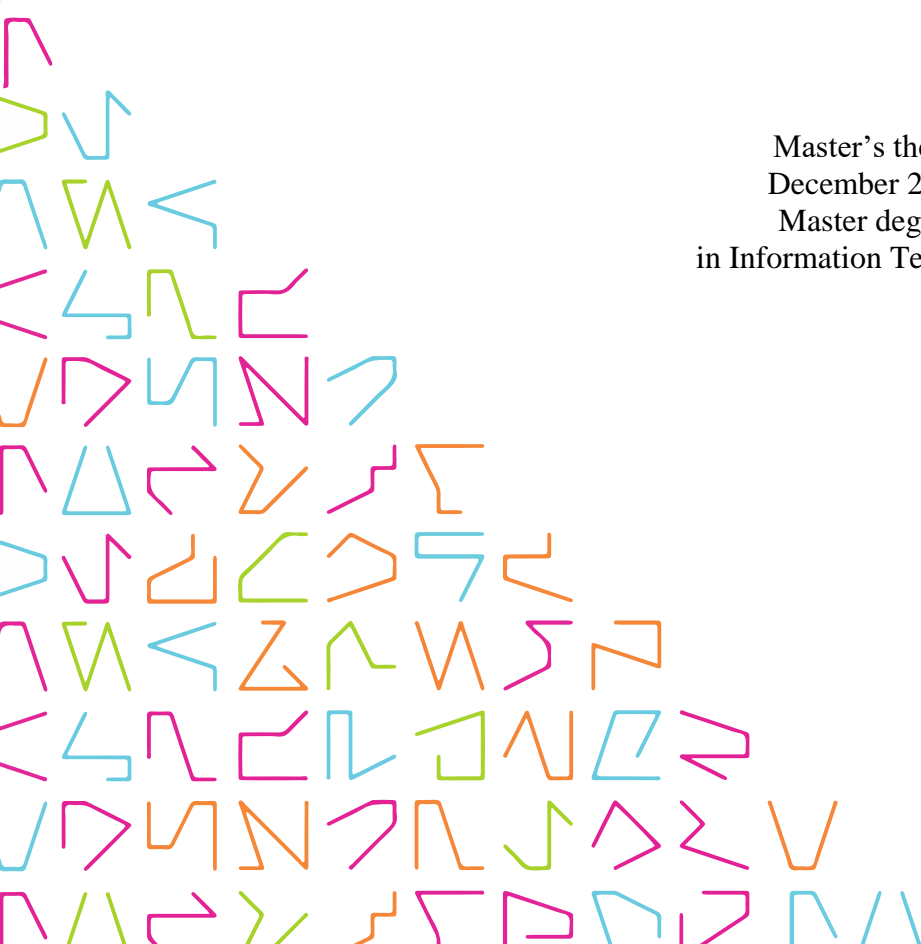


# ONLINE AUCTION SYSTEM

Begard Aljaf

Master's thesis  
December 2016  
Master degree  
in Information Technology



## **ABSTRACT**

Tampereen ammattikorkeakoulu  
Tampere University of Applied Sciences  
Master degree  
in Information Technology

Begard Aljaf:  
Online Auction System

Master's thesis 47 pages, appendices 7 pages  
December 2016

---

The objective of this thesis was to develop the Online Auction System in Iraq. The main reason behind implementing this project was the people needed a system where they can make the auctioning through the internet easily and reliably. The system was implemented to replace the tradition auctions by online auction, due to the fact that in the epoch of the internet and e-commerce online auction had an important role in business and it had been popular means of selling goods.

This system was the first online auction system that developed in Iraq. Nowadays' online auction system is the most popular web application that dozens of users can take part in it via the internet. Since now there was no online auction system there, where people can directly buy and sell items through it. Therefore, it was quite important to develop a reliable and easy system where to be used by users.

The system was implemented by using most well-known programming technologies and languages. A lot of time was spent on solving authentication problem. It was very important to develop the system that the users of the system trust each other during buying and selling. The client side was needed to be friendly user interface CSS entirely used with HTML and JavaScript. The server side program was needed for storing all required data into the database by using MySQL database with PHP which they easily can retrieve and store the data into the database through SQL queries. It was required a widely knowledge and skills of the system design and programming languages which greatly helped to implement the web application for the need.

The implementing this project was successfully done. Adding more features to the system will be implemented in near future.

---

Key words: online auction, auction, bid, bidder.

## **FOREWORD**

First of all, I would like to show my gratitude to my lovely husband 'Ali' for his support and patience with me during writing this thesis, without his support and aid I couldn't complete writing this thesis.

Special thanks to my supervisor 'Erkki Hietalahti' for his useful guidance and valuable comments.

Begard Aljaf

Tampere, December 2016

## CONTENTS

|       |   |    |
|-------|---|----|
| 1     | INTRODUCTION.....                             | 6  |
| 1.1   | Purpose.....                                  | 6  |
| 1.2   | Objective.....                                | 6  |
| 1.3   | Aim.....                                      | 6  |
| 1.4   | Scope.....                                    | 7  |
| 1.5   | The study limitations.....                    | 7  |
| 2     | LITERATURE REVIEW.....                        | 8  |
| 2.1   | History of Auction.....                       | 8  |
| 2.2   | Types of online auctions.....                 | 9  |
| 2.3   | Existing online auction systems.....          | 10 |
| 2.3.1 | Ebay.com.....                                 | 11 |
| 2.3.2 | Huuto.net.....                                | 11 |
| 2.3.3 | The functionalities of the system.....        | 11 |
| 2.4   | Existing system.....                          | 13 |
| 2.5   | Determination of problems.....                | 13 |
| 2.6   | Proposed system.....                          | 14 |
| 2.7   | Application technologies.....                 | 15 |
| 2.7.1 | Web server.....                               | 15 |
| 2.7.2 | Programming languages.....                    | 16 |
| 2.7.3 | Software tools.....                           | 19 |
| 2.7.4 | Three tier architecture.....                  | 20 |
| 3     | REQUIREMENTS AND SYSTEM DESIGN.....           | 22 |
| 3.1   | System overview.....                          | 22 |
| 3.2   | Use case.....                                 | 22 |
| 3.2.1 | User use cases.....                           | 23 |
| 3.2.2 | Administrative use cases.....                 | 32 |
| 3.3   | UI layout.....                                | 38 |
| 3.4   | Class diagram.....                            | 39 |
| 3.5   | ER-Diagram.....                               | 41 |
| 4     | SYSTEM ARCHITECTURE AND SOFTWARE TESTING..... | 43 |
| 4.1   | System Architecture.....                      | 43 |
| 4.2   | System testing.....                           | 43 |
| 5     | DISCUSSION.....                               | 45 |
|       | REFERENCES.....                               | 47 |
|       | APPENDICES.....                               | 48 |
|       | Appendix 1. Database SQL script.....          | 48 |

**ABBREVIATIONS AND TERMS**

|            |  |
|------------|--|
| CGI        | Common Gateway Interface                                   |
| CSS        | Cascading Style Sheet                                      |
| ER-diagram | Entity Relationship diagram                                |
| FastCGI    | Fast Common Gateway Interface                              |
| HTML       | Hypertext Mark-up Language                                 |
| HTTP       | Hypertext Transfer Protocol                                |
| JavaScript | Scripting programming language                             |
| MySQL      | Open source database                                       |
| PHP        | Server side scripting language                             |
| TAMK       | Tampere University of Applied Sciences                     |
| UI         | User Interface   |
| IIS        | Internet Information Services                              |
| URL        | Uniform Resource Identifier                                |
| WAMP       | Apache, MySQL, PHP/Perl/Python software bundle for Windows |

## **1 INTRODUCTION**

In the epoch of the internet most of the trading are done via the internet. The trading over the internet makes the business to access the huge number of clients faster and with smaller cost than making business physically. The most common type of e-business which has popularity in the past and now is Online Auction System.

Online Auction System is a web application which is built for buying and selling items through the internet by making a bid to the item. Increasing the bid over the time and the item will be won by the bidder who makes the highest bid. Many kind of auctions are existing but the most famous one are English auction ascending and Dutch auction descending.

In the developed project the English auction has been used where the price of items will increase.

### **1.1 Purpose**

The main purpose of developing this project is to make people who live in Iraq get familiar with the online auction systems and trading over the internet.

### **1.2 Objective**

- To design user friendly online auction system in Iraq.
- To make users participate in auctioning trusty.
- To authenticate users' account during registration.

### **1.3 Aim**

To provide a simple and easy way for buying and selling over the internet.

#### **1.4 Scope**

The implemented online auction system allowable for users who live in Iraq only. The users can participate in auctioning only if they have created an account through the registration form.

#### **1.5 The study limitations**

The lack of the auction sales in Iraq which is led to the difficulties to know more about the invisible sides of auction sales. The another reason was lack of time which made limitation of this research.

## 2 LITERATURE REVIEW

The term Auction extracted from the Latin root *auctus* which means “I augment”. It is a process of purchasing and selling items through the bid and selling to the bidder who makes the highest offer. Usually the bidders that participate in the same auction item will increase the price against each other. The increasing prices continues until it reaches the highest price by the bidder. Many types of auction exist; each of them have their rules and conditions. The auction is a fact which items goes to the highest bidder. By publishing the items to the audience an auction may have a wide range of bids which make the item surprisingly goes to the very high price.

The Auction sales are held in the auction houses or in specific places. The people who wish to sell their items by auctioning take part in auction houses and publish their items for sale. Also, the buyers have to exist in the auction places which takes the time to arrive. In the auction sales, the seller has to pay some amount to the auctioneer who is managing the auction by accepting offering price and declaring items sold.

### 2.1 History of Auction

The history of auction returns back to 500 B.C as stated by Herodotus. In Babylon where annually women were sold based on auction and being married to the person who has bought her. Women with less pretty were sold with returning money back to the bidder. The bids of these auctions were not mentioned whether they were ascending or descending.

The Roman Empire used auctions to dispose of debt. One example is selling the own home ‘s furniture by Marcus Aurelius to pay off debts.

The important historical auction happened in the year 193 A.D when the whole of Roman Empire was placed on auction block after being dismissed. In China also auction was used by Buddhist friars to put up money for temple creation. After that it became usual to auction of the possessions of deceased friars for this purpose.



The auction sales also existed in England, the London newspaper often notified the auctioning at coffeehouses and pubs everywhere in London in the late 17th century. The large auction houses were built in the early 18th century. Sotheby's was created in 1744 and Christie's was created in 1766.

In America auction sales appeared in the south when slaves were sold in auction. Often times, the merchandise owner did not allow to reveal himself because the present social customs might not view favorably to the auction which affect the auction negatively.

In Netherland and Germany auction has also been used, the auctioning used in Netherland for selling fruits and vegetables return back to year 1887. In Germany fish was sold by auctioning.

## **2.2 Types of online auctions**

There are many different types of online auctions, the most important ones are:

### **1. English auction**

The bids of English auction are declared either by auctioneer or by buyer. It is the most common type of online auction due to the easiness of use among other types. The format of bid auction is ascending. The bids are open for all to see. The bidder who makes the highest bid price is the winner. The online English auction is preferable by many people. Because it uses a simple mechanism that people find it familiar. Online English auction has become more popular than traditional English auction due to the fact that traditional English auction needs the physical existence of the bidders.

### **2. Dutch auction**

Dutch auction begins in high price and the price decreases orderly to the price that become acceptable by the bidder. It is the reverse of English auction which is ascending the price. The Dutch auction commonly uses a uniform-price rule in a single unit auction. Mostly the items that put in online Dutch auction ends with

the higher price than their first price which makes the bidder feel remorse about participating in Dutch auction.

### 3. Bidding fee auction

In this kind of auction all participants must pay for bid but only one participant will win who makes the highest bid price. The participants buy the penny for makings bids instead of paying real money. The auction website's owner makes the money via buying the penny bids by bidders and actual value of final price.

### 4. First-price sealed-bid

The variation between this kind of auction and English auction is in this type of auction the bidders cannot see the offers made by another bidder. It will remain confidentially until the end of auction. While in English auction the bids are public and viewable by all bidders.

### 5. Second-price sealed-bid

It is known as Vickery auction too. It is very similar to first bid auction and it is working in the same way. The only difference between this type of auction and first bid auction is that the bidder with the highest bid who wins will pay the second highest bid instead of the bid made by him.

## 2.3 Existing online auction systems

After knowing types of online auctions, it is a good thing to investigate about current online auction systems around the net. Providing their functionalities. Two famous auction systems are considered for this research eBay.com and huuto.net.

### 2.3.1 Ebay.com

eBay is the world's largest online marketplace which let people to trade locally and world-wide. The people are able to buy and sell on eBay through online auction formats, which is also called auction-style listings. Or direct buying through 'buy it now'. eBay was founded by Pierre Omidyar in 1995.

### 2.3.2 Huuto.net

Huuto.net is a Finnish online auction much the same as the global eBay. It was founded in 1999 by Lari Lohikoski in Helsinki. The name “huuto” is extracted from "huuto-kauppa" which means 'auction' in Finnish language.

### 2.3.3 The functionalities of the system

Table 1 below shows the main functionalities between both systems.

TABLE 1. Functionalities of the systems.

| User stories    | Ebay.com | Huuto.net |
|-----------------|----------|-----------|
| Login           | X        | X         |
| Register        | X        | X         |
| Home page       | X        | X         |
| Chat            | X        |           |
| Own page        | X        | X         |
| Place a bid     | X        | X         |
| Upload item     | X        | X         |
| feedback        | X        | X         |
| Change language |          |           |

|                    |   |   |
|--------------------|---|---|
| Search by category | X | X |
| Excluding a word   | X | X |
| By city            |   | X |
| Message            | X |   |
| Help               | X | X |
| Shopping cart      | X |   |
| Sitemap            | X |   |
| Administrator      | ? | ? |

As shown in the table 1 above, the two systems allow the user possibility to register, to login to the website and have a home page. Both websites also have a personal page, where each user can monitor the status of their auctions or of their bids and changing their personal information. Another function of both websites is having an item page, a page that represents each item on auction (with a name, a description, a photo etc.). The search functionality is also very important: in addition to a normal keyword search, both websites offer also the possibility to search excluding a given word, search in a given category. Huuto.net offers also to search for auctions regarding a given city. The two systems give also the possibility to place a bid, to upload an auction and have also some help pages. The feedback functionality is the important function which exists in both sites, it gives the users opportunity to leave their feedback about their experience when they are buying or selling something. On eBay, the user also has the ability to chat and sends a message to another user. Another important feature on eBay is shopping cart which is allowing the buyers to compare and see the final cost of the item before checkout.

The sitemap is another important functionality on eBay to make easier navigation through the website. None of the two websites has the ability to change language (huuto.net is in Finnish, while eBay.com is in English but has different versions in multiple languages for example ebay.es is in Spanish language and ebay.de is in Deutsch language and so on).

## **2.4 Existing system**

The current system of auction in Iraq until now is offline system. It means that the system is managing manually. The people who wish to participate in the auction have to be presented on the auction place on a specific time. The number of participants are limited due to the area of auction which leads to join a smaller number of participants. The smaller number of participants have a direct impact on the selling items, because fewer number of participants means having less bids to the items. Sometimes the seller sells his/her item in small value. In this traditional auction auctioneer has to follow each bidding process and record each seller and bidder until the end of the auction which is very exhausted and takes too much time.

## **2.5 Determination of problems**

The major problems with the current offline auction system in Iraq are the place of auction and the time of auction is very limited. It leads to participating a few number of people. A few number of participants make the items to sell with less price. Another problem with the traditional auction is keeping track of items, each seller, and bidder until the end of auction process. This problem makes a fewer number of items to be sold. Because a lot of time will be a waste by following every bidder and seller.

A further point is that the people in Iraq cannot participate in such online auction websites like ebay.com, tradera.com, etc. The Iraqis residents are unable to create an account on eBay.com. Iraq does not exist in the country lists of registration form for creating an account. For creating an account on tradera.com, the person needs to have PayPal account. PayPal service does not exist in Iraq. So that they are unable to participate in online actions. The aim of this development project is to create an online auction system so that everyone in Iraq can participate in it.

## 2.6 Proposed system

The proposed system is creating online auction system for Iraq that allows everyone using it. No matter where they are or when they use it. From now on, there is no need to go anywhere to participate in auction process. Alternatively, they can stay at home and join it. At any time, they are wish to participate in it. The only thing that users have to do to take part in bidding process is registering and confirming their emails. The email verification will be done through a confirmation link that they will receive in their email inbox after registration. Without this verification the users cannot access to the website.

The problem that ordinarily occurs in the online auction systems is the user's unreliability towards each other and uncertainty to the seller's item, because the buyer is unable to see the item physically. The users must verify their identities to reduce the fraud make by users and producing higher level of confidently. Thus a person is known who she/he is. By this way the users can trust each other, and providing confidence and willingness to deal with each other. The verification of user's identities is done by asking user to input his/her national card number at registration time. In Iraq currently national card is the only computerized identification that is used by the government. It is used for identifying each person's personal information. On that way, the user cannot do with fraud. In proposed system the users require to provide detail descriptions of items which they upload them for auctioning. Giving detail information about items leads to increase the sureness of bidders to seller's item.

What the proposed system is trying to achieve is to produce a higher level of confidence among users, the type of sellers, and items that they choose for bidding. By making the seller give the efficient information like visible and textual description of items. The outcome of implementing this system successfully will be an online auction system. It provides the evaluation of the item that is far much effective and that come close or equal the physical evaluation of the item.

## 2.7 Application technologies

The aim of this project is to construct an online auction system. The user should understand how this website work when opening a web browser and typing the online auction's URL. All steps have to be described in detail to get an overview and better understanding of the system. This is done by knowing the technologies that are used to implement this project.

### 2.7.1 Web server

The web server's role is to be a communication link between client side and server side and to come up with information released on the server by the web pages. The action of a web server is to take HTTP requests and return the requested resource in an HTTP response. The chosen application web server was Apache (Apache HTTP server). It is existed on the WAMP service. The Apache HTTP server is the common web server software.

#### **Apache web server**

Apache is the most popular web server software. It allows a computer to add websites that can be approached through the Internet by using a web browser. The reason behind popularity of Apache because it is free to utilize and open source. So, the web hosting companies are proposed to solve apache-based web hosting at minimal cost. Apache runs on multiple operating systems including UNIX, windows, Linux, mac OS. It is used by most popular web sites such as PayPal, Wikipedia, Apple and much more.

The reason of choosing Apache as a web server was because it is able to host dynamic web sites that uses server side language's code such as PHP, Perl that support them through module. The development project implements its server side by using PHP, therefore the Apache was the best choice.

## 2.7.2 Programming languages

According to necessity and knowledge the programming languages were selected to implement the development project. PHP was chosen as a server side programming language, due to its containment of the MySQL software library for supporting SQL statements executed in MySQL database. For backend database MySQL was selected as a database storage. In the client side JavaScript, html and CSS were selected.

### PHP

PHP is the powerful programming language for producing dynamic web pages. PHP essentially is used as a server side scripting language, but also it can be used in command line scripting or desktop applications. It is free software and available in several different versions. For running the server side program, a web server is required. In this project Apache was selected as a web server with connecting to PHP installation. For reaching the output of PHP program it needs a web browser and presenting the PHP page through the server.

PHP can be used on many operating systems including Windows, Mac OS, Unix and many others, and many web servers are supported it. Including Apache, IIS and others and any web server that is able to use FastCGI PHP binary. The works of PHP as module, or CGI processor.

The suitability of using PHP in this project because it is able to easily present image and PDF files in HTML pages. It can easily output XHTML and any XML files. So, in this project the large number of images are used because each item which is uploading in this web application must have an image to be viewed by buyers. Another benefit of PHP in this project was supported easily to databases like MySQL which was chosen as our database.



## **Client side programming**

The structure of the web pages is created by HTML. For styling the web pages CSS was used. JavaScript language was used for adding more functionality to the web pages. The design of the web pages was done by all of these three languages.

### **JavaScript**

JavaScript is object oriented scripting language. It is mostly supported by all browsers. JavaScript is used for adding additional powerful and complicate functionality roughly on all websites. It allows designing interactive sites by web developers. The dynamic conduct of web pages is mostly done by JavaScript. Shortly, JavaScript informs the browser according to events which occurred. It changes the web page as a response to that event (like button click).

The usage of JavaScript in this project was because of the responsibilities of it to handle dynamic behaviors of a web page. It was used for creating some features in this project such as a timer for countdown the item's remaining time, because each item has end date which is counted by timer. Also, JavaScript alert () was used for notifying the bidder if he/she was out of bid and used in much more features.

### **HTML**

HTML is used for structures of a web pages. The reading of HTML documents is done via the aid of web browsers such as (Chrome, Safari, Firefox). The tags inside the HTML documents do not appear by a web browser. The web browser just uses tags for determining how to present HTML documents.

## **CSS**

The user friendly web site needs to have a satisfying look for attracting users, the colors, nice fonts and layouts of HTML pages was done by CSS.

## **Backend technology**

For building the project development's database MySQL was chosen as a database.

## **MySQL**

MySQL is the most common database management system that is corroborative by Oracle corporation, is developed and handed out by Oracle.

MySQL is a relational database that keeps data in different tables instead of placing all data in one storeroom. The MySQL database is accessed through SQL language because it is open source everyone can use it. It can be freely downloaded from the internet and use it without paying any fees.

The MySQL was chosen as a database because it is very fast, reliable and due to PHP's ability to connect directly to MySQL database. MySQL and PHP are major elements for developing this online auction system. The stored, altered and retrieved data in MySQL is made obtainable to the Apache web server by the PHP interpreter that was located in the web server.

### **2.7.3 Software tools**

Every software project needs several tools to do various tasks. In this development project the free source tools were used. The tools were chosen according to necessary and having previous experience on them.

#### **Notepad++**

Notepad++ is free, small in size text editor. It is used for writing several language codes, because it supports plenty of languages. It was used for writing HTML, CSS, JavaScript and PHP codes of the development project.

#### **WAMP Server**

WAMP Server is the platform for designing and coding web pages on Windows. That is freely available which permits you to build dynamic web applications with Apache server, MySQL database and PHP scripting language. The phpMyAdmin also included in WAMP server that allows you without difficulties to manage databases made by PHP.

By using WAMP server you are capable of developing web application in an intuitive way. WAMP server is configurable with the built-in tools. You are able to set your server without changing its file settings.

The figure 1 below demonstrates the WAMP server architecture.

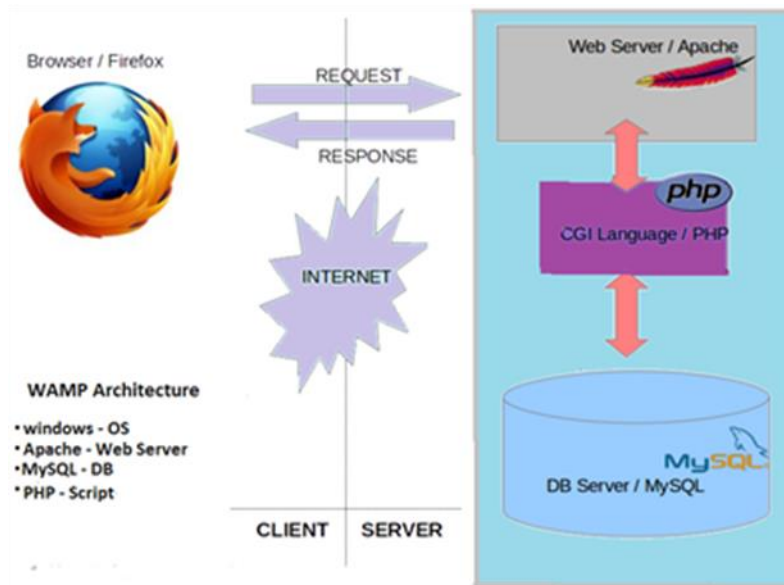


FIGURE 1. WAMP server architecture

## MySQL Workbench

MySQL workbench is free accessible tool for managing MySQL databases. That allows users to produce, design and model databases. MySQL workbench is used to create SQL queries for performing tasks on databases such as altering and getting data from databases.

### 2.7.4 Three tier architecture

Three-tier is one of the widely known and extremely adjustable structure for developing web application in credible and flexible way. The web application structure by 3-tier efficiently helps developers. It leads them to modify and reuse particular tier of an application without rewriting the entire codes of the application. The figure 2 below presents how three-tier architecture works.

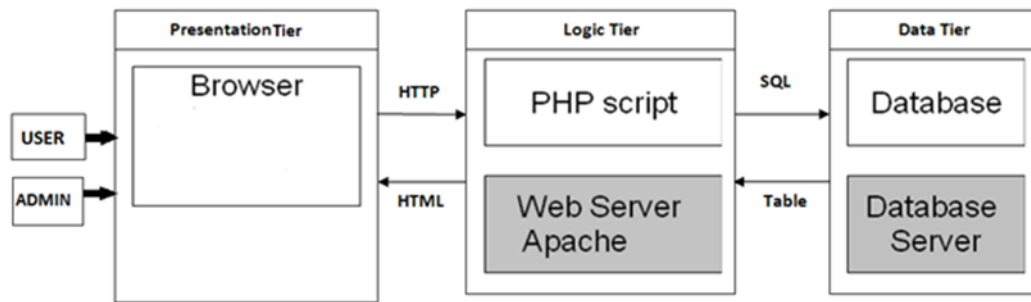


FIGURE 2. Three-tier architecture

As it is shown in the above figure the architecture consists of three layers which are:

1. Presentation tier that displays the web site to the client upon their request where they write in the browser. It will be sent as HTTP request to the web server and gets response as HTML by logic tier.
2. Logic tier is the middle layer that includes the system's logic. It is responsible for business logic of the system. This tier sends the client's request to the data tier and it does not allow clients to reach the database directly. In the development system for example administrator cannot access to the database directly, but he/she can manage data without access it.
3. Data tier is entitled to perform tasks on database like alter, delete, create and get operations. In the development project MySQL was used as a data server.

### 3 REQUIREMENTS AND SYSTEM DESIGN

#### 3.1 System overview

We are implementing web-based auction system in Iraq for selling and buying items. Users can domestically sign up and join to the auction through a web by using a web browser. Two different types of actors are using this system: user actor, which can be registered to the system and selling an item after uploading the item to the system or buying items by placing a bid. The bid must be higher than the start or the current bid. Administrative actor, which can login to the system with a particular username and password. Administrative performs many actions in the system that contain managing category, removing users, editing categories, etc. The functionality and more features are explained in more detail in this chapter.

The main characteristics of the system are:

- Each item on auction is distinguished by an id, name, a description, a picture, and a category (i.e. music, beauty, accessories, etc.)
- Every item on the auction has a commencement a time and date, and an end time.
- The user capable of adding an item on auction for sale, adjusting it and remove it by login to the auction site and accessing his/her personal page.
- A visitor can access the web site through a web browser and look at the items on auctioning but he/she unable to make a bid without register to the website.
- When a user signs up the system checks the national card number that he/she entered. An error message will occur if the national card number is wrong or if it has already existed in the system.

#### 3.2 Use case

For specifying the functionality of the system the use case was chosen. It provides a better view of the system. It shows the interaction between the user and the system. The use case is used for documenting the requirements of the system.

### 3.2.1 User use cases

The figure 3 below explains the user's use cases how the user will interact with the system.

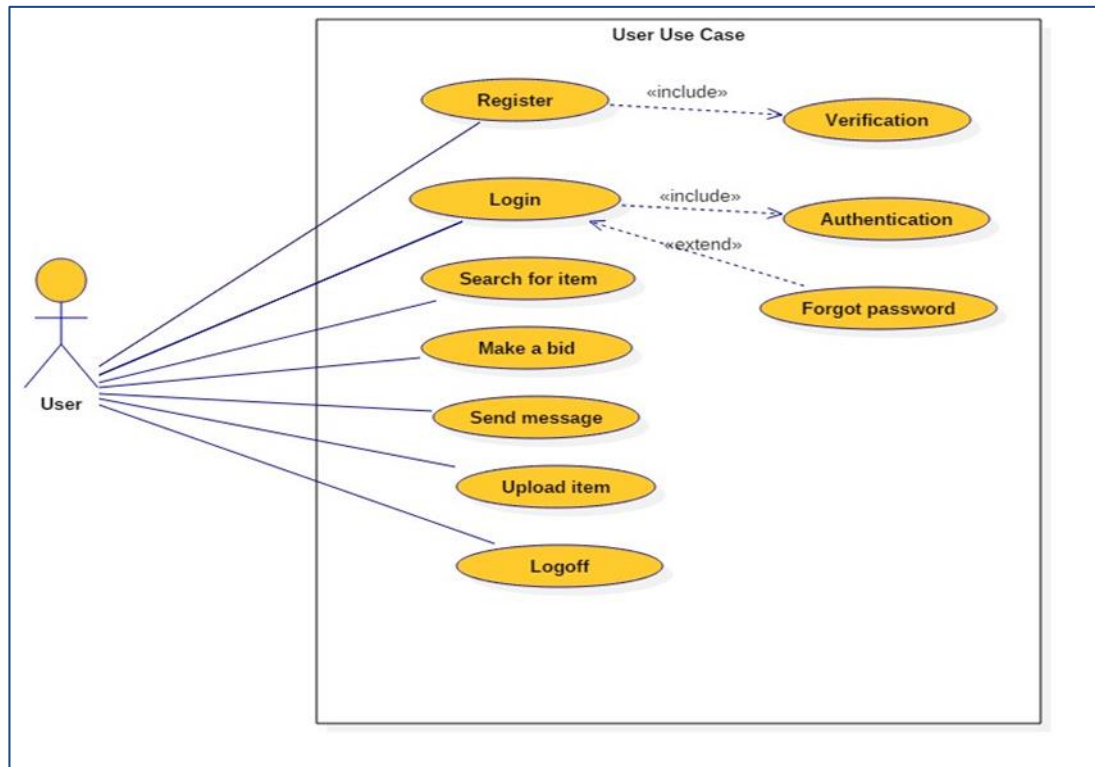


FIGURE 3. User use cases

The table 2 below is illustrated the index of user use cases.

TABLE 2. Index of user use cases

| Use case ID | Use case name      | Primary actor      | Scope | Priority |
|-------------|--------------------|--------------------|-------|----------|
| 1           | Register           | User               | in    | 1        |
| 2           | Verification       | User               | in    | 1        |
| 3           | Login              | User               | in    | 1        |
| 4           | Authentication     | The system         | in    | 1        |
| 5           | Forgot password    | User               | in    | 2        |
| 6           | Search for an item | User (as a bidder) | in    | 1        |

|    |                |                    |     |   |
|----|----------------|--------------------|-----|---|
| 7  | Make a bid     | User (as a bidder) | in  | 1 |
| 8  | Send a message | User               | in  | 2 |
| 9  | Upload an item | User (as a seller) | in  | 1 |
| 10 | Logoff         | User               | out | 1 |

### Use case 1: Register

**Primary actor(s):** generic user.

**Description:** the user must register into the system in order to take part to the auction.

**Pre-condition:** the user must have valid email and national card number.

**Basic use case flow:** the user fills the information that is required in the registration form, the main required information is:

- First name
- last name
- email
- national card number

### Main scenarios:

1. The user clicks on the sign up button.
2. The user provides his/her personal information and fills the form.
3. The user submits the form by clicking on the submit button.
4. If all the fields are filled and right national card number is entered the successful message will occur.
5. If user entered a fake national card number, the warning message will occur at the submission form time.



6. After submission, the account will be locked until the user confirms his/her email address by clicking on the confirmation link that will be sent to his/her email address.

**Exception:**

Invalid email address or national card number.

**Use case 2: Verification**

**Primary actor(s):** registered user.

**Description:** in order to activate an account, the user must confirm his/her account by clicking on the verification link that is sent to his/her email address.

**Pre-condition:** the user must have registered to the system.

**Basic use case flow:** the user confirms his/her email address by clicking on the verification email.

**Main scenarios:**

1. The system will send the verification email to the user's email address.
2. The user clicks on the verification link.
3. The user's email address was confirmed by the system.

**Exception:**

Invalid email address, national card number and/or username are already existed in the database on the server.

**Use case 3: Login**

**Primary actor(s):** user.

**Description:** whenever the user tries to access his/her personal page, buying an item, etc. he/she is required to provide his/her username and password.

**Pre-condition:** the user must have valid username and password.

**Basic use case flow:** the user writes his/her username and password in the login form and clicks the login button, the information is sent to the server for authentication.

**Main scenarios:**

1. The user fills the login form by writing his/her username and password.
2. The login information is sent to the server for authentication by clicking the login button.

**Exception:**

Invalid username or password.

**Use case 4: Authentication**

**Primary actor(s):** the system.

**Description:** whenever the user tries to login to his /her account by providing the username and password, the information is sent to the server for authentication.

**Pre-condition:** the user must have valid username and password.

**Basic use case flow:** the user writes his/her username and password in the login form and clicks the login button, the information is sent to the server for authentication.

**Main scenarios:**

1. The user fills the login form by writing his/her username and password.
2. The login information is sent to the server for authentication by clicking the login button.
3. The system checks the correctness of the information and responds by either valid or invalid.
4. If the username and password are valid, the user will successfully login to his /her account.
5. If the username and password are invalid, the user will get a message to check the username and password entered.

**Exception:**

None.

**Use case 5:** Forgot password

**Primary actor(s):** user.

**Description:** this use case is used in case the user forgot his/her password, the system provides him/her a new password.

**Pre-condition:** the user must have a valid email address stored in the system.

**Basic use case flow:** the user writes his/her email address; the information is sent to the system for validation. The email that is containing a link for creating new password will send to his/her email address, by clicking on the link he/she asking to provide the new password.

**Main scenarios:**

1. The user writes his/her email address which has given at registration time.
2. By clicking the forgot password button the email address is sent to the server for validation.
3. The system checks the correctness of the information and responds by either valid or invalid.
4. If the email address is valid, a link will send to user's email address.
5. By clicking on the link the user provides a new password.
6. If an email address is invalid, a message will occur asking the user to check entered email address.

**Exception:**

Invalid email address.

**Use case 6:** Search for an item

**Primary actor(s):** user, generic user

**Description:** this use case is used to search for the items on the auction by selecting its category or by entering a keyword on the search field. The registered and an unregistered user can access to this service.

**Pre-condition:** None.

**Basic use case flow:** the user searches on the items either by selecting a specific category or by entering a keyword on the search text box. On the base of his/her search a list of items will be shown.

**Main scenarios:**

1. The user selects the search type either by category or by a keyword.
2. If a user selects the specific category, then a list of items under this category will be shown with details.
3. If a user entered specific word on the search text box. The system will check the existence of this item, if it exists it will be shown with details.

**Exception:**

None.

**Use case 7: Make a bid**

**Primary actor(s):** user(bidder).

**Description:** this use case is used to let the user makes a bid on an item he/she interested in.

**Pre-condition:** the user has logged in.

**Basic use case flow:** the bidder chooses some item and make a bid on it. If the bid price is higher than the current bid the bid will be accepted and the item status is updated.

**Main scenarios:**

1. The bidder selects some item.
2. The system shows the item with its starting bid, current bid and the time left.

3. The bidder makes a bid by entering the price which must be higher than the current price.
4. By pressing the increase bid, the status of the item is updated and the highest bid will be the current bid.
5. If another bidder makes a bid its bid price will be the current highest bid, the system informs the bidders and asks to rebid.

**Exception:**

The bid price is lower than the current price.

**Use case 8:** Send a message

**Primary actor(s):** User.

**Description:** This use case is used to let the users send messages with title and content to each other by entering the username of the receiver. The message will be shown on the inbox of the receiver's personal page.

**Pre-condition:** the user has logged in.

**Basic use case flow:** the user goes to his/her account and selects a message on his/her personal page. She writes the message with title and content and sends it to another user by clicking on the send message button.

**Main scenarios:**

1. The user opens his/her personal page.
2. She /he selects a message on the personal page.
3. The user fills the empty form, writing the title and content of the message.
4. The username of receiver is written on the message.
5. The message is sent to the receiver by clicking on the send message button.

6. The message will appear in the message inbox of the receiver.

**Exception:**

Invalid username of the receiver.

**Use case 9:** Upload an item

**Primary actor(s):** user(seller)

**Description:** the seller uploads an item by selecting its category, description, picture, etc. The seller specifies the start bid price and starting time.

**Pre-condition:** the user has logged in.

**Basic use case flow:** the user goes to his/her personal page and clicks on the upload an item button, filling the item's form by writing the description, starting bid price, starting time, ending time and uploading item's picture.

**Main scenarios:**

1. The user opens his/her personal page.
2. She /he selects upload an item on the personal page.
3. The user fills the empty form, writing name, description, its category, starting time, starting bid price and ending time.
4. The item is uploaded by clicking on the make auction button.
5. The system accepts the item and presents it.

**Exception:**

If the seller does not write a description about an item or does not upload the item's photo.

**Use case 10: Logoff**

**Primary actor(s):** user.

**Description:** the use case used to sign out the user on the system.

**Pre-condition:** the user has logged in.

**Basic use case flow:** the user logged off from the system by clicking on the logoff button, the user's information removed from local storage.

**Main scenarios:**

1. The user clicks on the logoff button
2. His/her information removed on the local storage.
3. The system leads the user to the home page.

**Exception:**

None.

### 3.2.2 Administrative use cases

The figure 4 below shows the use cases of the activities of the administrative with the system.



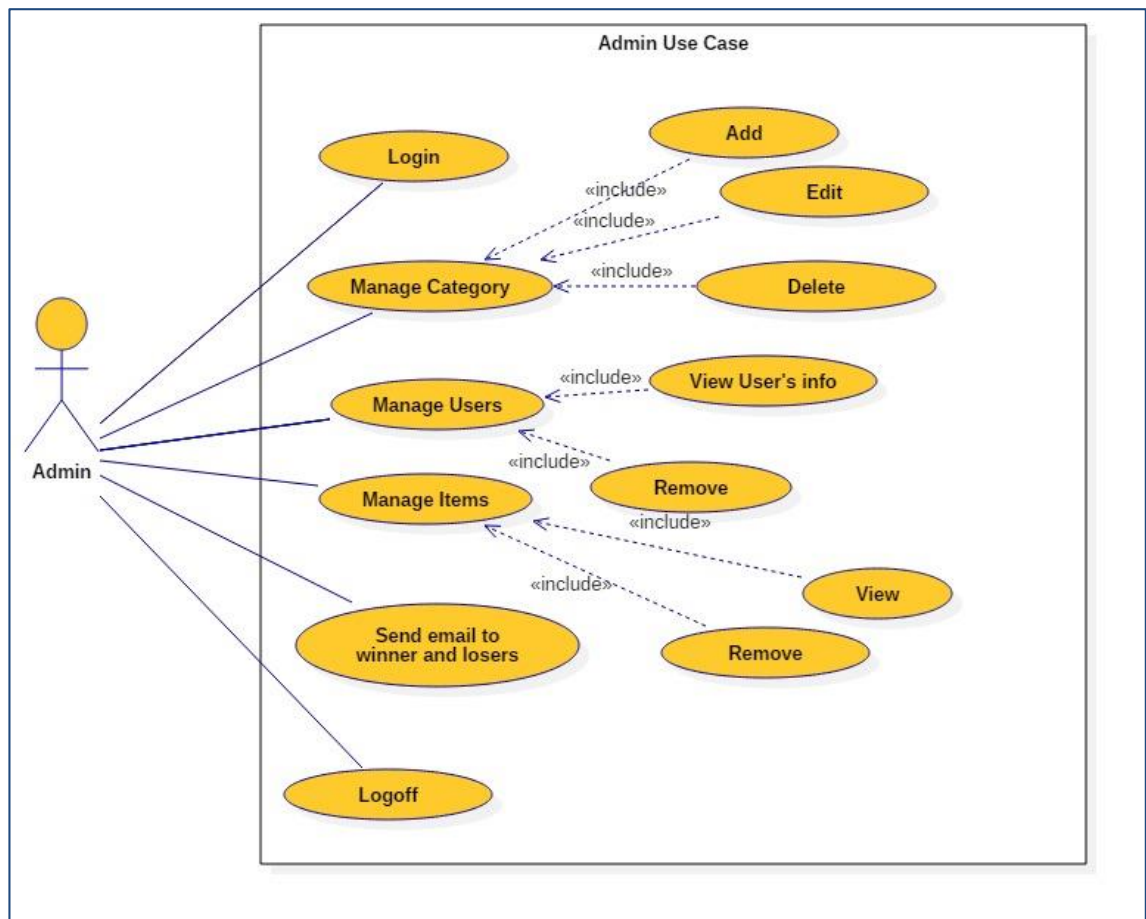


FIGURE 4. Admin use cases.

The table 3 below illustrates the admin use cases.

TABLE 3. Index table of admin use cases

| Use case ID | Use case name                       | Primary actor | Scope | Priority |
|-------------|-------------------------------------|---------------|-------|----------|
| 1           | Login                               | Admin         | in    | 1        |
| 2           | Manage category                     | Admin         | in    | 1        |
| 3           | Manage users                        | Admin         | in    | 1        |
| 4           | Manage items                        | Admin         | in    | 1        |
| 5           | Send email to the winner and losers | Admin         | in    | 2        |
| 6           | Logoff                              | Admin         | out   | 1        |

**Use case 1: Login**

**Primary actor(s): admin.**

**Description:** this use case is used to allow the admin access to the administrator page. He/she required to provide his/her username and password.

**Pre-condition:** the admin must have valid username and password.

**Basic use case flow:** the admin provides username and password and access to his/her page by clicking on the login button.

**Main scenarios:**

1. The admin provides his/her username and password. He /she clicks on the login button.
2. The login information is sent to the server for validation.
3. After successfully login, the admin access to administrator page.

**Exception:**

Invalid username or password.

**Use case 2:** Manage category

**Primary actor(s):** admin.

**Description:** this use case is used to add, edit and delete a category by admin.

**Pre-condition:** the admin has logged in.

**Basic use case flow:** the admin can access and modify the category's data stored on the database, by clicking on the category button from the main menu the list of a categories

will be shown. He/she able to add the new category, edit the information of a category and delete a category without directly access to the database.

**Main scenarios:**

1. The admin clicks on the category button.
2. The list of stored categories retrieved from the database.
3. In case of ‘add new category’, the admin must provide the name of new category and clicks on the add new category’s button.
4. In case of ‘edit category’, the admin is able to open the list of categories and edit category’s name by clicking on the edit category’s button.
5. In case of ‘delete category’, the admin has a right to delete some categories if no items exist under this category.

**Exception:**

Invalid username or password

**Use case 3:** Manage users

**Primary actor(s):** admin.

**Description:** this use case is used to manage the users; view and delete by admin.

**Pre-condition:** the admin has logged in.

**Basic use case flow:** the admin can view the user’s information and delete a user from the system if the user is created with a fake account.

**Main scenarios:**

1. The admin views the user's information by clicking on the view info.
2. The list of registered users is retrieved from the database.
3. The admin can view the information of the user, such as items uploaded, make bid and personal info.
4. If it seems like a fake account and does not have any activity, the admin has a right to delete it by clicking on the delete user button.

**Exception:**

Invalid username or password.

**Use case 4:** Manage items

**Primary actor(s):** admin.

**Description:** this use case is used to view and delete an item by admin.

**Pre-condition:** the admin has logged in.

**Basic use case flow:** the admin can view the information about each item on the system. If the item was sold or reached its ending time, it will be removed from the database.

**Main scenarios:**

1. The admin requested to view items by specific date.
2. The list of stored item retrieved from database by this date.
3. In case of 'sold item' or 'ended', the admin deletes the item by clicking on the delete item button.

**Exception:**

Invalid username or password.

**Use case 5:** Send email to winner and losers

**Primary actor(s):** admin.

**Description:** this use case is used to send emails after the ending of the auction by admin.

**Pre-condition:** the admin has logged in.

**Basic use case flow:** the admin is able to view the ongoing auctions and ended auctions. If the auction was ended the admin sends email to the participants and informs them about ending auctions.

**Main scenarios:**

1. The admin views the list of ongoing auctions.
2. Each of ongoing auctions have the end time.
3. After the auction was ended the email will be sent to all bidders.

**Exception:**

Invalid username or password.

**Use case 6:** Logoff

**Primary actor(s):** admin.

**Description:** the use case used to sign out the admin on the system.

**Pre-condition:** the admin has logged in.

**Basic use case flow:** the admin logged off from the system by clicking on the logoff button.

**Main scenarios:**

1. The admin clicks on the logoff button
2. His/her information removed on the local storage.
3. The system leads the admin to the admin's home page.

**Exception:**

None.

### 3.3 UI layout

As I mentioned before, this system will be the first online auction system in Iraq to be implemented. It is very important to choose the simple and common UI elements. Simple user interface makes the user feel more comfortable and capable of getting things done more quickly. The figure 5 below is a prototype of the system. As it is shown in the figure the home page has two buttons at the top left-hand side. They are login and register buttons. When the user is clicking on the register button, it takes the user to register page. By filling the required fields in the registration form and pressing the register button the user is able to create an account.

The second button is login button, when the user is clicking on the login button is taking the user to login view by entering the username and password and pressing the login button takes the user to the home page. As it is shown on the prototype the list of category exists on the right-hand side of the home page. The search field is located on the

top of the home page. the current ongoing items on auction are showing in the body of the home page.

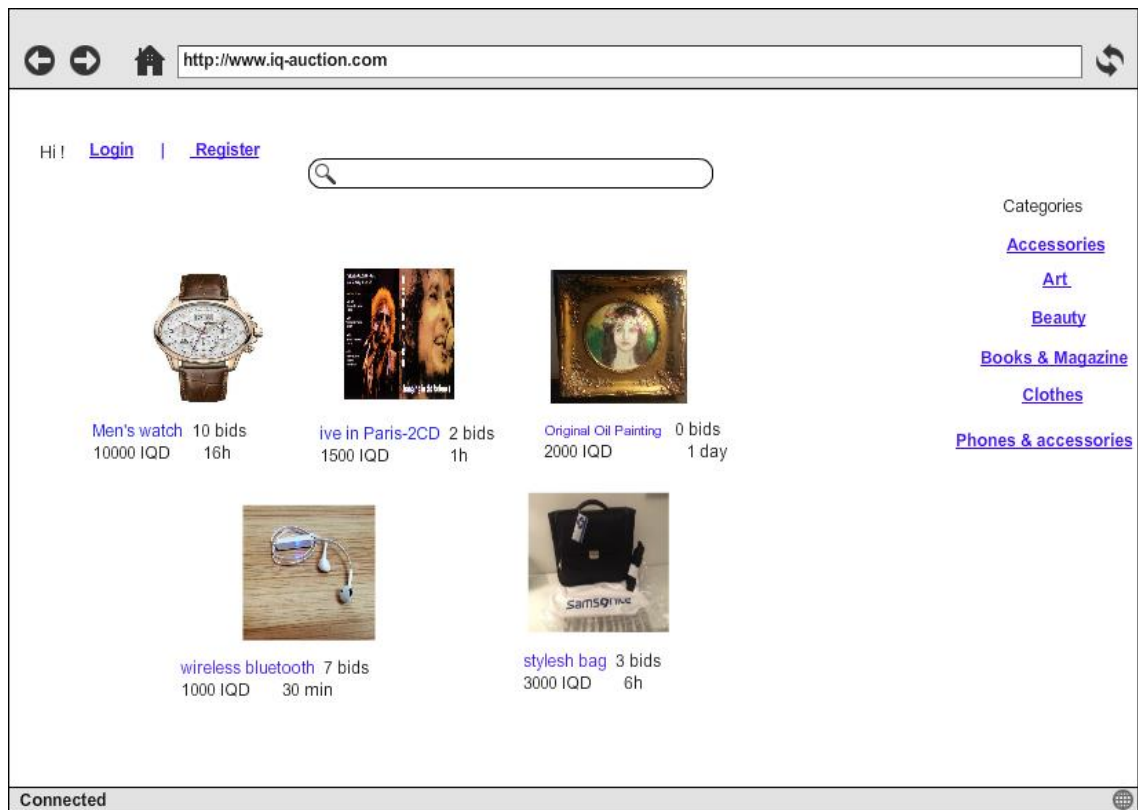


FIGURE 5. Prototype of system.

### 3.4 Class diagram

For presenting the design phase of the system class diagram was chosen. the object-oriented programming language was used by the author in the system development. Therefore, the class diagram was suitable to show the classes of the system, the attributes, operations and relationships of the classes. Figure 6 below is the class diagram of the system.

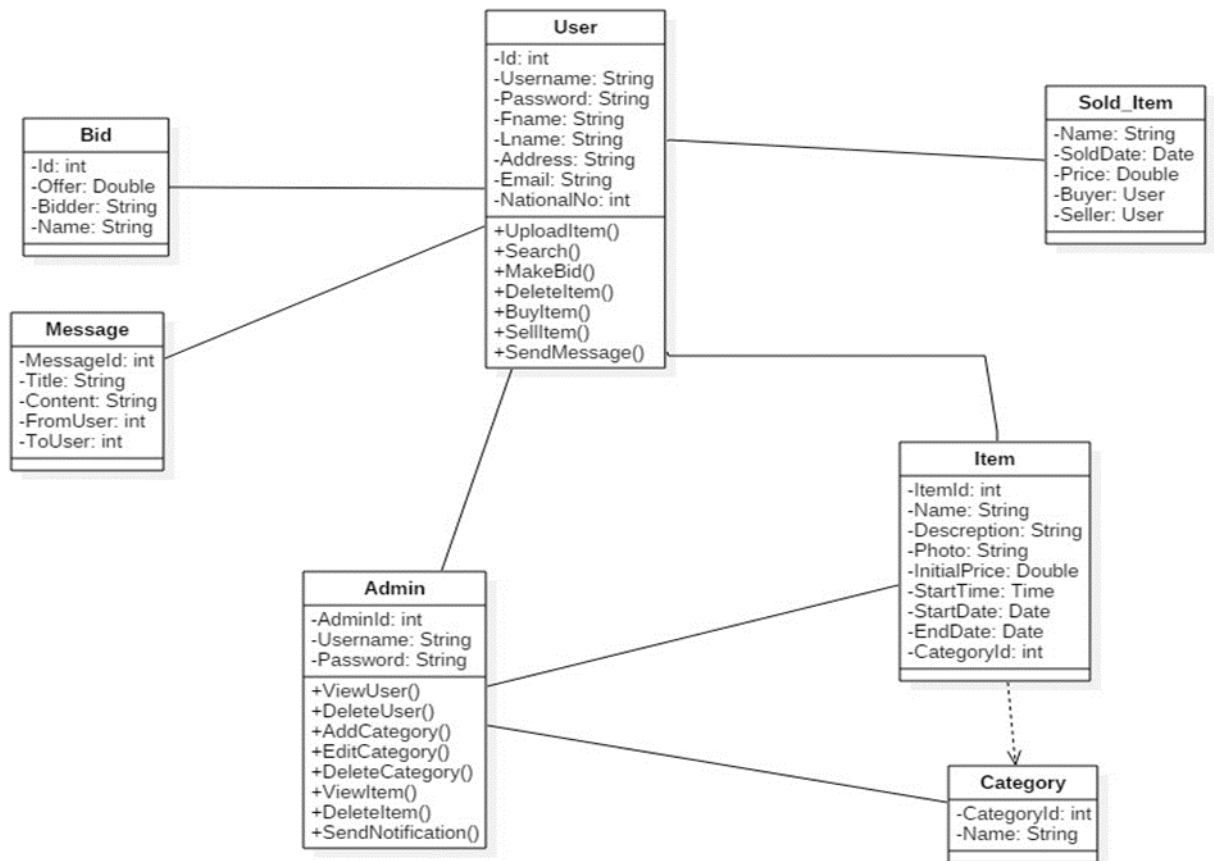


FIGURE 6. Class diagram

The class user contains multiple parameters (id, username, first name, last name, etc.) which are information about registered user on the system. The user class also contains several methods which are performed the operations of the user, the same user can bid for an item and upload an item for auction (uploading an item, search for the item, bid on the item, send a message to another user, etc.). The class bid has a bid id, the name of the item that placed to bid, the offer bid which is made by a user as a bidder, the name of the user which makes the bid and the item's name. The class item also has several parameters which are the information about the item that user uploads to the auction item id, item name, the description of the item, the name of the category it belongs to and so on. The class message has title of message, the message content, the id of the sender and the id of the receiver. The class sold item shows the information about items that were sold, its parameters are the name of sold item, the date it was sold, its price and id of seller and buyer. Finally, the class admin which has admin name and its pass-



word as a parameter and the methods that performs several operations such as: add category for adding new category, delete user for removing the user from the system, viewing item and sending notifications for users.

### **3.5 ER-Diagram**

For designing database ER diagram was chosen. Since the MySQL is relational database therefore ER modeling is very useful to design the database schema. The ER diagram clearly represents the relations between tables. The ER diagram of the system is represented in figure 7 below.

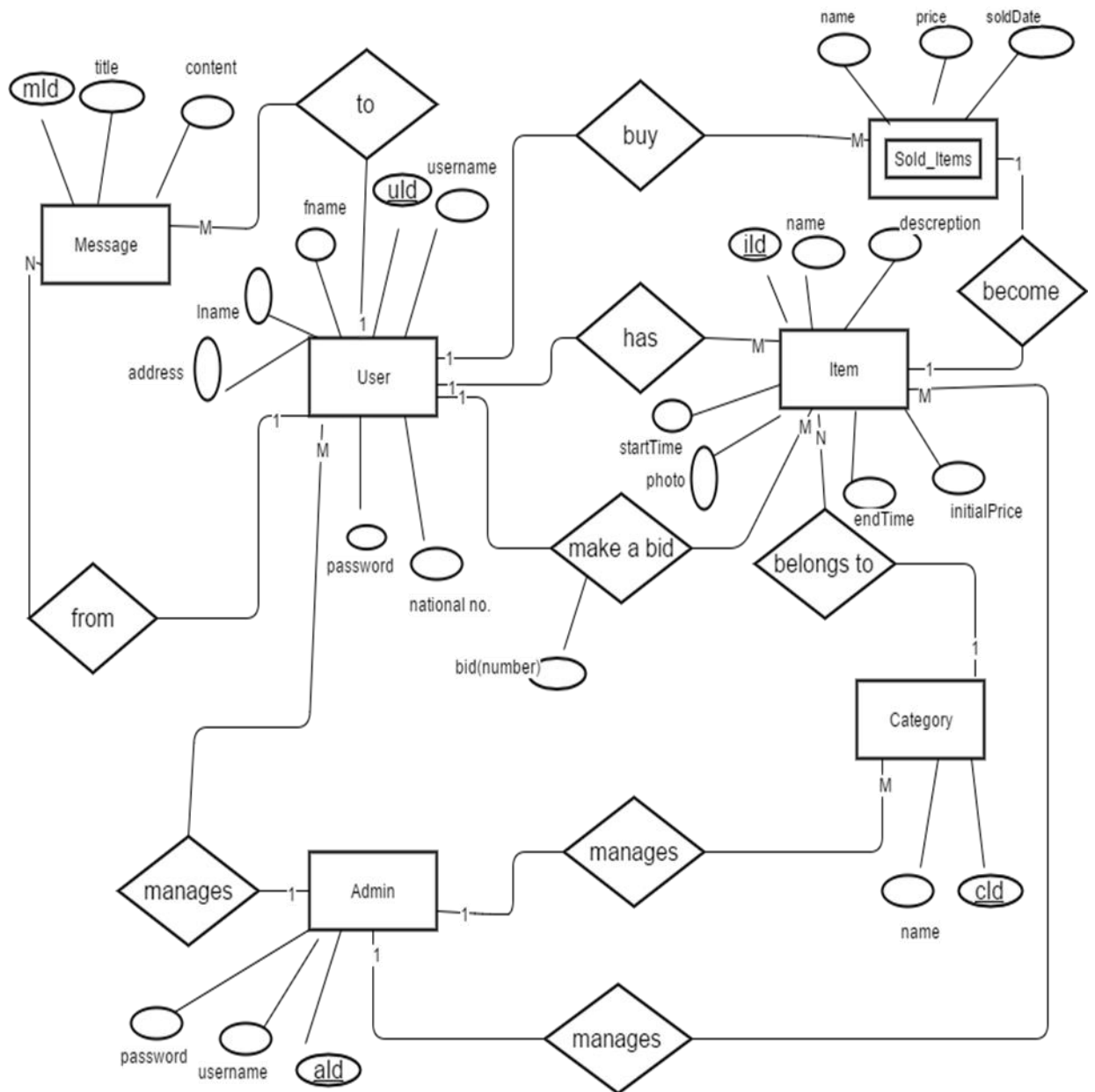


FIGURE 7. ER-diagram of the system.

The ER-diagram above has several entities (user, item, item, category...) with their attributes and relations between entities. Each user has one or more items, he/she is able to make a bid for one or more items, upload one or more items to the auction. Each item owns by one user and belongs to one category. The sold item is weak entity which has no its own key attribute, a sold item is the item when it is sold become sold item. The admin manages categories, users and items. The users are able to send message to each other; the message is sent from one user to another one user.

## 4 SYSTEM ARCHITECTURE AND SOFTWARE TESTING

### 4.1 System Architecture

The architecture of system is split into two sides; the client side which contains the user interface, and the server side which contains the web pages were written by PHP and MySQL database. By SQL queries that are written inside PHP pages can access to the database. The technologies were used in both sides were explained in more details in chapter 2. Figure 8 below shows the architecture of the system.

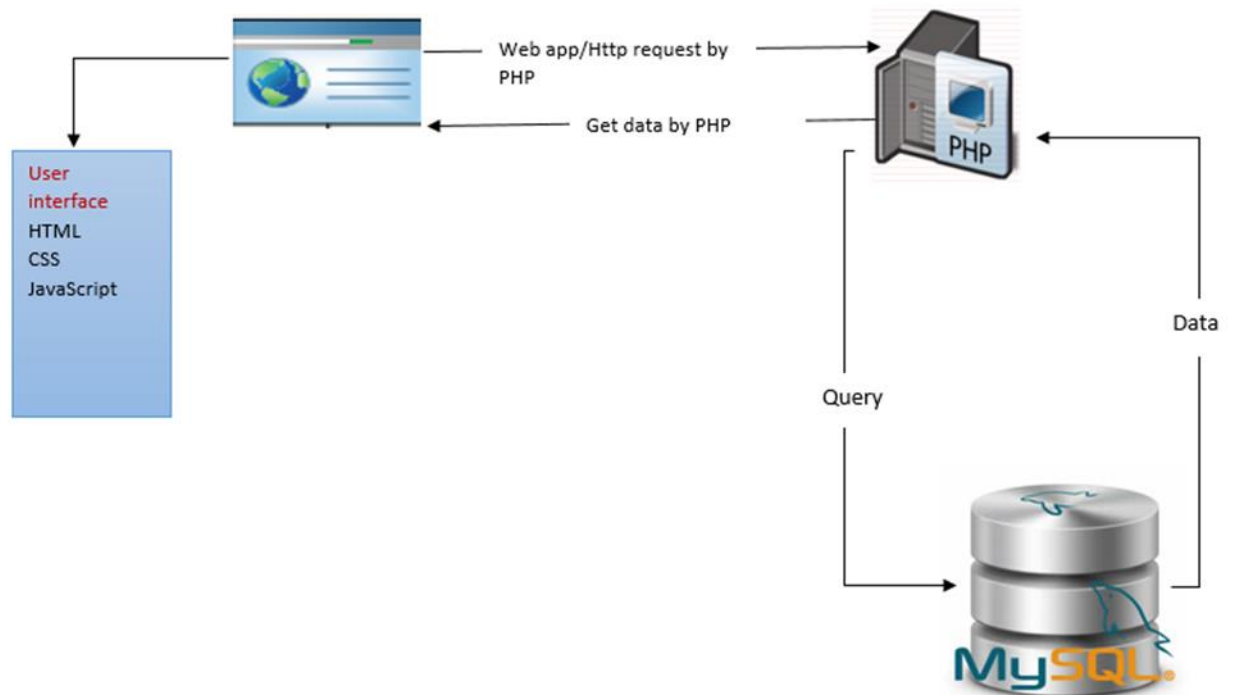


FIGURE 8. System architecture.

### 4.2 System testing

System testing is a significant phase of Software Quality Assurance. It represents the last inspection of design and coding. Testing is a process of performing a program with

the purpose of finding an error. The goal of testing is to point out and fix bugs in the developed system. The system is not complete without testing. Testing is the essential step to the success of the system. There are many approaches to system testing. The system testing was done based on integration testing to verify, whether the developed system meets the requirements or not. The testing of the system is shown on the table 4 below.

TABLE 4. System testing

| Test case ID | Test case name    | Precondition          | Priority | Input test data                         | Steps to be executed   | Expected result                              | Pass / Fail |
|--------------|-------------------|-----------------------|----------|---|--|--|-------------|
| 1            | User registration |                       | high     | Correct national card number.           | 1. User fills the required fields of registration form.<br>2. click 'Register' button. | Successfully registered & verify your email. | Pass        |
|              |                   |                       |          | Wrong or duplicate national card number |  | The system displays an error message.        | Fail        |
| 2            | User Login        | Registered previously | high     | Correct username & password             | 1. Enter username & password<br>2. click 'login' button.                               | The user successfully logged into system.    | Pass        |
|              |                   |                       |          | Wrong username and/or password          |  | The system displays an error message.        | Fail        |

## 5 DISCUSSION

When the company asked me to develop this system, at the first I was a little bit worried. There were a plenty of questions came to my mind. The first thing which thinking about was user's requirements and how they will interact with this system. The system which is going to be built should be such kind of systems that fills the user's requirements. But a big problem was the IT sectors until are not so strong in Iraq until now. Therefore, it was not so easy to build such a system like online ones there. For that reason, it was very important to develop simple and easy web application that the users easily will be familiar with it and they know how to use it. At the very first step it was significant to know about auctions sales in general and online auctions around the web. The history of auction, the features of some online auctions were gave me the idea of auction sales and helped me to find requirements soon. The main requirements that found was very useful to planning, designing phase and testing the system. Because the requirements were focused on the real needs.

After planning the system, it was important to define which technologies were going to be used for developing this system. Then reviewing the existence similar systems around the web were helpful a lot to know which types of technologies are required for this kind of web systems. The windows operating system was chosen. Therefore, the WAMP server software stack was preferable to be used. The WAMP server is helping the choices of other technologies like Apache, PHP and MySQL, so the server side programming languages were ready for the web application. For choosing client side programming it was better to choice such programing languages that are easily interact with the PHP. So HTML along with CSS and JavaScript were the best choice.

For writing and running software codes without doubt is required the software tools. The open source software tools were chosen which they were keeping the low development cost.

For knowing the behaviour of the system the use case was used. The use cases process was defining how the user will interact with the system. And how the system response with each use cases. The class diagram was used for previewing the structure of the system and defining which class contains what? Of course the system was needed to design

the database of it. ER diagram was used for handling database design. Since MySQL is the relational database and it was used as a system database. For that ER diagram was useful for designing MySQL database. The most significant phase in each web application is the user interface of it. Due to the fact that the user interface is the first point where the users are interacted with the system. For that reason, simple and easy user interface was defined for the developed system. The UI layout was created by using some mock-up tools for defining the basic requirements of the UI.

As it is clear testing should be provided for every software system, because it is going to clear the software from any bugs if it exists in the system. Some pieces of software codes were tested and was successfully passed from the test without any problem.

This system developing was expertise my knowledge, and how to handle similar working life problems in this area. This project was done successfully but it needs more works. But for this stage it was perfect, because it is quite important to start with small clear project and step by step provides it by new features.

## REFERENCES

Online auction types. Read 20.05.2016. [https://en.wikipedia.org/wiki/Online\\_auction](https://en.wikipedia.org/wiki/Online_auction)

Ralph Cassidy, Jr. Auctions and Auctioneering, University of California Press, 1967.

Philip A. Benson, Nelson L. North, Real Estate Principles and Practices, New York, 1924.

The features of eBay. Read 12.11.2016. <http://www.ebay.com/>

The features of Huuto. Read 15.11.2016. <http://www.huuto.net/>

Apache HTTP server. Read 22.05.2016.  
[http://en.wikipdia.org/wiki/Apache\\_HTTP\\_Server](http://en.wikipdia.org/wiki/Apache_HTTP_Server)

Cascading Style Sheets. Read 1.06.2016. [http://www.w3schools.com/css/css\\_intro.asp](http://www.w3schools.com/css/css_intro.asp)

Hypertext Mark-up Language, Introduction. Read 29.05.2016.  
[http://www.w3schools.com/html/html\\_intro.asp](http://www.w3schools.com/html/html_intro.asp)

JavaScript programming language, JS tutorial. Read 03.06.2016.  
<http://www.w3schools.com/js/>

MySQL, Open source database. Read 05.06.2016.  
<https://dev.mysql.com/doc/refman/5.7/en/what-is-mysql.html>

WAMP server, overview. Read 15.06.2016.  
<https://dev.mysql.com/doc/refman/5.7/en/what-is-mysql.html>

MySQL Workbench, Visual Database Management Tool. Read 20.06.2016.  
<http://www.mysql.com/products/workbench/>

Three tire architecture with PHP and MySQL, tutorial. Read 01.07.2016.  
[http://www.brainbell.com/tutors/php/php\\_mysql/Three\\_Tier\\_Architectures.html](http://www.brainbell.com/tutors/php/php_mysql/Three_Tier_Architectures.html)

PHP, what is PHP. Read 04.07.2016. <http://php.net/manual/en/intro-what-is.php>

## APPENDICES

1(7)

## Appendix 1. Database SQL script

```

SET @OLD_UNIQUE_CHECKS=@@UNIQUE_CHECKS, UNIQUE_CHECKS=0;
SET @OLD_FOREIGN_KEY_CHECKS=@@FOREIGN_KEY_CHECKS, FOREIGN_KEY_CHECKS=0;
SET @OLD_SQL_MODE=@@SQL_MODE, SQL_MODE='TRADITIONAL, ALLOW_INVALID_DATES';
CREATE SCHEMA IF NOT EXISTS `dbauction` DEFAULT CHARACTER SET utf8;
USE `dbauction`;

```

```

-----
-- Table `admin`
-----

```

```

DROP TABLE IF EXISTS `admin`;

```

```

CREATE TABLE IF NOT EXISTS `admin` (
  `admin_id` INT (11) NOT NULL AUTO_INCREMENT,
  `name` VARCHAR (45) NULL DEFAULT NULL,
  `password` VARCHAR (45) NULL DEFAULT NULL,
  PRIMARY KEY (`admin_id`))
ENGINE = InnoDB
AUTO_INCREMENT = 6
DEFAULT CHARACTER SET = utf8;

```

```

-----
-- Table `category`
-----

```

```

DROP TABLE IF EXISTS `category`;

```

```

CREATE TABLE IF NOT EXISTS `category` (
  `category_id` INT (11) NOT NULL AUTO_INCREMENT,
  `name` VARCHAR (45) NULL DEFAULT NULL,

```



```

`administrator` INT (11) NULL DEFAULT NULL,
PRIMARY KEY (`category_id`),
CONSTRAINT `fk_Category_Admin1`
FOREIGN KEY (`administrator`)
REFERENCES `admin` (`admin_id`)
ON DELETE SET NULL
ON UPDATE CASCADE)
ENGINE = InnoDB
AUTO_INCREMENT = 10
DEFAULT CHARACTER SET = utf8;

CREATE INDEX `fk_Category_Admin1_idx` ON `category` (`administrator` ASC);

```

```

-----
-- Table `user`
-----

```

```

DROP TABLE IF EXISTS `user`;

```

```

CREATE TABLE IF NOT EXISTS `user` (
  `user_id` INT (11) NOT NULL AUTO_INCREMENT,
  `username` VARCHAR (45) NOT NULL,
  `fname` VARCHAR (45) NULL DEFAULT NULL,
  `lname` VARCHAR (45) NULL DEFAULT NULL,
  `email` VARCHAR (45) NULL DEFAULT NULL,
  `password` VARCHAR (45) NULL DEFAULT NULL,
  `address` VARCHAR (45) NULL DEFAULT NULL,
  `national_card_no` VARCHAR (45) NOT NULL,
  `administrator` INT (11) NULL DEFAULT NULL,
  PRIMARY KEY (`user_id`),
  CONSTRAINT `fk_User_Admin`
  FOREIGN KEY (`administrator`)
  REFERENCES `admin` (`admin_id`)
  ON DELETE SET NULL
  ON UPDATE CASCADE)

```

ENGINE = InnoDB

3(7)

AUTO\_INCREMENT = 17

DEFAULT CHARACTER SET = utf8;

CREATE INDEX `fk\_User\_Admin\_idx` ON `user` (`administrator` ASC);

-----  
 -- Table `item`  
 -----

DROP TABLE IF EXISTS `item`;

CREATE TABLE IF NOT EXISTS `item` (  
 `item\_id` INT (11) NOT NULL AUTO\_INCREMENT,  
 `category` INT (11) NOT NULL DEFAULT '0',  
 `name` VARCHAR (45) NOT NULL DEFAULT NULL,  
 `photo` LONGBLOB NOT NULL DEFAULT NULL,  
 `description` TEXT NOT NULL DEFAULT NULL,  
 `start\_time` TIMESTAMP NULL DEFAULT NULL,  
 `end\_time` TIMESTAMP NULL DEFAULT NULL,  
 `initial\_price` DECIMAL (10,2) NULL DEFAULT NULL,  
 `seller` INT (11) NULL DEFAULT NULL,  
 `manage\_by` INT (11) NULL DEFAULT NULL,  
 `auction\_status` VARCHAR (50) NULL DEFAULT NULL,  
 PRIMARY KEY (`item\_id`, `category`),  
 CONSTRAINT `fk\_Item\_Admin1`  
 FOREIGN KEY (`manage\_by`)  
 REFERENCES `admin` (`admin\_id`)  
 ON DELETE SET NULL  
 ON UPDATE CASCADE,  
 CONSTRAINT `fk\_Item\_Category1`  
 FOREIGN KEY (`category`)  
 REFERENCES `category` (`category\_id`)  
 ON DELETE CASCADE  
 ON UPDATE NO ACTION,

```

CONSTRAINT `fk_Item_User1`
  FOREIGN KEY (`seller`)
  REFERENCES `user` (`user_id`)
  ON DELETE SET NULL
  ON UPDATE CASCADE)
ENGINE = InnoDB
AUTO_INCREMENT = 36
DEFAULT CHARACTER SET = utf8;

CREATE INDEX `fk_Item_Category1_idx` ON `item` (`category` ASC);

CREATE INDEX `fk_Item_User1_idx` ON `item` (`seller` ASC);

CREATE INDEX `fk_Item_Admin1_idx` ON `item` (`manage_by` ASC);

-----
-- Table `bid`
-----

DROP TABLE IF EXISTS `bid`;

CREATE TABLE IF NOT EXISTS `bid` (
  `bid_id` INT (11) NOT NULL AUTO_INCREMENT,
  `offer` DECIMAL (10,2) NULL DEFAULT NULL,
  `bidder` INT (11) NOT NULL,
  `item` INT (11) NOT NULL,
  PRIMARY KEY (`bid_id`),
  CONSTRAINT `fk_bid_item1`
    FOREIGN KEY (`item`)
    REFERENCES `item` (`item_id`)
    ON DELETE CASCADE
    ON UPDATE CASCADE,
  CONSTRAINT `fk_bid_user1`
    FOREIGN KEY (`bidder`)
    REFERENCES `user` (`user_id`)

```

```

    ON DELETE CASCADE
    ON UPDATE CASCADE)
ENGINE = InnoDB
AUTO_INCREMENT = 16
DEFAULT CHARACTER SET = utf8;

CREATE INDEX `fk_bid_user1_idx` ON `bid` (`bidder` ASC);

CREATE INDEX `fk_bid_item1_idx` ON `bid` (`item` ASC);

-----
-- Table `message`
-----

DROP TABLE IF EXISTS `message`;

CREATE TABLE IF NOT EXISTS `message` (
  `message_id` INT (11) NOT NULL AUTO_INCREMENT,
  `title` TEXT NULL DEFAULT NULL,
  `content` TEXT NULL DEFAULT NULL,
  `from_user` INT (11) NULL DEFAULT NULL,
  `to_user` INT (11) NULL DEFAULT NULL,
  PRIMARY KEY (`message_id`),
  CONSTRAINT `fk_Message_User1`
    FOREIGN KEY (`from_user`)
    REFERENCES `user` (`user_id`)
    ON DELETE SET NULL
    ON UPDATE CASCADE,
  CONSTRAINT `fk_Message_User2`
    FOREIGN KEY (`to_user`)
    REFERENCES `user` (`user_id`)
    ON DELETE SET NULL
    ON UPDATE CASCADE)
ENGINE = InnoDB
AUTO_INCREMENT = 4

```

```
DEFAULT CHARACTER SET = utf8;
```

6(7)

```
CREATE INDEX `fk_Message_User1_idx` ON `message` (`from_user` ASC);
```

```
CREATE INDEX `fk_Message_User2_idx` ON `message` (`to_user` ASC);
```

```
-----  
-- Table `sold_items`  
-----
```

```
DROP TABLE IF EXISTS `sold_items`;
```

```
CREATE TABLE IF NOT EXISTS `sold_items` (  
  `item_id` INT (11) NOT NULL,  
  `name` VARCHAR (45) NULL DEFAULT NULL,  
  `sold_date` DATE NULL DEFAULT NULL,  
  `price` DECIMAL (10,2) NULL DEFAULT NULL,  
  `buyer` INT (11) NULL DEFAULT NULL,  
  PRIMARY KEY (`item_id`),  
  CONSTRAINT `fk_Sold_Item_Item1`  
    FOREIGN KEY (`item_id`)  
    REFERENCES `item` (`item_id`)  
    ON DELETE CASCADE  
    ON UPDATE CASCADE,  
  CONSTRAINT `fk_Sold_Item_User1`  
    FOREIGN KEY (`buyer`)  
    REFERENCES `user` (`user_id`)  
    ON DELETE SET NULL  
    ON UPDATE CASCADE)  
ENGINE = InnoDB  
DEFAULT CHARACTER SET = utf8;
```

```
CREATE INDEX `fk_Sold_Item_User1_idx` ON `sold_items` (`buyer` ASC);
```

```
SET SQL_MODE=@OLD_SQL_MODE;
```

7(7)

```
SET FOREIGN_KEY_CHECKS=@OLD_FOREIGN_KEY_CHECKS;
```

```
SET UNIQUE_CHECKS=@OLD_UNIQUE_CHECKS;
```