

# **MOBILE APPLICATION FOR MEDICAL EMERGENCY REQUEST**

**(MER)**

**Supervised by:-**

**Dr.Ahmed Rafaat**

**Instructor :-**

**Eng.Ahmed EL-Refaiy**

# WORK GROUP



- ✖ *Menna Essam Elsayed*
- ✖ *Amany Elbakry*
- ✖ *Amira Mahmoud*
- ✖ *Dalia Ebrahim*
- ✖ *Monika karam*

# AGENDA

- ✖ Chapter 1 : introduction
- ✖ Chapter 2 : General Methodology.
- ✖ Chapter 3 : Design and Analysis
- ✖ Chapter 4 : Web Screens (design & develop)
- ✖ Chapter 5:Mobile Screens
- ✖ Security of Project and Future work

# CHAPTER 1

Introduction

# IDEA OF PROJECT

- *Help people in Satisfactory state of emergency.*
- *Useful for :-*
  - *older people*
  - *Disabled , paralyzed people*
  - *Deaf and speechless people*

- ❖ add voice record to help blind people
- ❖ add near Police Station for Distress police it is useful for :-

### ➤ Theft cases

### ➤ girl in Critical condition

- ❖ Note : may add Distress police and Satisfactory state of emergency it is useful in Accident

FUTURE WORK  
LOOK MOVE

# CHAPTER 2

General Methodology.

# GENERAL METHODOLOGY:

## ‘TOOLS’

- ✖ *Phone Gap / Cordova*
- ✖ *PHP*
- ✖ *My SQL*
- ✖ *Web Service (c#)*
- ✖ *Web Design Tools*  
*(Html5,Css3,JavaScript,Jquery,Bootsrtap)*

# PHONE GAP

- ✖ *Develop in 2012 to make the mobile application easier*
- ✖ *Phone Gap is a software development framework by Adobe System, which is used to develop mobile applications. To develop apps using Phone Gap, the developer does not require to have knowledge of mobile programming language but only web-development languages like, HTML, CSS, and JavaScript. Phone Gap produces apps for all popular mobile OS platforms such as iOS, Android, BlackBerry, and Windows Mobile OS, etc*

# PHP

- ✖ *PHP: PHP is an acronym for "PHP: Hypertext Preprocessor"*
- ✖ *PHP is a widely-used, open source scripting language*
- ✖ *PHP scripts are executed on the server*
- ✖ *PHP is free to download and use*
- ✖ *PHP files can contain text, HTML, CSS, JavaScript, and PHP code*

# MySQL

- ✖ *MySQL is a database system that runs on a server*
- ✖ *MySQL is ideal for both small and large applications*
- ✖ *MySQL is very fast, reliable, and easy to use*
- ✖ *MySQL uses standard SQL*
- ✖ *MySQL compiles on a number of platforms*
- ✖ *MySQL is free to download and use*
- ✖ *MySQL is developed, distributed, and supported by Oracle Corporation*
  
- ✖ *MySQL is a database system used on the web*

# CHAPTER 3

## CIVIL LAW

DataBase

# FRIST DATABASE TABLES

- ✖ Manager
- ✖ Employee
- ✖ Doctor
- ✖ Patient
- ✖ Medical Center
- ✖ Massage

# MANAGER

# MANAGER (CONTINUE)

- ✖ First Name , Last Name ,Email ,Birthday  
(optional)
- ✖ Password is Encrypted to protect it from Trojan horse programs .
  - Example : Noha password is 123 but it in database is “40bd001563085fc35165329ea”

# EMPLOYEE FINAL FO LEE

# EMPLOYEE (CONTINUE)

- ✖ City , birthday , email (optional )
- ✖ Password is Encrypted
- ✖ Shift is base because it determine who employee accept message and who reject

# DOCTOR

# DOCTOR (CONTINUE)

- ✖ Age , gender , address (optional)
- ✖ Specialized is base it help employee / manager to select who doctor available to patient
- ✖ Based on The patient's condition

# PATIENT LEVEL

# PATIENT (CONTINUE)

- ✖ All data is necessary to help employee .
- ✖ To select available doctor to help patient .

# MEDICAL CENTER

idMedicalCenter	location	specialized	phone	name
1	elgame	general	011235256	Elgama
2	magles elmadina st	kids	012321456 055/2241025	Elmontza
3		general	011232565 055/2236547	Elahrar
4	12 st el Qawmia	heart	010232563 011236454	Elhramin
NULL	NULL	NULL	NULL	NULL

# MEDICAL CENTER (CONTINUE)

- ✖ Name , location (optional)
- ✖ Contain all emergency area based on Location.

# MASSAGE

# MESSAGE (CONTINUE )

- ✖ Must contain ID\_Patient , ID\_Doctor , ID\_employee to determine who work good
- ✖ To keep on the security Manager hasn't the power to answer on msg

# CHAPTER 4

Design and Analysis

# SYSTEM ANALYSIS AND DESIGN:

## System Analysis:

The dissection of a system into its component pieces to study how those component pieces interact and work.

The survey and planning

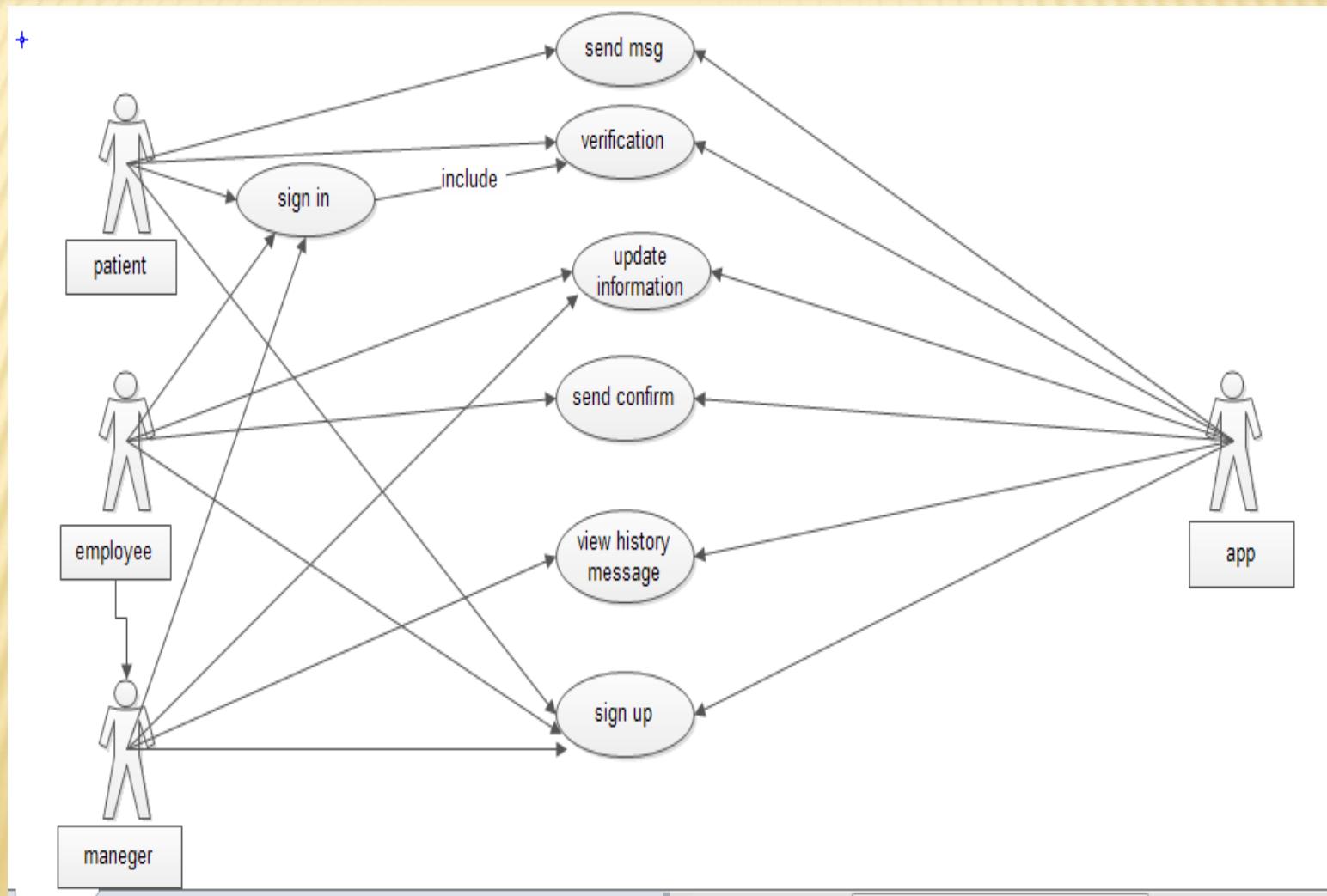
The study and analysis

The definition

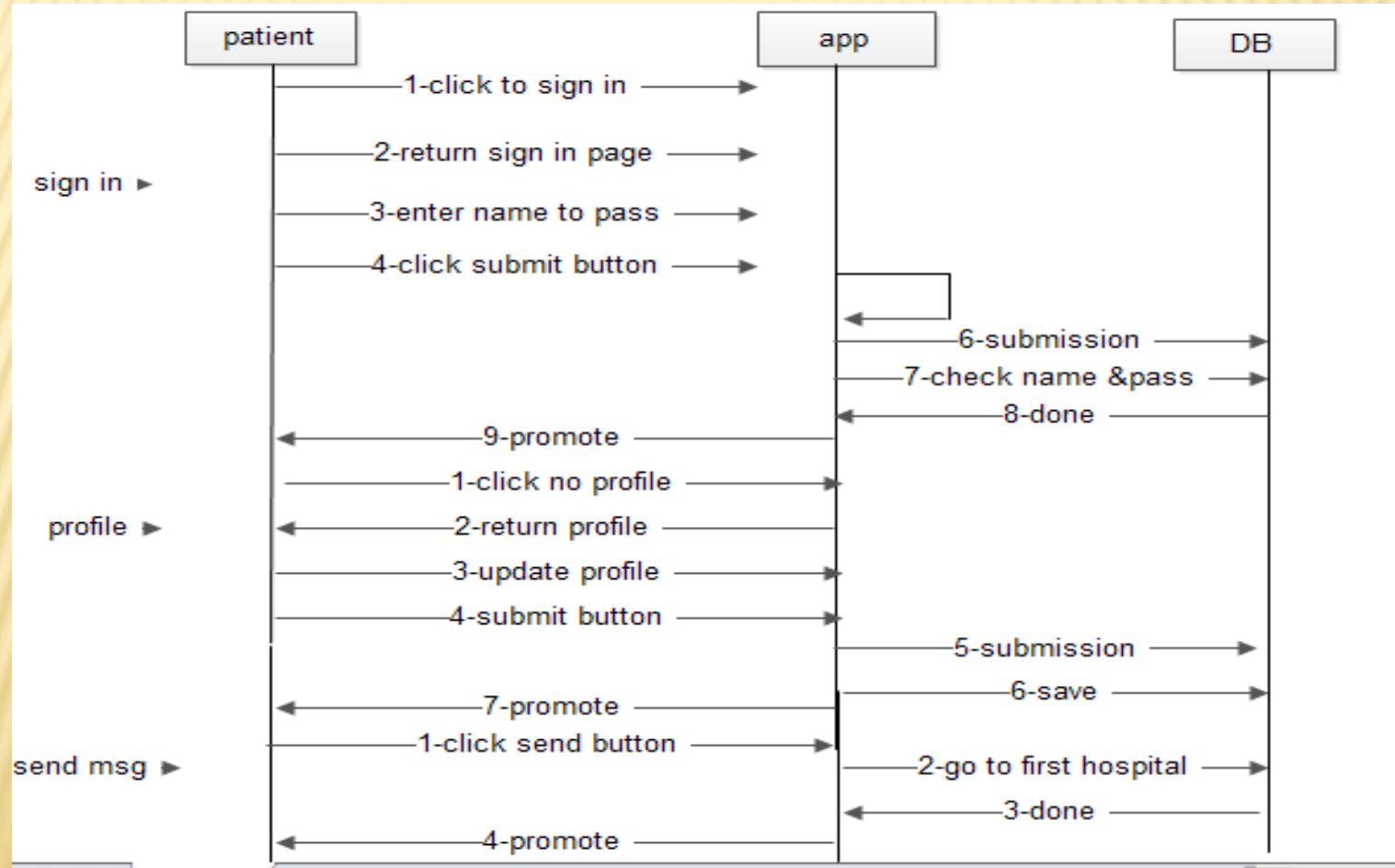
## System Design:

The process of defining the architecture , components , modules , interfaces and data for a system to satisfy specified requirements.

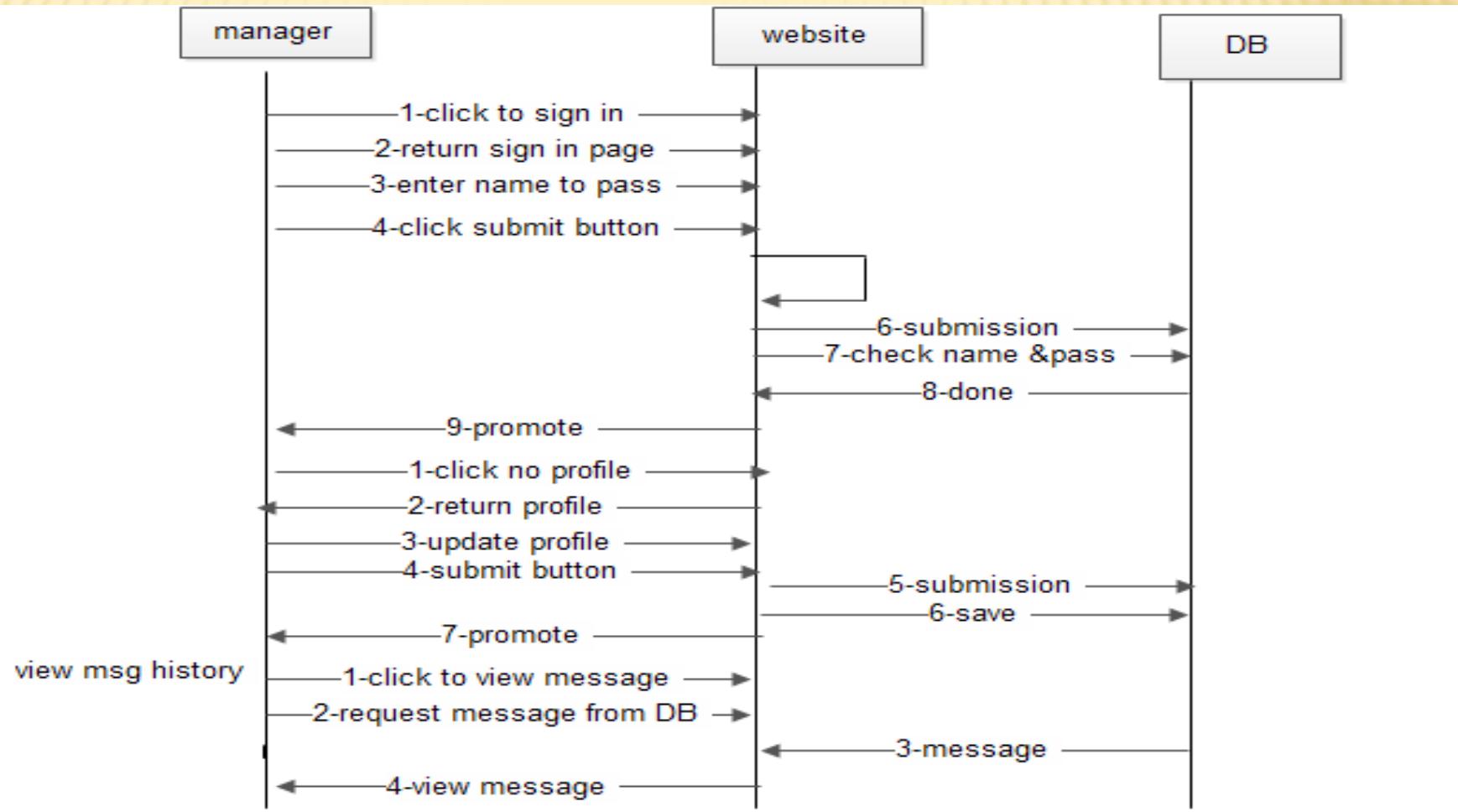
# 1-USE CASE



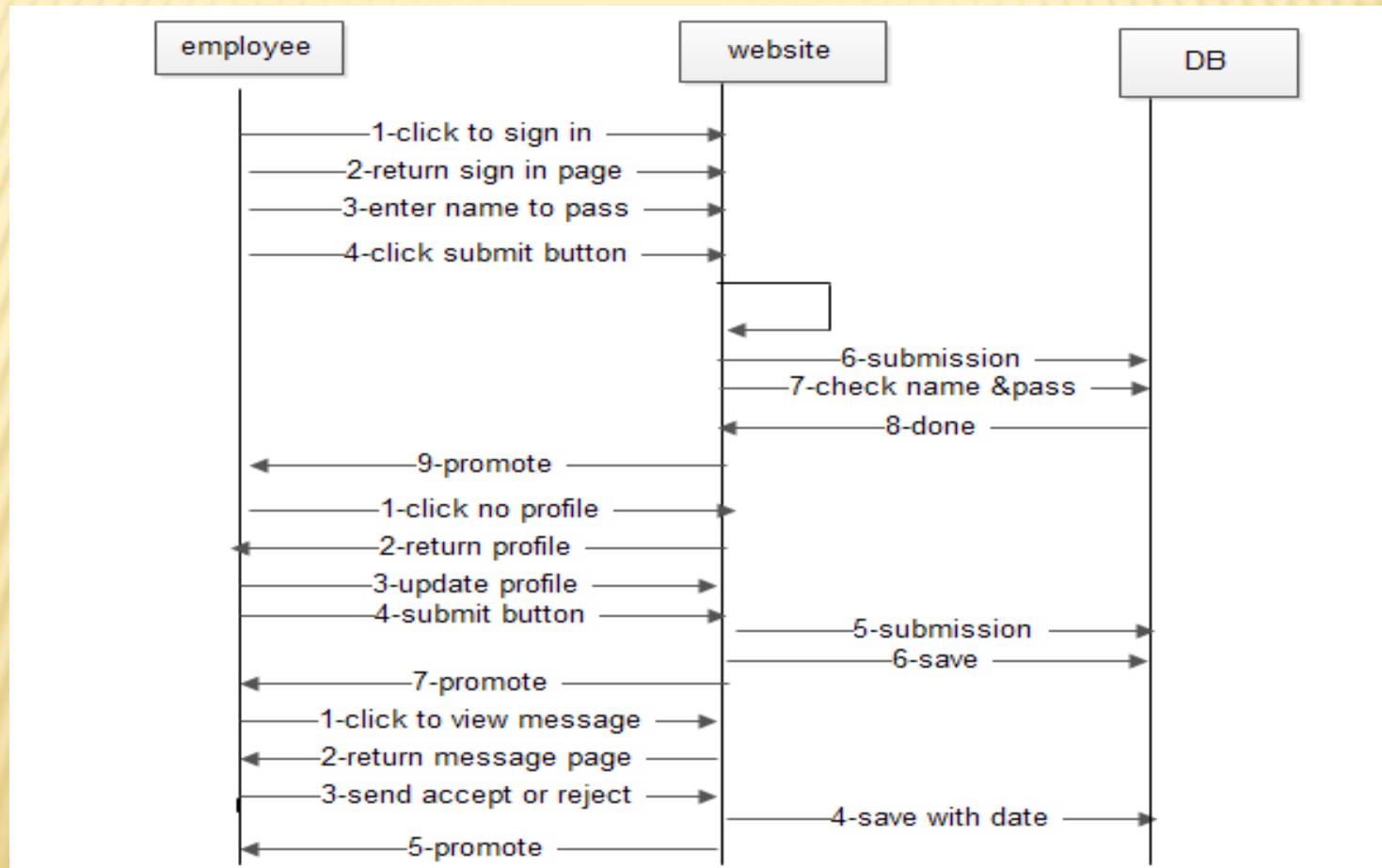
## 2- SEQUENTIAL FOR PATIENT:



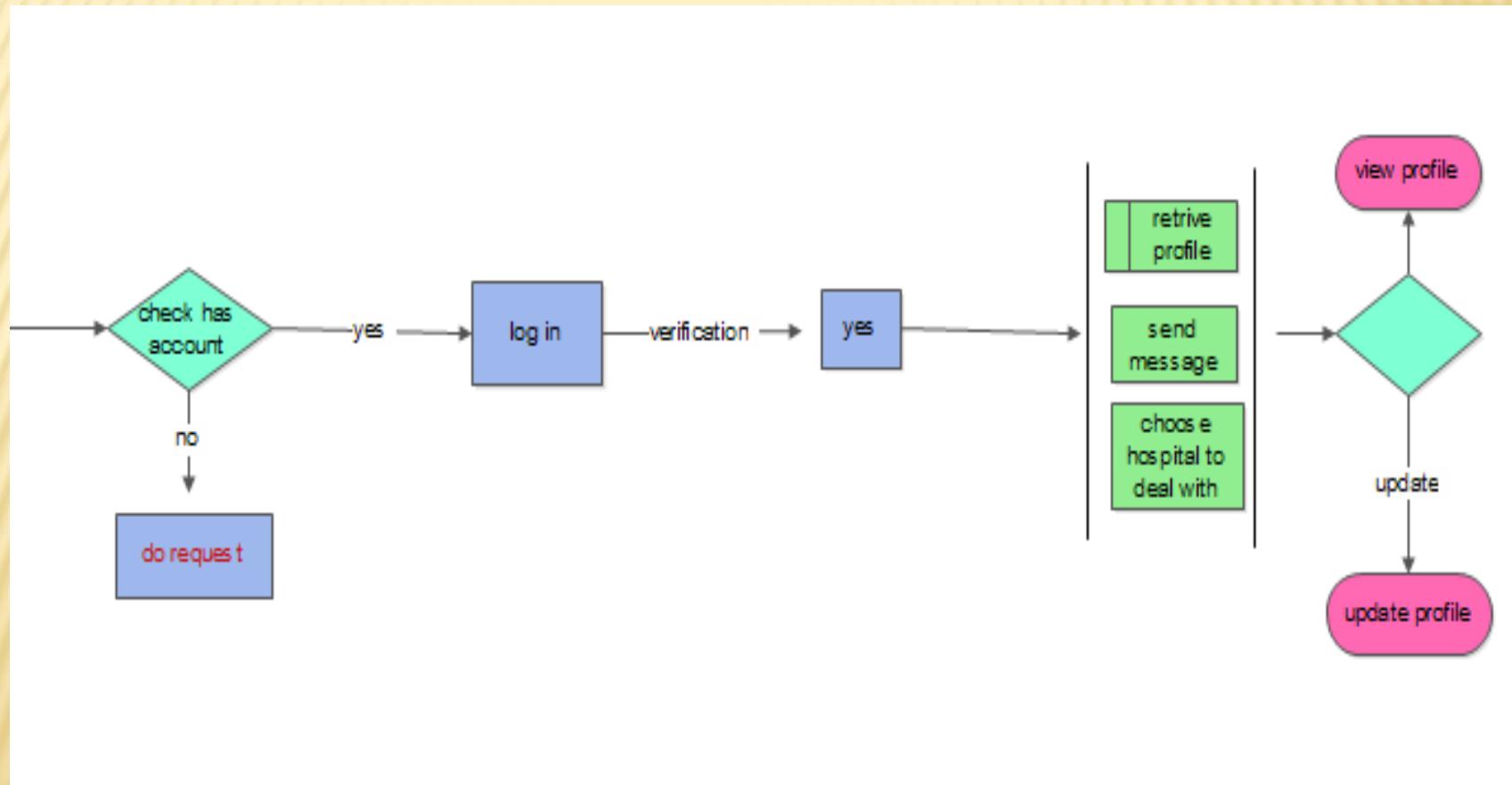
### 3- SEQUENTIAL FOR MANAGER:



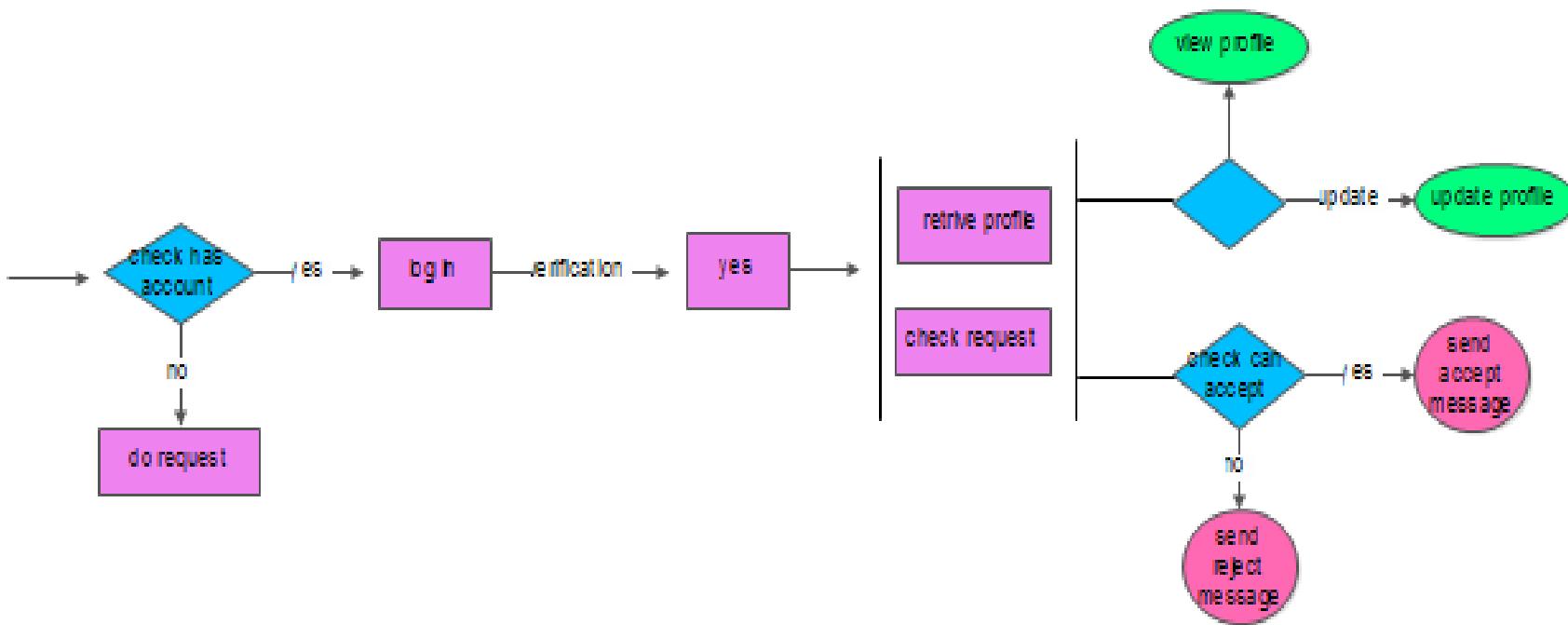
## 4- SEQUENTIAL FOR EMPLOYEE:



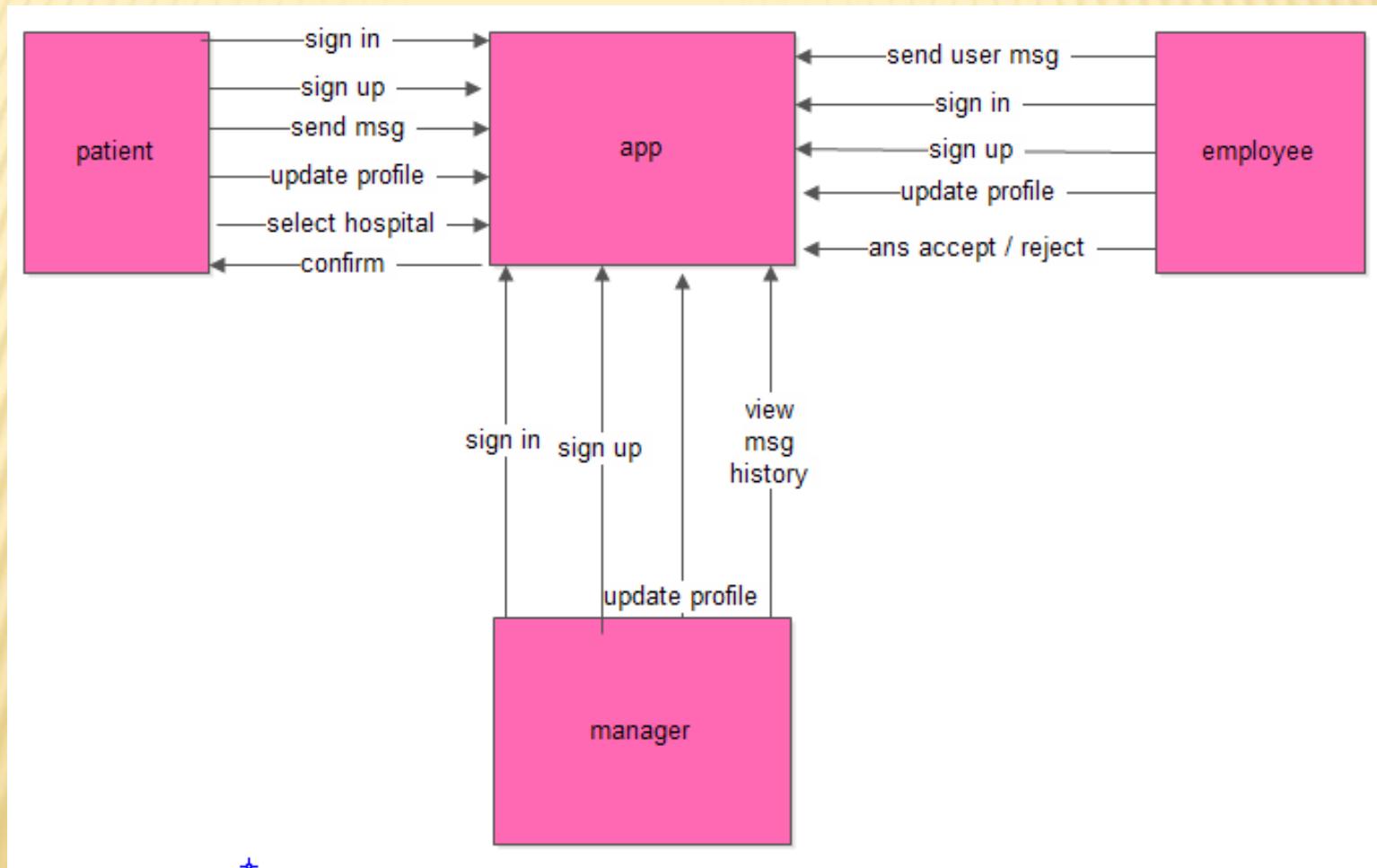
# 5- ACTIVITY FOR PATIENT:



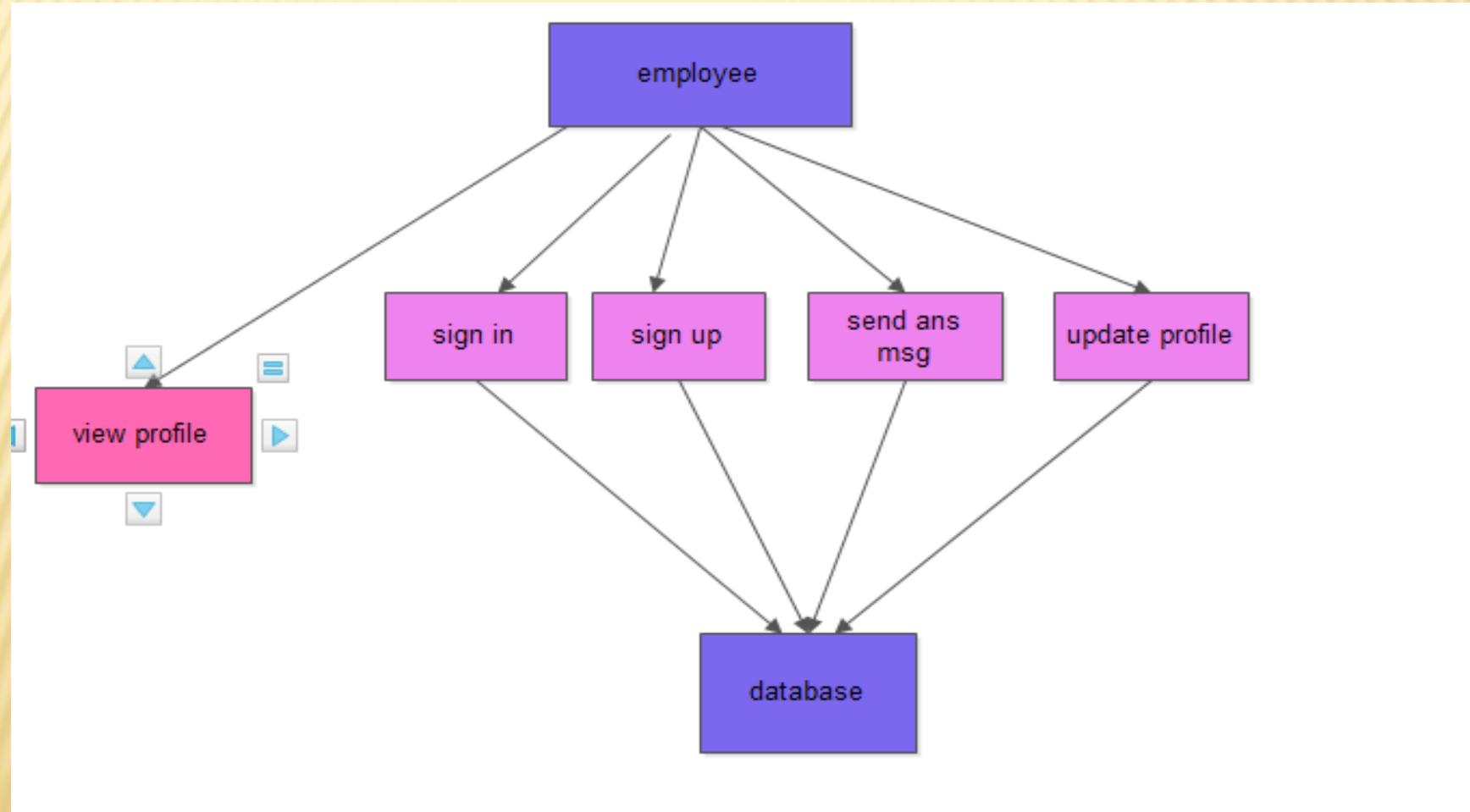
# 6- ACTIVITY FOR EMPLOYEE:



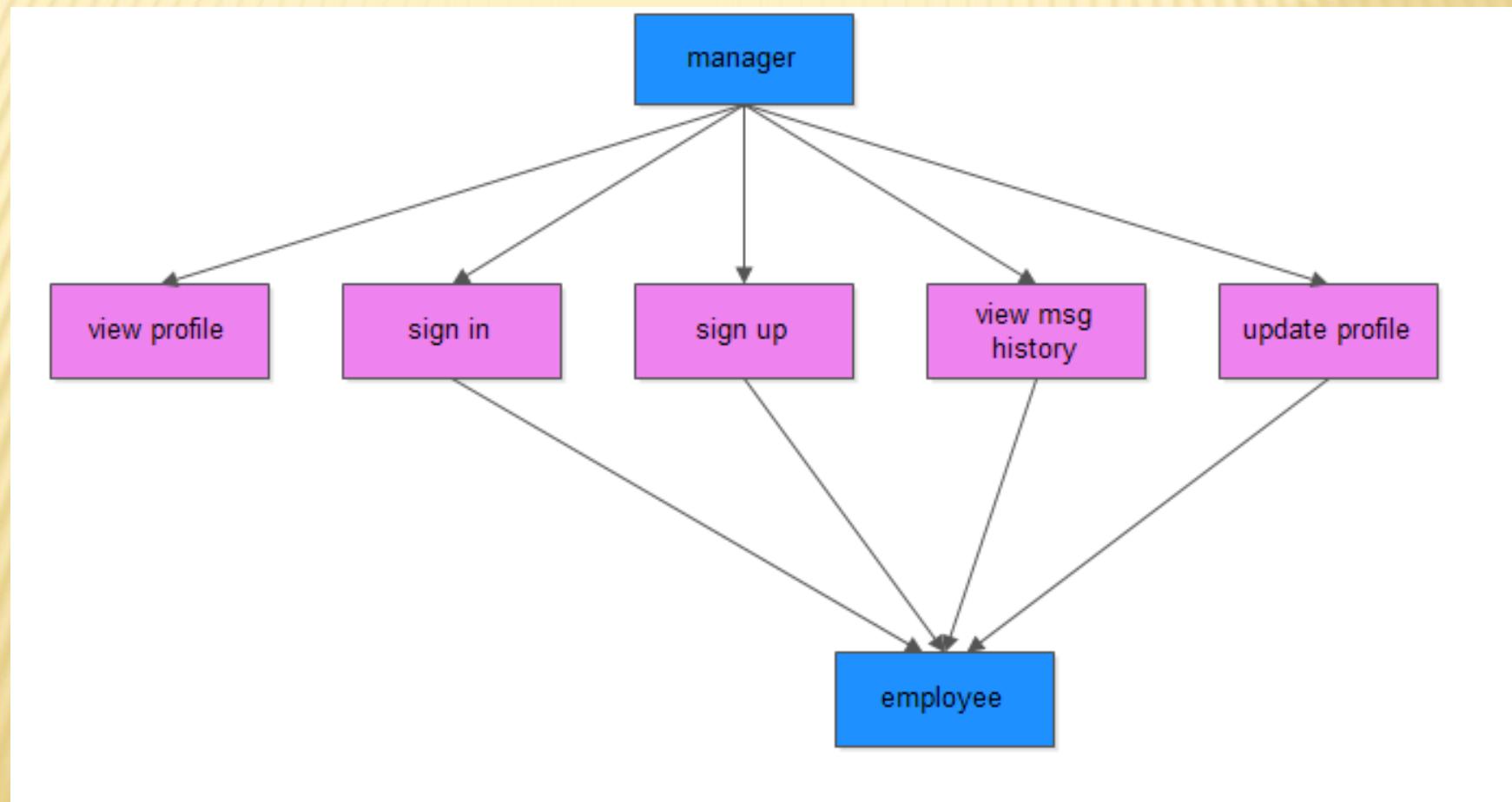
# 7- DFD CENTER LEVEL:



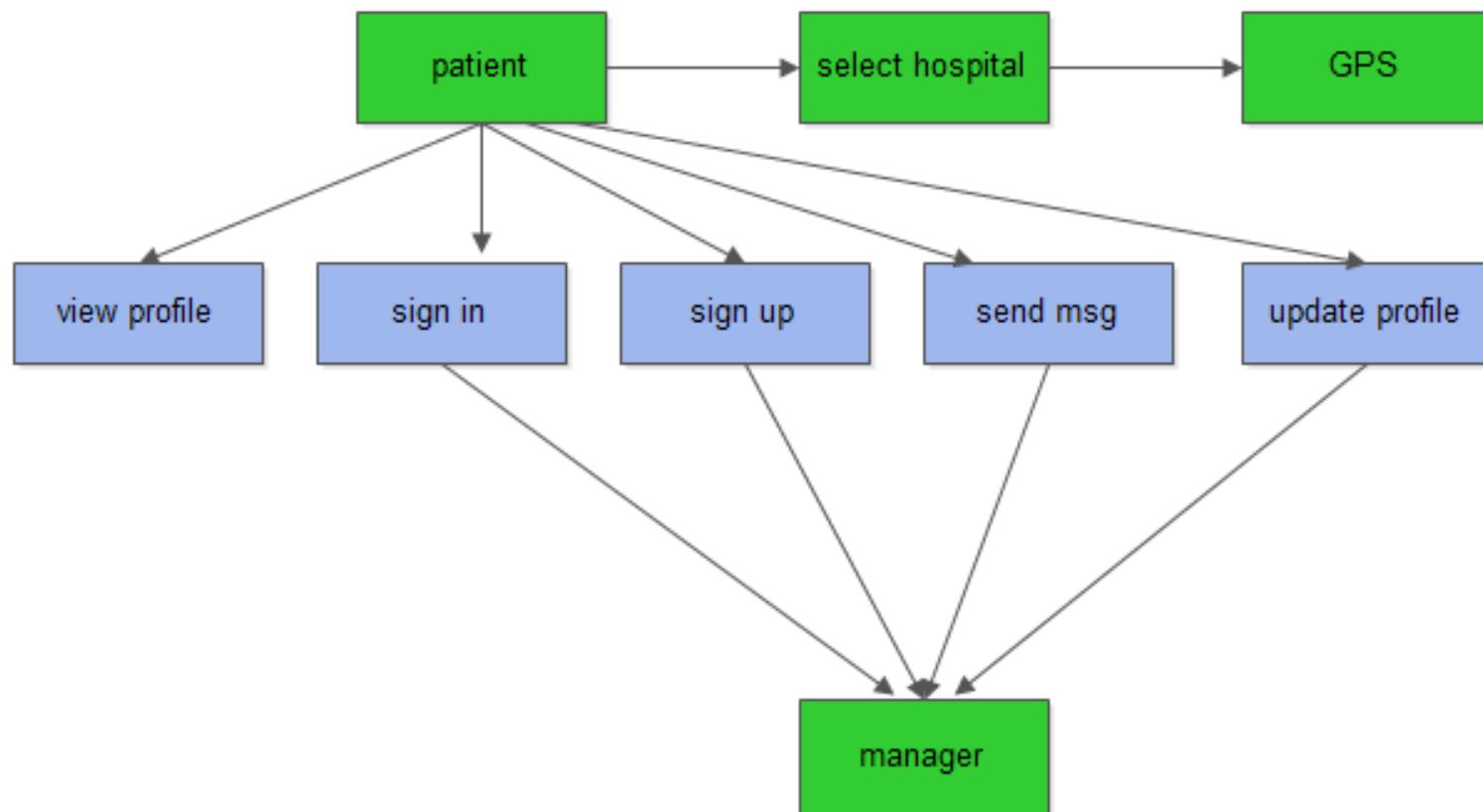
# 8- DFD FOR EMPLOYEE:



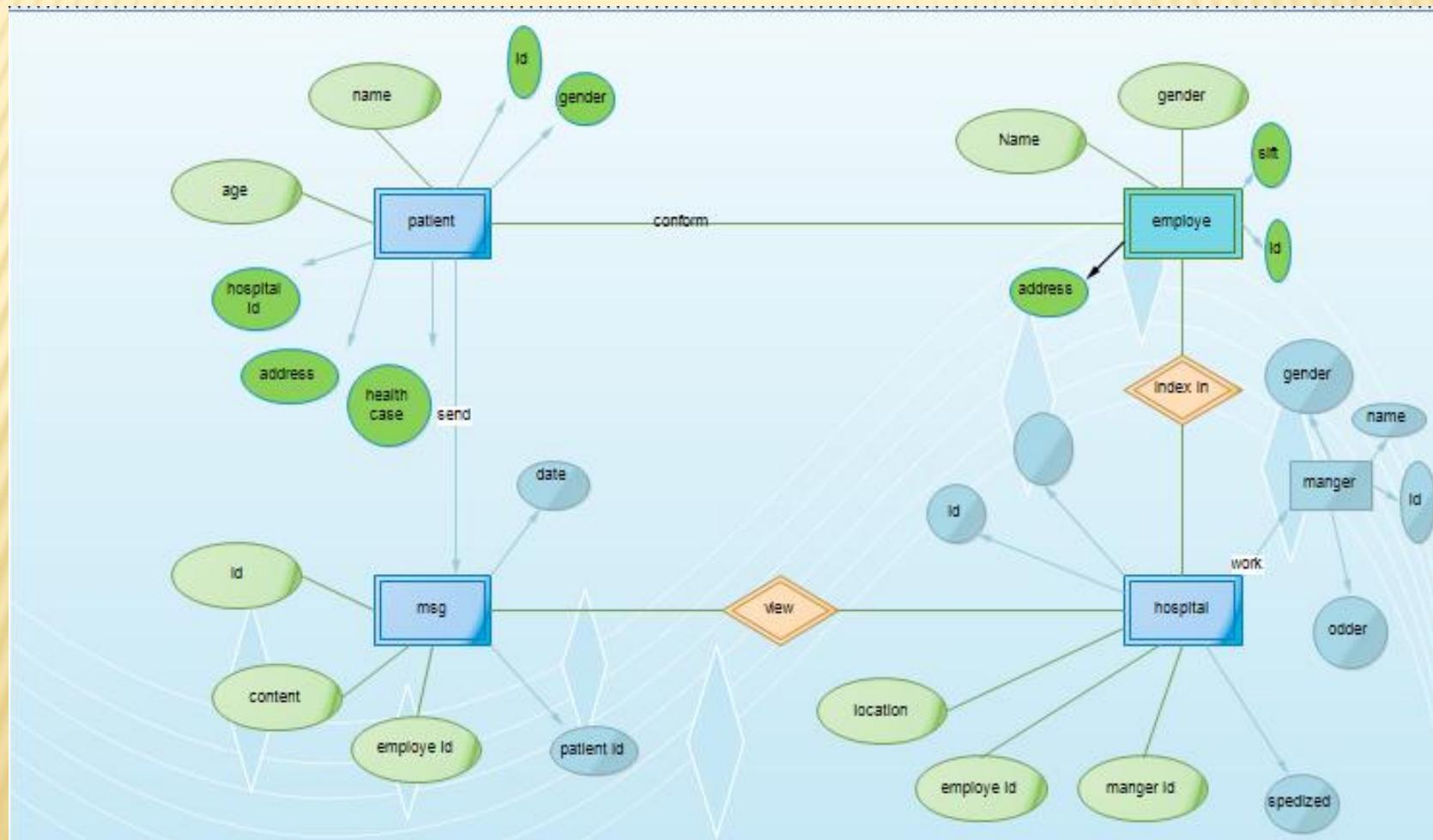
# 9-DFD FOR MANAGER:



# 10- DFD FOR PATIENT:



# 11-ER DIAGRAM



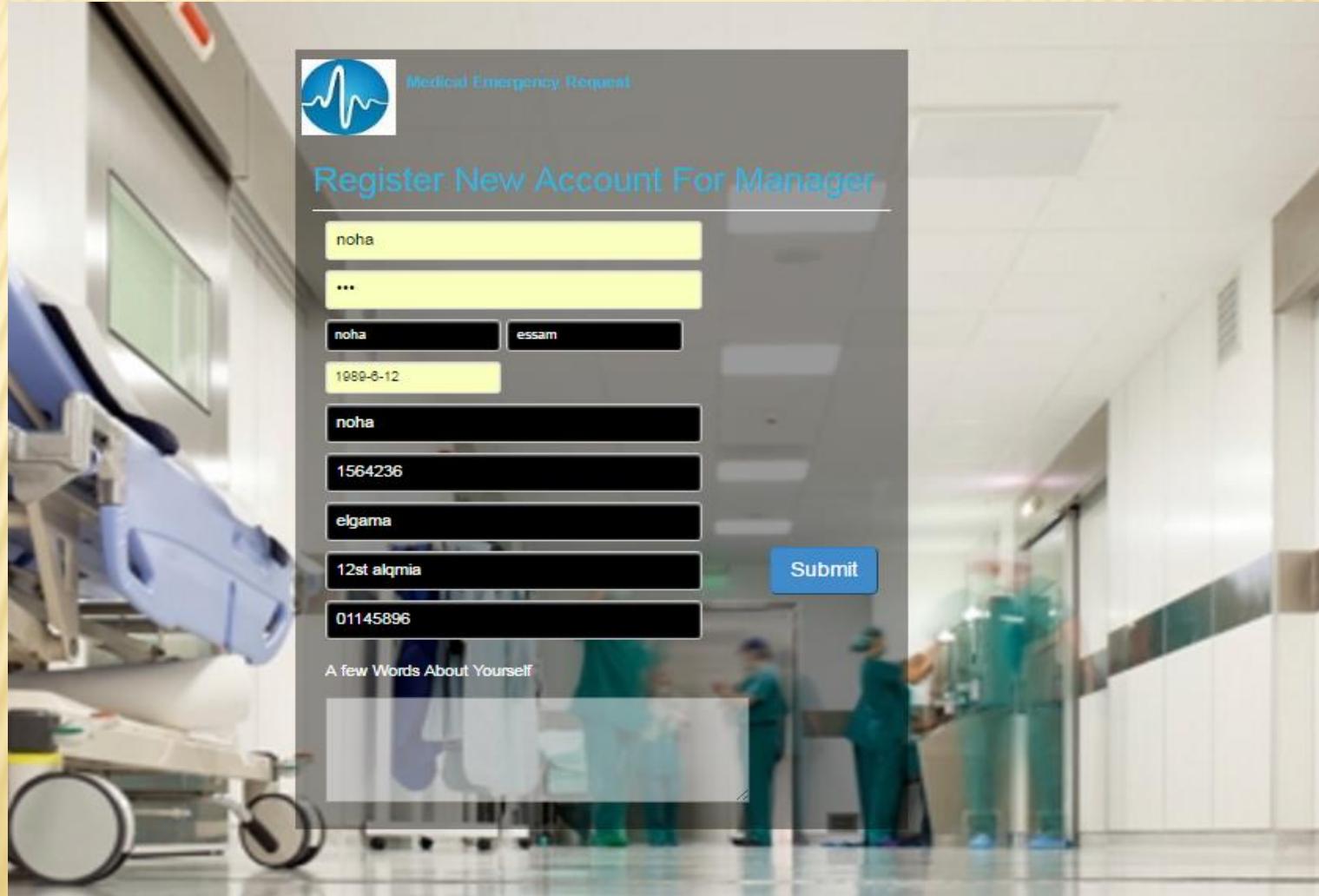
# CHAPTER 5

Web Screens (design & develop)

# SCREENS

- ✖ Sign in
- ✖ Registration to Manager
- ✖ Sign in
- ✖ Registration to Employee
- ✖ Home to Manager
- ✖ Home to Employee

# SIGN IN TO MANAGER



# HOME TO MANAGER

Medical Emergency Request

Manager Information

✓ UserName : Username  
✓ Address : 112 Ahmed Orabi  
✓ Phone : 0123456789

UPDATE

A doctor's profile picture is displayed.

idmsg	date	content	id_doctor	id_patient	id_employee	reaction
1	2016-01-02 132344.00000	help	1	1	1	done
2	2016-01-11 102344.00000	help me	2	2	1	no
3	2016-01-11 02344.00000	Sugar	1	1	2	no

Dr / Hazzem  
My rating: ★★★★★  
phone: 0123456789  
Address: 1st bahthi bathta

Dr / Noha  
My rating: ★★★★★  
phone: 01023456789  
Address: 1st bahthi bathta

Dr / Amira  
My rating: ★★★★★  
phone: 01123456789  
Address: Sanaa

Dr / Samia  
My rating: ★★★★★  
phone: 0123456789  
Address: Sanaa

Dr / Hassan  
Specialist: eyes

Dr / Noha  
Specialist: Cancer

Dr / Samia  
Specialist:

Doctors

Shift	Found Now	Not Found
Amal 10 AM - 3 PM	Amal	Nada
Nada	Amal	Nada
Amal 10 AM - 3 PM	Amal	Nada
Nada	Amal	Nada
Amal 3 PM - 6 PM	Amal	Nada
Nada	Amal	Nada
Amal 3 PM - 6 PM	Amal	Nada
Nada	Amal	Nada

ABOUT APPLICATION  
This application help in emergency request. It is useful for elderly, children, girls in problem.



#### INFORMATION

Home  
Messages  
Facilities  
Doctors  
Pricing

#### MER Team

If you have any problem contact us



If you have any problem calling us

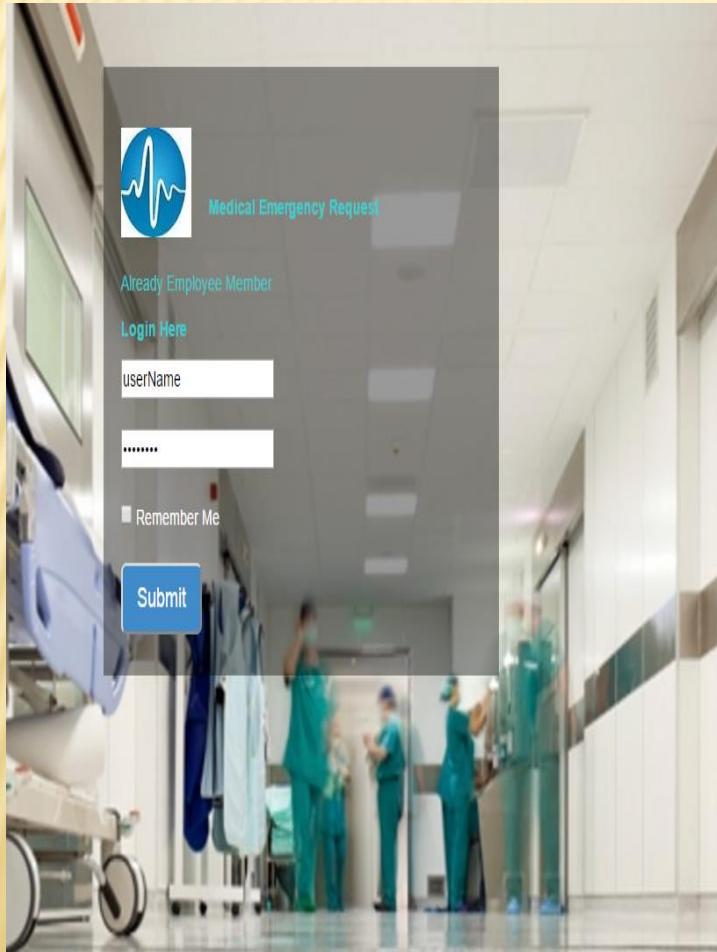


0123456789



merteam@yahoo.com

# SIGN IN , REGISTRATION TO EMPLOYEE



A photograph of a hospital corridor with medical staff in scrubs. A semi-transparent registration form is overlaid on the image.

Medical Emergency Request

Register New Account For Employee

Email

Password

First name Last name

Birth Of Date

Username

Shift

City/Town phone

gender

A few Words About Yourself

# HOME TO MANAGER (CONTINUE)

- ✖ Page contain table of massages that employee
- ✖ Receive to manager see if employee Lazy to answer them

# HOME TO EMPLOYEE

The screenshot shows a medical application interface. At the top, there's a navigation bar with links: HOME, HELP REQUEST, FACILITIES, DOCTORS, and PRICING. Below the navigation is a section titled "Medical Emergency Request" with a sub-section "employee information". It lists three items with green checkmarks: "UserName : Ahmad", "Address : 113 Sector", and "Phone : 01234567890". A blue "UPDATE" button is located at the bottom right of this section. To the right of this is a photo of a female doctor in a white coat and stethoscope. Below this main section is a large banner featuring a doctor in a white coat and a hexagonal grid of medical icons (heart, brain, eye, etc.). On the left side of the banner, there are three patient records:

- patient\_id : 1  
health case : heart  
phone : 0104567890  
address : 9 et  
Els...
- patient\_id : 3  
health case : sugar  
phone : 02222598380  
address : 5 et  
Els... , Els...
- patient\_id : 4  
health case : cancer  
phone : 0100235612399  
address : 16 et  
Els... , rns

At the bottom of the page, there are three doctor profile cards:

- Dr / Hazzem: My rating: ★★★★☆; phone: (012) 345400; address: 1 et bahria tower.
- Dr / Noha: My rating: ★★☆☆☆; phone: (012) 199176; address: bapue
- Dr / Amira: My rating: ★★★★★; phone: (012) 111547; address: Saudi

This screenshot shows a "Doctors" availability grid. The grid has three columns: "Found Now", "Shift", and "NOT Found". Each column contains a list of doctors with their availability status (e.g., Available or Not Available) and time range (e.g., 10 AM - 3 PM). A vertical scroll bar is visible on the right side of the grid.

Doctors	Doctors	Doctors
Found Now	Shift	NOT Found
Ahmed	Available	Nada
Amira	Available	Not Available
Amira	10 AM - 3 PM	
Hazem	Available	
Ahmed	10 AM - 3 PM	
Amira	3 AM - 9 PM	
Khalid	3 PM - 9 PM	
Nada	3PM - 9PM	
Mohamed	9 PM - 1 PM	
Amira	9PM - 4PM	

ABOUT APPLICATION  
This application help in emergency request. It is useful for elderly , children , girls in problem .  
It records GPS/LCD successfully

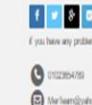
#### INFORMATION

Home  
Message  
Facilities  
Doctors  
Pricing

#### MER Team

If you have any problem calling us

#### FOLLOW US



# **HOME TO EMPLOYEE(CONTINUE)**

- Has massages that send from patient ✘
- It refresh outomatic to has new massages ✘

```
-->
<html>
  <head>
    <meta charset="UTF-8">
    <title></title>
  </head>
  <body>
    <?php
      $dsn = 'mysql:host=localhost;dbname=mer';
      $user = 'root';
      $pass = '';
      $option = array(
        PDO::MYSQL_ATTR_INIT_COMMAND => 'SET NAMES utf8 ',
      );
      try {
        $db = new PDO($dsn, $user, $pass, $option);
        $db->setAttribute(PDO::ATTR_ERRMODE, PDO::ERRMODE_EXCEPTION);
        echo 'connect';

      } catch (PDOException $e) {
        echo 'faild ' . $e->getMessage();
      }
    ?>
  </body>
</html>
```

```
-->
<html>
    <head>
        <meta charset="UTF-8">
        <title></title>
    </head>
    <body>
        <?php
        include 'connect.php';

        if (filter_input(INPUT_SERVER, 'REQUEST_METHOD', FILTER_SANITIZE_STRING) == 'POST') {
            $username = filter_input(INPUT_POST, 'user');
            $password = filter_input(INPUT_POST, 'pass');
            $hashedpass = sha1($password);
            $stmt = $db->prepare("select name ,password FROM employee WHERE name =? AND password = ?");
            $stmt->execute(array($username, $hashedpass));
            include './employee.php';
        } else
            echo ' error name or pass';
        ?>
    </body>
</html>
```

```
<!DOCTYPE html>
<html>
<head>
<style>
table, th, td {
    border: 1px solid black;
    color: black;
    margin: 20px;
    padding: 10px;
}
table
{
    margin-left: -50px;
}
</style>
</head>
<body>
<?php
$servername = "localhost";
$username = "root";
$password = "";
$dbname = "mer";

// Create connection
$conn = new mysqli($servername, $username, $password, $dbname);
// Check connection
if ($conn->connect_error) {
    die("Connection failed: " . $conn->connect_error);
}

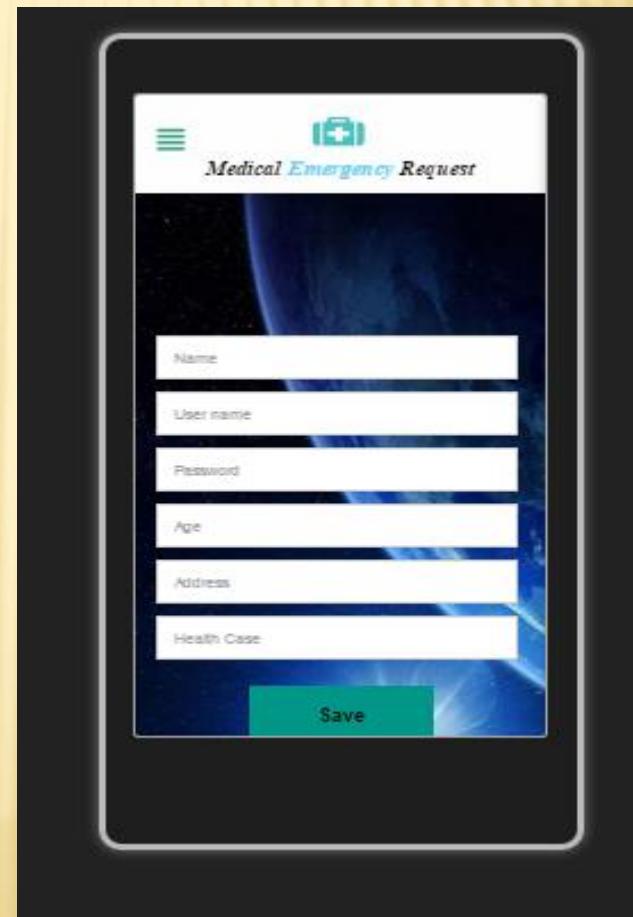
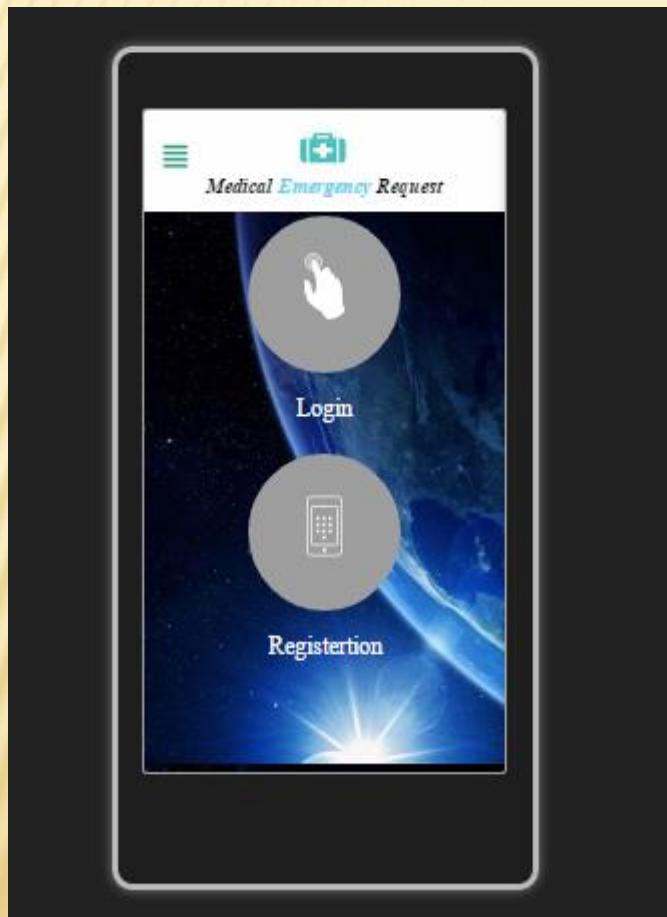
$sql = "SELECT idmsg,date,content,id_doctor,id_patient,id_employee
,reaction,doctorReport FROM msg";
$result = $conn->query($sql);

if ($result->num_rows > 0) {
    echo "<table><tr><th>Massage</th><th>date</th><th>Health
Problem</th><th>doctor Number</th><th>patient numbers</th><th> employee
number</th><th>response </th><th>report</th></tr>";
    // output data of each row
    while($row = $result->fetch_assoc()) {
        echo "<tr><td>" . $row["idmsg"] . "</td><td>" . $row["date"] . " </td><td>" .
        $row["content"] . " </td><td>" . $row["id_doctor"] . " </td><td>" . $row
        ["id_doctor"] . " </td><td>" . $row["id_employee"] . " </td><td>" . $row["reaction"] . " "
        "</td> <td>" . $row["doctorReport"] . "</td></tr>";
    }
    echo "</table>";
} else {
    echo "0 results";
}
```

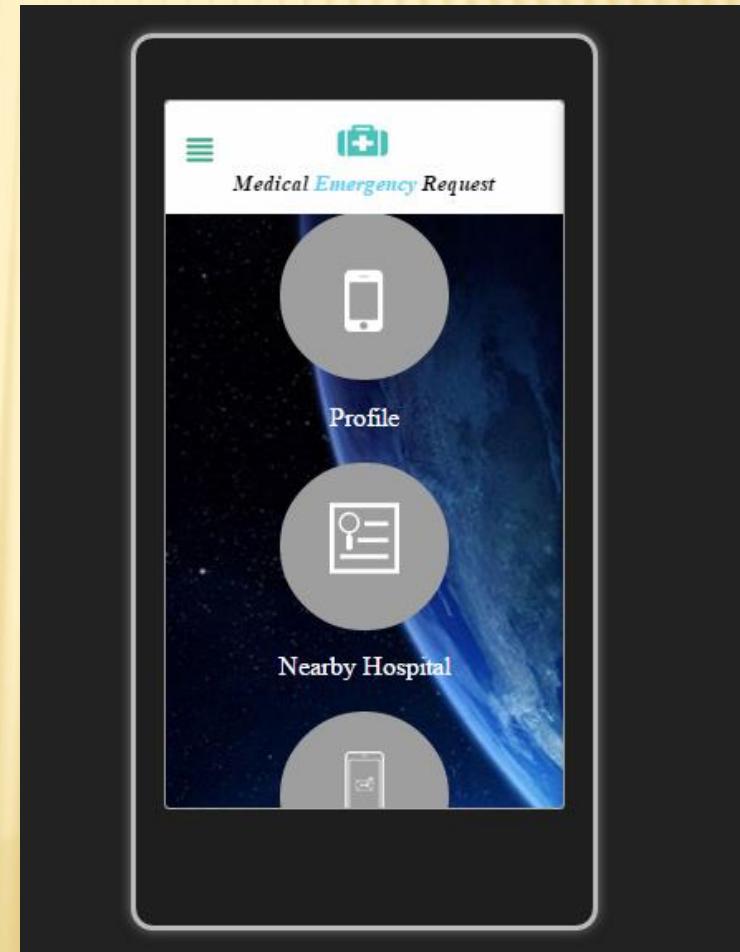
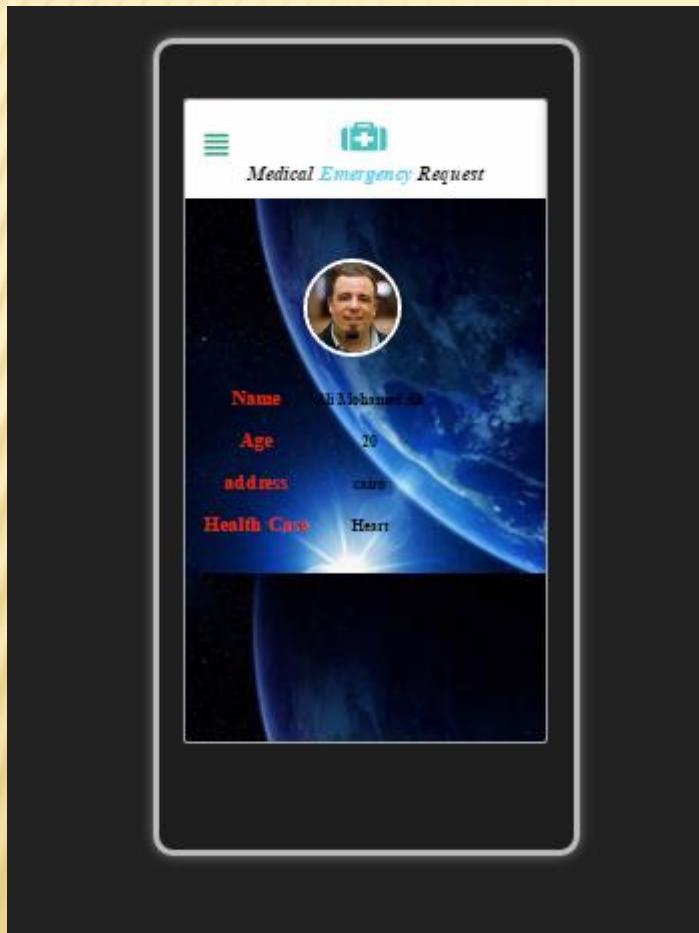
# Chapter 6

MOBILE SCREENS

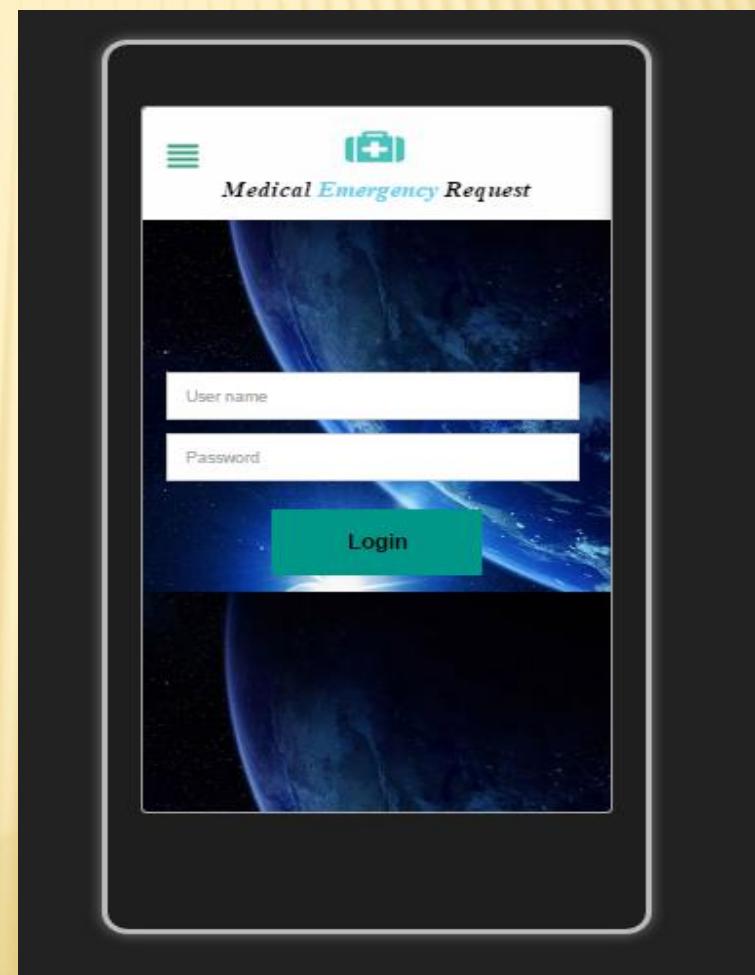
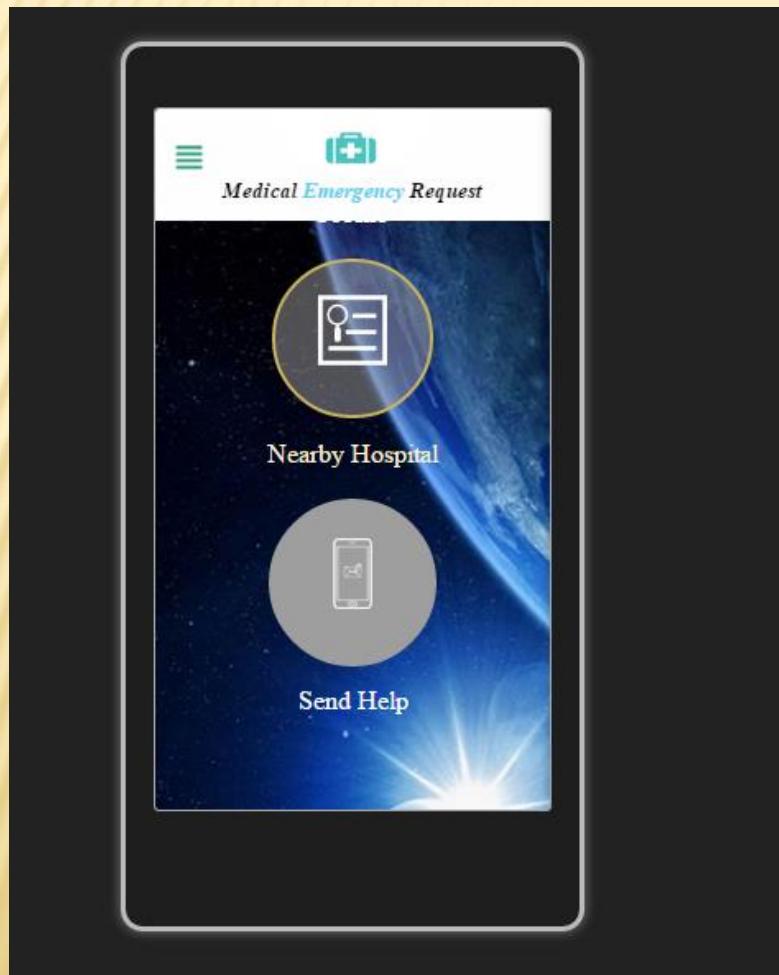
# MAIN MOBILE SCREEN



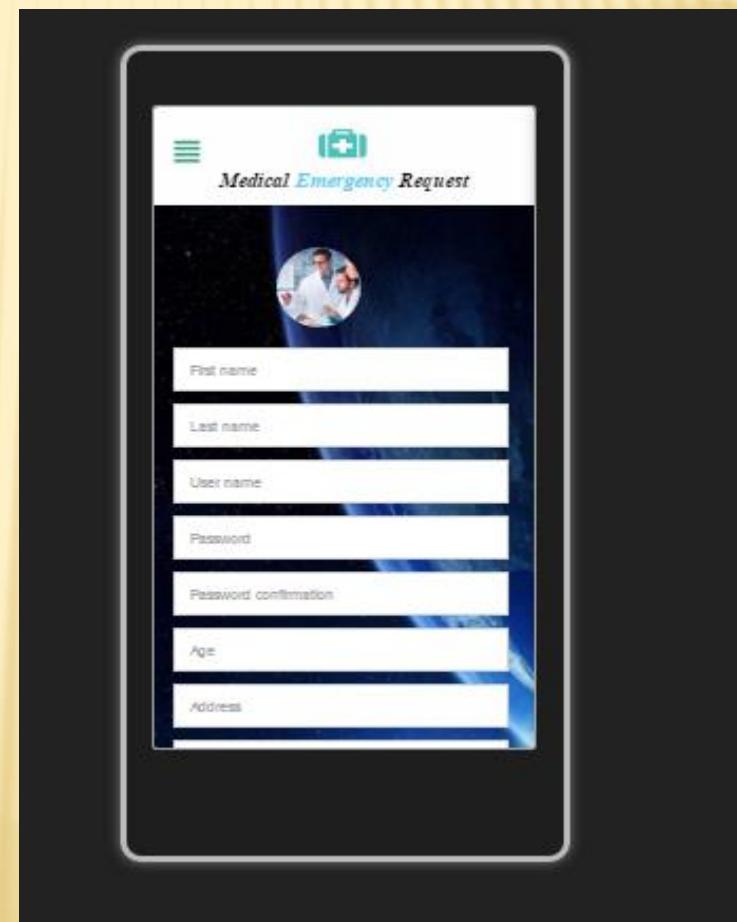
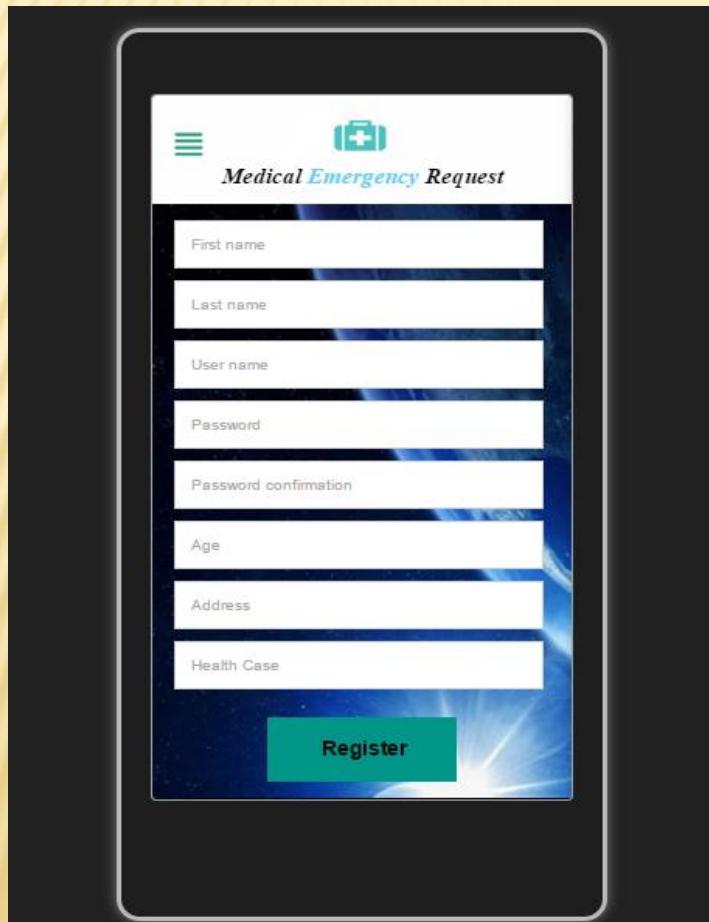
# MOBILE SCREEN



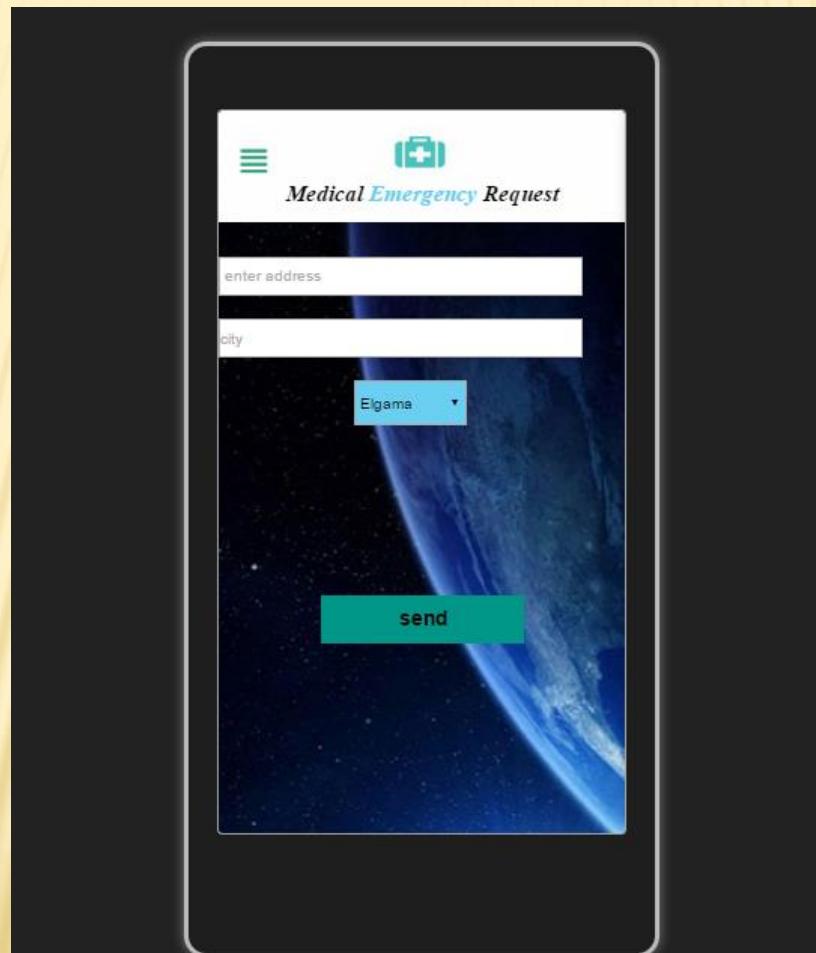
# MOBILE SCREEN(CONTINUE)



# MOBILE SCREEN(CONTINUE)



# MOBILE SCREEN(CONTINUE)



# SECURITY

- There is Security In Tables.
- There is Password is Encrypted to protect it from Trojan horse programs .
  - Example : noha password is 123 but it in database is “40bd001563085fc35165329ea”..

- Employee and manager have permissiveness
- Employee answer to massages and select doctor to go to patient

*Doctor has permissiveness to see massages tables*

- hospital has declaration to manager & employee to make sure that belong to this hospital
- before send help to hospital patient must sign in in application to protect it from send help by mistake

Chapter 4

# WEB SERVICE

```
using MySql.Data.MySqlClient;
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;

/// <summary>
/// Summary description for DBConnect
/// </summary>
public class DBConnect
{
    private MySqlConnection connection;
    private string server;
    private string database;
    private string uid;
    private string password;

    //Constructor
    public DBConnect()
    {
        Initialize();
    }

    //Initialize values
    private void Initialize()
    {
        /*server = "sql8.freemysqlhosting.net";
        database = "sql8125773";
        uid = "sql8125773";
        password = "NatCAi7uJU";*/

        server = "localhost";
        database = "mer";
        uid = "root";
        password = "1234";

        string connectionString;
        connectionString = "SERVER=" + server + ";" + "DATABASE=" +
        database + ";" + "UID=" + uid + ";" + "PASSWORD=" + password + ";";
    }

    //open connection to database
    public bool OpenConnection()
    {
        try
        {
            connection.Open();
        }
        catch (MySqlException ex)
        {
            throw ex;
        }
    }
}
```

```
try
{
    connection.Open();
    return true;
}
catch (MySqlException ex)
{
    //When handling errors, you can your application's response based
    //on the error number.
    //The two most common error numbers when connecting are as follows:
    //0: Cannot connect to server.
    //1045: Invalid user name and/or password.
    switch (ex.Number)
    {
        case 0:
            //MessageBox.Show("Cannot connect to server. Contact administrator");
            break;

        case 1045:
            //MessageBox.Show("Invalid username/password, please try again");
            break;
    }
    return false;
}

//Close connection
private bool CloseConnection()
{
    try
    {
        connection.Close();
        return true;
    }
    catch (MySqlException ex)
    {
        return false;
    }
}

//Insert statement
public string Insert(string fName, string Lname, string Uname, string pass, string age, string health, string address, string phone, string gender)
{
    if (FoundedUserName(Uname))
        return "-1";//mean founded
    string name = fName + " " + Lname;
    string query = "INSERT INTO patient(name, userName, pass, gender, address, phone, healthCase, age) VALUES('" + name + "', '" + Uname + "','" + pass + "','" + gender + "','" + address + "','" + phone + "','" + health + "')";

    //open connection
    if (this.OpenConnection() == true)
```

```
//open connection
if (this.OpenConnection() == true)
{
    //create command and assign the query and connection from the constructor
    MySqlCommand cmd = new MySqlCommand(query, connection);

    //Execute command
    cmd.ExecuteNonQuery();

    //close connection
    this.CloseConnection();
    return SelectIdByUserName(Unname)[0]; // done >> return id
}
return "0";//connection failed
}

//public string InsertHelp(string PatientID, string MedicalCenterId)
{
    string query = "INSERT INTO msg(content, reaction, id_MedicalCenter, id_patient) VALUES('help', 'no', " + MedicalCenterId + "," + PatientID + ")";

    //open connection
    if (this.OpenConnection() == true)
    {
        //create command and assign the query and connection from the constructor
        MySqlCommand cmd = new MySqlCommand(query, connection);

        //Execute command
        cmd.ExecuteNonQuery();

        //close connection
        this.CloseConnection();
        return "1"; // done >> return id
    }
    return "0";//connection failed
}

//Update statement
public string Update(string id, string name, string pass, string age, string health, string address, string phone, string gender)
{
    string query = "UPDATE patient SET name='" + name + "',pass = '" + pass + "',gender = '" + gender + "',address = '" + address + "',phone = '" + phone + "',healthCase = '" + health + "', age=" + age + " WHERE idPatient = " + id;

    //Open connection
    if (this.OpenConnection() == true)
    {
        //create mysql command
        MySqlCommand cmd = new MySqlCommand();
        //Assign the query using CommandText
        cmd.CommandText = query;
        //Assign the connection object to the command
        cmd.Connection = connection;
        //Execute command
        cmd.ExecuteNonQuery();
        //close connection
        this.CloseConnection();
        return "1"; // done >> return id
    }
    return "0";//connection failed
}
```

```
//Open connection
if (this.OpenConnection() == true)
{
    //Create mysql command
    MySqlCommand cmd = new MySqlCommand();
    //Assign the query using CommandText
    cmd.CommandText = query;
    //Assign the connection using Connection
    cmd.Connection = connection;

    //Execute query
    cmd.ExecuteNonQuery();

    //Close connection
    this.CloseConnection();
    return "1"; //done
}
return "0";//connection failed
}

//Delete statement
public void Delete()
{
}

public bool FoundedUserName(string userName)
{
    List<string> li = SelectIdByUserName(userName);
    if (li.Count != 0)
        return true;
    return false;
}
//Select statement
public List<string> SelectIdByUserName(string userName)
{
    string query = "SELECT idPatient FROM patient WHERE userName = '"+userName+"'";

    //Create a list to store the result
    List<string> list = new List<string>();

    //Open connection
    if (this.OpenConnection() == true)
    {
        //Create Command
        MySqlCommand cmd = new MySqlCommand(query, connection);
        //Create a data reader and Execute the command
        MySqlDataReader dataReader = cmd.ExecuteReader();

        //Read the data and store them in the list
        while (dataReader.Read())
```

```
//Read the data and store them in the list
while (dataReader.Read())
{
    list.Add(dataReader["idPatient"] + "");
}

//close Data Reader
dataReader.Close();

//close Connection
this.CloseConnection();

//return list to be displayed
return list;
}
else
{
    return list;
}
}

public List<string> SelectIdByUserNameAndPass(string userName, string pass)
{
    string query = "SELECT idPatient FROM patient WHERE userName = '" +userName+ "' and pass = '" +pass+ "'";

    //Create a list to store the result
    List<string> list = new List<string>();

    //Open connection
    if (this.OpenConnection() == true)
    {
        //Create Command
        MySqlCommand cmd = new MySqlCommand(query, connection);
        //Create a data reader and Execute the command
        MySqlDataReader dataReader = cmd.ExecuteReader();

        //Read the data and store them in the list
        while (dataReader.Read())
        {
            list.Add(dataReader["idPatient"] + "");
        }

        //close Data Reader
        dataReader.Close();

        //close Connection
        this.CloseConnection();

        //return list to be displayed
        return list;
    }
}
```

```
//return list to be displayed
    return list;
}
else
{
    list = null;
    return list;
}
}

public List<string>[] SelectMedicalCenters()
{
    string query = "SELECT * FROM madicalcenter";

    //Create a list to store the result
    List<string>[] list = new List<string>[5];
    list[0] = new List<string>();
    list[1] = new List<string>();
    list[2] = new List<string>();
    list[3] = new List<string>();
    list[4] = new List<string>();

    //Open connection
    if (this.OpenConnection() == true)
    {
        //Create Command
        MySqlCommand cmd = new MySqlCommand(query, connection);
        //Create a data reader and Execute the command
        MySqlDataReader dataReader = cmd.ExecuteReader();

        //Read the data and store them in the list
        while (dataReader.Read())
        {
            list[0].Add(dataReader["idMadicalCenter"] + "");
            list[1].Add(dataReader["location"] + "");
            list[2].Add(dataReader["speclized"] + "");
            list[3].Add(dataReader["phone"] + "");
            list[4].Add(dataReader["name"] + "");
        }
    }

    //close Data Reader
    dataReader.Close();

    //close Connection
    this.CloseConnection();

    //return list to be displayed
    return list;
}
else
```

```
//Open connection
if (this.OpenConnection() == true)
{
    //Create Command
    MySqlCommand cmd = new MySqlCommand(query, connection);
    //Create a data reader and Execute the command
    MySqlDataReader dataReader = cmd.ExecuteReader();

    //Read the data and store them in the list
    while (dataReader.Read())
    {
        list[0].Add(dataReader["idMadicalCenter"] + "");
        list[1].Add(dataReader["location"] + "");
        list[2].Add(dataReader["speclized"] + "");
        list[3].Add(dataReader["phone"] + "");
        list[4].Add(dataReader["name"] + "");
    }

    //close Data Reader
    dataReader.Close();

    //close Connection
    this.CloseConnection();

    //return list to be displayed
    return list;
}
else
{
    return list;
}

//Count statement
public int Count()
{
    return 0;
}

//Backup
public void Backup()
{

}

//Restore
public void Restore()
{
}
```

```
[WebMethod]
public string GetData()
{
    var output = new { a = "a", b = "b" };
    JavaScriptSerializer j2 = new JavaScriptSerializer();
    return j2.Serialize(output);
}

[WebMethod]
public string RegistrationFun(string fName, string Lname, string Uname, string pass, string age, string health, string address, string phone, string gender)
{
    DBConnect db = new DBConnect();
    string id = db.Insert(fName, Lname, Uname, pass, age, health, address, phone, gender);
    return id;
}
[WebMethod]
public string SignInfun(string username, string password)
{
    DBConnect db = new DBConnect();
    List<string> list = db.SelectIdByUserNameAndPass(username, password);
    string id = "0"; // connection failed
    id = (list != null && list.Count == 0) ? "-1" : list[0]; // -1 mean not found
    return id;
}
[WebMethod]
public string editProfileFun(string id, string name, string pass, string age, string health, string address, string phone, string gender)
{
    DBConnect db = new DBConnect();
    return db.Update(id, name, pass, age, health, address, phone, gender);
}
[WebMethod]
public string sendHelp(string PatientID, string MedicalCenterId)
{
    DBConnect db = new DBConnect();
    return db.InsertHelp(PatientID, MedicalCenterId);
}
[WebMethod]
public string nearbyhosp()
{
    DBConnect db = new DBConnect();
    List<string>[] li = db.SelectMedicalCenters();
    var output = new { id = li[0], location = li[1], specialized = li[2], phone = li[3], name = li[4] };
    JavaScriptSerializer j2 = new JavaScriptSerializer();
    return j2.Serialize(output);
}
```