POLITECNICO

MILANO

GUESSBID

Requirements Analysis and Specification Document

Author: Nguyen Van Minh

April,27, 2015

CONTENTS

1 INTRODUCTION

1.1 DESCRIPTION

This project will simulate the inverse auction system. By comparison with the traditional one, the buyer sets up bidding item hence the sellers will try to compete with another. Regarding the rule, the seller who issues *smallest unique bid* is winner.

Registration users in this system should have right to create and manage auctions and take part in another. In addition, after finishing auction the transaction will be started and the money is taken away from the winning seller's credit.

1.2 GOALS

This system has to provide main features respecting possible users:

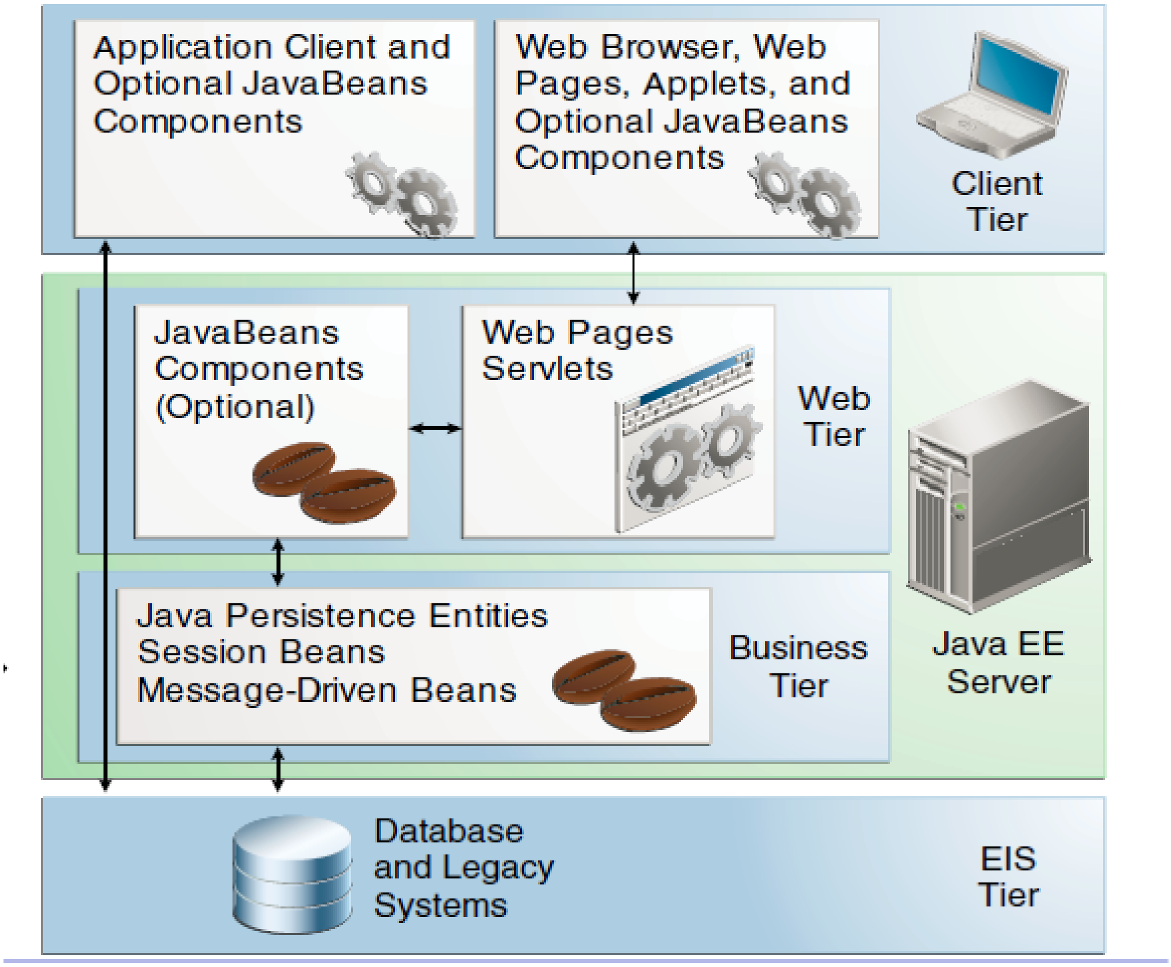
- Registration of person to the system.
- Linking user to a certain virtual credit.
- Finding and joining, checking status of any unexpired auctions from other users.
- Managing his/her own auction: set the expired date, close the auction before due date, change information about item.

1.3 ARCHITECTURE

This system applies multi-tier architecture of EJB web application in terms of giving nice solution on working logically and easily.

The main objective of Java EE application here is that enables developers to concentrate on business logic which makes the whole system is clean, well-organized.

Users can access to the web application (client tier) which interacts with the system through web interface. In the meantime, the system runs within Web tier, Business tier and Database (EIS tier). Developer has huge power to design the logic of system, oriented class by Java beans as well as creating separated User Interface by Web pages servlet.


Application Client and Optional JavaBeans Components
Web Browser, Web Pages, Applets, and Optional JavaBeans Components
Client Tier
JavaBeans Components (Optional)
Web Pages Servlets
Web Tier
Java Persistence Entities Session Beans Message-Driven Beans
Business Tier
Java EE Server
Database and Legacy Systems
EIS Tier

1.4 Glossary

In the first place, I need to clarify some words using in the system:

- User: apparently the registration users which followed successfully registration process.
- Auction: notified as real life auction which one item will be bring out to estimate. Only one among the competitions is winner respecting the specific rule.
- Buyer: user who set up the auction.
- Seller: user who try to win the auction.
- Bid: the amount of credit put on the auction to guestimate the item.

1.5 Identifying stakeholders

Our "financial" stakeholder is the professor who provide me this respectful project. As far as a mandatory project, the professor would like to have me pass by all the development process truly as a completed software.

Thus I need to follow from the beginning to the end of software process including requirement analysis documentation, design document, implementation, project reporting and final presentation. As a matter of fact, this project tends to run as well as EJB web application with required functionalities.

I might consider this project is attempt to lack some compatible functionalities in compared with another existed large systems as biddingo.com. By the way, my purpose is trying my best with respect to understanding well through all development process by multi-tier architecture of EJB platform and provide acceptable not only functions but also UI for users.

2 ACTORS

In view of a developer, defining actor is important in relation to . These actors have their specific roles and privileges in the system respectively. I indicate actors as below:

- Guest: the one who only access to home page of system which merely involves information about system.
- User: the registration user who possibly create/manage his/her own auctions and join another ones.
- Administrator: the only one is possible to check and manage all users in the system.
- System: take part in applying the rule of inverse auction, select the winner and automatically calculate the cost of each issuing bid as well as the bidding cost of winner (which can be considered as a hidden actor).

3 REQUIREMENTS

As aforementioned, for each goals I will derive:

- Registration of person to the system:
  - The system has to provide sign up functionality.
- Linking user to a certain virtual credit:
  - The system will create a virtual credit of 100 for registration user (as requiring from the description of project). Also bearing in mind that issuing bid costs of 2.
- Finding and joining, checking status of any unexpired auctions from other users:
  - The system should provide well-organized auction page to user with briefly description, status for each auctions.
- Managing his/her own auction: set the expired date, close the auction before due date, change information about item:
  - The system will be able to allow user all those above activities concerning every auctions. Thus user can intuitively control his/her auction so far.

3.1 FUNCTIONAL REQUIREMENTS

For each defined actors above, GuessBid project must identify some functional requirements as well:

- Guest: those can only have basic functionalities so he can:
  - See the public information of the system.
  - Sign up.
- User: is main actor of system who can:
  - Log in.
  - Modify his/her profile information.
  - Create/manage auction.
  - Compete in another auction.
- Administrator: is in charge of verifying users in the whole system.
  - Check users' public data.
  - Remove any users who can build up risks to the system (spammer, unreliable profile…).
- System: is responsible for running the rule of inverse auction, so it can:
  - Notify the competitors when auction is closed.
  - Pick up the winner (lowest unique bid).
  - Withdrawal from virtual credit of users.

3.2 NON FUNCTIONAL REQUIREMENTS

3.2.1 User interface

As mentioned earlier, the system is built over EJB platform and I decide to provide UI through web application.

3.2.2 Documentation

There are some documentations will be found somewhere in this project. I list all in dedication to clarify them:

- RASD: Requirement Analysis and Specification document, to illuminate the given problem, analyze the goals, requirements and specification.
- DD: Design documentation, to define the architecture of software elements in practice and logic.
- JavaDoc comments in source code, to give better understanding on the source code.
- Installation manual: guide to install GuessBid.
- User guide: guide to use GuessBid.

4 SPECIFICATION

5 SCENARIOS

We try to point out some possible scenarios:

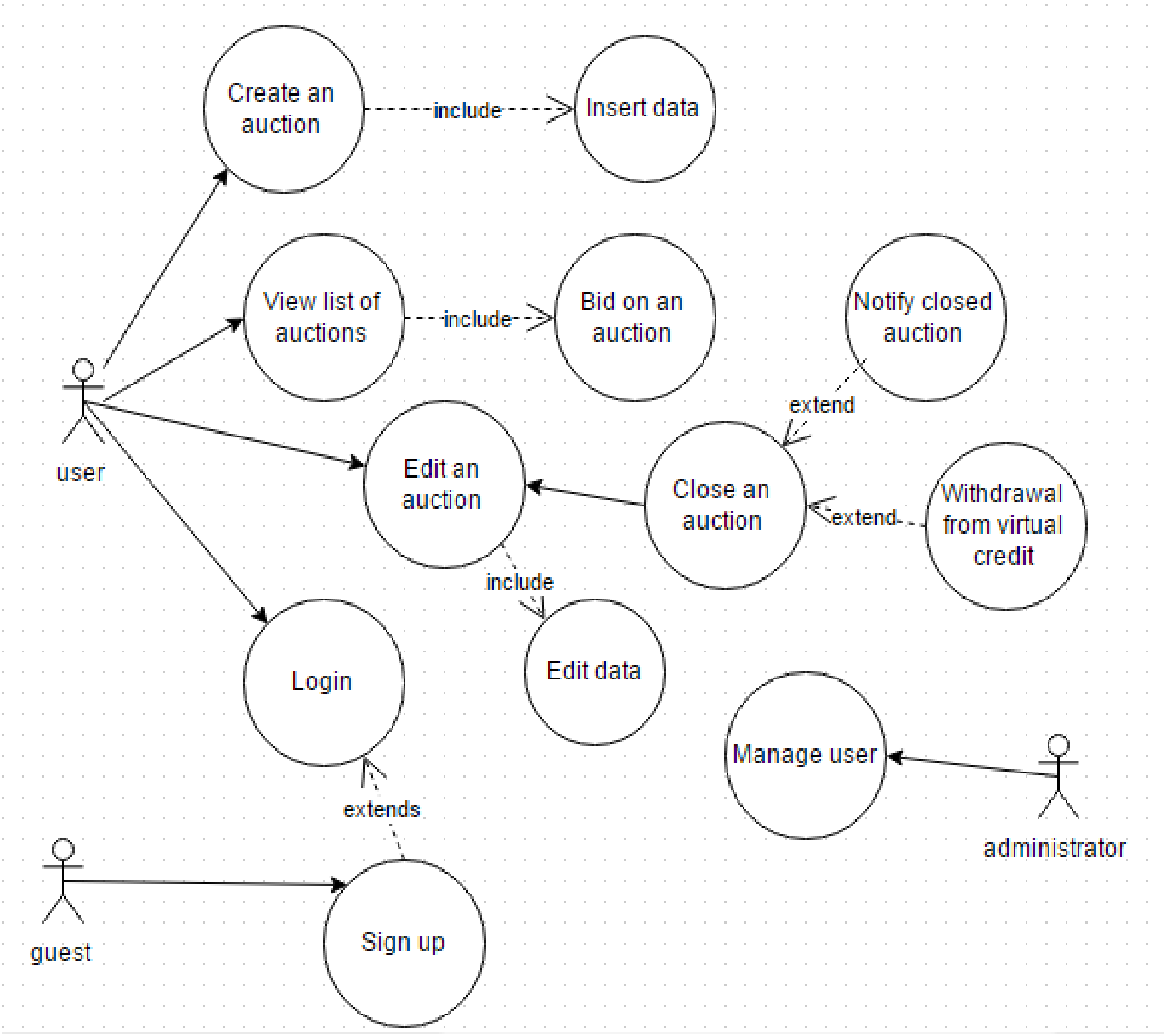
- Francesca is a doctor but she always want to open an orphanage to help lost children since she was a little girl. One day, she and her friends embark on the real work. Indeed, she is in charge of buying house equipment. She is wondering few days of reasonable sellers then accidentally a friend of her tells about the website which can handle her situation best. Therefore Francesca checks on the website, after reading the brief information of the systems in connection with huge potential sellers in many categories, she totally realize the benefit of the system and create an account immediately.
- Francesca brings her auction with careful description to the website after a week. She also sets the starting expected price, expired date within one month (which should be synchronized with her friends' work on building completely the outer of the orphanage).
- Francesca receives over 10 documentation of sellers in merely one week. They are all applicable to her willing. She is very satisfy as soon as the system pick the winner after closing the auction. The seller contacts directly to her afterwards and everything goes well to the new orphanage.
- Francesca now checks on the website unintentionally. She clicks on the home page and surprisingly finds an interesting auction. This auction concerns "Hans Andersen fairy tales" collection version 1997 which has meaningful portrait on the cover of every individual volume. The buyer has every parts but volume 2 and somehow Francesca has that book in her old bookshelf. The buyer – book collector seems excited of this collection and he puts on very high price for that book. In fact, Francesca has already bought a new completely collection of this lovely "Hans Andersen" stories. Accordingly, she joins in the auction, checks the current status of all other bidding competition. Finally, she brings out her bid which is smallest unique and still very high price as regard to an old book. Without doubt, she wins the auction by a smile of an ordinary nerd.

6 UML MODELS

6.1 USE CASES DIAGRAM

Deriving from scenario above, I bring here some use cases in details:

- Sign up.
- Login.
- Create an auction.
- Bid on an auction.
- Edit an auction.
- Check current status of bidding in auction.



6.2 USE CASES DESCRIPTIONS

Name	Sign up
Actors	Guest
Entry conditions	The guest who is new with the system and in the home page
Flow of events	• The guest clicks on the “Sign up” button. • The system shows him/her the signing-up page correspondingly; • The user fulfill the information into required fields (email address, password); • The user clicks the button “OK”. • The system shows the home page and a small area which displays his/her logged in state.
Exit conditions	The new user data is now updated in database and the guest becomes an user.
Exceptions	If the inserted information is wrong in some cases, the warning message will be shown.

Name	Log in
Actors	Users or administrator
Entry conditions	The user has successfully signed up to the system
Flow of events	• The user opens the home page; • The system shows him/her the page; • The user enters his e-mail address and password into the provided input form; • The user clicks the button “log in”. • The system shows the profile page.
Exit conditions	No exit conditions.
Exceptions	If the information put on the form is wrong, the warning message is shown.

Name	Create an auction
Actors	Users
Entry conditions	The user has already logged in to the system and in the home page.
Flow of events	• The user clicks on the “create an auction” button;

	- The system shows him/her the draft page;
Exit conditions	User might cancel if he/she changes his/her mind not to create new auction anymore.
Exceptions	No exception.

Name	Insert data
Actors	Users
Entry conditions	The user has already logged in to the system and in the new auction page.
Flow of events	- The user fulfil the information over the required fields and click the button "OK"; - The system shows the new auction page
Exit conditions	The data is now stored in database and the auction is basically starting.
Exceptions	If user doesn't fulfil all the left blank required fields or the inserted information isn't accepted, the warning message will be shown.

Name	View list of auctions
Actors	Users
Entry conditions	The user has already logged in to the system and in the home page.
Flow of events	- The system shows him the list of auctions which are currently available. - The user can click to expand viewing of the list.
Exit conditions	The user might click on any auction link to the specific auction page.
Exceptions	No exceptions.

Name	Bid on an auction
Actors	Users
Entry conditions	The user has already logged in to the system and in the auction page.
Flow of events	- The system shows him/her the auction page with all important information (involved also his/her

	currents status of bidding if he/she already did once); • The user puts the bidding value and clicks the button “OK”; • The system shows the updated auction page hence he/she can check again his/her current status of bidding (old and new).
Exit conditions	The data is now stored in database.
Exceptions	If user doesn’t fulfil all the left blank required fields or the inserted information isn’t accepted, the warning message will be shown.

Name	Edit an auction
Actors	Users
Entry conditions	The user has already logged in to the system and in the home page.
Flow of events	• The user clicks on any auction which is belong to him/her in an area display in homepage. • The system shows him/her the auction page correspondingly;
Exit conditions	The data is now updated in database.
Exceptions	If the inserted information isn’t accepted, the warning message will be shown.

Name	Edit data
Actors	Users
Entry conditions	The user has already logged in to the system and in the editing auction page
Flow of events	• The user modifies the fields which is changeable (description, expired date) ; • The user clicks the button “OK”. • The system shows the auction page with all updated information.
Exit conditions	The data is now updated in database.
Exceptions	If the inserted information isn’t accepted, the warning message will be shown.

Name	Close an auction
Actors	Users

Entry conditions	The user has already logged in to the system and in the editing auction page
Flow of events	- The user clicks the button "Cancel". - The system shows the warning message. - The user
Exit conditions	The data is now updated in database.
Exceptions	If the inserted information isn't accepted, the warning message will be shown.

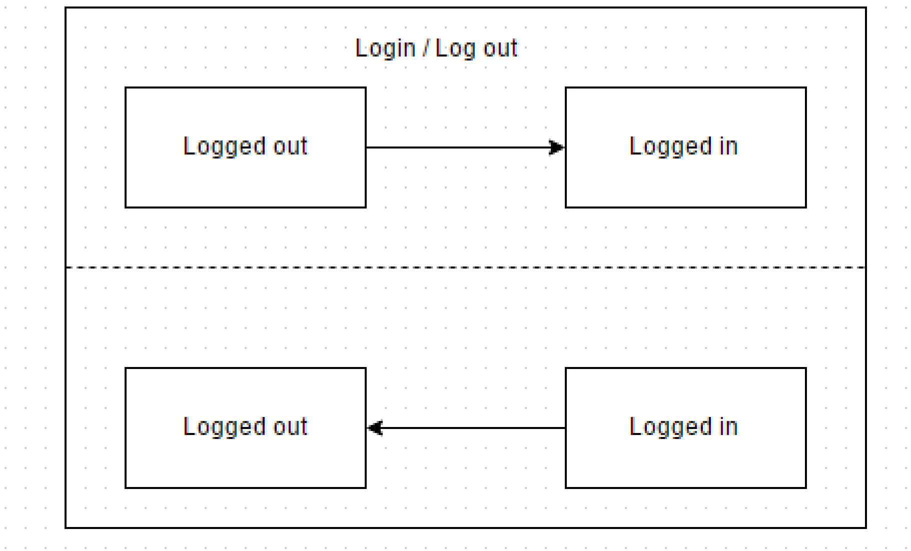
Name	Notify to users
Actors	System
Entry conditions	The user has already logged in to the system and closed an auction.
Flow of events	- The system will notify to users which are competitions in the auction.
Exit conditions	
Exceptions	No exceptions.

Name	Withdraw from virtual credit
Actors	System
Entry conditions	The user has already logged in to the system and one of the auction whose he/she took part in is just closed.
Flow of events	- The system will immediately withdraw money from virtual credit of users who took part in the auction.
Exit conditions	The virtual credit of users are updated.
Exceptions	No exceptions.

Name	Check current status of bidding in an auction
Actors	Users
Entry conditions	The user has already logged in to the system and in the specific auction page.
Flow of events	- The user clicks on "Bidding status" in a special area display in auction page. - The system shows him/her the current status of bidding in the auction (included him/her and another competitors);

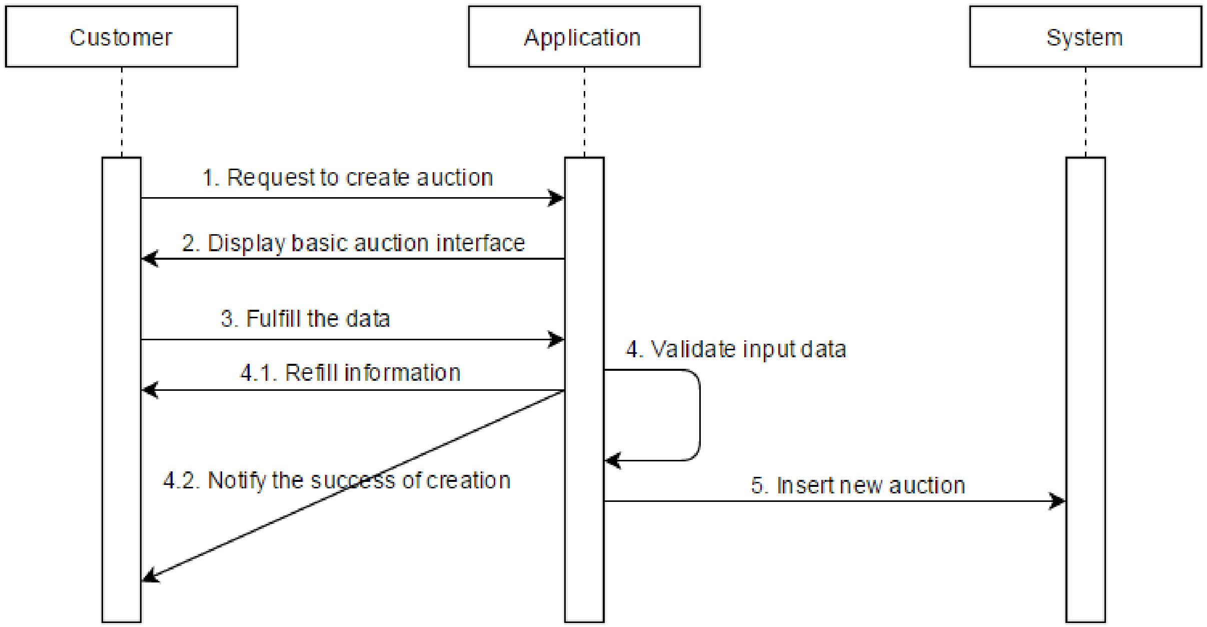
	- The user modifies the fields which is changeable (description, expired date) ; - The user clicks the button “OK”. - The system shows the auction page with all updated information.
Exit conditions	The data is now updated in database.
Exceptions	If the inserted information isn’t accepted, the warning message will be shown.

6.2.1 State diagram

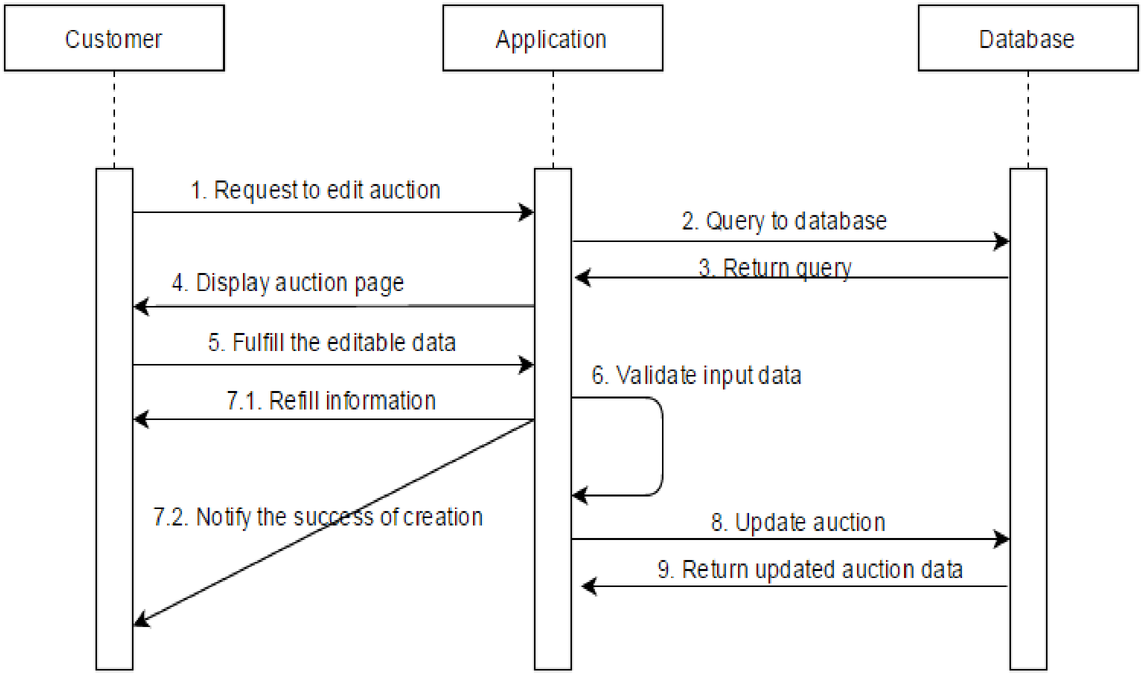


6.2.2 Sequence diagram

Create auction:

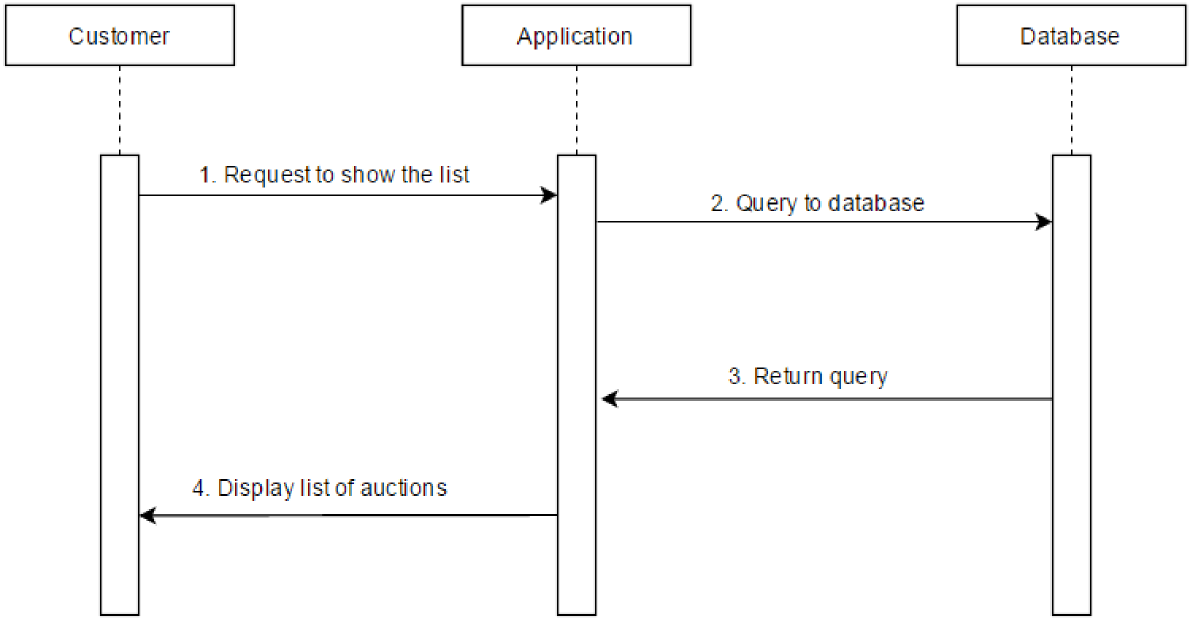


Edit auction:

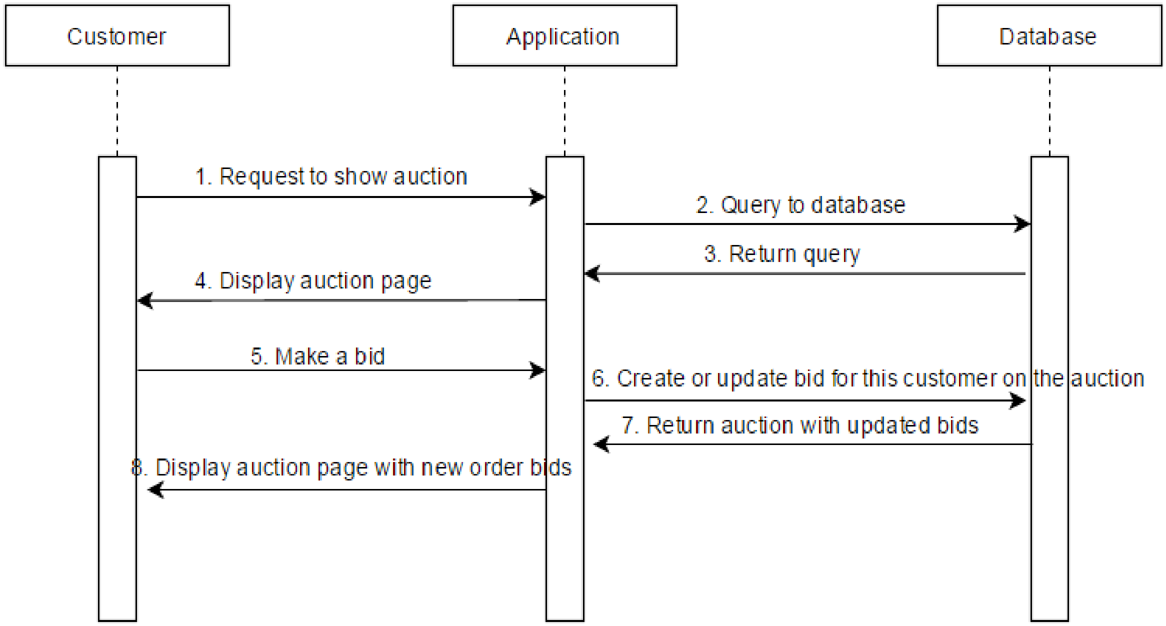


Display the list of auctions (could have parameter to show based on purpose of customer)

- List of auctions which is belong to customer
- List of auctions that customer is following
- Another auctions



Make a bid:


Customer
Application
Database
1. Request to show auction
2. Query to database
3. Return query
4. Display auction page
5. Make a bid
6. Create or update bid for this customer on the auction
7. Return auction with updated bids
8. Display auction page with new order bids

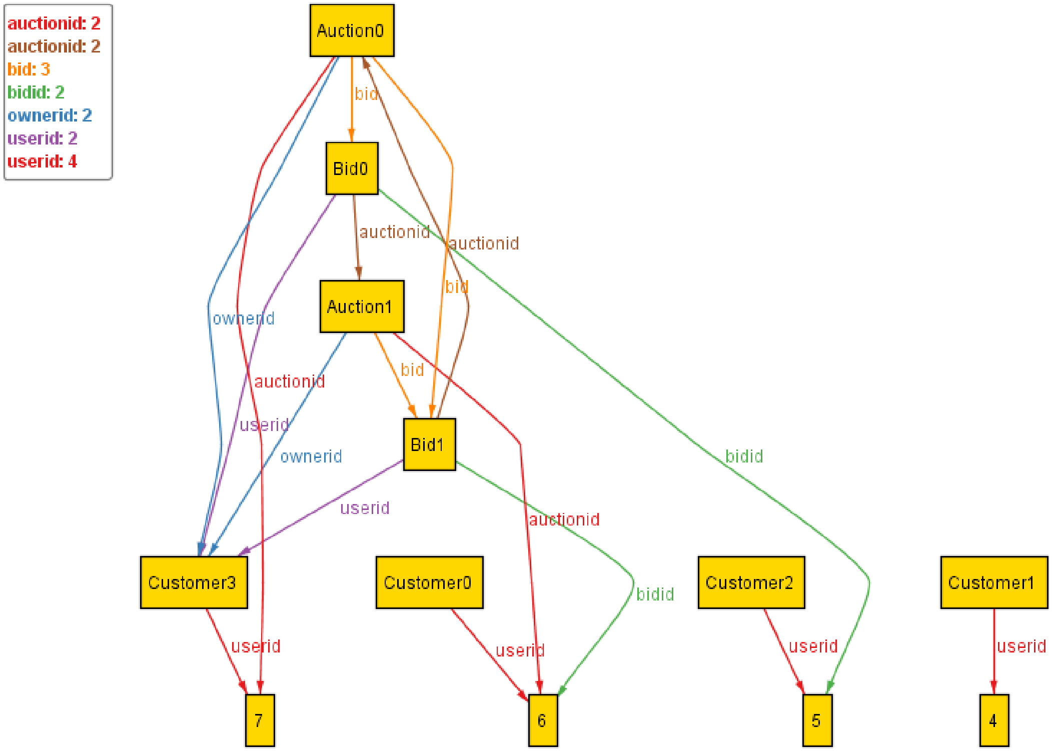
7 ALLOY MODELS

The system will be modelled using Alloy. The source code generates below and also the diagram

```
abstract sig User{
        userid: one Int
}
sig Admin extends User{}
sig Customer extends User{}
sig Auction{
        auctionid: one Int,
        ownerid: one Customer,
        bid: some Bid
}
sig Bid{
        bidid: one Int,
        auctionid: one Auction,
        userid: one Customer
}
fact Userid{
        // user id is unique
        all disj u1, u2: User | u1.userid != u2.userid
        // auction id is unique
        all disj a1, a2: Auction | a1.auctionid != a2.auctionid
        // bid id is unique
        all disj b1, b2: Bid | b1.bidid != b2.bidid
}

pred show[]{}

run show for exactly 4 Customer, 1 Admin, 5 Auction, 15 Bid
```



8 Used tools

To illustrate all the images, diagrams in this document, I have used these tools:

- Website: https://www.draw.io/ : in order to draw diagram
- Alloy program: regard to run Alloy model and show the Alloy test case.