TWO FACTOR AUTHENTICATION USING FACE AND SPEECH RECOGNITION FOR ONLINE DATABASE

FINAL REPORT

Team Members:

Nikhil Pinnamameni – 18BCI0053
Sumanth Dodda -- 18BCI0067
Sai Charan Muvva -- 18BCI0073
Sashank G -- 18BCI0075
Yashwanth -- 19BCE0030

VIT
VELLORE

ABSTRACT

The main purpose of this project is to develop an application interface or software application for managing real estate database helpful for real estate agencies and companies. Even though some technology is developed and available, most real estate companies are still using the traditional way to manage and store data like using documents or storing it in either files or excel sheets. Maintaining these documents and accessing it is difficult and most of the time, the documents are lost or damaged. So, it would be better if the humungous data is stored using software rather as it is easy to access and we can also use the data efficiently between clients and agents

A database administrator needs to access the management system to keep monitoring it. Usage of passwords is not a totally secure way of keeping the unwanted people out of the system. A lot of people re use the passwords that they have used on other systems or accounts. In such cases even if a password is strong enough it can't be trusted to keep the system safe from any intruder, potentially compromising the database security due a weak external software and the failure of the administrator to maintain separate passwords for different purposes. In such scenarios biometric authentication systems have been proven to be very useful.

With the increase in the average computation power of computers, running a pre trained basic AI program is a very simple and trivial task. With the computation power being available, running an AI program that can act as an authentication program is a very feasible alternative. In our proposed model we are going to use a hybrid model that uses both the speech of a person and face data of a person to identify them and give access to them. Our model will use AI to identify users even in case of minor changes in their face or speech.

We can implement this authentication on any database but we are going to implement it on Blood Donation Management System.

The principle point of building up this framework is to give blood to the general population who need blood. The quantity of people who need blood are expanding in huge number step by step. Utilizing this framework client can seek blood gather accessible in the city and he can likewise get contact number of the benefactor who has a similar blood assemble he needs. Keeping in mind the end goal to help individuals who need blood, this Online Blood Bank administration framework can be utilized viably to get the points of interest of accessible blood gatherings and client can likewise get contact number of the blood benefactors having a similar blood gathering and inside a similar city. So, if the blood amass isn't accessible in the blood donation center client can ask for the contributor to give the blood to him and spare somebody life. Utilizing this bank administration framework individuals can enrol himself or herself who need to give blood. To enlist in the framework, they need to enter their contact data like location versatile number and so forth.

TOOLS

Front-end: Html, CSS, Javascript ------ Back-end: PHP ------ Databse: Mysql

Python,Mysql,OpenCV, mysql, sys, numpy, os, etc…(some 3rd party apps)

Introduction:

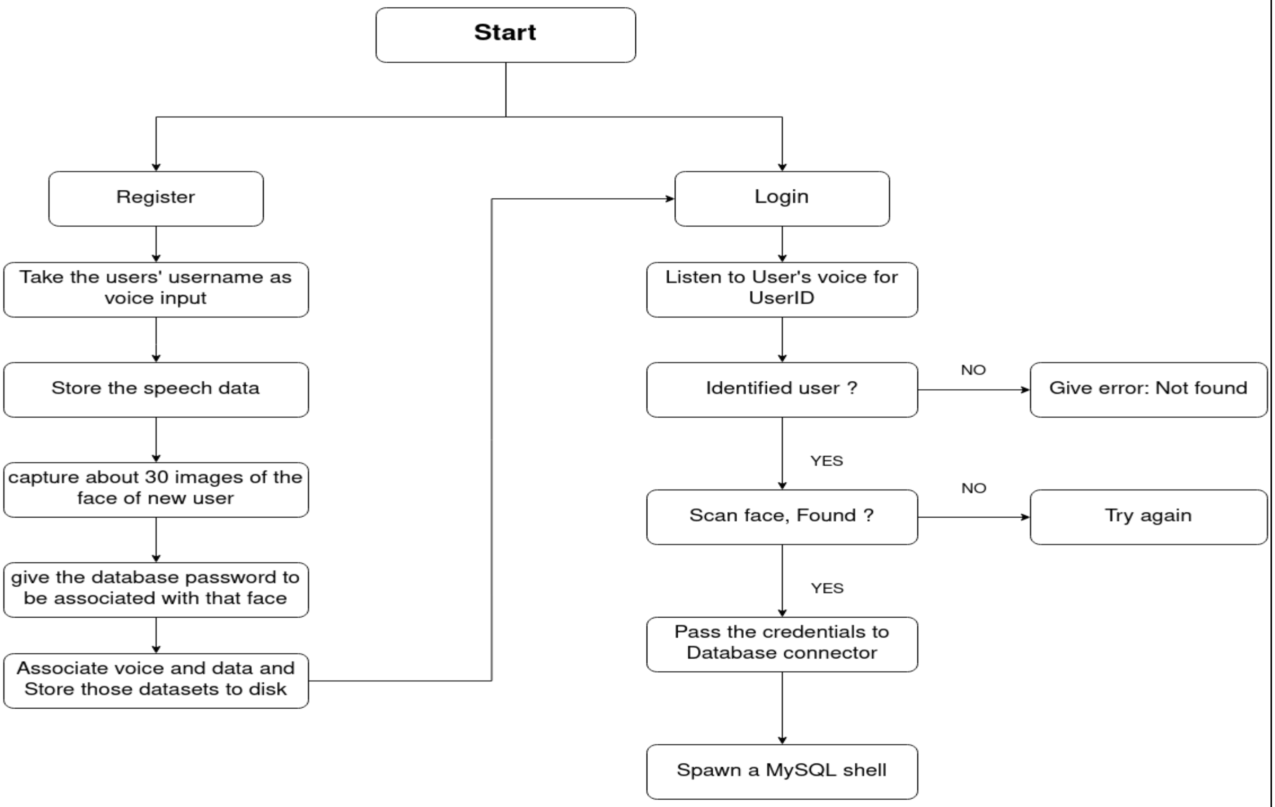
Face is a unique asset for every human being. All human beings have different faces (excluding twins that make up a very small part of the population). In real life people identify you using your face, cover up your face and no one can identify you without you providing some other form of identifiable information like your voice (or smell in case of animals that are sensitive to smell). Authentication using face detection has been around for quite some time. But they run on small computers that have been specifically designed for that purpose and they don’t use any artificial intelligence to detect minor changes in the face that occur with time and they can’t detect cosmetic changes, for example let's take our mess authentication system, in our mess during the face registration they ask you to take your spectacles off your face, And when we go back to get our face scanned and authenticate us , it can’t recognize our faces because we are wearing spectacles. These types of problems can be overcome by training the AI to detect spectacles. This increases both user convenience and system accuracy to detect and recognize faces.

Speech recognition has not been implemented heavily because human throat is very sensitive and our voice can change with even the simplest of external factors like weather (during the season transition from summer to rainy season you can suddenly wake up with a cold and not be able to authenticate yourself because of change in voice) or bad food (bad food can give you sore throat that can lead to voice change) . But in our proposed model we are using speech recognition to improve the ease of access for the end user. Our model uses a speech recognition model to recognize the user and fills their username in the required field automatically. In case if the system is unable to recognize the user then they will be given an option to fill it themselves because voice is a very sensitive feature to be used as a standalone feature in substitution for usual authentication. This way the user will be able to access the system even if they have some inconvenience with the speech recognition model.

Our model will use speech as an identification factor (like a username) and the face of an individual will be used as the password.

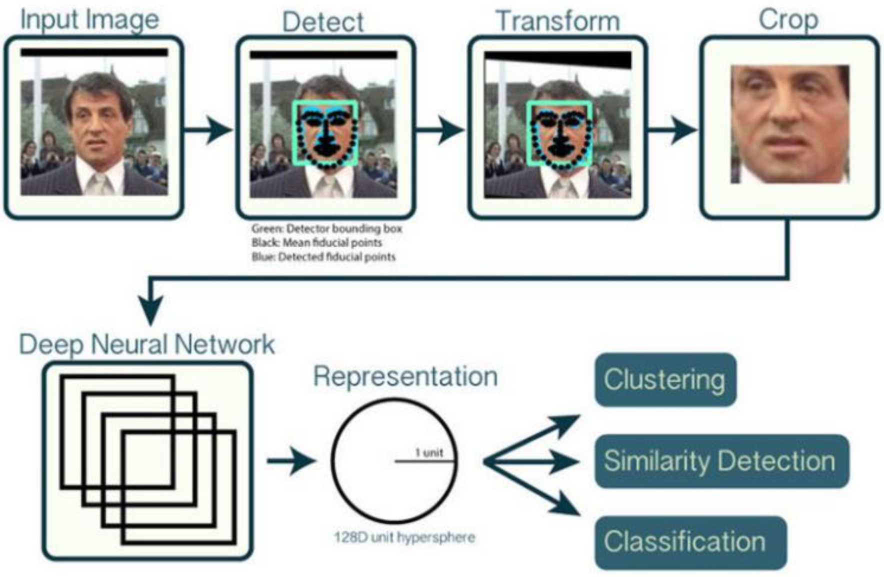
Proposed Methodology:

- **Simple Working Flow-Chart Representation:**


Start
Register
Take the users' username as voice input
Store the speech data
capture about 30 images of the face of new user
give the database password to be associated with that face
Associate voice and data and Store those datasets to disk
Login
Listen to User's voice for UserID
Identified user ?
NO
Give error: Not found
YES
Scan face, Found ?
NO
Try again
YES
Pass the credentials to Database connector
Spawn a MySQL shell

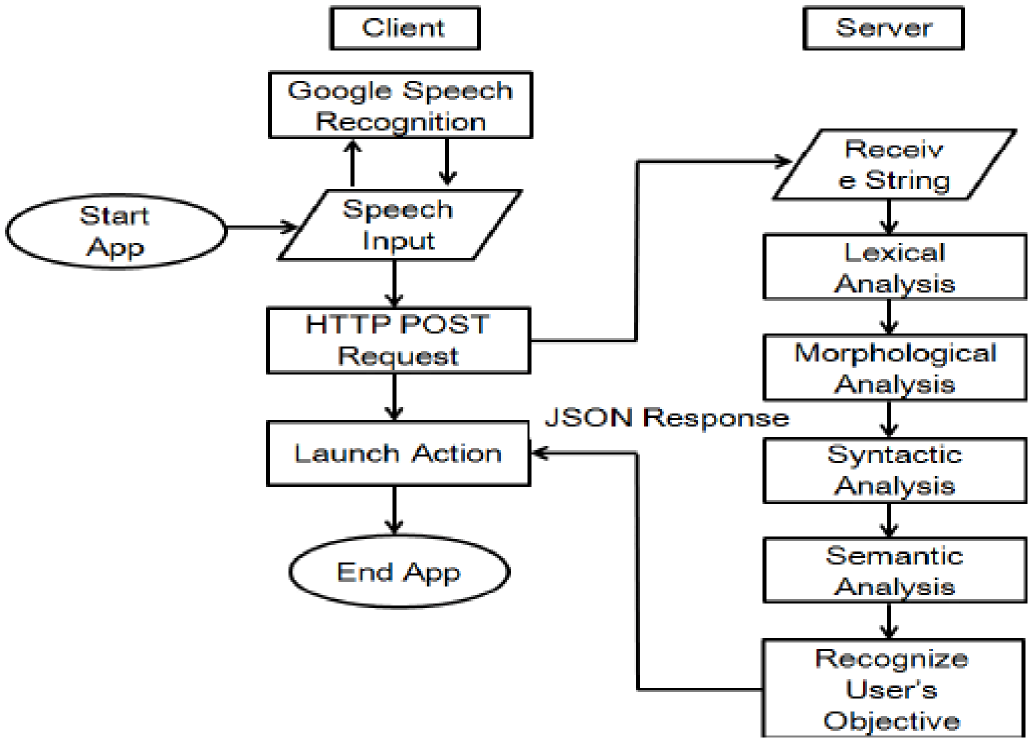
Source: Self-Made

- **Face Recognition Implemented using OpenCV Flow-Chart:**



<Source>

- **Speech Recognition API Flow-Chart:**



Source: Self-Made

Speech recognition is a process to convert speech sound to corresponding text. Speech recognition technology has been developed to a large extent in the last few years. But there exist many important research challenges e.g. speaker and language variability, environmental noise and the vocabulary size etc. The objective of this paper is to present a complete perspective on speech recognition describing various processes and summarizing various methods used in a typical speech system

Two-factor authentication, sometimes referred to as two-step verification or dual-factor authentication, is a security process in which users provide two different authentication factors to verify themselves. This process is done to better protect both the user's credentials and the resources the user can access. Two-factor authentication provides a higher level of security than authentication methods that depend on single-factor authentication, in which the user provides only one factor typically, a password or passcode. Two-factor authentication methods rely on a user providing a password, as well as a second factor, usually either a security token or a biometric factor, such as a fingerprint or facial scan. (In our case we are using face and speech recognition)

Two-factor authentication adds an additional layer of security to the authentication process by making it harder for attackers to gain access to a person's devices or online accounts because knowing the victim's password alone is not enough to pass the authentication check. Two-factor authentication has long been used to control access to sensitive systems and data, and online service providers are increasingly using 2FA to protect their users' credentials from being used by hackers who have stolen a password database or used phishing campaigns to obtain user passwords.

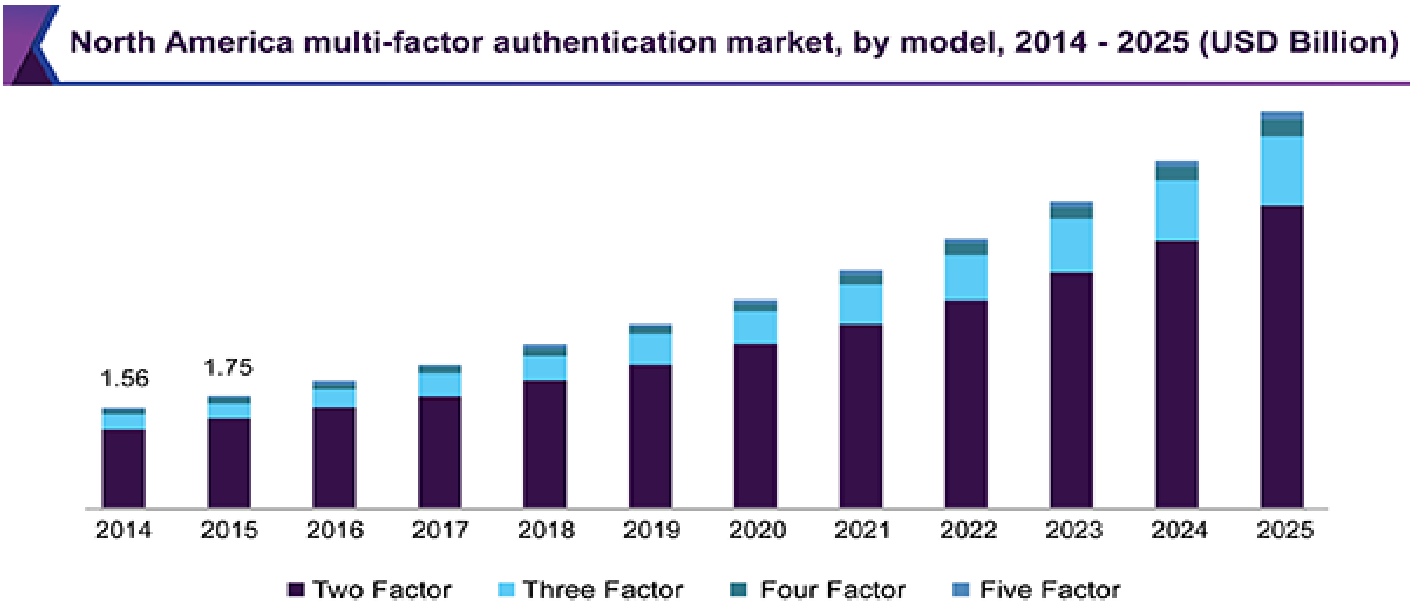
Advantages and scope:

Authentication is a major issue which has to be considered for any web-based application. The biometric system which will be implemented should ensure at most security for the application.

No biometric system can offer perfect, fool proof protection, but by combining two authentication systems will make it harder for an imposter to enter into the application. By this we are not only making our application to be more difficult to attack but we are also making our application to be less attractive for attackers to attack.

By above words one can feel that this is hassle to use but when it comes to security aspect it can be considered as a best authentication system (Not only this but combination of any two biometric systems will make the application strong in terms of security)

Scope of this two-factor authentication is very high in this digital era. We can find numerous applications using this two-factor authentication for allowing the access (But in a bit different way). Applications like Facebook, Amazon, Google, Dropbox etc..



We can see the increase in demand for multilevel authentication in the above prediction clearly Now when it comes to the advantages of this authentication system we can confidently say that these biometric systems can provide a high level of assurance when authenticating a person into the application.And the user acceptability of these biometrics which we are using is quite high so the authenticating process will be very easy for an individual. And considering the current issue of pandemic this can also be considered as a socially acceptable authentication system.

And compared to other authentication systems like password PIN etc this face and speech recognition has a technical edge because passwords and PINs can be guessed and can be stolen easily with some hacking approaches but these biometric systems cannot be hacked easily like them because in order to allow the access the physical presence of the person is must which makes this much reliable than compared to those traditional systems of authentication.

For a highly secured system the FAR of the authentication system should be low because for a highly secured system it is important to reject imposter attempts and FRR could be high because denying the access for a Genuine person is more acceptable than allowing a wrong person into the system.

Tables:

DROP TABLE IF EXISTS `admin`;

CREATE TABLE `admin` (

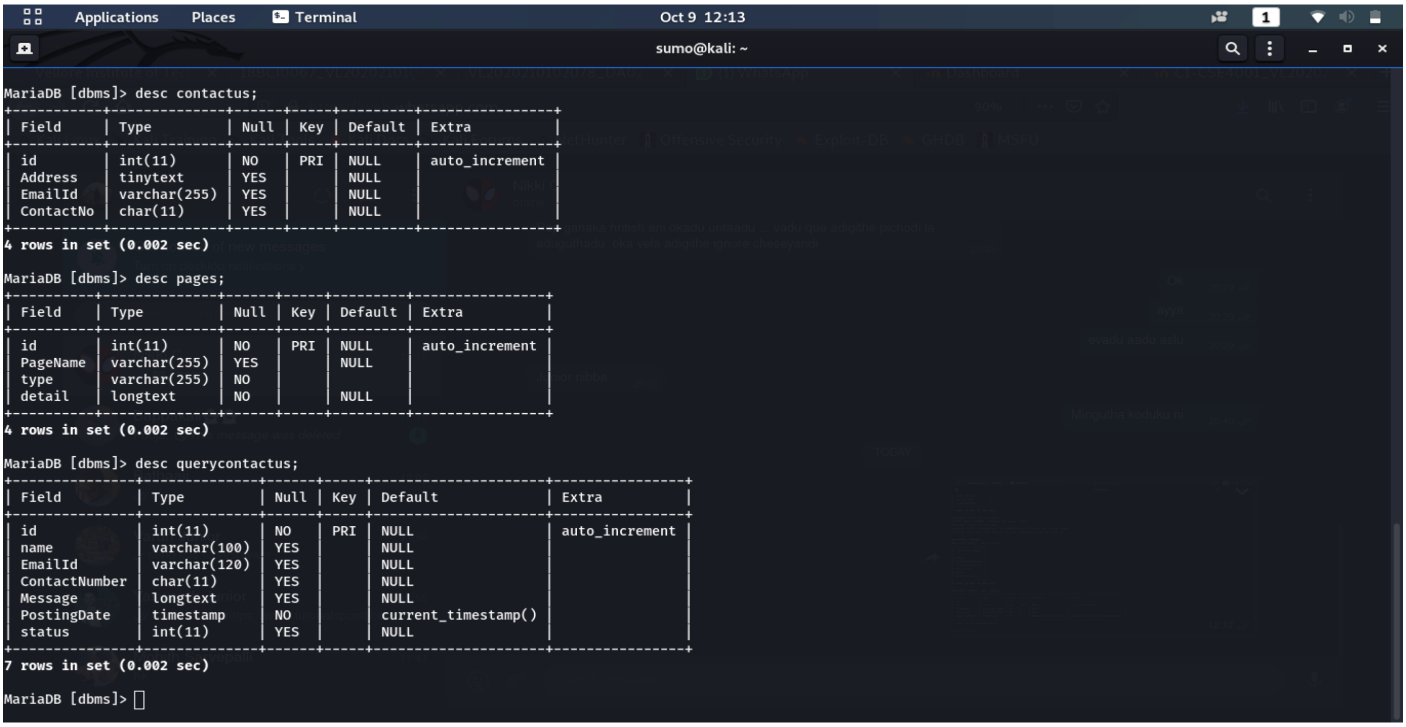
`id` int(11) NOT NULL AUTO_INCREMENT,

`UserName` varchar(100) NOT NULL,

`Password` varchar(100) NOT NULL,

`updationDate` timestamp NOT NULL DEFAULT '0000-00-00 00:00:00' ON UPDATE current_timestamp(),

PRIMARY KEY (`id`)

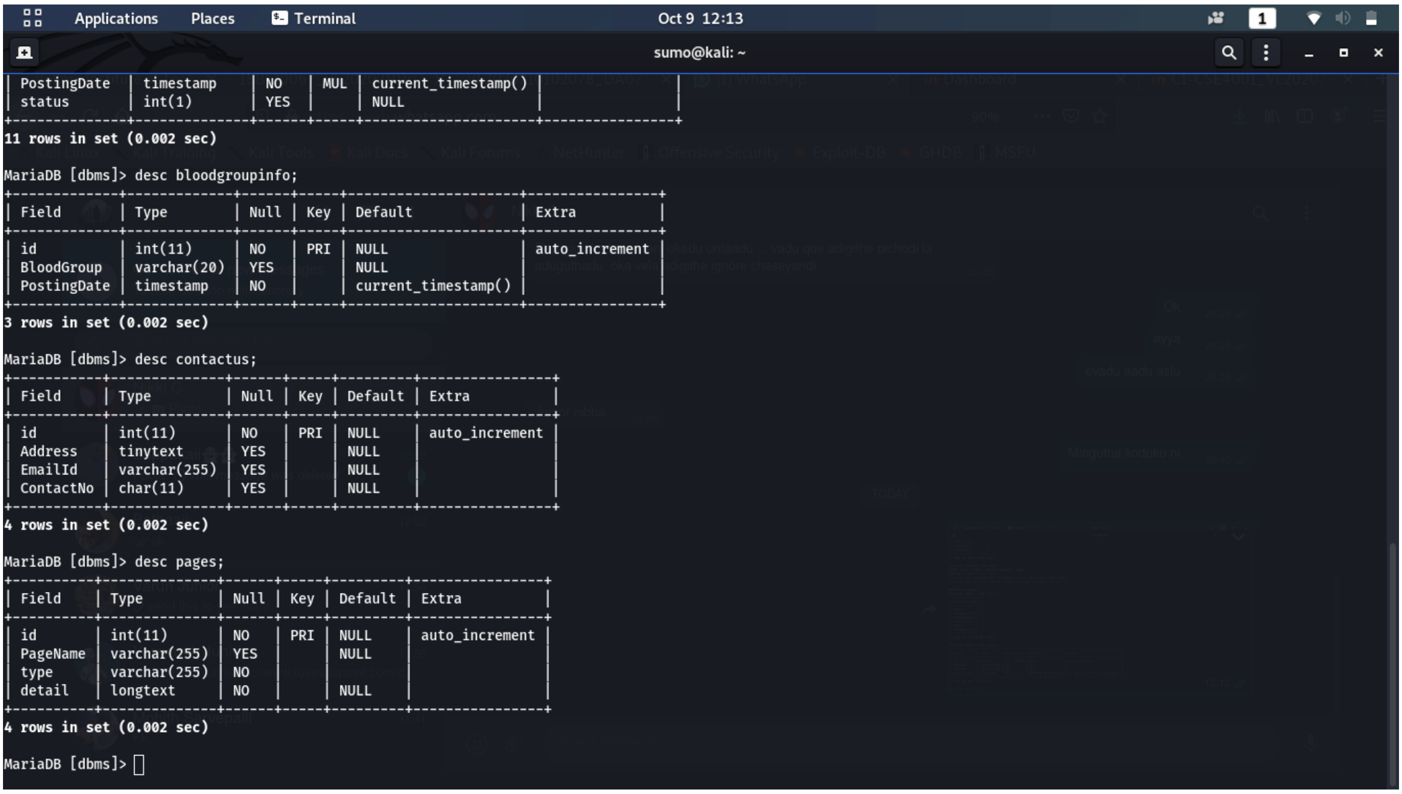


```
MariaDB [dbms]> desc contactus;
+-----------+--------------+------+-----+---------+----------------+
| Field     | Type         | Null | Key | Default | Extra          |
+-----------+--------------+------+-----+---------+----------------+
| id        | int(11)      | NO   | PRI | NULL    | auto_increment |
| Address   | tinytext     | YES  |     | NULL    |                |
| EmailId   | varchar(255) | YES  |     | NULL    |                |
| ContactNo | char(11)     | YES  |     | NULL    |                |
+-----------+--------------+------+-----+---------+----------------+
4 rows in set (0.002 sec)

MariaDB [dbms]> desc pages;
+----------+--------------+------+-----+---------+----------------+
| Field    | Type         | Null | Key | Default | Extra          |
+----------+--------------+------+-----+---------+----------------+
| id       | int(11)      | NO   | PRI | NULL    | auto_increment |
| PageName | varchar(255) | YES  |     | NULL    |                |
| type     | varchar(255) | NO   |     |         |                |
| detail   | longtext     | NO   |     | NULL    |                |
+----------+--------------+------+-----+---------+----------------+
4 rows in set (0.002 sec)

MariaDB [dbms]> desc querycontactus;
+---------------+--------------+------+-----+---------------------+----------------+
| Field         | Type         | Null | Key | Default             | Extra          |
+---------------+--------------+------+-----+---------------------+----------------+
| id            | int(11)      | NO   | PRI | NULL                | auto_increment |
| name          | varchar(100) | YES  |     | NULL                |                |
| EmailId       | varchar(120) | YES  |     | NULL                |                |
| ContactNumber | char(11)     | YES  |     | NULL                |                |
| Message       | longtext     | YES  |     | NULL                |                |
| PostingDate   | timestamp    | NO   |     | current_timestamp() |                |
| status        | int(11)      | YES  |     | NULL                |                |
+---------------+--------------+------+-----+---------------------+----------------+
7 rows in set (0.002 sec)

MariaDB [dbms]> 
```

DROP TABLE IF EXISTS `blooddonorsinfo`;

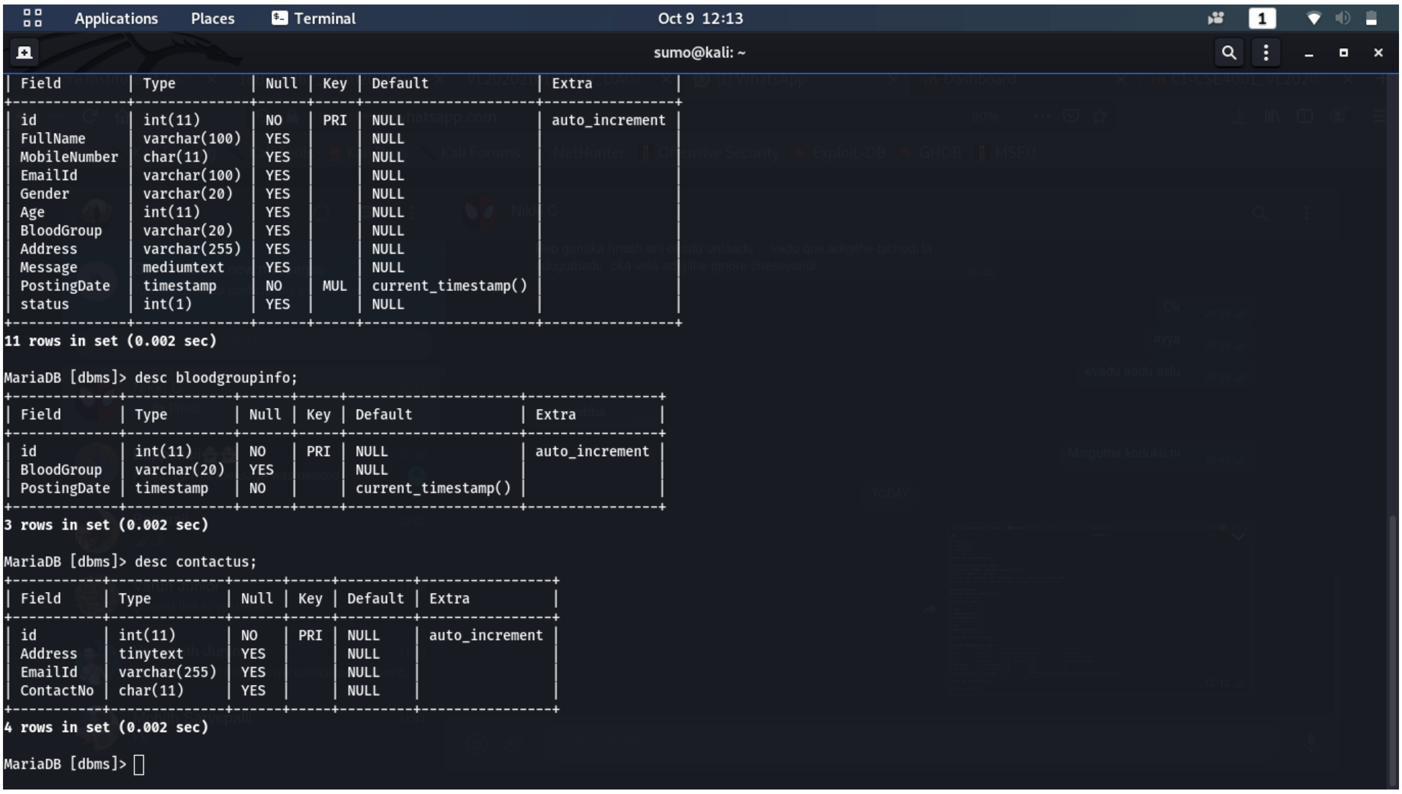
CREATE TABLE `blooddonorsinfo` (

```
 `id` int(11) NOT NULL AUTO_INCREMENT,
 `FullName` varchar(100) DEFAULT NULL,
 `MobileNumber` char(11) DEFAULT NULL,
 `EmailId` varchar(100) DEFAULT NULL,
 `Gender` varchar(20) DEFAULT NULL,
 `Age` int(11) DEFAULT NULL,
 `BloodGroup` varchar(20) DEFAULT NULL,
 `Address` varchar(255) DEFAULT NULL,
 `Message` mediumtext DEFAULT NULL,
 `PostingDate` timestamp NOT NULL DEFAULT current_timestamp(),
 `status` int(1) DEFAULT NULL,
 PRIMARY KEY (`id`),
 KEY `idx_blooddonorsinfo_PostingDate` (`PostingDate`)
```



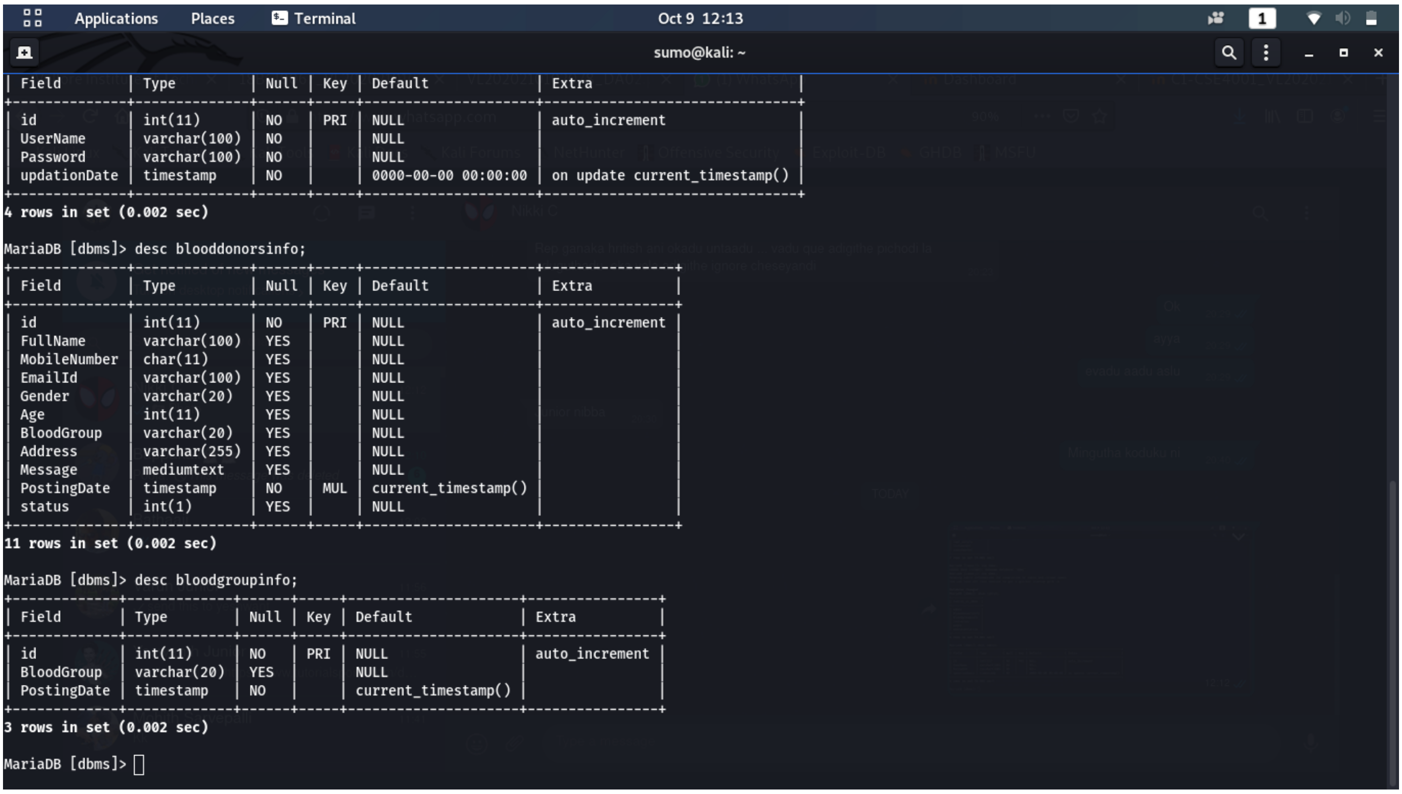
```
| PostingDate | timestamp    | NO   | MUL | current_timestamp() |                |
| status      | int(1)       | YES  |     | NULL                |                |
+-------------+--------------+------+-----+---------------------+----------------+
11 rows in set (0.002 sec)

MariaDB [dbms]> desc bloodgroupinfo;
+-------------+-------------+------+-----+---------------------+----------------+
| Field       | Type        | Null | Key | Default             | Extra          |
+-------------+-------------+------+-----+---------------------+----------------+
| id          | int(11)     | NO   | PRI | NULL                | auto_increment |
| BloodGroup  | varchar(20) | YES  |     | NULL                |                |
| PostingDate | timestamp   | NO   |     | current_timestamp() |                |
+-------------+-------------+------+-----+---------------------+----------------+
3 rows in set (0.002 sec)

MariaDB [dbms]> desc contactus;
+-----------+--------------+------+-----+---------+----------------+
| Field     | Type         | Null | Key | Default | Extra          |
+-----------+--------------+------+-----+---------+----------------+
| id        | int(11)      | NO   | PRI | NULL    | auto_increment |
| Address   | tinytext     | YES  |     | NULL    |                |
| EmailId   | varchar(255) | YES  |     | NULL    |                |
| ContactNo | char(11)     | YES  |     | NULL    |                |
+-----------+--------------+------+-----+---------+----------------+
4 rows in set (0.002 sec)

MariaDB [dbms]> desc pages;
+----------+--------------+------+-----+---------+----------------+
| Field    | Type         | Null | Key | Default | Extra          |
+----------+--------------+------+-----+---------+----------------+
| id       | int(11)      | NO   | PRI | NULL    | auto_increment |
| PageName | varchar(255) | YES  |     | NULL    |                |
| type     | varchar(255) | NO   |     |         |                |
| detail   | longtext     | NO   |     | NULL    |                |
+----------+--------------+------+-----+---------+----------------+
4 rows in set (0.002 sec)

MariaDB [dbms]> 
```

```
DROP TABLE IF EXISTS `bloodgroupinfo`;
CREATE TABLE `bloodgroupinfo` (
 `id` int(11) NOT NULL AUTO_INCREMENT,
 `BloodGroup` varchar(20) DEFAULT NULL,
 `PostingDate` timestamp NOT NULL DEFAULT current_timestamp(),
 PRIMARY KEY (`id`)
) ENGINE=InnoDB AUTO_INCREMENT=7 DEFAULT CHARSET=latin1;
```



```
| Field        | Type         | Null | Key | Default             | Extra          |
+--------------+--------------+------+-----+---------------------+----------------+
| id           | int(11)      | NO   | PRI | NULL                | auto_increment |
| FullName     | varchar(100) | YES  |     | NULL                |                |
| MobileNumber | char(11)     | YES  |     | NULL                |                |
| EmailId      | varchar(100) | YES  |     | NULL                |                |
| Gender       | varchar(20)  | YES  |     | NULL                |                |
| Age          | int(11)      | YES  |     | NULL                |                |
| BloodGroup   | varchar(20)  | YES  |     | NULL                |                |
| Address      | varchar(255) | YES  |     | NULL                |                |
| Message      | mediumtext   | YES  |     | NULL                |                |
| PostingDate  | timestamp    | NO   | MUL | current_timestamp() |                |
| status       | int(1)       | YES  |     | NULL                |                |
+--------------+--------------+------+-----+---------------------+----------------+
11 rows in set (0.002 sec)

MariaDB [dbms]> desc bloodgroupinfo;
+-------------+-------------+------+-----+---------------------+----------------+
| Field       | Type        | Null | Key | Default             | Extra          |
+-------------+-------------+------+-----+---------------------+----------------+
| id          | int(11)     | NO   | PRI | NULL                | auto_increment |
| BloodGroup  | varchar(20) | YES  |     | NULL                |                |
| PostingDate | timestamp   | NO   |     | current_timestamp() |                |
+-------------+-------------+------+-----+---------------------+----------------+
3 rows in set (0.002 sec)

MariaDB [dbms]> desc contactus;
+-----------+--------------+------+-----+---------+----------------+
| Field     | Type         | Null | Key | Default | Extra          |
+-----------+--------------+------+-----+---------+----------------+
| id        | int(11)      | NO   | PRI | NULL    | auto_increment |
| Address   | tinytext     | YES  |     | NULL    |                |
| EmailId   | varchar(255) | YES  |     | NULL    |                |
| ContactNo | char(11)     | YES  |     | NULL    |                |
+-----------+--------------+------+-----+---------+----------------+
4 rows in set (0.002 sec)

MariaDB [dbms]> 
```

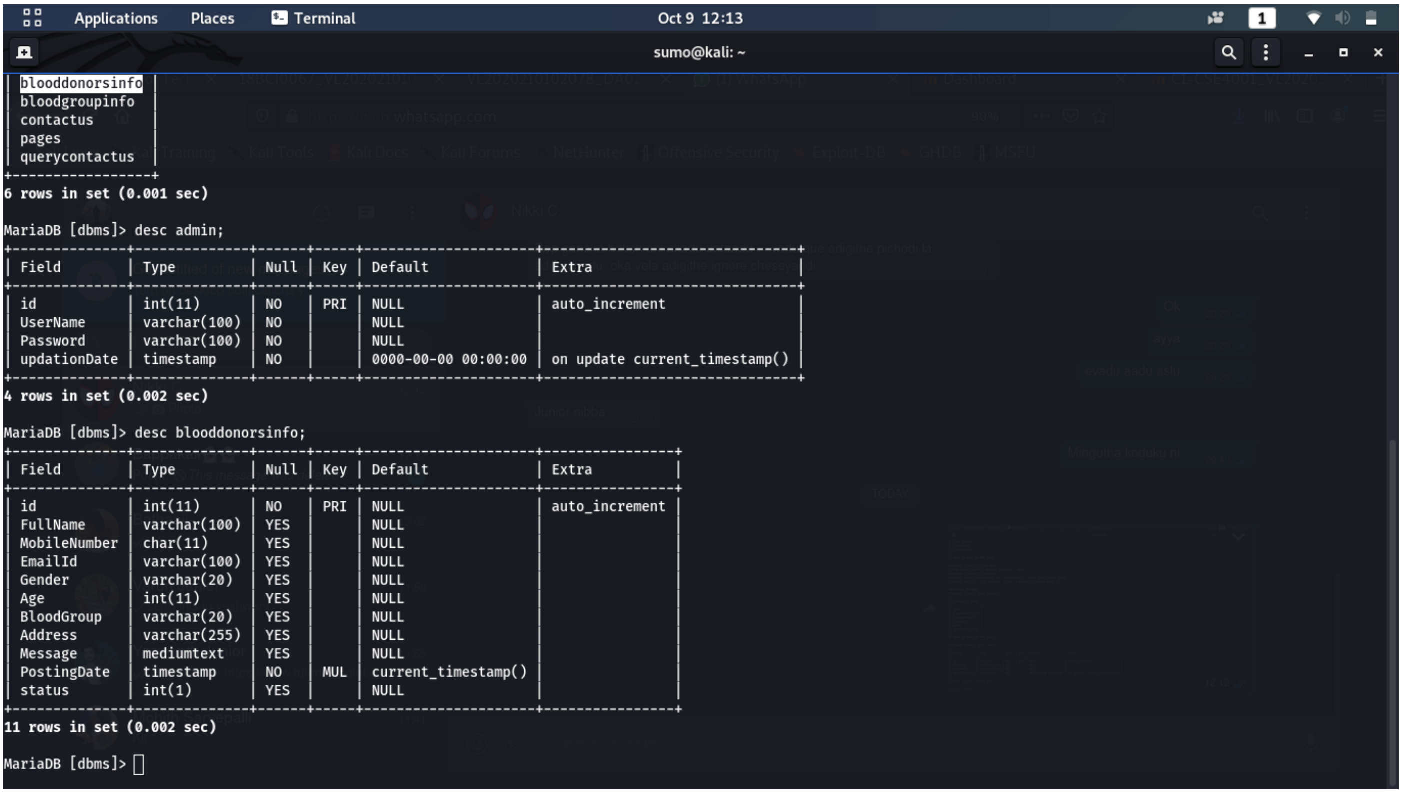
```
DROP TABLE IF EXISTS `contactus`;
CREATE TABLE `contactus` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `Address` tinytext DEFAULT NULL,
  `EmailId` varchar(255) DEFAULT NULL,
  `ContactNo` char(11) DEFAULT NULL,
  PRIMARY KEY (`id`)
) ENGINE=InnoDB AUTO_INCREMENT=2 DEFAULT CHARSET=latin1;
```



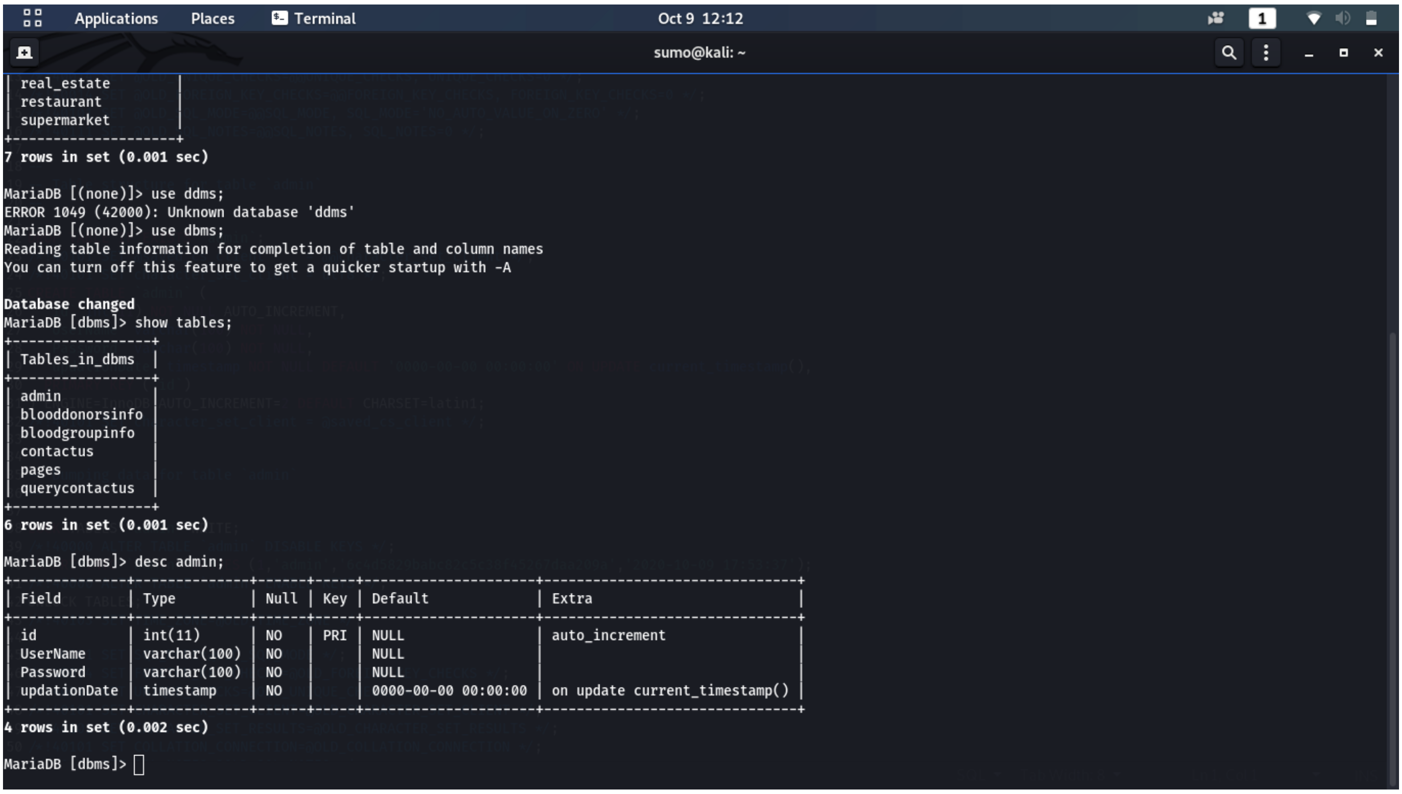
```
| Field        | Type         | Null | Key | Default             | Extra                       |
+--------------+--------------+------+-----+---------------------+-----------------------------+
| id           | int(11)      | NO   | PRI | NULL                | auto_increment              |
| UserName     | varchar(100) | NO   |     | NULL                |                             |
| Password     | varchar(100) | NO   |     | NULL                |                             |
| updationDate | timestamp    | NO   |     | 0000-00-00 00:00:00 | on update current_timestamp() |
+--------------+--------------+------+-----+---------------------+-----------------------------+
4 rows in set (0.002 sec)

MariaDB [dbms]> desc blooddonorsinfo;
+--------------+--------------+------+-----+---------------------+----------------+
| Field        | Type         | Null | Key | Default             | Extra          |
+--------------+--------------+------+-----+---------------------+----------------+
| id           | int(11)      | NO   | PRI | NULL                | auto_increment |
| FullName     | varchar(100) | YES  |     | NULL                |                |
| MobileNumber | char(11)     | YES  |     | NULL                |                |
| EmailId      | varchar(100) | YES  |     | NULL                |                |
| Gender       | varchar(20)  | YES  |     | NULL                |                |
| Age          | int(11)      | YES  |     | NULL                |                |
| BloodGroup   | varchar(20)  | YES  |     | NULL                |                |
| Address      | varchar(255) | YES  |     | NULL                |                |
| Message      | mediumtext   | YES  |     | NULL                |                |
| PostingDate  | timestamp    | NO   | MUL | current_timestamp() |                |
| status       | int(1)       | YES  |     | NULL                |                |
+--------------+--------------+------+-----+---------------------+----------------+
11 rows in set (0.002 sec)

MariaDB [dbms]> desc bloodgroupinfo;
+-------------+-------------+------+-----+---------------------+----------------+
| Field       | Type        | Null | Key | Default             | Extra          |
+-------------+-------------+------+-----+---------------------+----------------+
| id          | int(11)     | NO   | PRI | NULL                | auto_increment |
| BloodGroup  | varchar(20) | YES  |     | NULL                |                |
| PostingDate | timestamp   | NO   |     | current_timestamp() |                |
+-------------+-------------+------+-----+---------------------+----------------+
3 rows in set (0.002 sec)

MariaDB [dbms]> 
```

```
DROP TABLE IF EXISTS `pages`;

CREATE TABLE `pages` (

  `id` int(11) NOT NULL AUTO_INCREMENT,

  `PageName` varchar(255) DEFAULT NULL,

  `type` varchar(255) NOT NULL DEFAULT '',

  `detail` longtext NOT NULL,

  PRIMARY KEY (`id`)

) ENGINE=MyISAM AUTO_INCREMENT=22 DEFAULT CHARSET=latin1;
```

```
DROP TABLE IF EXISTS `querycontactus`;

CREATE TABLE `querycontactus` (

  `id` int(11) NOT NULL AUTO_INCREMENT,

  `name` varchar(100) DEFAULT NULL,
  `EmailId` varchar(120) DEFAULT NULL,
  `ContactNumber` char(11) DEFAULT NULL,
  `Message` longtext DEFAULT NULL,
  `PostingDate` timestamp NOT NULL DEFAULT current_timestamp(),
  `status` int(11) DEFAULT NULL,
  PRIMARY KEY (`id`)
) ENGINE=InnoDB AUTO_INCREMENT=10 DEFAULT CHARSET=latin1;
```



```
| blooddonorsinfo |
| bloodgroupinfo  |
| contactus       |
| pages           |
| querycontactus  |
+-----------------+
6 rows in set (0.001 sec)

MariaDB [dbms]> desc admin;
+--------------+--------------+------+-----+---------------------+-----------------------------+
| Field        | Type         | Null | Key | Default             | Extra                       |
+--------------+--------------+------+-----+---------------------+-----------------------------+
| id           | int(11)      | NO   | PRI | NULL                | auto_increment              |
| UserName     | varchar(100) | NO   |     | NULL                |                             |
| Password     | varchar(100) | NO   |     | NULL                |                             |
| updationDate | timestamp    | NO   |     | 0000-00-00 00:00:00 | on update current_timestamp() |
+--------------+--------------+------+-----+---------------------+-----------------------------+
4 rows in set (0.002 sec)

MariaDB [dbms]> desc blooddonorsinfo;
+--------------+--------------+------+-----+---------------------+----------------+
| Field        | Type         | Null | Key | Default             | Extra          |
+--------------+--------------+------+-----+---------------------+----------------+
| id           | int(11)      | NO   | PRI | NULL                | auto_increment |
| FullName     | varchar(100) | YES  |     | NULL                |                |
| MobileNumber | char(11)     | YES  |     | NULL                |                |
| EmailId      | varchar(100) | YES  |     | NULL                |                |
| Gender       | varchar(20)  | YES  |     | NULL                |                |
| Age          | int(11)      | YES  |     | NULL                |                |
| BloodGroup   | varchar(20)  | YES  |     | NULL                |                |
| Address      | varchar(255) | YES  |     | NULL                |                |
| Message      | mediumtext   | YES  |     | NULL                |                |
| PostingDate  | timestamp    | NO   | MUL | current_timestamp() |                |
| status       | int(1)       | YES  |     | NULL                |                |
+--------------+--------------+------+-----+---------------------+----------------+
11 rows in set (0.002 sec)

MariaDB [dbms]> 
```

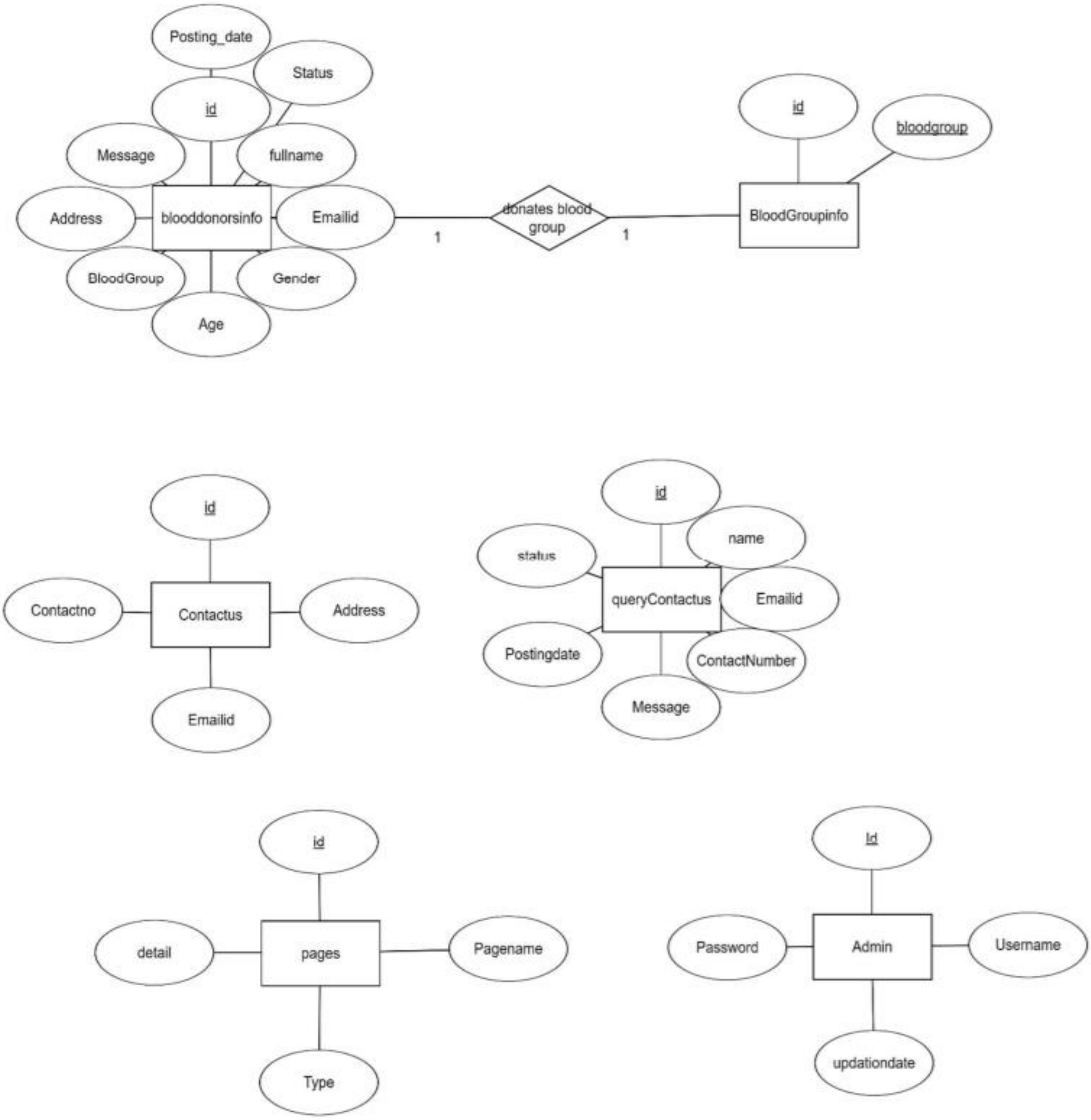


```
| real_estate        |
| restaurant         |
| supermarket        |
+--------------------+
7 rows in set (0.001 sec)

MariaDB [(none)]> use ddms;
ERROR 1049 (42000): Unknown database 'ddms'
MariaDB [(none)]> use dbms;
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Database changed
MariaDB [dbms]> show tables;
+-----------------+
| Tables_in_dbms  |
+-----------------+
| admin           |
| blooddonorsinfo |
| bloodgroupinfo  |
| contactus       |
| pages           |
| querycontactus  |
+-----------------+
6 rows in set (0.001 sec)

MariaDB [dbms]> desc admin;
+--------------+--------------+------+-----+---------------------+-------------------------------+
| Field        | Type         | Null | Key | Default             | Extra                         |
+--------------+--------------+------+-----+---------------------+-------------------------------+
| id           | int(11)      | NO   | PRI | NULL                | auto_increment                |
| UserName     | varchar(100) | NO   |     | NULL                |                               |
| Password     | varchar(100) | NO   |     | NULL                |                               |
| updationDate | timestamp    | NO   |     | 0000-00-00 00:00:00 | on update current_timestamp() |
+--------------+--------------+------+-----+---------------------+-------------------------------+
4 rows in set (0.002 sec)

MariaDB [dbms]> 
```

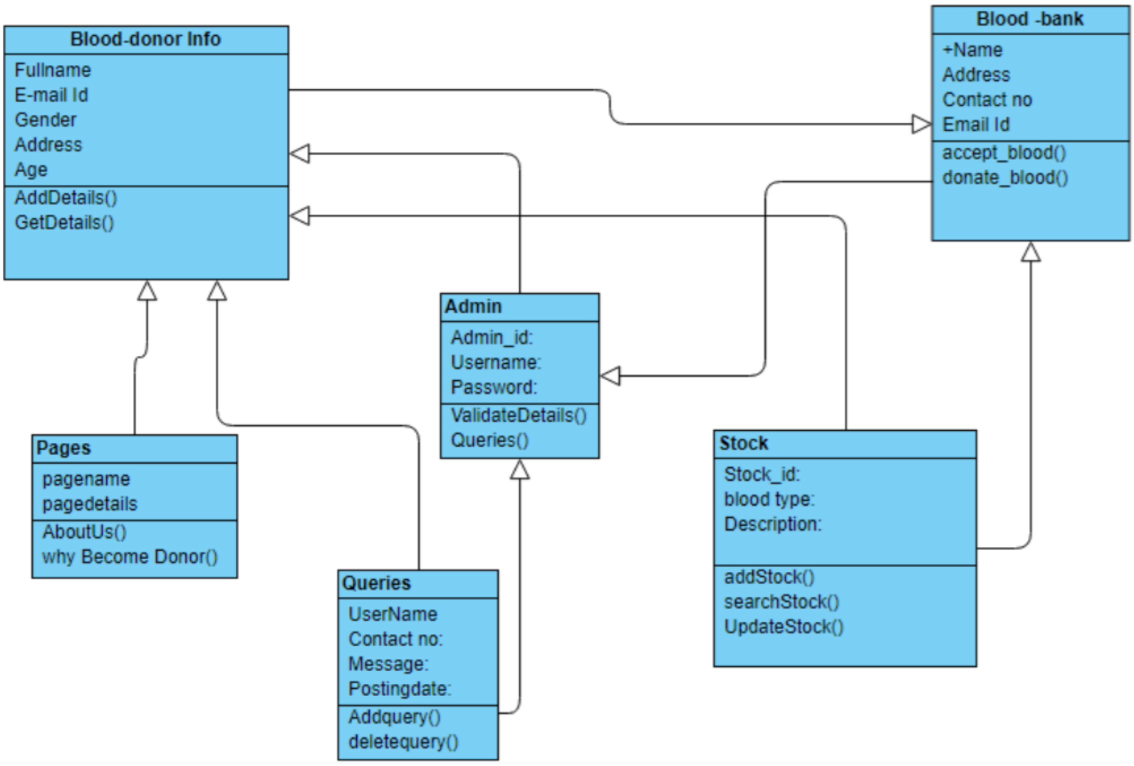
Normalization:

The tables are normalized

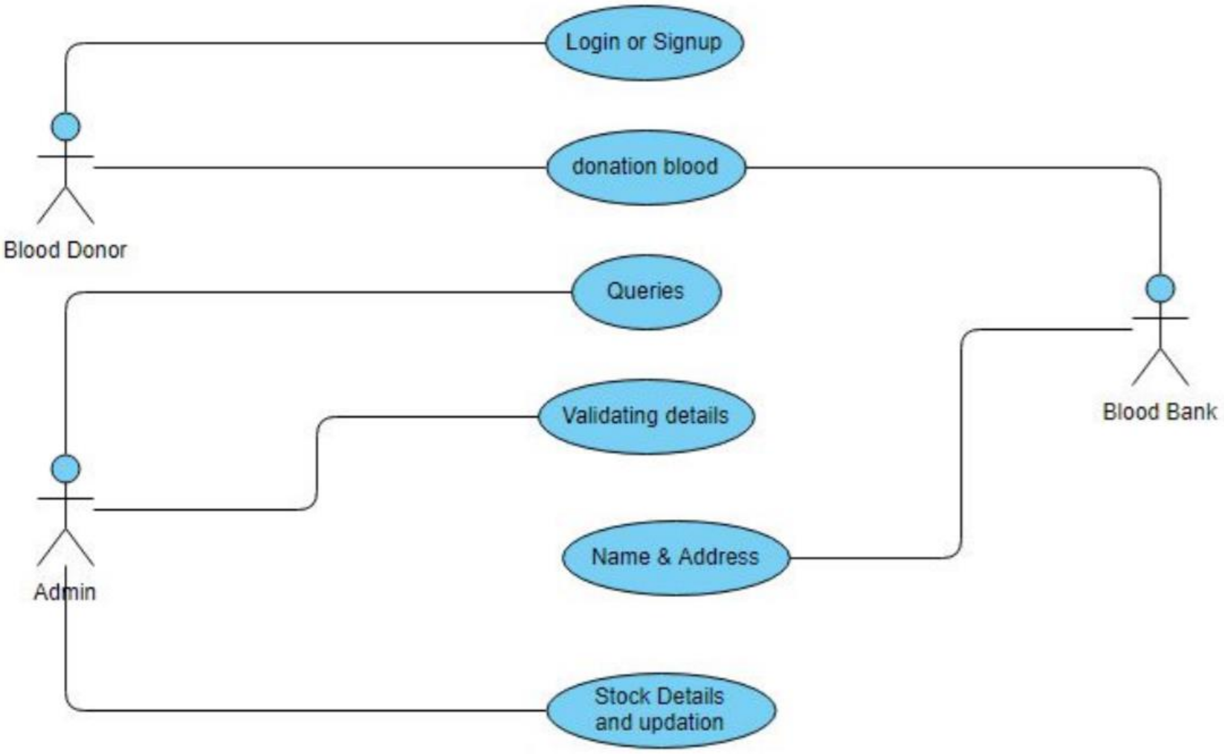
ER Diagram:


Posting_date
Status
id
Message
fullname
Address
blooddonorsinfo
Emailid
BloodGroup
Gender
Age
donates blood group
1
1
BloodGroupinfo
id
bloodgroup
id
Contactno
Contactus
Address
Emailid
id
status
name
queryContactus
Emailid
Postingdate
ContactNumber
Message
id
detail
pages
Pagename
Type
Id
Password
Admin
Username
updationdate

Class Diagram:

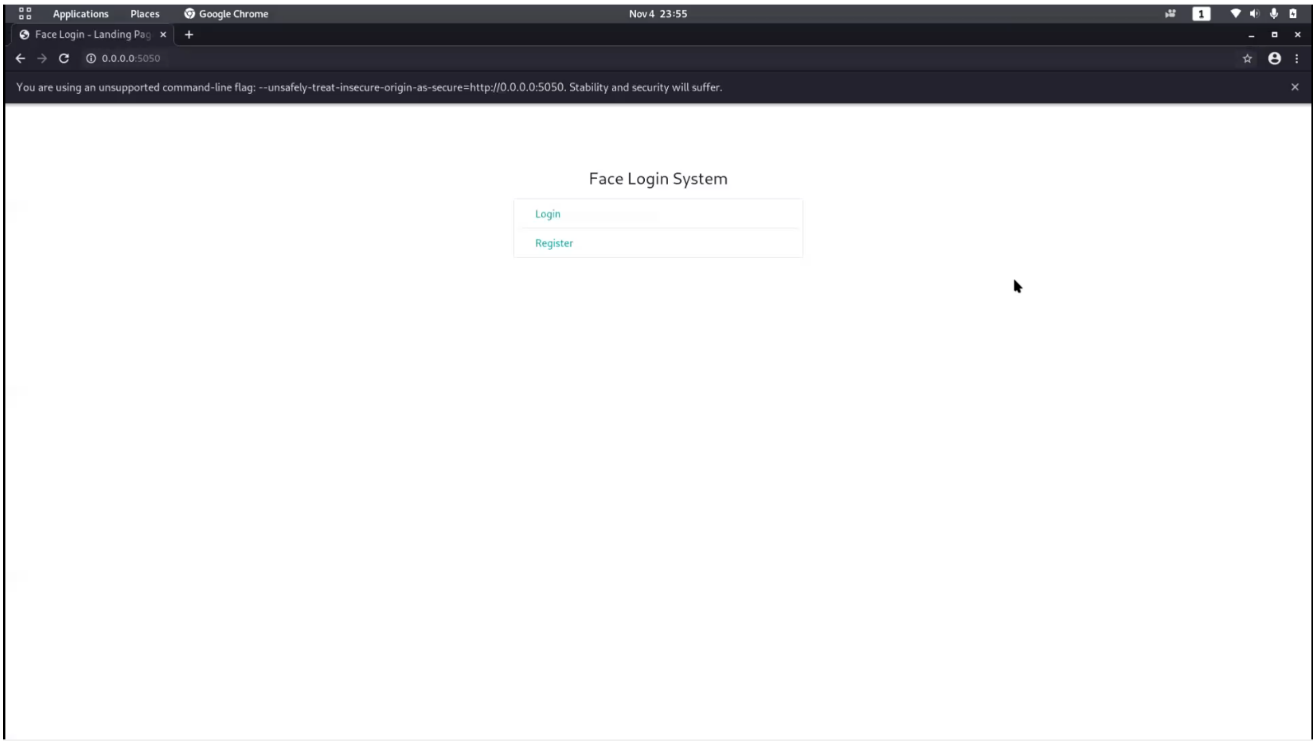


USE CASE Diagram :



Execution:

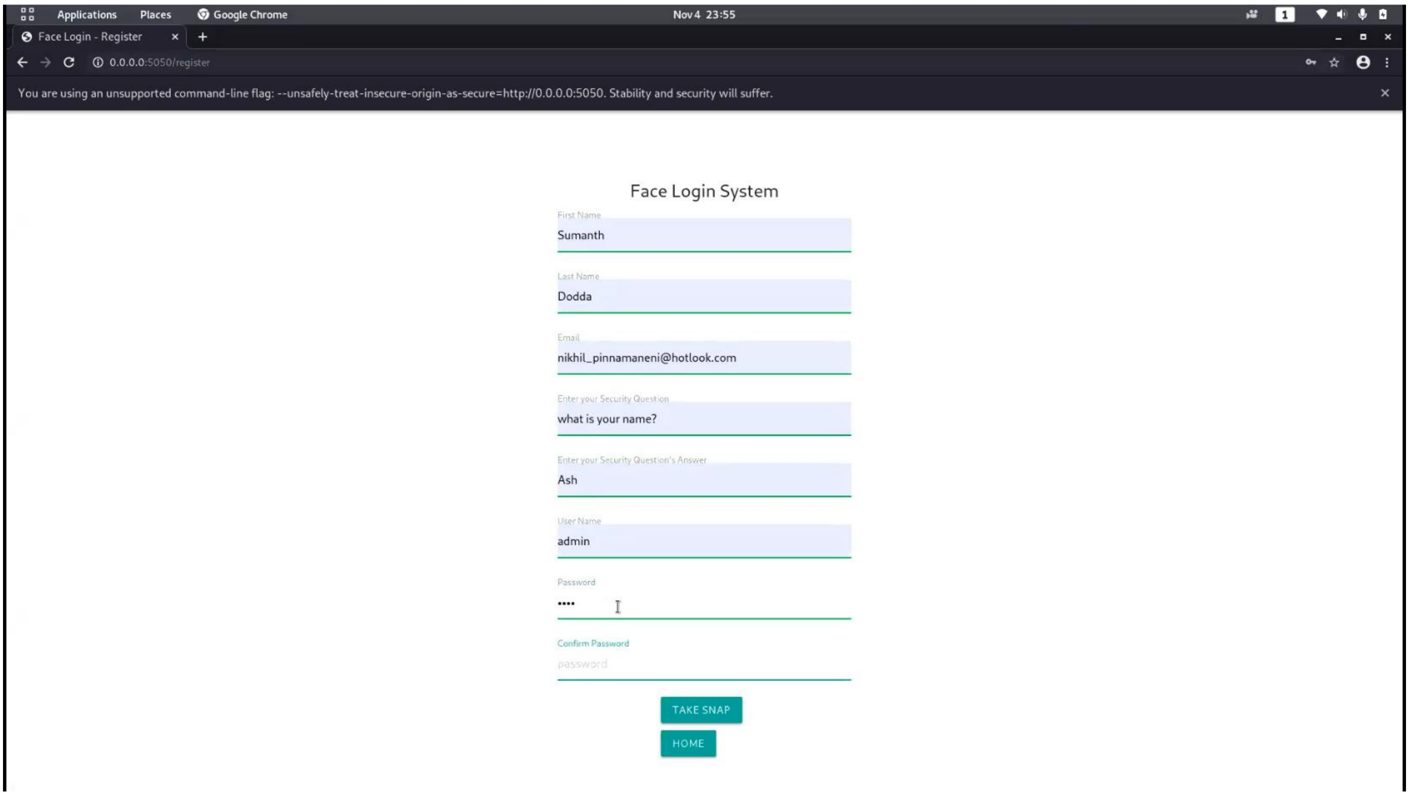
- **Homepage for Login**



You are using an unsupported command-line flag: --unsafely-treat-insecure-origin-as-secure=http://0.0.0.0:5050. Stability and security will suffer.

Face Login System

- **Enrolling Face & Security Question**



You are using an unsupported command-line flag: --unsafely-treat-insecure-origin-as-secure=http://0.0.0.0:5050. Stability and security will suffer.

Face Login System

First Name: Sumanth

Last Name: Dodda

Email: nikhil_pinnamaneni@hotlook.com

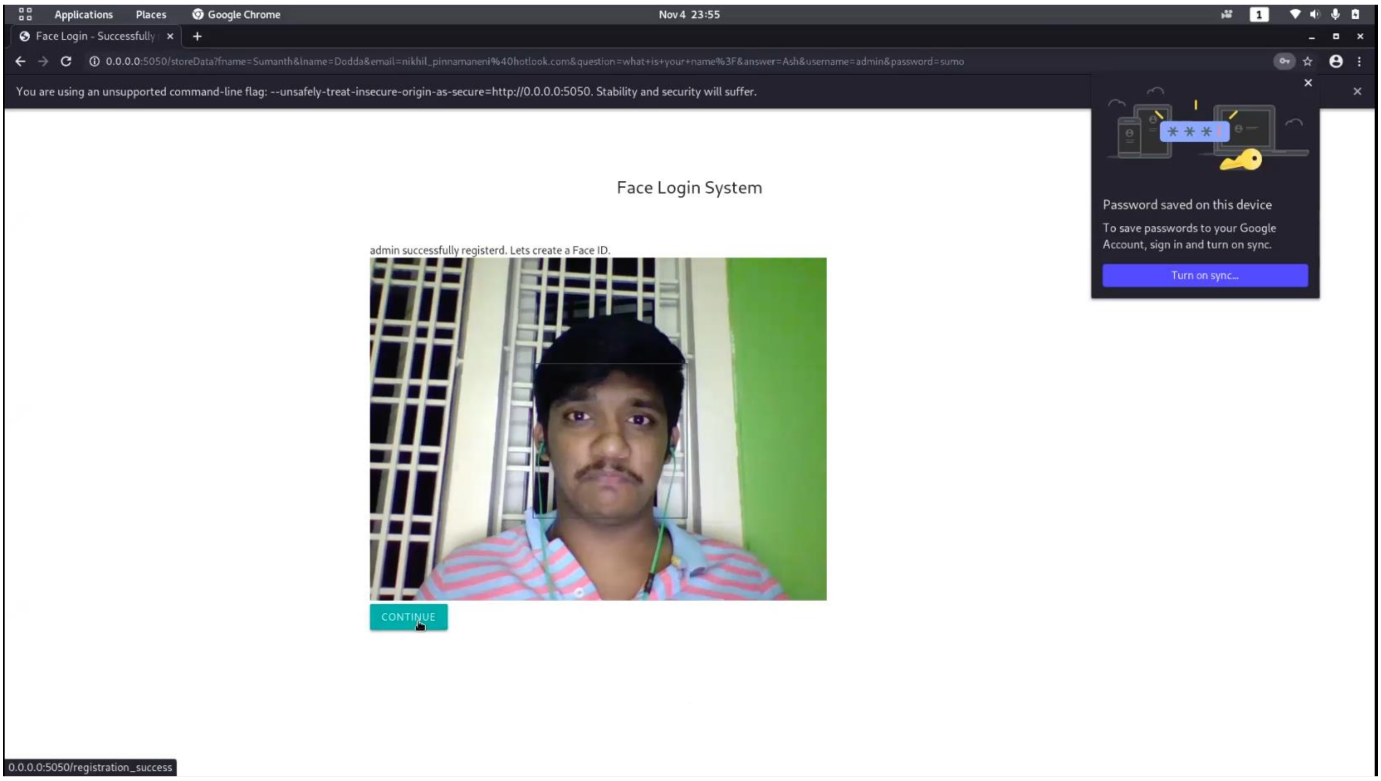
Enter your Security Question: what is your name?

Enter your Security Question's Answer: Ash

User Name: admin

Password: ••••

Confirm Password: password



0.0.0.0:5050/storeData?fname=Sumanth&lname=Dodda&email=nikhil_pinnamaneni%40hotlook.com&question=what+is+your+name%3F&answer=Ash&username=admin&password=sumo

You are using an unsupported command-line flag: --unsafely-treat-insecure-origin-as-secure=http://0.0.0.0:5050. Stability and security will suffer.

Face Login System

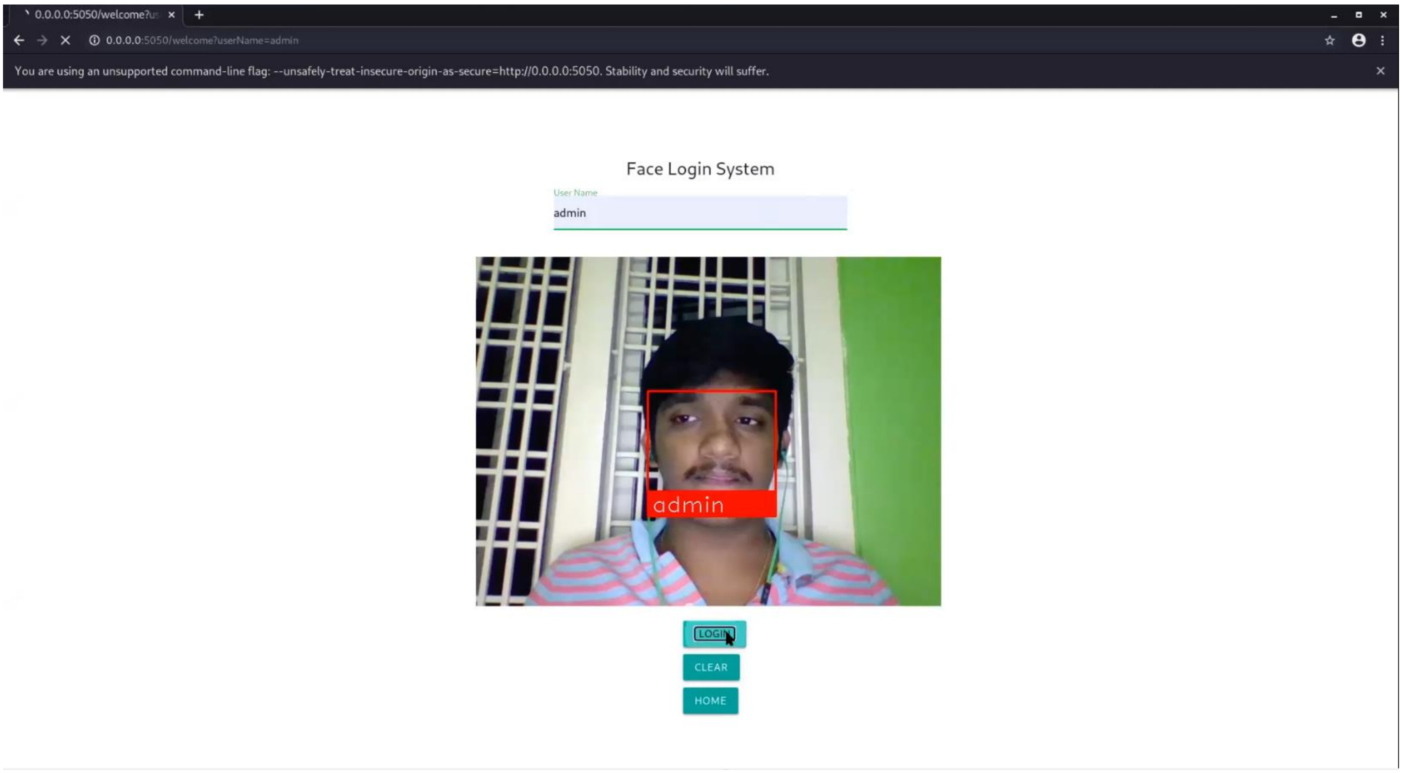
admin successfully registerd. Lets create a Face ID.

Password saved on this device

To save passwords to your Google Account, sign in and turn on sync.

0.0.0.0:5050/registration_success

- **Face Recognition Login (1st Level)**



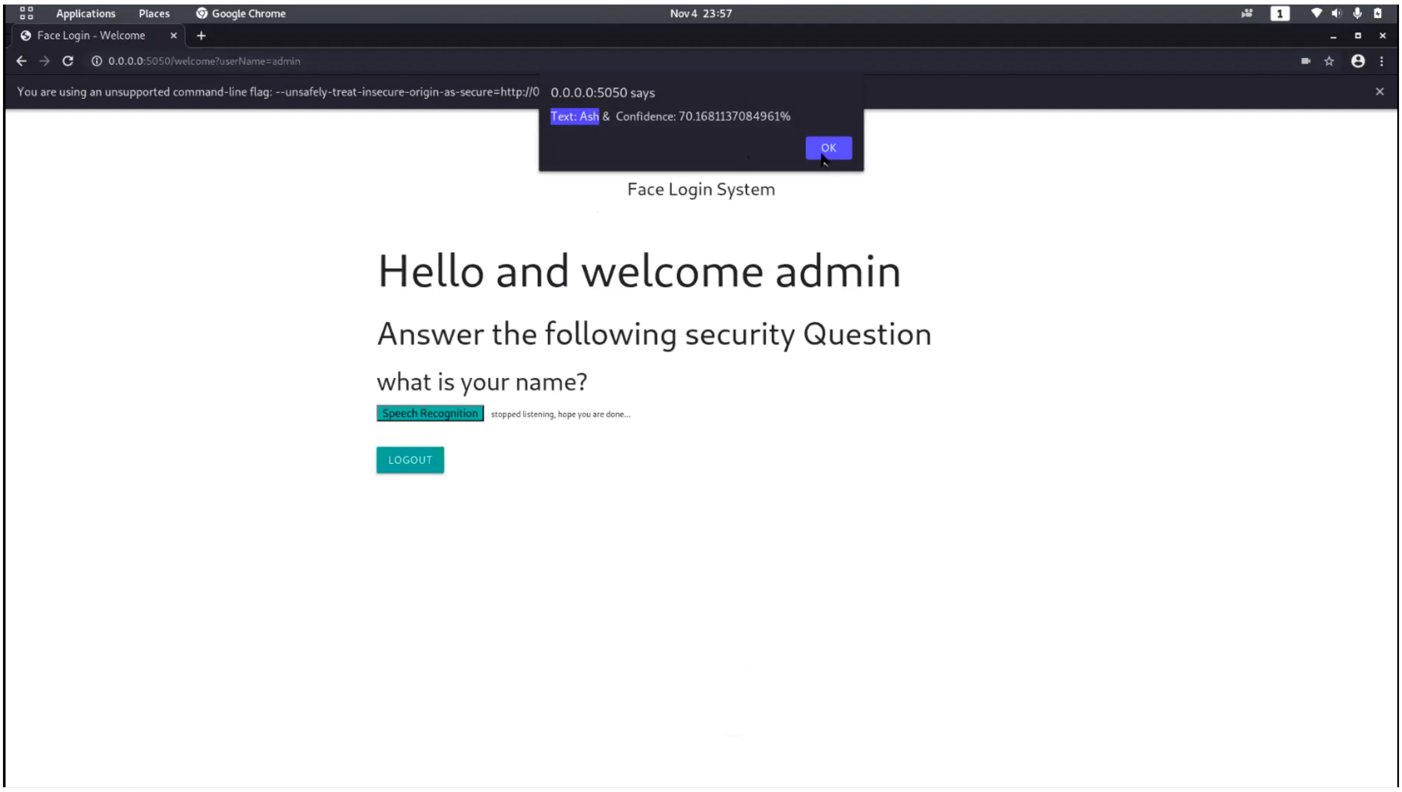
0.0.0.0:5050/welcome?userName=admin

You are using an unsupported command-line flag: --unsafely-treat-insecure-origin-as-secure=http://0.0.0.0:5050. Stability and security will suffer.

Face Login System

User Name: admin

- **Speech Recognition Login (2nd Level)**



You are using an unsupported command-line flag: --unsafely-treat-insecure-origin-as-secure=http://0

0.0.0.0:5050 says

Text: Ash & Confidence: 70.16811370849 61%

Face Login System

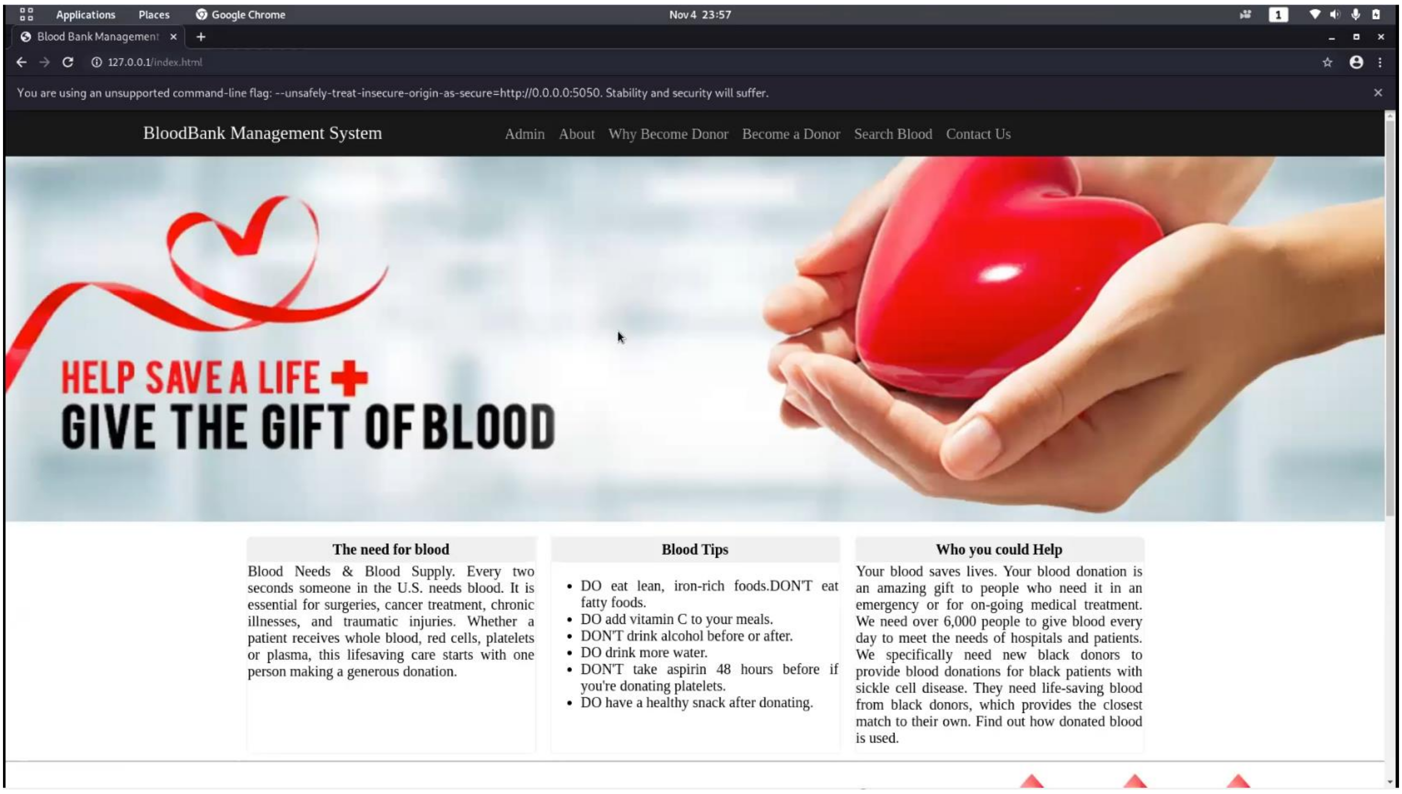
Hello and welcome admin

Answer the following security Question

what is your name?

stopped listening, hope you are done...

- **DB Home page**



You are using an unsupported command-line flag: --unsafely-treat-insecure-origin-as-secure=http://0.0.0.0:5050. Stability and security will suffer.

The need for blood

Blood Needs & Blood Supply. Every two seconds someone in the U.S. needs blood. It is essential for surgeries, cancer treatment, chronic illnesses, and traumatic injuries. Whether a patient receives whole blood, red cells, platelets or plasma, this lifesaving care starts with one person making a generous donation.

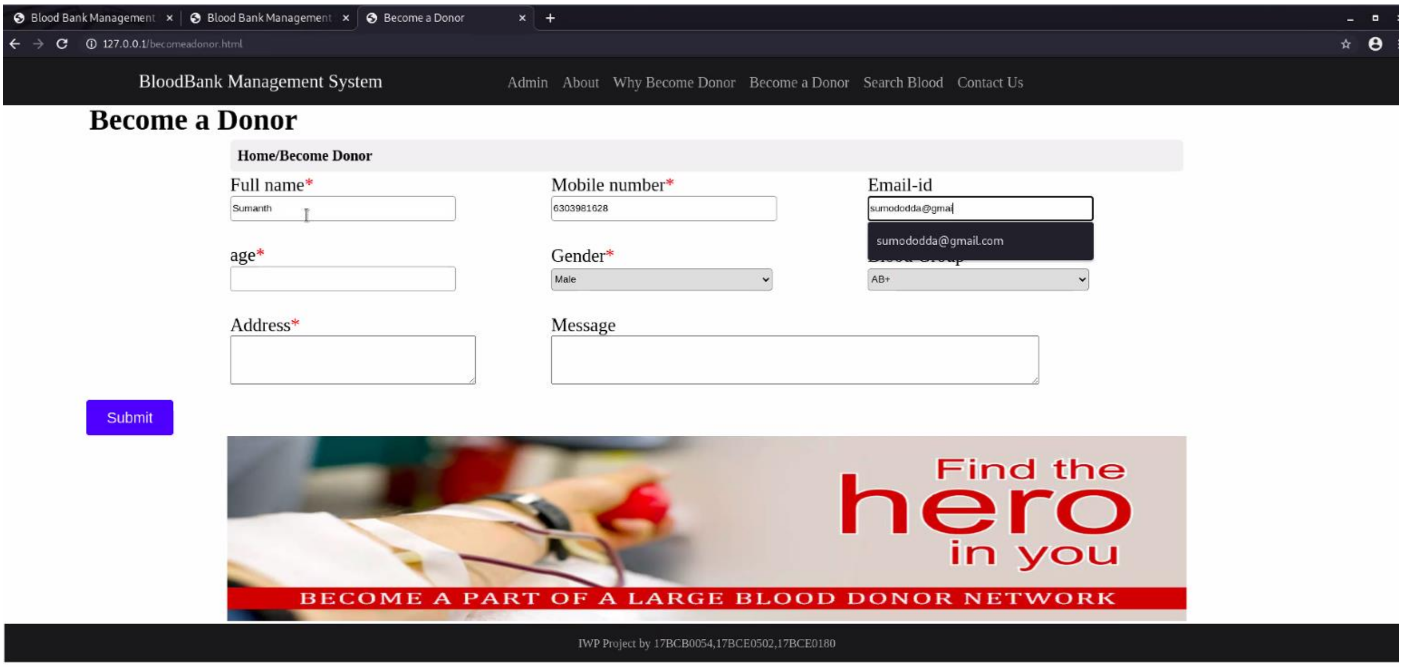
Blood Tips

- DO eat lean, iron-rich foods.DON'T eat fatty foods.
- DO add vitamin C to your meals.
- DON'T drink alcohol before or after.
- DO drink more water.
- DON'T take aspirin 48 hours before if you're donating platelets.
- DO have a healthy snack after donating.

Who you could Help

Your blood saves lives. Your blood donation is an amazing gift to people who need it in an emergency or for on-going medical treatment. We need over 6,000 people to give blood every day to meet the needs of hospitals and patients. We specifically need new black donors to provide blood donations for black patients with sickle cell disease. They need life-saving blood from black donors, which provides the closest match to their own. Find out how donated blood is used.

- **Enrolling in Database**



Become a Donor

Home/Become Donor

Full name*: Sumanth

Mobile number*: 6303981628

Email-id: sumododda@gmai

sumododda@gmail.com

age*:

Gender*: Male

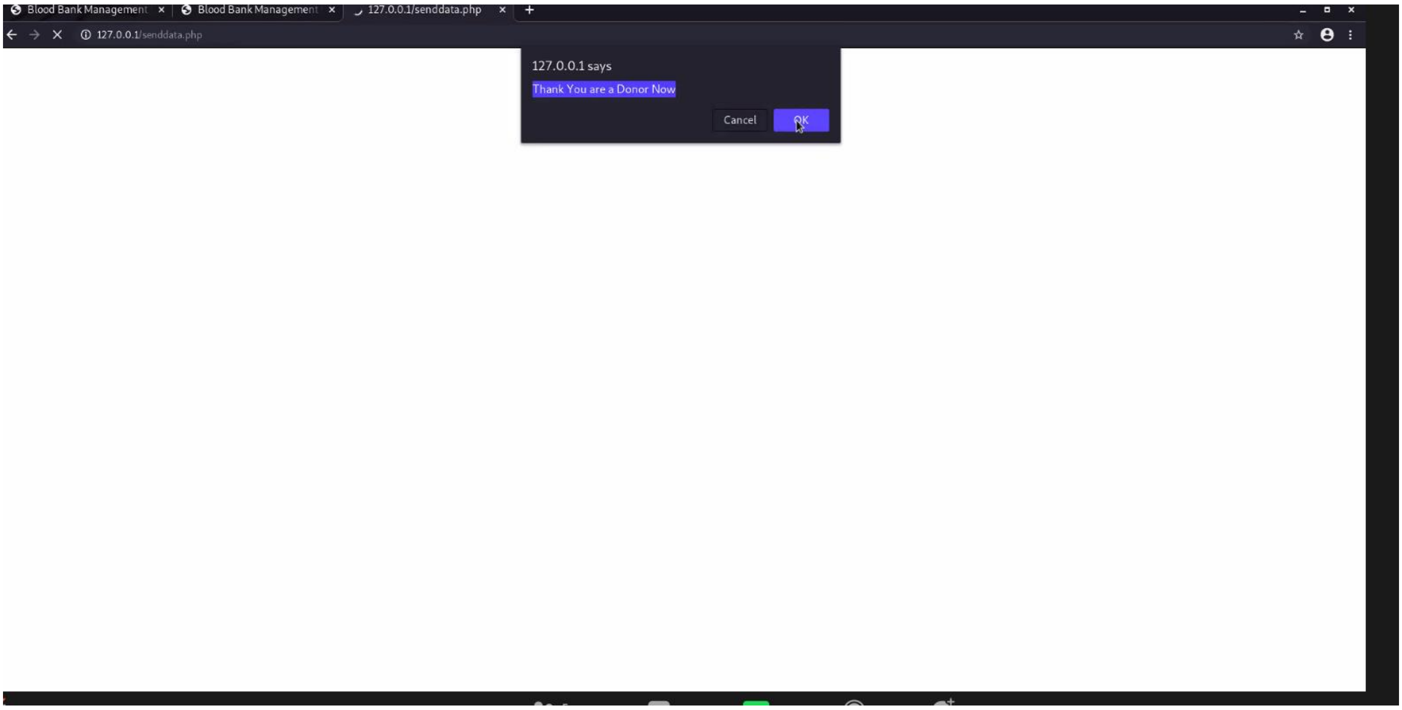
Blood Group: AB+

Address*:

Message:

IWP Project by 17BCB0054,17BCE0502,17BCE0180

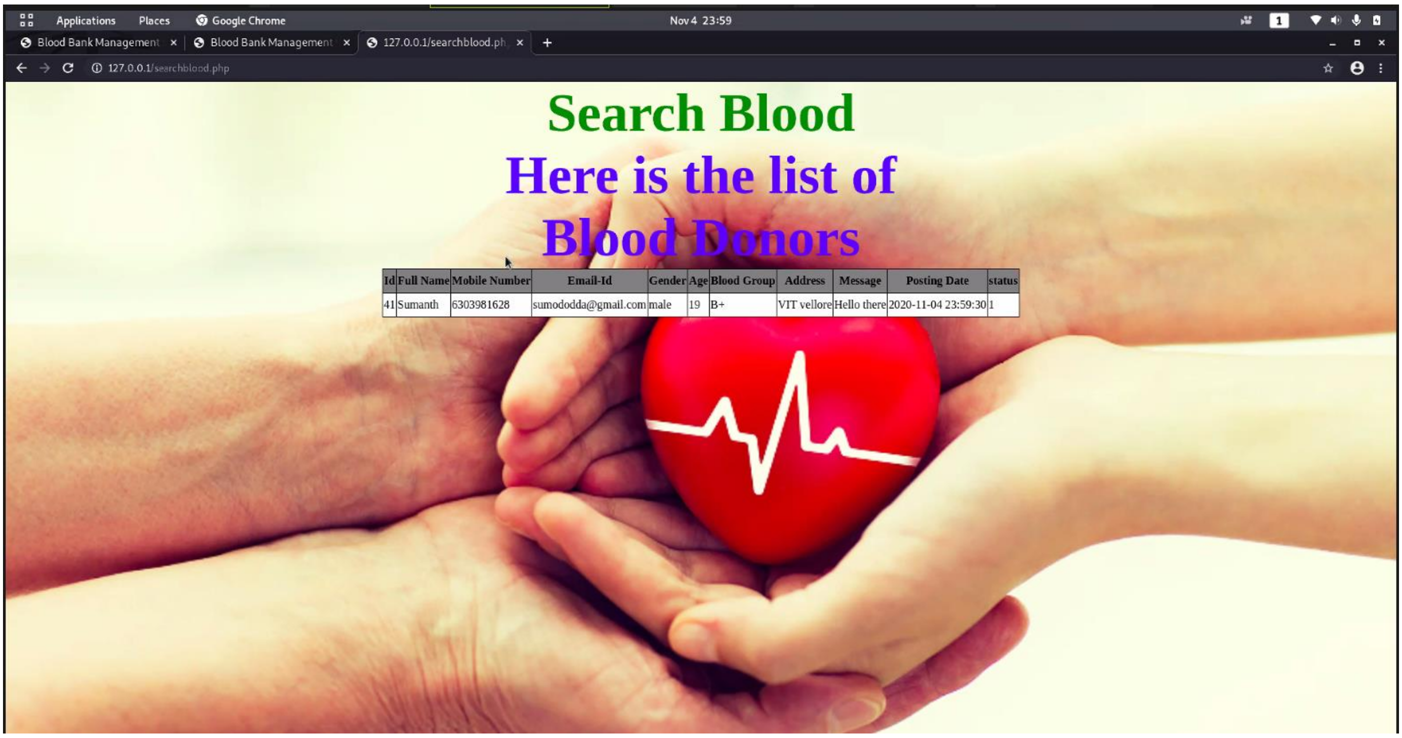
- **Enrolled**



127.0.0.1 says

Thank You are a Donor Now

- **All Participants**

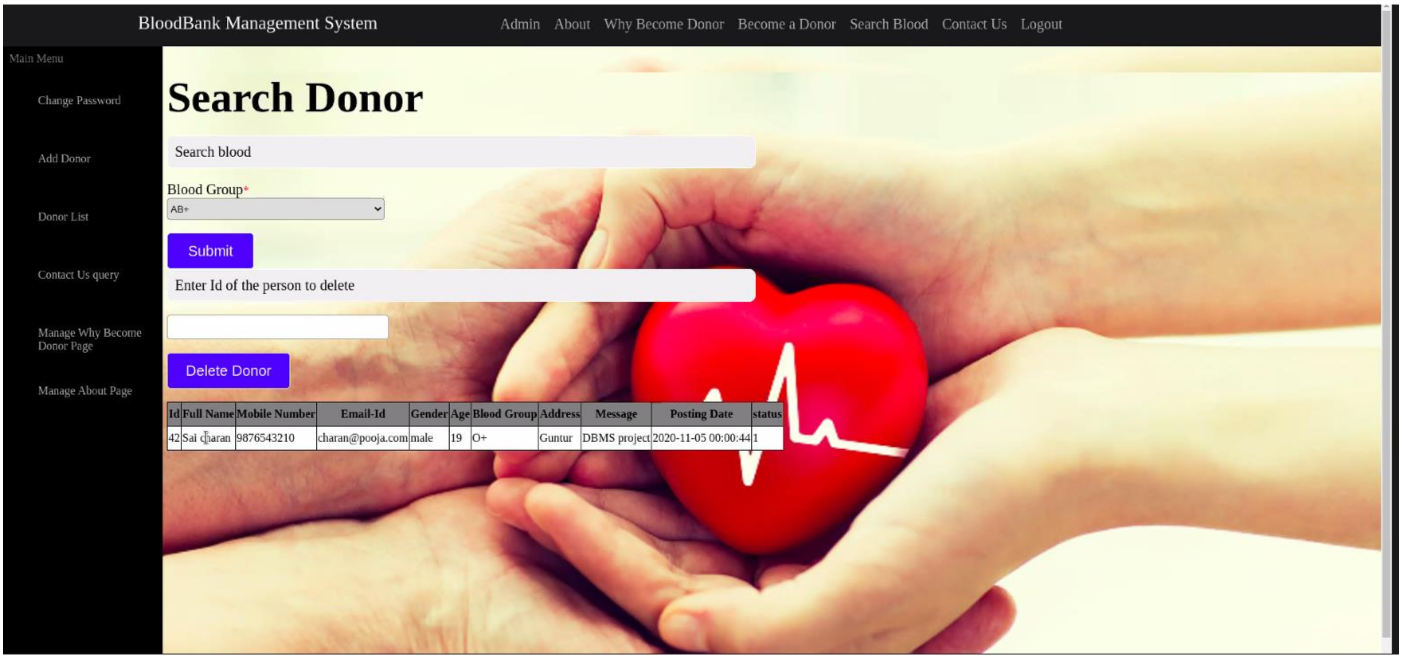


Search Blood

Here is the list of Blood Donors

Id	Full Name	Mobile Number	Email-Id	Gender	Age	Blood Group	Address	Message	Posting Date	status
41	Sumanth	6303981628	sumododda@gmail.com	male	19	B+	VIT vellore	Hello there	2020-11-04 23:59:30	1

- **Search Query**



Search Donor

Search blood

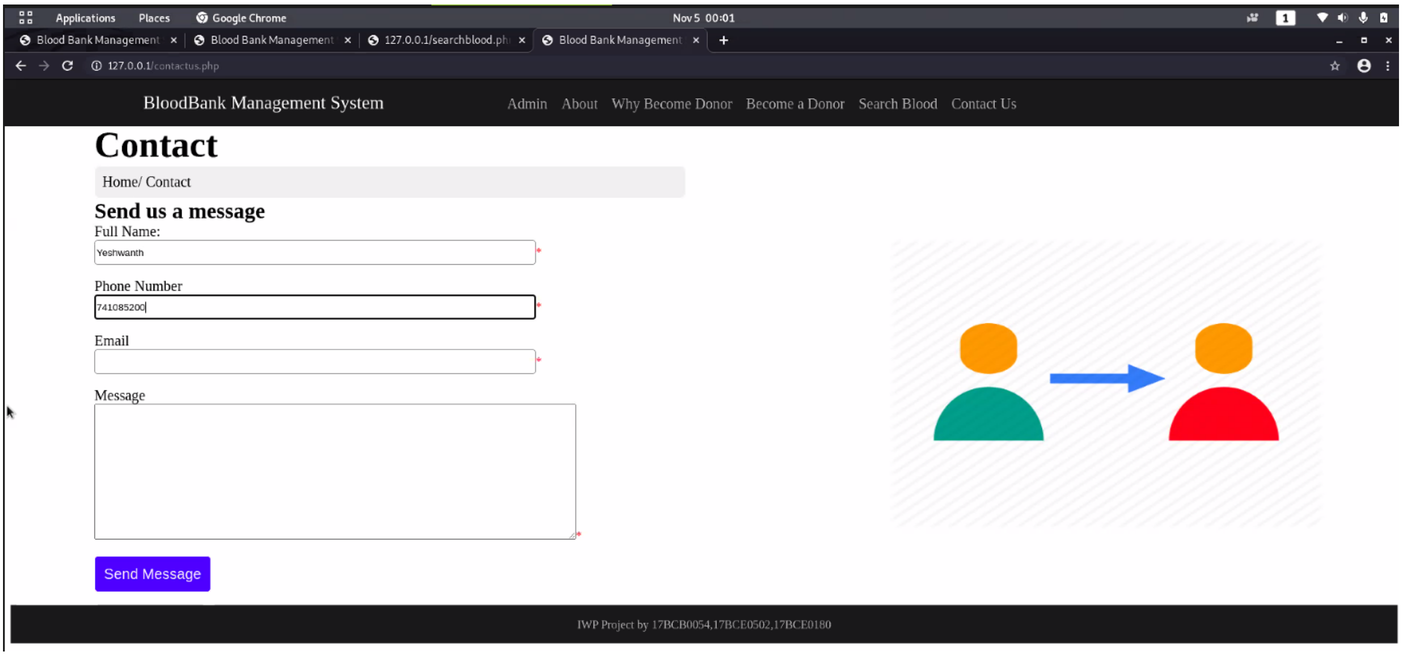
Blood Group*

AB+

Enter Id of the person to delete

Id	Full Name	Mobile Number	Email-Id	Gender	Age	Blood Group	Address	Message	Posting Date	status
42	Sai charan	9876543210	charan@pooja.com	male	19	O+	Guntur	DBMS project	2020-11-05 00:00:44	1

- **Request Query**



Contact

Home/ Contact

Send us a message

Full Name: Yeshwanth

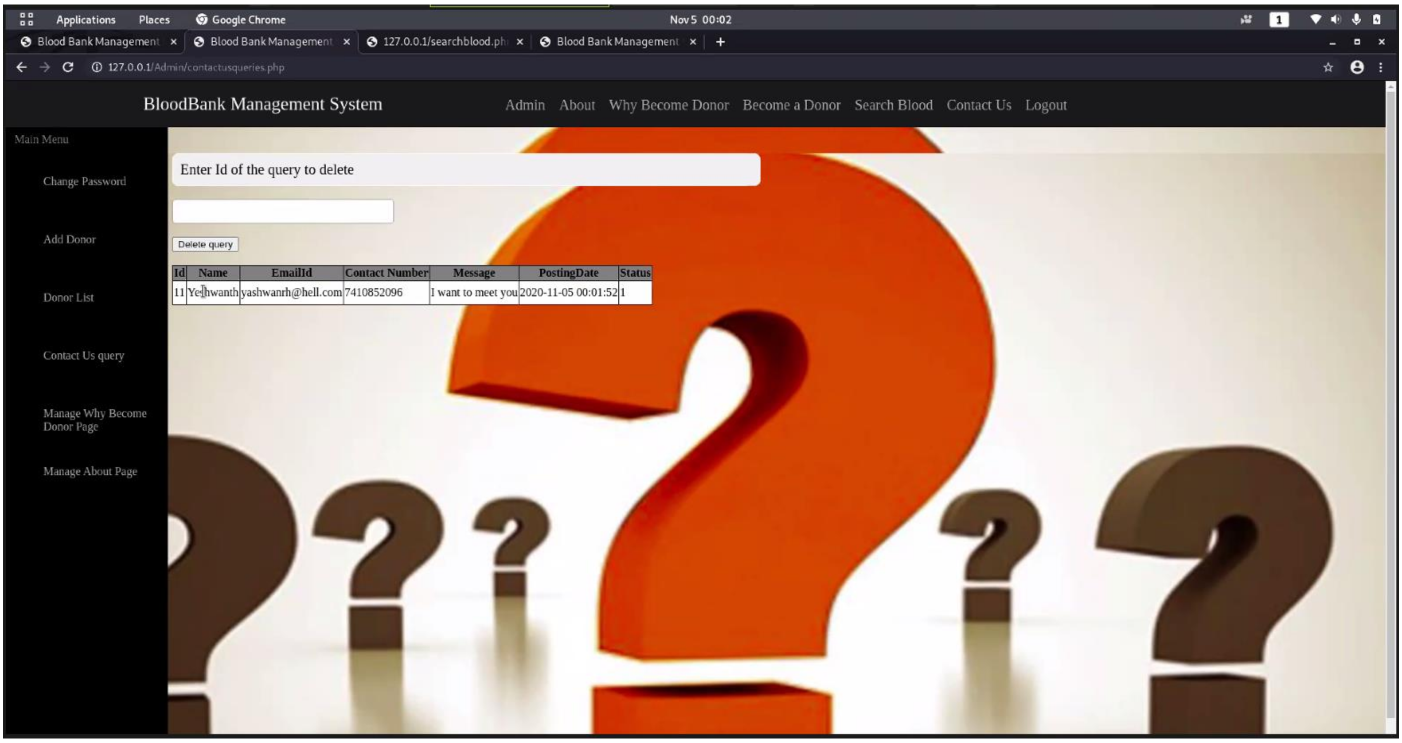
Phone Number 741085200

Email

Message

IWP Project by 17BCB0054,17BCE0502,17BCE0180

- **Admin Check Query**



Enter Id of the query to delete

Id	Name	EmailId	Contact Number	Message	PostingDate	Status
11	Yeshwanth	yashwanrh@hell.com	7410852096	I want to meet you	2020-11-05 00:01:52	1

USE CASE Diagram:

The project’s intention is to extend the usage of this biometric authentication to web-based applications. Which inspired us to employ a “Two – Factor Authentication”, (i.e., Face and speech recognition). We implemented this by employing some predominantly used computer vision techniques and speech recognition API's.

Our system is fast and secure which enhances the security as much as possible. As a result, the login process will be streamlined.

Hence employing a Multimodal authentication will always be a wise option for flawless authentication process in Database Management.

References:

1. Szegedy, Christian, et al. "Inception-v4, inception-resnet and the impact of residual connections on learning." Thirty-First AAAI Conference on Artificial Intelligence. 2017
2. Prasad, Puja S., Rashmi Pathak, Vinit Kumar Gunjan, and HV Ramana Rao. "Deep Learning Based Representation for Face Recognition." In ICCCE 2019, pp. 419-424. Springer, Singapore, 2020
3. Renu Bhatia, "Biometrics and Face Recognition Techniques" in the International Journal of Advanced Research in Computer Science and Software Engineering.
4. A. Rattani, D. R. Kisku, M. Bicego, Member, IEEE and M. Tistarelli of "Feature Level Fusion of Face and Fingerprint Biometrics" on https://arxiv.org
5. Real Time Detection and Recognition of Human Faces,Agnihotram Venkata Sripriya, Mungi Geethika, Vaddi Radhesyam,IEEE Xplore Part Number:CFP20K74-ART; ISBN: 978-1-7281-4876-2
6. Face Detection and Recognition System using Digital Image Processing, Gurlove Singh, Amit Kumar Goel, IEEE Xplore Part Number: CFP20K58-ART; ISBN: 978-1-7281-4167-1
7. A Computer Remote Control System Based on Speech Recognition Technologies of Mobile Devices and Wireless Communication Technologies
8. A Comparative study on end-to-end speech to text translation, Parnia Bahar, Tobias Bieschke, and Hermann Ney.
9. A. Mousa, M. Karabatak, and T. Mustafa, "Database Security Threats and Challenges," 8th Int. Symp. Digit. Forensics Secur. ISDFS 2020, pp. 2–6, 2020, doi: 10.1109/ISDFS49300.2020.9116436.
10. S. M. Toapanta Toapanta, F. G. Mendoza Qumi, D. H. Plua Moran, M. G. Tandazo Espinoza, L. E. Mafla Gallegos, and M. D. R. Maciel Arellano, "Analysis of Security Algorithms for a Distributed Database," Proc. - 2019 Int. Conf. Artif. Intell. Adv. Manuf. AIAM 2019, pp. 50–54, 2019, doi: 10.1109/AIAM48774.2019.00017.
11. S. Samaiya and M. Agarwal, "Real time database management system," Proc. 2nd Int. Conf. Inven. Syst. Control. ICISC 2018, no. Icisc, pp. 903–908, 2018, doi: 10.1109/ICISC.2018.8398931.
12. H. B. Hashim, "Challenges and Security Vulnerabilities to Impact on Database Systems," Al-Mustansiriyah J. Sci., vol. 29, no. 2, p. 117, 2018, doi: 10.23851/mjs.v29i2.332.

13. L. P. Herrero, R. Aliyev, and L. Peñalver, "Analyzing Vulnerability Databases," no. May, 2016.

14. B. Kumar and M. H. S. Al Hasani, "Database security - Risks and control methods," 2016 1st IEEE Int. Conf. Comput. Commun. Internet, ICCCI 2016, pp. 334–340, 2016, doi: 10.1109/CCI.2016.7778937.

15. C. Kaufman, R. Perlman, and M. Speciner, "Database Security - Concepts, Aproaches, and Challenges," vol. 2, no. 1, pp. 2–20, 2002.

16. S. Jajodia, "Database security and privacy," ACM Comput. Surv., vol. 28, no. 1, pp. 129–131, 1996, doi: 10.1145/234313.234370.