



# ONLINE AUCTION WEBSITE

GROUP 17

**GROUP LEAD:**  
SM AZEEM HUSSAIN

**GROUP MEMBERS:**  
ANUBHAV TYAGI  
SAMVID PRASOON  
SHIVANG GUPTA  
SHREYA VISHNOI  
SNEHA SINGH  
SM AZEEM HUSSAIN  
TANVI MANKHAPURE  
TARANPREET KAUR KOHLI

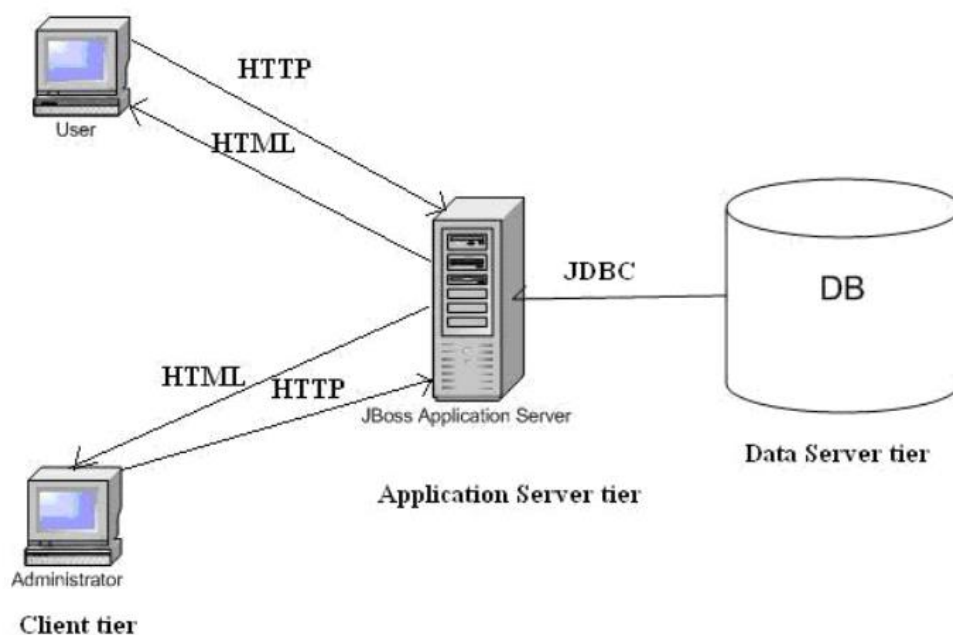
# Index

Topic	Page No.
An Introduction to our problem statement – ONLINE AUCTION	
Three-tier architecture	
Database and JDBC	
JSP	
Eclipse and MyEclipse	
Functional Requirements and System Design	
Use Cases	
User Stories	
UML Diagram	
ER Diagram	
Data Flow Diagram	

- **An Introduction to our problem statement – ONLINE AUCTION**

The purpose of this project is to build an “on-line auction management system”, a place for buyers and sellers to come together and trade almost anything. The system consists in a web-portal where registered users can propose new auctions, place bids in order to buy the items on auction. Auctions have a name, a description uploaded by users and an end period: users cannot place bids when the auction interval (start - end period) ends.

- **Three-tier architecture**



In such architecture, there are 3 main elements:

- **The client tier**, that is responsible for the presentation of data, receiving user elements and controlling the user interface.
- **The application server tier**, that is responsible for the business logic of the system. In fact, business-objects that implement the business rules "live" here, and are available to the client-tier. This tier protects the data from direct access by the clients. For the project, we used Tomcat as application server.
- **The data server tier**, that is responsible for data storage. As data server, we used MySQL, an open-source relational database.

- **Database and JDBC**

As database for the project, we used MySQL, that is an open-source relational database. A relational database is a type of database management system (DBMS) that stores data in the form of related

tables. Relational databases are powerful because they require few assumptions about how data is related or how it will be extracted from the database. As a result, the same database can be viewed in many different ways.

- **JSP**

For the realization of the web pages of the on-line auction portal, we used the JSP (Java Server Pages) technology. JSPs allow developers to combine HTML and Java code in the same document using special tags that identify Java code. JSP applies no restrictions to the Java language and its development is very simple. In fact, JSP pages are simply HTML pages with special tags including Java code performing elaborations and providing HTML code to be included into the page.

The first execution of a JSP includes three steps:

- Translation of the JSP code into a JavaServlet
- Compilation of the Servlet
- Execution of the Servlet.

The following executions include only the last step. Moreover, it is possible to precompile all JSP in order to reduce the time required by the first execution

- **Eclipse and MyEclipse**

For the programming part of the project we used Eclipse, that is a very powerful opensource integrated development environment (IDE). This IDE offers several services to the developer: it has compiler aware editing; syntax errors are highlighted when they are made, as are simple semantic errors such as missing declarations. Eclipse supports method completion, shows class interfaces concisely in graphical notation and supports interactive exploration of a program, through features such as fly-over name resolution. In addition, Eclipse supports Software Engineering principles such as packaging, debugging, testing, refactoring and versioning . Another very important advantage of using Eclipse is that several plug-ins can be installed on it, like MyEclipse, that is particularly useful when working with Enterprise JavaBeans. In fact, MyEclipse offers affordable tools for Java and J2EE (Java 2 Enterprise Edition) developers.

Features include:

- Web development tools, such as editors with code completion and syntax coloring for JSP, HTML, XML and CSS. They include also JSP syntax validation and native JSP debugging.
- Application Server integration, including integrated controls for starting and stopping application servers and full hot swap debugging support for deployed applications.

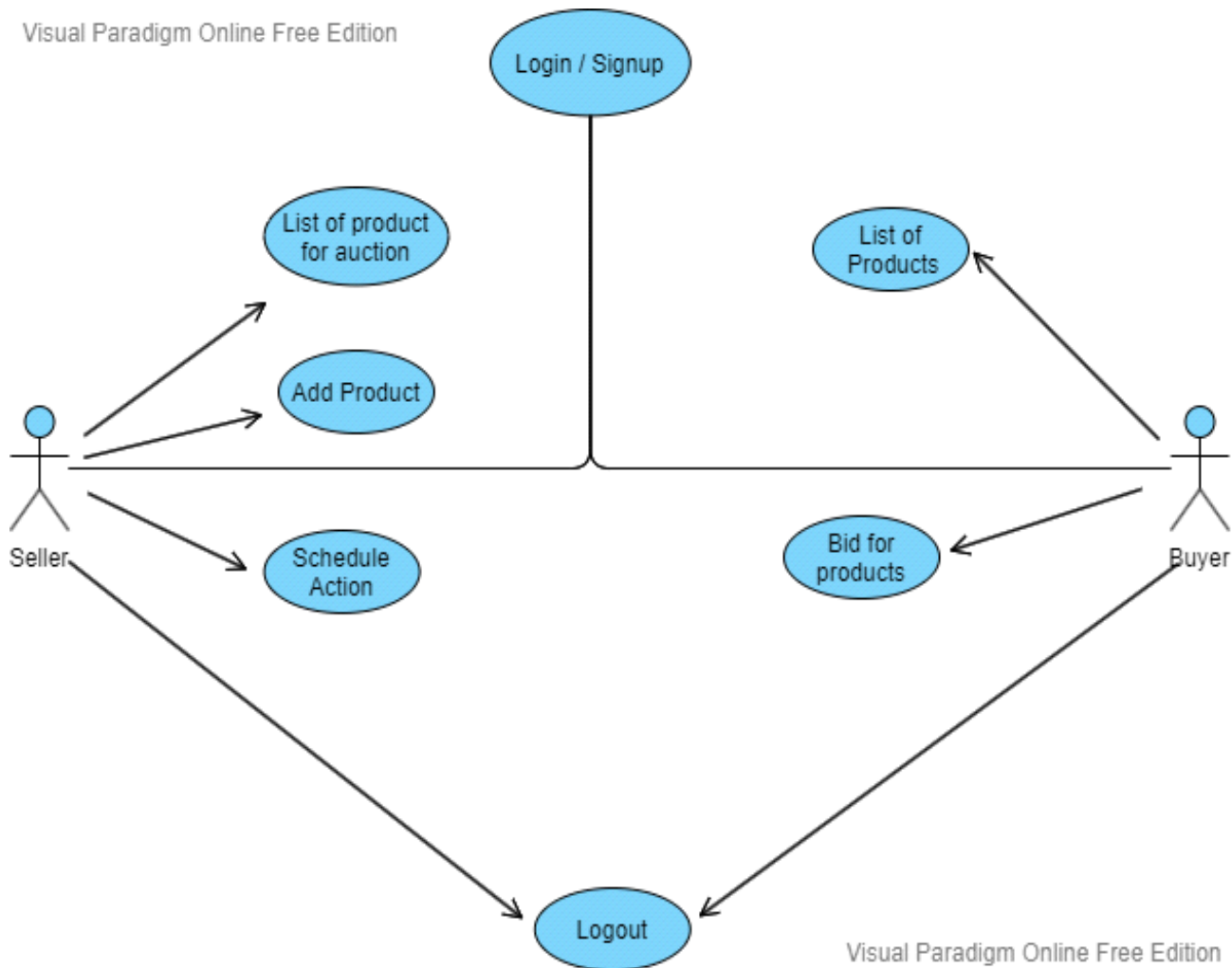
- **Functional Requirements and System Design**

The first step of the project is to find the functional requirement of the on line auction portal with the help of techniques like Use Cases and User Stories. After having found the functional requirements, the project goes on with the system design using the following techniques: the UML class diagram, the EER diagram for the database design and the page flow diagram.

- **Use Cases**

The first step for the functional requirement collection are the use cases. Use cases are “a description of set of sequences of actions, including variants, that a system performs that yield an observable result of value to an actor”. They are used in order to: design system from user’s perspective, communicate system behaviour in user’s term and enumerate all

externally visible behaviour . Here are the use cases for the on line auction system project (there are two actors for the system: a seller and a buyer).



As shown on the schema, a buyer can view list of products on auction, and bid them for purchases.

A seller, on the other hand, can insert and modify available data about items, users and categories of items.

## • User Stories

After collecting all use cases, user stories can be written. A user story “is the smallest amount of information (a step) necessary to allow the customer to define (and steer) a path through the system” . The user stories are divided into 2 main categories: buyer side and seller side.

User Side Stories:

**Home Page:** The home page shows also a list of categories to simplify items searching and the latest auctions. **Registration:** The registration page allows user to provide his/her personal data (name, address, date of birth, fiscal code, email address, phone number, userID, password) and receive a

userID and a password. UserID and password allow the user to access to his/her personal page, to take part to the auction and to post a new auction. It performs basic checks on entered data and provides user registration or an error message if the userID and/or user fiscal code are already present in the system.

**Login** Every time the user tries to access to non-public areas (personal page, bid, post an auction...), he/she is asked to provide his/her personal ID and password. These are entered through a form. If userID and password are correct, the user is logged in and is no more asked to login throughout the session. Otherwise an error message is raised.

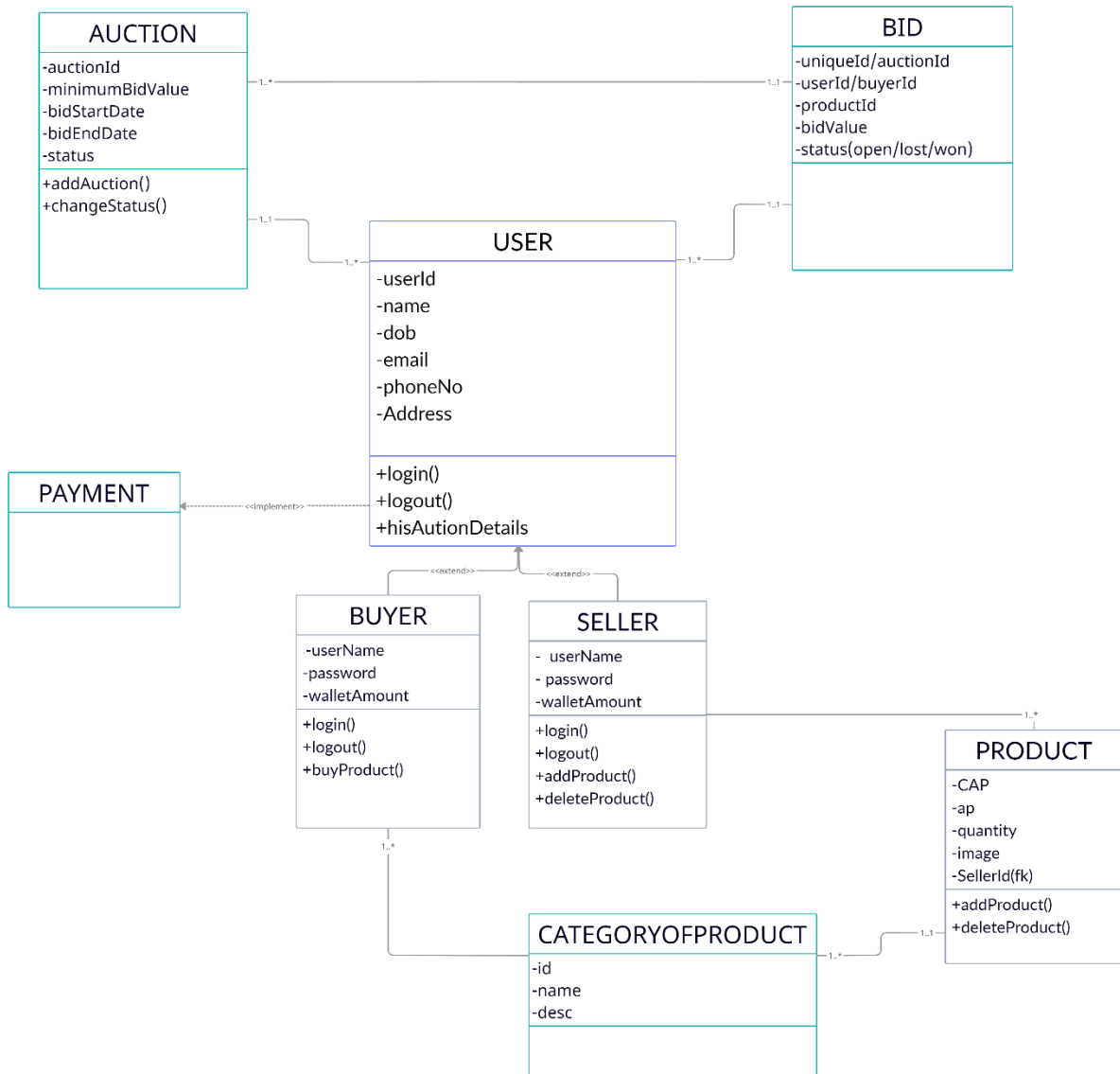
**Personal page:** To access the personal page the user is asked to login, or to register. The personal page keeps track of all the items the user is presently trying to buy and has bought in the recent past and of all items he/she is trying to sell. From this page it is also possible to post a new auction.'

**Browse:** The user can browse the auctions selecting among several categories of items (e.g. cars, books etc.). The results will be shown in a table and the user can sort them by price, by auction interval (by lasting period of the auction).

**Search:** The user can search for items on auction providing a key word. Auctions having price from a given value in Euro to another value Both registered and unregistered users can access to this service. Item page: Item characteristics are shown in the item page. From this page the user can place a bid pushing the button "PLACE A BID" and view the chronology of the bids. Bid: The user that makes a bid is asked to login if not already logged. If the bid is accepted by the system, the item is listed in the user personal page. Bids can only be placed during the auction interval and they must be at least one minimum increment bid above the current price. Post an auction: From his/her personal page, the user can post an auction from a specific form, providing the characteristics of the item he/she is willing to sell. If the auction is accepted by the system, the item is listed in the user personal page and other users can place a bid for it.

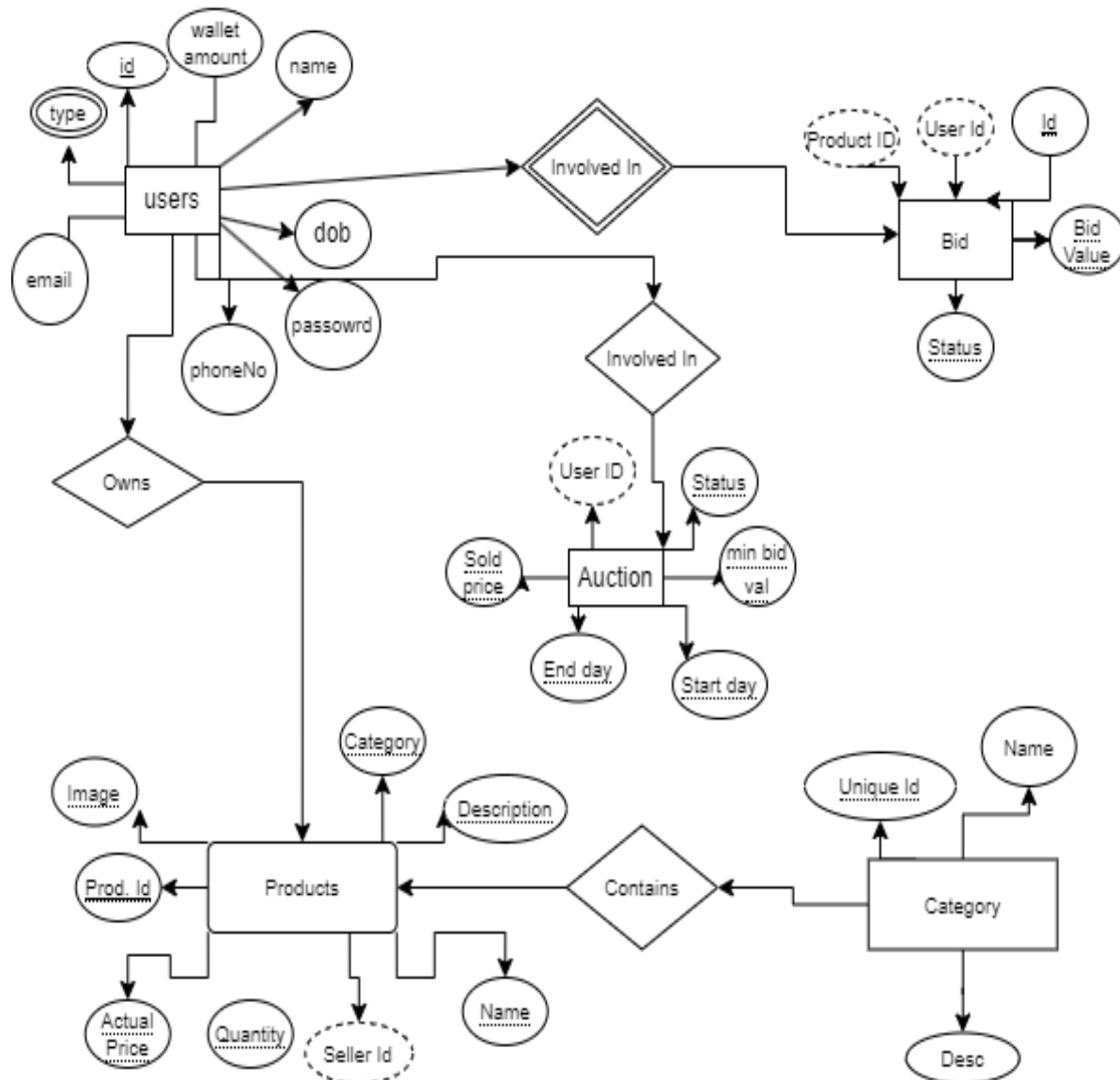
- **UML Class Diagram**

The next step of the design phase is to draw an UML Class Diagram of the system. Since the programming language of the system is an object oriented one, an UML Class Diagram is particularly adapted to show the classes of the system, their interrelationships, and the operations and attributes of the classes. Here is the class diagram of the project.



- EER Diagram for Database Design**

After having drawn the UML Class Diagram for the On Line Auction System, it is clear what kind of data should be stored in the database. Since MySQL is a relational database, the EER modelling approach is very useful to design the database schema since it maps well to the relational model and the constructs used in the ER model can easily be transformed into relational tables. Here is the EER Diagram for the database of the system.





## Data Flow Diagram

Since the on line auction portal consists of web pages, it is useful to draw a data flow diagram. A page flow is a diagram that visually organizes the flow and actions of the web pages. Here is the page flow that represents the path, for a normal user, to register to the system, search for an item and place a bid and make a logon. The following diagram shows the flow of data throughout the project. As shown on the diagram, from the home page it is possible to reach the registration page and enter the personal data of the user. If the data are right, the user can reach a confirmation page, otherwise (e.g. the username that he/she choose is already in use) an error page. From the homepage it is also possible logon inserting username and password and reach a search page to search for an item. The result of the search are given in another page where there are the links to see the item's detail (description, photo...) and place a bid for that item.

