GuessBid

Requirements Analysis and Specification Document

Milica Shulevska
POLITECNICO DI MILANO | SOFTWARE ENGINEERING 2
27.04.15

CONTENTS

1 INTRODUCTION

1.1 PURPOSE

The purpose of this Requirements Analysis and Specification Document is to describe the general functionalities of the GuessBid application assigned as a project in the Software Engineering 2 course. Moreover, it contains all the critical aspects of the application and establishes an agreement between a client and a developer.

The intended audience of this document is all the people actively participating in the software engineering course, including professors and tutors. Not only shall the document serve as a reference for the developers to follow the requirements, but it will also serve to the testers to check whether the stated requirements and goals are met or not.

1.2 SCOPE

The software product that is going to be developed is the GuessBid application. The application will be implementing an inverse auction system. An inverse auction works like a regular auction. The difference is that in an inverse auction a user has to propose the lowest unique bid to win the auction.

The product has the following general objectives:

- Let user register on the system.
- Let user login to the system.
- Let user create an auction for each good he/she has, defining an expiration date after which the auction expires.
- Let user browse existing auctions.
- Let user bid for an existing auction.
- Let user be informed by the system about the current status of his/her biddings for opened auctions.
- Let user be notified about the outcome of the auction when it is closed.

The product will have the following general functionalities:

- **Manage users**
- **Manage auctions**
- **Manage bids**

The product will have the following limitations:

- **Password change**

The user will not be able to change or reset his/her password. The system will not have this option.

- **One auction at a time**

The users will not be able to create multiple auctions at the same time. The system will only offer the option to create only one auction at a given time by one user.

The software will have the following goals:

[G1] Users can register on the system.

[G2] Users can create an auction.

[G3] Users can browse existing auctions.

[G4] Users can bid on existing auctions.

[G5] Users can provide more than one bid before closing time of the auction.

[G6] User with a lowest unique bid is the winner.

[G7] Issuing bids has a cost of 2 per bid.

[G8] Users are informed by the system about the current status of his/her biddings on the open auctions at each bid.

[G9] Users are notified about the outcome of the auctions they have a connection to.

[G10] Users can login.

[G11] Users can logout.

1.3 Definitions, Acronyms and Abbreviations

1.3.1 Definitions

Keyword	Definition
User	The person who will use the system.
Bid	Offer, a certain price for something, especially at an auction.
Auction	A public sale in which goods or property are sold to the highest bidder.
Inverse auction	A unique bid auction is a type of strategy game related to traditional auctions where the winner is usually the individual with the lowest unique bid.
Bidder	Someone who makes bids for something, for example at an auction.

Table 1: Definitions

1.3.2 Acronyms and abbreviations

Acronym/Abbreviation	Definitions
XML	Extensible Markup Language
RASD	Requirements Analysis and Specification Document
NFR	Non-Functional Requirements
FR	Functional Requirements
QA	Quality Attributes
G	Goal
DBMS	Data Base Management System
AS	Application Server
JEE	Java Enterprise Edition

Table 2: Acronyms and abbreviations

1.4 References

1. IEEE Recommended Practice for Software Engineering Requirements Specification (http://ieeexplore.ieee.org/xpl/mostRecentIssue.jsp?reload=true&punumber=5841)
2. Alloy model file: guessBid.als

1.5 Overview

This document is organized as follows:

1. **Introduction**

   This section provides a synopsis of the software product to be developed.

2. **Overall Description**

   This section describes the general factors that affect the software product and its requirements. For instance, the different interfaces, operations, functions, user characteristics, constraints, assumptions and dependencies and future requirements.

3. **Specific Requirements**

   This section contains all the analysis done to the project requirements. It describes all of the software requirements to a level of detail sufficient to be externally perceivable. It contains the different use cases, interfaces, scenarios and sequence diagrams, among others.

4. **Appendices**

   This section provides supporting information showing how the alloy model contributed to the analysis model and requirement analysis.

2 OVERALL DESCRIPTION

This section describes the general factors that affect the software product and its requirements, and it provides a background for specifying concrete requirements in the next section of this document.

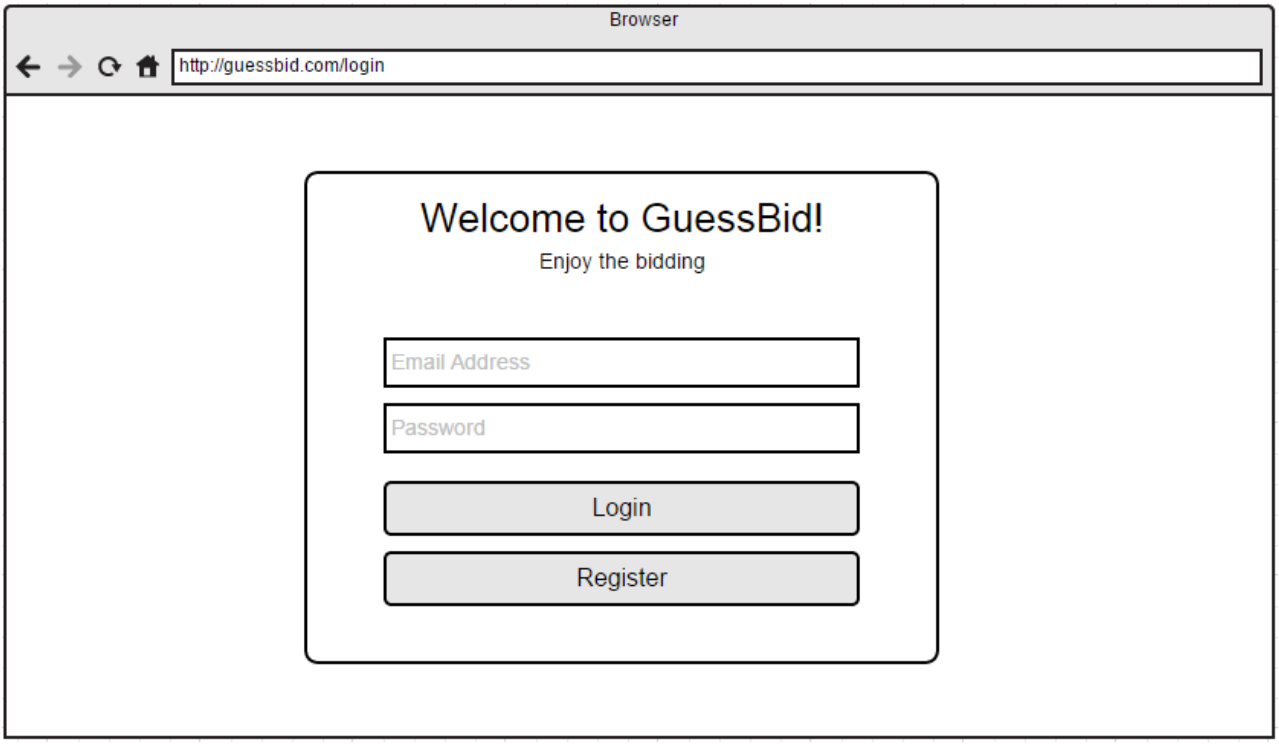
2.1 PRODUCT PERSPECTIVE

The software product is a completely self-contained system, independent from other systems.

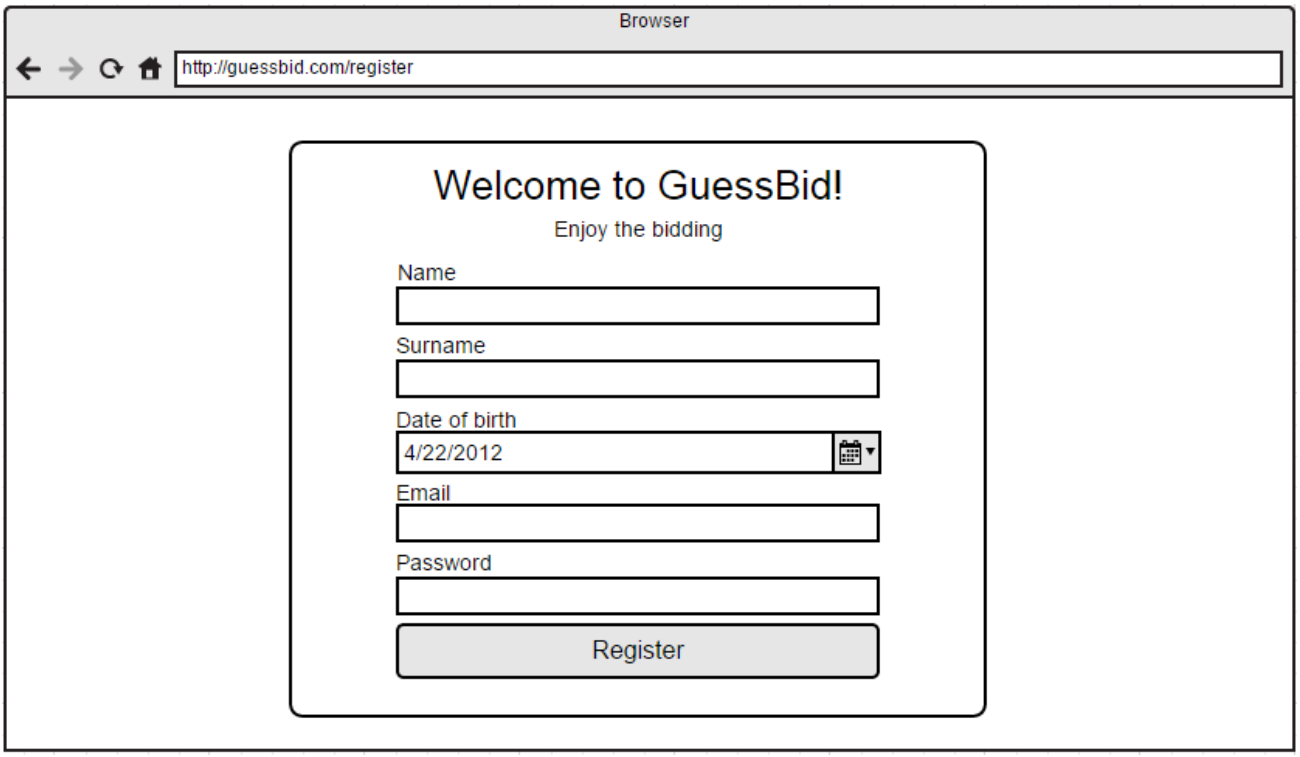
2.1.1 System Interfaces

The software product does not provide any external interfaces.

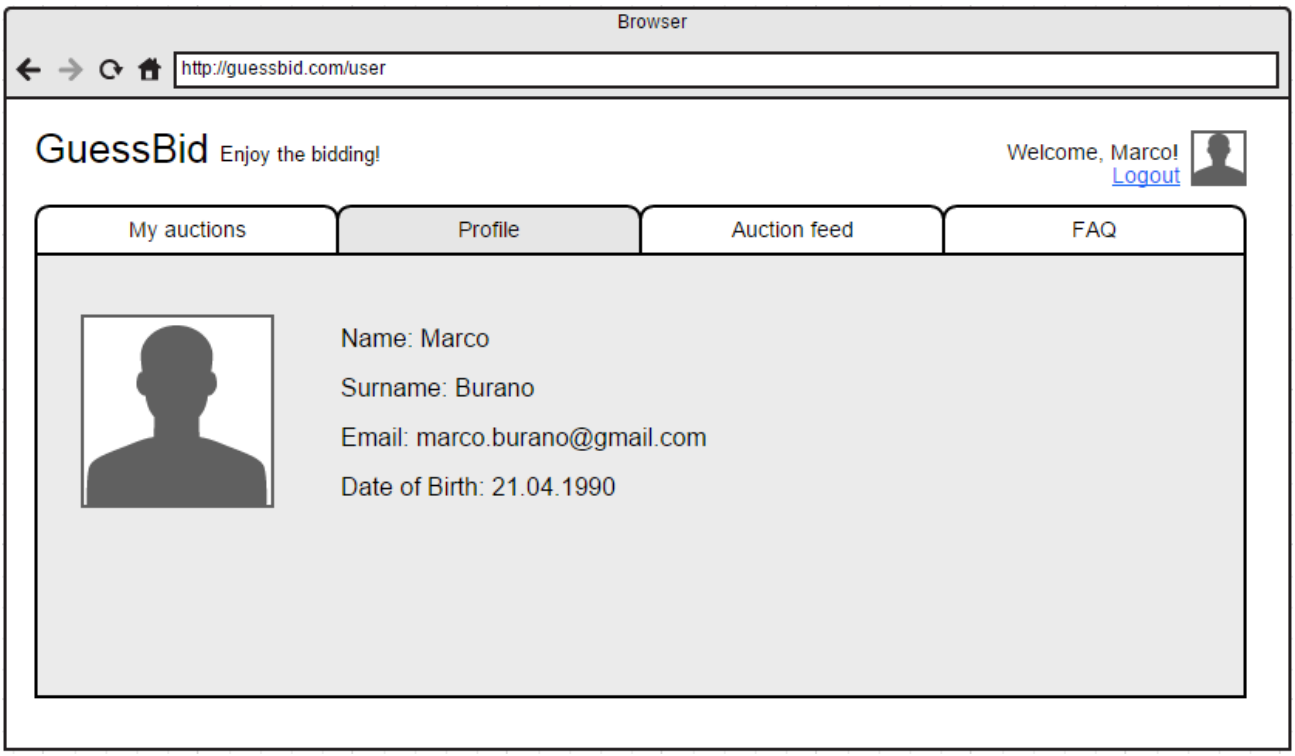
2.1.2 User Interfaces



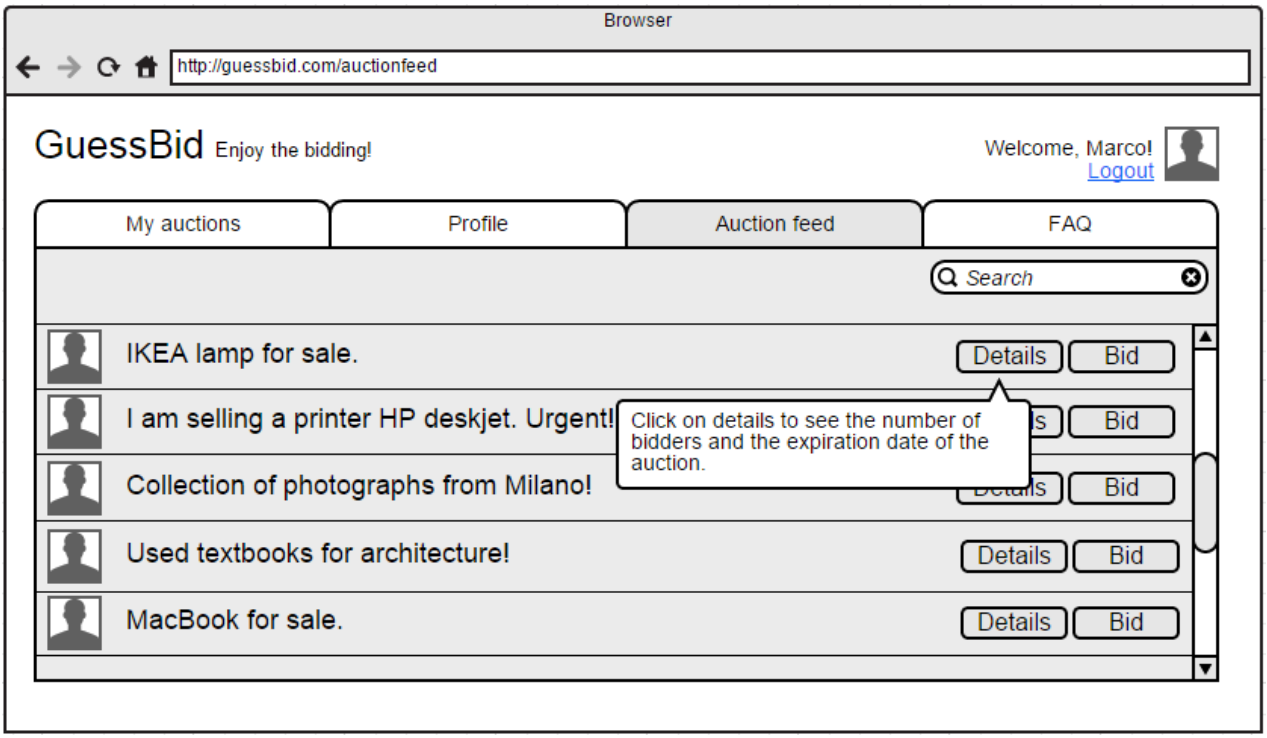
User Interface 1: Login



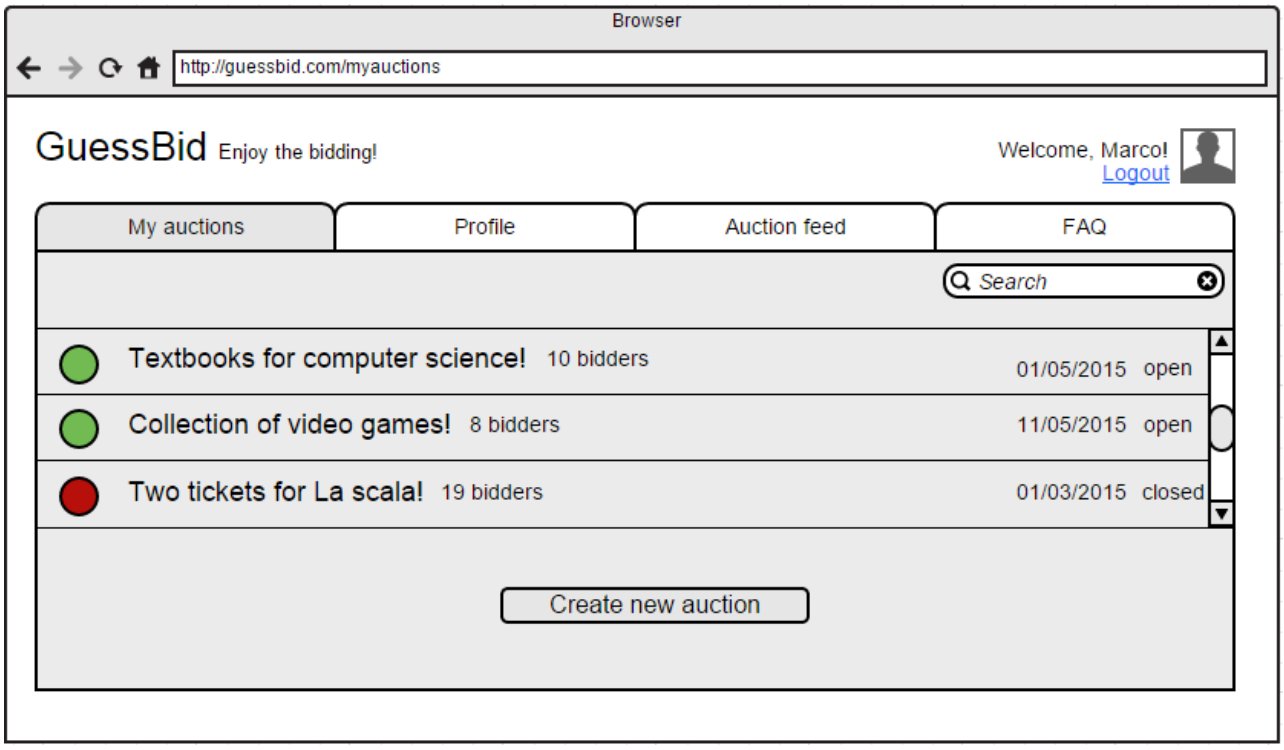
User Interface 2: Register form



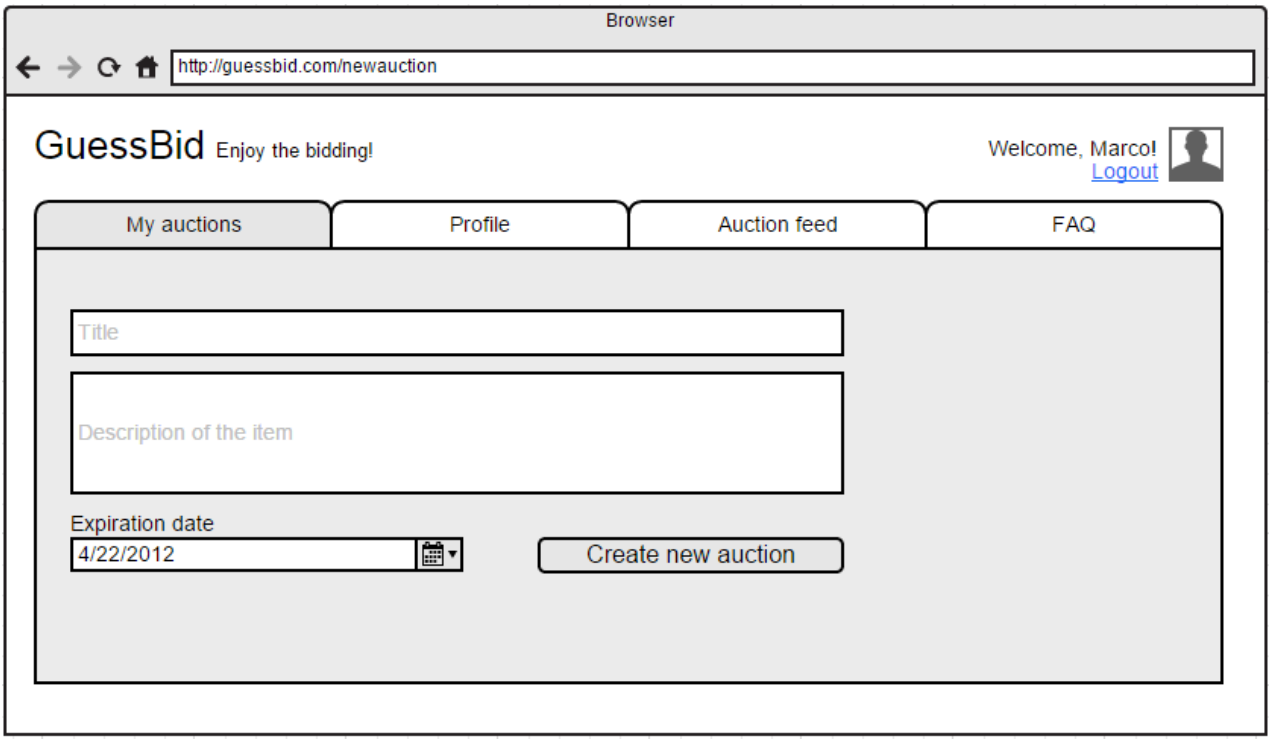
User Interface 3: User profile



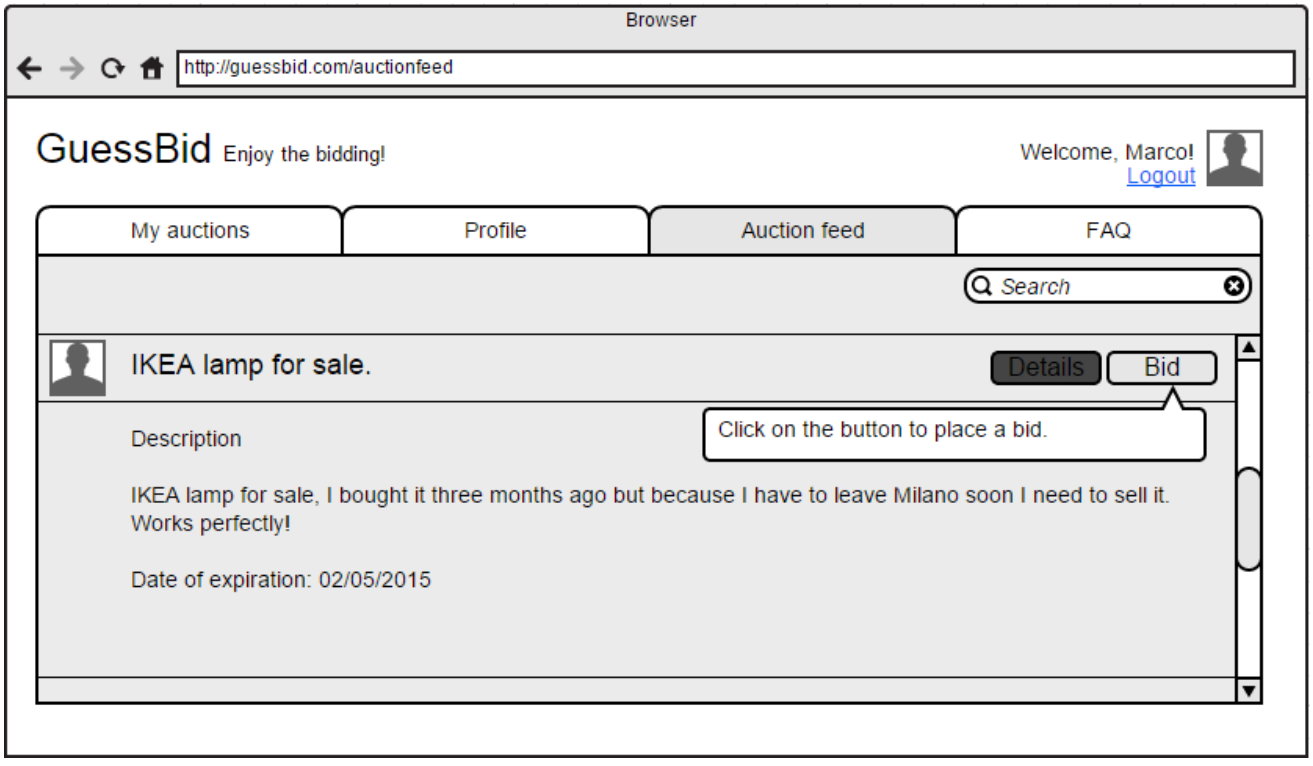
User Interface 4: Browse auctions in the auction feed



User Interface 5: Auctions created by the user



User Interface 6: Creation of a new auction



User Interface 7: Auction page

User Interface 8: Place a bid on an open auction

2.1.3 Hardware Interfaces

The software product does not provide any hardware interfaces.

2.1.4 Software Interfaces

- *Database Management System*

  Name: MySQL

  Mnemonic: MySQL

  Specification number: Community Server

  Version number: 5.6.21

  Source: http://dev.mysql.com/downloads/mysql/

- *Application Server*

  Name: GlassFish

  Mnemonic: GlassFish

  Version number: 4.1

  Source: https://glassfish.java.net/download.html

- *Operating System*

  The software product will run on any operating system which supports Java Virtual Machine, Database Management System and Application Server described above.

2.1.5 Communication Interfaces

Protocol	Port	Service
TCP	80	TCP
TCP	3306	MySQL (only if it is in a different physical server)

Table 3: Communication Interfaces

It is important to note that for the development of the first (current) version of the software we will assume that the Database Management System and Application Server reside on the same physical server.

2.1.6 Memory

The minimum memory requirements are:

- Primary memory: 2GB+
- Secondary memory: 40GB+

Note that the secondary memory is recommended to be physically on a different server from where the software product is installed, as it can exponentially grow without affecting system's performance. For the current production we assume that they are at least installed on the same server.

2.1.7 Operations

A user can interact with the system as a functional user. The functional operations for all the users are described in the product functions section.

2.1.8 Site Adaptation Requirements

The product software requires the following conditions to be satisfied in order to run successfully:

- Java Virtual Machine
- Application Server
- Database Management System

- Primary memory required space
- Secondary memory required space

Furthermore, users are required to have installed any of the following web browsers: IE 7.0+, FF 10+, Chrome 20+ or Safari 5+.

2.2 PRODUCT FUNCTIONS

This subsection describes a summary of major functions, functional and non-functional requirements of the software product.

2.2.1 Functional Requirements

The functional requirements are divided in three principal topics:

- *Manage users*
- *Manage auctions*
- *Manage bids*

The functional and non-functional requirements are defined and explained detailed in the following subsections.

2.2.1.1 Manage users

- Functional Requirements

  [FR1] Register to the system.

  [FR2] Login/Logout.

  [FR3] Consult auctions

  [FR4] Consult auction information

  [FR5] Bid on an auction

  [FR6] Receive notification about the status of the biddings
- Non-functional Requirements

  [NFR1] The user password must be stored securely.

  [NFR2] The system must manage high number of users.

2.2.1.2 Manage auctions

- Functional Requirements

[FR7] The user can create an auction.

[FR8] Auction should have a date of creation.

[FR9] Auction should have a date of expiration.

- Non-functional Requirements

  [NFR3] A user can manage only one auction at a time.

  [NFR4] A user can bid on multiple auctions.

  [NFR5] User cannot bid on his/her own auction.

2.2.1.3 Manage bids

- Functional Requirements

  [FR12] Users can issue bids on open auctions.

- Non-functional Requirements

  [NFR6] Issued bids date should be before the closing date of the auction.

2.3 User Characteristics

The user characteristics are:

- Knowledge in using a browser

2.4 Constraints

The following constraints apply to the software product:

2.4.1 Regulatory Policies

The project does not have any regulatory policies.

2.4.2 Hardware Limitations

The project does not have any hardware limitations.

2.4.3 Interfaces to other applications

The project does not have any interface with other applications.

2.4.4 Parallel operation

The software product must control different levels of concurrency, for instance concurrent users; creation and management of the auctions etc. It is necessary the management of parallel operations at the application level, and database level. These operations are important for the

general functionality of the product and they will be resolved in the design planning phase of the project.

2.4.5 Audit Function

The software does not perform any audit.

2.4.6 Control Functions

The software product does not control any device or any other system.

2.4.7 Higher-order language requirements

The software product requires basic knowledge of HTML, Java and JEE technologies.

2.4.8 Signal handshake protocol

The software product does not manage any handshake protocol.

2.4.9 Reliability requirements

The software product does not require any specific requirements to perform and maintain its functions under normal operation.

2.4.10 Criticality of the application

The software product requires proper support for concurrent users.

2.4.11 Safety and security considerations

The software product does not require any safety and security considerations.

2.5 Assumptions and Dependencies

The requirements in this document are grounded on the following assumptions:

- The Java virtual machine is already installed on the operating system.
- Users have access to a decent Internet connection.
- The software product supports any number of users.

2.6 Apportioning of requirements

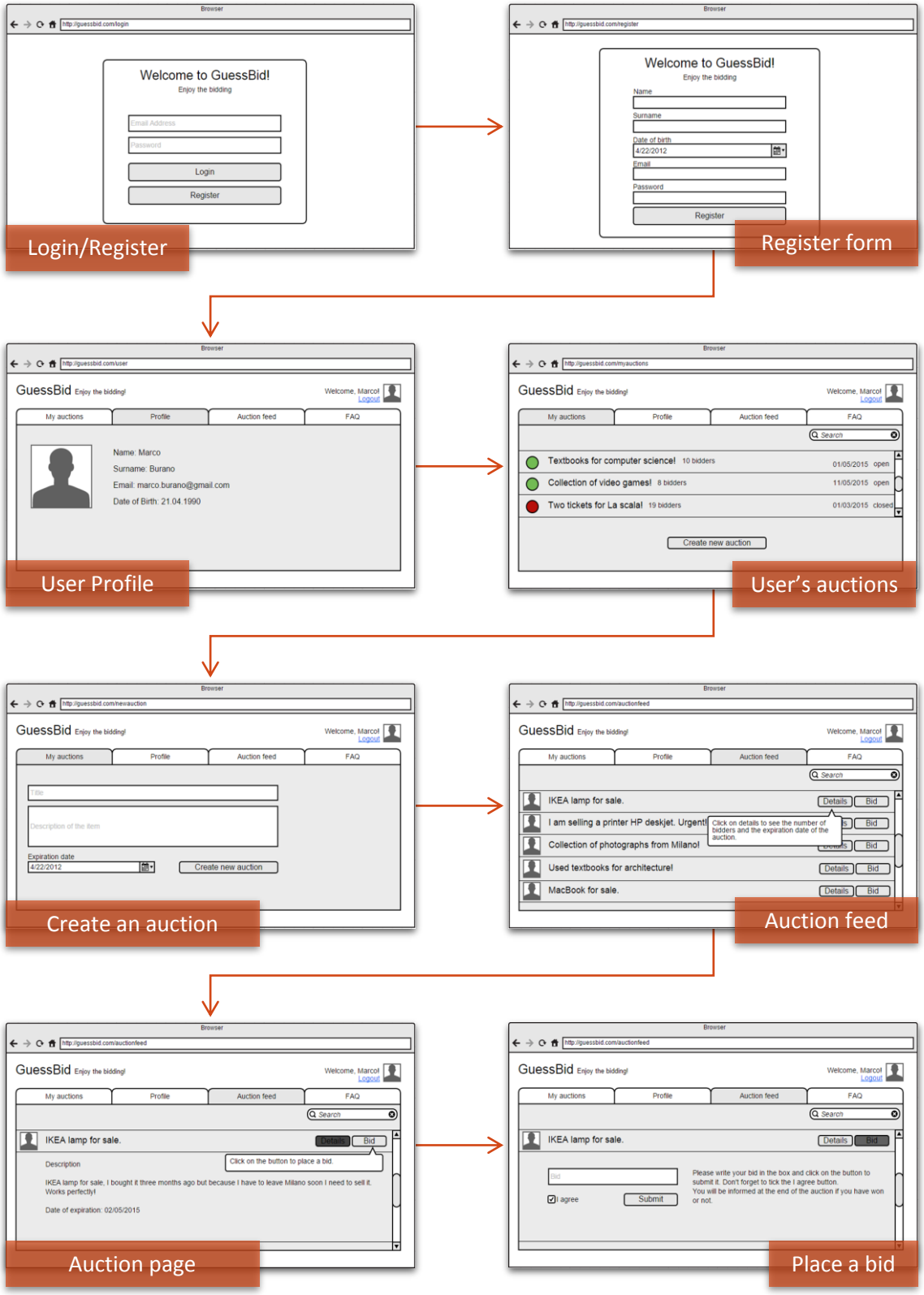
Future releases of the software product may provide support for:

1. Password modifying and resetting.
2. One user can create more auctions at the same time.

3 SPECIFIC REQUIREMENTS

3.1 EXTERNAL INTERFACES

3.1.1 User Interfaces



Storyboard: User navigation through GuessBid.

3.1.2 Hardware Interfaces

The system does not have hardware interfaces.

3.1.3 Software Interfaces

The system does not have software interfaces.

3.1.4 Communication Interfaces

The system does not have communication interfaces.

3.2 FUNCTIONAL REQUIREMENTS

3.2.1 Scenarios

User registers to the system	
Code	SC-001
Description	Describe how a user registers to the system
Goal	**[G1]** Users can register on the system.
Assumptions	The user is not yet registered on GuessBid
Scenario	
Lara is a student who would like to use a system for auctions. One of her friends told her about GuessBid, so she decides to use it. Lara connects to the internet and opens up a browser installed on her computer. She goes to the internet address of the system and when the first page loads, Lara notices that there is a register option. After clicking the "Register" button, the register form opens up where she writes her name, surname, email, password and other information. When she finishes filling out the form, she clicks on the button "Confirm", after which she receives a message stating that the registration has been successful. Lara is now registered on the system.	

Scenario 1: User registers to the system

User logs in on the system	
Code	SC-002
Description	Describes how a user logs in the system
Goal	**[G1]** Users can register on the system..
Assumptions	The user is registered but not logged in GuessBid.
Scenario	
Victor is Lara's friend who told her about the GuessBid system. He has been using the system for a while now and he is already registered on it. Today Luca told him that there are some really cool stuff for sale on the system so he decides to check them out and maybe bid on some of the auctions. He connects to the internet and go to the internet address of the system using a browser installed on his computer. When the system loads the first page, Victor notices the form for Login, he writes his username and password and clicks on the "Login" button. He is redirected to his profile on the system where he can see all the active auctions.	

Scenario 2: User logs in the system.

User logs out from the system	
Code	SC-003
Description	Describes how a user logs out the system
Goal	**[G1]** Users can register on the system.
Assumptions	The user is registered and logged in GuessBid.
Scenario	
Victor has been checking some auctions on the GuessBid system. He is already logged in on the system. Lara comes along and asks him if she can use his computer in order to log in on GuessBid To do that, Victor has to log out. Victor navigates to the "Logout" button in the upper right corner of the page and he clicks on it. The system loads the first page again with the Login form, so Lara can login with her username.	

Scenario 3: User logs out from the system.

User creates an auction	
Code	SC-004
Description	Describes how a user creates an auction
Goal	**[G2]** Users can create an auction.
Assumptions	The user is registered and logged in GuessBid. The user has not created any other auction at this time.
Scenario	
Lara has some lamp to sell so she decides to create an auction on the GuessBid system and sell the lamp to the winner. She clicks on the create auction button. After clicking it, a form appears where she needs to enter all the information about the auction: expire date, description of the item. After finishing filling out the form, she clicks on the "Confirm" button bellow the form. By creating this, the auction now is open for bidding. The page is redirected to the auction view where she can see all the necessary information concerning the auction.	

Scenario 4: User creates an auction.

User consults an auction	
Code	SC-005
Description	Describes how a user consults an auction
Goal	**[G3]** Users can browse auctions.
Assumptions	The user is registered and logged in GuessBid.
Scenario	
Maria wants to buy some stuff for her apartment. Since she knows that on GuessBid there are a lot of good things for good prices, she decides to go on the system and check some of the auctions. She opens one of the auctions and she sees the information on the auction page.	

Scenario 5: User consults an auction.

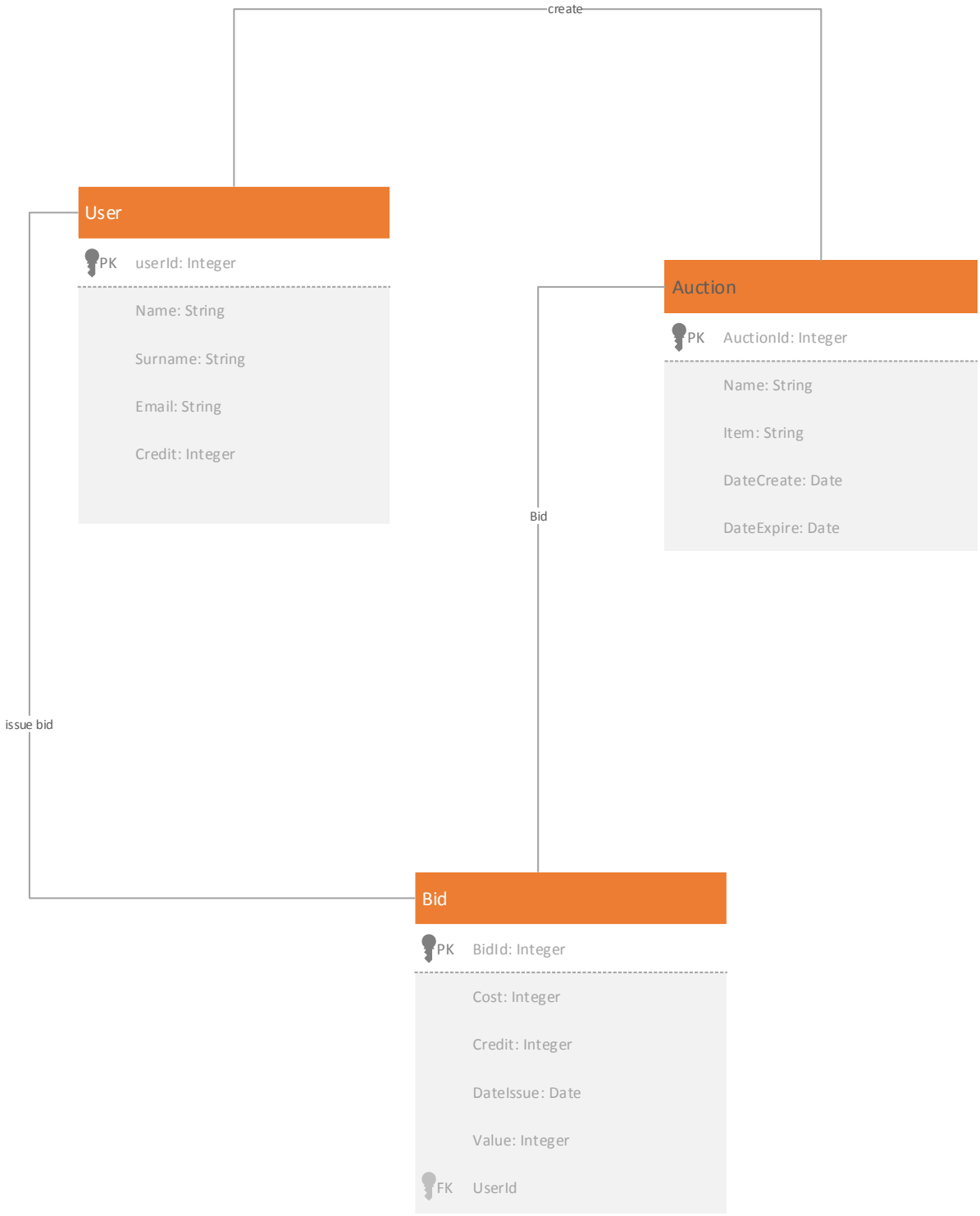
User can bid on an auction	
Code	SC-006
Description	Describes how a user can bid on auctions.
Goal	**[G4]** Users can bid on existing auctions. **[G5]** Users can provide more than one bid before closing time of the auction.
Assumptions	The user is registered and logged in GuessBid. There is at least one open auction.
Scenario	
Luka logs in to GuessBid system and browses the open auctions. He likes one of the items offered so he decides to place a bid. He clicks on add a bid button and places his offer. He receives confirmation message that the bid is acknowledged and after a message on the status of the bid. He gets a message that his bid is not the lowest unique bid so he decides to place another bid to this auction.	

Scenario 6: User places a bid.

User is notified about the status of his/her biddings	
Code	SC-007
Description	Describes how a user can bid on auctions.
Goal	**[[G8]** Users are informed by the system about the current status of his/her biddings on the open auctions at each bid.
Assumptions	The user is registered and logged in GuessBid. There is at least one open auction.
Scenario	
Luka logs in to GuessBid system and browses the open auctions. He likes one of the items offered so he decides to place a bid. After placing the bid he receives a notification about the status of his bid. Currently his bid is not the lowest unique bid for this auction. He then goes on another auction and places a bid. He gets a message that currently he is the lowest unique bid.	

Scenario 7: Notification about the status of the user's biddings.

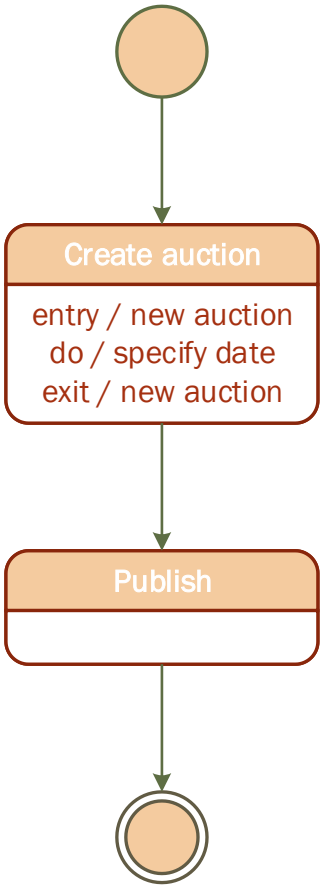
3.2.2 Analysis model



Class diagram 1: Analysis model

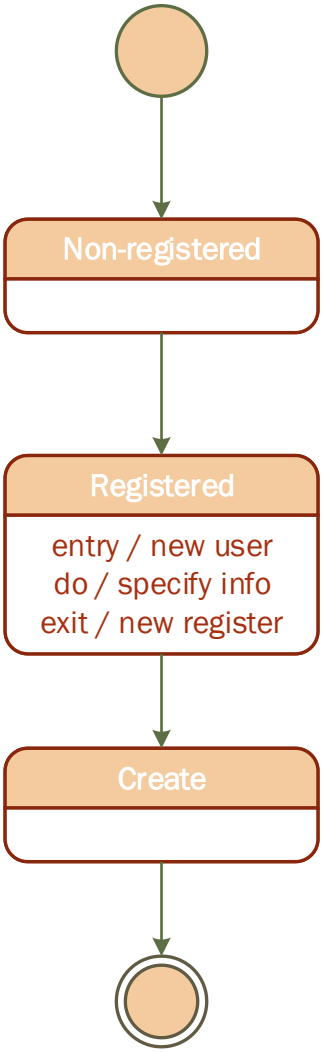
3.2.3 State chart model

3.2.3.1 State diagram of an auction



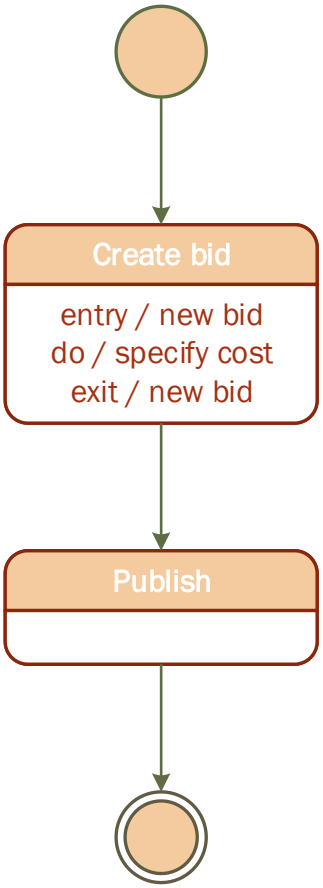
State diagram 1: Auction

3.2.3.2 State diagram of a user

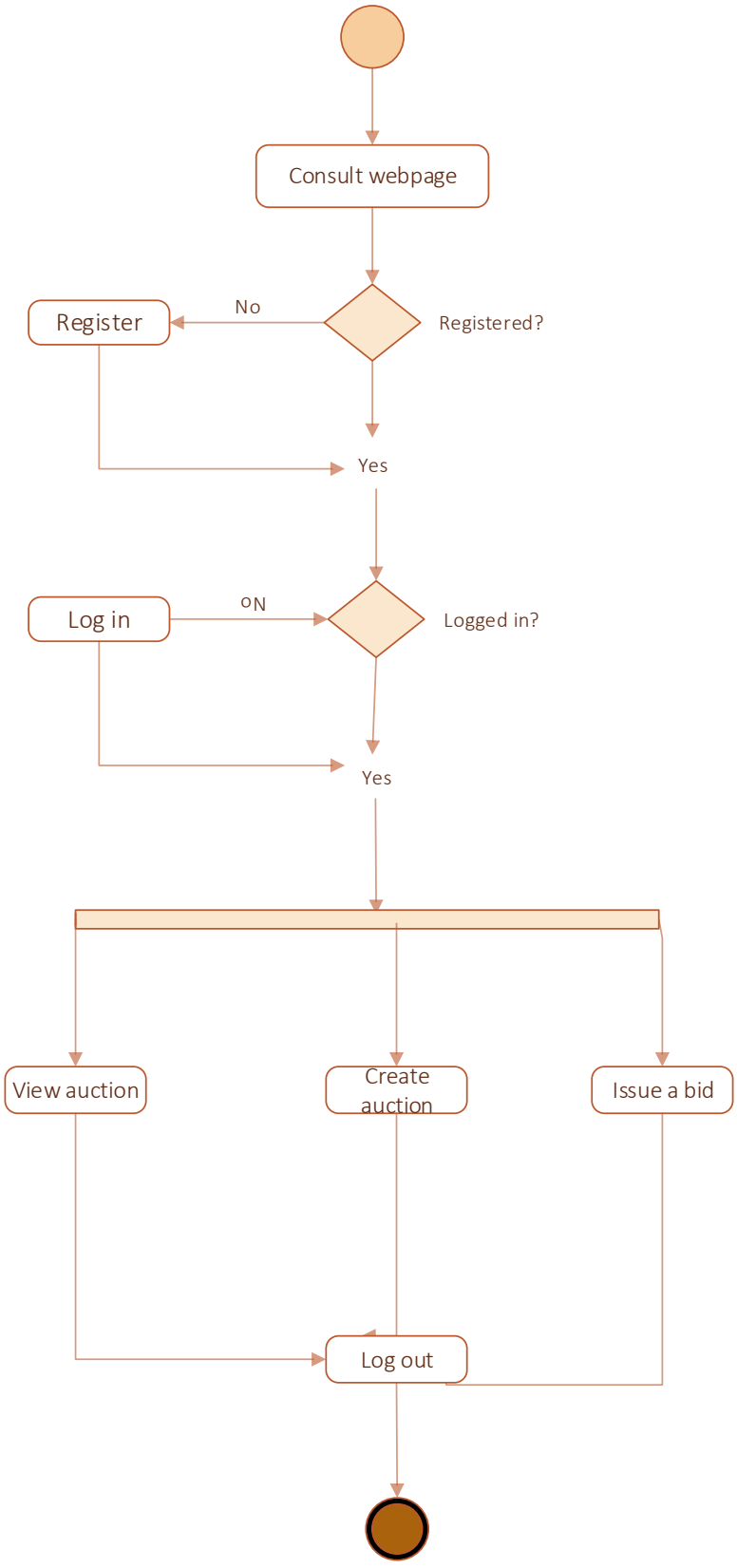


State diagram 2: User

3.2.3.3 State diagram of a bid



3.2.4 Activity Model

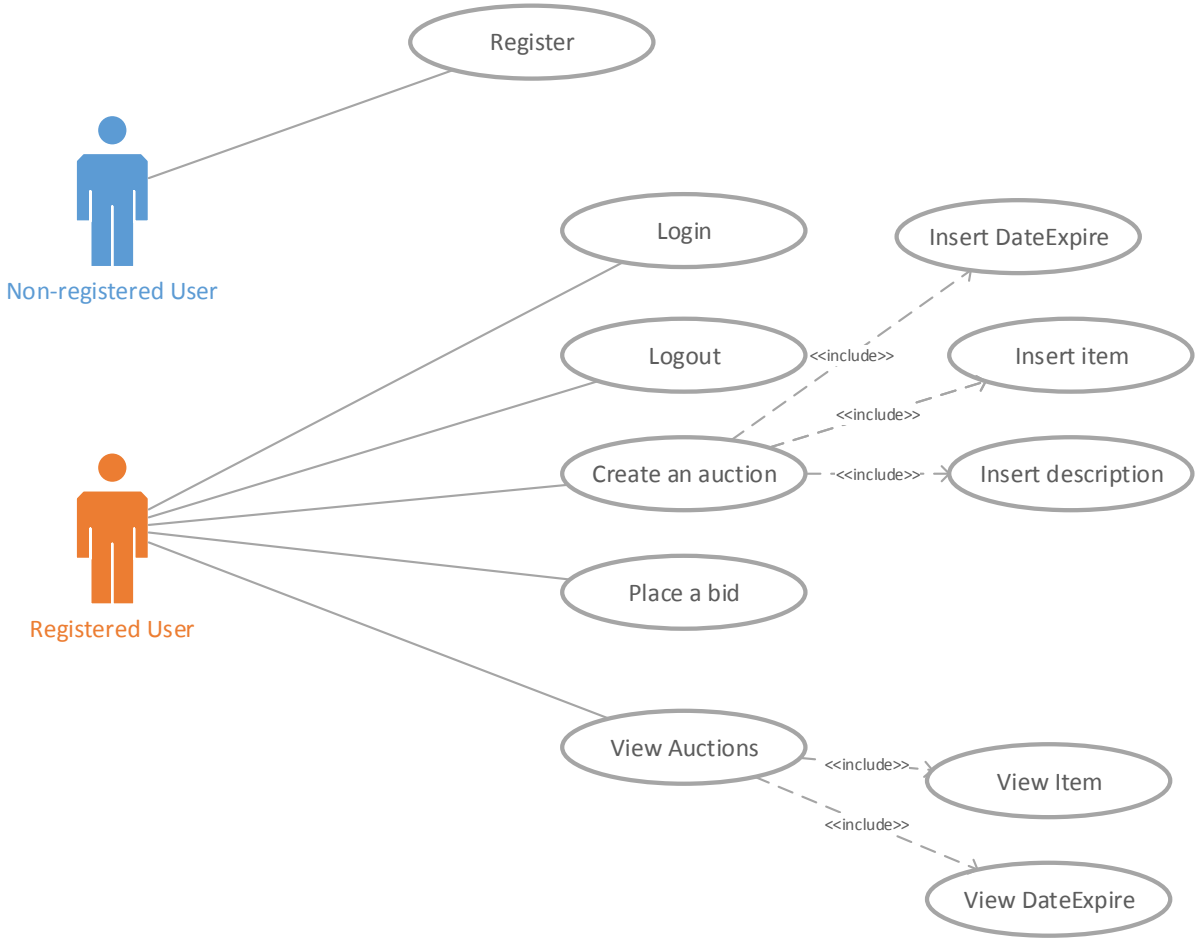


Activity diagram 1: Actions of the user

3.2.5 Use cases

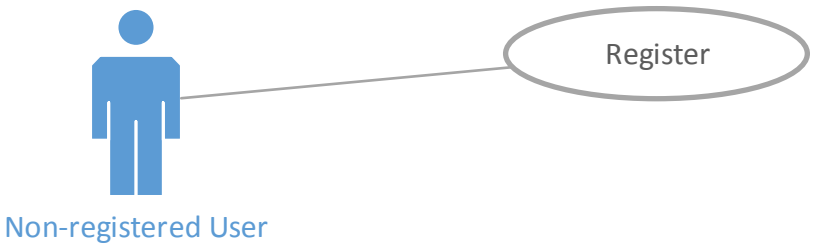
The software contains two main actors:

- Non-registered user
- Registered user



Use Case Diagram 1: General use case model.

3.2.5.1 Non-registered User



Use Case Diagram 2: Non-registered user

- **Registering in the system**
  Code: USC-001

  Description: Registering in the system

  Goal: [G1]

  Actors: Non-registered user

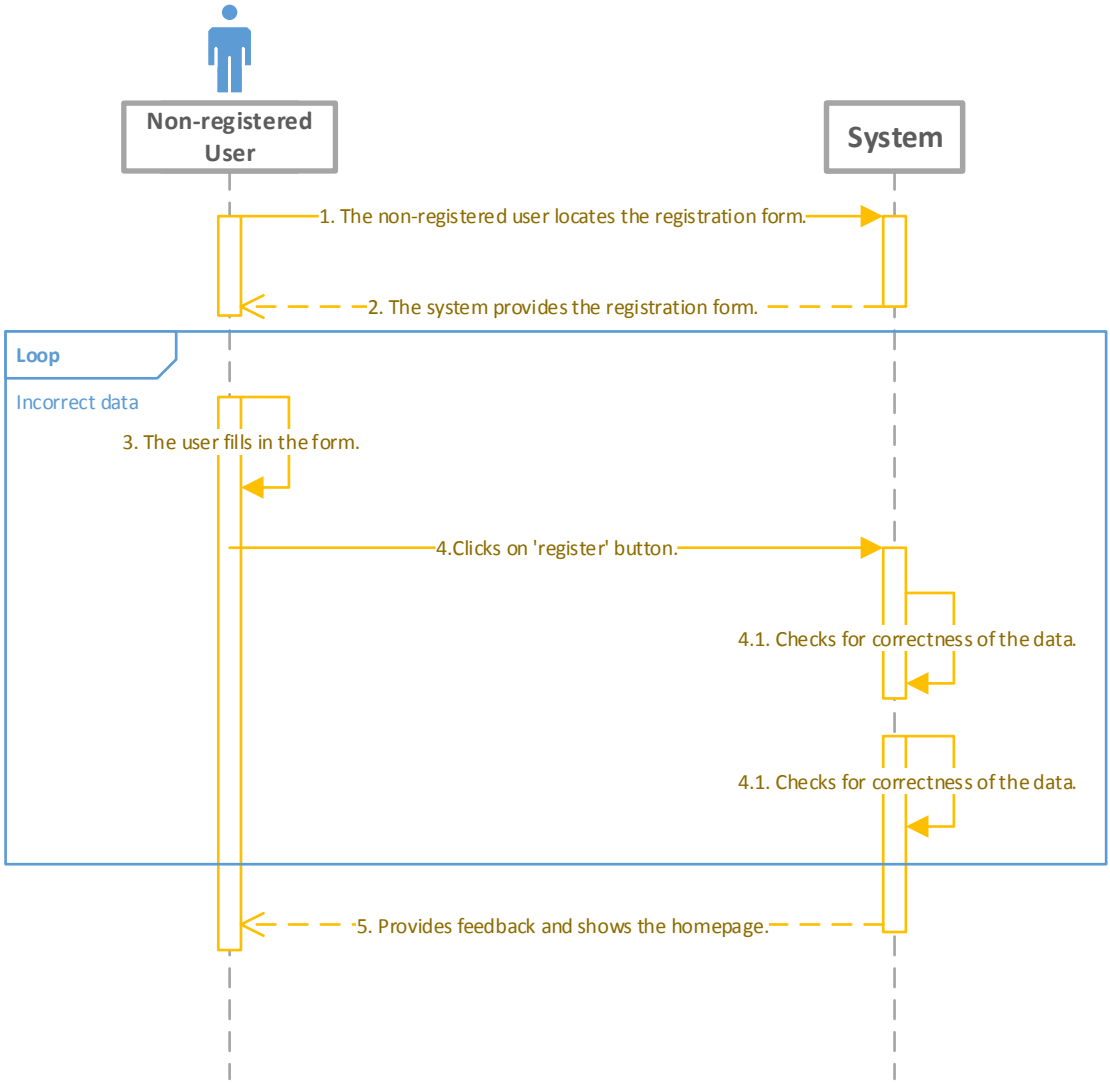
  Entry condition: Non-registered user navigates to the homepage of GuessBid

  Exit condition: Non-registered user is registered.

  Flow of events:

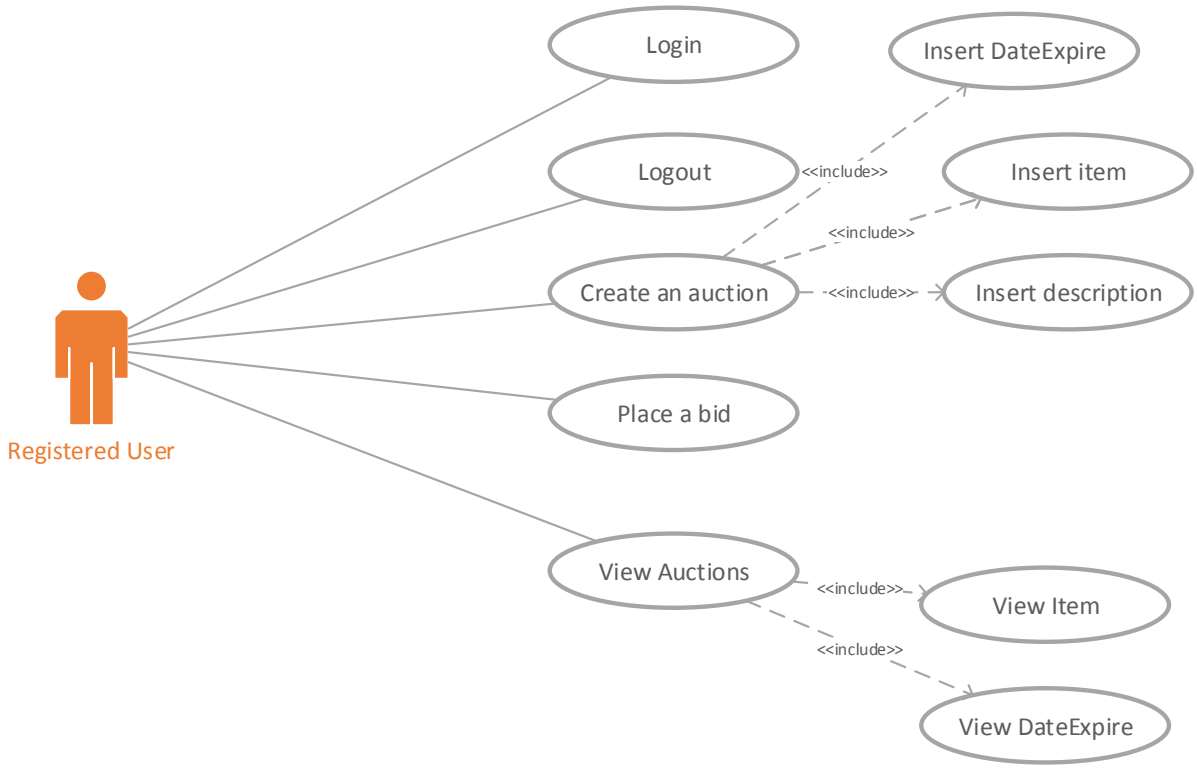
  1. Non-registered user locates the registration form
  2. The system provides the registration form
  3. Non-registered user fills in the form
  4. Non-registered-user clicks “Register” button
  5. The system provides feedback and shows the homepage

  Exceptions: the user adds incorrect data



Sequence Diagram 1: Registering in the system. (USC-001)

3.2.5.2 *Registered User*



Use Case Diagram 3: Registered user

- **View auctions**
  Code: USC-002

  Description: View auctions.

  Goal: [G3]

  Actors: Registered user

  Entry condition: Registered user navigates to the homepage of GuessBid

  Exit condition: View an auction.

  Flow of events:

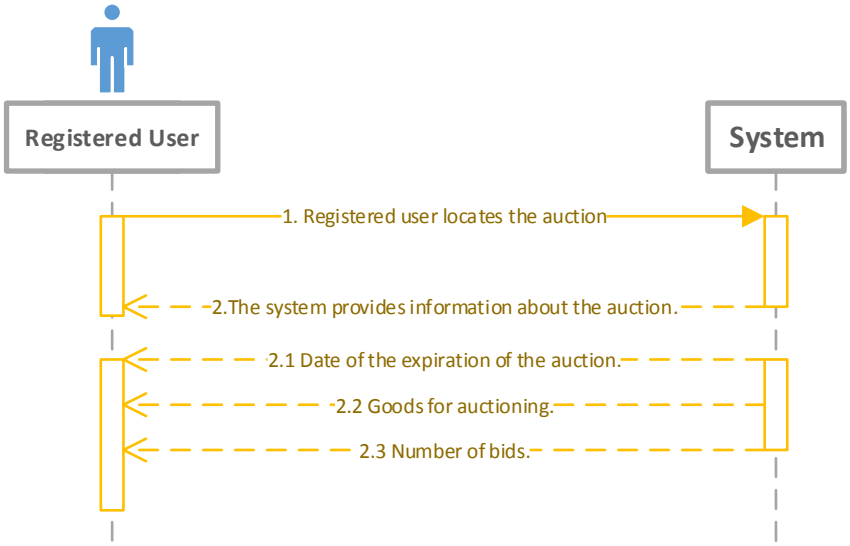
  1. Registered user locates the auction.

  2. The system provides information about the auction

  2.1. The system provides the expiration date.

  2.2. The system provides the goods for auctioning.

  2.3. The system provides the number of bidders.



Sequence Diagram 2: View auctions. (USC-002)

- **Registered user login**

  Code: USC-003

  Description: Registered user login.

  Goal: [G10]

  Actor: Registered user

  Entry condition: Registered user navigates to the homepage of GuessBid.

  Exit condition: User is logged in.

  Flow of events:

  1. The registered user locates the login form.

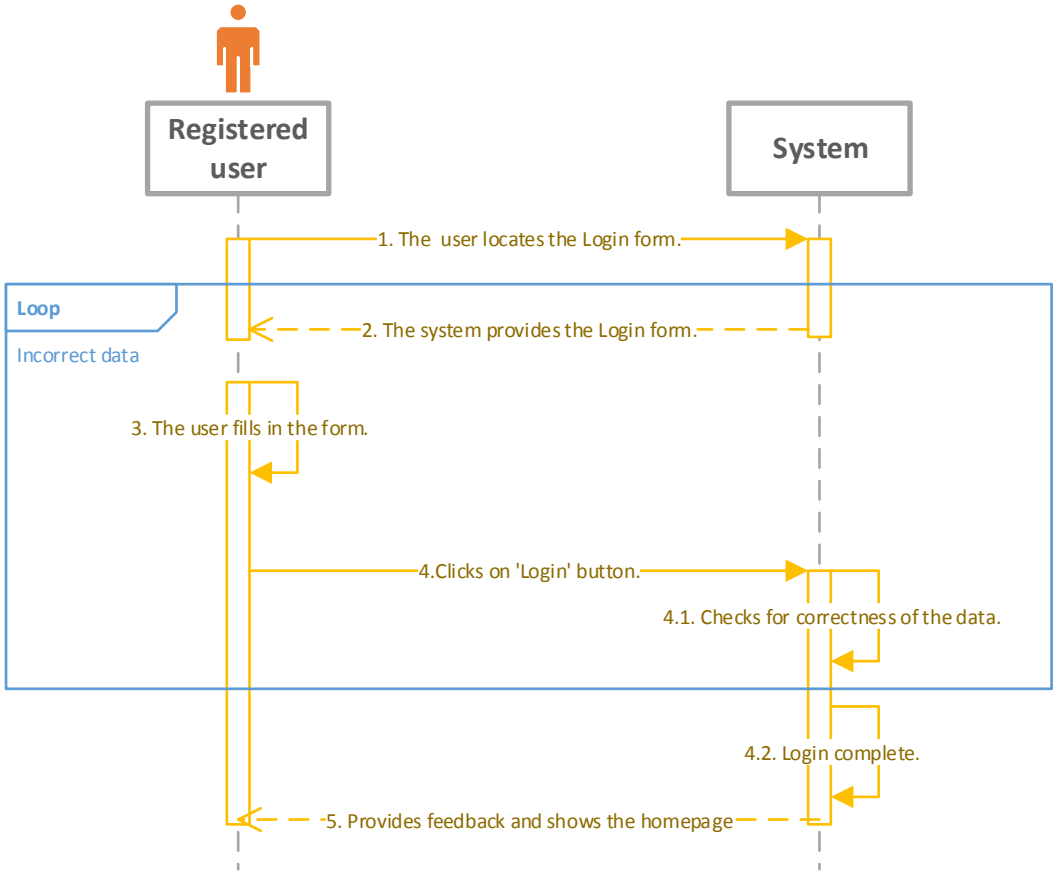
  2. The system provides the login form.

  3. The registered user fills in the form.

  4. The registered user clicks “Login” button.

  5. The system provides feedback and shows the homepage of GuessBid.

  Exceptions: the user adds incorrect data.



Sequence Diagram 3: Login of already registered user. (USC-003)

- **Registered user logs out from the system**

  Code: USC-004

  Description: Registered user logout.

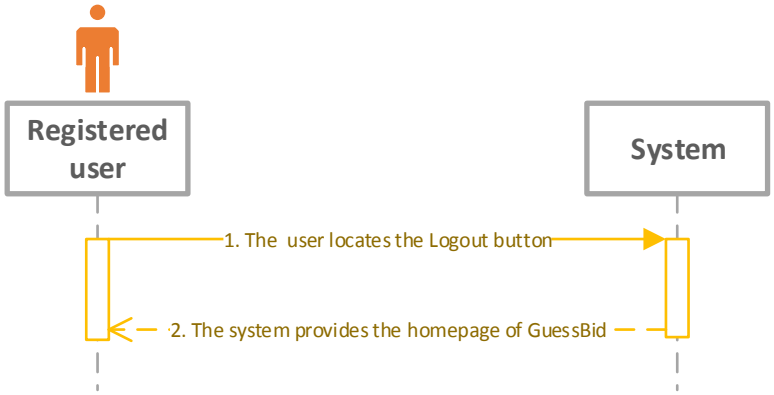
  Goal: [G10]

  Entry condition: The user is logged in.

  Exit condition: The user is logged out.

  Flow of events:

  1. The registered user clicks on the logout button.
  2. The system provides the homepage for non-registered users.



Sequence Diagram 4: Registered User logout (USC-004)

- **User creates a new auction**

  Code: USC-005

  Description: User creates a new auction.

  Goal: [G2]

  Entry condition: User is logged in and there is not an open auction created by him/her at the moment.

  Exit condition: A new auction is created.

  Flow of events:

  There is no ongoing auction created by that user at that time.

  1. The user clicks the button “New auction”.

  2. The system provides the form for a new auction.

  3. The user enters the information about the auction (title, description, expiration date).

  4. Saves the form.

  5. The user submits the auction

  5.1. The system checks for correctness of the data.

  6. Confirmation that the auction is created.

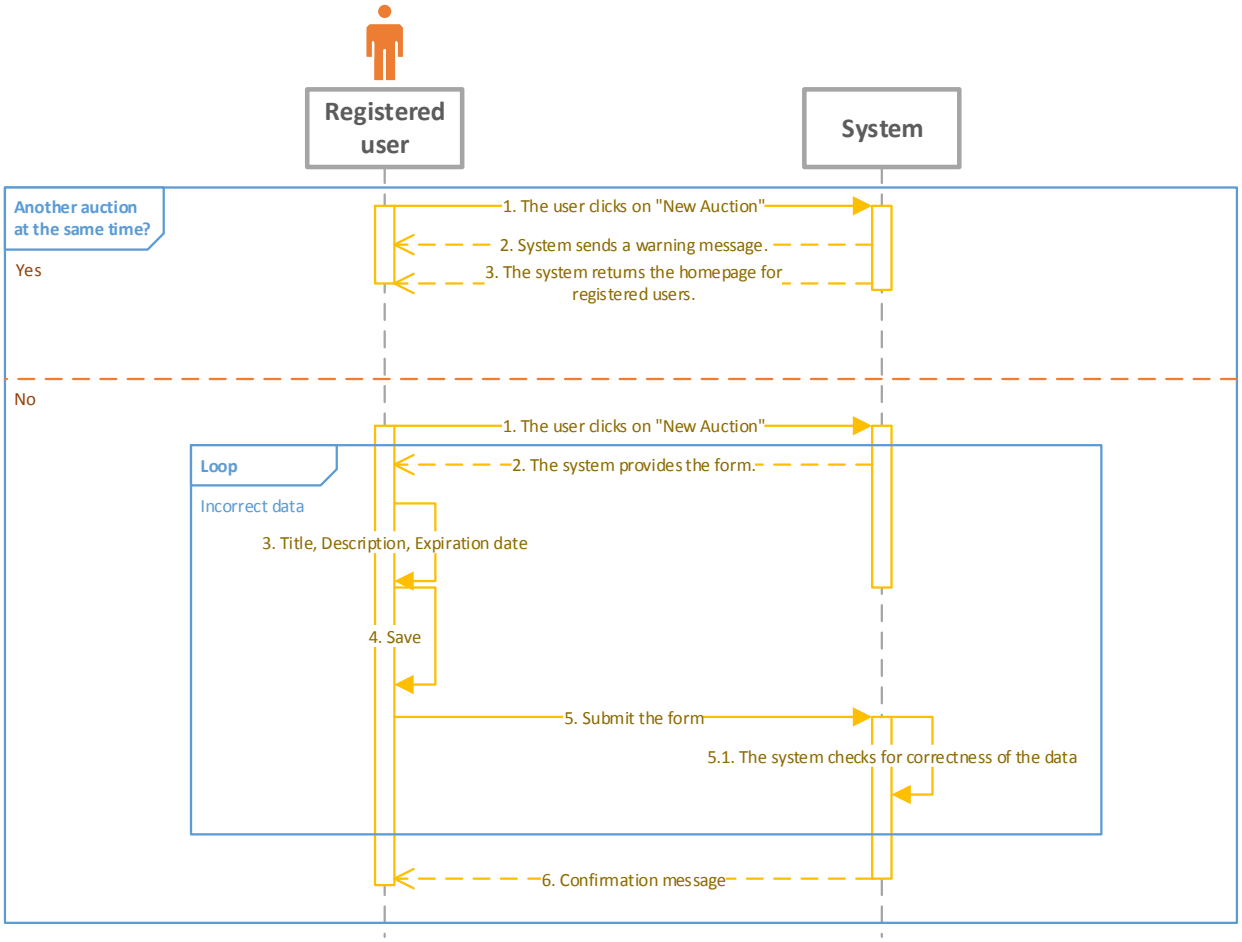
  The user already organizes an auction at the same period.

  1. The user clicks the button “New auction”.

  2. The system replies a warning message: “You have an ongoing open auction at the moment”.

  3. The system provides the homepage for registered users.

  Exceptions: wrong user data.



Sequence Diagram 5: Creation of a new auction (USC-005)

- **User places a bid on an auction**

Code: USC-006

Description: User places a bid on an auction

Goal: [G4]

Entry condition: User is logged in and there is at least one open auction by another user.

Exit condition: A bid is placed on the auction.

Flow of events:

1. Registered user locates the auction.

2. The system provides the information about the auction.

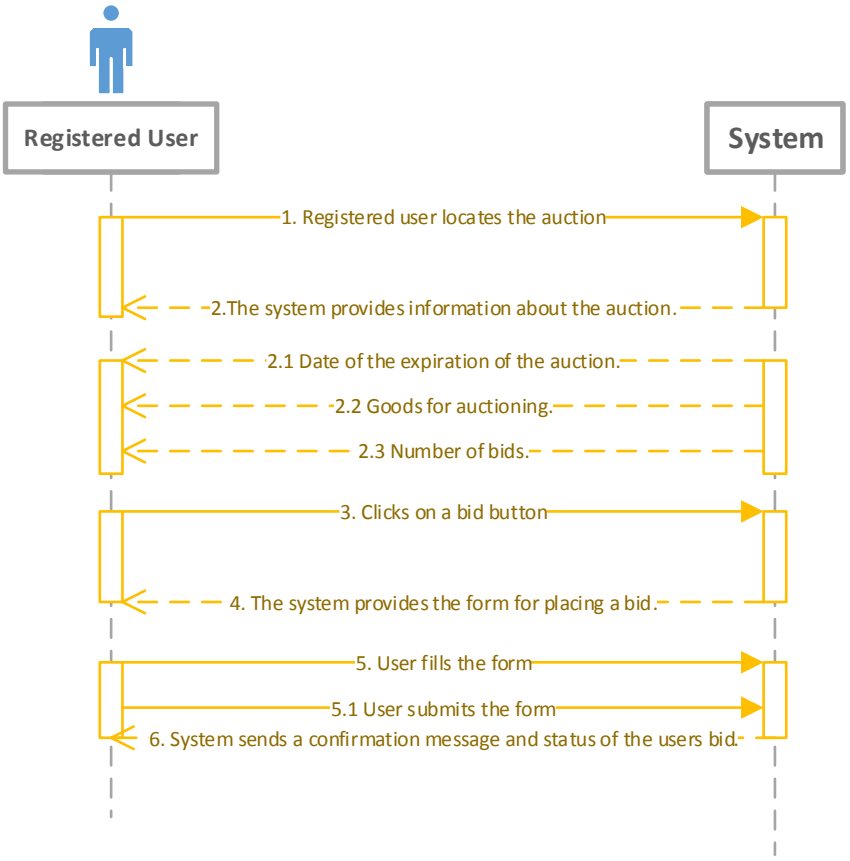
3. User clicks on the “Bid” button.

4. The system provides the form for a new bid.

5. The user fills the form.

5.1. The user submits the form.

6. System sends a confirmation message that the bid is placed and a status message about the user’s bid.



Sequence Diagram 6: Placing a bid on an existing auction. (USC-006)

3.3 Performance Requirements

The software product requires that each web page shall load in less than 15 seconds. Also, as mentioned before, it is necessary to support a high level of concurrent users.

3.4 Logical Database Requirements

The database must save all the entities identified in the analysis diagram, because they are necessary for managing the software product information.

3.5 Design Constraints

The software product must be designed and implemented in JEE and its related technologies.

3.6 Standard Compliance

The software product must be developed following recommended standards in order to be easily readable and updatable.

3.7 Software System Attributes

3.7.1 Reliability

The system shall assure the integrity of the users.

3.7.2 Availability

The system should be available 24 hours per day, 7 days per week, and 365 days per year.

3.7.3 Security

The software product must encrypt users' password. This can be achieved using cryptographic techniques.

3.7.4 Maintainability

The database of the software product should be backed up periodically, so that in case of failure existing data can be reconstructed easily. The system needs maintainability in clearing the users that are inactive for longer periods, server files system backup and periodical cleaning of old auctions, in order not to affect the performance of the system.

3.7.5 Portability

The software product can be installed on any operating system which supports Java Virtual Machine and its dependent components.

3.8 Other Requirements

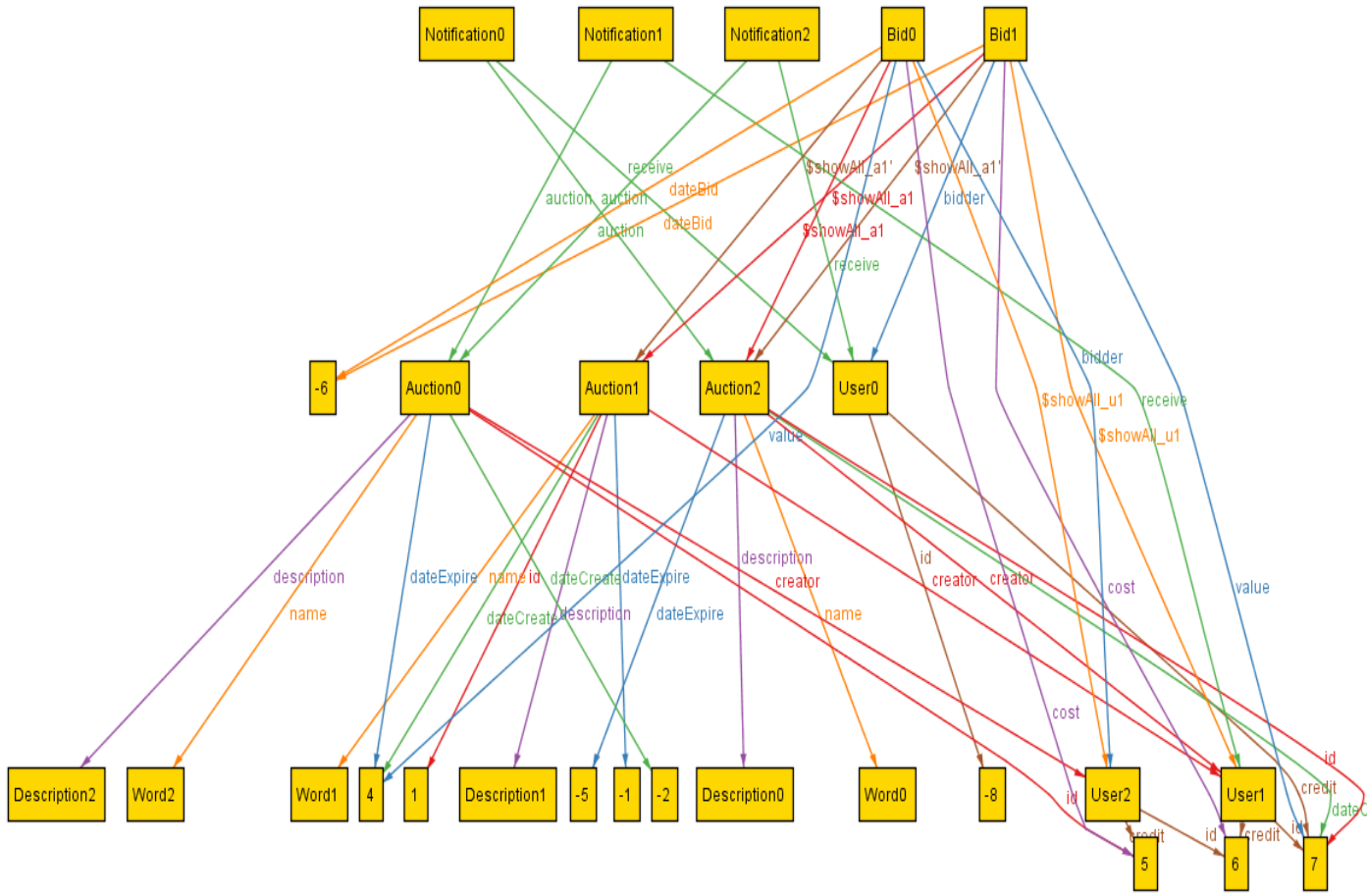
The software product must provide understandable messages in text form in the event of an error and instruct the user on what to do.

4 APPENDICES

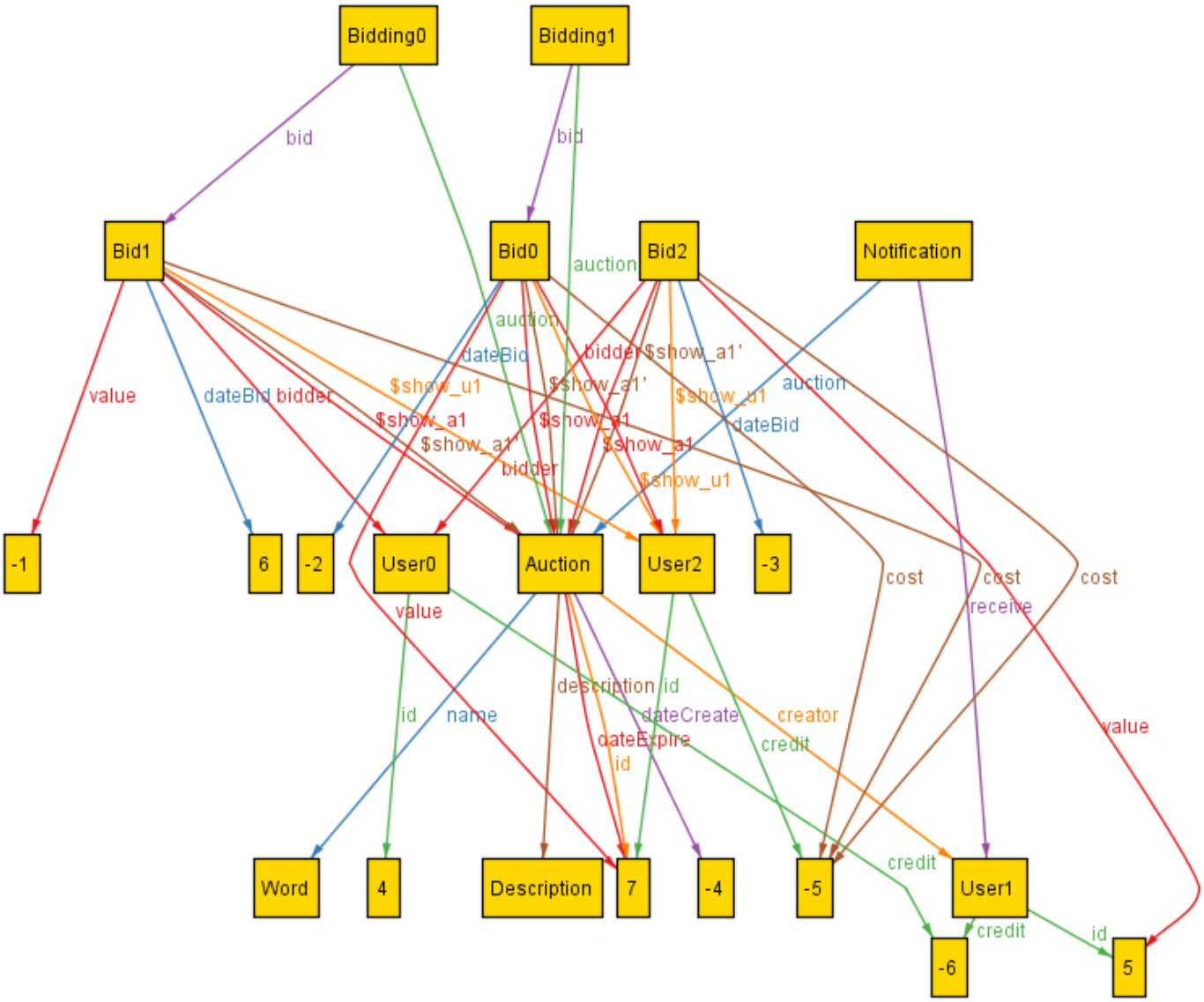
4.1 ALLOY MODEL

Alloy model file: guessBid.als

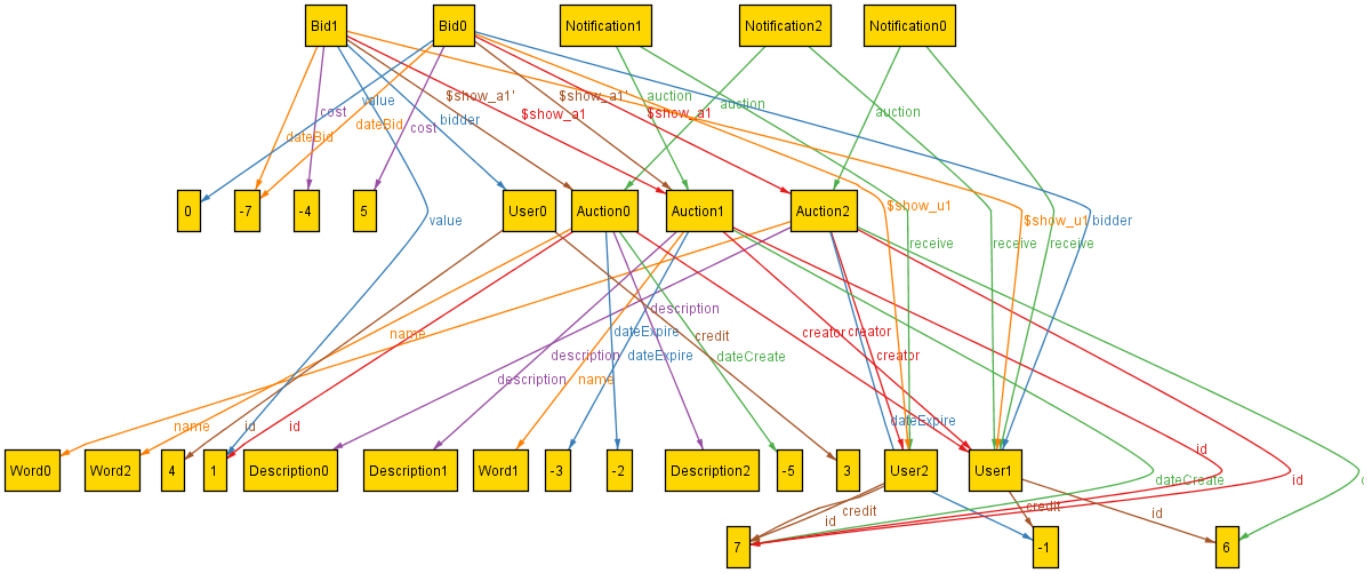
Following there are some screenshots from the alloy model.



Alloy instance 1: run showAll



Alloy instance 2: One auction, three users, and three bids.



Alloy instance 3: Three auctions, three users and two bids.