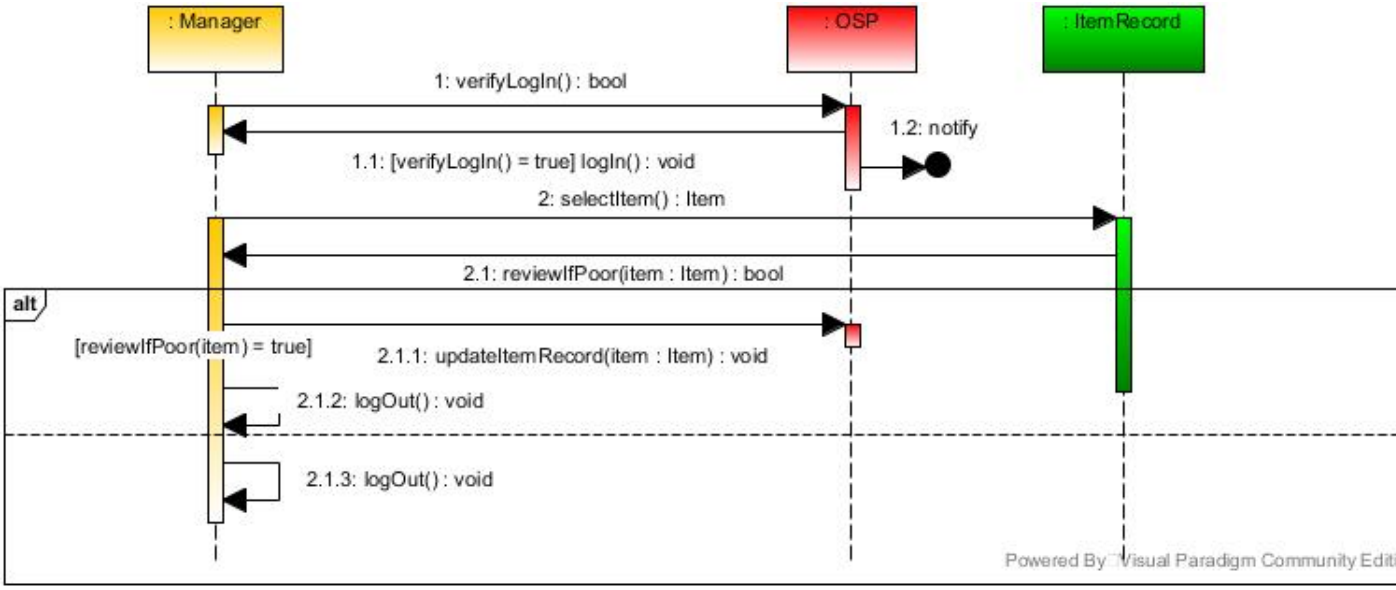


Figure 9: **Sequence Diagram For Reviewing an Item**

- Regardless of the quality of item the manager can log out of the portal by clicking Sign Out Button.

### 3.2.2 Collaboration Diagrams

The below diagrams are the collaboration diagrams for the sequence diagrams shown above for each process respectively. These are 'spatially'-oriented UML diagrams which show the static and dynamic relationships and interactions between various objects. The order of messages is expressed by means of decimal classification only i.e., without any explicit specification to time.
These graphically represent the communications between the objects as described above but no attention is given to the chronological ordering of communications.

The above collaboration diagram is for the Registration process for a user to register as a buyer or a seller to be able to use the portal. It depicts the various communications involved between the objects in the diagram for the process as described above in the sequence diagram section.

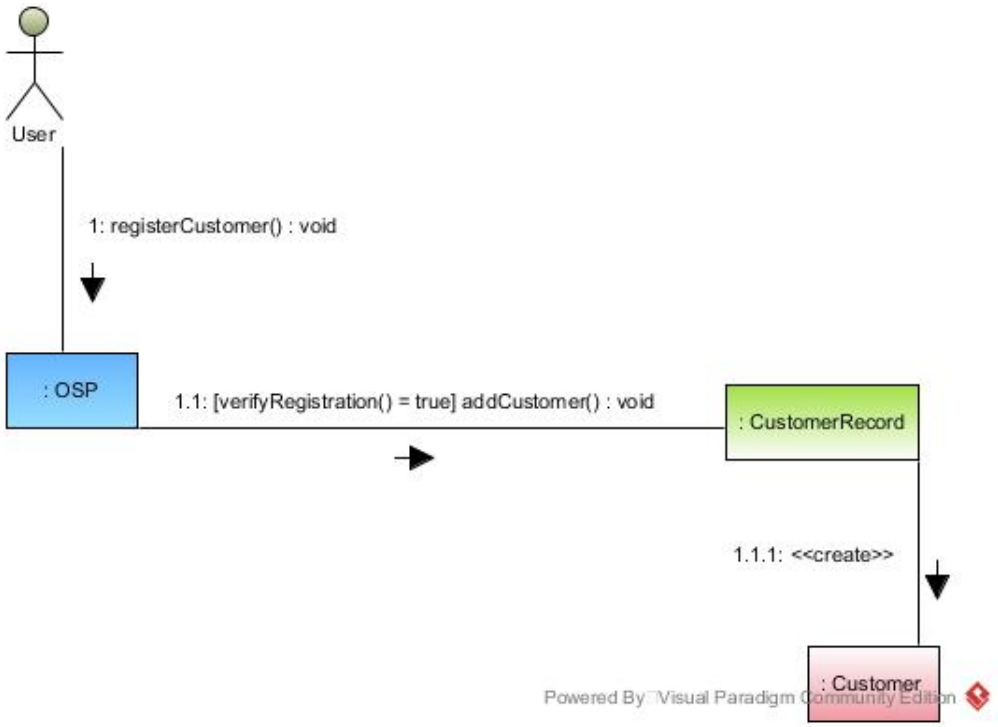


Figure 10: **Collaboration Diagram For Registration**

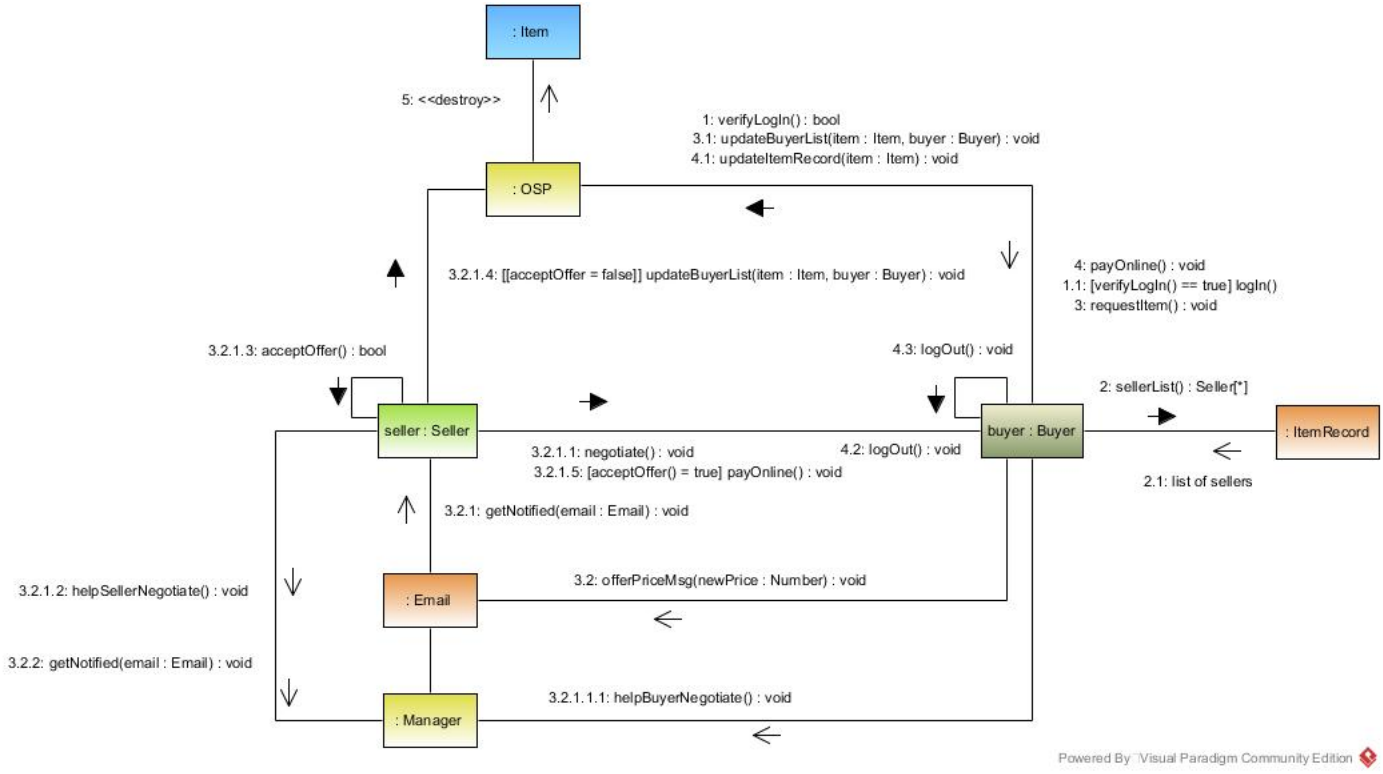


Figure 11: **Collaboration Diagram For Buying an Item**

The above collaboration diagram is for the Buying an item by a buyer registered in the OSP. It depicts the various communications involved between the objects in the diagram for the process as described above in the sequence diagram section.

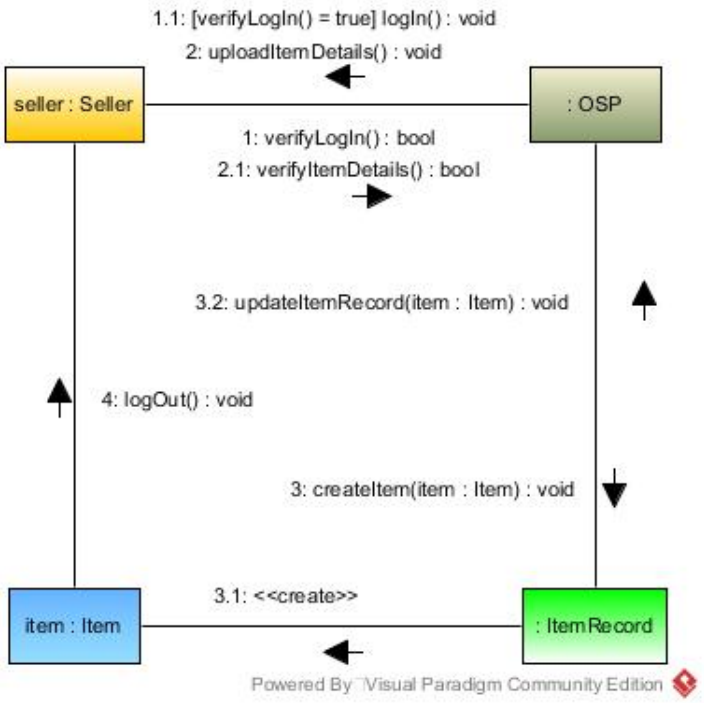


Figure 12: **Collaboration Diagram For Selling an Item**

The above collaboration diagram is for the Selling an item by a seller registered in the OSP. It depicts the various communications involved between the objects in the diagram for the process as described above in the sequence diagram section.

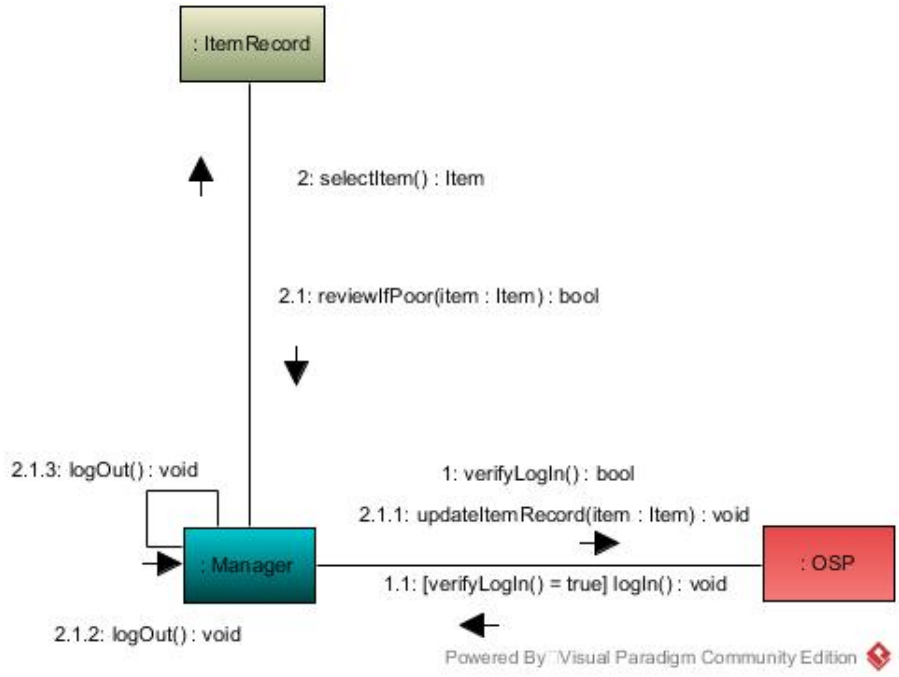


Figure 13: **Collaboration Diagram For Reviewing an Item**

The above collaboration diagram is for the Reviewing an item by the manager which is uploaded by a seller registered in the OSP . It depicts the various communications involved between the objects in the diagram for the process as described above in the sequence diagram section.

### 3.2.3 Statechart Diagram

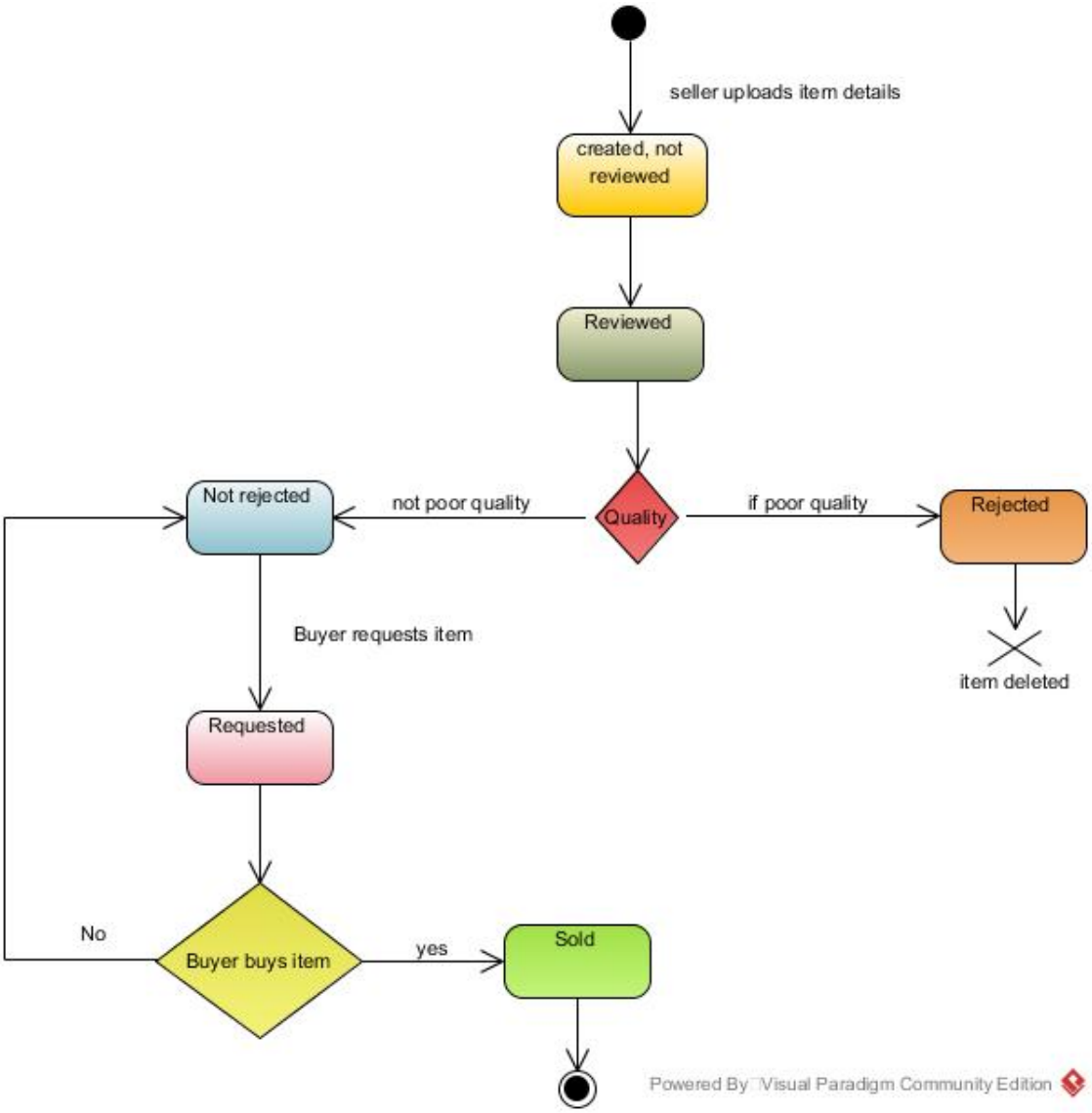


Figure 14: **Statechart Diagram Of An Item**

The statechart diagram is another UML diagram that is used to specify the various state changes of a particular object. It describes the life cycle of the instances of a class as well as the execution of various operations on an instance of a class. It models :

- the possible states of the instances of a class

- the possible transitions from one state to another
- the events firing transitions and the operations(actions and activities) which are executed within states or during a transition.

The above diagram represents the life cycle of an item uploaded by a seller in the OSP.

- Its life cycle begins from the start state when the seller uploads the item details i.e., its status becomes 'created but not yet reviewed'.
- After the manager reviews the item its status becomes 'reviewed'.
- If the manager finds the quality of the item poor, its status becomes 'rejected' thereafter which it is deleted.
- Instead if it is not of poor quality its status is 'not rejected'.
- When a buyer requests for the item its status becomes 'requested'.
- After all the optional negotiations between the buyer and the seller, if the buyer does not buy it its status returns to 'not rejected'.
- If the buyer buys it by paying bill online its status becomes 'sold' thus reaching its final state.

### 3.2.4 Activity Diagrams

Activity Diagram is an important UML diagram that focuses on the behavioural model in the analysis process in the development of the software product. It describes a process consisting of :

- Actions and Activities
- Control Flow
- Input and Output Objects, Object flow.
- All the responsible objects necessary to complete the process.

An Activity can be defined as a meaningful sequence of actions and hence can be further decomposed hierarchically into sub activity states upto the level of an action state.

The following diagrams depict the three important activities of the portal :

- Buying Item
- Selling Item
- Reviewing Item

This activity diagram describes all the actions involved in the process of buying an item. First the buyer has to log into the portal. Then he must search for desired item. If no sellers are found for that item then the activity is terminated. If there are sellers for that item, he/she can choose one among them and raise a request to that seller. The buyer may not negotiate and proceed to online bill payment and terminate the activity. If the buyer chooses to negotiate or offer a price, then the seller can also negotiate with the buyer over email. If the seller does not accept the offer then the buyer can retry with another seller or if not interested , terminates the activity. If the seller accepts the offer then the buyer proceeds to online bill payment and finally the activity is terminated.

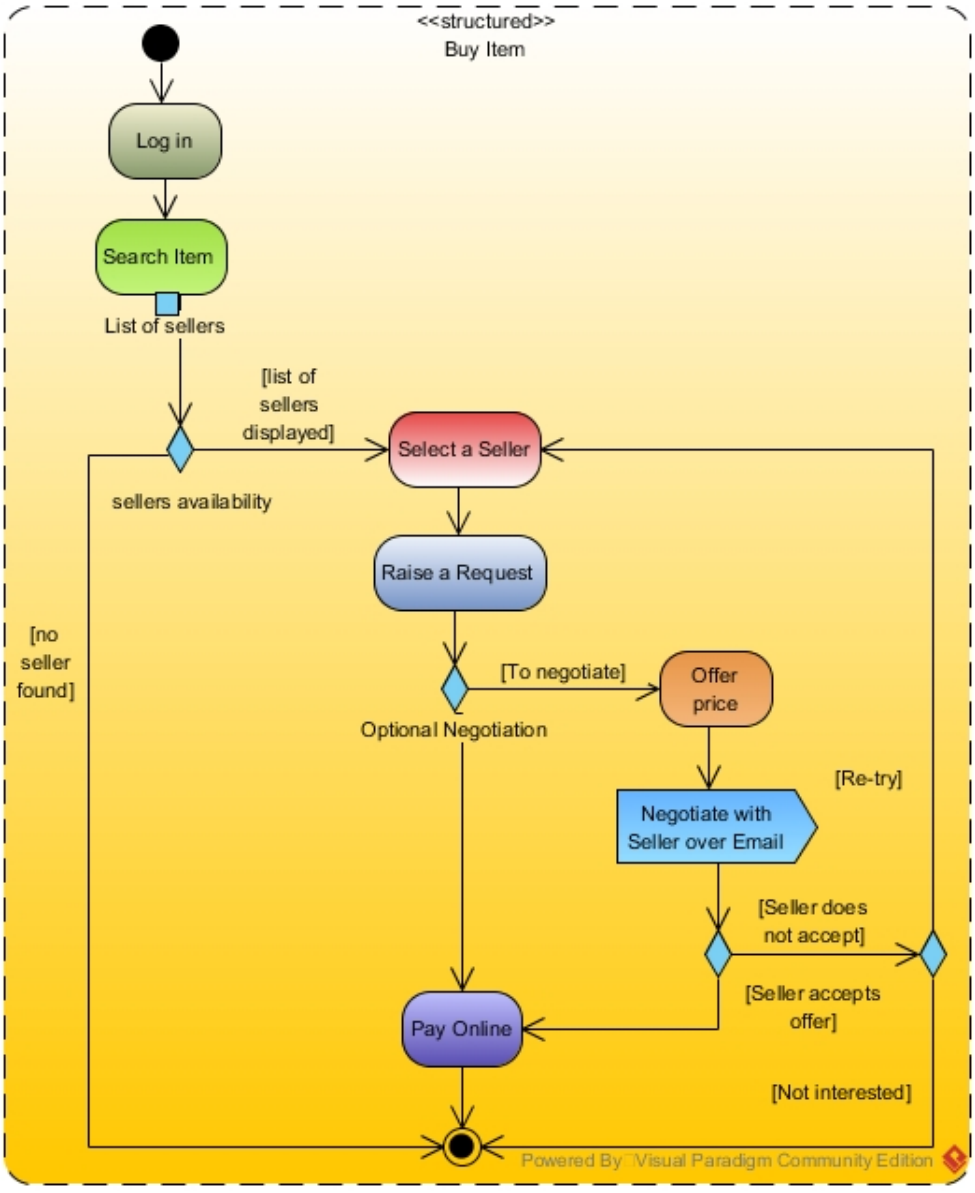


Figure 15: **Activity Diagram For Buying An Item**

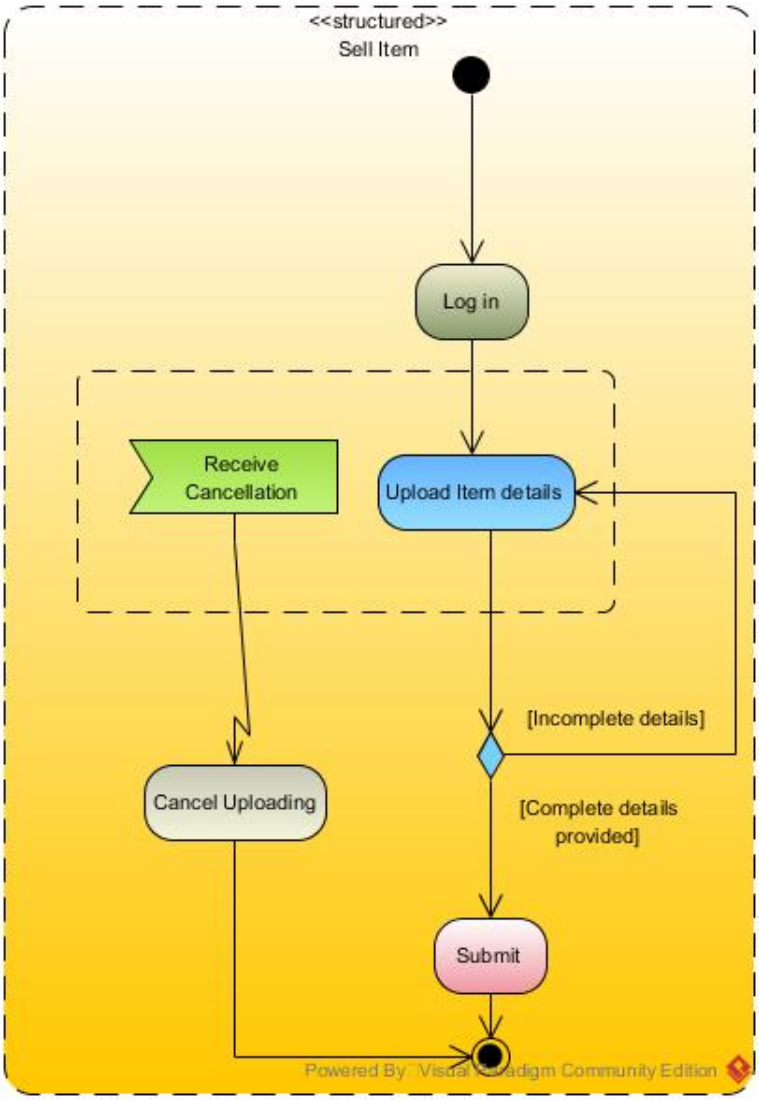


Figure 16: **Activity Diagram For Selling An Item**

This activity diagram describes all the actions involved in the process of uploading an item for sale. First the seller has to log into the portal. The seller has to enter all the details of the item to be uploaded in case of incomplete details entered, the seller has to continue uploading the details. Once the complete details are provided the seller can submit the item to the portal. Hence the activity is terminated. As an exception situation, if while uploading details, if the seller chooses to cancel the upload , the activity is terminated unsuccessfully.

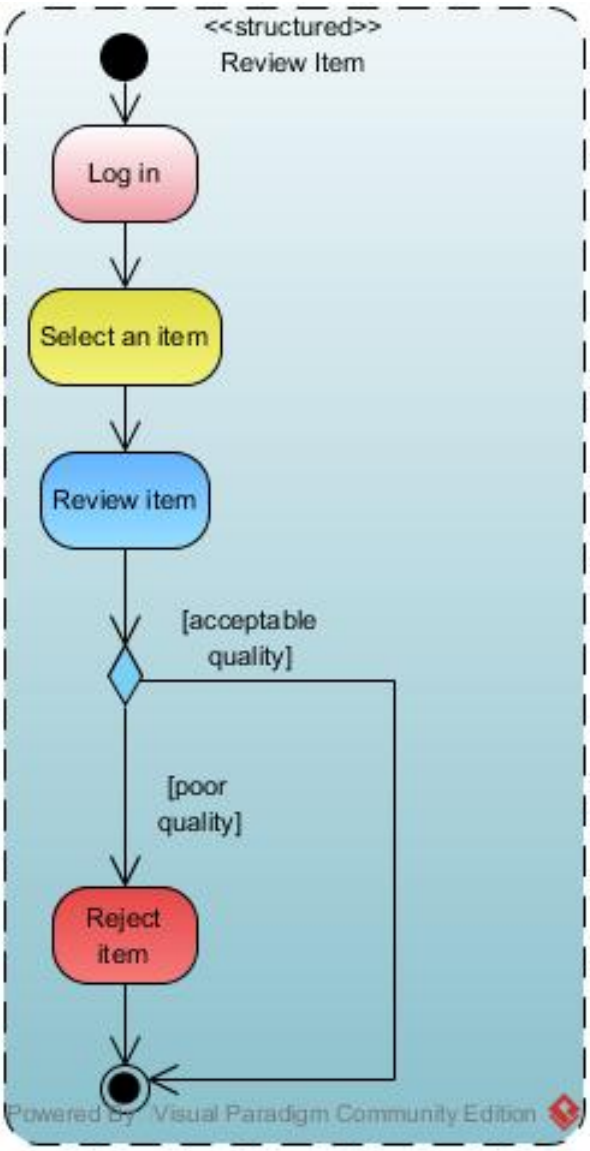


Figure 17: **Activity Diagram For Reviewing An Item**

This activity diagram describes all the actions involved in the process of reviewing an item. First the manager has to log into the portal. The manager must select an item to review it and then if he/she finds the item of poor quality , the manager rejects it and the activity is terminated. If he/she finds the item not of poor quality the activity is terminated without any rejection.

# 4 Detailed Design

## 4.1 Global System Architecture

The overall system architecture is a 2-tier architecture which includes client at one end and the database at the other. There is no server based middle tier in the software being designed.

## 4.2 Platform

### Minimum system requirements :

**Hardware Requirements**:

- Operating System - Windows XP/98 or later versions, Linux
- Processor - Pentium II processor or equivalent
- Hard Disk space - 500MB
- RAM - 512MB

**Software Requirements:**

- Database - MySQL

### Recommended system requirements :

**Hardware Requirements:**

- Operating System - Windows XP/98 or later versions, Linux
- Processor - 400MHz Pentium III processor or above
- Hard Disk space - 500MB
- RAM - 1 GB

## 4.3 Software Architecture

Object-oriented architecture forms the basis of the OSP software. In this style data representations and their associated primitive operations are encapsulated in an abstract data type or object. The components of this style are the objects or instances of the abstract data types. Objects interact through function and procedure invocations. Two important aspects of this style are that

- An object is responsible for preserving the integrity of its representation (usually by maintaining some invariant over it), and
- The representation is hidden from other objects.

Thus the aspects of OOA mentioned justify our choice.

## 4.4 Report

Under the detailed design section of the software design, the global system architecture was discussed. The OSP software has a 2-tier architecture comprising of the client and the database with no server. Then the platform requirements for the OSP software was discussed in terms of the operating system, the processor required, the minimum and recommended hard disk space and RAM requirements, etc. The software architecture of the OSP software was later stated to be of the object-oriented type using PHP as the core technology. The important aspects of OOD used for the OSP are data abstraction and the preservation of integrity of the software.