# ABSTRACT

**"Blood Bank & Donor Management System"** is a browser-based system that is designed to store, process, retrieve and analyse information concerned with the administrative and inventory management within a blood bank. This system is mainly based on collection, storage, and usage of blood in needy situations. This project aims at maintaining all the information pertaining to blood donors, different blood groups available in each blood bank and help them manage in a better way.Our client is not interested in blood stocking, instead we are stocking blood donors' information. The donors who are interested in donating blood have to register in the database. There is no storage of blood, so no complications in the project. The software is fully integrated with CRM (customer relationship management) as well as CMS (content management system) solution. It is developed in a manner that is easily manageable, time-saving and relieves one from manual work. The requirement of the blood must be requested and we supply the information about the donor. The donors can update their status whether they are available or not.

# 1. INTRODUCTION

A blood donation is a process whereby a person voluntarily has blood drawn to be used for future transfusions when in need at hospitals for treatment procedures that require them. Donation may be of whole blood (blood drawn directly from the body) or of specific components of the blood, such as red blood cells, white blood cells, plasma, and platelets. Blood banks often participate in the process of collecting blood and other procedures such as managing stocks, approving blood requests, and updating donation information.To develop a blood bank information system which focuses on making an online system that is accessible for both donors and administrators.They can also update the personal information through the system, without having to contact the blood bank registry. The system is also developed for the administrators, who are the main authority in the system. Administrators can add, modify, delete, and query any donation 1 information if necessary. The administrator is also responsible for responding to the hospital's blood requests and checking the stocks in the blood bank's inventory.

The purpose of this "**BLOOD BANK & DONOR MANAGEMENT SYSTEM PROJECT"** was to develop a blood management information system to assist in the management of blood donor records and ease/or control the distribution of blood in various parts of the country based on the hospital demands. Without quick and timely access to donor records, creating market strategies for blood donation, lobbying and sensitization of blood donors becomes very difficult. The blood management information system offers functionality to quick access to donor records collected from various parts of the country.

The project analyses the system requirements and then comes up with the requirement specifications. It studies other related systems and then comes up with system specifications. The system is then designed in accordance with specifications to satisfy the requirements. The system design is then implemented with MYSQL, PHP and HTML. The system is designed as an interactive and content management system. The content management system deals with data entry, validation confirm and updating while the interactive system deals with systems and product development projects or, indeed, for any activity where you must manage a relationship.

Data is the code-word of the computer industry. Data refers to a collection of facts usually collected as a result of observation and experiments or processes within a computer system. This may consist of numbers, words or images or observations of a set of variables. Data is often viewed as the lowest level of abstraction from which information and knowledge are derived.

# 2. STATEMENT OF PROBLEMS

The following problem arises when using a typical blood bank's existing system:

## (i) Personal profile accessibility :

The donor's information can only be updated by the administrators of the blood bank. A donor can update their information by calling, faxing, e-mailing, but not by themselves. This is a waste of time just for updating a piece of information and it may be troublesome for some donors.

## (ii) Lost or damaged card :

A typical membership card can easily get damaged if it is exposed to the sunlight or weather and this causes ruin of the card's bar code which,, is significant important for retrieving records. If the card gets lost or stolen, the donor has to make a replacement card to keep their membership at the blood bank

## (iii) Donation record accessibility :

The donor ID card is the only tangible evidence that contains the donor's recent donation records, if the card gets lost, donors may find it difficult to schedule their next appointment since they are not able to see the last time, they had donated blood.

## (iv) Blood result notifications :

After the process of blood donation, the donor will receive a card that only contains their name and blood type. They will not be notified of their blood result unless they request that information from the blood bank.

## (v) Blood stock management :

Blood banks are required to maintain account of blood bags in the inventory. This increases with each blood donation recorded in our system, and decreases as they are checked out upon hospital requests. Our system will need to keep the information up-to-date to ensure correctness of the inventory.

## (vi) Mailing by postal system :

Blood banks will only mail donors when the donated blood is disqualified, however, this mail is sent through the postal system to the donor's given address. If the donor's address is recorded incorrectly, the mail will be sent to the wrong address and the donor will never be notified that their blood is rejected and given the reason for that.

# 3. PROJECT OVERVIEW

## Project Modules :

In this projects, we use PHP and Mysql and it contains two modules.

i. Admin

ii. Donor

### (i) Admin Module :

**Dashboard:** In this section, admin can view all the details in brief like total blood group listed, registered donor list, and total enquiries received.

**Add Donor:** In this section, admin can add donor.

**Donor List:** In this section, admin can view a list of donors and have the right to delete and hide the detail of donor.

**Manage Contact us Query:** In this section, admin can manage query which is received by users.

**Manage Pages:** In this section, admin can manages website pages.

**Update Contact info:** In this section, admin can update the contact details of the website.

Admin can also change the password .

### (ii) User Module :

**Home:** Its is welcome page for users and donor. If any users want to donate the blood they must register with us.

**About Us:** Users can view the about us page.

**Contact Us:** Users can contact with admin the through contact us page.

**Donor List:** Users can view and contact donors.

**Search Donor:** Users can search the donor according to city and blood group

# 4. ANALYSIS

## 4.1 SYSTEM ANALYSIS:

Systems Analysis and Design is an active field in which analysts repetitively learn new approaches and different techniques for building the system more effectively and efficiently. The primary objective of systems analysis and design is to improve organizational systems There are two types of process in the existing system:

### The blood donation process by donors :

When a new donor comes to donate blood, they are required to fill out their personal information during the registration process before donating. After the donation, the donor is given a donor identification card with their name, blood type and a bar code to be used as a reference for future donations. The bar code is used to retrieve the donor's record containing their personal information, medical history, and donation information, including blood results. Only blood bank administrators have the authority to access the donor's records, since the system is only available for their use within the organization. This makes it difficult for donors to make changes to their personal information within the system. That is, for donors to update their personal information, such as their phone number, mailing address, or e-mail, they cannot update the information by themselves, but must contact the blood bank centre to update their information. At the back the card is a table that contains number of donations, date, location, and the blood collector's signature. Existing donors can submit their donor ID cards to retrieve their personal information and donation records and start the blood donation process, and they will be given a new card after they have donated blood for a total of eight times. Having a donor ID card may be a tangible reminder to people that they are helping lives as a blood donor; however, possessing a physical card comes with drawbacks such as loss or damage. To ensure donors can still identify themselves with the system, other credentials, such as username and password, can be used as a safeguard if their donor ID card is lost or damaged. If the donated blood is disqualified, the donor will be notified through postal mail that their blood component is reactive to viruses [4], meaning that there is a positive result of the blood being infected, and the organization will also inform the donor to perform another blood test at the blood bank to confirm the result of blood. If the blood is qualified, the administrator then will deposit the blood into the inventory for future requests.

Blood Request Process by Hospitals can request for blood by calling in or e-mailing the blood bank the type of blood and the quantity that is in need. The administrator is responsible in checking the availability of the blood type according to the request. If the requested blood type is available, the administrator will withdraw the blood from the inventory and transfer it to the hospital. However, if the requested blood is unavailable, the administrator will send an e-mail to inform the hospital.

## 4.2 FEASIBILITY STUDY

Feasibility is conducted to identify the best system that meets all requirements. It is both necessary and important to evaluate the feasibility of a project at the earliest possible time. Feasibility study includes an identification description, an evaluation of proposed system and feasibility study of the proposed system is to be carried out. This is to ensure that the proposed system is not a burden to the shop. The feasibility study should be relatively cheap and quick. The results should inform the decision of whether to go ahead with a more detailed analysis for feasibility analysis, some understanding of the major requirements for the system is essential. Four key considerations involved in the feasibility analysis are:-

**(i)** ECONOMICAL FEASIBILITY

**(ii)** TECHNICAL FEASIBILITY

**(iii)** OPERATIONAL FEASIBILITY

## 4.2.1 ECONOMICAL FEASIBILITY:

Economical feasibility is the most frequently used method for evaluating the effectiveness of the candidate system. It is very essential because the main goal of the proposed system is to have economically better result along with increased efficiency. A cost evaluation is weighed against the ultimate income or product. Economic justification is generally the bottom-line consideration that includes cost benefit analysis, long term corporate income strategies, and cost of resources needed for development and potential market growth. When compared to the advantage obtained from implemented the system its cost is affordable. This organization. The amount of fund that company can pour into the research and development of the system is

limited. The expenditures must be justified. Thus, the developed system as well within the budget and this was achieved because most of the technologies used are freely available.Installation of new system will reduce administrative and operational cost. The newly developed software that does not require any existing manual paper works and files.

So cost also can reduced by removing these types of materials. Proposed system was developed with available resources. Since cost input for the software is almost nil the output of the software is always a profit. Hence software is economically feasible.

### 4.2.2 TECHNICAL FEASIBILITY:

The study is carried out to check the technical feasibility, that is, the technical requirements of the system. Any system developed must not have a high demand on the available technical resources. This will lead to high demands on the available technical resources. The developed system must have a modest requirement, as only minimal or null changes are required for implementing this system. This is related to the technicality of the project. This evaluation determines whether the technology needed for the proposed system is available or not. It deals with hardware as well as software requirements. That is, type of hardware, software and the methods required for running the system are analysed. A study of function, performance and constraints may improve the ability to create an acceptable system, technical feasibility is frequently the most difficult area to achieve at the stage of product engineering process. The scope was whether the work for the project is done with the current equipment and the existing system technology has to be examined in the feasibility study. The result was found to be true. This feasibility is carried out to check the technical requirements of the system.

This system is implemented by using **PHP**. So, it can be used in any **Windows OS** computer. This system requires very low system resources and it will work in almost all configurations. In the existing system all functions are doing manually. So, if they get this designed software, the problems can be avoided and thus the system will run smoothly. In the proposed system, data can be easily stored and managed using database management system software. The reports and results for various queries can be generated easily. Our proposed system is technically feasible to use by any user.

## 4.2.3 OPERATIONAL FEASIBILITY :

The purpose of the operational feasibility is to determine whether the new system will be used if it is developed and implemented and whether there will be resistance from user that will undermine the possible application benefits. The aspect of study is to check the level of acceptance of the system by the user. This includes the process of training the user to use the system efficiently. The user must not feel threatened by the system, instead must accept it as a necessity. The level of acceptance by the user solely depends on the methods that are employed to educate the user about the system and to make him familiar with it. His level of confidence must be raised so that he is also able to make some constructive. The proposed system is an upgrade version of the current system new fields have been implemented according to the user need, hence it ensures a user-friendly environment in such a way that it ensures all the aspects. The proposed system is very much user friendly and the system is easily understood by simple training and it is operationally feasible.

# 5. SYSTEM SPECIFICATIONS

## 5.1 HARDWARE REQUIREMENTS :

- Pentium-IV(Processor).
- 2 GB RAM
- Hard disk with at least free space of 2 GB
- Microsoft Compatible 101 or more Key Board

## 5.2 SOFTWARE REQUIREMENTS :

- Operating System: Windows (Windows 10 Recommended for ease)
- Web-Technology: PHP
- Front-End: HTML, CSS, JAVASCRIPT
- Back-End: MySQL, PHP
- Web Server: Apache SERVER (XAMPP)
- Web Browser: Chrome

# 6. ADVANTAGES & DISADVANTAGES OF THE EXISTING SYSTEM

## 6.1 ADVANTAGES OF THE SYSTEM :

**(i)** User view all blood bank information is location wise.

**(ii)** Donor easily donates blood near blood bank location.

**(iii)** Patient easily request for blood near blood bank location.

**(iv)** In this system also supported inquiry for user.

## 6.2 DISADVANTAGES OF THE SYSTEM:

The following drawbacks of existing system emphasize the need for computerization:

**(i)** A lot of users/ Blood Donator came at a time on **BLOOD BANK & DONOR MANAGEMENT SYSTEM** website, at this situation it is become too difficult maintain it in proper way.

**(ii)** A lot of blood donators came in same time for queries, which makes the task of admin tough.

**(iii)** It lacks of data security.

**(iv)** Retrieval of data takes lot of time.

**(v)** Percentage of accuracy is less.

**(vi)** Reports take time to produce.

# 7. OBJECTIVES OF THE SYSTEM :

The goal of the project is to develop a web application for blood banks to manage information about their donors and blood stock. The main objectives of this website development can be defined as follows:

(i) To develop a system that provides functions to support donors to view and manage their information conveniently.

(ii) To maintain records of blood donors, blood donation information and blood stocks in a centralized database system.

(iii) To inform donors of their blood result after their donation.

(iv) To support searching, matching and requesting for blood convenient for administrators.

(v) To provide a function to send an e-mail directly to the donor for their user account and the hospital, the availability of the blood bag.

# 8. DESIGN APPROACH

## 8.1 SYSTEM DESIGN :

System design is the process of developing specification for candidate system that meet the criteria established in the system analysis. Major step in system design is the preparation of the input forms and the output reports in a form to the user. The main objectives of the system design is to the packages easily by any computer operator system design is the creative act of invention, developing new outputs, a database, offline files, methods, procedures and out for processing business to meet an organization objectives. System builds information gathered during the system analysis. System design has four phases.

## 8.2 DESIGN TOOLS :

### 8.2.1 USE CASE DIAGRAM :

To model a system the most important aspect is to capture the dynamic behaviour. To clarify a bit in details, dynamic behaviour means the behaviour of the system when it is running /operating. So only static behaviour is not sufficient to model a system rather dynamic behaviour is more important than static behaviour. In UML there are five diagrams available to model dynamic nature and use case diagram is one of them. Now as we must discuss that the use case diagram is dynamic in nature there should be some internal or external factors for making the interaction. These internal and external agents are known as actors. So, use case diagrams are consists of actors, use cases and their relationships. The diagram is used to model the system/subsystem of an application. A single use case diagram captures a particular functionality of a system.

The purpose of use case diagram is to capture the dynamic aspect of a system. But this definition is too generic to describe the purpose. Because other four diagrams (activity, sequence, collaboration, and State chart) are also having the same purpose. So we will look into some specific purpose which will distinguish it from other four diagrams. Use case diagrams are used to gather the requirements of a system including internal and external influences. These requirements are mostly design requirements. So, when a system is analysed to gather its functionalities use cases are prepared and actors are identified.

USE CASE FOR ADMIN:

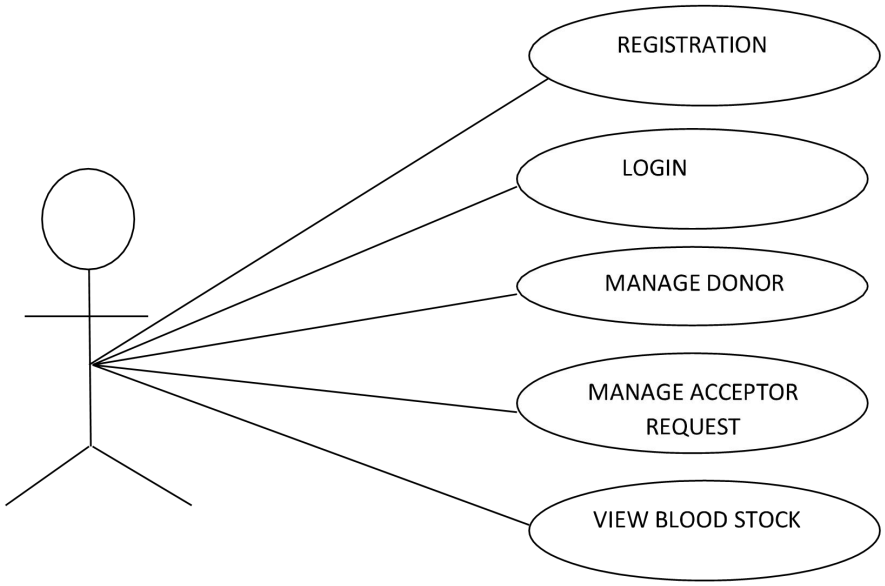



**FIGURE 8.1**

USE CASE FOR DONOR:

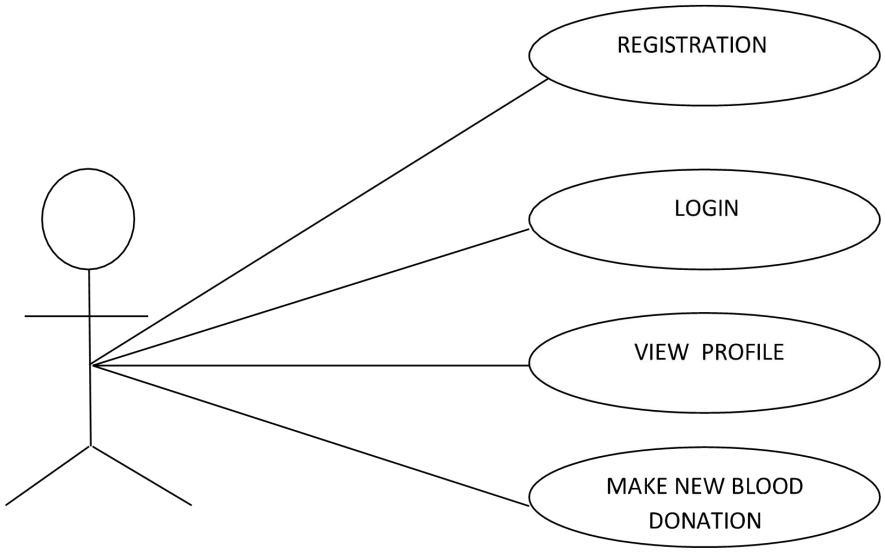



**FIGURE 8.2**

### 8.2.2 DATA FLOW DIAGRAM (DFD) :

A DFD, also known as a “bubble chart” has the purpose of clarifying system-requirements and identifying major transformations that will become programs in system design. A DFD consists of a series of bubbles joined by lines. The bubbles represents data transformations and the lines represents data flow in the system. A data flow diagram may be used to represent a system or software at any level of abstraction. A DFD is a diagram that describes the flow of data and the processes that change or transform data throughout a system. It is a structured analysis and design tool that can be used or flowchart in place of, or in association with, information-oriented and process-oriented system flowcharts. When analyst prepare the DFD, they specify the user needs at a level of detail that virtually determines the information flow into and out of the system and the required data resources. This network is constructed by using a set of symbols that do not imply a physical implementation. The DFD implementation plan reviews the current physical system, prepare input and output specification.

### 8.2.3 CONTEXT DIAGRAM :

A context diagram is a level-0 DFD and represents the entire system elements as a single bubble with input and output data indicated by incoming and outgoing arrows respectively. The user gives data or commands as input and the user will get the details as output.

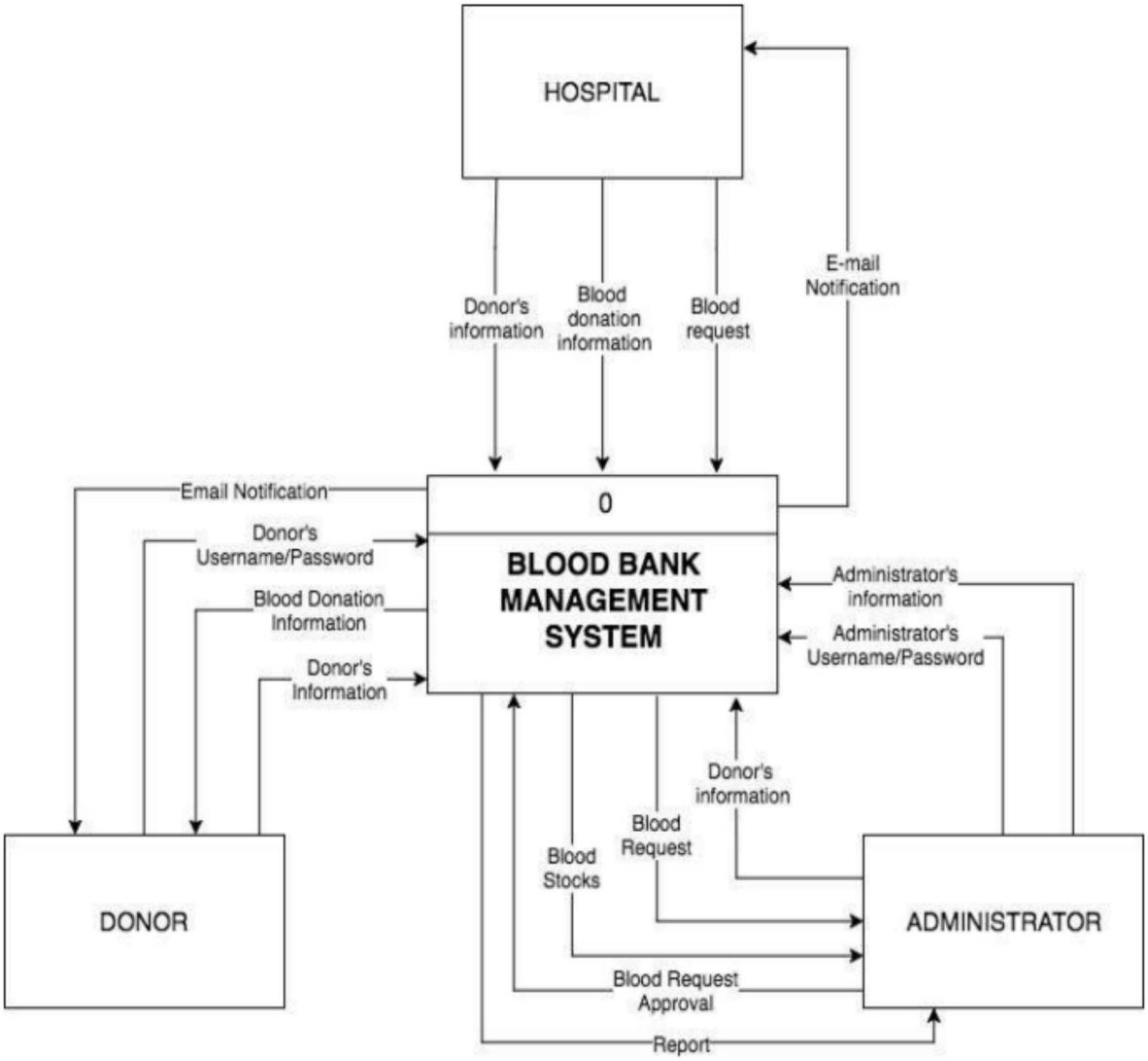



**FIG 8.3 CONTEXT DIAGRAM OF BLOOD BANK & DONOR MANAGEMENT SYSTEM**

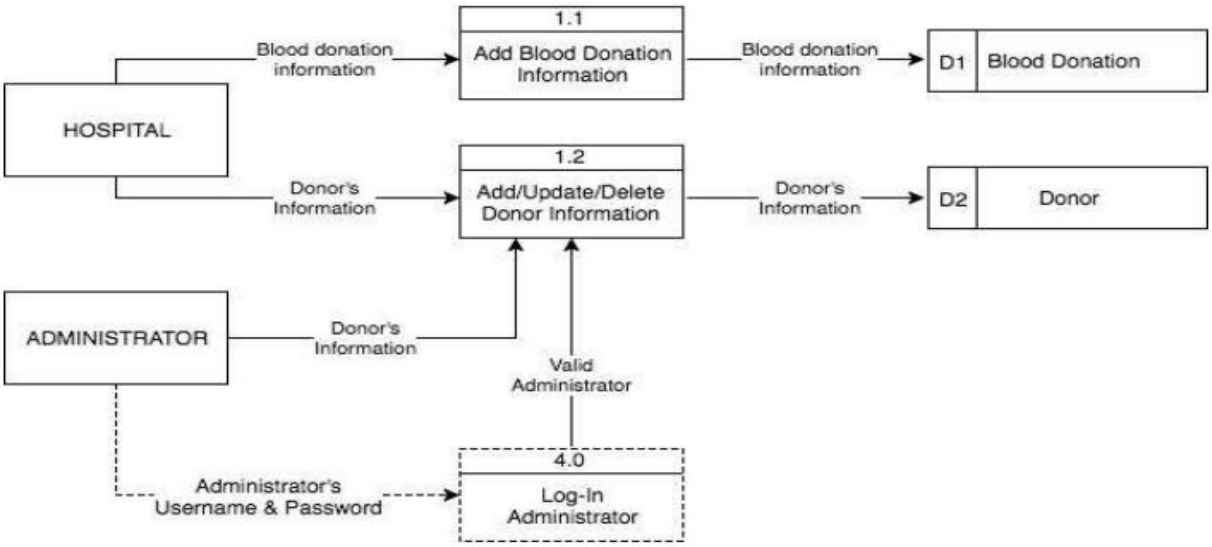



**FIG 8.4 LEVEL 1 DFD**

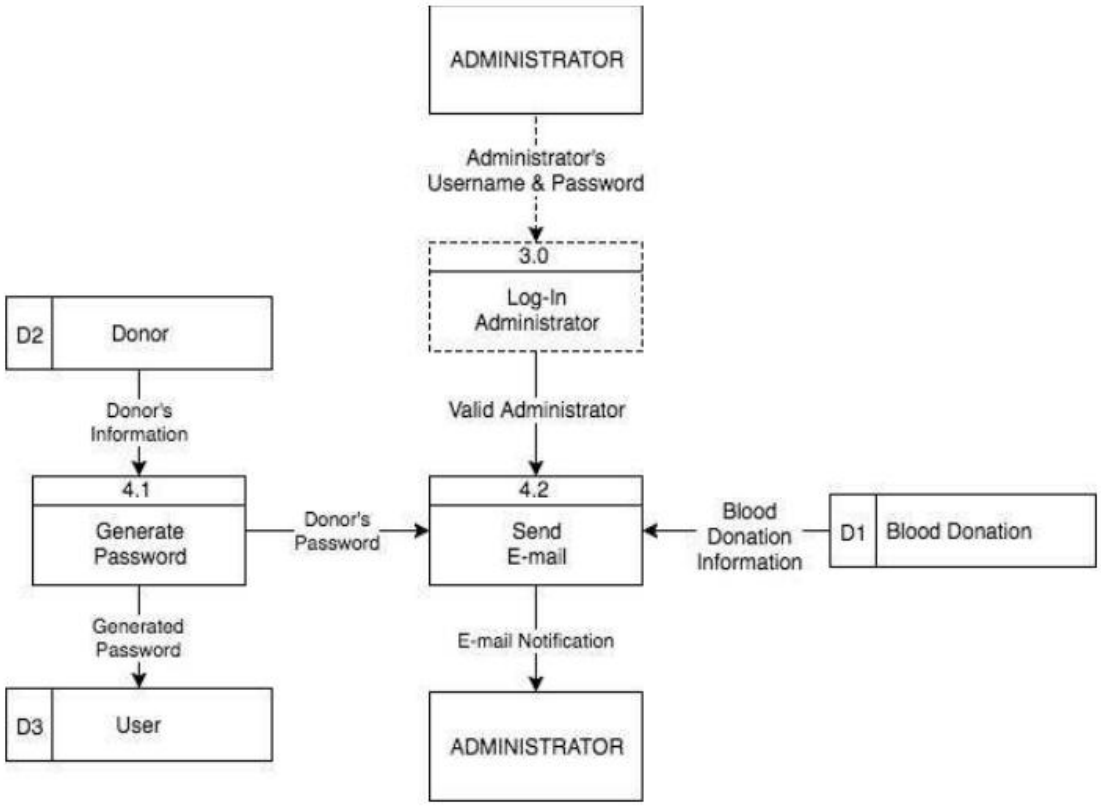



**FIG 8.5 ADMIN STRUCTURE DFD DIAGRAM**

## 8.2.4 ENTITY RELATIONSHIP DIAGRAMS (ERD)

# E-R DIAGRAM

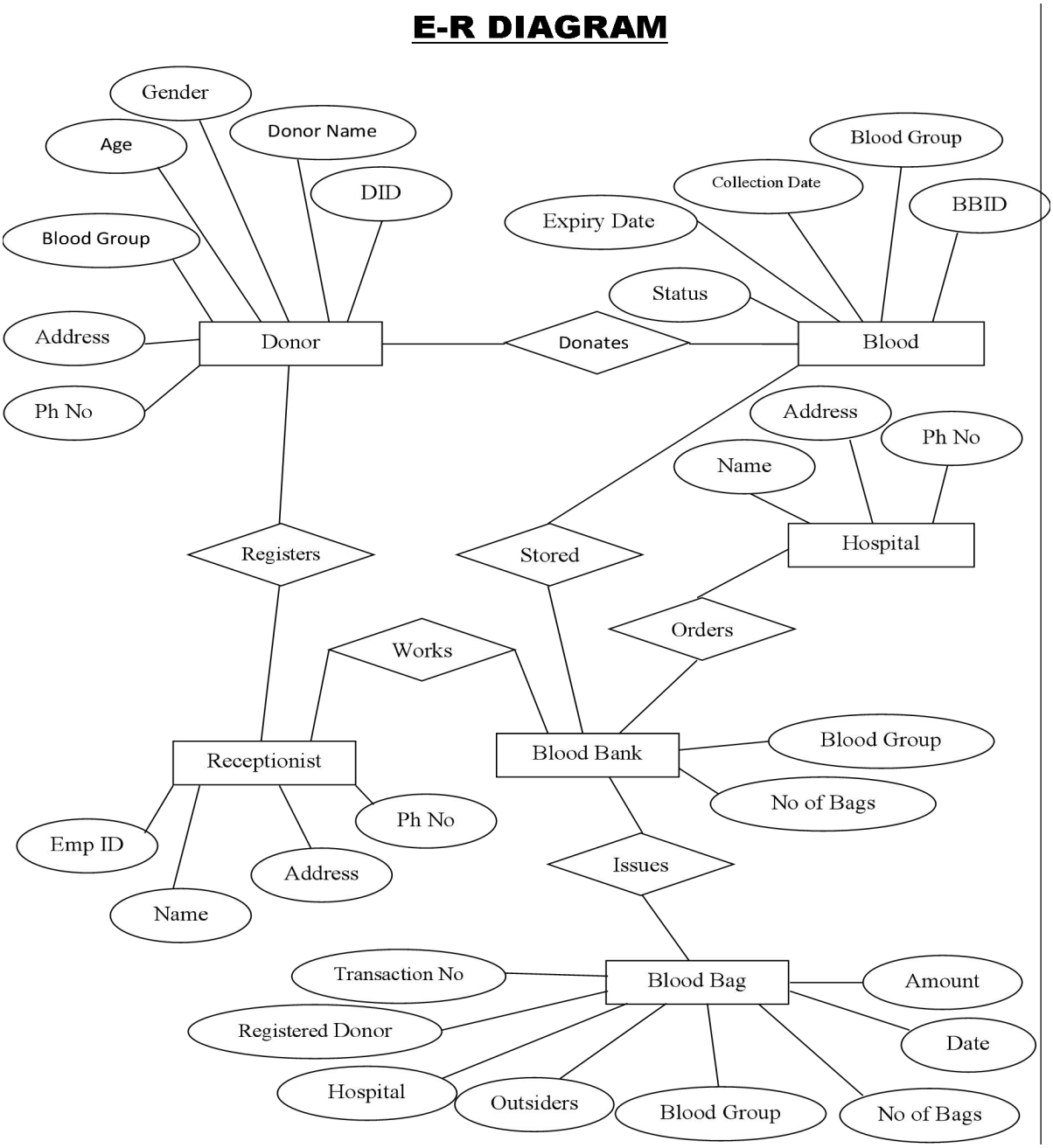



**FIG 8.6 ER DIAGRAM**

# 9. SYSTEM TESTING

Testing is a process of executing a program with the indent of finding an error. Testing is a crucial element of software quality assurance and presents ultimate review of specification, design, and coding. System Testing is an important phase. Testing represents an interesting anomaly for the software. Thus, a series of testing are performed for the proposed system before the system is ready for user acceptance testing. A good test case is one that has a high probability of finding an as undiscovered error. A successful test is one that uncovers an as undiscovered error.

## 9.1 Testing Objectives :

(i) Testing is a process of executing a program with the intent of finding an error.

(ii) A good test case is one that has a probability of finding a yet undiscovered error.

(iii) A successful test is one that uncovers an undiscovered error.

## 9.2 Testing Principles :

(i) All tests should be traceable to end user requirements.

(ii) Tests should be planned long before testing begins.

(iii) Testing should begin on a small scale and progress towards testing in large.

(iv) Exhaustive testing is not possible.

(v) To be most effective testing should be conducted by a independent third party. The primary objective for test case design is to derive a set of tests that has the highest livelihood for uncovering defects in software. To accomplish this objective two different categories of test case design techniques are used.

## 9.3 White-box Testing :

White box testing focus on the program control structure. Test cases are derived to ensure that all statements in the program have been executed at least once during testing and that all logical conditions have been executed.

## 9.4 Black-box Testing :

Black box testing is designed to validate functional requirements without regard to the internal workings of a program. Black box testing mainly focuses on the information domain of the software, deriving test cases by partitioning input and output in a manner that provides through test coverage. Incorrect and missing functions, interface errors, errors in data structures, error in functional logic are the errors falling in this category.

## 9.5 Testing Strategies :

A strategy for software testing must accommodate low-level tests that are necessary to verify that all small source code segment has been correctly implemented as well as high-level tests that validate major system functions against customer requirements.

## 9.6 Testing Fundamentals :

Testing is a process of executing program with the intent of finding error. A good test case is one that has high probability of finding an undiscovered error. If testing is conducted successfully, it uncovers the errors in the software. Testing cannot show the absence of defects, it can only show that software defects present.

## 9.7 Testing Information Flow :

Information flow for testing flows the pattern. Two class of input provided to test the process. The software configuration includes a software requirements specification, a design specification and source code. Test configuration includes test plan and test cases and test tools. Tests are conducted and all the results are evaluated. That is test results are compared with expected results. When erroneous data are uncovered, an error is implied and debugging commences.

## 9.8 Unit Testing :

Unit testing is essential for the verification of the code produced during the coding phase and hence the goal is to test the internal logic of the modules. Using the detailed design description as a guide, important paths are tested to uncover errors with in the boundary of the modules. These tests were carried out during the programming stage itself. All units of ViennaSQL were successfully tested.

## 9.9 Integration Testing :

Integration testing focuses on unit tested modules and build the program structure that is dictated by the design phase.

## 9.10 System Testing :

System testing tests the integration of each module in the system. It also tests to find discrepancies between the system and it's original objective, current specification and system documentation. The primary concern is the compatibility of individual modules. Entire system is working properly or not will be tested here, and specified path ODBC connection will correct or not, and giving output or not are tested here these verifications and validations are done by giving input values to the system and by comparing with expected output. Top-down testing implementing here.

## 9.11 Acceptance Testing:

This testing is done to verify the readiness of the system for the implementation. Acceptance testing begins when the system is complete. Its purpose is to provide the end user with the confidence that the system is ready for use. It involves planning and execution of functional tests, performance tests and stress tests in order to demonstrate that the implemented system satisfies its requirements.

Tools to special importance during acceptance testing include:

- Test coverage Analyzer – records the control paths followed for each test case.
- Timing Analyzer – also called a profiler, reports the time spent in various regions of the code are areas to concentrate on to improve system performance.
- Coding standards – static analyzers and standard checkers are used to inspect code for deviations from standards and guidelines.

## 9.12 Test Cases :

Test cases are derived to ensure that all statements in the program have been executed at least once during testing and that all logical conditions have been executed. Using White-Box testing methods, the software engineer can drive test cases that **: -**

(i) Guarantee that logical decisions on their true and false sides.

(ii) Exercise all logical decisions on their true and false sides.

(iii) Execute all loops at their boundaries and within their operational bounds.

(iv) Exercise internal data structure to assure their validity.

The test case specification for system testing must be submitted for review before system testing commences. elements consisting of "tags" surrounded by angle brackets within the web page content. It can include or can load scripts in languages such as JavaScript which affect the behaviour of HTML processors like Web browsers; and Cascading Style Sheets (CSS) to define the appearance and layout of text and other material. The W3C, maintainer of both HTML and CSS standards, encourages the use of CSS over explicit presentational markup.

Hyper Text Markup Language (HTML) is the encoding scheme used to create and format a web document. A user need not be an expert programmer to make use of HTML for creating hypertext documents that can be put on the internet. Most graphical e-mail clients allow the use of a subset of HTML (often ill-defined) to provide formatting and semantic markup not available with plain text. This may include typographic information like coloured headings, emphasized and quoted text, inline images, and diagrams. Many such clients include both a GUI editor for composing HTML e-mail messages and a rendering engine for displaying them. Use of HTML in e-mail is controversial because of compatibility issues It is possible to simulate many class-based features with prototypes in JavaScript. Functions double as object constructors along with their typical role. Prefixing a function call with new creates a new object and calls that function with its local this keyword bound to that object for that invocation. The constructor's prototype property determines the object used for the new object's internal prototype. JavaScript's built-in constructors, such as Array, also have prototypes that can be modified. Unlike many object-oriented languages, there is no distinction between a function definition and a method definition. Rather, the distinction occurs during function calling.

# 10. Software & Tools Used

## 10.1 About Front End:

The front end is an interface between the user and the back end. The front and back ends may be distributed amongst one or more systems.

In network computing, *front end* can refer to any hardware that optimizes or protects network traffic. It is called application front-end hardware because it is placed on the network's outward-facing front end or boundary. Network traffic passes through the front-end hardware before entering the network.

In compilers, the front end translates a computer programming source code into an intermediate representation, and the back end works with the intermediate representation to produce code in a computer output language. The back end usually optimizes to produce code that runs faster. The front-end/back-end distinction can separate the parser section that deals with source code and the back end that generates code and optimizes.

These days, front-end development refers to the part of the web users interact with. In the past, web development consisted of people who worked with Photoshop and those who could code HTML and CSS. Now, developers need a handle of programs like Photoshop and be able to code not only in HTML and CSS, but also JavaScript or jQuery, which is a compiled library of JavaScript.

Most of everything you see on any website is a mixture of HTML, CSS, and JavaScript, which are all controlled by the browser. For example, if you're using Google Chrome or Firefox, the browser is what translates all of the code in a manner for you to see and with which to interact, such as fonts, colors, drop-down menus, sliders, forms, etc. In order for all of this to work, though, there has to be something to support the front-end; this is where the backend comes into play.

### 10.1.1 HTML :

**HTML** or **Hyper Text Markup Language** is the standard markup language used to create web pages.HTML was created in 1991 by Tim Berners-Lee at CERN in Switzerland. It was designed to allow scientists to display and share their research.

HTML is written in the form of HTML elements consisting of *tags* enclosed in angle brackets(like <html>). HTML tags most commonly come in pairs like <h1> and </h1>, although some tags represent *empty elements* and so are unpaired, for example <img>. The first tag in a pair

is the *start tag*, and the second tag is the *end tag* (they are also called *opening tags* and *closing tags*).

The purpose of a web browser is to read HTML documents and compose them into visible or audible web pages. The browser does not display the HTML tags, but uses the tags to interpret the content of the page. HTML describes the structure of a website semantically along with cues for presentation, making it a markup language rather than a programming language.

HTML elements form the building blocks of all websites. HTML allows images and objects to be embedded and can be used to create interactive forms. It provides a means to create structured documents by denoting structural semantics for text such as headings, paragraphs, lists, links, quotes and other items. It can embed scripts written in languages such as Java Script which affect the behavior of HTML web pages.

HTML is descriptive markup language. Library of various markup languages is defined in various browsers.

### 10.1.2 CSS :

**CSS tutorial** or CSS 3 tutorial provides basic and advanced concepts of CSS technology. Our CSS tutorial is developed for beginners and professionals. The major points of CSS are given below:

- CSS stands for Cascading Style Sheet.
- CSS is used to design HTML tags.
- CSS is a widely used language on the web.
- HTML, CSS and JavaScript are used for web designing. It helps the web designers to apply style on HTML tags.

**Cascading Style Sheets** (**CSS**) is a style sheet language used for describing the look and formatting of a document written in a markup language. While most often used to style web pages and user interfaces written in HTML and XHTML, the language can be applied to any kind of XML document, including plain XML, SVG and XUL. CSS is a cornerstone specification of the web and almost all web pages use CSS style sheets to describe their presentation

.

CSS is designed primarily to enable the separation of document content from document presentation, including elements such as the layout, colors, and fonts. This separation can improve content accessibility, provide more flexibility and control in the specification of presentation characteristics, enable multiple pages to share formatting, and reduce complexity and repetition in the structural content (such as by allowing for table less web design).

CSS can also allow the same markup page to be presented in different styles for different rendering methods, such as on-screen, in print, by voice (when read out by a speech-based browser or screen reader) and on Braille-based, tactile devices. It can also be used to allow the web page to display differently depending on the screen size or device on which it is being viewed. While the author of a document typically links that document to a CSS file, readers can use a different style sheet, perhaps one on their own computer, to override the one the author has specified.

With plain HTML you define the colors and sizes of text and tables throughout your pages. If you want to change a certain element you will therefore have to work your way through the document and change it. With CSS you define the colors and sizes in "styles". Then as you write your documents you refer to the styles. Therefore: if you change a certain style it will change the look of your entire site. Another big advantage is that CSS offers much more detailed attributes than plain HTML for defining the look and feel of your site.

### 10.1.3 JAVA SCRIPT :

**JavaScript** (**JS**) is a dynamic computer programming language. It is most commonly used as part of web browsers, whose implementations allow client-side scripts to interact with the user, control the browser, communicate asynchronously, and alter the document content that is displayed. It is also being used in server-side network programming (with Node.js), game development and the creation of desktop and mobile applications.

JavaScript is a prototype-based scripting language with dynamic typing and has first-class functions. Its syntax was influenced by C. JavaScript copies many names and naming conventions from Java, but the two languages are otherwise unrelated and have very different semantics. The key design principles within JavaScript are taken from the Self and Scheme programming languages. It is a multi-paradigm language, supporting object-oriented, imperative, and functional programming styles.

The application of JavaScript in use outside of web pages—for example, in PDF documents, site-specific browsers, and desktop widgets—is also significant. Newer and faster JavaScript VMs and platforms built upon them (notably Node.js) have also increased the

Popularity of JavaScript for server-side web applications. On the client side, JavaScript was traditionally implemented as an interpreted language but just-in-time compilation is now performed by recent (post-2012) browsers.

JavaScript was formalized in the ECMA Script language standard and is primarily used as part of a web browser (client-side JavaScript). This enables programmatic access to objects within a host environment.

JavaScript is the most popular programming language in the world.

It is the language for HTML, for the Web, for computers, servers, laptops, tablets, smart phones, and more.

You can use JavaScript to:

- Change HTML elements
- Delete HTML elements
- Create new HTML elements
- Copy and clone HTML elements

### 10.1.4 PHP :

PHP is now officially known as "**PHP: Hypertext Preprocessor**". It is a server-side scripting language usually written in an HTML context. Unlike an ordinary HTML page, a PHP script is not sent directly to a client by the server; instead, it is parsed by the PHP binary or module, which is server-side installed. HTML elements in the script are left alone, but PHP code is interpreted and executed. PHP code in a script can query databases, create images, read and write files, talk to remote servers – the possibilities is endless. The output from PHP code is combined with the HTML in the script and the result sent to the user's web-browser, therefore it can never tell the user whether the web-server uses PHP or not, because the entire browser sees is HTML.

PHP's support for Apache and MySQL further increases its popularity. Apache is now the most-used web-server in the world, and PHP can be compiled as an Apache module. MySQL is a powerful free SQL database, and PHP provides a comprehensive set of functions for working with it. The combination of Apache, MySQL and PHP is all but unbeatable.

That doesn't mean that PHP cannot work in other environments or with other tools. In fact, PHP supports an extensive list of databases and web-servers. While in the mid-1990s it was ok to build sites, even relatively large sites, with hundreds of individual hard-coded HTML pages,

today's webmasters are making the most of the power of databases to manage their content more effectively and to personalize their sites according to individual user preferences.

## 10.2 About Back End:

In a previous blog, we talked about how web programmers are concerned with launching websites, updates, and maintenance, among other things. All of that works to support the front-end of the website. The back-end has three parts to it: server, application, and database.

To better explain how all of this works, let's use the example of a customer trying to purchase a plane ticket using a website. Everything that the customer sees on the webpage is the front-end, as we have explained before, but once that customer enters all of his or her information, ssssuch as their name, billing address, destination, etc, the web application stores the information in a database that was created previously on the server in which the website is calling for information.

The web application creates, deletes, changes, renames, etc items in the database. For example, when a customer purchases a ticket, that creates an item in the database, but when they have a change in their order or they wish to cancel, the item in the database is changed.

.In short, when a customer wants to buy a ticket, the backend operation is the web application communicating with the server to make a change in a database stored on said server. Technologies like PHP, Ruby, Python, and others are the ones backend programmers use to make this communication work smoothly, allowing the customer to purchase his or her ticket with ease.

### 10.2.1 MySQL :


**Introduction:**

There are a large number of database management systems currently available, some commercial and some free. Some of them : Oracle, Microsoft Access, MySQL and PostgreSQL.

These database systems are powerful, feature-rich software, capable of organizing and searching millions of records at very high speeds.

The database has become an integral part of almost every human's life. Without it, many things we do would become very tedious, perhaps impossible tasks. Banks, universities, and libraries are three examples of organizations that depend heavily on some sort of database system. On the Internet, search engines, online shopping, and even the website naming convention would be impossible without the use of a database. A database that is implemented and interfaced on a computer is often termed a database server

One of the fastest SQL (Structured Query Language) database servers currently on the market is the MySQL server, developed by T.c.X. DataKonsultAB. MySQL, available for download at www.mysql.com, offers the database programmer with an array of options and capabilities rarely seen in other database servers. MySQL is free of charge for those wishing to use it for private and commercial use. Those wishing to develop applications specifically using MySQL should consult MySQL's licensing section, as there is charge a for licensing the product.

### 10.2.2 .Understanding Databases, Records, and Primary Keys :

Every database is composed of one or more tables. These tables, which structure data into rows and columns, impose organization on the data. The records in the table(below) are not arranged in any particular order. To make it easy to identify a specific record, therefore, it becomes necessary.

### 10.2.3 Relational Database Management System (RDBMS) :

You already know that a single database can hold multiple tables. In a Relational database management system (RDBMS), these tables can be linked to each other by one or more common fields, called foreign keys.

:

### 10.2.4 Database administrator (DBA) :

The database administrator is the super user of databases; he has unrestricted rights and privileges to access databases, granting permission to other database users.

# 11. FUTURE SCOPE OF THE PROJECT

The system functions and features of our system will include the following:

## (i) Registration:

This function allows the donor and administrator to register as a user to interact with the system. The system requires the user to login before viewing and editing any information.

## (ii) View and edit information online:

Donors are allowed to view their blood donation records online by their given account. They can also edit their personal information through the system.

## (iii) Data is entered by the Administrators:

The donors' information and donation records can be sent from the hospital to the administrator by call or e-mail. The administrator is responsible for keying the received data into the system.

## (iv) Recording donation records :

The system can record data of whole blood which is sent from the hospital.

## (v) Manage blood inventory:

The system uses a First-In-First-Out stock management, where the blood stock that is checked-in to the system first will be the first one given to the hospital

when requested. When the blood stock is expired, the administrator is responsible for removing the stock from the inventory and updating the system.

### (vi) Blood requests :

The hospital can request blood via e-mail and by calling the blood bank.

### (vii) Notify by E-mail :

The donor's account and generated password will be sent via e-mail, following by their blood result of the previous donation sent in a separated e-mail. Hospitals can also receive e-mail responding to their requested blood whether it is available in our stock or not.

### (viii) Summary Report :

The system can generate a report to summarize all records, including blood donations, blood requests and blood stock for the administrator.

# 12. SOURCE CODE :

## User :

### 1. home.php :

```
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="utf-8">
   <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">
   <meta name="description" content="">
   <meta name="author" content="">
  <linkrel="stylesheet"
href="https://maxcdn.bootstrapcdn.com/bootstrap/4.5.2/css/bootstrap.min.css">
  <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.5.1/jquery.min.js"></script>
  <script src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.16.0/umd/popper.min.js"></script>
  <script src="https://maxcdn.bootstrapcdn.com/bootstrap/4.5.2/js/bootstrap.min.js"></script>
<style>
</style>
</head>
<body>
<div class="header">
<?php
$active="home";
```

```
include('head.php'); ?>
</div>
<?php include'ticker.php'; ?>
  <div id="page-container" style="margin-top:50px; position: relative;min-height: 84vh;   ">
    <div class="container">
    <div id="content-wrap"style="padding-bottom:75px;">
  <div id="demo" class="carousel slide" data-ride="carousel">
 <!-- Indicators -->
    <ul class="carousel-indicators">
      <li data-target="#demo" data-slide-to="0" class="active"></li>
      <li data-target="#demo" data-slide-to="1"></li>
      <li data-target="#demo" data-slide-to="2"></li>
    </ul>
<!-- The slideshow -->
    <div class="carousel-inner">
      <div class="carousel-item active">
        <img src="image\donor.png" alt="image\donor.png" width="100%" height="500">
      </div>
      <div class="carousel-item">
        <img      src="image\Blood-facts_10-illustration-graphics__canteen.png"      alt="image\Blood-
facts_10-illustration-graphics__canteen.png" width="100%" height="500">
      </div>
      <div class="carousel-item">
        <img src="image\save.jpg" alt="image\save.jpg" width="100%" height="500">
      </div>
</div>
```

```
<!-- Left and right controls -->
 <a class="carousel-control-prev" href="#demo" data-slide="prev">
    <span class="carousel-control-prev-icon"></span>
  </a>
  <a class="carousel-control-next" href="#demo" data-slide="next">
    <span class="carousel-control-next-icon"></span>
  </a>
 </div>
<br>
 <h1 style="text-align:center;font-size:45px;font-family: 'Franklin Gothic Medium', 'Arial Narrow', Arial, sans-serif;">Welcome to Blood Bank & Donor Management System</h1>
<br>
 <div class="row">
          <div class="col-lg-4 mb-4">
             <div class="card">
                <h4 class="card-header card bg-info text-white" >The Need for Blood</h4>
 <p class="card-body overflow-auto" style="padding-left:2%;height:120px;text-align:left;">
                     <?php
                       include 'conn.php';
                       $sql=$sql= "select * from pages where page_type='needforblood'";
                       $result=mysqli_query($conn,$sql);
                       if(mysqli_num_rows($result)>0)   {
                          while($row = mysqli_fetch_assoc($result)) {
                            echo $row['page_data'];
                          }
                        }
```

```
                    ?>
                  </p>
              </div>
          </div>
          <div class="col-lg-4 mb-4">
              <div class="card">
                  <h4 class="card-header card bg-info text-white">Blood Tips</h4>
  <p class="card-body overflow-auto" style="padding-left:2%;height:120px;text-align:left;">
                   <?php
                    include 'conn.php';
                    $sql=$sql= "select * from pages where page_type='bloodtips'";
                    $result=mysqli_query($conn,$sql);
                    if(mysqli_num_rows($result)>0)   {
                       while($row = mysqli_fetch_assoc($result)) {
                         echo $row['page_data'];
                       }
                     }
  ?>
    </p>
</div>
          </div>
          <div class="col-lg-4 mb-4">
              <div class="card">
                  <h4 class="card-header card bg-info text-white" >Who You Could Help</h4>
<p class="card-body overflow-auto" style="padding-left:2%;height:120px;text-align:left;">
                   <?php
```

```
                include 'conn.php';
   $sql=$sql= "select * from pages where page_type='whoyouhelp'";
                $result=mysqli_query($conn,$sql);
                if(mysqli_num_rows($result)>0)   {
                   while($row = mysqli_fetch_assoc($result)) {
                     echo $row['page_data'];
                   }
                  }
 ?>
               </p>
 </div>
          </div>
</div>
 <h2>Blood Donor Names</h2>
 <div class="row">
        <?php
          include 'conn.php';
          $sql= "select * from donor_details join blood where donor_details.donor_blood=blood.blood_id order by rand() limit 6";
          $result=mysqli_query($conn,$sql);
          if(mysqli_num_rows($result)>0)
          {
          while($row = mysqli_fetch_assoc($result)) {
         ?>
          <div class="col-lg-4 col-sm-6 portfolio-item" ><br>
          <div class="card" style="width:300px">
```

```
            <img class="card-img-top" src="image\blood_drop_logo.jpg" alt="Card image" style="width:100%;height:300px">
            <div class="card-body">
              <h3 class="card-title"><?php echo $row['donor_name']; ?></h3>
              <p class="card-text">
                <b>Blood Group : </b> <b><?php echo $row['blood_group']; ?></b><br>
                <b>Mobile No. : </b> <?php echo $row['donor_number']; ?><br>
                <b>Gender : </b><?php echo $row['donor_gender']; ?><br>
                <b>Age : </b> <?php echo $row['donor_age']; ?><br>
                <b>Address : </b> <?php echo $row['donor_address']; ?><br>
              </p>
 </div>
           </div>
       </div>
     <?php }} ?>
</div>
<br>
    <!-- /.row -->
<!-- Features Section -->
       <div class="row">
          <div class="col-lg-6">
             <h2>BLOOD GROUPS</h2>
             <p>
               <?php
                 include 'conn.php';
```

```
            $sql=$sql= "select * from pages where page_type='bloodgroups'";
            $result=mysqli_query($conn,$sql);
            if(mysqli_num_rows($result)>0)   {
               while($row = mysqli_fetch_assoc($result)) {
                 echo $row['page_data'];
               }
              }
 ?></p>
</div>
      <div class="col-lg-6">
          <img class="img-fluid rounded" src="image\blood_donationcover.jpeg" alt="" >
       </div>
    </div>
    <!-- /.row -->
<hr>
<!-- Call to Action Section -->
    <div class="row mb-4">
      <div class="col-md-8">
      <h4>UNIVERSAL DONORS AND RECIPIENTS</h4>
      <p>
        <?php
          include 'conn.php';
          $sql=$sql= "select * from pages where page_type='universal'";
          $result=mysqli_query($conn,$sql);
          if(mysqli_num_rows($result)>0)   {
```

```
                while($row = mysqli_fetch_assoc($result)) {
                  echo $row['page_data'];
                 }
               }
 ?></p>
           </div>
          <div class="col-md-4">
            <a class="btn btn-lg btn-secondary btn-block" href="donate_blood.php" style="align:center; background-color:#7FB3D5;color:#273746 ">Become a Donor </a>
          </div>
       </div>
</div>
  </div>
  <?php include('footer.php');?>
</div>
</body>
</html>
```

## 2. about_us.php :

```
<html>
<head>
  <meta charset="utf-8">
  <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">
  <meta name="description" content="">
  <meta name="author" content="">
```

```
 <linkrel="stylesheet"
href="https://maxcdn.bootstrapcdn.com/bootstrap/4.5.2/css/bootstrap.min.css">
 <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.5.1/jquery.min.js"></script>
 <script src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.16.0/umd/popper.min.js"></script>
 <script src="https://maxcdn.bootstrapcdn.com/bootstrap/4.5.2/js/bootstrap.min.js"></script>
</head>
<style>
</style>
<body>
<?php
$active ='about';
include('head.php');
?>
div id="page-container" style="margin-top:50px; position: relative;min-height: 84vh;">
 <div class="container">
 <div id="content-wrap" style="padding-bottom:50px;">
<div class="row">
   <div class="col-lg-6">
      <h1 class="mt-4 mb-3">About Us</h1>
      <p> <?php
        include 'conn.php';
        $sql=$sql= "select * from pages where page_type='aboutus'";
        $result=mysqli_query($conn,$sql);
        if(mysqli_num_rows($result)>0)   {
           while($row = mysqli_fetch_assoc($result)) {
             echo $row['page_data'];
```

```
          }
        }
 ?>
</p>
   </div>
   <div class="col-lg-6">
      <img class="img-fluid rounded" src="image\banner_590x300.jpg" style="height:400px" alt="error" >
   </div>
</div>
</div></div>
<?php include('footer.php')
?>
</div>
</body>
</html>
```

### 3. why_donate_blood :

```
<html>
<head>
  <meta charset="utf-8">
  <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">
  <meta name="description" content="">
  <meta name="author" content="">
  <linkrel="stylesheet"
href="https://maxcdn.bootstrapcdn.com/bootstrap/4.5.2/css/bootstrap.min.css">
```

```
 <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.5.1/jquery.min.js"></script>
 <script src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.16.0/umd/popper.min.js"></script>
 <script src="https://maxcdn.bootstrapcdn.com/bootstrap/4.5.2/js/bootstrap.min.js"></script>
</head>
<body>
<?php
$active ='why';
include('head.php');
?>
<div id="page-container" style="margin-top:50px; position: relative;min-height: 84vh;">
 <div class="container">
 <div id="content-wrap" style="padding-bottom:50px;">
<div class="row">
   <div class="col-lg-6">
      <h1 class="mt-4 mb-3">Why Should I Donate Blood ? </h1>
      <p>
        <?php
          include 'conn.php';
          $sql=$sql= "select * from pages where page_type='donor'";
          $result=mysqli_query($conn,$sql);
          if(mysqli_num_rows($result)>0)   {
             while($row = mysqli_fetch_assoc($result)) {
               echo $row['page_data'];
             }
            }
```

```
 ?>
</p>
</div>
   <div class="col-lg-6">
      <img class="img-fluid rounded" src="image\08f2fccc45d2564f74ead4a6d5086871.png" style="height:600px; width:500px" alt="error" >
   </div>
</div>
</div>
</div>
<?php include('footer.php')
?>
</div>
</body>
</html>
```

### 4. donate_blood :

```
<html>
<head>
 <meta charset="utf-8">
 <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">
 <meta name="description" content="">
 <meta name="author" content="">
<linkrel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/4.5.2/css/bootstrap.min.css">
<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.5.1/jquery.min.js"></script>
```

```
<script src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.16.0/umd/popper.min.js"></script>
<script src="https://maxcdn.bootstrapcdn.com/bootstrap/4.5.2/js/bootstrap.min.js"></script>
</head>
<body>
<?php
$active ='donate';
 include('head.php')?>
<div id="page-container" style="margin-top:50px; position: relative;min-height: 84vh;">
  <div class="container">
  <div id="content-wrap" style="padding-bottom:50px;">
<div class="row">
    <div class="col-lg-6">
       <h1 class="mt-4 mb-3">Donate Blood </h1>
     </div>
</div>
<form name="donor" action="savedata.php" method="post">
<div class="row">
<div class="col-lg-4 mb-4">
<div class="font-italic">Full Name<span style="color:red">*</span></div>
<div><input type="text" name="fullname" class="form-control" required></div>
</div>
<div class="col-lg-4 mb-4">
<div class="font-italic">Mobile Number<span style="color:red">*</span></div>
<div><input type="text" name="mobileno" class="form-control" required></div>
</div>
```

```
<div class="col-lg-4 mb-4">
<div class="font-italic">Email Id</div>
<div><input type="email" name="emailid" class="form-control"></div>
</div>
</div>
<div class="row">
<div class="col-lg-4 mb-4">
<div class="font-italic">Age<span style="color:red">*</span></div>
<div><input type="text" name="age" class="form-control" required></div>
</div>
<div class="col-lg-4 mb-4">
<div class="font-italic">Gender<span style="color:red">*</span></div>
<div><select name="gender" class="form-control" required>
<option value="">Select</option>
<option value="Male">Male</option>
<option value="Female">Female</option>
</select>
</div>
</div>
<div class="col-lg-4 mb-4">
<div class="font-italic">Blood Group<span style="color:red">*</span></div>
<div><select name="blood" class="form-control" required>
  <option value=""selected disabled>Select</option>
  <?php
    include 'conn.php';
```

```
   $sql= "select * from blood";
   $result=mysqli_query($conn,$sql) or die("query unsuccessful.");
  while($row=mysqli_fetch_assoc($result)){
   ?>
  <option value=" <?php echo $row['blood_id'] ?>"> <?php echo $row['blood_group'] ?> </option>
  <?php } ?>
</select>
</div>
</div>
</div>
<div class="row">
<div class="col-lg-4 mb-4">
<div class="font-italic">Address<span style="color:red">*</span></div>
<div><textarea class="form-control" name="address" required></textarea></div></div>
</div>
<div class="row">
  <div class="col-lg-4 mb-4">
  <div><input type="submit" name="submit" class="btn btn-primary" value="Submit" style="cursor:pointer"></div>
  </div>
</div>
</div>
</div>
<?php include('footer.php') ?>
</div>
```

```
</body>

</html>
```

## 5. search_blood_group :

```
<?php

$bg=$_POST['blood'];

$conn=mysqli_connect("localhost","root","","blood_donation") or die("Connection error");

$sql= "select * from donor_details where donor_blood='{$bg}' order by rand() limit 5";

$result=mysqli_query($conn,$sql) or die("query unsuccessful.");

 if(mysqli_num_rows($result)>0)   {

 while($row = mysqli_fetch_assoc($result)) {

   ?>

   <div class="row">

   <div class="col-lg-4 col-sm-6 portfolio-item" ><br>

   <div class="card" style="width:300px">

      <img class="card-img-top" src="image\blood_drop_logo.jpg" alt="Card image" style="width:100%;height:300px">

      <div class="card-body">

        <h3 class="card-title"><?php echo $row['donor_name']; ?></h3>

        <p class="card-text">

          <b>Blood Group : </b> <b><?php echo $row['blood_group']; ?></b><br>

          <b>Mobile No. : </b> <?php echo $row['donor_number']; ?><br>

          <b>Gender : </b><?php echo $row['donor_gender']; ?><br>

          <b>Age : </b> <?php echo $row['donor_age']; ?><br>

          <b>Address : </b> <?php echo $row['donor_address']; ?><br>

        </p>
```

```
</div>
    </div>
</div>
<?php
  }
}
  else
  {
 echo '<div class="alert alert-danger">No Donor Found For your search Blood group </div>';
  } ?>
</div>
```

## 6. *contact_*us :

```
<html>
<head>
  <meta charset="utf-8">
  <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">
  <meta name="description" content="">
  <meta name="author" content="">
  <linkrel="stylesheet"
href="https://maxcdn.bootstrapcdn.com/bootstrap/4.5.2/css/bootstrap.min.css">
  <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.5.1/jquery.min.js"></script>
  <script src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.16.0/umd/popper.min.js"></script>
  <script src="https://maxcdn.bootstrapcdn.com/bootstrap/4.5.2/js/bootstrap.min.js"></script>
</head>
<body>
```

```
<?php $active ='contact';

include 'head.php'; ?>

<?php

if(isset($_POST["send"])){

  $name=$_POST['fullname'];

$number=$_POST['contactno'];

$email=$_POST['email'];

$message=$_POST['message'];

$conn=mysqli_connect("localhost","root","","blood_donation") or die("Connection error");

$sql= "insert into contact_query (query_name,query_number,query_mail,query_message) values('{$name}','{$number}','{$email}','{$message}')";

$result=mysqli_query($conn,$sql) or die("query unsuccessful.");

  echo '<div class="alert alert-success alert_dismissible"><b><button type="button" class="close" data-dismiss="alert">&times;</button></b><b>Query Sent, We will contact you shortly. </b></div>';

}?>

<div id="page-container" style="margin-top:50px; position: relative;min-height: 84vh;">

  <div class="container">

  <div id="content-wrap" style="padding-bottom:50px;">

    <h1 class="mt-4 mb-3">Contact</h1>

    <div class="row">

      <div class="col-lg-8 mb-4">

        <h3>Send us a Message</h3>

        <form name="sentMessage"  method="post">

            <div class="control-group form-group">

                <div class="controls">
```

```
            <label>Full Name:</label>
            <input type="text" class="form-control" id="name" name="fullname" required>
            <p class="help-block"></p>
          </div>
        </div>
        <div class="control-group form-group">
          <div class="controls">
            <label>Phone Number:</label>
            <input type="tel" class="form-control" id="phone" name="contactno"  required >
          </div>
        </div>
        <div class="control-group form-group">
          <div class="controls">
            <label>Email Address:</label>
            <input type="email" class="form-control" id="email" name="email" required>
          </div>
        </div>
        <div class="control-group form-group">
          <div class="controls">
            <label>Message:</label>
            <textarea rows="10" cols="100" class="form-control" id="message" name="message"
required  maxlength="999" style="resize:none"></textarea>
          </div>
        </div>
        <button type="submit" name="send"  class="btn btn-primary">Send Message</button>
```

```
    </form>
  </div>
  <div class="col-lg-4 mb-4">
    <h2>Contact Details</h2>
    <?php
      include 'conn.php';
      $sql= "select * from contact_info";
      $result=mysqli_query($conn,$sql);
      if(mysqli_num_rows($result)>0)   {
         while($row = mysqli_fetch_assoc($result)) { ?>
    <br>
    <p>
       <h4>Address :</h4><?php echo $row['contact_address']; ?>
    </p>
    <p>
       <h4>Contact Number :</h4><?php echo $row['contact_mail']; ?>
    </p>
    <p>
       <h4>Email : </h4><a href="#"><?php echo $row['contact_phone']; ?></a>
      </a></b>
    </p>
    <?php }
   } ?>
  </div>
</div>
```

```
<!-- /.row -->
</div>
</div>
<?php include 'footer.php' ?>
</div>
</body>
</html>
```

## Admin

### 7. Dashboard :

```
<html>
<head>
  <meta charset="utf-8">
  <meta name="viewport" content="width=device-width, initial-scale=1">
  <linkrel="stylesheet"
href="https://maxcdn.bootstrapcdn.com/bootstrap/3.4.1/css/bootstrap.min.css">
  <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.5.1/jquery.min.js"></script>
  <script src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.16.0/umd/popper.min.js"></script>
  <script src="https://maxcdn.bootstrapcdn.com/bootstrap/3.4.1/js/bootstrap.min.js"></script>
<style>
#sidebar{position:relative;margin-top:-20px}
#content{position:relative;margin-left:210px}
@media screen and (max-width: 600px) {
  #content {
    position:relative;margin-left:auto;margin-right:auto;
```

```
  }
}
.block-anchor {
            color:red;
cursor: pointer;
        }
</style>
</head>
<body style="color:black;" >
<?php
  include 'conn.php';
    include 'session.php';
    if (isset($_SESSION['loggedin']) && $_SESSION['loggedin'] == true) {
    ?>
<div id="header">
<?php include 'header.php';
?>
</div>
<div id="sidebar">
<?php
$active="dashboard";
include 'sidebar.php'; ?>
</div>
<div id="content">
 <div class="content-wrapper">
```

```
  <div class="container-fluid">
   <div class="row">
    <div class="col-md-12 lg-12 sm-12">
     <h1 class="page-title">Dashboard</h1>
    </div>
   </div>
   <hr>
   <div class="row">
    <div class="col-md-12">
     <div class="row">
      <div class="col-md-3">
       <div class="panel panel-default panel-info" style="border-radius:50px;">
        <div class="panel-body panel-info bk-primary text-light" style="background-color:#D6EAF8; border-radius:50px">
         <div class="stat-panel text-center">
          <?php
           $sql =" SELECT * from donor_details ";
           $result=mysqli_query($conn,$sql) or die("query failed.");
           $row=mysqli_num_rows($result);
?>
<div class="stat-panel-number h1"><?php echo $row?></div>
          <div class="stat-panel-title text-uppercase">Blood Donors Available </div>
          <br>
           <button class="btn btn-danger" onclick="window.location.href = 'donor_list.php';">
            Full Detail <i class="fa fa-arrow-right"></i>
```

```
                </button>
</div>
            </div>
  </div>
         </div>
 <div class="col-md-3">
           <div class="panel panel-default panel-info" style="border-radius:50px;">
            <div class="panel-body panel-info bk-primary text-light" style="background-color:#ABEBC6;border-radius:50px;">
              <div class="stat-panel text-center">
               <?php
                $sql1 =" SELECT * from contact_query ";
                $result1=mysqli_query($conn,$sql1) or die("query failed.");
                $row1=mysqli_num_rows($result1);
               ?>
 <div class="stat-panel-number h1 "><?php echo $row1?></div>
               <div class="stat-panel-title text-uppercase"> All User Queries </div>
               <br>
               <button class="btn btn-danger" onclick="window.location.href = 'query.php';">
                Full Detail <i class="fa fa-arrow-right"></i>
               </button>
              </div>
             </div>
</div>
         </div>
```

```
<div class="col-md-3">
        <div class="panel panel-default panel-info" style="border-radius:50px;">
         <div class="panel-body panel-info bk-primary text-light" style="background-color:#E8DAEF ;border-radius:50px; ">
           <div class="stat-panel text-center">
            <?php
              $sql2 ="SELECT * from contact_query where query_status=2 ";
              $result2=mysqli_query($conn,$sql2) or die("query failed.");
              $row2=mysqli_num_rows($result2);
 ?>
<div class="stat-panel-number h1 "><?php echo $row2 ?></div>
           <div class="stat-panel-title text-uppercase"> Pending Queries </div>
           <br>
           <button class="btn btn-danger" onclick="window.location.href = 'pending_query.php';">
             Full Detail <i class="fa fa-arrow-right"></i>
           </button>
          </div>
         </div>
</div>
      </div>
</div>
   </div>
  </div>
 <?php
```

```
 } else {
    echo '<div class="alert alert-danger"><b> Please Login First To Access Admin Portal.</b></div>';
    ?>
    <form method="post" name="" action="login.php" class="form-horizontal">
      <div class="form-group">
        <div class="col-sm-8 col-sm-offset-4" style="float:left">
<button class="btn btn-primary" name="submit" type="submit">Go to Login Page</button>
        </div>
      </div>
    </form>
 <?php }
  ?>
</body>
</html>
```

### 8. add_donor :

```
<?php include 'session.php'; ?>
<html>
<head>
  <meta charset="utf-8">
  <meta name="viewport" content="width=device-width, initial-scale=1">
  <linkrel="stylesheet"
href="https://maxcdn.bootstrapcdn.com/bootstrap/3.4.1/css/bootstrap.min.css">
  <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.5.1/jquery.min.js"></script>
  <script src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.16.0/umd/popper.min.js"></script>
```

```
  <script src="https://maxcdn.bootstrapcdn.com/bootstrap/3.4.1/js/bootstrap.min.js"></script>
<style>
#sidebar{position:relative;margin-top:-20px}
#content{position:relative;margin-left:210px}
@media screen and (max-width: 600px) {
  #content {
    position:relative;margin-left:auto;margin-right:auto;
  }
}
</style>
</head>
<body style="color:black">
  <?php
  include 'conn.php';
    if (isset($_SESSION['loggedin']) && $_SESSION['loggedin'] == true) {
    ?>
<div id="header">
<?php $active="add"; include 'header.php';
?>
</div>
<div id="sidebar">
<?php include 'sidebar.php'; ?>
</div>
<div id="content">
  <div class="content-wrapper">
```

```
  <div class="container-fluid">
    <div class="row">
      <div class="col-md-12 lg-12 sm-12">
 <h1 class="page-title">Add Donor</h1>
      </div>
    </div>
    <hr>
    <form name="donor" action="save_donor_data.php" method="post">
    <div class="row">
    <div class="col-lg-4 mb-4"><br>
    <div class="font-italic">Full Name<span style="color:red">*</span></div>
    <div><input type="text" name="fullname" class="form-control" required></div>
    </div>
    <div class="col-lg-4 mb-4"><br>
    <div class="font-italic">Mobile Number<span style="color:red">*</span></div>
    <div><input type="text" name="mobileno" class="form-control" required></div>
    </div>
    <div class="col-lg-4 mb-4"><br>
    <div class="font-italic">Email Id</div>
    <div><input type="email" name="emailid" class="form-control"></div>
    </div>
    </div>
<div class="row">
    <div class="col-lg-4 mb-4"><br>
    <div class="font-italic">Age<span style="color:red">*</span></div>
```

```
    <div><input type="text" name="age" class="form-control" required></div>
    </div>
 <div class="col-lg-4 mb-4"><br>
    <div class="font-italic">Gender<span style="color:red">*</span></div>
    <div><select name="gender" class="form-control" required>
    <option value="">Select</option>
    <option value="Male">Male</option>
    <option value="Female">Female</option>
    </select>
    </div>
   </div>
    <div class="col-lg-4 mb-4"><br>
    <div class="font-italic">Blood Group<span style="color:red">*</span></div>
    <div><select name="blood" class="form-control" required>
    <option value=""selected disabled>Select</option>
    <?php
      include 'conn.php';
      $sql= "select * from blood";
      $result=mysqli_query($conn,$sql) or die("query unsuccessful.");
    while($row=mysqli_fetch_assoc($result)){
     ?>
     <option value=" <?php echo $row['blood_id'] ?>"> <?php echo $row['blood_group'] ?>
</option>
   <?php } ?>
    </select>
    </div>
```

```
    </div>
</div>
    <br>
    <div class="row">
    <div class="col-lg-4 mb-4">
    <div class="font-italic">Address<span style="color:red">*</span></div>
    <div><textarea class="form-control" name="address" required></textarea></div></div>
   </div> <br>
    <div class="row">
      <div class="col-lg-4 mb-4">
      <div><input type="submit" name="submit" class="btn btn-primary" value="Submit" style="cursor:pointer" onclick="popup()"></div>
      </div>
    </div>
   </form>
</div>
    </div>
    </div>
    <?php
   } else {
      echo '<div class="alert alert-danger"><b> Please Login First To Access Admin Portal.</b></div>';
      ?>
      <form method="post" name="" action="login.php" class="form-horizontal">
        <div class="form-group">
          <div class="col-sm-8 col-sm-offset-4" style="float:left">
```

```
<button class="btn btn-primary" name="submit" type="submit">Go to Login Page</button>
          </div>
         </div>
       </form>
   <?php }
    ?>
    <script>
    function popup() {
      alert("Data added Successfully.");
    }
    </script>
</body>
</html>
```

**9. donor_list :**

```
<html>
<head>
  <meta charset="utf-8">
  <meta name="viewport" content="width=device-width, initial-scale=1">
  <linkrel="stylesheet"
href="https://maxcdn.bootstrapcdn.com/bootstrap/3.4.1/css/bootstrap.min.css">
  <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.5.1/jquery.min.js"></script>
  <script src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.16.0/umd/popper.min.js"></script>
  <script src="https://maxcdn.bootstrapcdn.com/bootstrap/3.4.1/js/bootstrap.min.js"></script>
<style>
#sidebar{position:relative;margin-top:-20px}
```

```
#content{position:relative;margin-left:210px}
@media screen and (max-width: 600px) {
 #content {
   position:relative;margin-left:auto;margin-right:auto;
 }
}
 #he{
     font-size: 14px;
     font-weight: 600;
     text-transform: uppercase;
     padding: 3px 7px;
     color: #fff;
     text-decoration: none;
     border-radius: 3px;
     align:center
 }
</style>
</head>
<?php
include 'conn.php';
 include 'session.php';
 if (isset($_SESSION['loggedin']) && $_SESSION['loggedin'] == true) {
 ?>
<body style="color:black">
<div id="header">
```

```
<?php include 'header.php';
?>
</div>
<div id="sidebar">
<?php $active="list"; include 'sidebar.php'; ?>
</div>
<div id="content" >
  <div class="content-wrapper">
    <div class="container-fluid">
      <div class="row">
        <div class="col-md-12 lg-12 sm-12">
 <h1 class="page-title">Donor List</h1>
 </div>
 </div>
      <hr>
      <?php
        include 'conn.php';
  $limit = 10;
        if(isset($_GET['page'])){
          $page = $_GET['page'];
        }else{
          $page = 1;
        }
        $offset = ($page - 1) * $limit;
        $count=$offset+1;
```

```
$sql= "select * from donor_details join blood where donor_details.donor_blood=blood.blood_id LIMIT {$offset},{$limit}";
$result=mysqli_query($conn,$sql);
if(mysqli_num_rows($result)>0)  {
?>
<div class="table-responsive">
<table class="table table-bordered" style="text-align:center">
<thead style="text-align:center">
<th style="text-align:center">S.no</th>
<th style="text-align:center">Name</th>
<th style="text-align:center">Mobile Number</th>
<th style="text-align:center">Email Id</th>
<th style="text-align:center">Age</th>
<th style="text-align:center">Gender</th>
<th style="text-align:center">Blood Group</th>
<th style="text-align:center">Address</th>
<th style="text-align:center">Action</th>
</thead>
<tbody>
<?php
while($row = mysqli_fetch_assoc($result)) { ?>
<tr>
<td><?php echo $count++; ?></td>
<td><?php echo $row['donor_name']; ?></td>
<td><?php echo $row['donor_number']; ?></td>
```

```
                <td><?php echo $row['donor_mail']; ?></td>
                <td><?php echo $row['donor_age']; ?></td>
                <td><?php echo $row['donor_gender']; ?></td>
                  <td><?php echo $row['blood_group']; ?></td>
                  <td><?php echo $row['donor_address']; ?></td>
                  <td id="he" style="width:100px">
                  <a          style="background-color:aqua"          href='delete.php?id=<?php          echo
$row['donor_id']; ?>'> Delete </a>
               </td>
              </tr>
            <?php } ?>
          </tbody>
      </table>
    </div>
    <?php } ?>
<div class="table-responsive"style="text-align:center;align:center">
    <?php
    $sql1 = "SELECT * FROM donor_details";
    $result1 = mysqli_query($conn, $sql1) or die("Query Failed.");
 if(mysqli_num_rows($result1) > 0){
 $total_records = mysqli_num_rows($result1);
$total_page = ceil($total_records / $limit);
echo '<ul class="pagination admin-pagination">';
      if($page > 1){
echo '<li><a href="donor_list.php?page='.($page - 1).'">Prev</a></li>';
```

```
    }
    for($i = 1; $i <= $total_page; $i++){
      if($i == $page){
        $active = "active";
      }else{
        $active = "";
      }
      echo '<li class="'.$active.'"><a href="donor_list.php?page='.$i.'">'.$i.'</a></li>';
    }
    if($total_page > $page){
      echo '<li><a href="donor_list.php?page='.($page + 1).'">Next</a></li>';
    }
 echo '</ul>';
   }
   ?>
 </div>
 </div>
</div>
</div>
<?php }
  else {
     echo '<div class="alert alert-danger"><b> Please Login First To Access Admin Portal.</b></div>';
     ?>
 <form method="post" name="" action="login.php" class="form-horizontal">
```

```
        <div class="form-group">
          <div class="col-sm-8 col-sm-offset-4" style="float:left">
<button class="btn btn-primary" name="submit" type="submit">Go to Login Page</button>
          </div>
        </div>
      </form>
  <?php }
 ?>
</body>
</html>
```

**10. Query :**

```
<html>
<head>
 <meta charset="utf-8">
  <meta name="viewport" content="width=device-width, initial-scale=1">
  <linkrel="stylesheet"
href="https://maxcdn.bootstrapcdn.com/bootstrap/3.4.1/css/bootstrap.min.css">
  <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.5.1/jquery.min.js"></script>
  <script src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.16.0/umd/popper.min.js"></script>
  <script src="https://maxcdn.bootstrapcdn.com/bootstrap/3.4.1/js/bootstrap.min.js"></script>
<style>
#sidebar{position:relative;margin-top:-20px}
#content{position:relative;margin-left:210px}
@media screen and (max-width: 600px) {
```

```
 #content {
   position:relative;margin-left:auto;margin-right:auto;
 }
}
 #he{
     font-size: 14px;
     font-weight: 600;
     text-transform: uppercase;
     padding: 3px 7px;
     color: #fff;
     text-decoration: none;
     border-radius: 3px;
     align:center
 }
</style>
</head>
<?php
include 'conn.php';
 include 'session.php';
 if (isset($_SESSION['loggedin']) && $_SESSION['loggedin'] == true) {
 ?>
<body style="color:black">
<div id="header">
<?php include 'header.php';
?>
```

```
</div>
<div id="sidebar">
<?php $active="query"; include 'sidebar.php'; ?>
</div>
<div id="content" >
  <div class="content-wrapper">
    <div class="container-fluid">
      <div class="row">
        <div class="col-md-12 lg-12 sm-12">
<h1 class="page-title">User Query</h1>
  </div>
  </div>
      <hr>
      <script>
      function clickme(){
        if(confirm("Do you really Want to Read ?"))
        {
            document.getElementById("demo").innerHTML = "Read";
            <?php
            echo '<div class="alert alert-info alert_dismissible"><b><button type="button" class="close" data-dismiss="alert">&times;</button></b><b>Pending Request "Read".</b></div>';
 $que_id = $_GET['id'];
             $sql1="update contact_query set query_status='1' where  query_id={$que_id}";
              $result=mysqli_query($conn,$sql1);
            ?>
```

```
        }
      }
 </script>
<?php
        include 'conn.php';
$limit = 10;
          if(isset($_GET['page'])){
            $page = $_GET['page'];
          }else{
            $page = 1;
          }
          $offset = ($page - 1) * $limit;
          $count=$offset+1;
        $sql= "select * from contact_query LIMIT {$offset},{$limit}";
        $result=mysqli_query($conn,$sql);
        if(mysqli_num_rows($result)>0)   {
       ?>
<div class="table-responsive">
      <table class="table table-bordered" style="text-align:center">
          <thead style="text-align:center">
          <th style="text-align:center">S.no</th>
          <th style="text-align:center">Name</th>
          <th style="text-align:center">Mobile Number</th>
          <th style="text-align:center">Email Id</th>
          <th style="text-align:center">Message</th>
```

```
        <th style="text-align:center">Posting Date</th>
        <th style="text-align:center">Status</th>
        <th style="text-align:center">Action</th>
        </thead>
        <tbody>
          <?php
          while($row = mysqli_fetch_assoc($result)) { ?>
        <tr>
<td><?php echo $count++; ?></td>
                <td><?php echo $row['query_name']; ?></td>
                <td><?php echo $row['query_number']; ?></td>
                <td><?php echo $row['query_mail']; ?></td>
                <td><?php echo $row['query_message']; ?></td>
                <td><?php echo $row['query_date']; ?></td>
                <?php if($row['query_status']==1)
{
?><td>Read<br></td>
<?php } else {?>
<td><a href="query.php?id=<?php echo $row['query_id'];?>" onclick="clickme()"><b
id="demo">Pending</b></a><br>
</td>
<?php } ?>
                <td id="he" style="width:100px">
                <a style="background-color:aqua" href='delete_query.php?id=<?php echo
$row['query_id']; ?>'> Delete </a>
    </td>
```

```
            </tr>
           <?php } ?>
          </tbody>
      </table>
    </div>
    <?php } ?>
<div class="table-responsive"style="text-align:center;align:center">
        <?php
        $sql1 = "SELECT * FROM contact_query";
        $result1 = mysqli_query($conn, $sql1) or die("Query Failed.");
  if(mysqli_num_rows($result1) > 0){
$total_records = mysqli_num_rows($result1);
 $total_page = ceil($total_records / $limit);
 echo '<ul class="pagination admin-pagination">';
          if($page > 1){
            echo '<li><a href="query.php?page='.($page - 1).'">Prev</a></li>';
          }
          for($i = 1; $i <= $total_page; $i++){
            if($i == $page){
              $active = "active";
            }else{
              $active = "";
            }
            echo '<li class="'.$active.'"><a href="query.php?page='.$i.'">'.$i.'</a></li>';
  }
```

```
        if($total_page > $page){
          echo '<li><a href="query.php.php?page='.($page + 1).'">Next</a></li>';
        }
 echo '</ul>';
       }
       ?>
</div>
     </div>
   </div>
<?php
  } else {
     echo '<div class="alert alert-danger"><b> Please Login First To Access Admin Portal.</b></div>';
     ?>
     <form method="post" name="" action="login.php" class="form-horizontal">
       <div class="form-group">
         <div class="col-sm-8 col-sm-offset-4" style="float:left">
<button class="btn btn-primary" name="submit" type="submit">Go to Login Page</button>
         </div>
       </div>
     </form>
  <?php }
   ?>
</body>
   </html>
```

## 11. update_page_details :

```
<html>
<head>
  <meta charset="utf-8">
  <meta name="viewport" content="width=device-width, initial-scale=1">
  <linkrel="stylesheet"
href="https://maxcdn.bootstrapcdn.com/bootstrap/3.4.1/css/bootstrap.min.css">
  <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.5.1/jquery.min.js"></script>
  <script src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.16.0/umd/popper.min.js"></script>
  <script src="https://maxcdn.bootstrapcdn.com/bootstrap/3.4.1/js/bootstrap.min.js"></script>
  <script type="text/javascript" src="nicEdit.js"></script>
<style>
#sidebar{position:relative;margin-top:-20px}
#content{position:relative;margin-left:210px}
@media screen and (max-width: 600px) {
  #content {
    position:relative;margin-left:auto;margin-right:auto;
  }
  #area1, #area4{
    width: 70vw;
    min-height: 50vh;
    font-family: tahoma;
  }
nicEdit-panel > div > * {
    opacity: 1 !important;
```

```
  }
.nicEdit-buttonContain {
      padding: .5em;
  }
  .nicEdit-panelContain{
  }
.nicEdit-selectContain{
  margin-top: 8px;
  padding:.5em
  }
 .nicEdit-selectTxt{
       font-family: Tahoma !important;
       font-size: 12px !important;
   }
.nicEdit-main{
    outline: 0;
  }
}
</style>
</head>
<body style="color:black">
 <?php
  include 'conn.php';
    include 'session.php';
    if (isset($_SESSION['loggedin']) && $_SESSION['loggedin'] == true) {
```

```
   ?>
<div id="header">
<?php include 'header.php';
?>
</div>
<div id="sidebar">
<?php $active ="";
include 'sidebar.php'; ?>
</div>
<div id="content">
  <div class="content-wrapper">
    <div class="container-fluid">
      <div class="row">
        <div class="col-md-12 lg-12 sm-12">
 <h1 class="page-title">Update Page Details</h1>
        </div>
      </div>
      <hr>
      <div class="row">
        <div class="col-md-10">
          <div class="panel panel-default">
            <div class="panel-heading">Page Details</div>
            <div class="panel-body">
<form name="updata_page"  method="post">
              <div class="hr-dashed"></div>
```

```
<div class="form-group">
            <label class="col-sm-4 control-label">Selected Page : </label>
             <?php
             include 'conn.php';
          switch($_GET['type'])
          {
              case "aboutus" :
                   echo "About US";
                   break;
              case "donor" :
                   echo "Why Donate Blood";
                   break;
              case "needforblood" :
                   echo "The Need For Blood";
                   break;
              case "bloodtips" :
                   echo "Blood Tips";
                   break;
              case "whoyouhelp" :
                   echo "Why you could Help";
                   break;
              case "bloodgroups" :
                   echo "Blood Groups";
                   break;
              case "universal" :
```

```
                    echo "Universal Donors And Recepients";
                    break;
            }
 ?>
    </div>
            <div class="form-group">
                <label class="col-sm-4 control-label">Page Details: </label>
                <div class="col-sm-4">
                  <div id="sample">
<textarea cols="60" rows="10" id="area4" name="data">
                </textarea>
</div>
      </div>
    </div>
 <div class="form-group">
      <div class="col-sm-8 col-sm-offset-4"><br>
<button class="btn btn-primary" name="submit" type="submit">Update</button>
      </div>
    </div>
</form>
 </div>
   </div>
   </div>
</div>
 <?php if(isset($_POST['submit']))
```

```
  {
   $type=$_GET['type'];
   $data=$_POST['data'];
   $conn=mysqli_connect("localhost","root","","blood_donation") or die("Connection error");
   $sql= "update pages set page_data='{$data}'where page_type='{$type}'";
   $result=mysqli_query($conn,$sql) or die("query unsuccessful.");
  echo '<div class="alert alert-success"><b>Page Data Updated Successfully.</b></div>';
  }
?>
</div>
      </div>
     </div>
 <?php
} else {
   echo '<div class="alert alert-danger"><b> Please Login First To Access Admin Portal.</b></div>';
   ?>
   <form method="post" name="" action="login.php" class="form-horizontal">
     <div class="form-group">
       <div class="col-sm-8 col-sm-offset-4" style="float:left">
 <button class="btn btn-primary" name="submit" type="submit">Go to Login Page</button>
       </div>
     </div>
   </form>
<?php }
```

```
 ?>
</body>
</html>
```

## 12. update_contact :

```
<html>
<head>
<meta charset="utf-8">
 <meta name="viewport" content="width=device-width, initial-scale=1">
 <linkrel="stylesheet"
href="https://maxcdn.bootstrapcdn.com/bootstrap/3.4.1/css/bootstrap.min.css">
 <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.5.1/jquery.min.js"></script>
 <script src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.16.0/umd/popper.min.js"></script>
 <script src="https://maxcdn.bootstrapcdn.com/bootstrap/3.4.1/js/bootstrap.min.js"></script>
<style>
#sidebar{position:relative;margin-top:-20px}
#content{position:relative;margin-left:210px}
@media screen and (max-width: 600px) {
 #content {
   position:relative;margin-left:auto;margin-right:auto;
 }
}
</style>
</head>
<body style="color:black">
```

```
<?php
  include 'conn.php';
    include 'session.php';
    if (isset($_SESSION['loggedin']) && $_SESSION['loggedin'] == true) {
    ?>
<div id="header">
<?php include 'header.php';
?>
</div>
<div id="sidebar">
<?php
$active="contact";
 include 'sidebar.php'; ?>
</div>
<div id="content">
  <div class="content-wrapper">
    <div class="container-fluid">
      <div class="row">
        <div class="col-md-12 lg-12 sm-12">
<h1 class="page-title">Update Contact Info</h1>
        </div>
      </div>
      <hr>
      <?php if(isset($_POST['update']))
      {
```

```
$address=$_POST['address'];
$number=$_POST['email'];
$email=$_POST['contactno'];
$conn=mysqli_connect("localhost","root","","blood_donation") or die("Connection error");
$sql= "update contact_info set contact_address='{$address}', contact_mail='{$email}', contact_phone='{$number}' where contact_id='1'";
$result=mysqli_query($conn,$sql) or die("query unsuccessful.");
echo '<div class="alert alert-success"><b>Contact Details Updated Successfully.</b></div>';
mysqli_close($conn);
}
?>
<div class="row">
<div class="col-md-10">
<div class="panel panel-default">
<div class="panel-heading">Contact Details</div>
<div class="panel-body">
<form method="post"  name="change_contact" class="form-horizontal">
<div class="form-group">
<label class="col-sm-4 control-label"> Address</label>
<div class="col-sm-8">
<textarea class="form-control" name="address" id="address" required></textarea>
</div>
</div>
<div class="form-group">
<label class="col-sm-4 control-label"> Email id</label>
<div class="col-sm-8">
```

```
                <input type="email" class="form-control" name="email" id="email" value="" required>
                </div>
              </div>
    <div class="form-group">
                <label class="col-sm-4 control-label"> Contact Number </label>
                <div class="col-sm-8">
                 <input type="text" class="form-control" value="" name="contactno" id="contactno" required>
                </div>
              </div>
<div class="hr-dashed"></div>
 <div class="form-group">
                <div class="col-sm-8 col-sm-offset-4">
 <button class="btn btn-primary" name="update" type="submit">Update</button>
                </div>
              </div>
</form>
</div>
         </div>
       </div>
  </div>
 </div>
     </div>
   </div>
  <?php
```

```
 } else {
    echo '<div class="alert alert-danger"><b> Please Login First To Access Admin Portal.</b></div>';
    ?>
    <form method="post" name="" action="login.php" class="form-horizontal">
      <div class="form-group">
        <div class="col-sm-8 col-sm-offset-4" style="float:left">
<button class="btn btn-primary" name="submit" type="submit">Go to Login Page</button>
        </div>
      </div>
    </form>
 <?php }
  ?>
</body>
</html>
```

# 13. LIST OF TABLES :

**Table name:** admin_info

**Primary key:** admin_id

**Description:** This table store the login information.

| Field Name | Data Type | Size | Constraints | Description |
|---|---|---|---|---|
| admin_id | int | 10 | Primary key | To store login id |
| admin_name | varchar | 50 | Not null | To store the admin name |
| admin_username | varchar | 50 | Not null | To store the admin username |
| admin_password | varchar | 50 | Not null | To store the admin password |

**Table name:** blood

**Primary key:** blood_id

**Description:** This table store the blood groups

| Field Name | Data Type | Size | Constraints | Description |
|---|---|---|---|---|
| blood_id | int | 11 | Primary key | To store the blood id |

| | | | | |
|---|---|---|---|---|
| blood_group | varchar | 10 | Not null | To store the blood groups |

**Table name:** contact_info

**Primary key:** contact_id

**Description:** This table store the contact information.

| Field Name | Data Type | Size | Constraints | Description |
|---|---|---|---|---|
| contact_id | int | 11 | Primary key | To store the contact id |
| contact_address | varchar | 100 | Not null | To store the contact address |
| contact_phone | bigint | 100 | Not null | To store the contact number |
| contact_mail | varchar | 50 | Not null | To store the contact email |

**Table name:** contact_query

**Primary key:** query_id

**Description:** This table store the query information.

| Field Name | Data Type | Size | Constraints | Description |
|---|---|---|---|---|
| query_id | int | 11 | Primary key | To store the query id |

| | | | | |
|---|---|---|---|---|
| query_name | varchar | 10 | Not null | To store the query name |
| query_number | char | 11 | Not null | To store the query number |
| query_mail | varchar | 120 | Not null | To store the query email |
| query_message | longtext | | Not null | To store the query message |
| query_date | timestamp | | Not null | To store the query date |
| query_status | int | 11 | Null | To store the query status |

**Table name:** donor_details

**Primary key:** donor_id

**Description:** This table store the donor information.

| Field Name | Data Type | Size | Constraints | Description |
|---|---|---|---|---|
| donor_id | int | 11 | Primary key | To store the donor id |
| donor_name | varchar | 50 | Not null | To store the donor name |
| donor_number | char | 10 | Not null | To store the donor number |

| | | | | |
|---|---|---|---|---|
| donor_mail | varchar | 120 | Not null | To store the donor email |
| donor_age | int | 10 | Not null | To store the donor age |
| donor_gender | varchar | 10 | Not null | To store the donor gender |
| donor_blood | varchar | 11 | Not null | To store the donor bloodgroup |
| donor_address | varchar | 50 | Not null | To store the donor address |

**Table name:** pages

**Primary key:** page_id

**Description:** This table store the page information.

| Field Name | Data Type | Size | Constraints | Description |
|---|---|---|---|---|
| page_id | int | 11 | foreign key | To store page id |
| page_name | varchar | 255 | Not null | To store the page name |
| page_type | varchar | 50 | Null | To store the page type information |
| page_data | longtext | | Not null | To store the page data |

**Table name:** query_stat

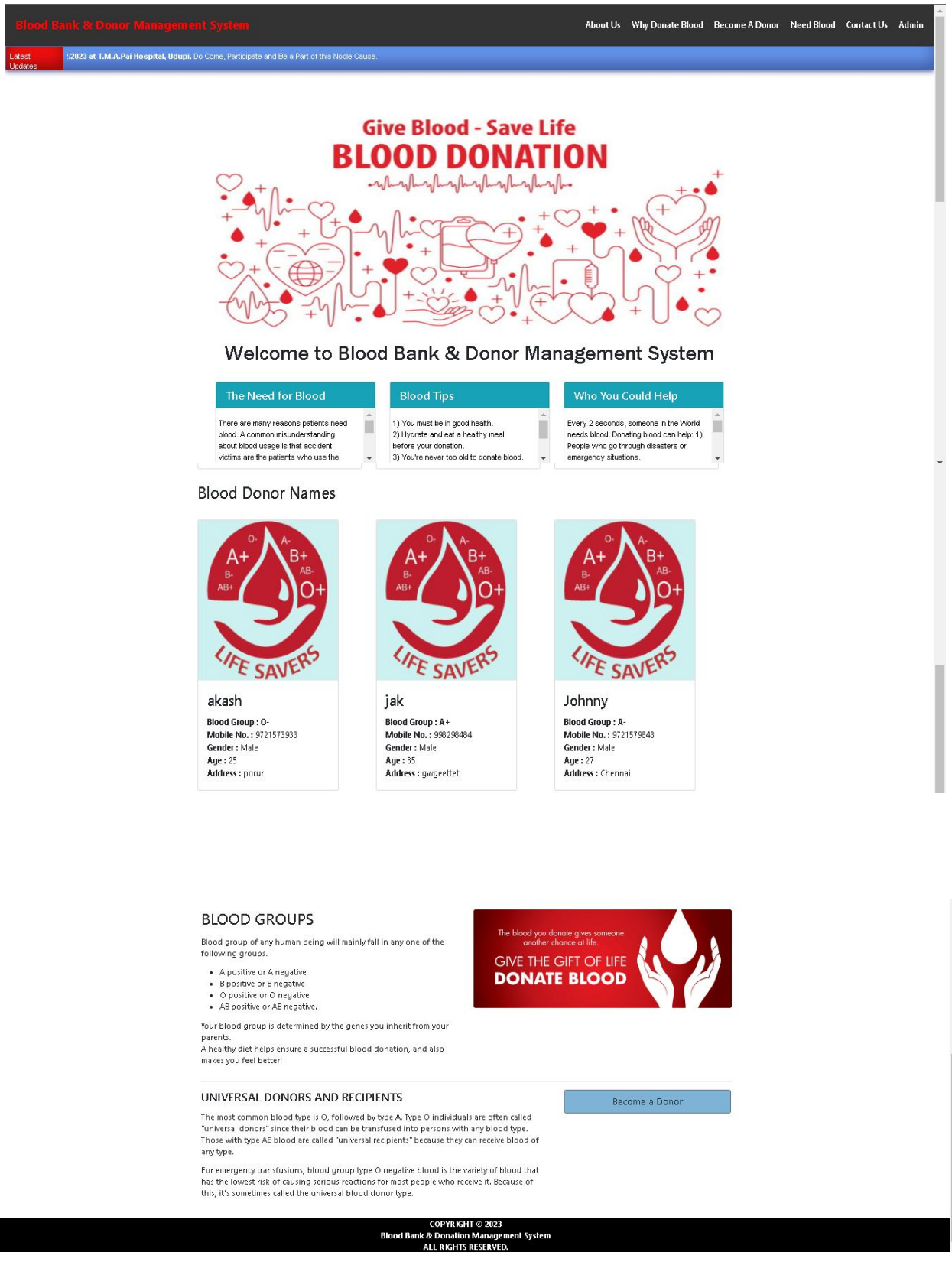
**Primary key:** id

**Description:** This table store the query status

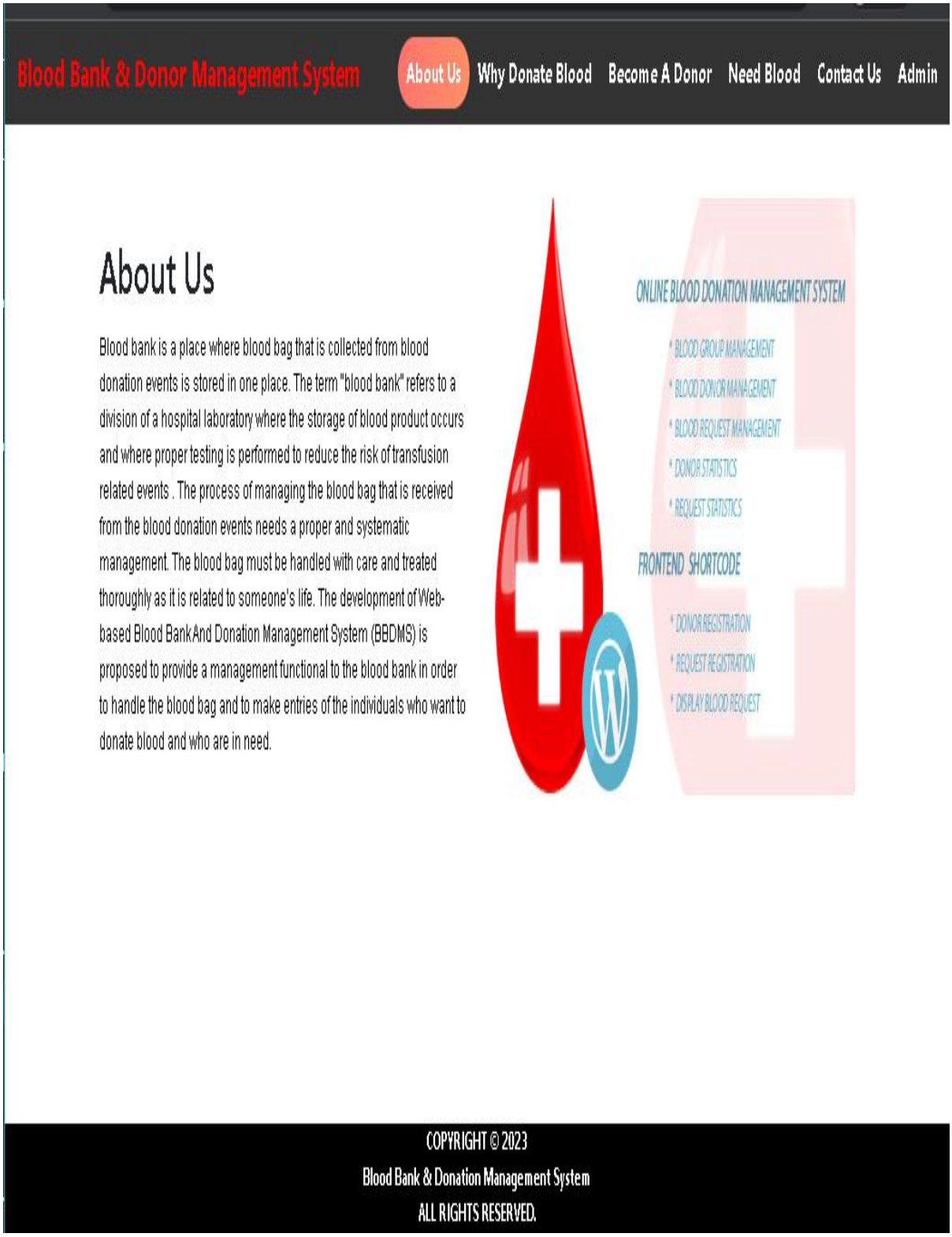
| Field Name | Data Type | Size | Constraints | Description |
|---|---|---|---|---|
| id | int | 11 | Primary key | To store the query id |
| query_type | varchar | 45 | Not null | To store the query status |

# 14. PROJECT SNAPSHOTS

## (i) home.php (Home page of the this project): -

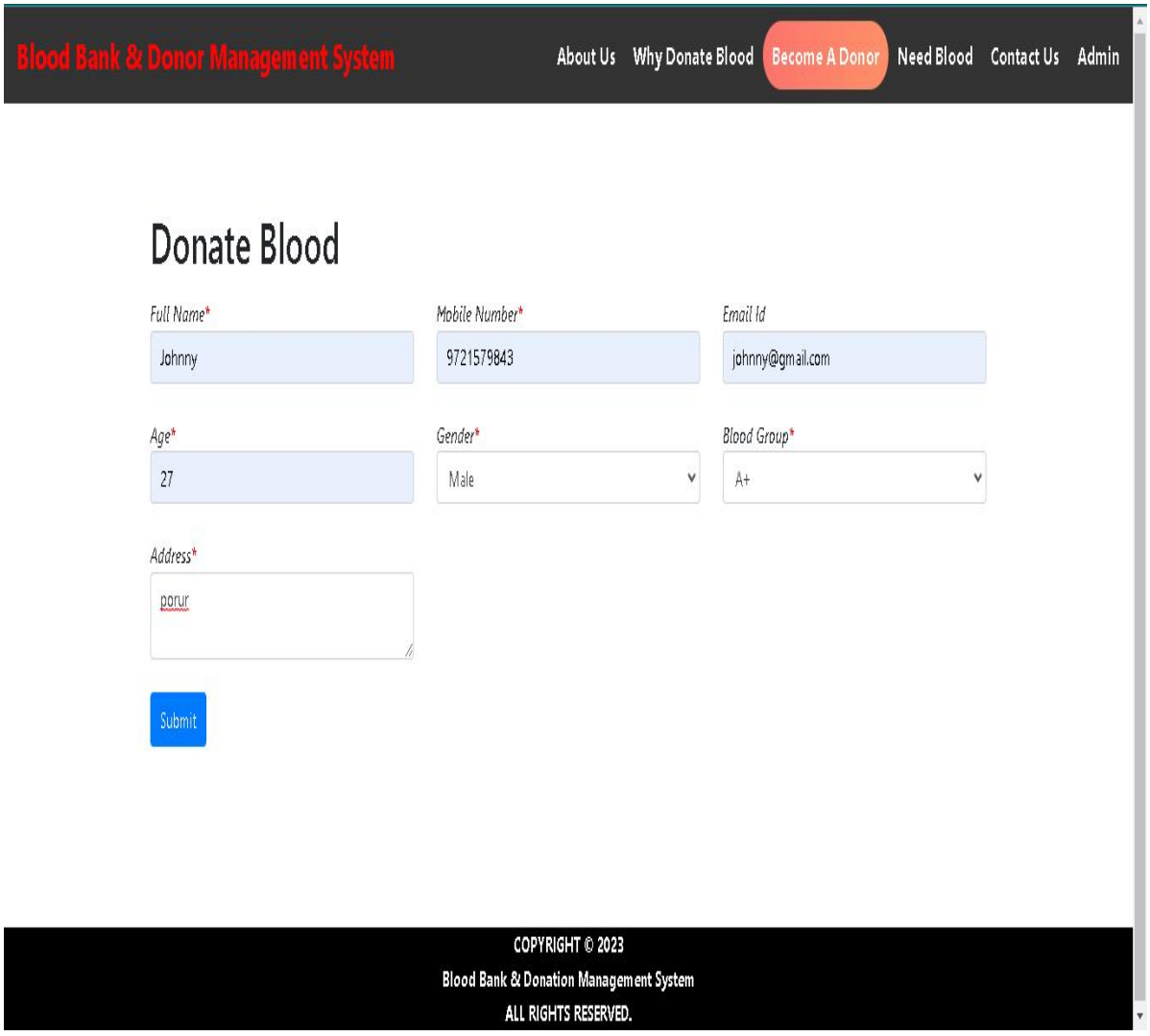


**(ii) about_us.php (About us page) :-**

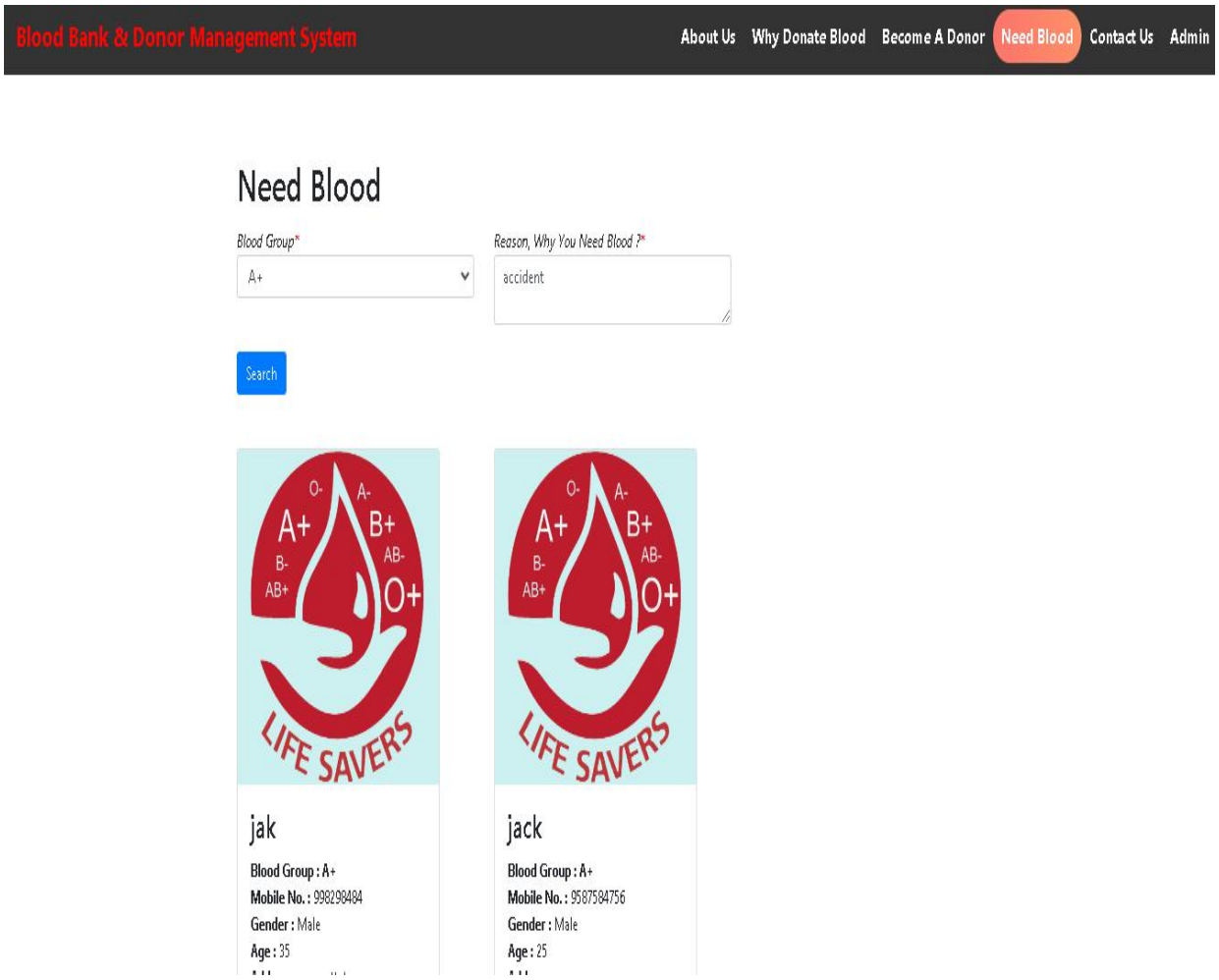


## (iii) Why_donate_blood.php

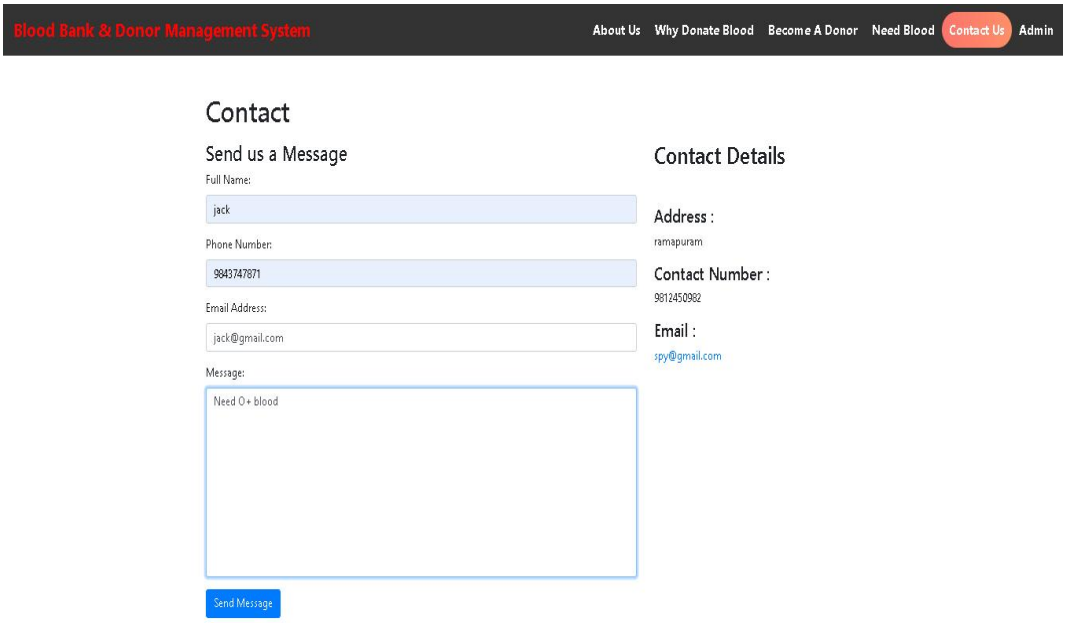
**(iv) donate_blood.php**



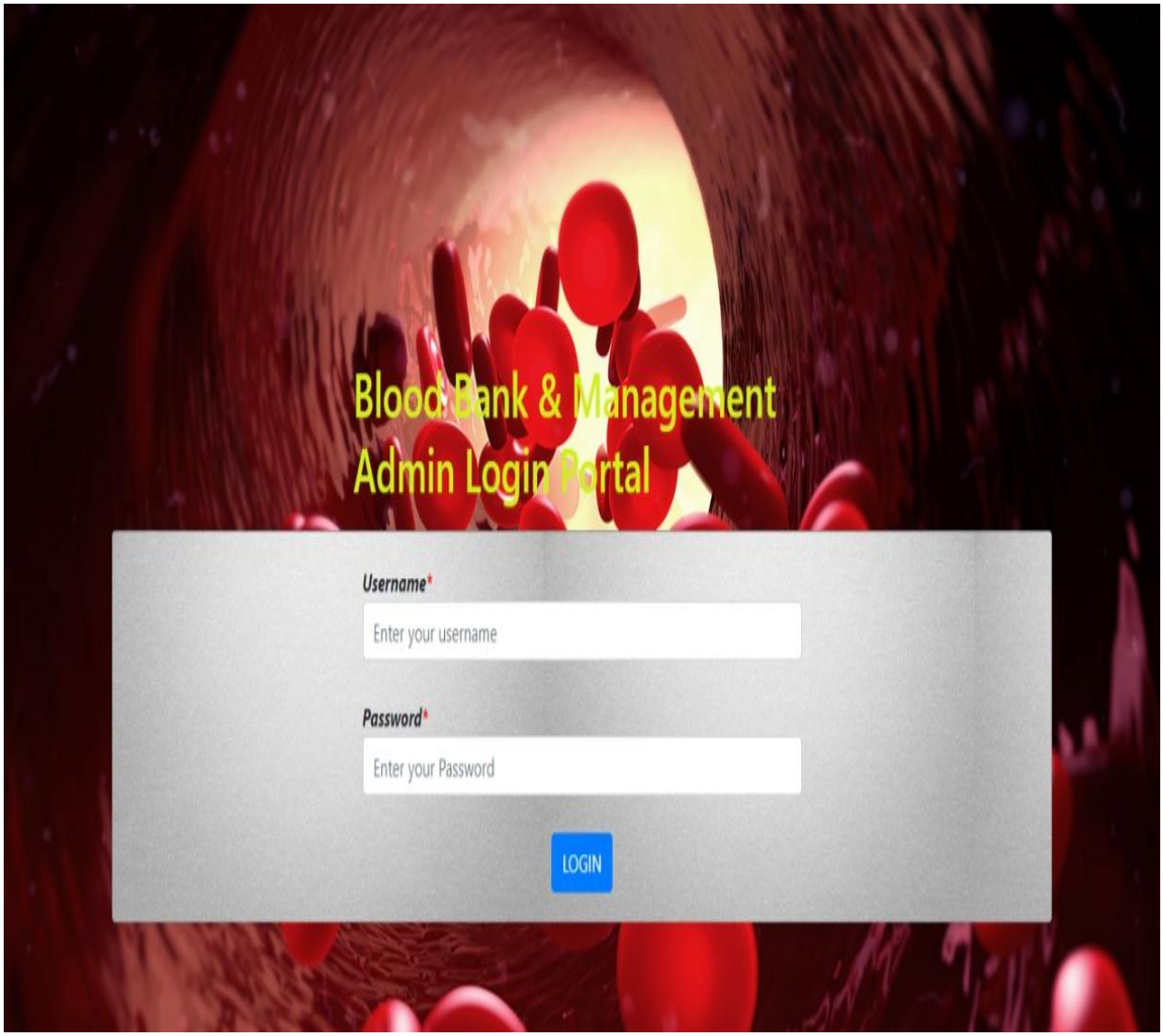
**(v) need_blood.php**



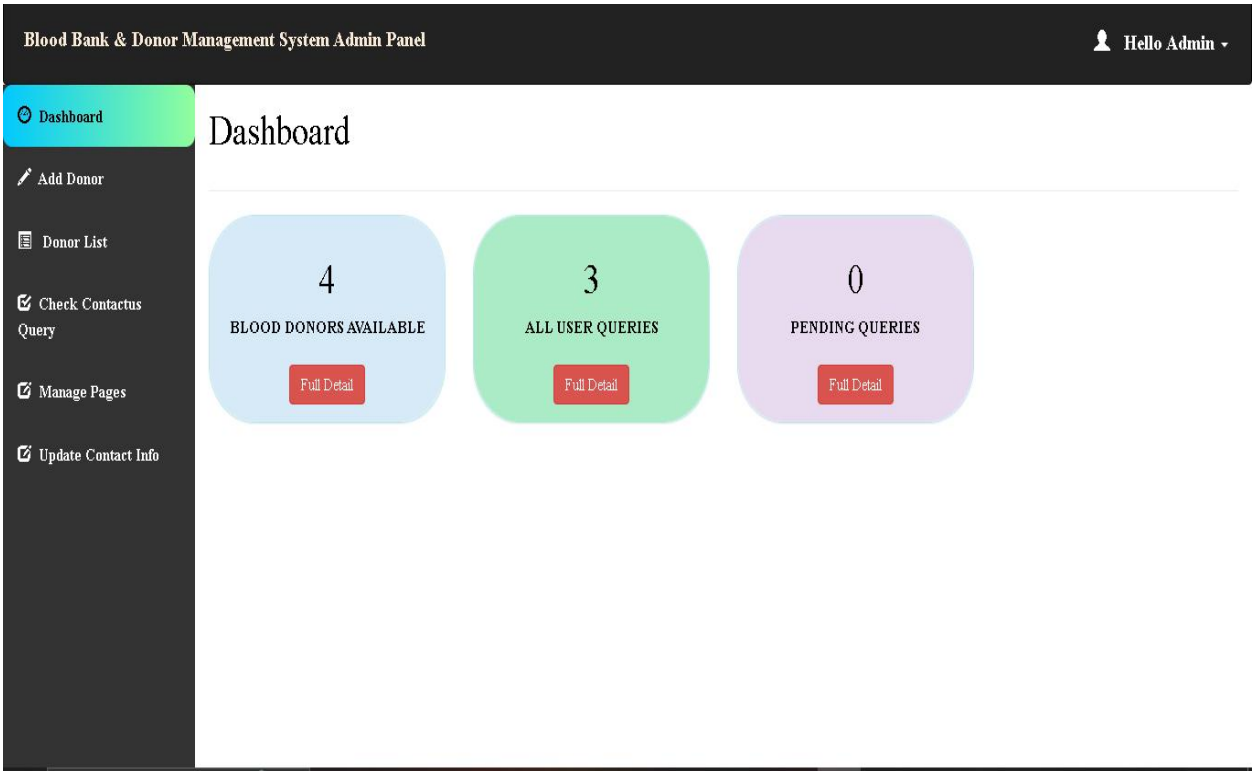
## (vi) contact_us.php



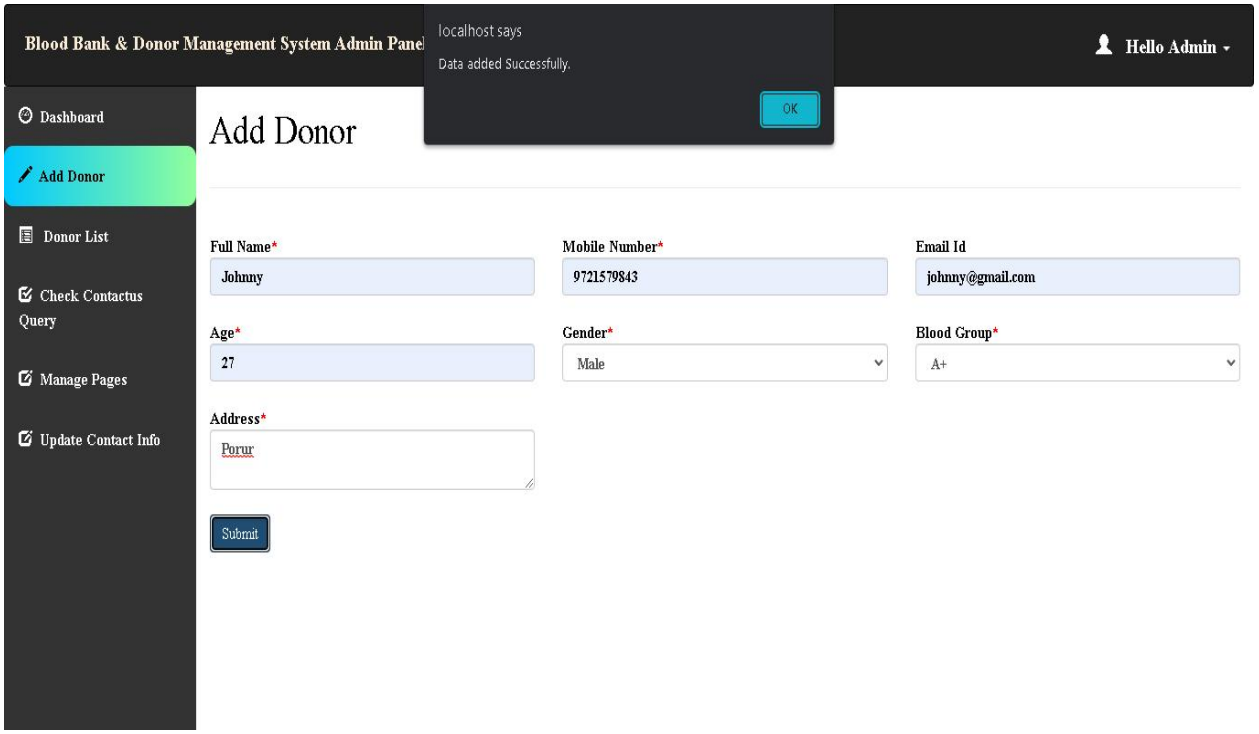
**(vii) admin/login.php (Administrator's Login Page): -**



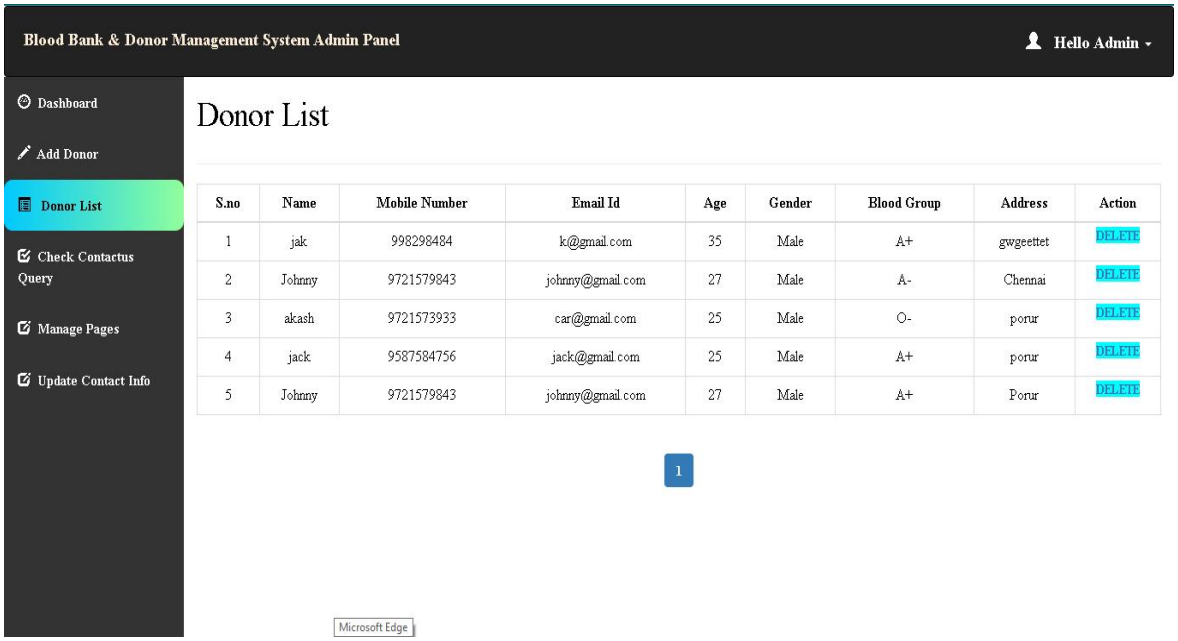
## (viii) admin/dashboard.php



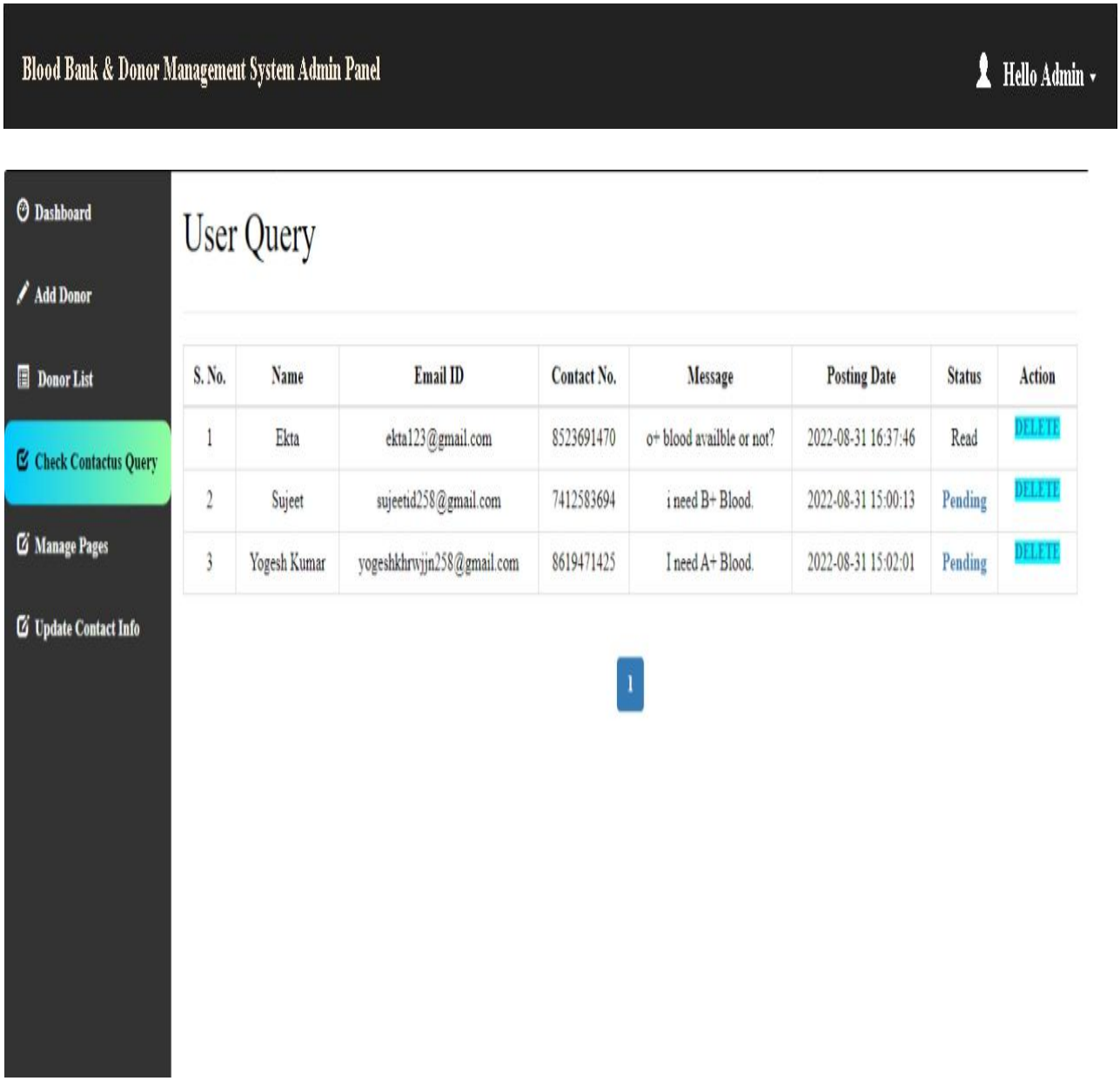
## (ix )admin/add_donor.php



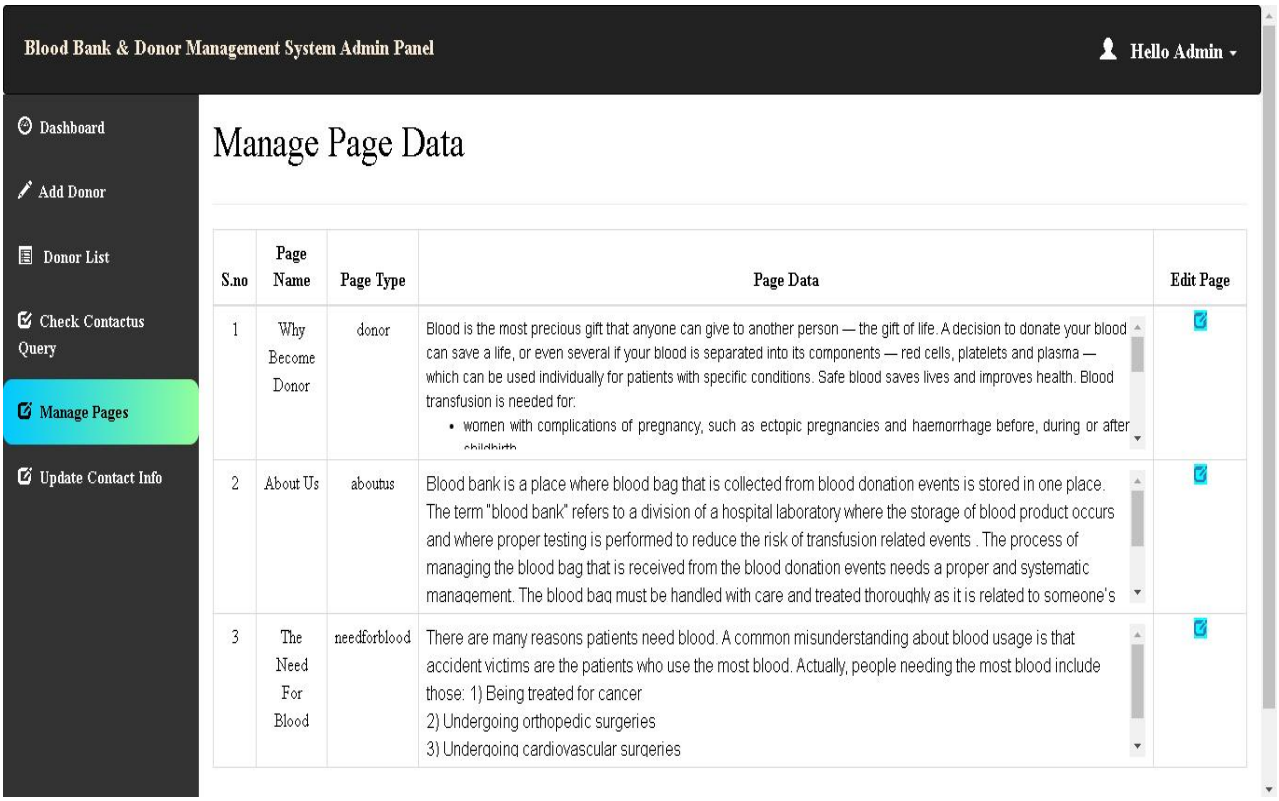
## (ix) admin/donor_list.php



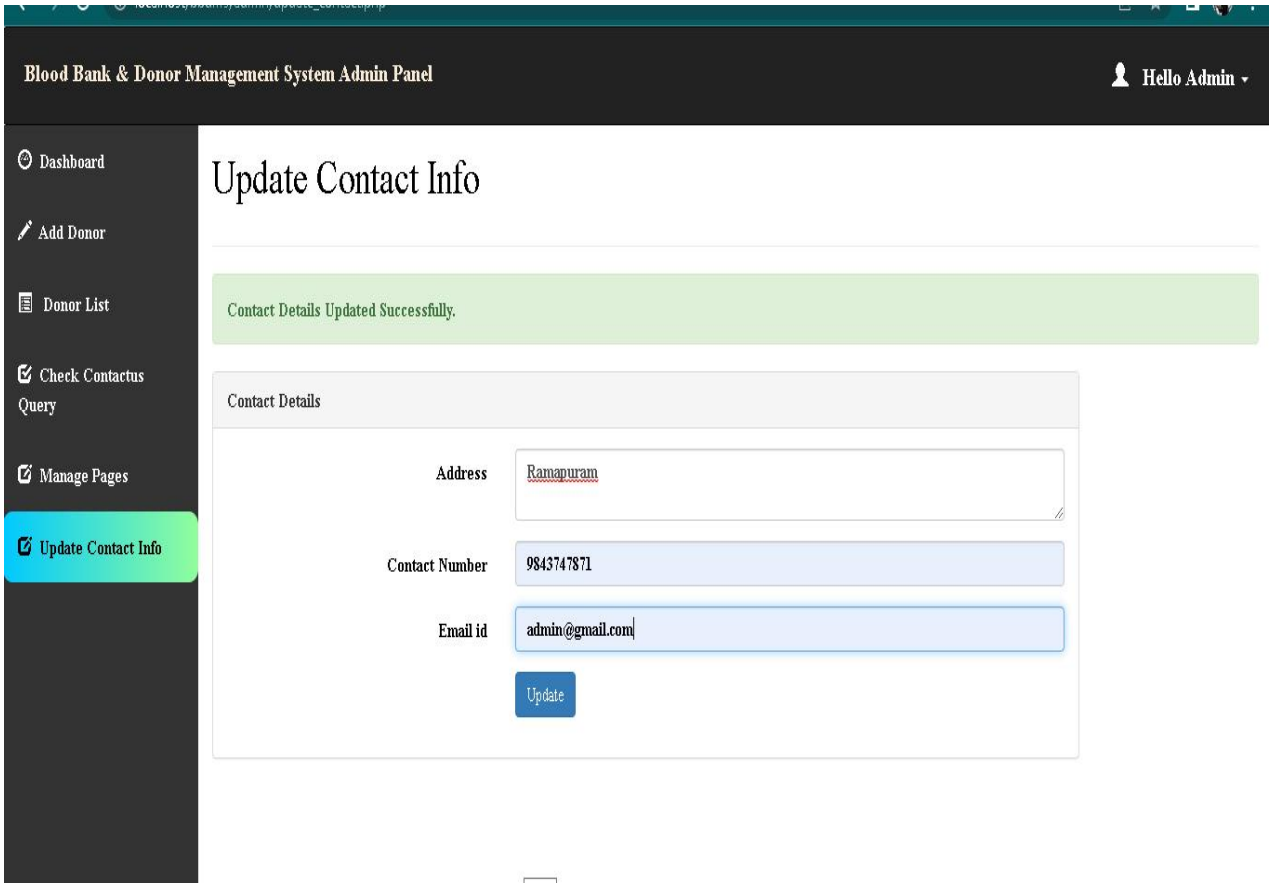
**(xi) admin/query.php**



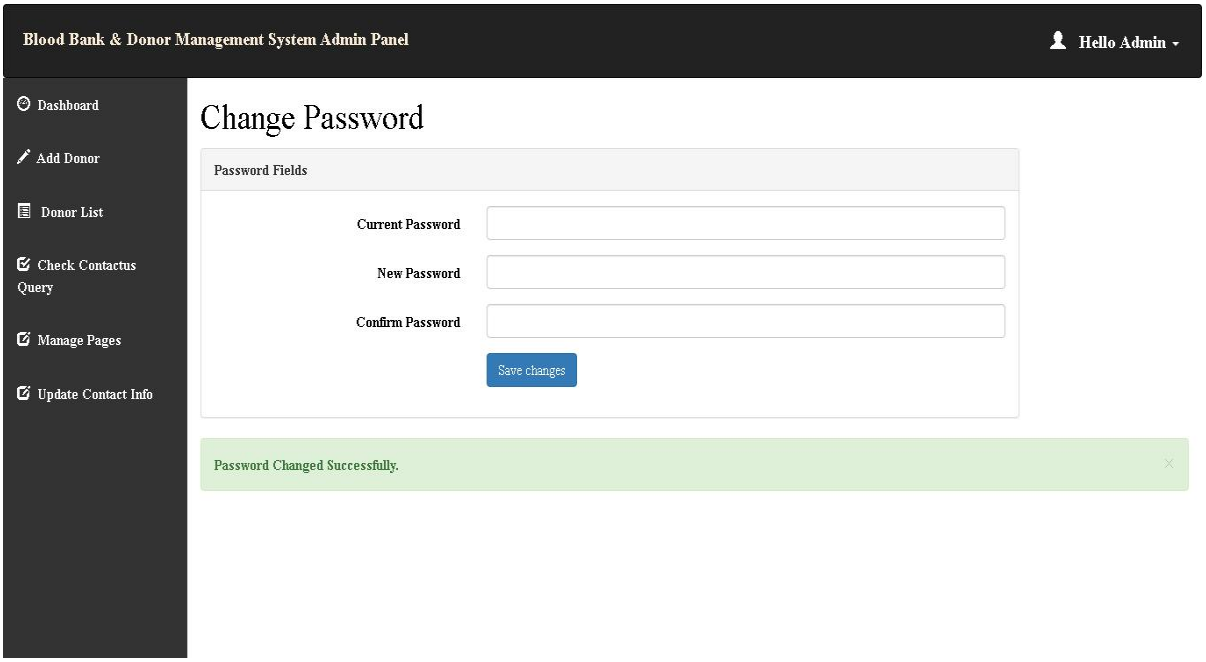
## (xii) admin/pages.php: -



## (xiii) admin/update_contact.php



## (xiv) admin/change_password.php



# CONCLUSION

In this project, **"BLOOD BANK & DONOR MANAGEMENT SYSTEM"** we have tried to computerize various processes of Blood Bank.

Blood Bank Management System is very flexible project and can be used in any branch of BLOOD BANK for keeping record. In this php project we have tried to provide all the Blood bank management system related record keeping facilities which helps to keep record and employees who belongs to it.

The main focus of this project is to less in human efforts. The maintenance of the record is made efficient, as all the records are stored in the SQL database.

It is user interactive and effective than the existing system. The flexibility of visual basic helps to maintain the **"BLOOD BANK & DONOR MANAGEMENT SYSTEM"** more efficiently.

Finally, I am thankful to all the people including my project partner who have supported me in the development of this project.

# REFERENCES

[1] Kevin Tatroe, Peter MacIntyre, Rasmus Lerdorf, Programming PHP: Creating Dynamic Web Pages, O'Reilly Media, Inc., 2013. ISBN: 9781449365837

[2] Dave W. Mercer, Beginning Php 5, John Wiley & Sons, 2009. ISBN: 9788126505395

[3] Peter Moulding, PHP Black Book, Coriolis Group Books, 2001 ISBN : 9781588800534

[4] https://www.w3schools.com/php

[5] Atkinson Leon, PHP Core Programming, Beijing:Tsinghua university press, 2000