MINISTRY OF SCIENCE AND HIGHER EDUCATION OF THE RUSSIAN FEDERATION
Federal State Autonomous Educational Institution of Higher Education
“South Ural State University (National Research University)”
School of Electrical Engineering and Computer Science
Department of Computer Science

THESIS IS CHECKED

Reviewer,
Director of the planning department
JSC "Digital Iron Pipe"

_____ O.N. Fedianin

“___”_____ 2019

ACCEPTED FOR THE DEFENSE

Head of the department,
Dr. Sci., Prof.

_____ L.B. Sokolinsky

“___”_____ 2019

DEVELOPMENT OF WEB-APPLICATION FOR DIGITAL IMAGE PROTECTION

GRADUATE QUALIFICATION WORK
SUSU–02.04.02.2019.308-642.GQW

Supervisors:
Cand. Sci., Assoc. Prof.

_____ O.N. Ivanova

Senior Lecturer

_____ R.S. Fedyanina

Author,
student of the group CE-229
_____ H.A. Alkhattan

Normative control
_____ O.N. Ivanova
“___”_____ 2019

Chelyabinsk–2019

MINISTRY OF SCIENCE AND HIGHER EDUCATION OF THE RUSSIAN FEDERATION
Federal State Autonomous Educational Institution of High Education
“South Ural State University (National Research University)”
School of Electrical Engineering and Computer Science
Department of Computer Science

APPROVED

Head of the department,
Dr. Sci., Prof.

_____ L.B. Sokolinsky

“___”_____ 2019

TASK
of the master graduate qualification work

for the student of the group CE-229
Alkattan Hussein Ali Ibrahim
in master direction 02.04.02
“Fundamental Informatics and Information Technologies”
(master program “Database Technologies”)

1. **The topic** (approved by the order of the rector from __.__.2019 No. ___)
Development of web-application for digital image protection.
2. **The deadline for the completion of the work:** 05.06.2019.
3. **The source data for the work**

3.1. Kohan B. Guide to Web Application Development. [Электронный ресурс] URL: https://www.comentum.com/guide-to-web-application-dev

3.2. Cope G. Tips and Techniques to Protect Images on the Internet [Электронный ресурс] URL: http://www.naturefocused.com/articles/image-protectio

3.3. Reza M.D., Khan M.S.A., Alam M.G.R., Islam S. An Approach of Digital Image Copyright Protection by Using Watermarking Technology [Электронный ресурс] URL: https://arxiv.org/ftp/arxiv/ppers/1205/1205.6229.pdf

3.4. MS SQL Server, tutorialspoint simply easy learning.

4. **The list of the development issues**

4.1. To analyze the subject area and technologies for creating applications with a web interface.
4.2. To study and select the most suitable for the current task algorithms for embedding digital watermarks into images.
4.3. Develop a web application to protect images using a digital watermark.
4.4. Test developed web application.

5. Issuance date of the task: 08.02.2019.

Supervisor

Cand. Sci., Assoc. Prof.	O.N. Ivanova
Senior Lecturer	R.S. Fedyanina
The task is taken to perform	H.A. Alkhattan

TABLE OF CONTENTS

INTRODUCTION

When considering digital watermark image protection, it is very important to explicitly define and understand some related terms such as information security, cipher, steganography, watermark and digital watermarking. The description of these terms are provided as follows.

Information Security

Information security has become the subject of great interest by researchers and interested parties who are trying to obtain new and updated solutions and techniques to ensure the protection of information transmitted and received through the internet without any penetration or disclosure by the interveners [18]. Therefore, it was necessary to keep pace with the development of information security and the establishment of advanced techniques and methods, hence the emergence of Information Hiding and the development of the adoption of technical concealment Steganography.

Encryption is a method of protection that makes incoming and outgoing data invisible by hiding certain messages within a particular cover. The objective of the concealment process is not to raise any point of doubt about the presence of hidden data, whereas the goal of the concealment analyzer is to suspect all messages sent, and check them for hidden data. Steganalysis is a process of detecting, reading, changing and deleting hidden information [21].

The hidden or invisible messages are all in the same meaning, where the sender and the receiver only know the message. We can translate this into the word "art of hiding the word." This art has been used for hundreds of years [5]. Kings and princes used slaves to write the letter and used animals for this purpose [21]. Then appeared hidden ink in the picture and became a means of hiding the new writing and when the emergence of computers and technology entered the art of concealment into a new era can be called the digital age [12].

Therefore, there is a need to find multiple means for communicating information and data correctly, which protects this information from unauthorized access. Cryptography is the science that deals with the methods used to protect the storage and transfer of information in a wide range. A secret key is used to encrypt data [18].

Although encryption is a good way to protect information, it is easy to detect and can be manipulated by any intruder [11]. There is a need for more sophisticated, more confidential and data-intensive technologies, especially with the emergence and evolution of the internet which is sufficient to induce the attacker or the attacker to believe that important or sensitive data exists in the random or in the P encrypted, begins using anti-encryption techniques to try to get their content, and even if it is unable to achieve this, he may tamper with or use some of the available means to prevent access to its target [18].

The major challenges faced by information security is the emergence of computer networks and means of communication in order to store enter and provide information internally within remote hosts. A new term has been added to the information security glossary, network security, which is defined as the correct protection for all components connected to a computer network [5].

Steganography - the science of concealment

It is the science of writing hidden messages in such a way that only the sender and recipient concerned can doubt the existence of the message, a kind of secrecy through obscurity. Steganography is mainly for encryption and hidden writing on the image of files. Overall, the advantage of steganography over abstract encryption is that the message itself does not attract attention [1]. Encrypted messages are clearly protected, regardless of their decryption sounds suspicious; the author in countries that prevent encryption may condemn them and while protecting message content encryption, steganography can protect both the message and parties involved in the messaging. Steganography includes hiding information in computer files. In digital

steganography, the electronic connection includes a hidden markup at the carrier center level, such as a document file, an image file, a protocol. Multimedia files are ideal for hidden transmissions due to their sizes. As a simple example, a sender can use a non-catchy image, and adjust the color of a pixel for every hundred points, to match an alphabetical letter. There will be slight, inconceivable and unlikely changes that will be detected by the naked eyes [1].

Encryption

In encryption process, cipher refers to an algorithm for performing encryption or decryption - a series of well-defined steps that can be followed as a procedure. An alternative, less common term for cipher is encryption. To encrypt means to convert information to a crypto or code. In common language, "cipher" is synonymous with "code", which is a set of steps that encrypts the message; however, the concepts are distinct in encryption, especially classical encryption [22].

Symbols generally replace different length strings of characters in the output, while zero is generally replace the same number of characters entered. There are exceptions and some encryption schemes may use slightly more or less characters when output against the entered number. Icons run by substitution according to a large codebook that links a random string of characters or numbers to a word or phrase. In cipher, code such as "TYGHOR" can mean text message "continue to the following coordinates". When using this code, the original information is known as encrypted text, and the encoded form is encoded. An encrypted text message contains all encrypted message information, but it is not in a form that can be read by a person or computer without the appropriate mechanism to decrypt it [5].

Watermark

A watermark is the hidden information within a digital signal (such as image, video, audio, polygonal model). It is integrated into the content of host signal itself, and requires no additional file header or conversion of data format as well. Moreover,

it is designed to permanently reside in the host data. Unlike encryption, watermark does not restrict access to the host data [4].

Digital watermark

This is also referred to as simply watermarking a pattern of bits inserted into a digital image, audio or video file that identifies the file's copyright information (author, rights, etc.). The name comes from the faintly visible watermarks imprinted on stationery that identify the manufacturer of the stationery. The purpose of digital watermarks is to provide copyright protection for intellectual property that is in digital format [10].

Unlike printed watermarks, which are intended to be somewhat visible, digital watermarks are designed to be completely invisible, or in the case of audio clips, inaudible. Moreover, the actual bits representing the watermark must be scattered throughout the file in such a way that they cannot be identified and manipulated easily. The digital watermark must be robust enough so that it can withstand normal changes to the file, such as reductions [2].

Andrew Terchel and Charles Osborne coined the term “digital watermark” in December 1992. The first successful sign of the inclusion and extraction of the spectrum marker deployed in 1993 was by Andrew Terchel, Charles Osborne and Gerard Rankin. Watermarks are the identification marks produced during the papermaking process. The first watermarks appeared in Italy during the 13th century, but their use spread rapidly throughout Europe. It was used as a means of identifying the papermaker or trade union that made the paper. A thread sewn on the paper template often creates labels. Today's watermarks continue to be used as markers of the plant and to prevent forgery [23].

1. THE ANALYSIS OF SUBJECT AREA

1.1. Problem statement

The main problem related to digital watermark is basically how to maintain balance between imperceptibility, robustness and capacity as increasing one factor negative effect on other and a good digital watermarking system possess above feature. To achieve good imperceptibility, watermark should be embedded in high frequency component whereas robustness occurs in low frequency component. Another issue is the human vision ability to view images vividly [9]. In RGB color images, the less sensitive color for hiding watermark is the blue. Other problems involves the fragile watermarking, content recovery against cropping is also a challenging issue. Fragile watermarking slightly distort the results in the destruction of the watermark. Next one is payload size, payload size is how amount of information it carries. As more is payload size, it compromises with the imperceptibility. Next issue is robustness in spatial domain [16]. As in spatial domain, there is change in pixel values. It is hardly resist against various attacks like JPEG compression, high pass filtering, low pass filtering, cropping etc. Other issue is cost of computation, which includes the cost of inserting, and detecting watermark that should be minimized. Next issue is false positive rate, which is important property of digital watermarking system. Other issue is to design universal technique for all the digital media that is robust against various type of attacks.

The size of image logo for every digital watermark system is usually small but this was corrected by adjusting the size to bigger size of logo [16].

1.2. Aim and objective of this project

Create a simple and convenient tool to protect digital images using a digital watermark. The created tool must be accessible from any device and from anywhere in the world. Using the tool should not require the installation of additional software and its configuration. The developed tool should allow:

- to browse and access the images with hidden/ invisible watermark;

– to browse and access the images with visible watermark;

– to generate a watermark;

– to browse logo or watermark from user's computer;

– to be able to make comparison between the original image with the watermark;

– to store uploaded and created images.

1.3. Comparative analysis of analogous sites

Three sites with similar functionality were selected for analysis: www.watermarquee.com, www.watermarkly.com, www.www.watermarktool.com

Water Marquee is a site with same functionality. This site enables user to easily use watermark to one photo or dozen photos at the same time; to totally customize by adjusting the font, size, color, and position of the watermark until it becomes perfect; and it works on any operating system, which permits user to watermark their photos right in browser on internet. The URL of this site: www.watermarquee.com. The screenshot of the homepage icon is shown in figure 1:

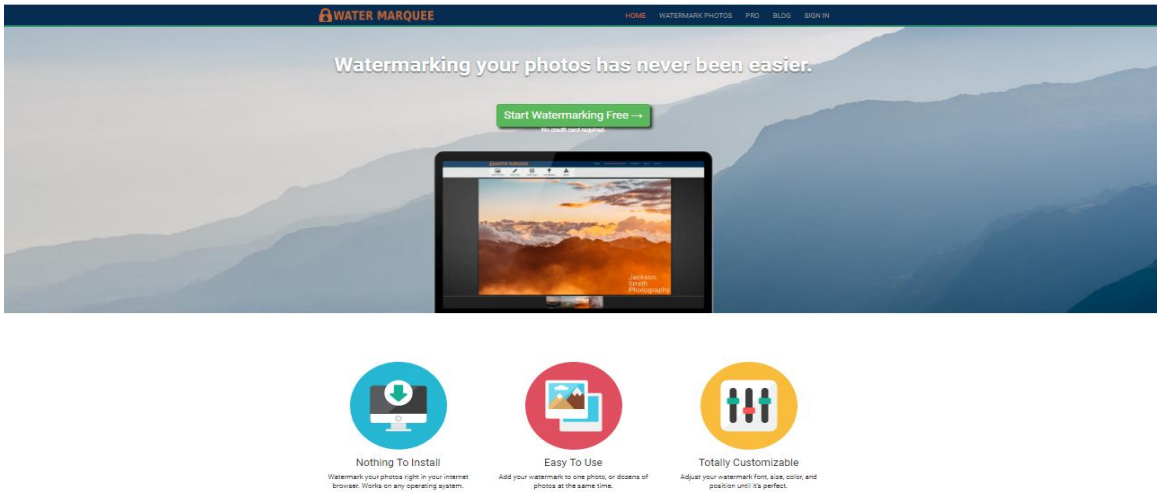


Fig. 1. Home page of the Water Marquee site

In Warsaw Poland, the WaterMarkly is another analogical site with the same functionality. The URL of this site: www.watermarkly.com. The main page of this

site consist of four top navigation menus which are “Watermark Photos”; “Image Resizer”; “Support”; “Blog” and “Run Watermarkly”. When the user clicks on the “Watermark Photo” icon, the screenshot of the page is displayed as it is in figure 2.

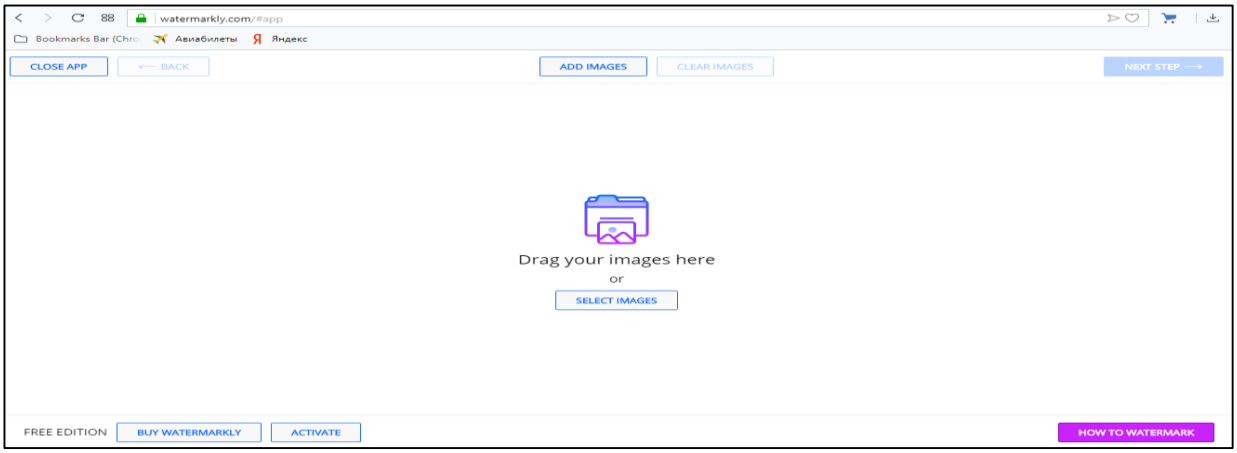


Fig. 2. Watermark photo page on WaterMarkly site

Watermarktool is another related site with the similar functionality as this system. The site consist of six top navigation menu, which are “HOME”; “FEATURES”, “WMT PLUS”; “FAQ (Frequently Ask Question)”; “ADVERTISING”; “CONTACT FORM”. Watermarktool is one of the powerful online watermarking software that allows you to quickly and easily protect images with a visible watermark. With the many watermarking options available, you are able to personalize your images in a variety of ways - including text size, color, and position. The free version requires no sign up and gives you full access to the text watermark feature. For a small monthly fee, adds extra functionality such as a higher file size limit, the ability to save your watermark settings for future use, and the option to use an image as a watermark. The URL of this site: www.watermarktool.com. The home page of this site is shown in figure 3.

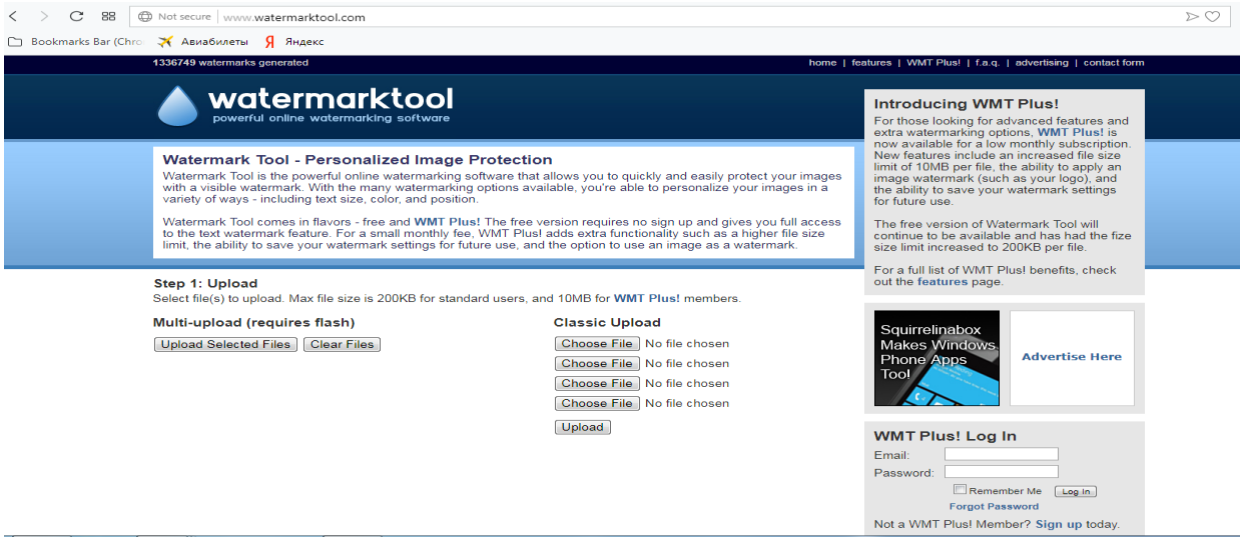


Fig. 3. Watermarktool homepage

The study selected analogs of the main advantages and disadvantages were identified. The results are shown in table 1.

Thus, it was decided that in order to ensure the competitiveness of the developed application it is necessary to implement:

– the ability to store images on the server (at least in a certain amount);

– provide the ability to process images of any resolution for free;

– provide several tools for working with digital watermarks (visible / invisible / logo / text).

The selected features are taken into account at the design stage of the application.

Table. 1. Advantages and disadvantages of the analyzed analogues

№	Website	Advantages	Disadvantages
1.	Water Marquee	1. Simple and convenient interface.	1. There is no possibility to contact the service provider. 2. Full functionality is available only after payment. 3. Restrictions on a free account.

End of the table 1

2.	Water-markly	1. The ability to resize the image. 2. The user can create individual watermarks of his choice. 1. Support service. The site has a blog which provides information about manual watermark position, how to fill the picture with available watermark. 3. There are four other languages (French, Russian, Dutch and Spanish) for users. 4. Processing multiple images at once.	4. There is no possibility to Store images on the server.
3.	Watermark tool	5. Easy navigation. 6. Restrictions on a free account. 7. There is an advertising platform for all users. 8. The site provides a platform for user F.A.Q. 9. The site doesn't have a blog but it has a page where users can fill a contact form for any advertisement.	5. Access to full functionality is expensive. 6. Payment by PayPal only.

2. DESIGN OF WEB-APPLICATION FOR DIGITAL WATERMARK IMAGE PROTECTION

2.1. Functional requirements

The analysis of the subject area identified the following objects and actor:

Original image – the image, in which the digital watermark will be adding. The image can be uploaded by the user or selected from previously uploaded (stored on the server).

Watermark image – the image used as a watermark.

Watermark text – the text used as a watermark.

Protected image – the finished image with a watermark (the result of processing).

User – authorized user of the service. The system will be able to function with only authorized user. If the site visitor does not have a login and password for authorization, he needs to reiterate. The existing user can login with their respective usernames and passwords.

Analysis of the subject area and a review of existing solutions allowed to clearly formulating the functional requirements for the system being developed. In determining functional requirements, the main goal was to create a simple and easy-to-use tool, without unnecessary functionality, but with the ability to store images. The authorized user is can able to choose any type of watermark where can chooses any type of processing such as loading your own logo or creating your own logo from the writers in the program where it appears on the image in the form of copy wright or protect its own logo on the image by adding a copy wright hidden inside the picture. Thus, the developed web service should solve the following main tasks:

- upload image to protect;
- upload logo;
- to embed a digital watermark to protect the digital image;
- to generate a watermark from user entered text;

– realize the ability to place a visible or invisible digital watermark on the image;

– specify the position of the digital watermark on the image;

– to store uploaded and created images;

– to browse and access the images with hidden/visible watermark;

– to check the image for the presence of a digital watermark (extract the text embedded in this application);

– download image with digital watermark.

2.2. Non-functional requirement

The created tool must be accessible from any device and from anywhere in the world. Using the tool should not require the installation of additional software and its configuration. Therefore, it was decided to implement a web-based application. Thus, the non-functional requirements of this system is the same frontend display in browsers such as Google Chrome, Opera Mini, Mozilla Firefox

2.3. Use case diagram and flow chart of the system

The use case diagram for the system is shown in figure 4 and it provides a description of all the functional requirements the user is expected to perform. In the use case diagram, one actor includes the user. The system has eleven (8) use cases as depicted in the figure 4.

The authorized user can perform these functions:

– *Upload images*

The authorized user is able to upload the image and manipulate it through the site also can able to upload more than one image in each processing process is uploaded one image. The authorized user can control the size of the logo where it is

capable of size option (small, big or middle). The user is also able to control the location of the logo (right, left, top, bottom, or middle).

– *Upload or choose logo*

The user can be able to upload or choose watermark (logo) from computer, and can generate copyright.

– *Enter text*

An authorized user can enter text to create a copyright.

– *Choose the method*

The user selects the method so that can be able selection of the watermark is visible or invisible. The selection of the method is created by clicking on the button to raise the logo to put it on the image or by typing the watermark in the form of copy wright or clicking the (invisible watermark) button or clicking (visible watermark) button.

– *Add text and logo*

The authorized user is able to start processing to choose the appropriate method or location of the logo or the appropriate size of the logo in the image.

– *Image list*

After processing is complete, an authorized user can save all kinds of image on the server.

– *Download image*

An authorized user has access to saved images from anywhere in the world and can download them to a local computer.

– *Check image*

The authorized user is able to show the hidden copyright within the image by uploading the image to the site and extracting the hidden copyright inside the image for its own protection.

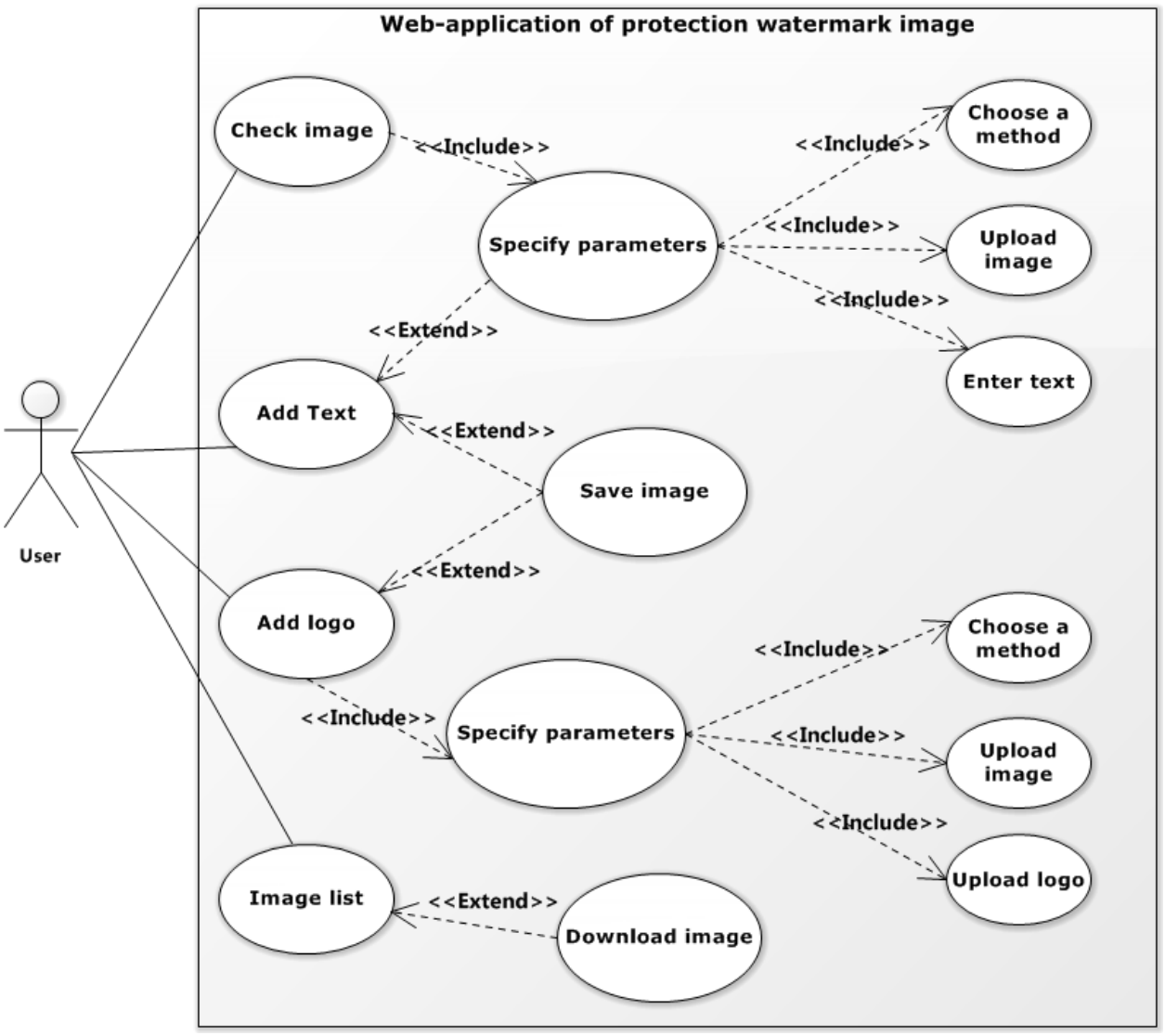


Fig. 4. Use case diagram

2.4. Flow chart of the System

A flowchart is a visual representation of the sequence of steps and decisions needed to perform a process. The figure 5 shows that, there are 19 processes depicted with the oval shape and a decision with the diamond shape. The "start" shows the initial process while the "end" shows the process of termination after the flow through other intermediate processes.

The "Login" is a decision that determines if the user can perform and proceed to the next process or not when the condition is satisfied "Yes" and when not satisfied

“No”. If the condition is not satisfied “No”, the user returns to the “Registration” process and to the initial “Start” process otherwise “Yes”, the user proceeds to the “Upload image” process, next to the “set parameters” process which is sub divided into the process “Positioning” that permit the user to “specify size”, “choose logo” and “print text”. The next process involves the “visible or not”, “execute” and “save image” until it is terminated at the “end” process. After the “Login” process the user can also perform the process named “Go to the form check copyright”, next “upload image”, “Check” and “end” process. Alter-natively, the user after the “Login” can decide to proceed to the “Select list of images” process which is sub-divided into the “download” and “delete” process, then the users finally terminates their process at the “end”.

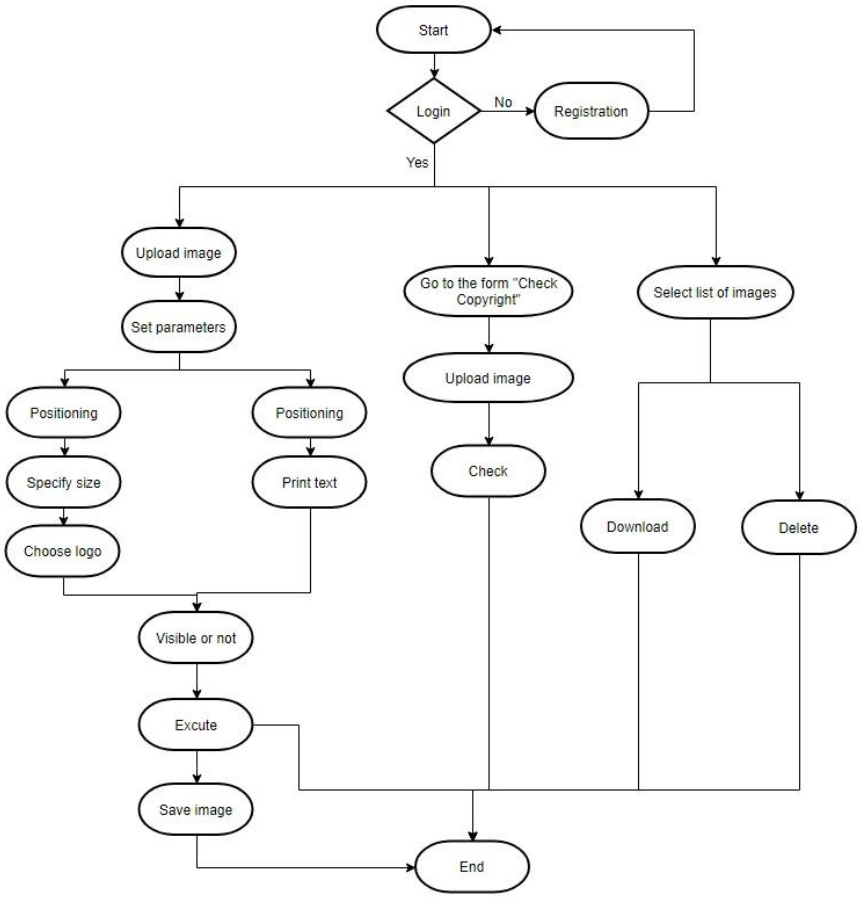


Fig. 5. Flow chart diagram

2.5. Database schema

A database is used to store information about the users of the system and the images uploaded by them. The database schema for the system is developed and represented in fig.6.

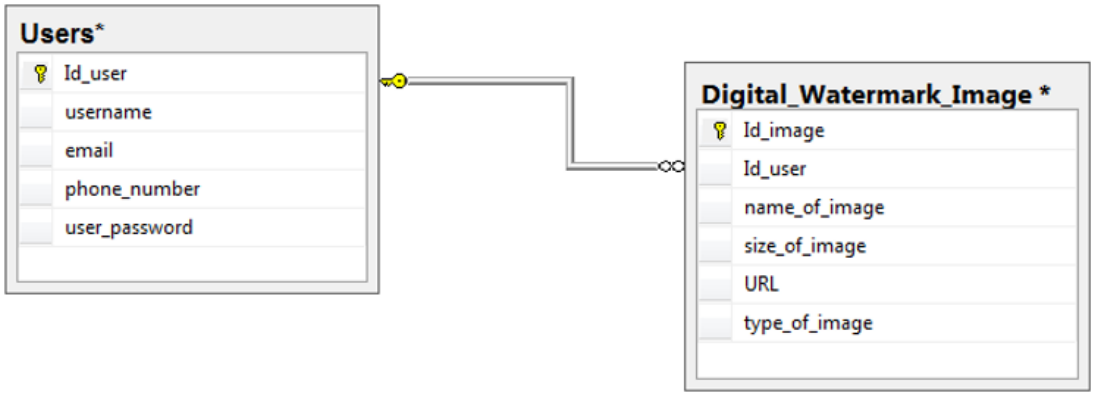


Fig. 6. Database schema

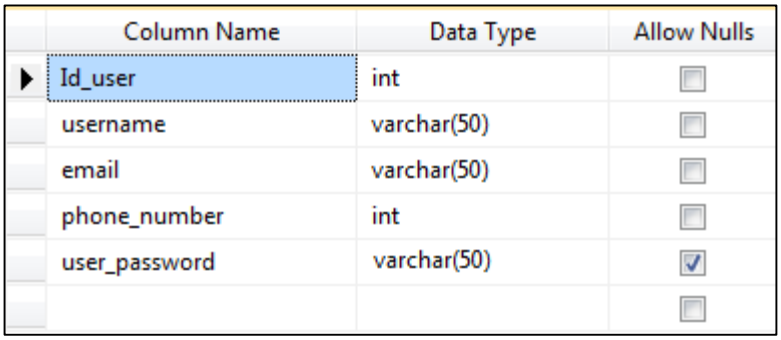
The database schema consists of 2 tables (fig.6). These tables are “Users” and “Digital_Watermark_image”. The column “Id_user in the Users table is the primary key and it has a foreign key also in the Digital_Watermark_image table. The Users table have 5 fields which are (Id_user*, username, email, phone number and password) while the Digital_Watermark_image table have 6 fields which are (Id_image*, Id_user, name_of_image, size_of_image, URL and type_of_image)

The internal structures of the two tables are shown in figure 7-8 with the name of fields, data types (varchar, int, nvchar). These internal structures was created in Microsoft SQL server management studio 2014.

Column Name	Data Type	Allow Nulls
Id_image	int	☐
Id_user	int	☐
name_of_image	varchar(50)	☐
size_of_image	varchar(50)	☑
URL	nvarchar(MAX)	☐
type_of_image	int	☐
		☐

Fig. 7. Internal structure of the table “Digital_Watermark_image”

The internal structure of the Digital_Watermark_Image table in figure 7 consists of: the Id_image, Id_user and type_of_image that is an integer data type and it is set to be not null. This means that the values in the field can contain only integers (numbers) and not be empty; the name_of_image and the size_of image are set as the varchar (50), this means that it can only be a variable or characters with a length restriction of 50. The URL has a data type of nvarchar (MAX).



Column Name	Data Type	Allow Nulls
Id_user	int	☐
username	varchar(50)	☐
email	varchar(50)	☐
phone_number	int	☐
user_password	varchar(50)	☑
		☐

Fig. 8. Internal structure of the table "Users"

The internal structure of the Users table in figure 8 consists of the id_user and phone_number that is an integer data type and it is set to be not null. This means that the values in the field can contain only integers (numbers) and not be empty; user_password is set null because the users can decide to make password null; the username and email ID are set as varchar (50) data type, which means that it is mainly characters with length restriction of 50.

The tables with all the data for the user to access the system is shown in fig. 9 - 10.

	Id_user	username	email	phone_number	user_password
1	1	Hussein	hussein_199227@hotmail.com	79049794122	1234
2	2	Alina	alina@gmail.com	79053452123	12234

Fig. 9. Users Table with data

	Id_image	Id_user	name_of_image	size_of_image	URL	type_of_image
1	4	1	IMG22678	2mb	/img/visible	visible
2	5	1	IMG22988	3mb	/img/invisible	invisible
3	7	2	IMG23787	4mb	/img/invisible	invisible

Fig. 10. Digital_Watermark_Image Table with data

2.6. Algorithms used to embed a watermark

The Hash function algorithm for the encryption of text

The Hash function is a Hash function, meaning that it takes any number of data segments and returns a constant string of bits called the Hash value, so that any change in the original data (accidental or intentional) will result in a significant change in the Hash value very). Typically, encrypted data is called "message" and the amount of encryption is called digest [24].

This type of algorithm does not require a cryptographic key because it is not used to encrypt the text, but to make sure that the content of the message is reliable and has not been modified. By comparing, the abstract sent with the digest generated by the message to verify the validity of its content [7]:

- a feed can be easily calculated for any given message;
- it is not possible to generate a message from a given abstract;
- it is not possible to change a message without changing its conclusion;
- it is not possible to generate two messages with the same conclusion.

Common description of the LSB algorithm

In computing, the least significant bit (LSB) is the bit position in a binary integer giving the units value, that is, determining whether the number is even or odd [6]. The LSB is sometimes referred to as the low-order bit or right-most bit, due to the convention in positional notation of writing less significant digits further to the right. It is analogous to the least significant digit of a decimal integer, which is the digit in the ones (right-most) position [3].

It is common to assign each bit a position number, ranging from zero to $N - 1$, where N is the number of bits in the binary representation used [15]. Normally, this is simply the exponent for the corresponding bit weight in base-2. Although a few CPU manufacturers assign bit numbers the opposite way (which is not the same as different bendiness), the term least significant bit itself remains unambiguous as an alias for the unit bit [14].

By extension, the least significant bits (plural) are the bits of the number closest to, and including, the LSB.

The method is used by arranging the text in one row, isolating every three bits separately, and storing in the least important bit in red, green and blue (RGB) [8].

2.7. Development of the interface

Web interface development is the pattern of developing the interaction between users and software running on the web server. Most web interfaces are designed with a focus on simplicity, so that no extraneous information and functionality might not be difficult for the user to understand. The interfaces of the system developed is shown in this session. The interface consists of the homepage, the browse visible watermark, the browse invisible watermark and the check watermark interface. The home page interface for creating watermark is shown in figure 11.

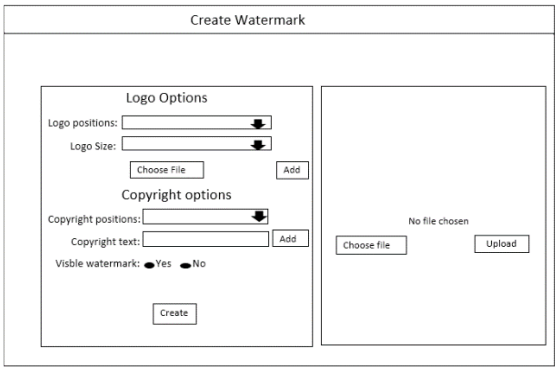


Fig. 11. Home page interface

The Invisible watermark image interface displays the features of the invisible images. The interface for creating invisible watermark images are shown in figure 12.

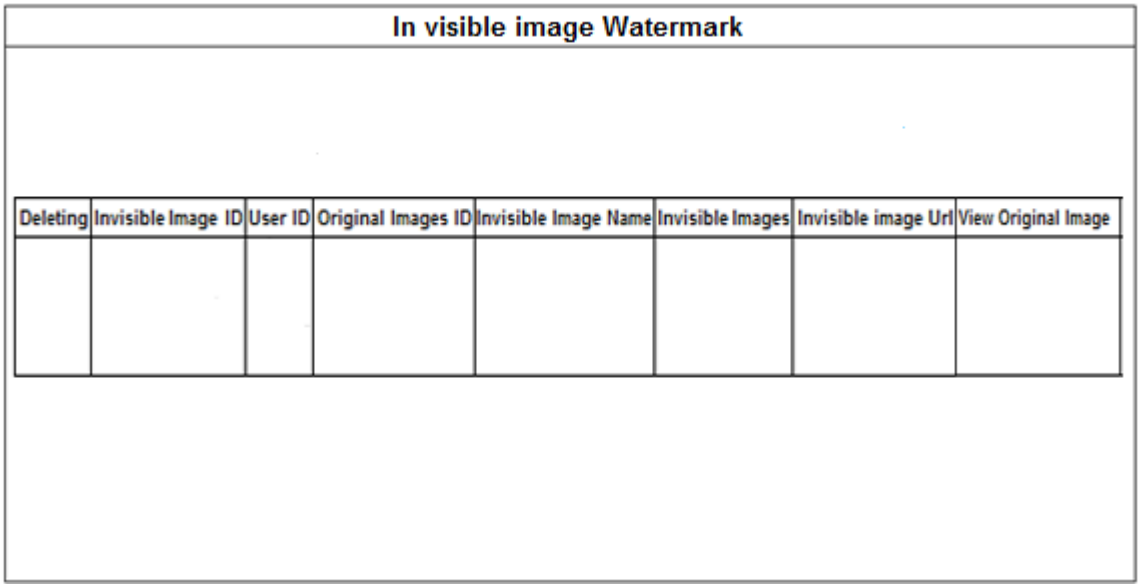


Fig. 12. Invisible Watermark images Interface

The visible watermark image interface displays the features of the visible images. The interface for creating visible watermark images are shown in figure 13.

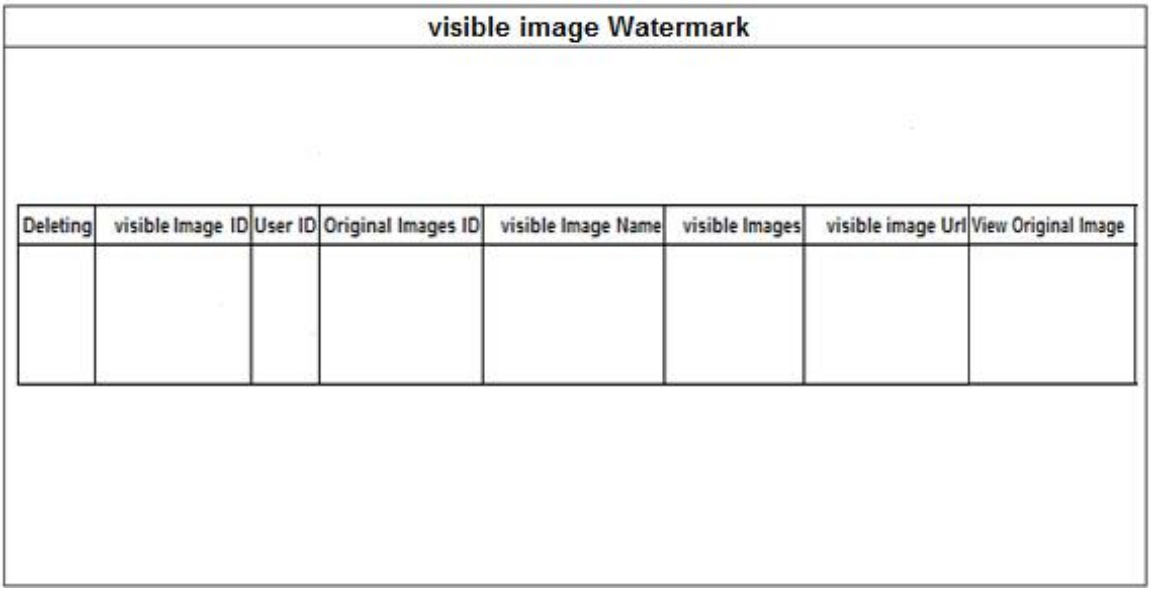


Fig. 13. Visible watermark images interface

The check copyright interface is used to embedding text inside image to protect image or logo in image. The interface for the check copyright is shown in figure 14.

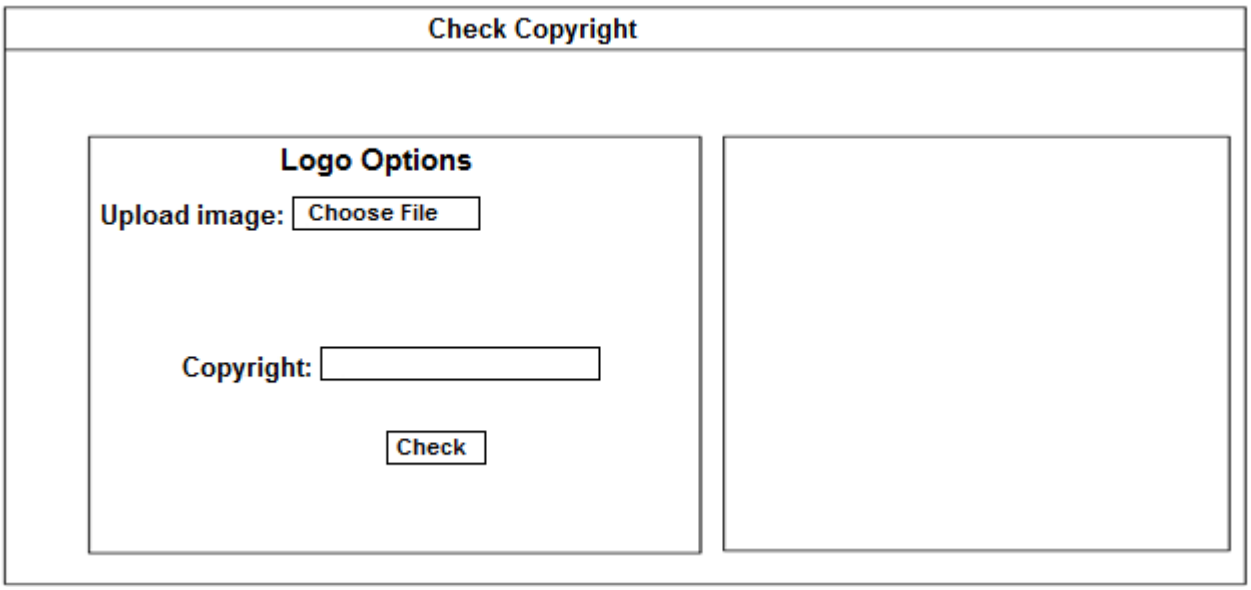


Fig.14. Check copyright Interface

3. IMPLEMENTATION

3.1. Choice of development tools

For this work, it was decided to work with ASP.NET (Active Server Pages) because of its flexibility and ease of implementation.

ASP.NET is also an open source web application which produces web pages dynamically. ASP.NET web pages consist of web forms which are the main building platform for the development of most online designs. Web forms consists in files with an extension "aspx", these files also contain (X) HTML markup platform. The database management system utilized for this work is SQL server management studio 2014 which supports database for ASP.NET web forms. We used the programming language C# for the implementation of this system. The project consist of a class named “Steganography.cs”; and main master page named “master.Master” and seven web forms. We chose ASP.NET web form because it is not complicated to understand and it is more flexible for me to easily implement.

3.2. Algorithms for embedding digital watermarks

Hide (copyright)

The masking algorithm in color images can be summarized as follows:

- View the text file to hide by itself
- Choose the appropriate image size for the hiding parts.
- Convert the location of the image element to which the first letter of text is hidden.
- Contains (2) the first two parts, while the second and third parts of the word contains (3) bits in sequence.
- Read the color value of the image element by extracting the values of the three bytes representing the basic colors (red, green and blue) (R, G, B).

– Replacing the byte bits of each of the three colors of bits of one of the parts in the location of the least necessary bits.

Restore (copyright)

– Read the color value of the image element and extract the values of the three bytes that show red, green and blue.

– The replaced bits are then taken into the masking process of each color that is collected to form the character byte.

– Converts the value of the character from the ASCII value to the shape it represents.

– The next item that will be hidden will then be identified by calculating the hiding distance and adding it to the location of the current item.

LSB for embedding watermark (logo) on image

– For the logo, we have three matrix for each color 256 color resolution of 2 bits. The bits to the left are the most important because they have the higher power.

– For the image, we have the same.

– If we want to embed, we usually take the most important bit of the logo and exchange them with the least important bits of the Image.

– If we want, make logo more obvious we add another bit to the exchange bits and vice versa.

– For the logo, we take the most important bits and put them in the least important bits in the image.

3.3. Database

The CREATE table statement is used to create the Users and Digital_Water-mark_image tables in the database. The CREATE SQL command is one of the Data Definition Languages (DDL), which deals database schemas and descriptions, of

how the data should reside in the database. The SQL queries for creating the Users table is shown in figure 15.

```
Create table Users
(
  id_user int primary key not null,
  username varchar(50) not null,
  email varchar(50) not null,
  phone_number int not null,
  user_password varchar(50),
);
```

Fig. 15. SQL query for Users table

The SQL queries for creating the Digital_Watermark_image table is shown in figure 16.

```
Create table Digital_Watermark_image
(
  id_image int primary key not null,
  id_user int not null,
  name_of_image varchar(50) not null,
  size_of_image varchar(50),
  URL nvarchar(MAX) not null,
  type_of_image int not null
);
```

Fig. 16. SQL query for creating Digital_Watermark_image table

The SQL queries for selecting and inserting values in the Users table is shown in figure 17.

```
select from Users
insert into Users (id_user,username,email,phone_number, user_pass-
word) values
(1, Hussein', 'hussein_199227@hotmail',79049794, 1234' );
insert into Users (id_user, username, email, phone_number,
user_password) values
(2, 'Alina', 'alina@gmail.com', 7905345, '12234'');
```

Fig. 17. SQL query for select and insert values in User table

The SQL queries for selecting and inserting values in the Digital_Watermark_image table is shown in figure 18.

```
use Digital
select*from Digital_Watermark_image
insert into Digital_Watermark_image (Id_image, Id_user_name_of_im-
age, size_of_image_URL type_of_image) values
(4,1, "In622678", "/img/visible", "visible");
insert into Digital_Watermark_image (Id_image, Id_user,name_of_im-
age,size_of_image,URL,type_of_image) values
(5,1, "ING22988","3mb","/ing/invisible","invisible");
insert into Digital_Watermark_image (Id_image,Id_user_name_of_im-
age,size_of_image,URL,type_of_image) values
(7, 2, "IMG236787", "4mb", "/img/invisible", invisible");
```

Fig. 18. SQL query for select and insert values in Digital_Watermark_image table

The SQL queries for deleting values in the Digital_Watermark_image table is shown in figure 19.

```
use Digital
delete from Digital_Watermark_Image where Id_image= 5;
delete from Digital_Watermark_Image where Id_image= 7;
delete from Digital_Watermark_Image where size_of_image='3mb';
```

Fig. 19. SQL query delete values in Digital_Watermark_image table.

The SQL queries for updating values in the User table is shown in fig 20. The column “email” was updated from “alina@gmail.com” to “alinakey@yahoo.com”

```
select*from Users
UPDATE Users set email = 'alinakey@yahoo. com' where
email='alina@gmail. com';
```

Fig. 20. SQL query update values in Users table

3.4. Basic functionality

In this system, some fragments of ASP.NET C# codes text for different interfaces are shown. The registration page provides for the user to input the “Name”; “email”; “phone number” and “password”. The login page provides for the user to input their “Email or phone number” and “password”. Figure 21 shows the code for the registration and login page in email because for the number phone the same.

```
qlConnection con = new SqlConnection(WebConfigurationManager.
ConnectionStrings["digital_WatermarkConnectionString"].
ConnectionString);
if(Regex.IsMatch(TxtEmail.Text, "@"))
{
SqlCommand cmdRead = new SqlCommand
("select * from [user] where email=
@email and password=@password", con);
string encryptedPassword =
FormsAuthentication.HashPasswordForStoringInConfigFile
(TxtPass.Text, "SHA1");
cmdRead.Parameters.AddWithValue("@email", TxtEmail.Text.ToLower());
cmdRead.Parameters.AddWithValue("@password", encryptedPassword);
con.Open();
SqlDataReader drRead = cmdRead.ExecuteReader();
if(drRead.Read())
{
FormsAuthentication.RedirectFromLoginPage(TxtEmail.Text.ToLower(),
chkRemember.Checked);
master.UsrName = "Hello, " + drRead["name"].ToString();
Session["uID"] = drRead["uID"].ToString();
}
else
{
LbMsg.ForeColor = System.Drawing.Color.Red;
LbMsg.Text = "Email or password incorrect";
}
con.Close();
```

Fig. 21. Code Sample to Registration and Login page

The code shown in figure 22 provides details on how to check the logo size, which is 2mb. The code is based on the if statement, that have the logo coordinate of x and y as 0 respectively. When the image is less than 2mb, the logo will not be accepted.

```
if(DrpLogoSize.SelectedValue == 2.ToString())
{
logoX = 0;
logoY = 0;
}
if (chkImgWm == false)
{
imgWithLogo = new Bitmap(imgOriginal, 400, 400);
}
else
{
imgWithLogo = new Bitmap(imgWithWm);
}
WatermarkImage(imgWithLogo, imgLogo, new Point(logoX, logoY), 00.50f);
var varImage = new Bitmap(imgWithLogo, 400, 400);
imgFinal = new Bitmap(varImage);
imgWithLogo.Save(msImgLogo, ImageFormat.Png);
string base64 = Convert.ToBase64String(msImgLogo.ToArray());
imgMain.ImageUrl = "" + base64;
chkImgLogo = true;
lblresult.Visible = false;
}
else
{
lblresult.ForeColor = Color.Red;
lblresult.Text = "Please the size of Logo must be less than 2 MB";
lblresult.Visible = true;
}
}}
```

Fig. 22. Code Sample to Check Logo Size

The code shown in figure 23 is for adding logo to the original image. The code has different if statements. When the logo size X and Y is 100, the logo produce is small; when the logo size X and Y is 200, the logo size produce is medium; when the logo size X and Y is 400, the logo size produce is big.

```
if (fs != null & upLogo.HasFile)
{
if (upLogo.PostedFile.ContentLength <= 1950000)
{
MemoryStream msImgLogo = new MemoryStream();
Stream fsLogo = upLogo.PostedFile.InputStream;
Bitmap imgLogoBig = new Bitmap(fsLogo);
Bitmap imgLogo = new Bitmap(imgLogoBig,
imgLogoBig.Width / 5,
imgLogoBig.Height / 5);
if(DrpLogoSize.SelectedValue == 0.ToString())
{
logoSizeX = 100;
logoSizeY = 100;
}
if (DrpLogoSize.SelectedValue == 1.ToString())
{
logoSizeX = 200;
logoSizeY = 200;
}
if(DrpLogoSize.SelectedValue == 2.ToString())
{
logoSizeX = 400;
logoSizeY = 400;
}
imgLogo = new Bitmap(imgLogoBig, logoSizeX, logoSizeY);
}
```

Fig. 23. Code Sample to adding logo to original image

The code in figure 24 is used to add copyright, which can be visible or invisible embedding inside image to original image.

```
if (fs != null & txtCpyRit.Text != "")
{
MemoryStream msWatermark = new MemoryStream();
if (chkImgLogo == false)
{
imgWithWm = new Bitmap(imgOriginal, 400, 400);
}
else
{
imgWithWm = new Bitmap( imgWithLogo);
}
Graphics watermarkGraphic = Graphics.FromImage(imgWithWm);
Brush brush = new SolidBrush(Color.FromArgb(80, 255, 0, 0));
Point watermarkPosition = new Point(wmX, wmY);
watermarkGraphic.DrawString(txtCpyRit.Text,
new Font("Century Gothic", 30, FontStyle.Bold, GraphicsUnit.Pixel),
brush, watermarkPosition);
imgWithWm.Save(msWatermark, ImageFormat.Png);
var varImage = new Bitmap(imgWithWm, 400, 400);
imgFinal = new Bitmap(varImage);
string base64 = Convert.ToBase64String(msWatermark.ToArray());
imgMain.ImageUrl = "" + base64;
chkImgWm = true;
}
```

Fig. 24. Code Sample to add copyright to original image

The code in figure 25 is used to check which pixel element has the turn to hide a bit in its LSB. The code is for extracting copyright on the image in form text. The rightmost bit in the character will be extracted. To also remove the added rightmost bit of the character.

```
switch (pixelElementIndex % 3)
{
case 0:
{
if (state == State.Hiding)
{
R += charValue % 2;
charValue /= 2;
}
}
break;
case 1:
{
if (state == State.Hiding)
{
G += charValue % 2;
charValue /= 2;
}
}
break;
case 2:
{
if (state == State.Hiding)
{
B += charValue % 2;
charValue /= 2;
}
```

Fig. 25. Code Sample for checking pixel element to hide LSB

The code in figure 26 is to hid characters in the image. It also holds the index of the character that is being hidden, thereby holding the value of the character converted to integer. The code holds the pixel that is currently being processed and clears the least significant bit (LSB) from each pixel element.

```
public static Bitmap embedText(string text, Bitmap bmp)
{
State state = State.Hiding;
int charIndex = 0;
int charValue = 0;
long pixelElementIndex = 0;
int zeros = 0;
int R = 0, G = 0, B = 0;
for (int i = 0; i < bmp.Height; i++)
{
for (int j = 0; j < bmp.Width; j++)
{
Color pixel = bmp.GetPixel(j, i);
R = pixel.R - pixel.R % 2;
G = pixel.G - pixel.G % 2;
B = pixel.B - pixel.B % 2;
for (int n = 0; n < 3; n++)
{
if (pixelElementIndex % 8 == 0)
{
if (state == State.Filling_With_Zeros && zeros == 8)
{
if ((pixelElementIndex - 1) % 3 < 2)
{
bmp.SetPixel(j, i, Color.FromArgb(R, G, B));
}
}
```

Fig. 26. Code Sample for hiding copyright inside the image

3.5. User interface

User interface consists of the registration page and login page. The registration page provides the platform for non-existing users to register to the system by entering their name, email, phone and password before they can login in as an existing user as shown in figure 27:

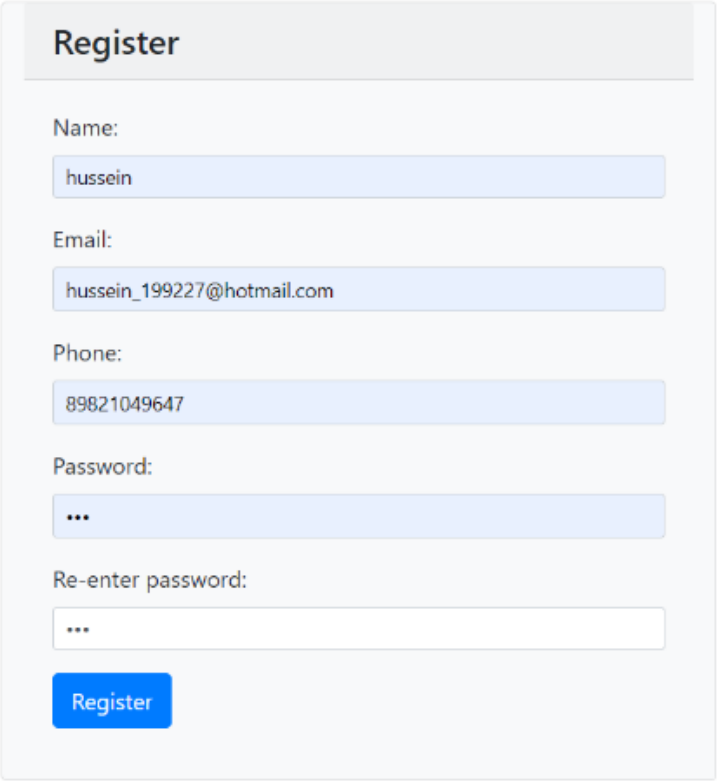


Fig. 27. User interface with the registration rage

The login page is a user interface that provides access for only the existing users who have successfully registered and their necessary details are stored in the system database. The login interface consist of the email or phone number and the password. If the user clicks on the “remember me” box, they can easily login next time without providing their login details. The login interface is shown in figure 28:

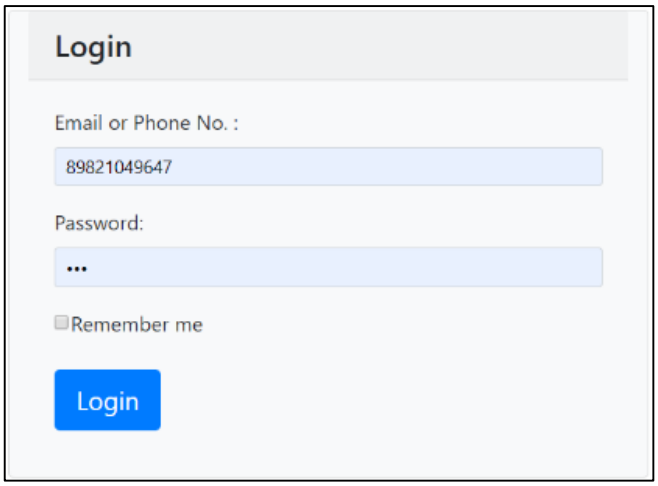


Fig. 28. User Interface with the Login Page

The user interface for creating watermark with two menus (logo option and copyright option) is shown in figure 29:

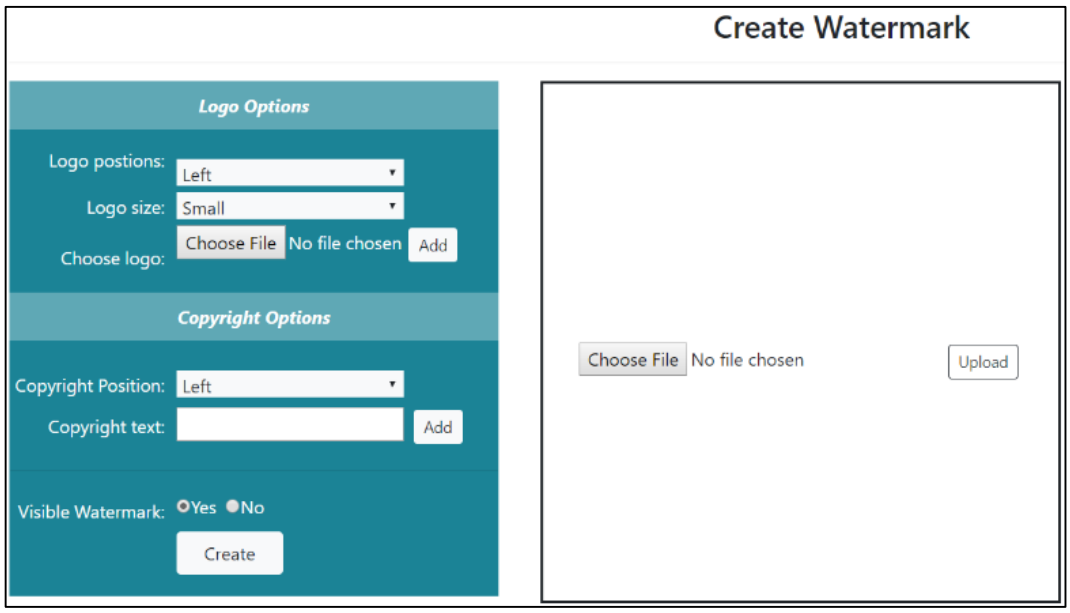


Fig. 29. User Interface with the Create Watermark

User also can check invisible watermark (copyright) that is most important to protection image, extracting copyright (text) inside image, user can upload image when click check we will show which text (copyright) witting inside image is shown in figure 30.

Fig. 30. User Interface with the Check Watermark

4. TESTING

Software testing involves the testing the websites or web-applications for bugs. There are several methods of testing such as functionality testing; usability testing; integration testing; crowd testing; database testing; compatibility testing; and performance testing [13]. As part of this work, functional testing of the implemented web application was carried out.

4.1. Functional testing

Functional testing is a type of software testing which involves the testing of all components to clarify they are working properly as expected or not. This testing also known as component testing of a system [20]. The functional testing was conducted based on the functional requirements of this system [19]. The results of functional testing are shown in table 2.

Table. 2. The results of the functional testing for the system

No.	Function	Expected result	Obtained result	Conclusion
1.	To show the Home page.	Any user can access and view every menu provided on this page	Any user can access and view every menu provided on this page	The function works properly
2.	To give the users the access to login their details.	Only registered user can access the login page after they have registered. The page requires them to enter their “email or phone number” and “password”.	Only registered user can access the login page after they have registered. The page requires them to enter their “email or phone number” and “password”.	The function works properly

Continuation of the table 2

3.	To give all the user access to the registration page.	Any new User can register their details, which includes “Name”, “Email”, “Phone” and “Password”.	Any new User can register their details, which includes “Name”, “Email”, “Phone” and “Password”.	The function works properly
4.	To give users the access to create watermark.	The registered users can create watermark on image; by selecting the logo options and the copyright options, which includes making the water either visible or invisible.	The registered users can create watermark on image; by selecting the logo options and the copyright options, which includes making the water either visible or invisible.	The function works properly
5.	To provide user to browse visible watermark image	The user can view visible watermark images and view the visible image they originally created. Users can also download original image with watermark, by clicking the visible image URL.	The user can view visible watermark images and view the visible image they originally created. Users can also download original image with watermark, by clicking the visible image URL.	The function works properly

End of the table 2

6.	To provide user to browse invisible watermark image	The user can view invisible watermark images and view the invisible image they originally created. Users can also download original image with watermark, by clicking the invisible image URL.	The user can view invisible watermark images and view the invisible image they originally created. Users can also download original image with watermark, by clicking the invisible image URL.	The function works properly
7.	To delete visible and invisible watermark images	The user can delete the watermark they created on the image, either visible or invisible.	The user can delete the watermark they created on the image, either visible or invisible.	The function works properly
8.	To check copyright	The user can check the copyright on images by extracting the text on the image.	The user can check the copyright on images by extracting the text on the image.	The function works properly

4.2. The main forms and interface of the implemented application

The user is able to access the site to create a watermark and phone number or email does this through registration on the site and access after the user login the

website by entering his special email and password can will see the homepage after that the user “upload” the image the user can add “Logo” to the image by browsing about some Logo and click the button “Add”, we can set the position of logo by clicking the drop list “Logo positions” and we can chose “Left”, “Right”, “Top”, “Bottom” and “Middle” for position of logo also we resize logo to “Small” ,“Middle” and “Large” size as shown in figure 31.

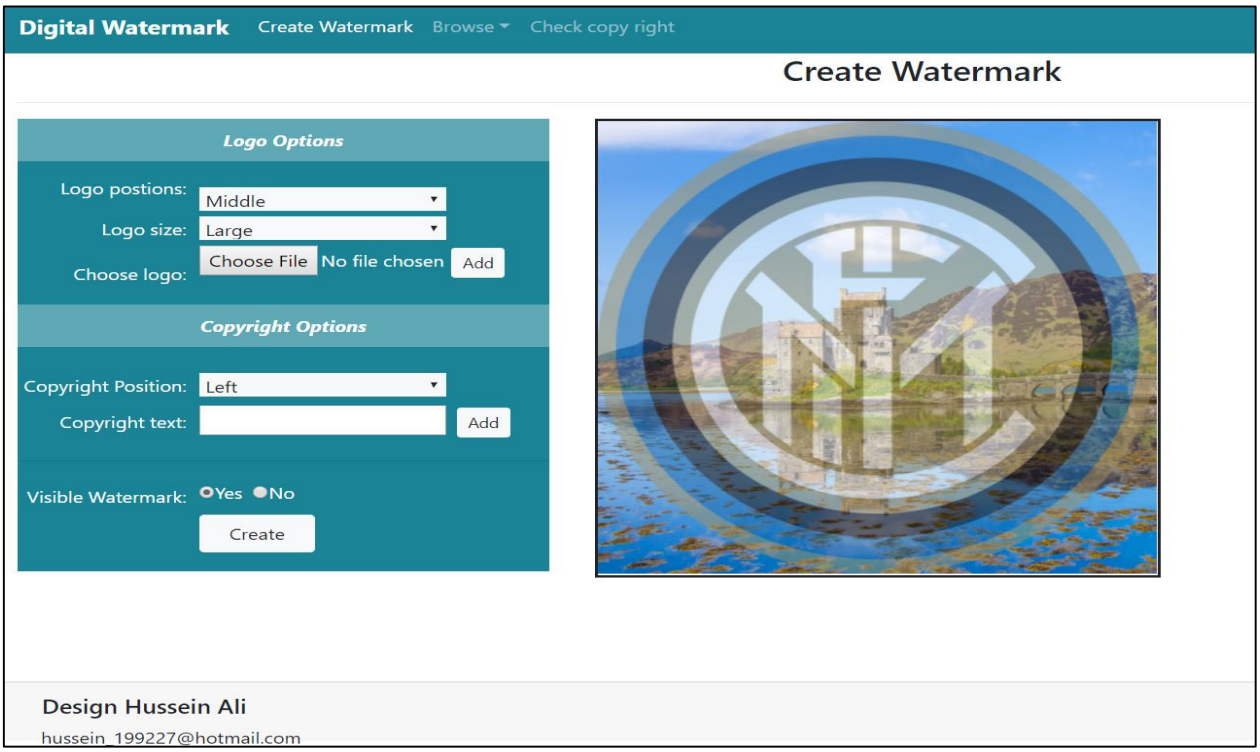


Fig. 31. The add logo to the image

After the user upload the image the user can add copyright to the image by writing some text and click the button “Add”, we can set the position of the copyright text by clicking the drop list “copyright positions” and we can chose “Left”, “Right”, “Top”, “Bottom” and “Middle” for position of copyright also we can make invisible watermark (copyright) inside image for protection image, if we click “Yes” visible watermark (copyright) or click “No” will be invisible (copyright) embedding inside image as shown in the figure 32.

Figure 33–34 show how the user can view all images with visible watermark and all images with invisible watermark by clicking on bottom “browse” and choose visible image or Invisible image also we can download from “URL” or “View original image” we can “deleting” image also.

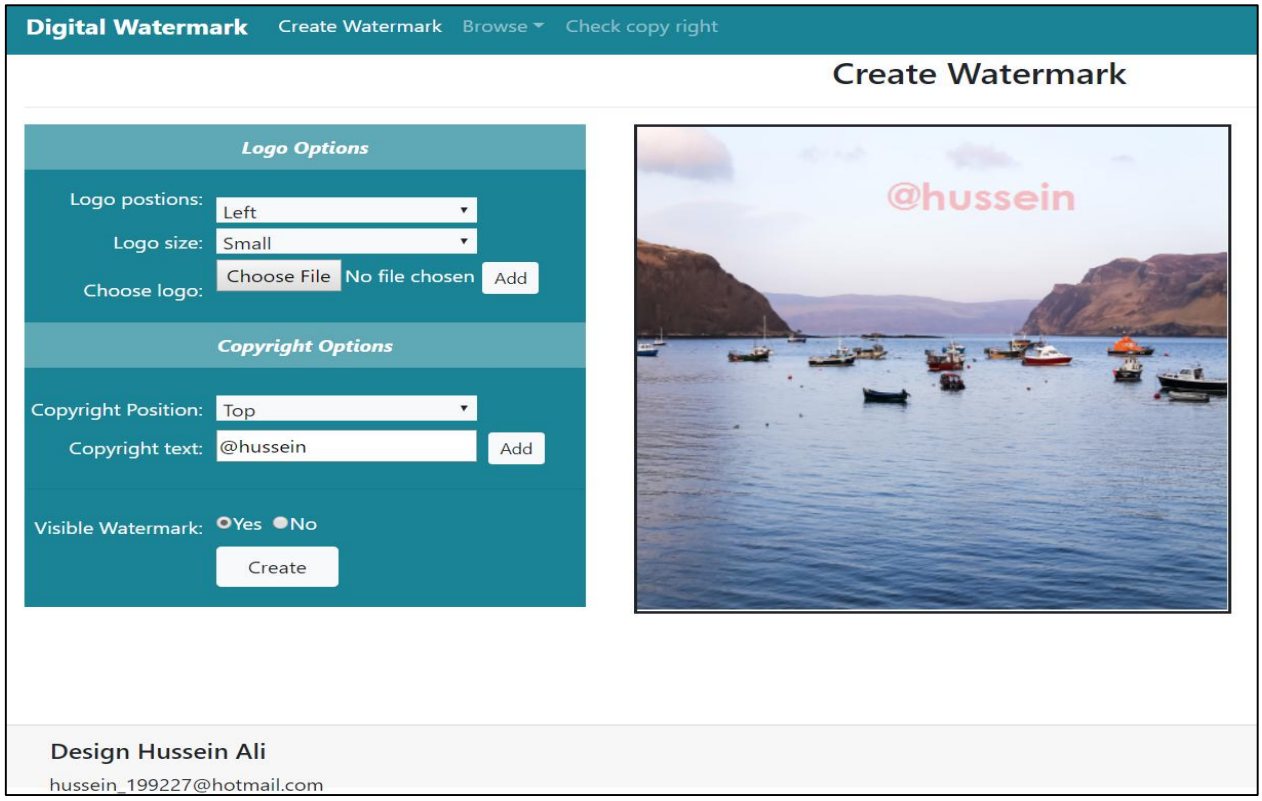


Fig. 32. The add copyright to the image

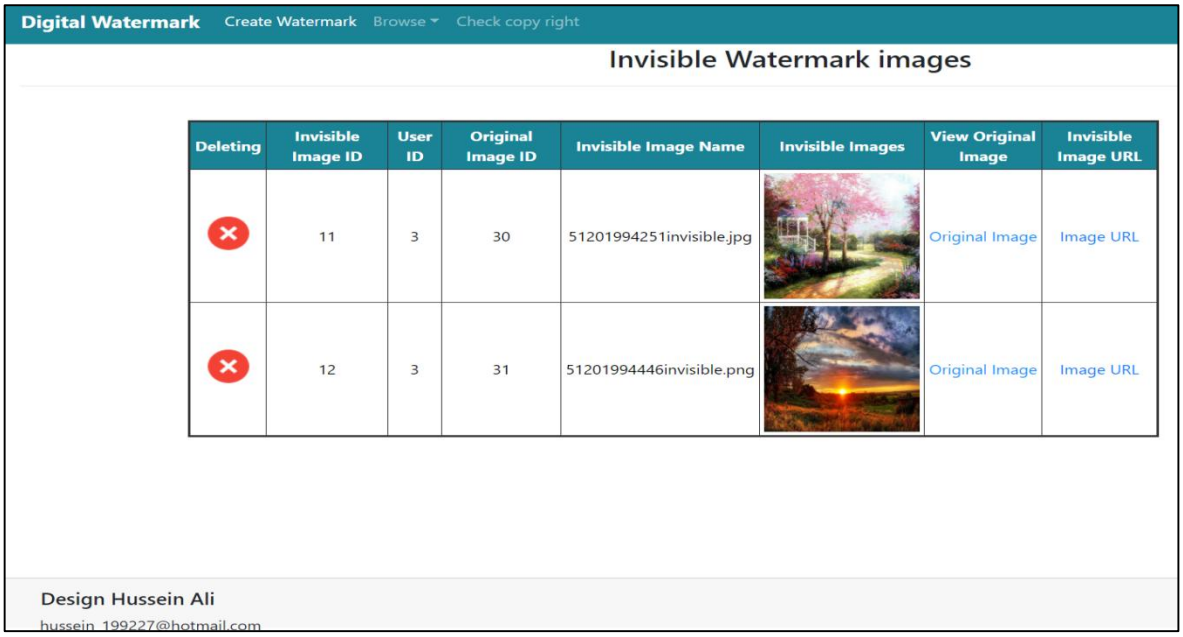


Fig. 33. Invisible image page

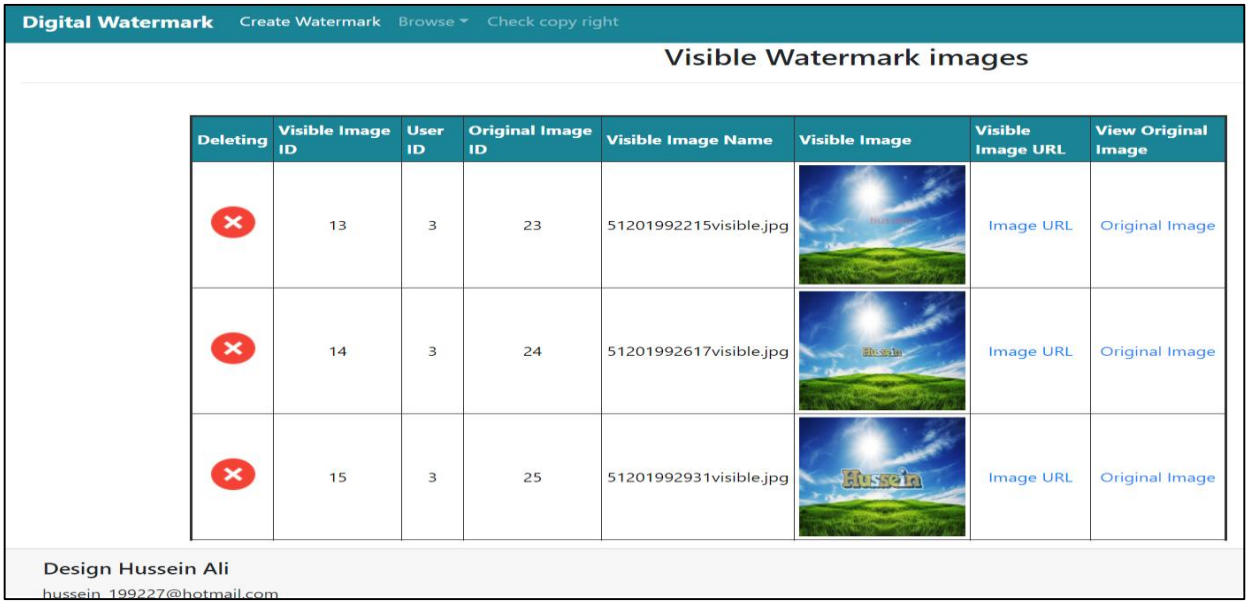


Fig. 34. Visible image page

Figure 35 show how the user can check all images with watermark by clicking on bottom “browse” and add image after that click the bottom “Check” at last we will get the copyright text (extract text from image) also we use invisible watermark for protection logo.

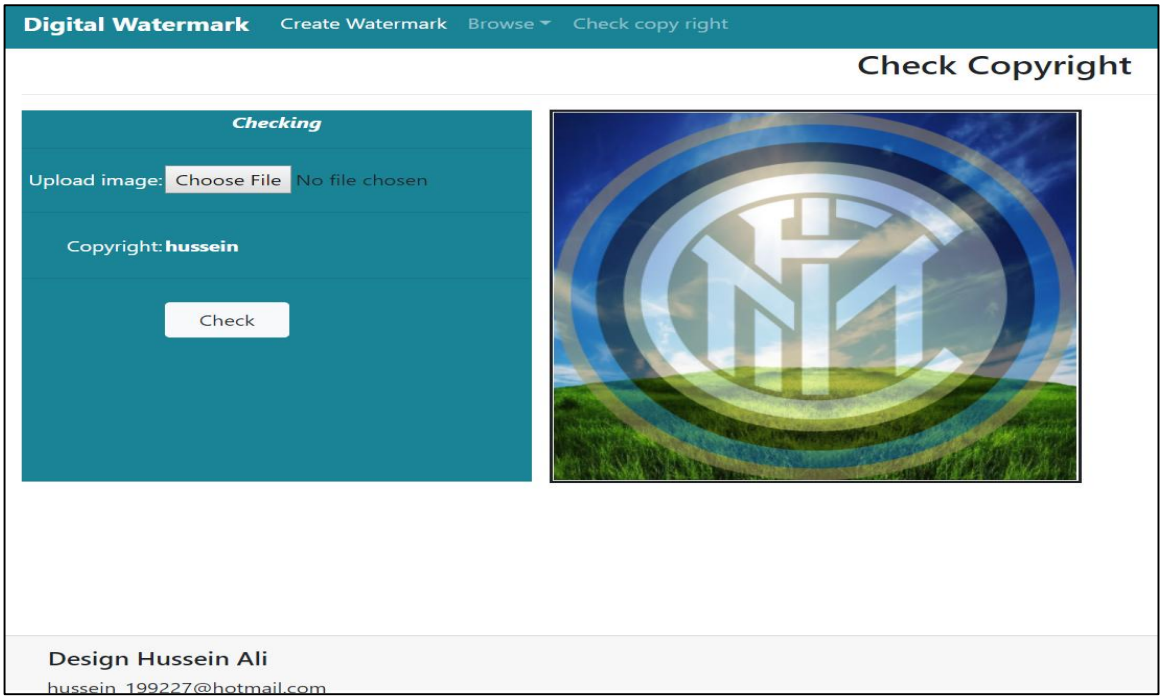


Fig. 35. Check copyright image

CONCLUSION

Undoubtedly, the task of protecting the copyright of digital images is still an urgent task. The main purpose of this work was to create a convenient, potentially popular web application for embedding digital watermarks in digital images, to ensure the protection of copyright on the Internet. The web application developed should be able to solve the problems associated with the inability of a digital watermark to withstand various attacks, such as JPEG compression, high-pass filtering, low-pass filtering, and cropping. The size of image logo for every digital watermark is usually smaller in nature but this can be corrected by a bigger logo.

To achieve the goals it was necessary to solve several problems, including:

– to study the existing analogues and identify the main disadvantages and advantages of ready-made solutions;

– to study the methods and technologies for protecting digital images;

– design a potentially relevant and easy-to-use web application to protect digital images;

– design and implement a database for storing information about users of the application;

– analyze and select the most appropriate means of implementation;

– implement front-end and back-end applications;

– perform functional testing of the implemented application.

The tasks were solved completely. The developed application is available at [http://husseinaliweb-001-site1.htempurl.com]

The future prospective for the developed application that was not implement in this system includes the following.

1. The implementation of watermark in videos and copyright by embedding copyright in the video to protect the videos like visible and invisible watermark.

2. The application of copyright for both visible and invisible watermark into a sound clips with extension such as .mp3, .mp4, .wmv file and extract the copy of such watermarked audio file for the purpose of security.

3. To improve the user interfaces in the future with more advanced features.

REFERENCES

1. Archana C., Steganography Tutorial – A Complete Guide For Beginners, edureka, [Electronic Resource] URL: https://www.edureka.co/blog/steganography-tutorial (the date of access: 01.02.2019).

2. Athansios Z. / Z. Athanasios, P. Achilleas, p. Constantinos. // IEEE Xplore, Social Network Content Management through Watermarking. – United Kingdom, Liverpool: IEEE, 2012. – 6 p.

3. Bit Calculator, Understanding the most and least significant bit. [Electronic Resource] URL: https://bit-calculator.com/most-and-least-significant-bit (the date of access: 10.02.2019).

4. Cherdantseva Y., Hilton J. Information Security and Information Assurance. The Discussion about the Meaning, Scope and Goals. – United Kingdom, Cardiff: Cardiff University, 2013. – 5 p.

5. COMODO, What is Information Security(IS). [Electronic Resource] URL: https://enterprise.comodo.com/what-is-information-security.php (the date of access: 20.02.2019).

6. Computer Hope, Least significant bit. [Electronic Resource] URL: https://www.computerhope.com/jargon/l/leastsb.htm (the date of access: 05.02.2019).

7. Cornell, CS 3110 Lecture 21 Hash functions. [Electronic Resource] URL: http://www.cs.cornell.edu/courses/cs3110/2008fa/lectures/lec21.html (the date of access: 15.02.2019).

8. Definitions, least significant bit. [Electronic Resource] URL: https://www.definitions.net/definition/least+significant+bit (the date of access: 25.02.2019).

9. DIGITAL WATERMARKING. [Electronic Resource] URL: http://digitalwatermarkingalliance.org/digital-watermarking-works/

10. Frank Y.S. Digital watermarking and steganography: fundamentals and techniques. – USA, Boca Raton: CRC Press, 2008. – 200 p.

11. Fridrich J., Wet paper codes with improved embedding efficiency IEEE Trans Information Forensics and Security. // J. Fridrich, M. Goljan, D. Soukal. – USA, Binghamton: Binghamton University, 2006. – 110 p.

12. GeeksforGeeks, What is Information Security? [Electronic Resource] URL: https://www.geeksforgeeks.org/what-is-information-security/ (the date of access: 28.02.2019).

13. GURU99, what is Software Testing? Introduction, Definition, Basics & Types. [Electronic Resource] URL: https://www.guru99.com/software-testing-introduction-importance.html (the date of access: 01.02.2019).

14. Ipfs Least significant bit. [Electronic Resource] URL: https://ipfs.io/ipfs/QmopizjW3WknFiJnKLwHCnL72vedxjQkDDP1mXWo6uco/wiki/Least_significant_bit.html (the date of access: 12.02.2019).

15. Jonathan K. SECURITY MEASURES. // PCWorld. [Electronic Resource]URL:https://www.pcworld.com/article/3021316/why-stolen-laptops-still-cause-data-breaches-and-whats-being-done-to-stop-them.html

16. Kumar S., Dutta A. A novel spatial domain technique for digital image watermarking using block entropy In Recent Trends in Information Technology. – USA, Piscataway: IEEE, 2016. – 6 p.

17. Paramhans K. Design and Development of Secured Image Watermarking for Embedded System Applications. // K. Paramhans, S. Pradeepkumar, C. Veerendra. – India, Karnataka: Shri Dharmasthala Manjunatheswara College of Engineering & Technology Dharwad, 2012. – 53 p.

18. Samonas S., Coss D. The CIA Strikes Back: Redefining Confidentiality, Integrity and Availability in Security. – USA, Virginia: Virginia Commonwealth University, 2014. – 45 p.

19. Software Testing Help, Functional Testing: A Complete Guide with Types and Example. [Electronic Resource] URL: https://www.softwaretest-inghelp.com/guide-to-functional-testing/

20. Software Testing, Functional Testing. [Electronic Resource] URL: http://softwaretestingfundamentals.com/functional-testing/

21. St Petersburg University, Programming and Information Technology. [Electronic Resource] URL: http://english.spbu.ru/education-at-spbu/undergradu-ate/bachelor/85-program-bac/1653-programming-and-information-technology (the date of access: 01.02.2019).

22. Stewart M., James W. CISSP Study Guide. – Canada: John Wiley & Sons, Canada, Indianapolis: Indiana, 2012. – 257 p.

23. Ukessays, The History Of The Digital Watermarking Techniques. [Elec-tronic Resource] URL: https://www.ukessays.com/essays/information-technol-ogy/the-history-of-the-digital-watermarking-techniques-information-technology-essay.php (the date of access: 06.02.2019).

24. Yorku,HashFunctions. [ElectronicResource] URL: http://www.cse.yorku.ca/~oz/hash.html (the date of access: 11.02.2019).