POLITECNICO DI MILANO

DEPARTMENT OF ELECTRONICS, INFORMATION AND BIOENGINEERING

SOFTWARE ENGINEERING II

Professor Mirandola Raffael

"GuessBid" Project

Requirements Analysis and Specification Document

Authors:

-Nery, Abel Sebsebe (816863)

-Beshir, Addisalem Wondie (816955)

April 27, 2015

Table of contents

1. Introduction

1.1 Description of the given problem

We have prepared this document to implement an inverse auction system to help facilitate the need of selling and buying of goods online.

Our system enable a user to register and login online to able to create, delete and update auctions with information of goods to be sold. The buyer will login to the system and can browse auctions in order to bid by guessing the lowest bid value on the list of products already placed by the seller.

The system will assign a credit of 100 for each register user to enable them bid on goods by withdrawing from their credit. The system will automatically generate the bidding status for users and it will also determine and declare the winner of the bid.

1.2 Goals

"GuessBid" will be able to provide services with the following features:

- Registration of a person to the system
- Logging in to the system
- Creation of auctions
- Placing of goods
- Updating of goods
- Updating of auctions (modifying and deleting)
- Bidding on goods
- Checking the status of bidding
- Receive notification of winner announcement and bid closing

1.3 Domain Properties

The following conditions are under consideration of our system:

- Any user can register to GuessBid Application
- Sellers will insert a list of goods to be sold with the necessary information such as name, price, specification, and pictures.
- Sellers will create auctions for each good by providing the necessary information about the auction such as the auction name and the expiry date.
- Only sellers will be able to update or delete their auction at any point of the bidding process
- Buyers will log in to the system and browse auctions, select the type of good they need and insert the lowest possible bid value.
- Buyers will also be able to check the bidding status and their standings in each bid they are participating.
- Buyers will receive information about any update of the auction they are currently participating in.
- When the system announces the winner automatically based on the auction expiry date the bidders/buyers will be notified and the auction will be closed.

1.4 Glossary

The following words will be used frequently in our document

- **Goods/Products**: are items the can be bought and sold
- **Seller**: is a person who can access the system to create, update and delete auctions for each goods needed to be sold.
- **Buyer**: is a person who can access the system to browse auctions, see all information about the list of goods placed by the seller, select products and bid on them by selecting the minimum bidding value.

- **Guest:** a guest is a person who can access the home page of the system but not yet registered to the system and cannot be able to sign in. A guest can sign up to the system to become a seller or buyer.
- **User**: when we say user we mean the buyer or seller or both
- **Inverse Auction**: is a process where goods are sold to the lowest bidder.
- **Bid**: a bid is a price offer that will be given by the buyers at a given auction.
- **Bidding status**: is the current position of buyers with respect to each other based on their price offer. The highest position will be taken by the buyer with the lowest bid value.
- **Notification**: is a message about auction modification, buyer standings or position changes of the bidding status. It also could be the winner announcement and the auction closing message.

1.5 Assumptions

Here we have some points which may create ambiguity or which are not a part of this document but still worth mentioning:

- Here the main assumption we have made is about the inverse auction system. Since the aim of auction is to sell good to the highest bidder, in the inverse auction we assume that seller have a logical reason to sell their good to the lowest bidder. Inverse auction works perfectly especially for services; for example if a person wants to hire a consultant, he/she may compare select the applicants based on the lowest price offered. In conclusion, we assume that the service provision reality mentioned above works for items to be sold.
- This system doesn't include product shipping and delivery. We will focus on the goods to be sold and the bidding process until the closing

of the auction. After that, we assume that the products will be shipped to the buyer properly.

1.6 Proposed system

We propose a GuessBid Web Application to help users involve in an inverse auction systems to sell and buy goods online. This web based application will provide the following capabilities:

Users will easily access this web application, to join the system, to create auctions, to browse auctions, to bid on goods available on existing auctions, to check bidding status, and to update and delete their own auctions.

Sellers will have a specific interface with the ability to list all their goods to be sold, to create auctions for each good and manage their own auctions until its closing.

The system will generate all the list of goods and their associated auctions. It will also generate the status of each buyer during bidding process, at the end; the system will announce the winner and generate notification messages.

After the winner is announced by the system the seller will confirms that and sells the product. The sell has a right to cancel the auction anytime even after the winner is announced.

After the buyer is received the winner notification, if he/she is the winner, they will confirm to pay and buy the product from their credit account.

1.7 Identifying Stakeholders

For this project we don't exactly have a financial stakeholder since we are developing this application for learning purposes to meet the requirements set by our Professor. Any user who intends to use our system is our customer or client and considered as one of our stakeholders.

1.8 Other considerations about the system

We have planned to make our system better having the following considerations:

- **Easy to us**e: simplicity to use our application is one of the features that we have found important. We will try our best to create well-organized reports, we will also implement a simple way of navigation through our system with a full set of functionalities in such a way that the user will see a list of users and do some actions to bid and manage auctions.

- **Stability:** We will grant some basic functions in a stable way. For example there has not to be any fault in registering a new user.

- **Have a nice look and feel:** we will try to make a well-designed application in the sense that we hope it will be nice to see and will fit to any user.

2. Actors Identifying

The actors of our informative system are basically four:

- **Guest:** a person who has not registered and can only exploit basic functionalities such as accessing the home page and signup.
- **Buyer:** a person that has registered and so has provided his personal information to the system.

- **Seller:** is a user who is responsible for the management of specific auctions for each good that are listed by him.

3. Requirements

i) Registration of user to the system:

The system has to provide a sign up functionality.

ii. Creating auction

The system will allow sellers to create auction for each good

iii. Modify auction information

The system will allow the organizer of the system to modify goods and associated auction information;

iv. Search for goods

The system will allow the user to search for available goods in the system to precede bidding process

v. Participate in the bidding

The system will allow the buyer to see available goods for auction and participate by putting the lowest bid value.

vi. Check bidding status

The system will allow users to see the status of the bid process for example the standing of each buyer at each bid.

vii. Receive notification message

The system will allow users to receive a message about the winner of each auction and the auction closing.

viii. Send message

The system will allow sellers and buyers to communicate. The buyer can send message to seller to request more explanation about the product and the seller can reply.

3.1 Functional Requirements

Functional requirement describes the functionality of the proposed system. It describes the function of the system irrespective of the implementation

Guest: he can only exploit basic functionalities, so he can only:

- Access the home page of the system
- view goods
- search goods
- Register/Sign up.

Seller: he can:

- Log in
- List goods to be sold
- Create auctions
- Modify auction information
- Delete auction
- Check bidding status of his/her auction
- Receive auction notification
- Confirm winner and sell
- Receive and send messages

Buyer: he can:

- Login with user name and password
- Search goods
- Browse auctions
- Participate in the bidding
- Checking bidding status
- Receive auction notification
- Buy goods/pay

3.2 Non Functional Requirements

In addition to the above functionality the following non-functional requirements are identified from user perspective and are expected requirements of the system

Workability

The system should be Suitable for the variety of users. It should be accurate in performing its functions and secured enough from intruders and attacks by external users. Moreover, it must support interoperability with other subsystems and components, and it should be complete i.e. it should be fully functional in terms of providing all the functions expected it to perform.

Reliability

The system should be reliable and matured enough in giving its service. It should have a fault tolerance mechanism in which it can recover faster from problems that may occur. The system should support backup in case the original was accidentally damaged or erased.

Efficiency

The system should be efficient and the response time should be minimal. It should be capable of running on minimum hardware requirements and with the accustomed operating systems.

Portability

The system needs to be portable on all major plat forms. This system should not be restricted by any specific technology such as database, web server, and operating system. There should always be alternative environment.

Concurrency

The system should support multiple accesses of users. It should give service to multiple users concurrently.

Error Handling

The system should handle exceptions and extreme conditions and behave accordingly. It should notify the users about the type and location of exception and appropriate action to be taken.

User Interface

The interface of the system should be user friendly, that is, it should be understandable.

Documentation

There will be a user documentation that could be found on the system which shows all the functionalities of the system clearly. It will show the steps to follow to perform each task.

Security

All input need to be encoded and validated to prevent SQL injection. User can only access the system if it is already registered in the system. Some user groups can be configured that they can never have certain permission. For example, buyer should not have the access to update and delete auctions.

.

4. Specifications

- The Seller might decide to modify auction information for example in the case of goods quantity and price change.

- The buyer can see the standings of each buyer who are participating in the same auction but the buyer cannot see the offered price of other participants.
- Only sellers can create a list of goods and their associated auctions.
- Sellers can search goods to create auctions
- Buyers can search the list of available goods for auctions or can simply browse existing auctions
- Only registered user can be a buyer or seller
- Buyers can participate in an auction by guessing the lowest bid
- Both buyers and sellers can see the list of auctions and its association expiry date
- Both buyers and sellers can receive the notification message about the winner of the bid and the auction closing.

5. Identifying scenarios

No	Scenario Name	Description
1	Enter Good	A seller that has already registered to the system can log in using his/her user name and password and by selecting Enter Goods and when Goods form is displayed the seller will insert all new good information then save the data. After saving the data the seller logout form the system.
2	Update Good	The seller who has already created goods information can login in to the system using

		his/her username and password and by selecting goods and then by selecting the update goods the seller can modify the specific good information for each product.
3	Delete Good	The seller who has already created goods can login in to the system using his/her username and password and by selecting goods the seller can easily delete each product or group of products easily.
4	Create Auction	A seller that has already registered to the system can log in using his/her user name and password and by selecting create new auction and when auction information form render the seller will insert all new auction information and after saving the data the seller logout form the system
5	Update Auction	The seller who has already created auction information can login in to the system using his/her username and password and by selecting auction and then by selecting the update auction the seller can modify the information about that auction or the goods associated to that auction.
6	Delete Auction	The seller who has already created auction can login in to the system using his/her username and password and by selecting auction and then by selecting the update auction the seller can Delete the auction information

7	Receive Auction Notification	A user(Seller of Buyer) that is already logged in to the system can automatically receive auction winner notification and can confirm the buying or selling process by selecting confirm sale and confirm buy. This will lead to payment process and auction closing.

6. System Modeling

System models describe the model of the envisioned system in a dynamic, static and analytic aspect. Use case documentation and sequence diagram is used to show the dynamic model of the system and Class diagram is utilized to show the static aspect of the system.

6.1 Use Case Diagram

We can derive some use cases from the scenarios identified in the previous paragraph:

We decided to split the Use Case Diagram into smaller ones because we wanted to make the situation clear. We can provide some "macro Use Cases" below (this is not a Use Case Diagram, but only a diagram that helps the reader to understand the composition of the diagrams drawn below:

Use case description

We describe in a detailed way below the main use cases under each diagram. Some use cases that extends other use cases and some others are omitted because they are really similar to others.

6.1.1 Main Use Case Diagram

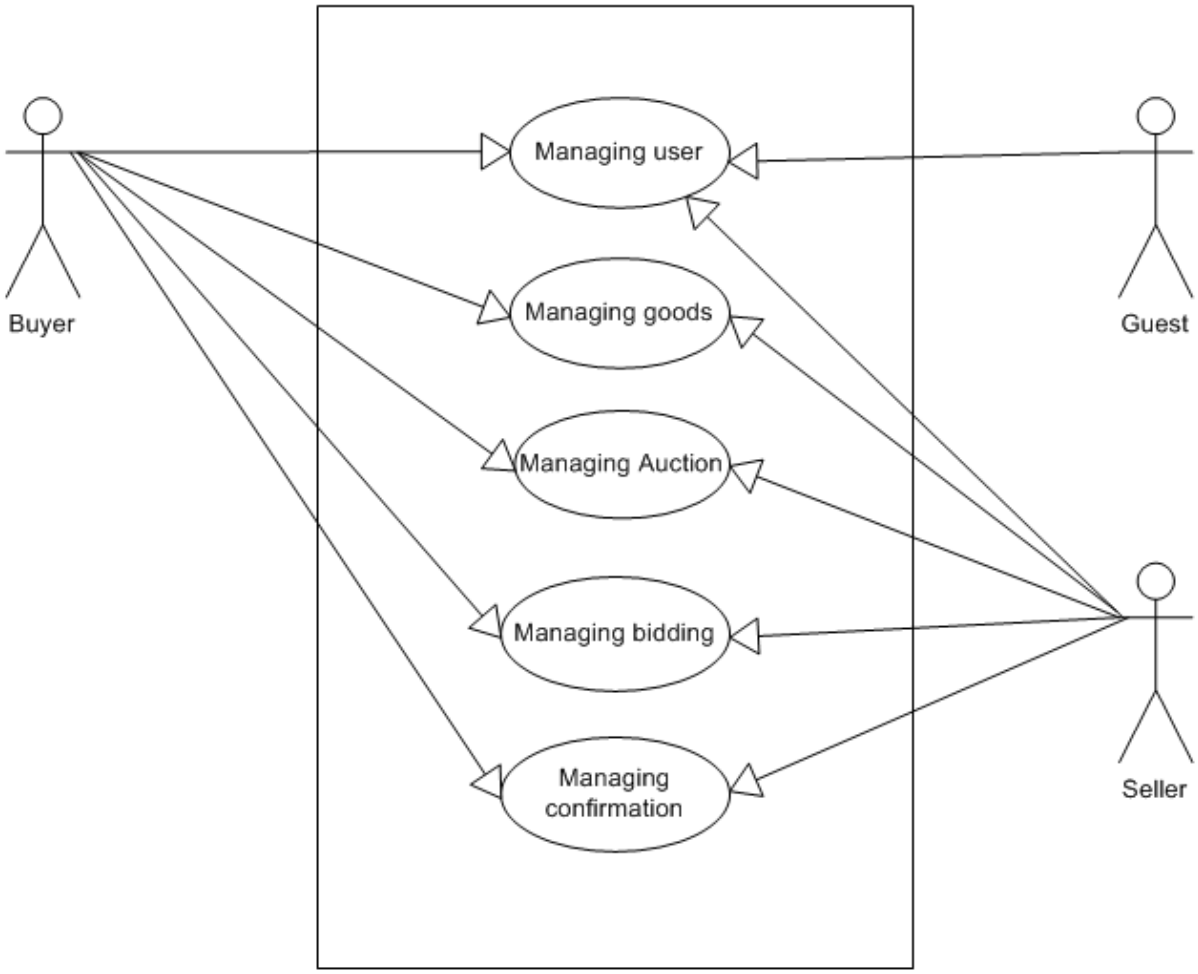


Figure 1. Main Use case Diagram

6.1.2 Managing Goods Use Case Diagram

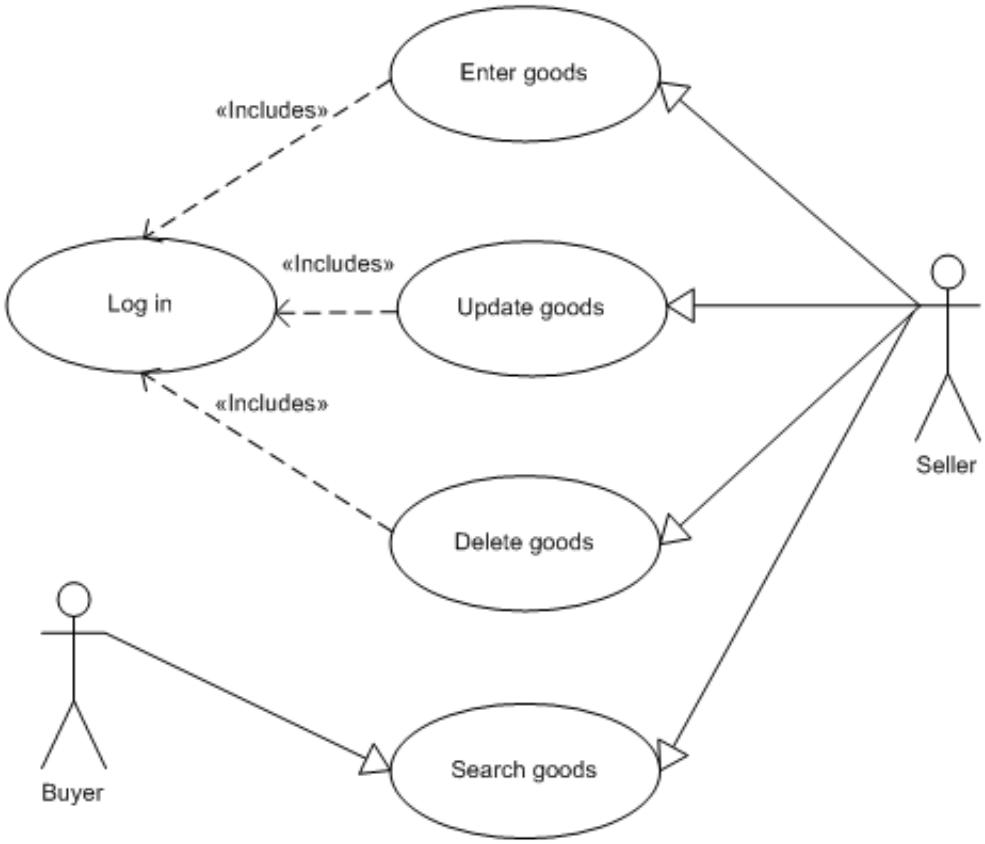


Figure 2. Managing Goods Use case

Managing goods Use Case Discription

Use case ID	MGUC-01
Name	Enter goods
Actor	Seller
Precondition	The user have been authenticated
Normal flow	1. The user select Enter goods sub menu 2. The system display a form to create

	goods information 3. The user fill Goods ID, Type, Quantity, Unit price, Discription, and insert picture of the product and press **save** button 4. The system validates and saves it if it is valid or else Displays error message
Post condition	Goods information will be stored in the system to be used whenever required
Exceptions	• The seller failed to login • The seller has not entered some mandatory fields. In this case an error message is shown.

Use case ID	MGUC -02
Name	Update goods
Actor	Seller
Precondition	The user have been authenticated
Normal flow	1. The user select **Update goods** sub menu 2. The system displays a list of goods 3. The seller selects the product he/she wants to update 4. The system will populate goods information form with selected data 5. The seller modify product information except Goods ID and click **update** button 6. The system validates and update it if it is

	valid or else Displays error message
Post condition	Product information is updated
Exceptions	• The seller failed to login • The seller has inserted invalid data. In this case an error message is shown.

Use case ID	MGUC -03
Name	Delete Goods
Actor	Seller
Precondition	The seller have been authenticated
Normal flow	1. The user select **Delete goods** sub menu 2. The system display a list of goods 3. The user select good/product 4. The seller press **delete** button and the system confirm the user with [yes/no] option 5. The user select yes option 6. The system delete good/product
Post condition	Product information is deleted from the system
Exceptions	• The seller failed to login • The seller has selected no option. The system abort delete operation

Use case ID	MGUC -04

Name	Search goods
Actor	Seller, Buyer
Precondition	The user has to view the goods search page
Normal flow	1. The user type name/ID of the product on search box select search button 2. The system displays search result
Post condition	No exit condition
Exceptions	The search gives no results.

6.1.3 User Management Use Case Diagram

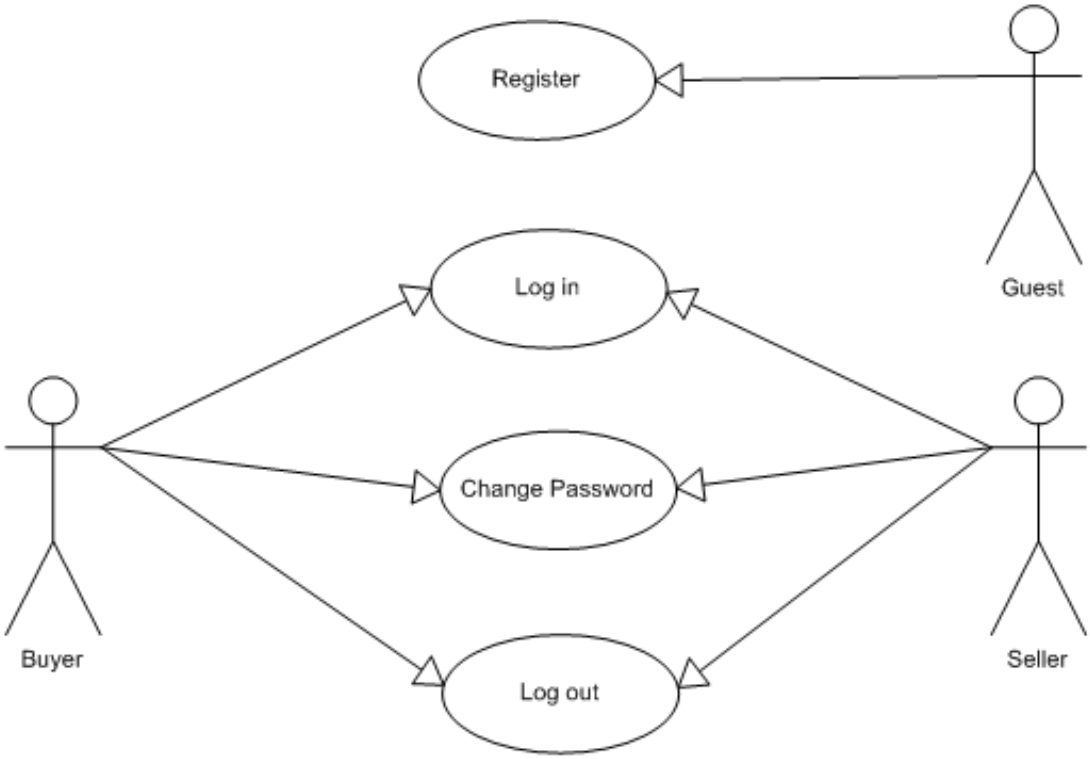


Figure 3. User Management Use case

User Management Use Case Description

Use case ID	UMUC-01
Name	Signup/Register
Actor	Guest
Precondition	The guest has not signed up to the system
Normal flow	1. The user clicks on the button "sign up"; 2. The system shows a page which contains an input form. The input form asks for some personal information and an e-mail address;

	3. The guest fulfills the input form and clicks "sign up"; 4. The link redirects him to GuessBid, the system confirms the registration.
Post condition	The information about the guest is stored into the database in the way to grant his log in. The log in is now possible
Exceptions	The guest enters a non valid password, or doesn't fill some mandatory fields. In this case the page is reloaded showing an error message.

Use case ID	UMUC-02
Name	Login
Actor	Seller, Buyer
Precondition	User successfully signup to the system
Normal flow	1. The user opens the home page of the platform; 2. The system shows the page; 3. The user enters his e-mail address and password in the input form provided; 4. The user clicks the button "log in". 5. The system shows the user personal page.
Post condition	There is no exit condition
Exceptions	The information inserted in the form is wrong, an error message is shown.

6.1.4 Managing Auctions Use Case Diagram

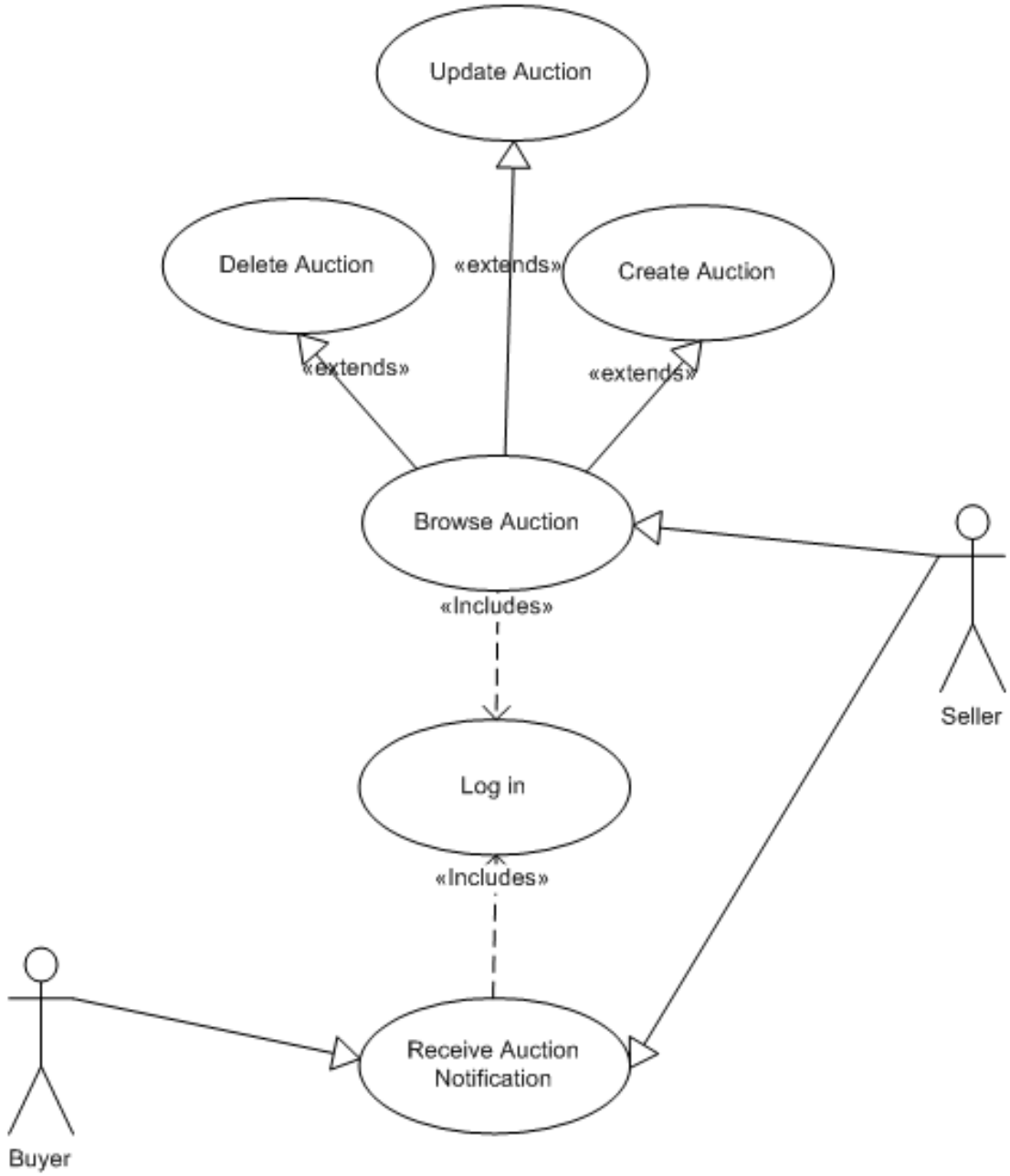


Figure 4. Managing Auctions Use case

Managing Auctions Use Case Description

Use case ID	MAUC-01
Name	Create Auction
Actor	Seller
Precondition	The user have been authenticated, Good/ Product is selected
Normal flow	1. The user select Create Auction menu 2. The system display a form to create auction 3. The user fill Auction name, Auction ID, Expiry date, and initial price, and press **save** button 4. The system validates and saves it if it is valid or else Displays error message
Post condition	Auction information will be stored in the system to be used whenever required
Exceptions	• The seller failed to login • The seller has not entered some mandatory fields. In this case an error message is shown.

Use case ID	MAUC-02
Name	Update Auction
Actor	Seller
Precondition	The user have been authenticated, Auction exist

Normal flow	1. The user select **Update Auction** sub menu 2. The system displays a list of auctions 3. The seller selects the auction he/she wants to update 4. The system will populate auction information form with selected data 5. The seller modify auction information except Auction ID and click **update** button 6. The system validates and update it if it is valid or else Displays error message
Post condition	Auction information is updated
Exceptions	• The seller failed to login • The seller has inserted invalid data. In this case an error message is shown.

Use case ID	MAUC-03
Name	Delete Auction
Actor	Seller
Precondition	The seller have been authenticated
Normal flow	1. The user select **Delete Auction** sub menu 2. The system display a list of auctions 3. The user selects auction 4. The seller press **delete** button and the

	system confirm the user with [yes/no] option 5. The user select yes option 6. The system deletes auction
Post condition	Auction information is deleted from the system
Exceptions	• The seller failed to login • The seller has selected no option. The system abort delete operation

Use case ID	MAUC-04
Name	Search/Browse Auction
Actor	Seller, Buyer
Precondition	The user has to view the search auction page
Normal flow	1. The user type name/ID of the auction on search box and selects search button 2. The system displays search result
Post condition	No exit condition
Exceptions	The search gives no results.

6.1.5 Managing Bidding Use Case Diagram

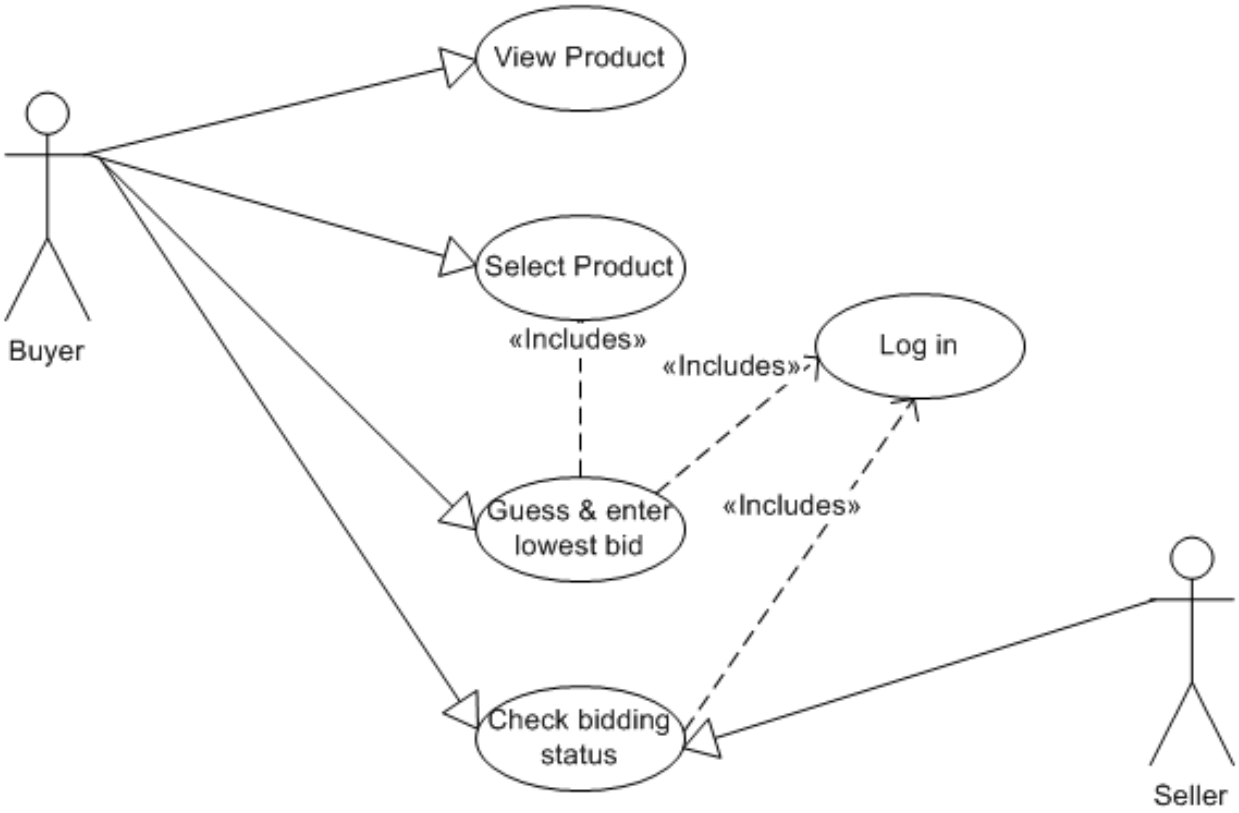


Figure 5. Managing Bidding Use case

Managing Bidding Use Case Description

Use case ID	MBUC-01
Name	Guess and enter lowest bid
Actor	Buyer
Precondition	The buyer has to select product
Normal flow	1. The user type the lowest bid guess value and click bid 2. The system saves the bid information

Post condition	The user check the bidding status
Exceptions	The product auction date has expired

6.1.6 Managing Confirmation Use Case Diagram

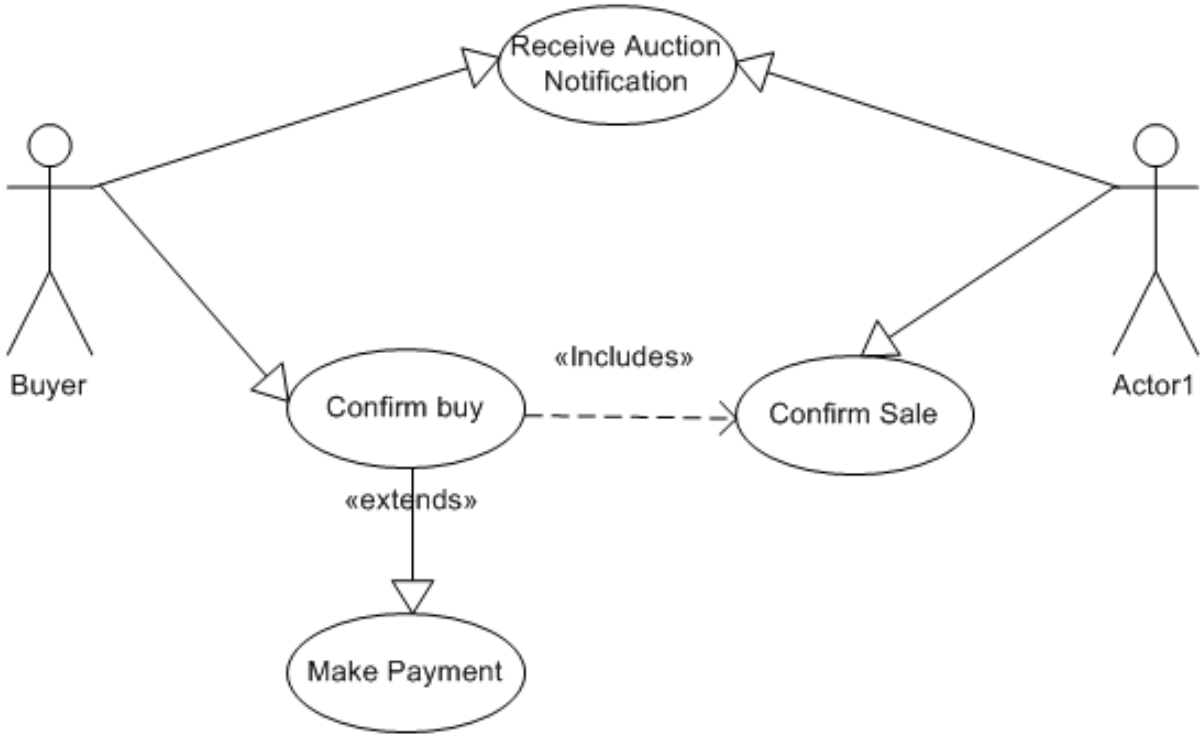


Figure 6. Managing Confirmation Use case

6.2 Class Diagram

We have depicted our class diagram as follows:

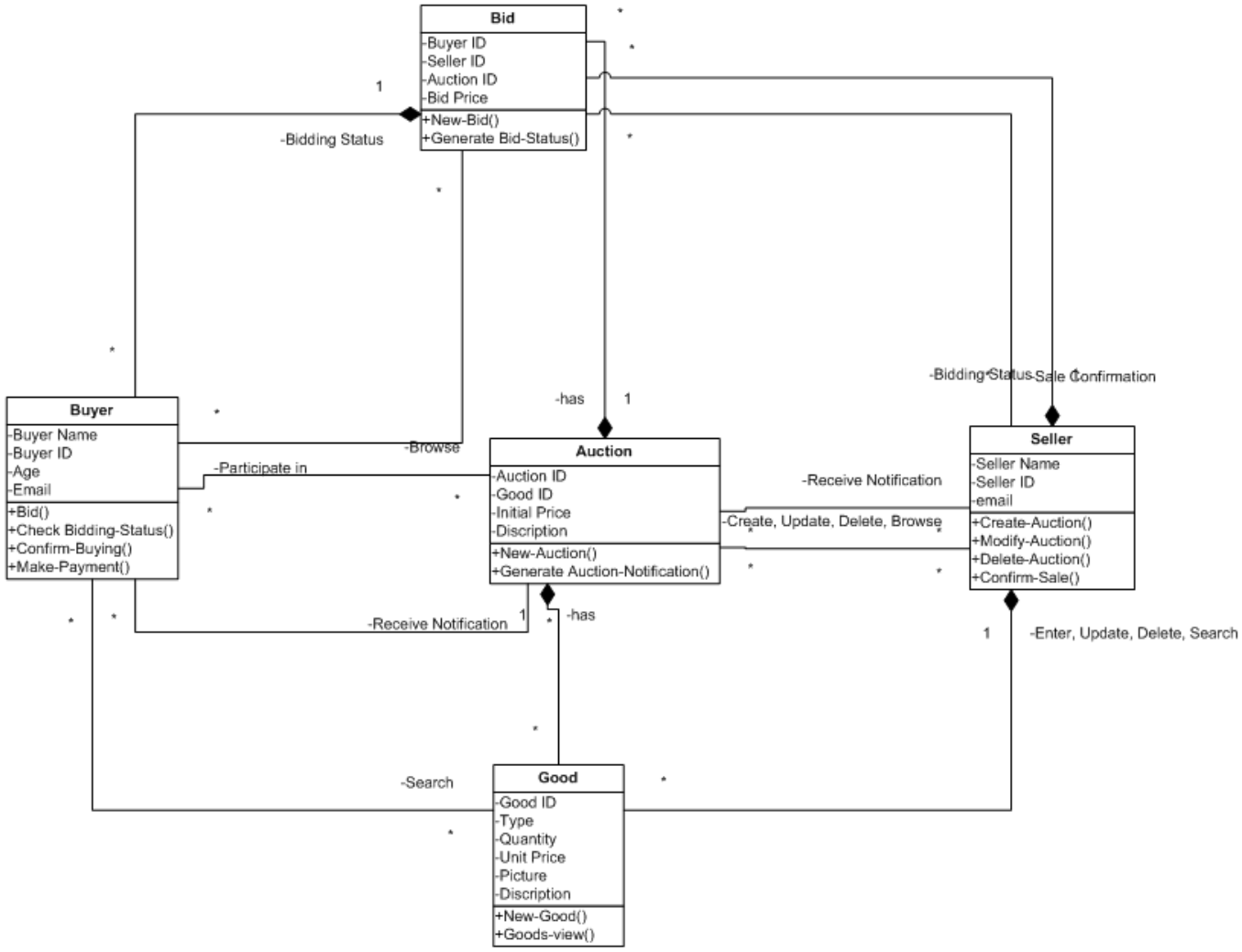


Figure 6. Class Diagram

6.3 Sequence Diagrams

6.3.1 Log in

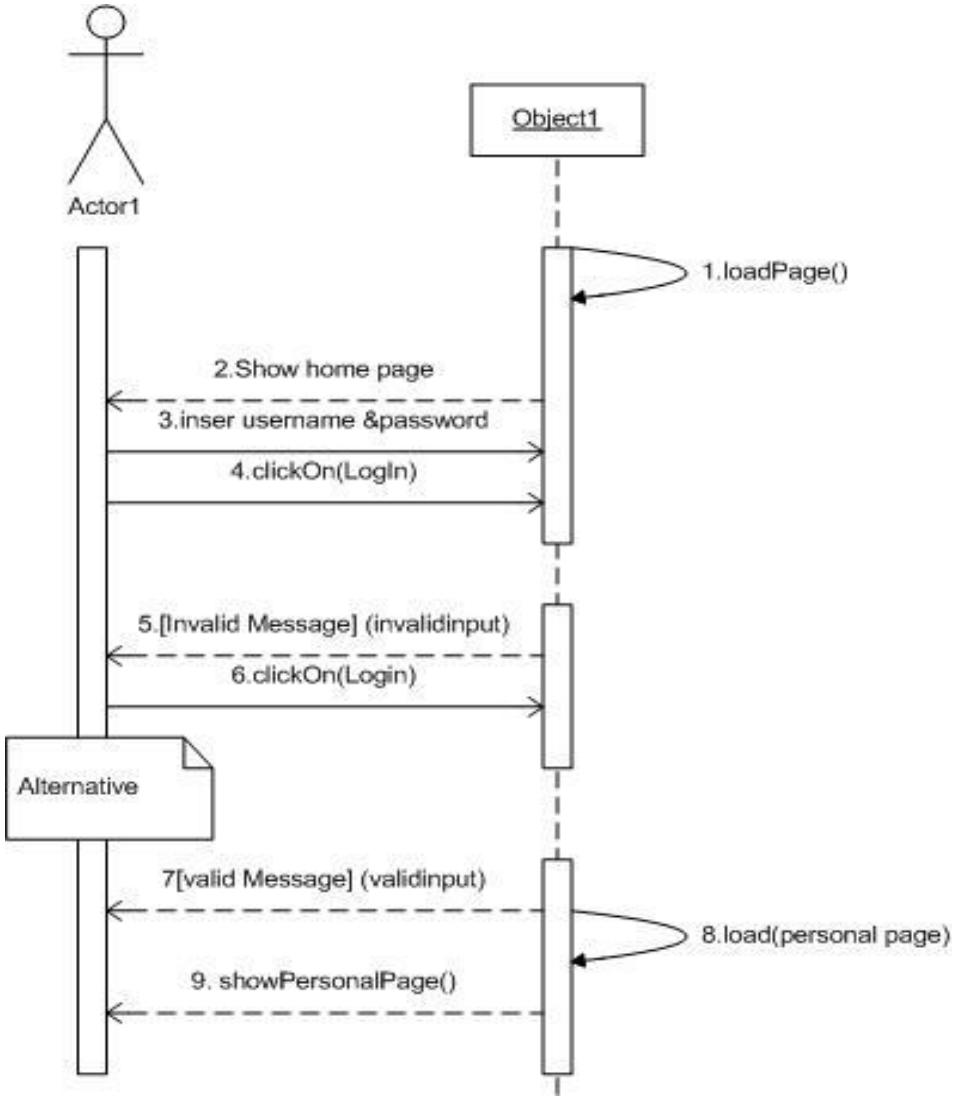


Figure 7. Log in Sequence Diagram

6.3.2 Enter Good

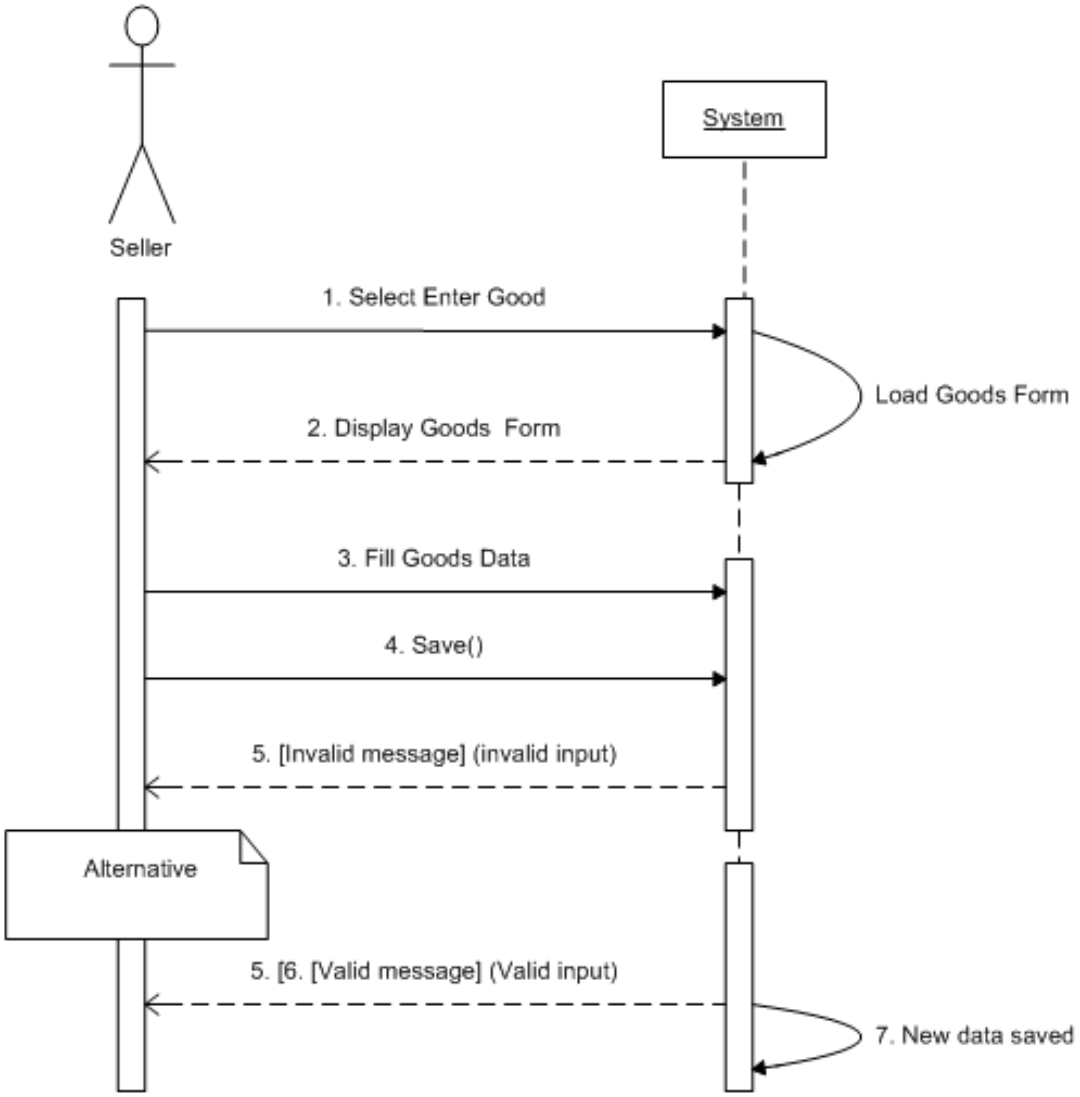


Figure 8. Enter Good Sequence Diagram

6.3.3 Update Good

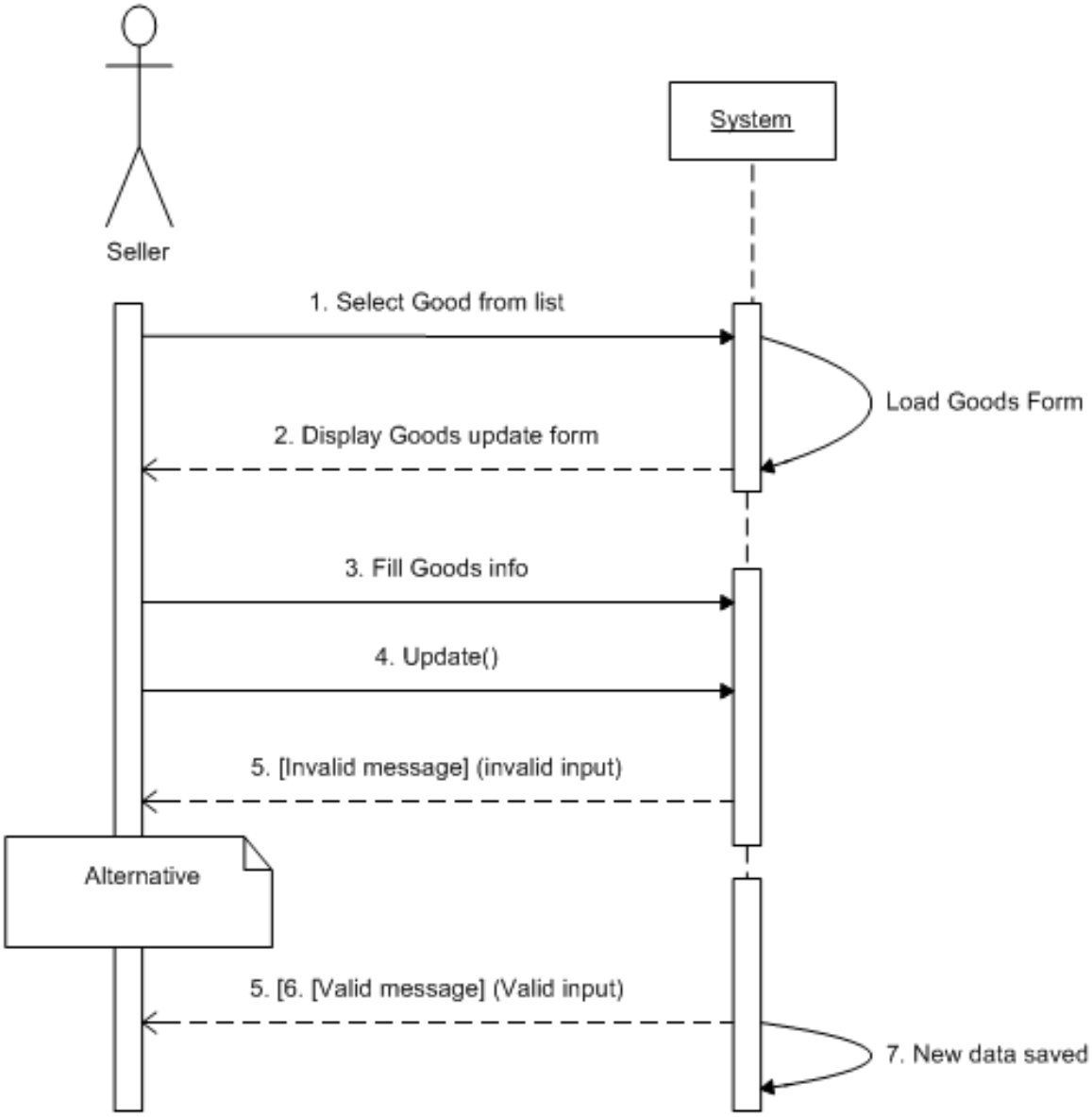


Figure 9. Update Good Sequence Diagram

6.3.4 Search Good

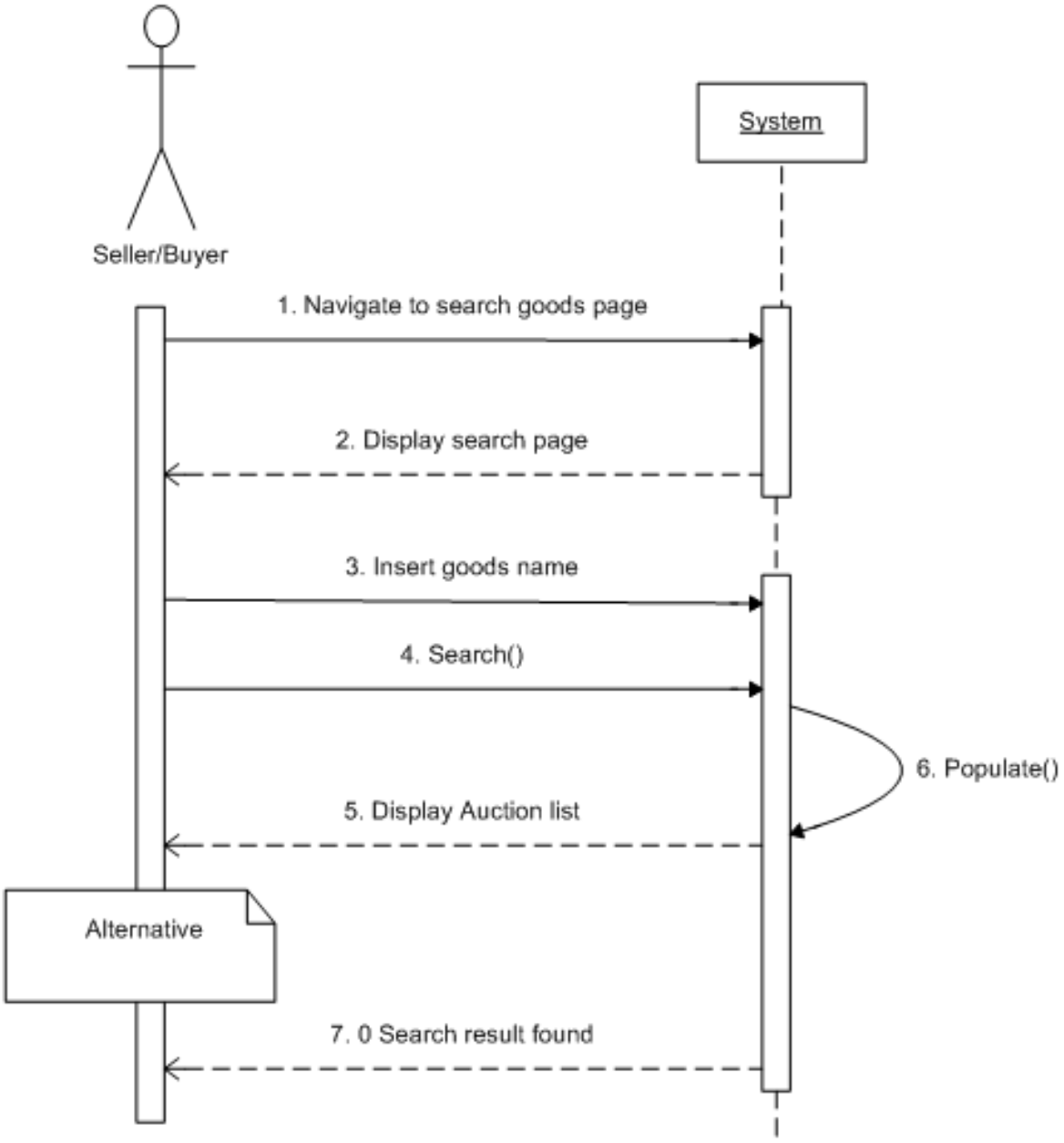


Figure 10. Search Good Sequence Diagram

6.3.5 Create Auction

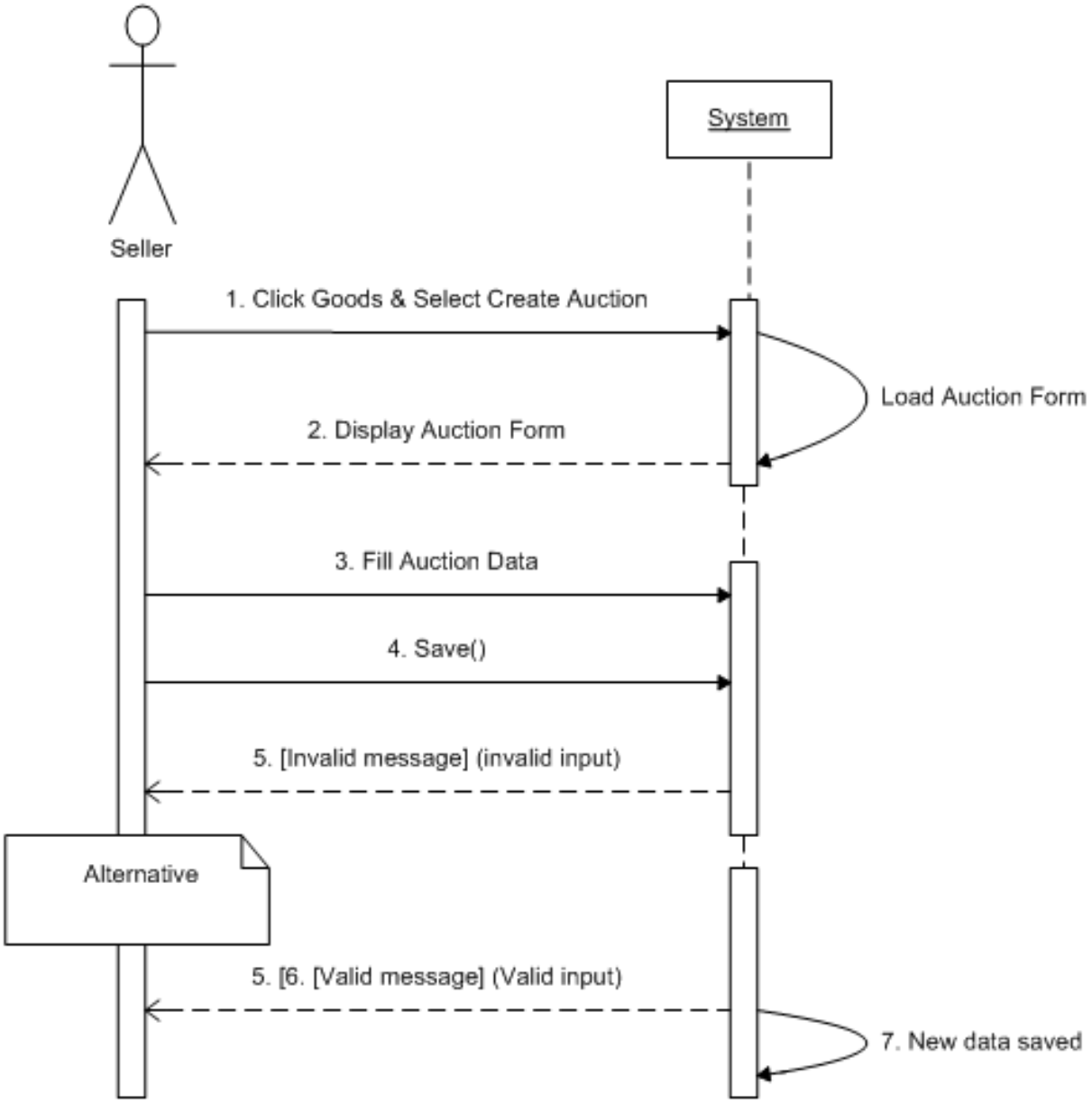


Figure 11. Create Auction Sequence Diagram

6.3.6 Update Auction

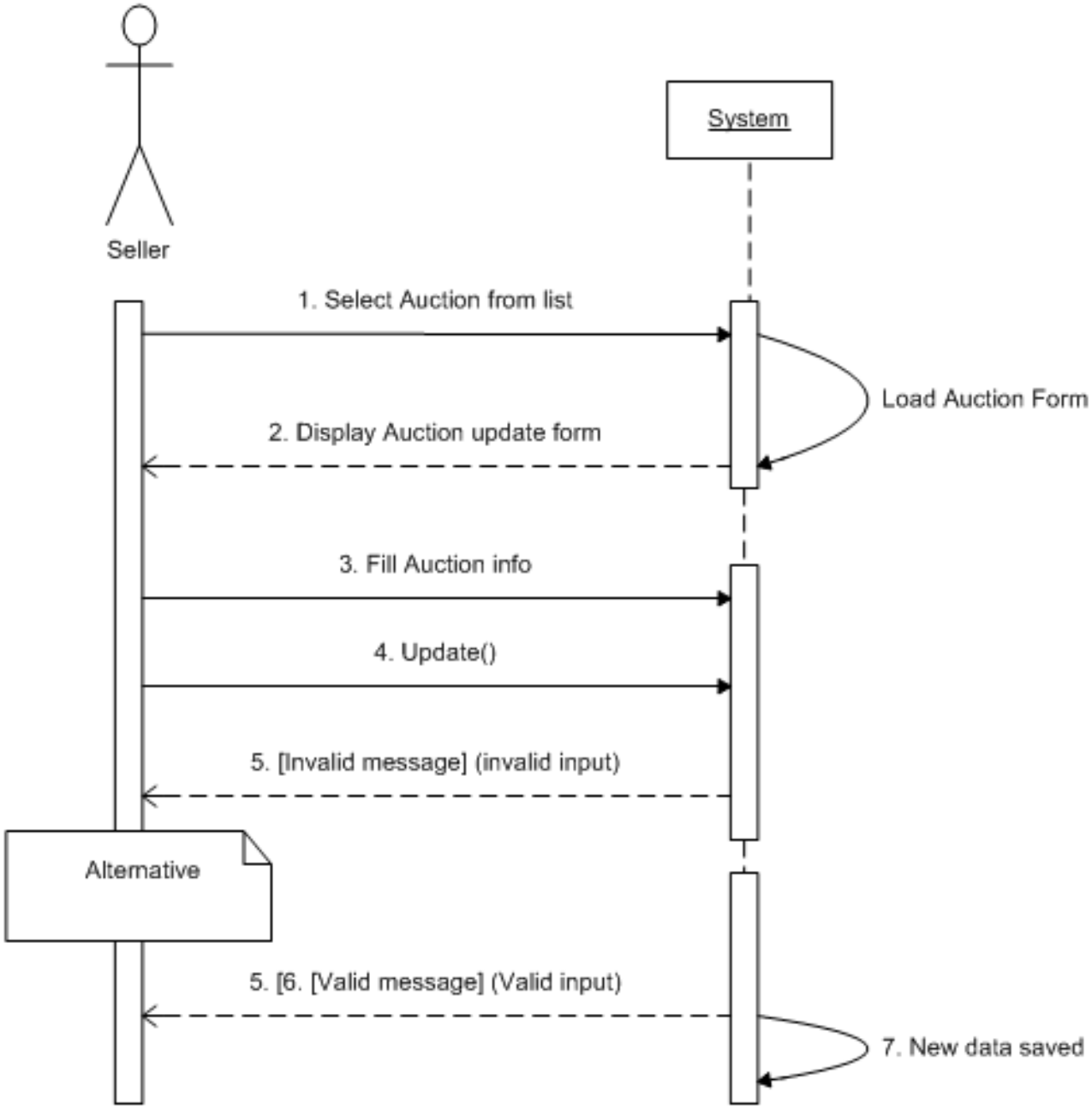


Figure 12. Update Auction Sequence Diagram

6.3.7 Browse/Search Auction

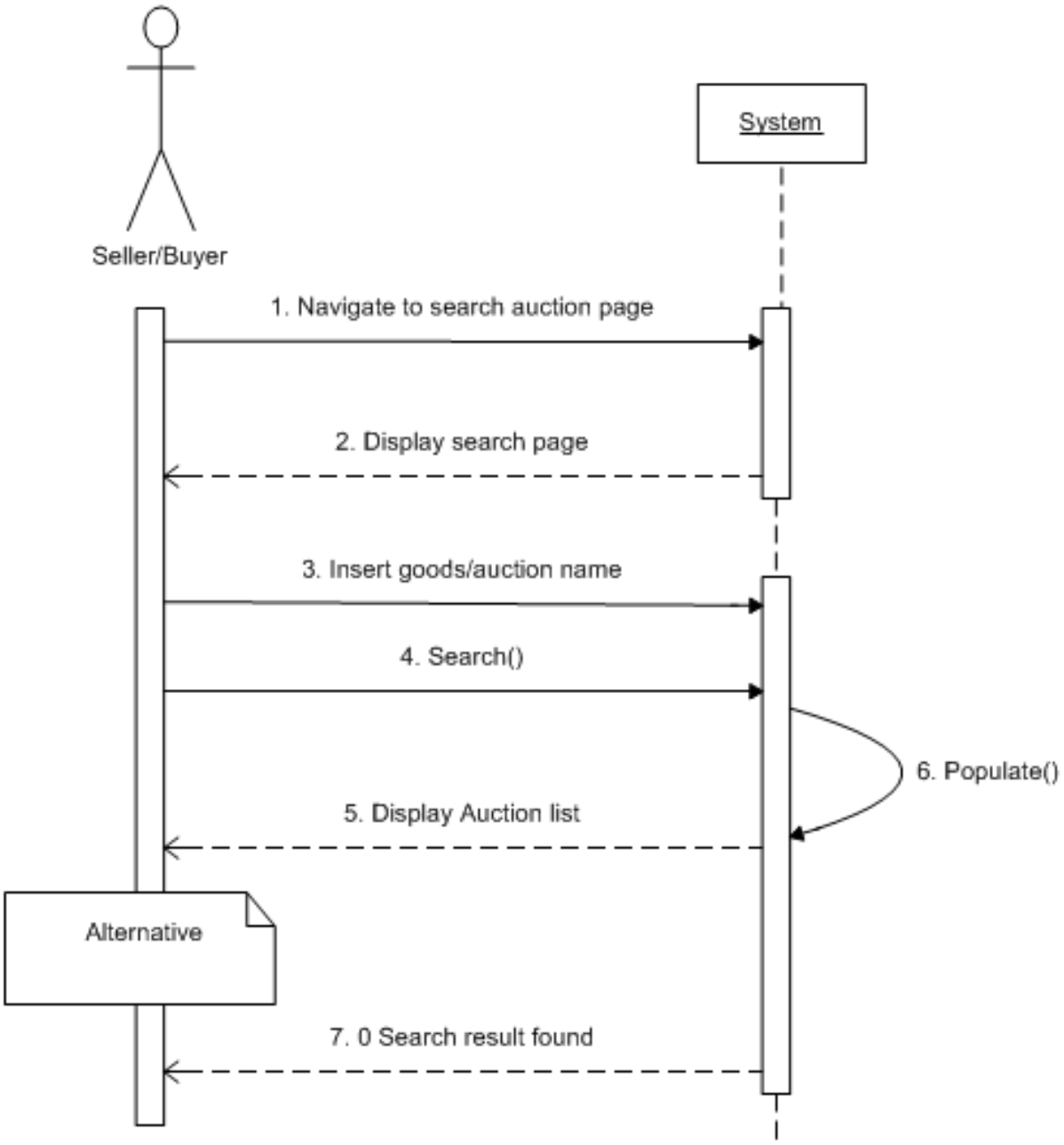


Figure 13. Search Auction Sequence Diagram

6.3.8 Bidding

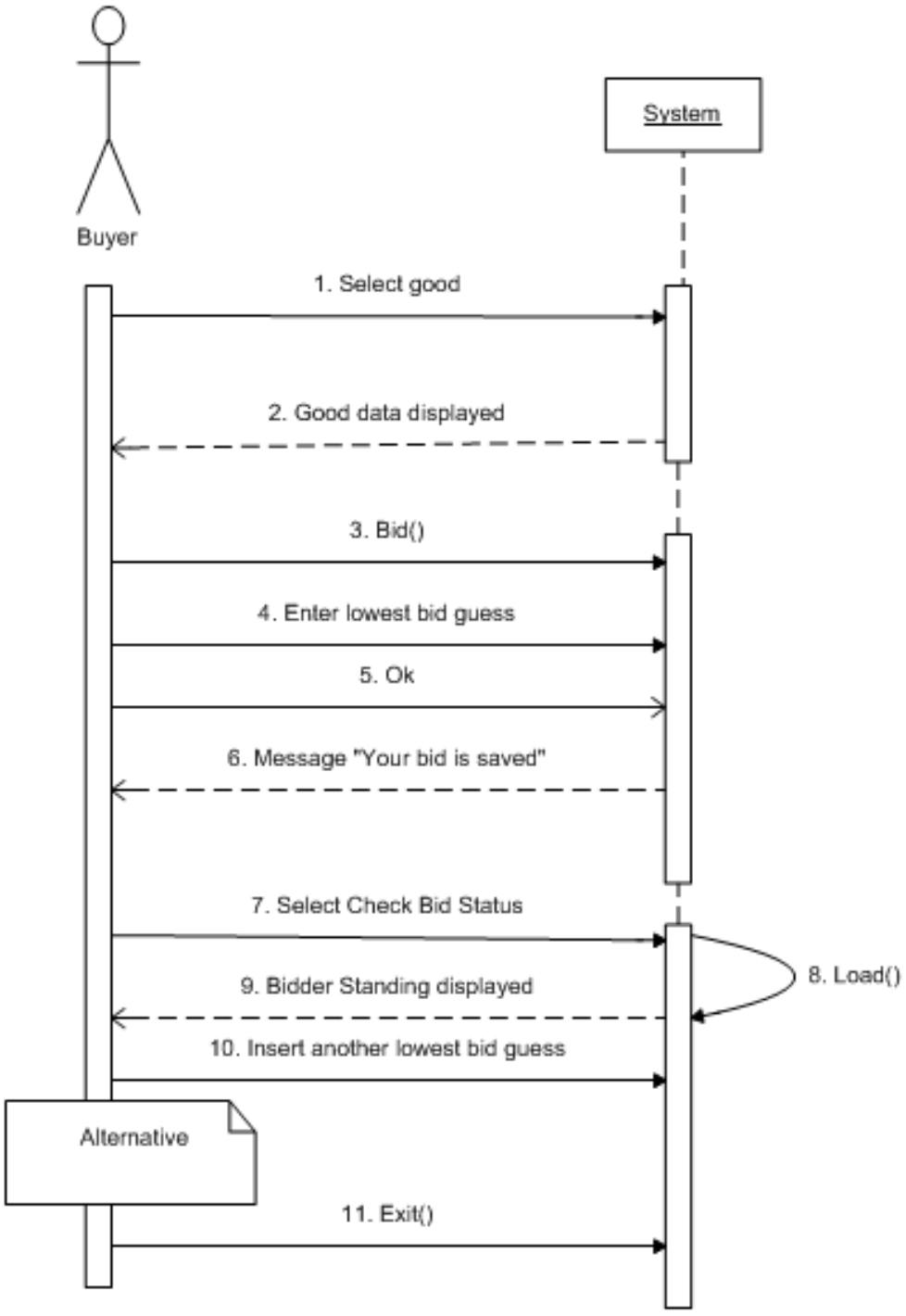


Figure 14 . Bidding Sequence Diagram

6.3.9 Buyer/Seller Confirmation

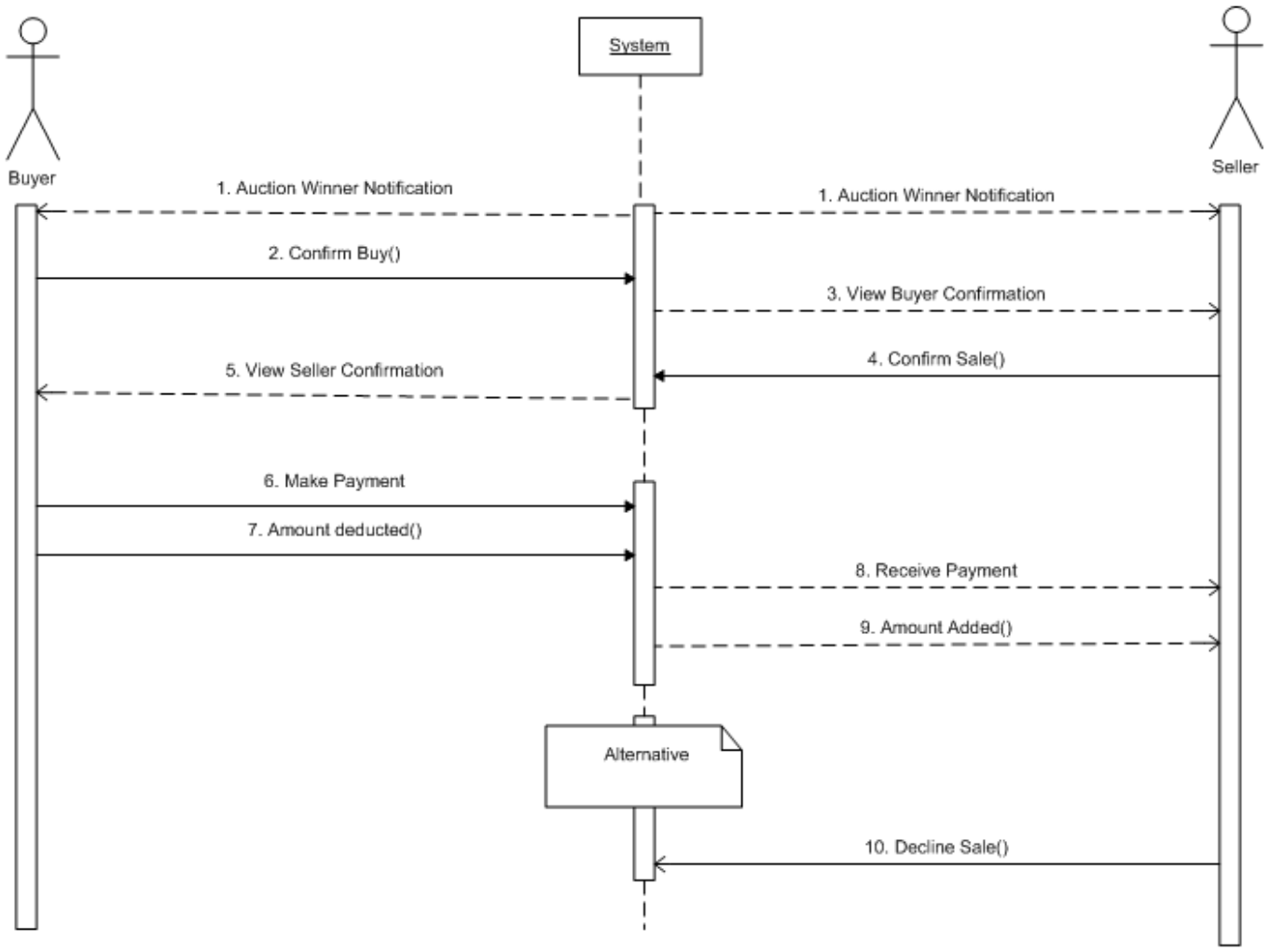


Figure 15 . Buyer/Seller Sequence Diagram

6.4 State Chart Diagram

6.4.1 Auction Class State Chart Diagram

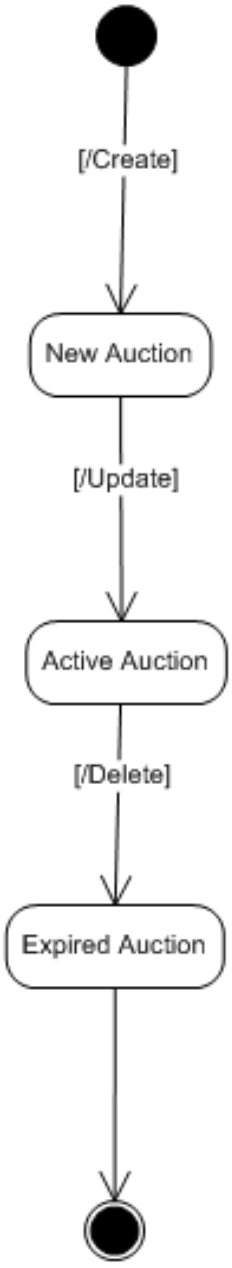


Figure 16. Auction Class State Chart Diagram

6.4.2 Bidding Class State Chart Diagram

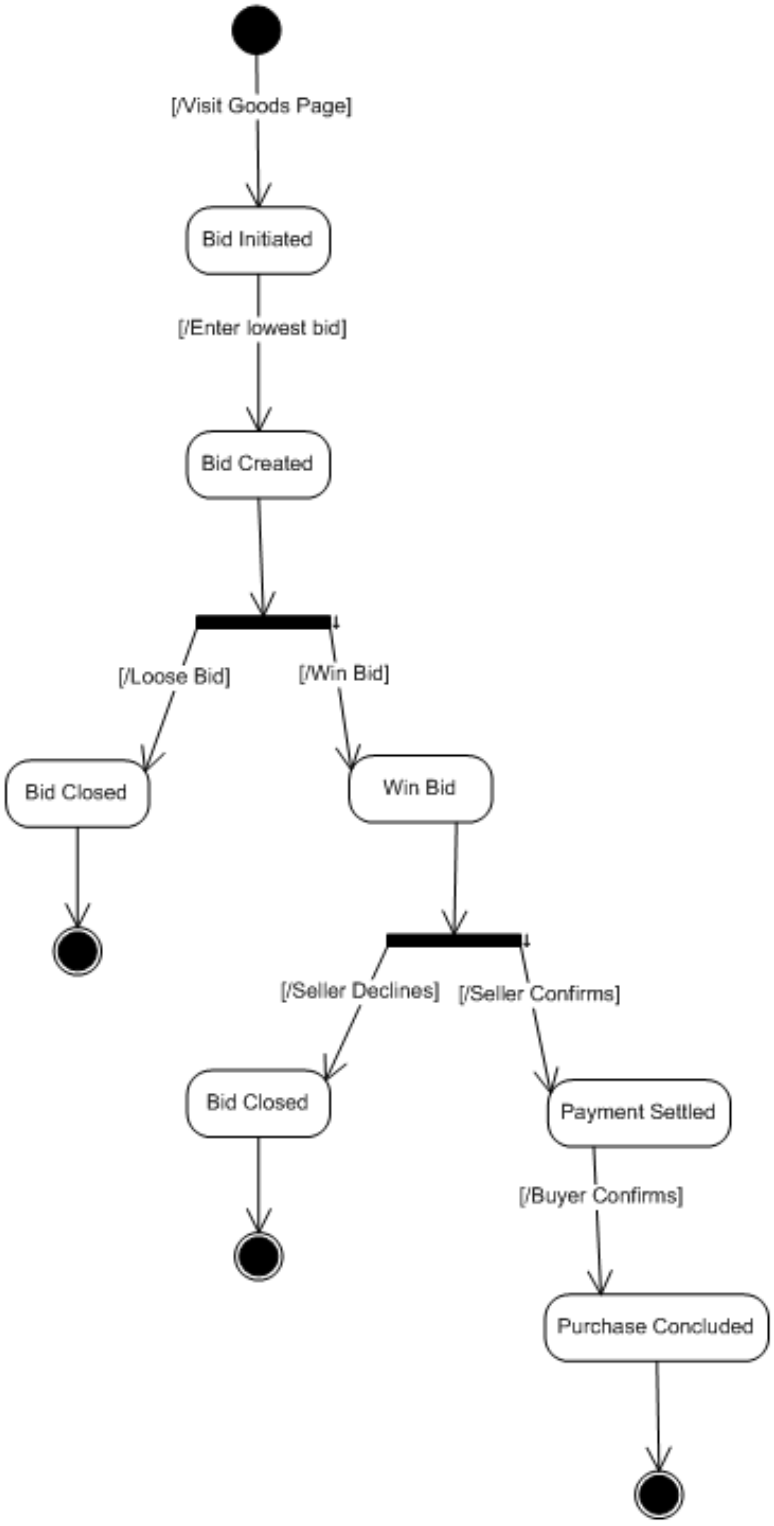


Figure 17. Bidding Class State Chart Diagram

6. Alloy Modeling

Here we are trying to show that our Class Diagram is consistent using Alloy Analyzer. We have included the alloy code and respective diagrams.

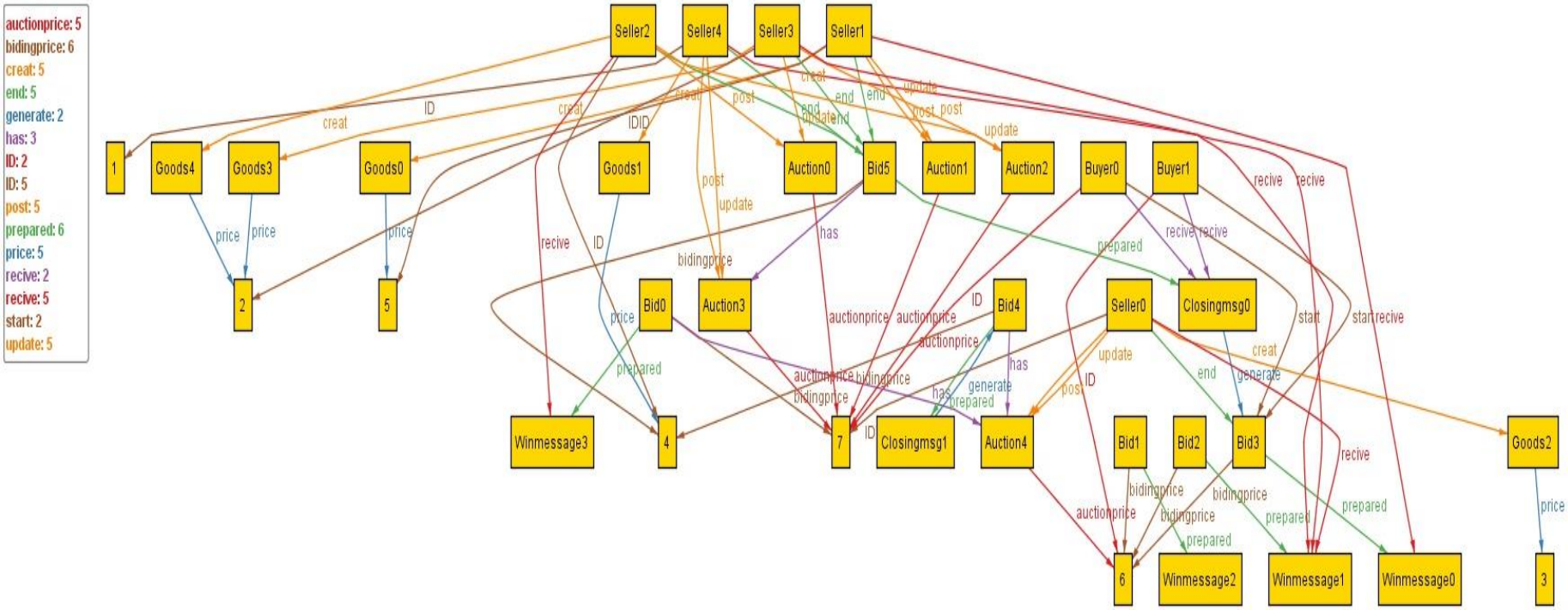
6.1 Alloy Code

```
abstract sig User {
}

sig Buyer extends User
{

start : set Bid,
recive: one Closingmsg,

ID:one Int
}{ID>0}

sig Seller extends User
{
creat: some Goods,
recive :one Winmessage,
post:some Auction,
end :one Bid,
update :one Auction,

ID:one Int
}{ID>0}
```

```
sig Goods
{

price : one Int

}{price>0}

sig Auction {

auctionprice:one Int
}{auctionprice>0}

sig Bid {

prepared : one Notification,

has : lone Auction,
bidingprice:one Int
}{bidingprice>0}

sig Closingmsg extends Notification {

generate: one Bid

}

sig Winmessage extends Notification{

}
sig Notification{


}

// no share id bettwen users
fact UniqId{
all b1,b2: Buyer |b1!=b2 implies b1.ID!=b2.ID
all s1,s2: Seller |s1!=s2 implies  s1.ID!=s2.ID

}
```

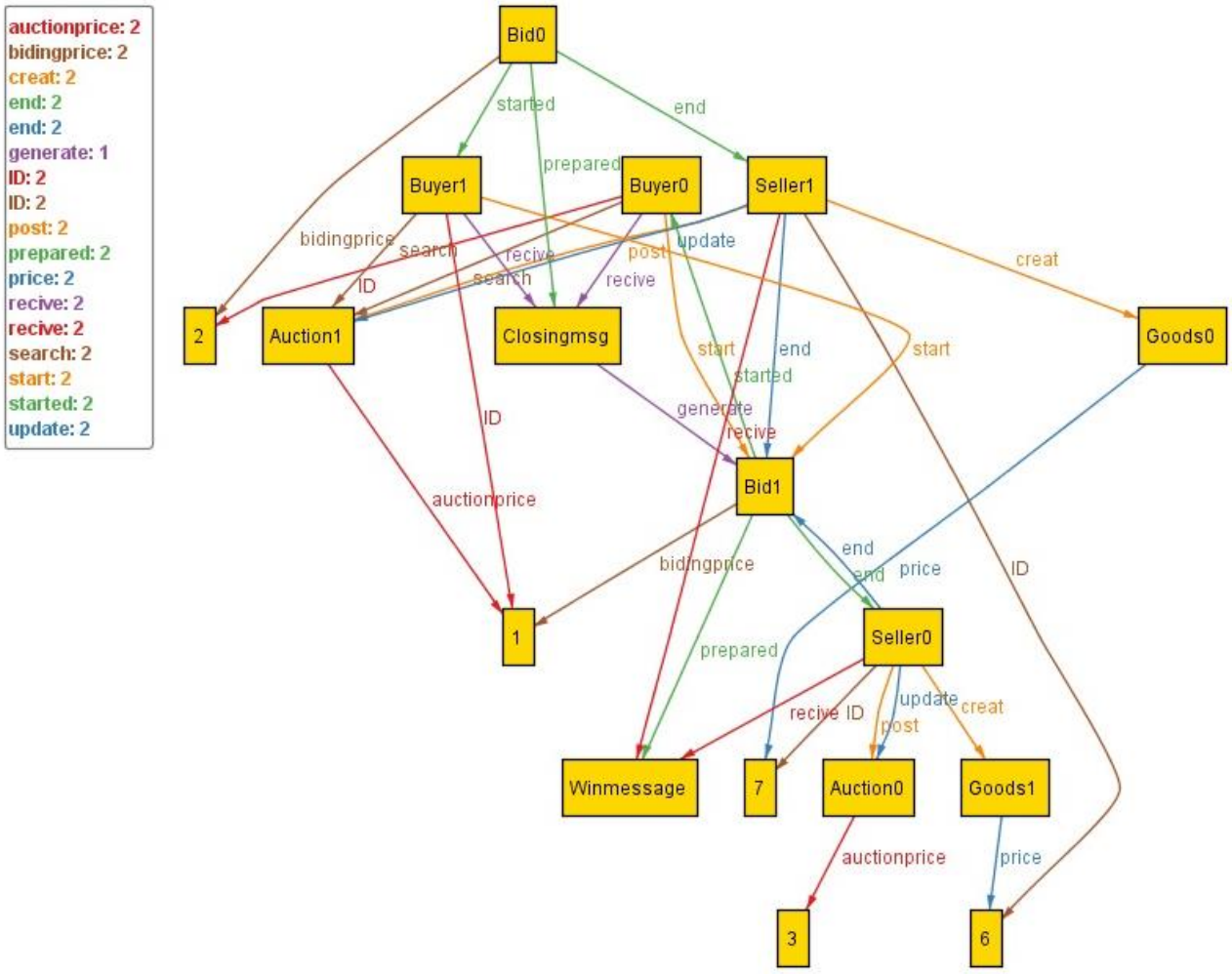
```
//acuton can only be modifed by Seller

fact AcutionCreater{
//each acuton  is created by seller
all a:Auction  | one s: Seller | a in s.post

}
//no Notification exist with out the Bid
fact NotificationProp{
all n: Notification  |  one b: Bid  |  n=b.prepared
}

fact acutionFunctional{
//acution can only be modifed by seller
all a:Auction  | one s: Seller | a in s.update
}

// seller can only creat product
fact GoodsCreater{
all g:Goods | one s:Seller | g in s.creat
}

fact  notificationProp{
// notification is generated if he/she start the biding
//buyer recieve closing notifcation for bid he/she participated
all b:Buyer | b.start = b.recive.generate


}

pred show(){


}

run show for 3 but exactly 2 Seller , 2 Buyer
```

6.2 Main Alloy Diagram



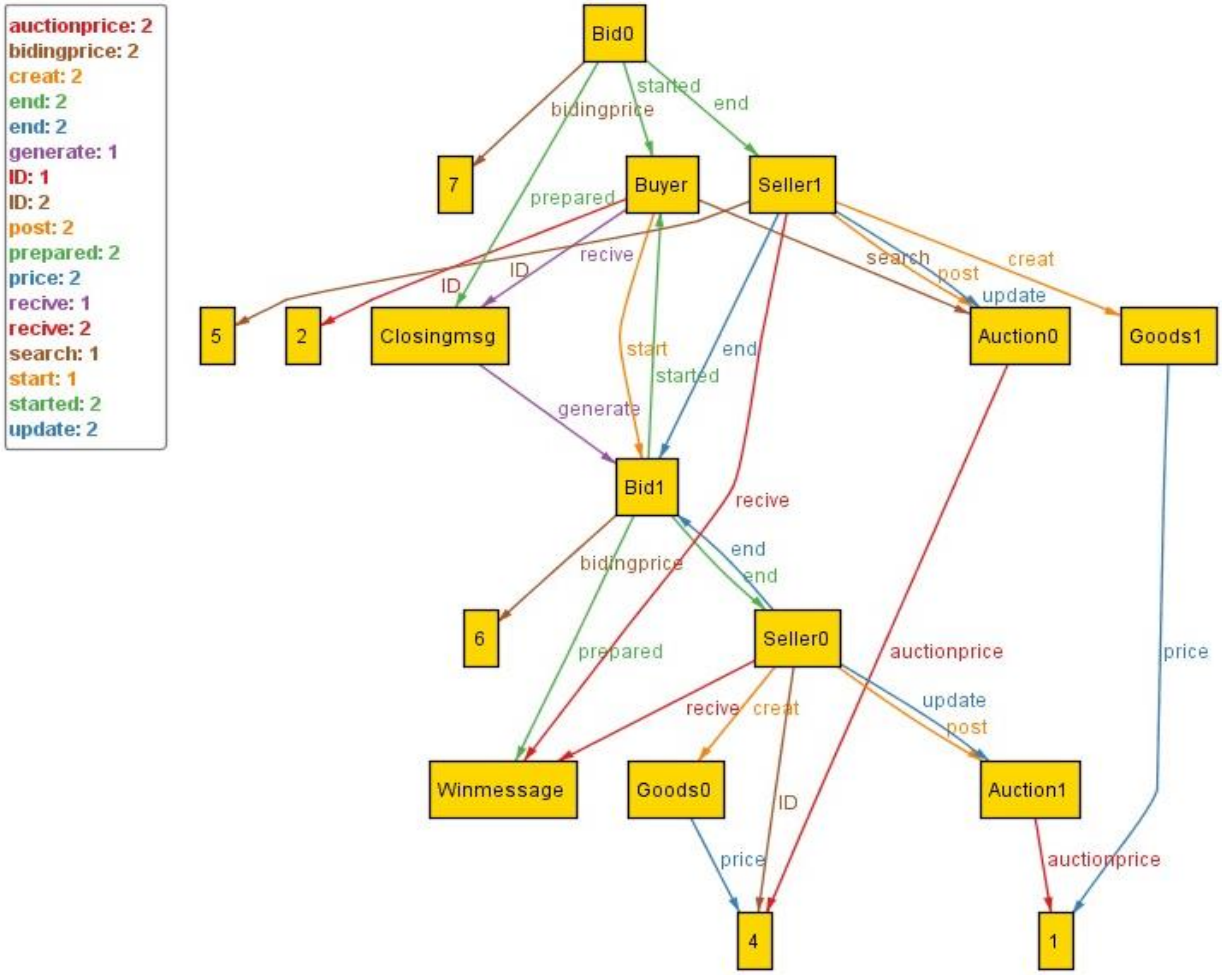
6.3 User Management Diagram

All users(Buyer and Seller) have uniqe ID



6.4 Bid/Auction Closing Diagram

Only Seller can end the bid by putting closing date or closing the bid



7. Used Tools

- **Microsoft Word 2010:** to prepare and compile document
- **Microsoft Visio 2007:** to create UML diagrams
- **Alloy Analyzer 4.2:** to show the consistency of our models