GuessBid

Design Document

Authors:

Telemtaev Ruslan

Kazakov Sergey

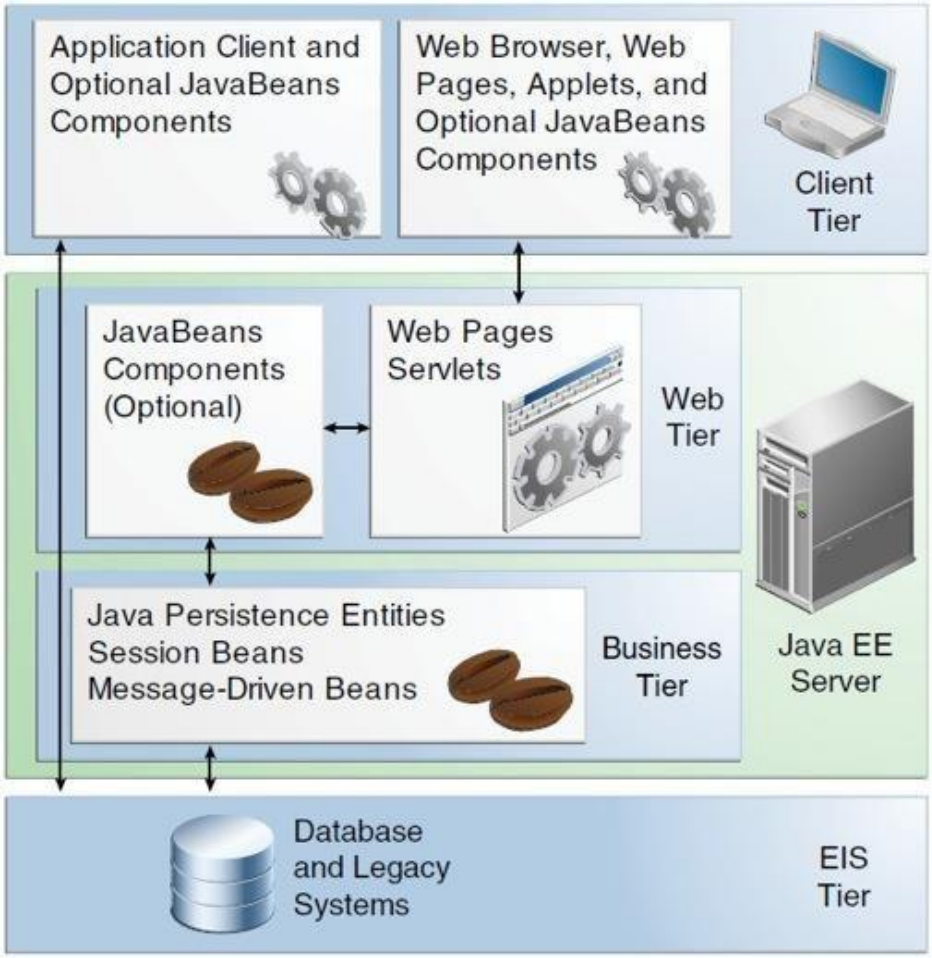
Contents

1. Architecture Description

1.1 Technology Architecture

The technology to be used for implementing GuessBid has been decided directly by the customer, and has been already considered as a constraint on the implementation in the RASD. The system musts run on the Java Enterprise Edition (JEE) platform, that has a structure that best fits the needs of a web application like GuessBid.

The general multitier architecture is shown in figure



JEE architecture divided in four different logical tier:

Client Tier: it contains Application Clients and Web Browsers and it is the layer that interacts directly with the actors. As our project will be a web application the client will use a web browser to access pages;

Web Tier: it contains the Servlets and Dynamic Web Pages that needs to be elaborated.
This tier receives the requests from the client tier and forwards the pieces of data collected to the business tier waiting for processed data to be sent to the client tier, eventually formatted;

Business Tier: it contains the Java Beans, that contain the business logic of the application, and Java Persistence Entities.

Enterprise Information System (EIS): it contains the data source. In our case it is the database allowed to store all the relevant data and to retrieve them.

1.2 Subsystems Individuation

In this chapter we decompose the whole GuessBid system in smaller subsystems, by using a top-down approach. A subsystem identifies one or more components of the system that are responsible of a single functionality, either explicitly offered to the user or implicitly used by other parts of the system. During this procedure we will follow common design principles. In particular our goal is to have a system with high cohesion, so our system will be easier to understand and change, and reduce coupling where possible.
Also in this way will be simple to test each subsystem and then test the whole system. Furthermore we could test easily the interactions between modules.
But before doing this division we need a remark on functionalities of our web-application.

1.2.1 Functionalities

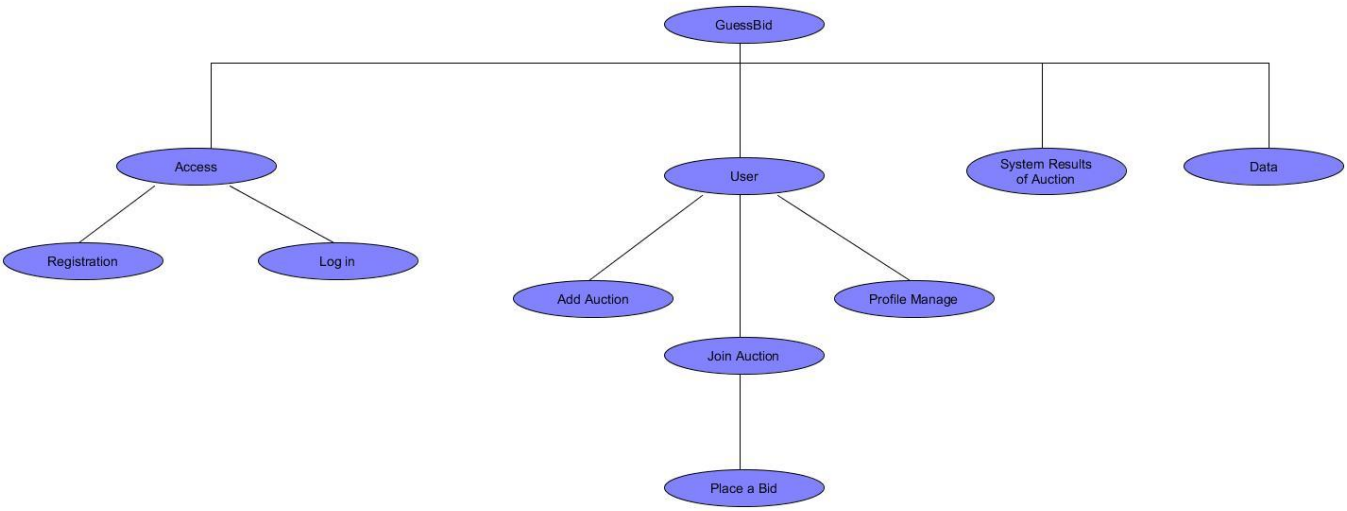
• Registration for a new user. A new user can sign up to the web application.

• Login functionality for registered user.

• A user can create his auction

• A user can find auction , join auction , and place a bid

• The system can send notification to the users.



1.2.2 Sub-System

GuessBid is divided in the following subsystems:

Access Sub-System is the module that allows a person to access the application.

- Registration Sub-System
- Log-In Sub-System

User Sub-System divided this module in other sub modules:

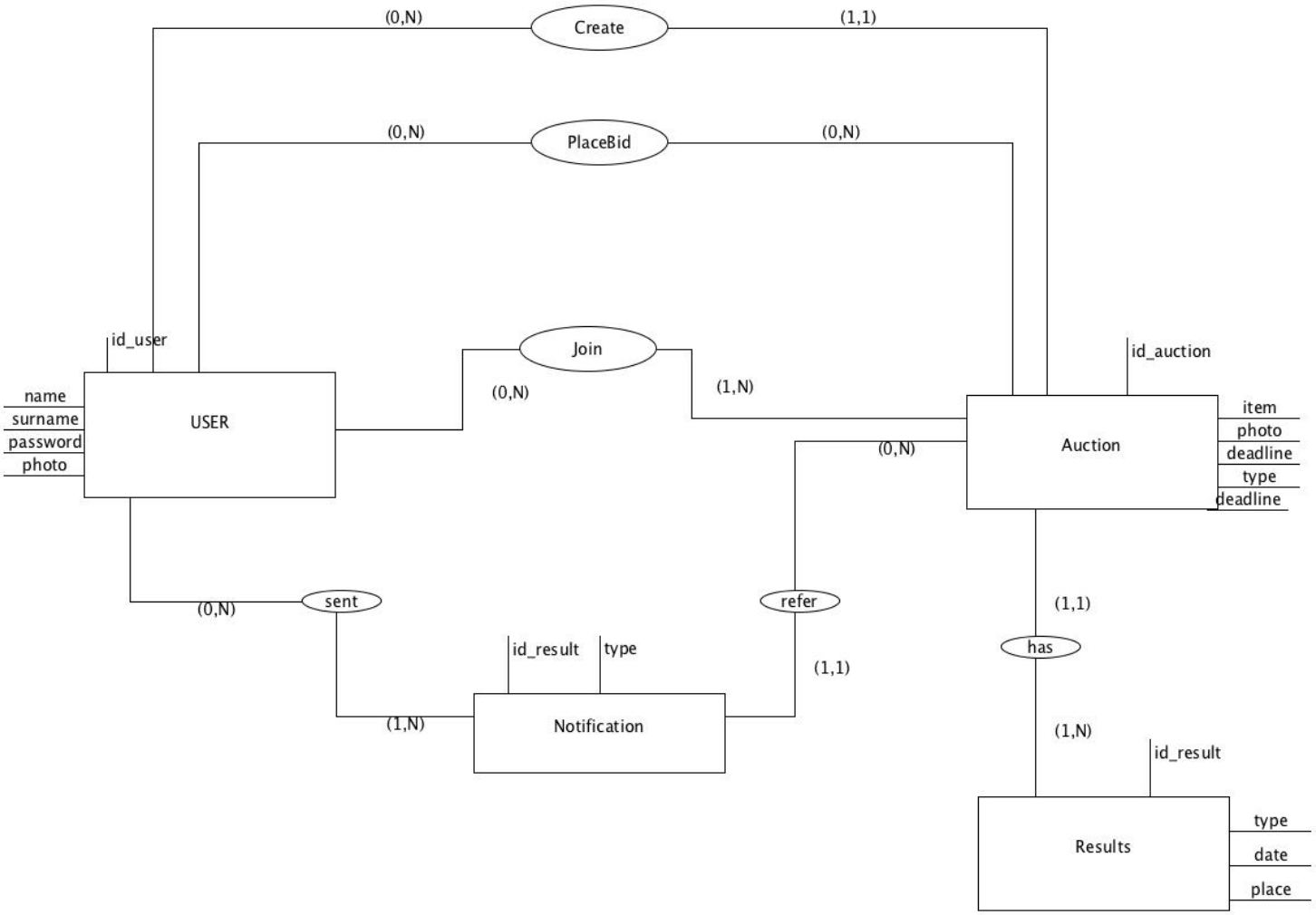
- Add Auction Sub-System
- Join Auction
    - Place a Bid
- Profile Management Sub-System

System Information Sub-System (will take results of auction).

DATA Sub-System

2. Data Management

In this chapter we will explain in detailed way entities, relations and, in general, the motivations of our design, analyze the information that are needed to be stored and organize them in a compact and functional schema.



3. Logical View

An Auction can have attribute type: that's help user to find auction with condition that user prefers more .

Result information the result of certain auction , at a certain type and deadline.

Notification the existence of a notification directed to one or more user, take part in certain auction .

User creator auction a user can create zero or more auctions, an auction must have exactly one creator.

User join auction a user can participate to zero or more auction, an auction must have at least two participants.

User Bid auction a user after join auction can place a bid , user can place at least one bid .

Notification refers auction each notification refers to exactly one auction

3. Logical View

The final step of the database is schema of the tables, the so called logical view.

The final model has the following physical structure:

USER (id_User, photo , email, name, phone, password)

Join (USER_ id_User , AUCTION_id_Auction)

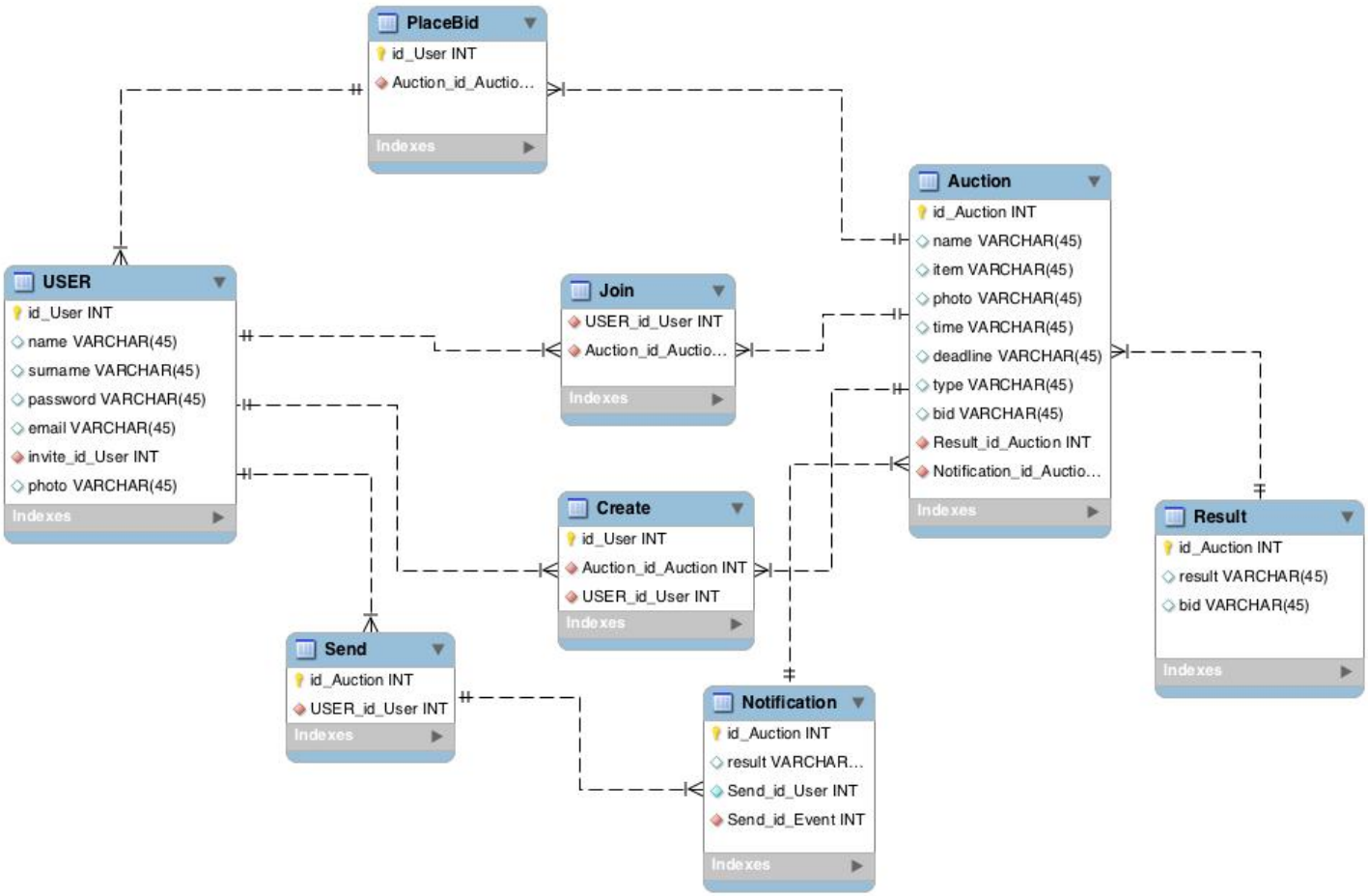
Auction (id_Auction, item, photo, type, date, deadline, bid, Result _id_Auction, Notification_ id_Auction)

Result (id_Auction, result, bid)

Notification (id_Auction, result, Send_id_Auction)

Send (id_Action , USER_id_User)

PlaceBid (USER_ id_User , AUCTION_id_Auction)


PlaceBid
id_User INT
Auction_id_Auctio...
Indexes
Auction
id_Auction INT
name VARCHAR(45)
item VARCHAR(45)
photo VARCHAR(45)
time VARCHAR(45)
deadline VARCHAR(45)
type VARCHAR(45)
bid VARCHAR(45)
Result_id_Auction INT
Notification_id_Auctio...
Indexes
USER
id_User INT
name VARCHAR(45)
surname VARCHAR(45)
password VARCHAR(45)
email VARCHAR(45)
invite_id_User INT
photo VARCHAR(45)
Indexes
Join
USER_id_User INT
Auction_id_Auctio...
Indexes
Create
id_User INT
Auction_id_Auction INT
USER_id_User INT
Indexes
Result
id_Auction INT
result VARCHAR(45)
bid VARCHAR(45)
Indexes
Send
id_Auction INT
USER_id_User INT
Indexes
Notification
id_Auction INT
result VARCHAR...
Send_id_User INT
Send_id_Event INT
Indexes

4. Software Design

4.1 User Experience

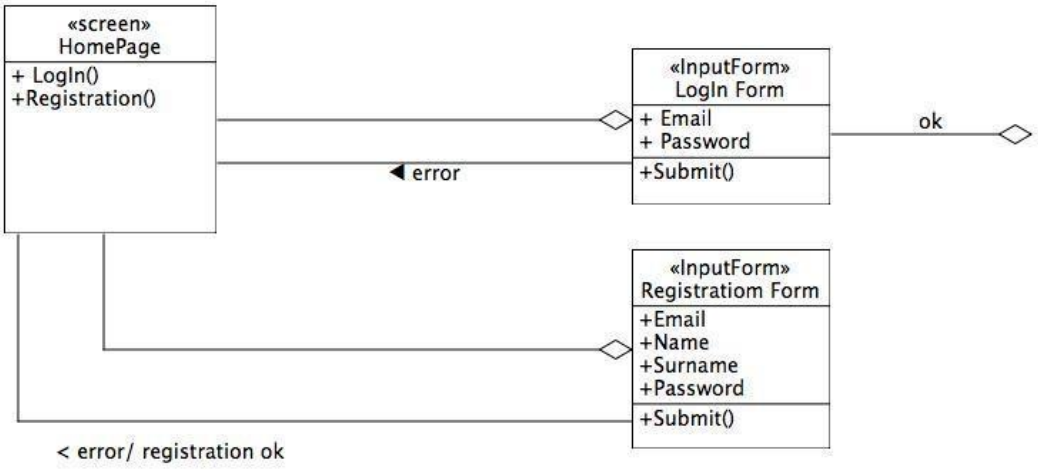
We want to show how our application work using UML program. The diagram contains two types of elements:

“screen” represents user page and user possibilities in GuessBid system, all pages are dynamical and user using provide links can access to other screen.

“Input form” is a page which provides user input some information about yourself, auction and other.

4.2 Registration and Login

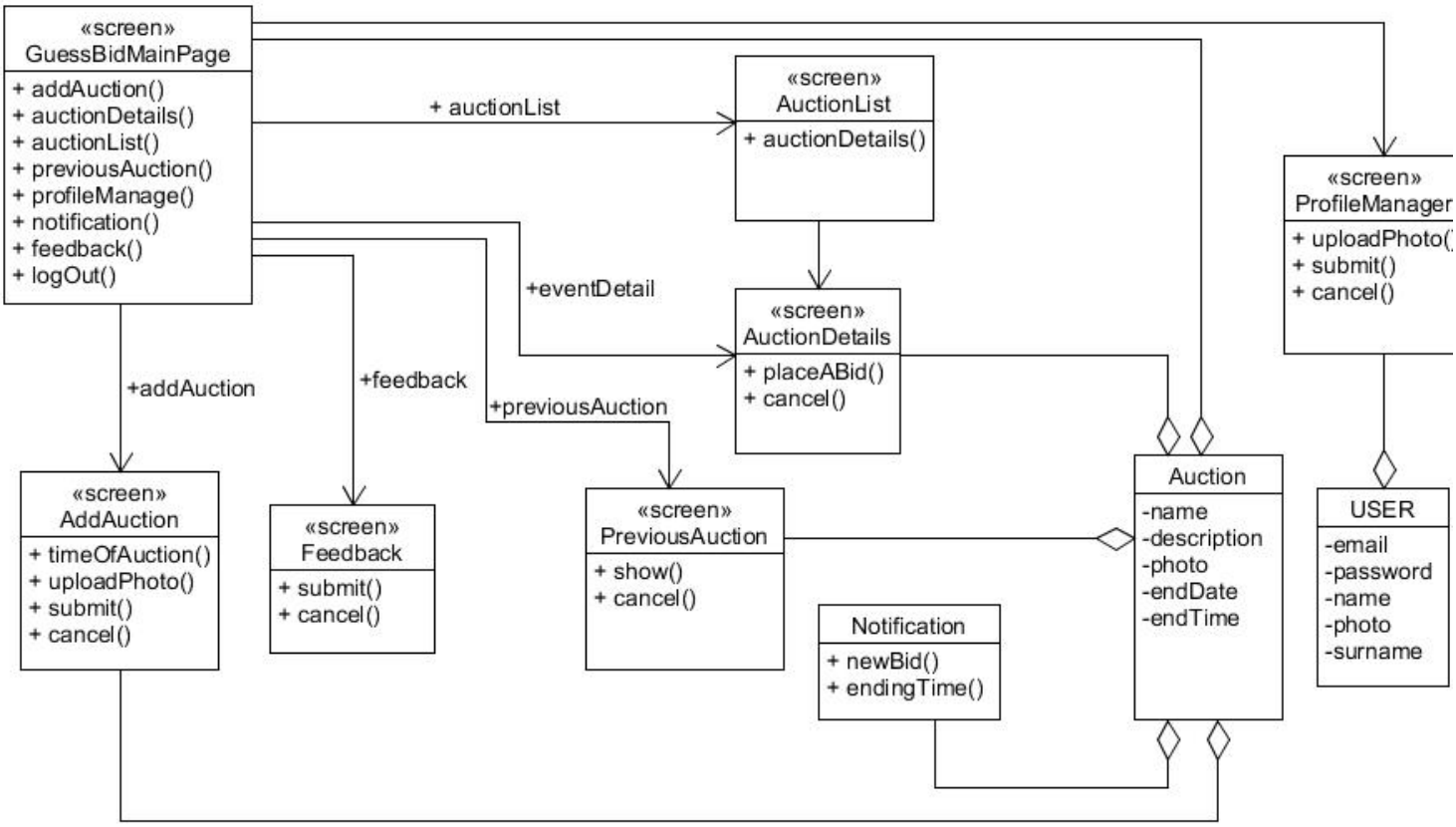
Diagram represent us how user can registrate or login in GuessBid system. If user has registrated in GuessBid, he can fill e-mail and password fields and login in the System. If user want to create new account he go to register page and fill all fields, if he fill correctly all fields, he redirect to GuessBid login page, in other case he return to the registration page.



4.3 GuessBid Home Page

Diagram represents the home page of our web application. To access this page the user must login using e-mail and password.

On this page user can join auction, create auction, search auction, write to developers of GuessBid system, and actually log out. System notify user about results of auction.



5. BCE

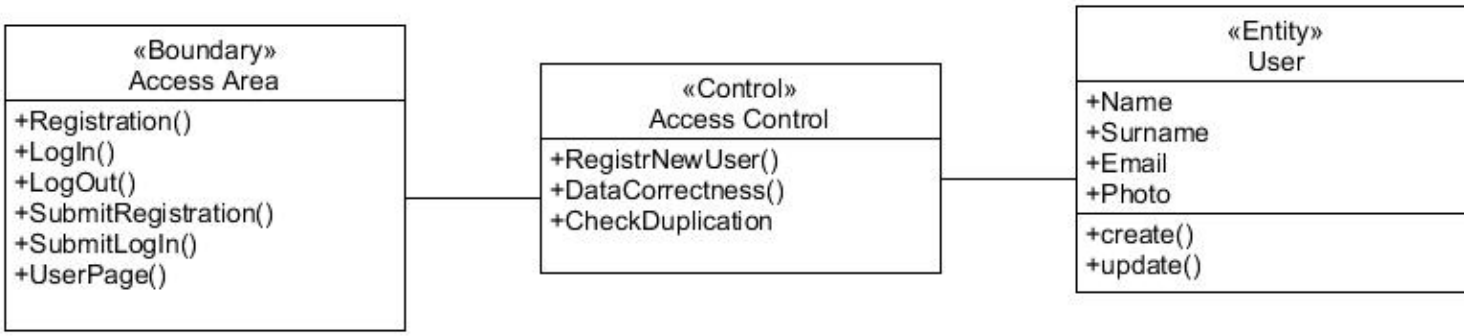
“Boundary” provides similar functionalities screens, which represents in 5.1.

“Control” represents modules of the system, which connects boundary and entity, represents in chapter 3.

“Entity” provides the application component that is reply to access the data stored in the database, represents in chapter 4

We create two **BCE diagrams** that shows us how user access the system and how user interact with system when he login.

5.1 User Access



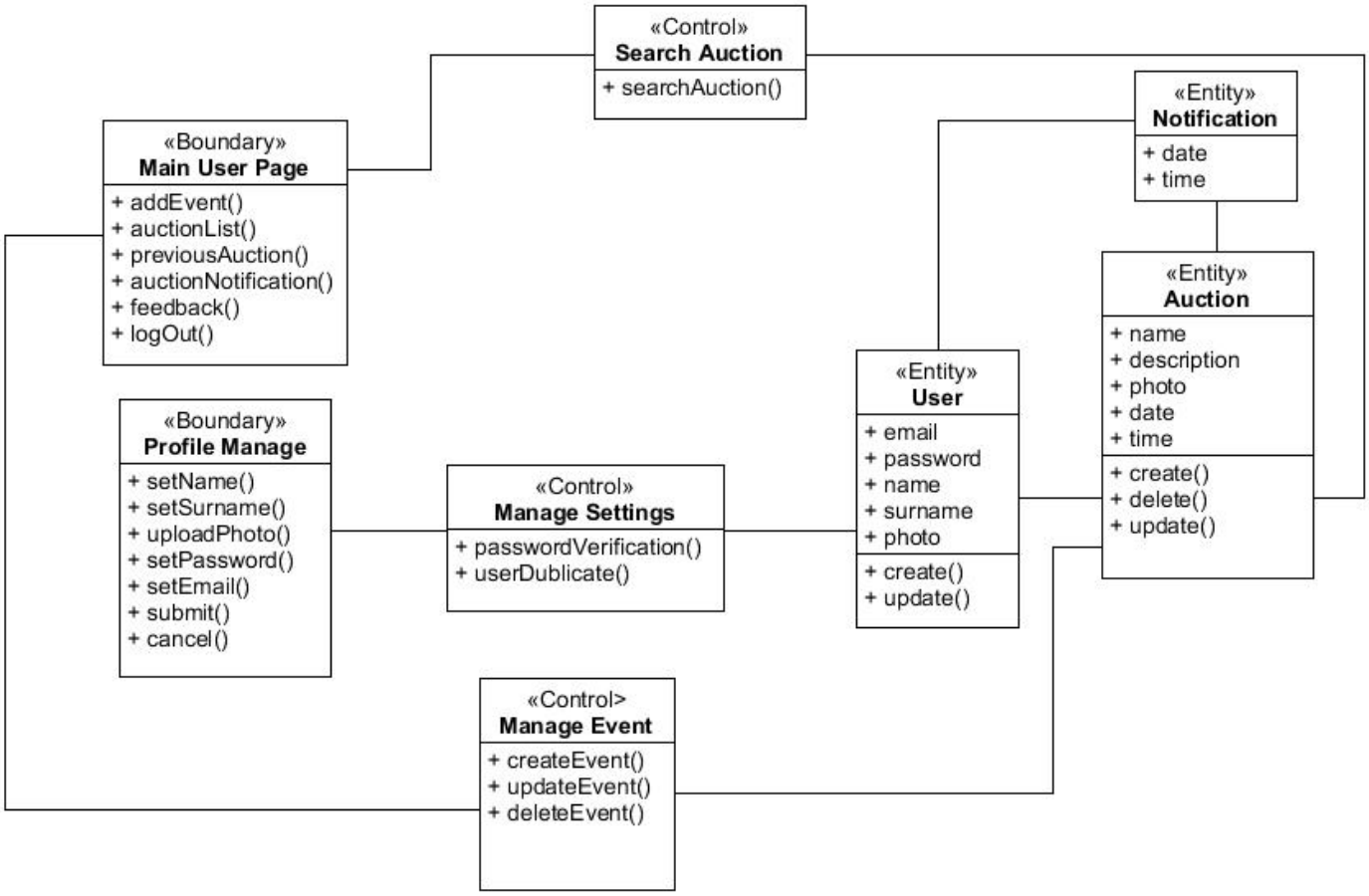
Boundaries:

- **"Access Area"** provide application screen: registration, login and logout method, to exit the system.

Control:

- **"Access Control"** is the module that control access to the system, that verificate, validates data and check data for correctness.

5.2 User Logged



Boundaries:

- **"Main User Page"** contains the screens concerning the main user page functionality.

- **"Profile Manage"** contains the screens of user profile settings.

Control:

- **“Search Auction”** module that is provide to “Search Auction” functionality.

- **“Join Auction”** module that is provide to “SearchAuction” functionality.

- **“Manage Settings”** module that allow a user to modify his profile information.

- **“Manage Auction”** module that allow the user to create his own auction.

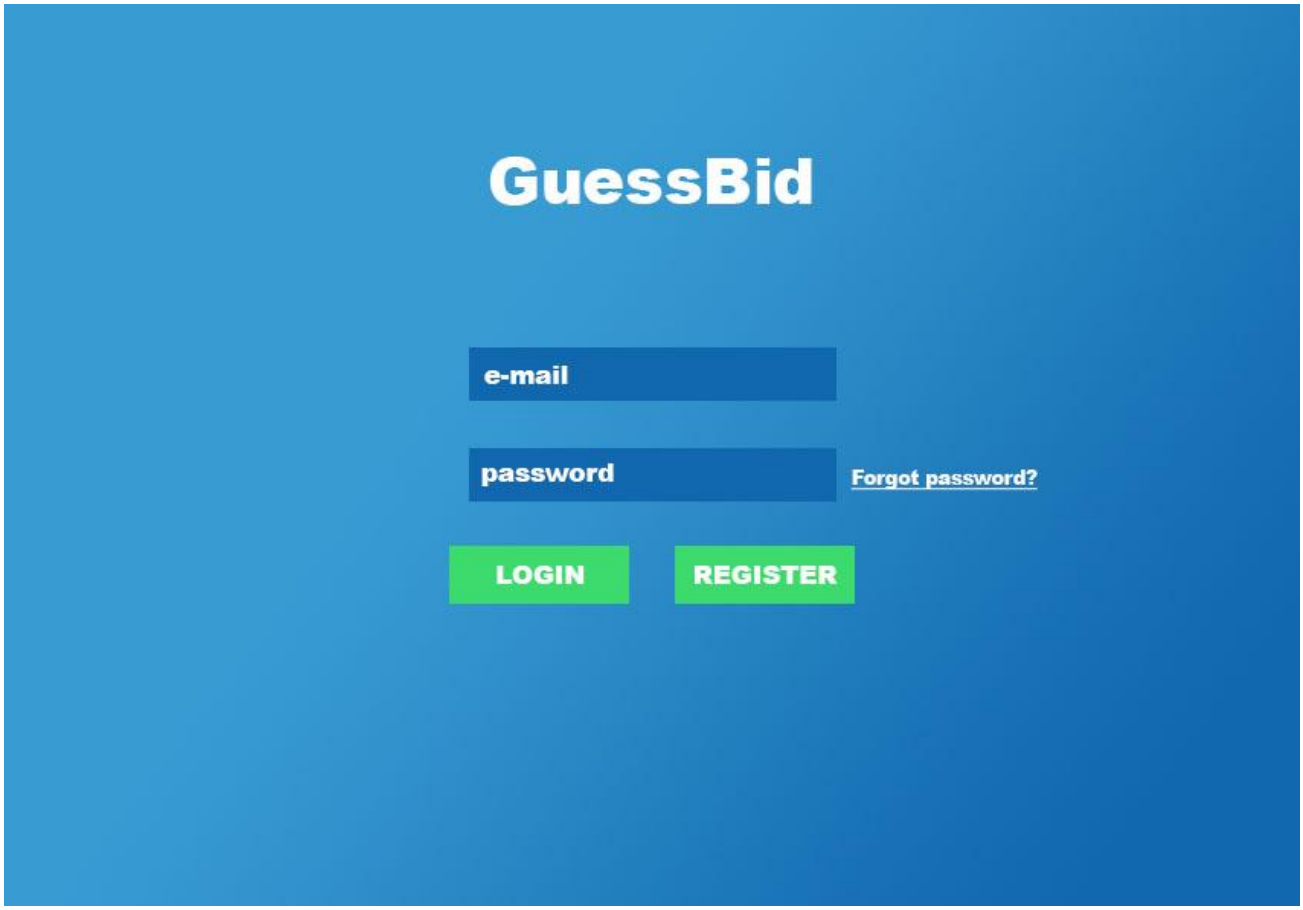
- **“Place a Bid”** module that allow the user to place a bid in auction.

6. User Interface

In this paragraph we will present of the main pages that include out application to present design and structure in each page.
We want to show the following pages that user can access: USER MAIN PAGE, AUCTION PAGE, PROFILE MANAGING, SEARCH AUCTION, BROWSE PREVIOUS AUCTIONS, Log out from GUESSBID.

6.1 Login and Registration Page

This is the page in which a user can do log or do a registration.



6.2 User Main Page

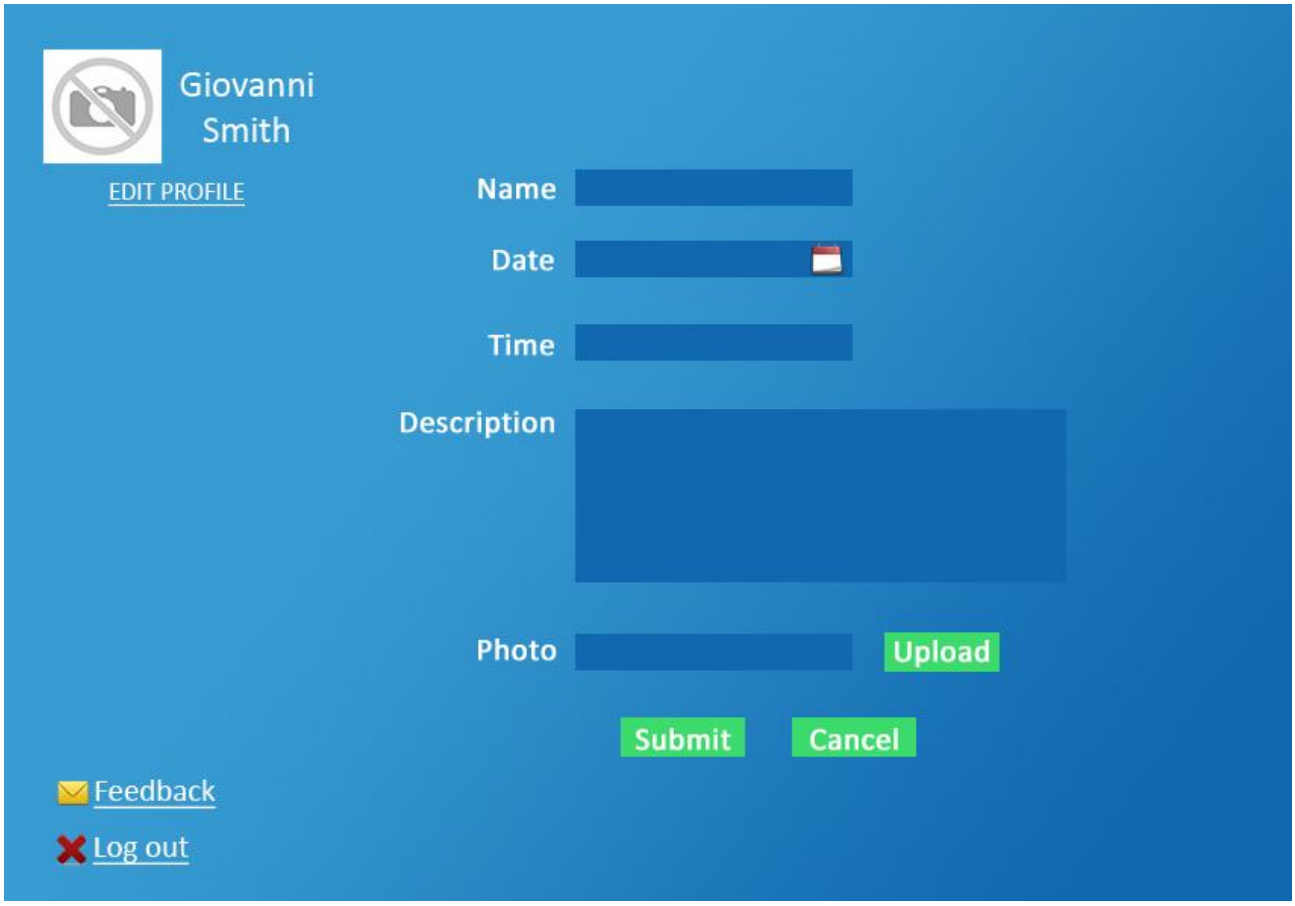
This is the page, which a user that had done the log in procedure can do all actions that our application provide. User can change his profile information, accept or decline invitation, create event, search events, use feedback, log out.

6.3 Auction Page

This is the page which view details about an event. User can go to this page from his personal main page.

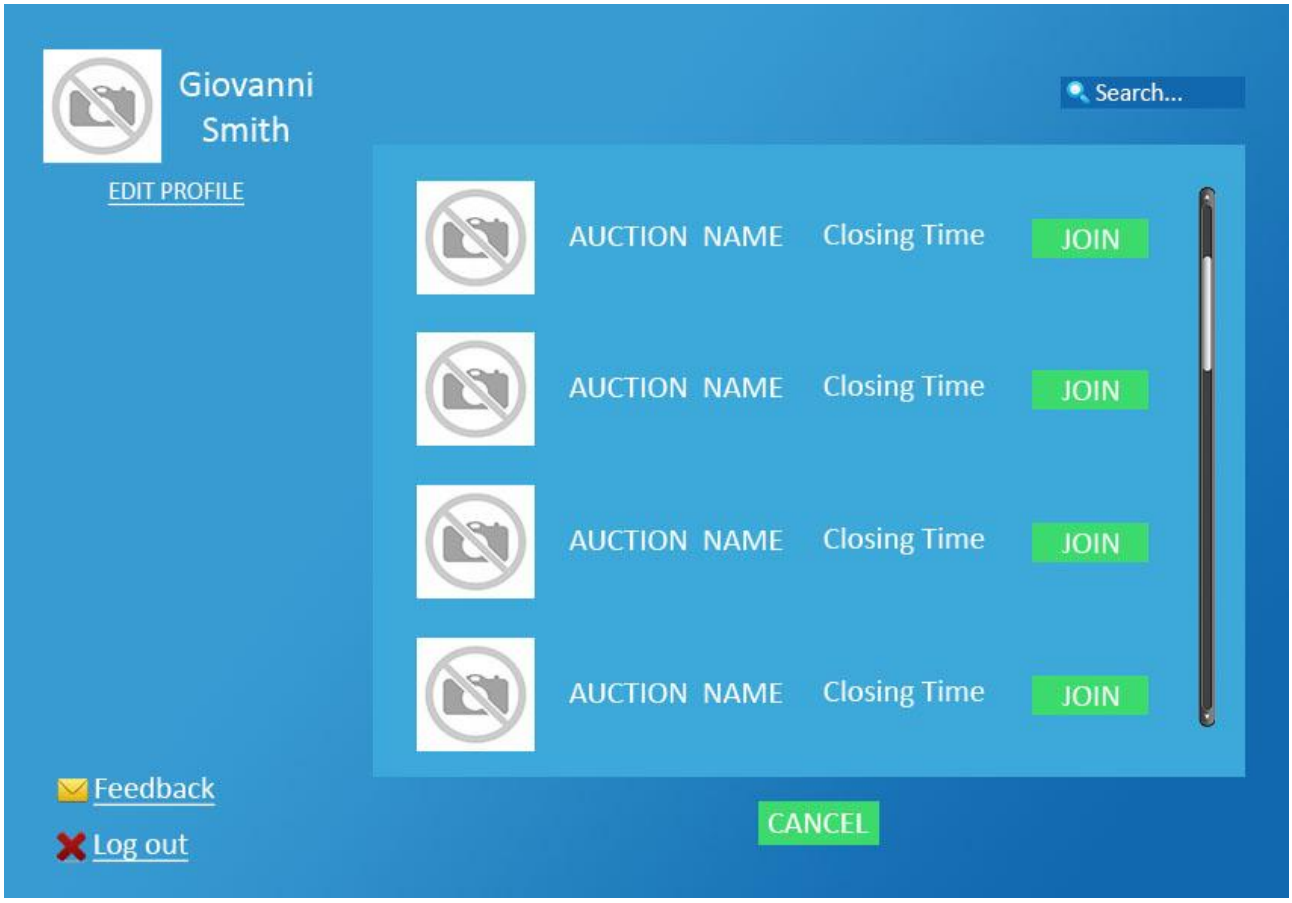
6.4 Add Auction Page

This is the page in which a user have to fill all textfields. If the creation will be fine the user is redirect to the event detail page.



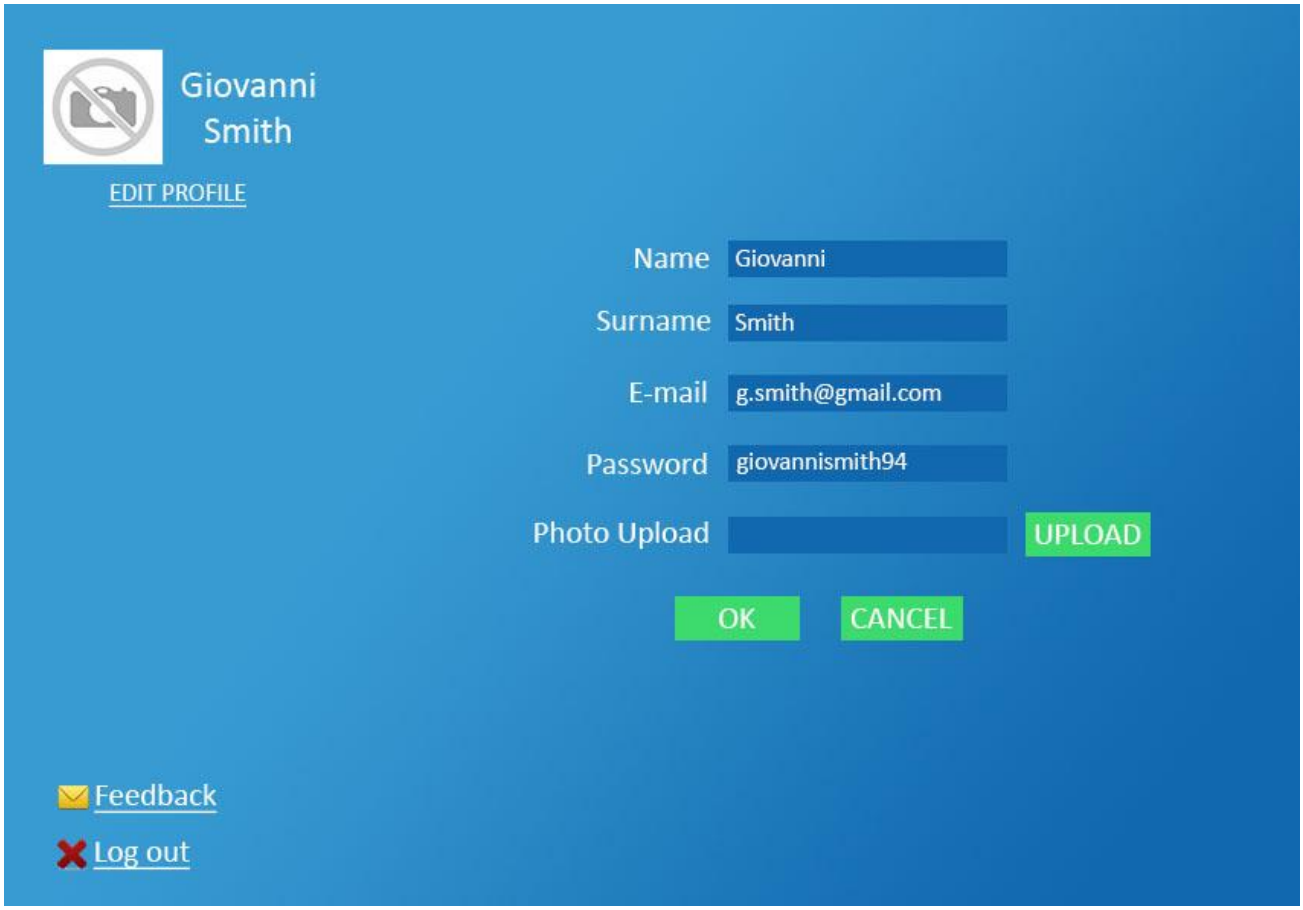
6.5 Auction List Page

The user will be redirected in this page when he wants to sent request for another user. User can see all users who join this application, and user can sent request for them by push the button “invite’.



6.6 Profile Managing Page

This is the page in which a user can change his personal information: his password, his photo, name, surname.



7. Appendix: Used Tools

To create this document the following tool has been used:

• Google Documents

• UMLet to create all UML diagrams.

• MySQL Workbench 6.2 CE to create the logical view diagram.