**BLOOD DONATION WEBSITE**

**ABSTRACT**

The Blood Donation Website is an online platform that can be used to register blood donors and helps people search for donors registered in the database. The admin of this website can approve new donor registrations and can check their details. This project is developed using PHP, HTML language, and MYSQL for database connection. Creating and managing requirements is a challenge for IT, systems, and product development projects or indeed for any activity where you have to manage a contractual relationship. Organizations need to effectively define and manage requirements to ensure they are meeting the needs of the customer while proving compliance and staying on schedule and within budget. The impact of a poorly expressed requirement can bring a business out of compliance or even cause injury or death. Requirements definition and management is an activity that can deliver a high, fast return on investment. The project analyzes the system requirements and then comes up with the requirements specifications. It studies other related systems and then comes up with system specifications. The system is then designed by 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 confirmation, and updating while the interactive system deals with system interaction with the administration and users. This collects the personal data of users willing to donate blood and allows Administrators to approve their profiles and people to freely search for donors and contact them. It has two modules, User and Admin.

1. **INTRODUCTION**

This site collects the data of donors through a registration form and once their profile is approved by the admin the profile becomes active. Users who need blood during emergencies can open the website and search for donors according to their areas, towns, districts, or blood groups. This greatly increases the availability of donors and can increase the chance of saving a person’s life. The process is free and simple so that users do not have any troubles while searching for donors during emergencies. The admin can also use this website to search for donors in a particular area and give them calls regarding any blood donation drives or camps to help people in need

**IMPLEMENTATION**

The aim of designing and developing this Driving school website using PHP is to provide an easy and hassle-free registration process online, inquire about courses and sessions, and look at their scheduled sessions without the need to directly approach the school. Using Apache server pages and MySQL database the user can log in using their Email id and password after registration and approval and look at schedules. This also helps the Administrator to keep a record and manage all data within their PC or Smartphone anytime and anywhere. This makes new users to be able to enquire and enroll in classes even after closing hours and during holidays.

**EXISTING SYSTEM**

Hospitals usually keep a record of people who have taken blood tests or patients without any concerning illness. During emergencies, if the concerned person doesn’t have a person with a similar blood group in the family, friends or in contact, the hospital uses the record and calls the person with the same blood group. This record is limited as not all people visit the same hospital and sometimes requests are made with nearby hospitals for their records. Not everyone will be ready to donate blood and a lot of them would be in a situation where they cannot be able to donate blood. So the potential people who can donate blood becomes close to one to five people. This is a tedious process for a person running to save a life.

**PROPOSED SYSTEM**

The system of this research work was web-based. PHP (Personal Home Page) is a scripting language and an interpreter that is used to develop the website.

**Benefits:**

1. **It makes the whole process simple –** A donor can register their profile in the website, their profile gets approved, and donors can be searched and called without any hassle.
2. **Donors can be searched without any delays –** People can search for donors by simply typing the area or town or district and their blood group.
3. **Makes more donors available –** Since only donors register in this site, there are a lot of donors who can be contacted when some donors are not available.
4. **Hassle free contacting –** The recipients can contact the donor with the touch or a click of a button without going through needless procedures like in conventional websites.
5. **DESCRIPTION OF MODULES**

Module 1: User Authentication and Registration

This module handles the registration and authentication process for donors. Users can securely register for an account, providing necessary personal details. Authentication mechanisms ensure that only registered users can access the system, enhancing security and privacy.

Module 2: Donor Profile Management

In this module, donors can manage their profiles, update personal information, and specify blood donation preferences. Users have the flexibility to modify their profiles as needed, ensuring that their information is always accurate and up-to-date.

Module 5: Search and Filtering

Hospitals or blood banks can search for donors based on specific criteria such as blood type, location, and availability. Advanced filtering options help in identifying suitable donors efficiently, facilitating timely access to blood donations when needed.

Module 7: Admin Dashboard

Administrators have access to a comprehensive dashboard where they can manage donor profiles, appointments, and donation records. Additionally, administrators can generate reports for analysis and decision-making, providing valuable insights into donation trends and operational efficiency.

Module 8: Data Security

Ensuring the security of donor information is paramount. This module implements robust data security measures to protect sensitive donor data and comply with privacy regulations. Encryption, access control, and data backup mechanisms are employed to safeguard donor information.

Module 9: Reporting and Analytics

This module provides administrators with tools to generate reports and analyze donation data. Reports can include metrics such as donation frequency, blood type distribution, and donor demographics. Analyzing these insights helps in making informed decisions and optimizing donation processes.

Module 10: System Administration

System administrators oversee the overall functioning of the Blood Donors Management System. They are responsible for system maintenance, user management, and ensuring the smooth operation of all modules. System administrators play a crucial role in supporting and maintaining the system infrastructure.

**4. SYSTEM SPECIFICATION**

**HARDWARE SPECIFICATION**

|  |  |
| --- | --- |
| System | HP 15s |
| Processor | Ryzen 5 2.1 GHz |
| Storage | 512 GB SSD |
| RAM | 16 GB |
| Monitor | Integrated Monitor |
| Mouse | Integrated Trackpad |
| Keyboard | Integrated Keyboard |

**OPERATING SYSTEM**

|  |  |
| --- | --- |
| Operating System | Windows 11 |
| Front End | PHP Version 8 |
| Back End | MySQL Version 8 |
| Server | XAMPP |

**SOFTWARE SPECIFICATION**

**SOFTWARE DESCRIPTION**

**XAMPP:**

XAMPP is an [open-source](https://en.wikipedia.org/wiki/Free_software) [cross-platform](https://en.wikipedia.org/wiki/Cross-platform) [web server](https://en.wikipedia.org/wiki/Web_server) [solution stack](https://en.wikipedia.org/wiki/Solution_stack) package developed by Apache Friends, consisting mainly of the [Apache HTTP Server](https://en.wikipedia.org/wiki/Apache_HTTP_Server), [Maria DB](https://en.wikipedia.org/wiki/MariaDB) [database](https://en.wikipedia.org/wiki/Database), and [interpreters](https://en.wikipedia.org/wiki/Interpreter_%28computing%29) for scripts written in the [PHP](https://en.wikipedia.org/wiki/PHP) and [Perl](https://en.wikipedia.org/wiki/Perl) [programming languages](https://en.wikipedia.org/wiki/Programming_language). XAMPP stands for Cross-Platform (X), Apache (A), Maria DB (M), PHP (P), and Perl (P). It is a simple, lightweight Apache distribution that makes it extremely easy for developers to create a local web server for testing and deployment purposes.

Everything needed to set up a web server – server application (Apache), database (Maria DB), and scripting language (PHP) – is included in an extractable file. XAMPP is also cross-platform, which means it works equally well on Linux, Mac, and Windows.

XAMPP's designers intended it for use only as a development tool, to allow website designers and programmers to test their work on their computers without any access to the Internet.

**CROSS-PLATFORM**

Cross-platform software is a type of software application that works on multiple operating systems or devices, which are often referred to as platforms. A platform means an operating system such as Windows, Mac OS, Android, or iOS. When a software application works on more than one platform, the user can utilize the software on a wider choice of devices and computers.

**BENEFITS OF CROSS-PLATFORM**

The benefit of a cross-platform software app or program is that you can use the same program whether you’re on a Windows PC or whether you’re logging in from your laptop or smartphone. The Microsoft Office suite of applications, which includes Word, Excel, and PowerPoint, is available on Windows, Mac OS, iOS (iPhone/iPad), and Android. While there are differences based on how the platforms work, you’ll have a similar experience within the application between all of your devices.

Having a similar experience across any platform means there’s a much smaller learning curve if one even exists at all, so you’ll be more productive and be able to use a software product you’re familiar with regardless of the operating system or device you choose. In addition, your files can be moved much more easily between your devices so you can use the software with whatever device you have with you at the time. And there’s a way to keep all of your work in sync across all of your devices, by using the cloud.

**EXAMPLES OF CROSS-PLATFORM**

**Unity 3D**

First, let’s talk about Unity3D. I think the game engine should be preferred by people who want to write mobile games.  
You can develop games on 17 platforms using multiple languages, including Linux. Of course, iOS, Android, and Windows Phone is also the most ideal game engine to develop games.

You can develop your application using C #, JS, and C ++.

Link to: [https://unity3d.com](https://unity3d.com/)

# Xamarin

Xamarin Some time ago, it was purchased by Microsoft and is a perfect fit for developers using C #.

Because it is a C # language, it has a lot of documentation, and because of Microsoft support, Xamarin is the choice for C # developers.

In addition, you can do everything you can do in Objective-C, Swift, and Java with the Xamarin library.

Link to: [https://xamarin.com](https://xamarin.com/)

# React Native

React Native is an open-source JavaScript library developed by the new generation of React — Facebook, which was open to Github in 2013. Native application creation means writing applications only for a specific operating system. React Native helps developers reuse their code over the web and on mobile. Developers will not have to create the same app from scratch for iOS and Android. They will be able to reuse the code in each operating system. The great thing about React Native is that there is little difference between a finished application in Objective-C or Java and an application built using React Native. Android and iOS code development environments are very different. So it takes time to remove the application to two different platforms. However, with React Native, only one developer can write on different mobile operating systems.

**APACHE:**

The Apache HTTP Server, colloquially called Apache is a [free and open-source](https://en.wikipedia.org/wiki/Free_and_open-source) [cross-platform](https://en.wikipedia.org/wiki/Cross-platform) [web server](https://en.wikipedia.org/wiki/Web_server) software, released under the terms of [Apache License](https://en.wikipedia.org/wiki/Apache_License) 2.0. Apache is developed and maintained by an open community of developers under the auspices of the [Apache Software Foundation](https://en.wikipedia.org/wiki/Apache_Software_Foundation).

The vast majority of Apache HTTP Server instances run on a [Linux distribution](https://en.wikipedia.org/wiki/Linux_distribution), but current versions also run on [Microsoft Windows](https://en.wikipedia.org/wiki/Microsoft_Windows), [OpenVMS](https://en.wikipedia.org/wiki/OpenVMS),  and a wide variety of [Unix-like](https://en.wikipedia.org/wiki/Unix-like) systems. Past versions also ran on [NetWare](https://en.wikipedia.org/wiki/NetWare), [OS/2](https://en.wikipedia.org/wiki/OS/2), and other operating systems,  including ports to mainframes.

Originally based on the HTTP server, the development of Apache began in early 1995 after work on the NCSA code stalled. Apache played a key role in the initial growth of the [World Wide Web](https://en.wikipedia.org/wiki/World_Wide_Web), quickly overtaking NCSA HTTP as the dominant [HTTP](https://en.wikipedia.org/wiki/HTTP) server. In 2009, it became the first web server software to serve more than 100 million [websites](https://en.wikipedia.org/wiki/Website). As of January 2021, [Netcraft](https://en.wikipedia.org/wiki/Netcraft" \o "Netcraft) estimated that Apache served 24.63% of the million busiest websites, while [Nginx](https://en.wikipedia.org/wiki/Nginx) served 23.21% and Microsoft is in third place at 6.85% (for some of Netcraft's other stats Nginx is ahead of Apache), while according to W3Techs, Apache is ranked first at 35.0% and Nginx second at 33.0% and Cloudflare Server third at 17.3%.

**LANGUAGE SPECIFICATION**

**PHP**

**INTRODUCTION OF PHP**

PHP started as a small open-source project that evolved as more and more people found out how useful it was. Rasmus Lerdorf unleashed the first version of PHP way back in 1994.

* PHP is a recursive acronym for "PHP: Hypertext Preprocessor".
* PHP is a server-side scripting language that is embedded in HTML. It is used to manage dynamic content, databases, session tracking, and even build entire e-commerce sites.
* It is integrated with several popular databases, including MySQL, PostgreSQL, Oracle, Sybase, Informix, and Microsoft SQL Server.
* PHP is pleasingly zippy in its execution, especially when compiled as an Apache module on the Unix side. The MySQL server, once started, executes even very complex queries with huge result sets in record-setting time.
* PHP supports a large number of major protocols such as POP3, IMAP, and LDAP. PHP4 added support for Java and distributed object architectures (COM and CORBA), making n-tier development a possibility for the first time.
* PHP is forgiving: PHP language tries to be as forgiving as possible.
* PHP Syntax is the same as C language.

**What is a PHP File?**

* PHP files can contain text, HTML, CSS, JavaScript, and PHP code.
* PHP code is executed on the server, and the result is returned to the browser as plain HTML.
* PHP files have the extension ".php".

**What Can PHP Do?**

* PHP can generate dynamic page content and it can create, open, read, write, delete, and close files on the server and it can collect form data.
* PHP can send and receive cookies it can add, delete, and modify data in your database and it can be used to control user-access and encrypt data.

**Why PHP?**

* PHP runs on various platforms (Windows, Linux, Unix, Mac OS X, etc.).
* PHP is compatible with almost all servers used today (Apache, IIS, etc.).
* PHP supports a wide range of databases.
* PHP is free.
* PHP is easy to learn and runs efficiently on the server side.

## **What is a Database?**

* A database is a separate application that stores a collection of data. Each database has one or more distinct APIs for creating, accessing, managing, searching, and replicating the data it holds.
* Other kinds of data stores can be used, such as files on the file system or large hash tables in memory but data fetching and writing would not be so fast and easy with those types of systems.
* Nowadays, we use relational database management systems (RDBMS) to store and manage huge volumes of data. This is called a relational database because all the data is stored in different tables and relations are established using primary keys or other keys known as foreign keys.

**MySQL Database**

* MySQL is released under an open-source license. So you have nothing to pay to use it. MySQL is a very powerful program in its own right. It handles a large subset of the functionality of the most expensive and powerful database packages.
* MySQL uses a standard form of the well-known SQL data language. MySQL works on many operating systems and with many languages including PHP, PERL, C, C++, JAVA, etc. MySQL works very quickly and works well even with large data sets.
* MySQL is very friendly to PHP, the most appreciated language for web development. MySQL supports large databases, up to 50 million rows or more in a table.
* The default file size limit for a table is 4GB, but you can increase this (if your operating system can handle it) to a theoretical limit of 8 million terabytes (TB). MySQL is customizable.
* The open-source GPL license allows programmers to modify the MySQL software to fit their specific environments.

**TABLE CREATION**

* Name of the table
* Names of fields
* Definitions for each field
* Field Attribute **NOT NULL** is being used because we do not want this field to be NULL. So if the user tries to create a record with a NULL value, then MySQL will raise an error.
* Field Attribute **AUTO\_INCREMENT** tells MySQL to go ahead and add the next available number to the id field.
* Keyword **PRIMARY KEY** is used to define a column as the primary key. You can use multiple columns separated by a comma to define a primary key.

## **ADMINISTRATIVE MYSQL COMMAND**

* **USE DATABASE NAME**: This will be used to select a particular database in the MySQL work area.
* **SHOW DATABASES:** Lists the databases that are accessible by the MySQL DBMS.
* **SHOW TABLES:** Shows the tables in the database once a database has been selected with the use command.
* **SHOW COLUMNS FROM Table name:** Shows the attributes, types of attributes, key information, whether NULL is permitted, defaults, and other information for a table.
* **SHOW INDEX FROM Table name:** Presents the details of all indexes on the table, including the PRIMARY KEY

## **CREATING TABLES USING PHP SCRIPT:**

To create a new table in any existing database you would need to use PHP function **mysqli\_query()**.

## **Dropping Tables Using PHP Script:**

Drop an existing table in any database, you would need to use the PHP function **mysqli\_query()**.

## **INSERTING DATA USING PHP SCRIPT:**

**CREATE**

Create table statement is used to create a table in MySQL.

**SELECT**

The SELECT statement is used to select data from one or more tables.

**UPDATE**

The UPDATE statement is used to update existing records in a table:

## **DELETE**

The DELETE statement is used to delete records from a table:

**DATABASE DESIGN:**

The data in the system has to be stored and retrieved from the database. Designing the database is part of system design. Data elements and data structures to be stored have been identified at the analysis stage.

They are structured and put together to design the data storage and retrieval system. A database is a collection of interrelated data stored with minimum redundancy to serve many users quickly and efficiently.

The general objective is to make database access easy, quick, inexpensive, and flexible for the user. Relationships are established between the data items and unnecessary data items are removed.

Normalization is done to get an internal consistency of data and to have minimum redundancy and maximum stability. This ensures minimizing data storage required, minimizing chances of data inconsistencies, and optimizing for updates.

**INPUT DESIGN**

The Input design is the main feature of the system. Input design determines the format and validation criteria for data entering the system. Inputs originate with end-users; human factors play a significant role in input design. The input design is designed to control the input, avoid delay, and errors in data, avoid extra steps, to keep the process simple. The design of input focuses on controlling the amount of input required, controlling the errors, avoiding delay, avoiding extra steps, and keeping the process simple. The input is designed in such a way that it provides security and ease of use while retaining privacy.

The following are the general principles, that are considered in designing inputs,

* + - Enter only variable data
    - Do not input data that can be calculated
    - List of values
    - Sequence entry

**OUTPUT DESIGN**

Designing the output is more important than working up with a few layout charts and reports. The outputs are designed based on the issue encountered. It will also take care of who will receive the output, what for it is produced how many details are needed, when it is needed, and by what method.

The outputs designed in this system are easy to use and useful for their jobs. The outputs are simple to read and interpret. The outputs obtained from this system are designed by using a few guidelines, which are given below. The information should be clear and accurate, yet concise and restricted to relevant data. Reports should have titles, data, and descriptive headings for columns of data, numbered pages, and so on.

**SYSTEM TESTING**

System testing is the process of exercising software with the intent of finding and ultimately correcting errors. This fundamental philosophy does not change for web applications, because Web-based systems and applications reside on a network and interoperate with many different operating systems, browsers, hardware platforms, and communication protocols; the search for errors represents a significant challenge for web applications.

The distributed nature of client/server environments, the performance issues associated with transaction processing, the potential presence of several different hardware platforms, the complexities of network communication, the need to serve multiple clients from a centralized database, and the requirements imposed on the server all combine to make testing of client\server architectures.

Testing issues

* Client GUI considerations
* Target environment and platform diversity considerations
* Distributed database considerations
* Distributed processing considerations

**TYPES OF TESTING**

1. Unit Testing

2. Integration Testing

3. Validation Testing

4. User Acceptance Testing

5. System Testing

**Unit Testing**

All modules were tested and individually as soon as they were completed were checked for their correct functionality. Unit testing is carried out by verifying and recovering errors within the boundary of the smallest unit or a module. In this testing step, each module was found to be working satisfactorily per the expected output of the module. In the package development, each module is tested separately after it has been completed and checked with valid data.

**Integration Testing**

The entire project was split into small programs; each of these single programs gives a frame as an output. These programs were tested individually; at last, all these programs were combined by creating another program where all these constructions were used. It causes a lot of problems by not functioning in an integrated manner.

The user interface testing is important since the user has to declare that the arrangements made in the frames are convenient and it is satisfied. When the frames are tested, the end user gives suggestions. Since they were much exposed to do the work manually.

**Validation Testing**

At the culmination of the black box testing software is completely assembled as a package. Interfacing errors have been uncovered and corrected and a final series of tests i.e., validation succeeds when the software functions in a manner that can be reasonably accepted by the customer.

**User Acceptance Testing**

User acceptance testing of the system is the key factor in the success of any system. The system under consideration is tested for user acceptance by constantly keeping in touch with prospective systems at the time of development and making changes whenever required. This is done concerning the input screen design and output screen design.

**System Testing**

This is to verify that all the system elements have been properly integrated and perform allocated functions. Testing is executing a program to test the logic changes made in it to find errors. Tests are also conducted to find discrepancies between the system and its original objective, current specifications, and documents.

**SYSTEM IMPLEMENTATION**

Implementation is the stage in the project where the theoretical design is turned into a working system. The most crucial stage is achieving a successful new system & and giving the user confidence that the new system will work efficiently & and effectively in the implementation stage.

The stage consists of

* + - Testing the developed program with simple data.
    - Detections and correction of errors.
    - Creating whether the system meets user requirements.
    - Testing whether the system.
    - Making necessary changes as desired by the user.
    - Training user personnel.

**Implementation Procedures**

The implementation phase is less creative than the system design. A system project may be dropped at any time before implementation, although it becomes more difficult when it goes to the design phase.

The final report to the implementation phase includes procedural flowcharts, record layouts, report layouts, and a workable plan for implementing the candidate system design into an operational one. Conversion is one aspect of implementation.

**System Maintenance**

Maintenance is the implementation of the review plan. As important as it is, many programmers and analysts are to perform or identify themselves with the maintenance effort. There are psychological, personality, and professional reasons for this. Analysts and programmers spend far more time maintaining programs than they do writing them. Maintenance accounts for 50-80 percent of total system development.

Maintenance is expensive. One way to reduce maintenance costs is through maintenance management and software modification audits.

* Maintenance is not as rewarding or exciting as developing systems. It is perceived as requiring neither skill nor experience.
* Users are not fully cognizant of the maintenance problem or its high cost.
* Few tools and techniques are available for maintenance.
* A good test plan is lacking.
* Standards, procedures, and guidelines are poorly defined and enforced.
* Programs are often maintained without care for structure and documentation.
* There are minimal standards for maintenance.
* Programmers expect that they will not be in their current commitment by the time their programs go into the maintenance cycle.

**SYSTEM DESIGN**

System design is "the process of studying a procedure or business to identify its goals, purposes and create systems and procedures that will efficiently achieve them". Another view sees system analysis as a problem-solving technique that breaks down a system into its component pieces for the study of how well those parts work and interact to accomplish their purpose.

The field of system analysis relates closely to requirements analysis or operations research. It is also "an explicit formal inquiry carried out to help a decision maker identify a better course of action and make a better decision than they might otherwise have made.

**DESIGN NOTATION**

Design notations are used when planning and should be able to communicate the purpose of a program without the need for formal code. Commonly used design notations are:

* DFD
* ERD
* **DFD (DATA FLOW DIAGRAM):**

A data flow diagram (DFD) is a graphical representation of the "flow" of data through an [information system](https://en.wikipedia.org/wiki/Information_system), modeling its process aspects. A DFD is often used as a preliminary step to create an overview of the system without going into great detail, which can later be elaborated. DFDs can also be used for the [visualization](https://en.wikipedia.org/wiki/Data_visualization) of [data processing](https://en.wikipedia.org/wiki/Data_processing) (structured design). A DFD shows what kind of information will be input to and output from the system, how the data will advance through the system, and where the data will be stored. It does not show information about the timing of the process or information about whether processes will operate in sequence or parallel, unlike a [flowchart](https://en.wikipedia.org/wiki/Flowchart) which also shows this information.

Data flow diagrams were popularized in the late 1970s, arising from the book Structured Design, by computing pioneers Ed Yourdon and Larry Constantine. They based it on the “data flow graph” computation models by David Martin and Gerald Estrin. The structured design concept took off in the software engineering field, and the DFD method took off with it. It became more popular in business circles, as it was applied to business analysis than in academic circles.

**DFD SYMBOLS**

The process that transforms data flow

Source or Destination of Data

Data Flow

Data source

**ENTITY RELATIONSHIP DIAGRAM**

The relation upon the system is structured through a conceptual ER-Diagram, which not only specifies the existential entities but also the standard relations through which the system exists and the cardinalities that are necessary for the system state to continue. The Entity Relationship Diagram (ERD) depicts the relationship between the data objects. The ERD is the notation that is used to conduct the data modeling activity The attributes of each data object noted in the ERD can be described resign a data object description.

The set of primary components that are identified by the ERD are

* + Data object
  + Relationships
  + Attributes
  + Various types of indicators

The primary purpose of the ERD is to represent data objects and their relationships.

**ER-DIAGRAM SYMBOL**

Entity

Relationship

Flow

* **INPUT DESIGN**

The input design is the link between the information system and the user. It comprises the developing specifications and procedures for data preparation and those steps are necessary to put transaction data into a usable form for processing can be achieved by inspecting the computer to read data from a written or printed document or it can occur by having people keying the data directly into the system. The design of input focuses on controlling the amount of input required, controlling the errors, avoiding delay, avoiding extra steps, and keeping the process simple. The input is designed in such a way that it provides security and ease of use while retaining privacy.

Input Design is the process of converting a user-oriented description of the input into a computer-based system. This design is important to avoid errors in the data input process and show the correct direction to the management for getting correct information from the computerized system. It is achieved by creating user-friendly screens for the data entry to handle large volumes of data.

The goal of designing input is to make data entry easier and to be free from errors. The data entry screen is designed in such a way that all the data manipulations can be performed. It also provides record viewing facilities. When the data is entered it will check for its validity. Data can be entered with the help of screens.

* **DATABASE DESIGN**

The database is designed to manage large bodies of information. The management of data involves both the definitions of structures for the storage of information. In addition, the database system must provide for the safety of the information solved, despite system crashes or attempts at unauthorized access. For developing an efficient database users have to fulfill certain conditions such as controlled redundancy.

* Defining the data
* Inputting the data
* Locating the data
* Accessing the data
* Communicating the data

Revising the data

**Objectives of Database Design**

For designing a database design several objectives have to be met as follows:

* Ease of use
* Control of data integrity
* Control of redundancy
* Control of security
* Data independence (logical & physical)
* Data storage protection
* System performance
* **OUTPUT DESIGN**

A quality output is one, which meets the requirements of the end user and presents the information. In any system results of processing are communicated to the users and other systems through outputs. In output design it is determined how the information is to be displaced for immediate need and also the hard copy output. It is the most important and direct source of information to the user. Efficient and intelligent output design improves the system’s relationship to help user decision-making.

Output design generally refers to the results and information that are generated by the system for many end-users; output is the main reason for developing the system and the basis on which they evaluate the usefulness of the application.

In this Online Repository System project output is to view customer details, employee lists, order tracking details, and attendance percentage results.

**SYSTEM FLOW DIAGRAM**

**DATA FLOW DIAGRAM**

**LEVEL 0:**

User

Admin

Register themselves as donors or search for donors

Request for becoming donor

Gets Donor list

Approval of Donor

**LEVEL 1:**

User

Name and other personal details

**Level 2**

Admin

Login

Approval page

(Donor approval

Search for donors and contact them

**Use case diagram**

Admin

User

**ER DIAGRAM**

Login

Register as donors / Search for donors

Approval of Donors

Search for donors and contact them

Admin

User

**DATABASE DESIGN**

**Table name: admin**

|  |  |  |  |
| --- | --- | --- | --- |
| **FIELD NAME** | **DATA TYPE** | **LENGTH** | **DESCRIPTION** |
| id | int | 10 | unique id |
| name | varchar | 255 | Admin name |
| email | varchar | 255 | Admin Email ID |
| pswd | varchar | 255 | Admin Password |

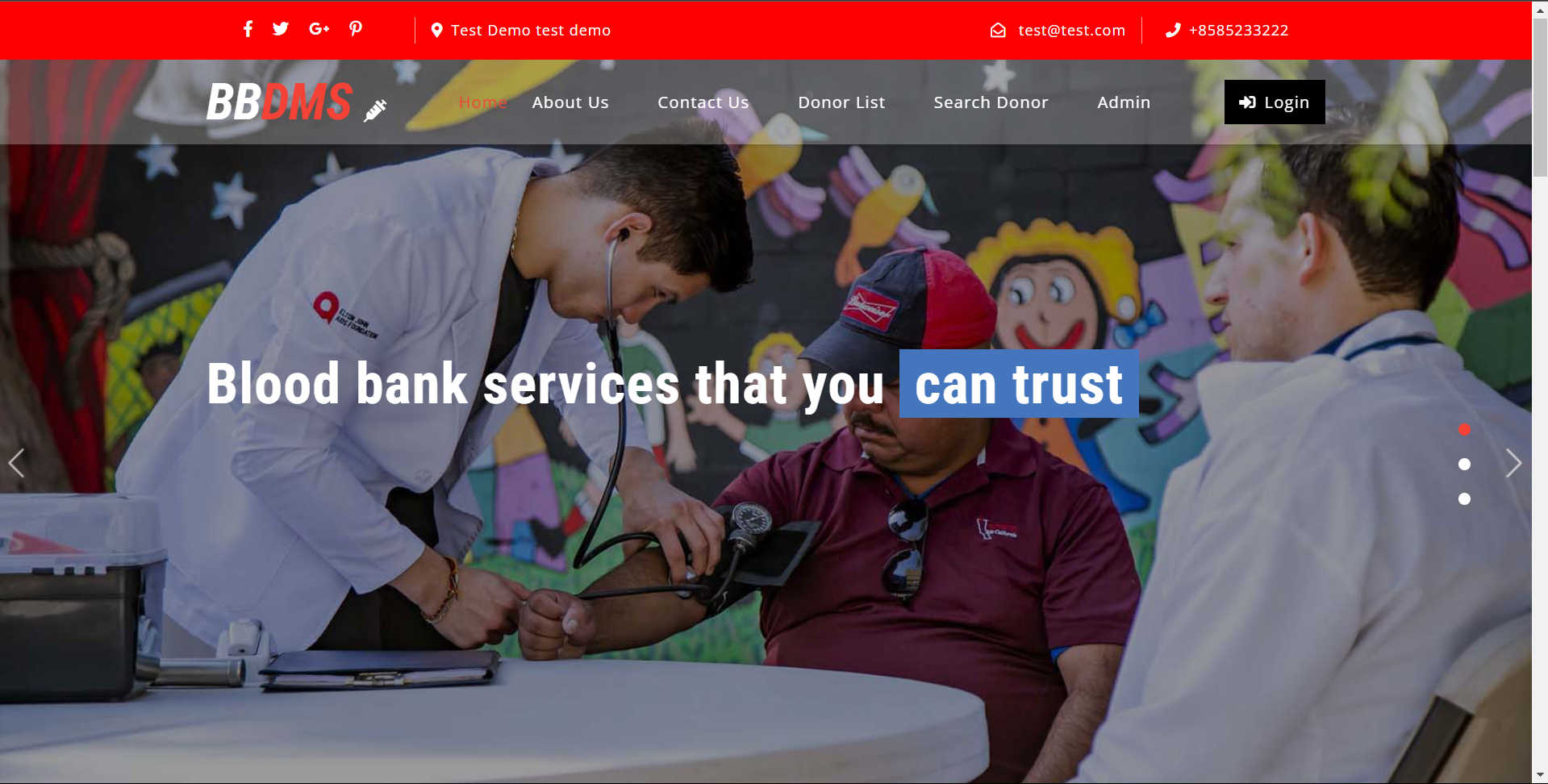
**Table name: record**

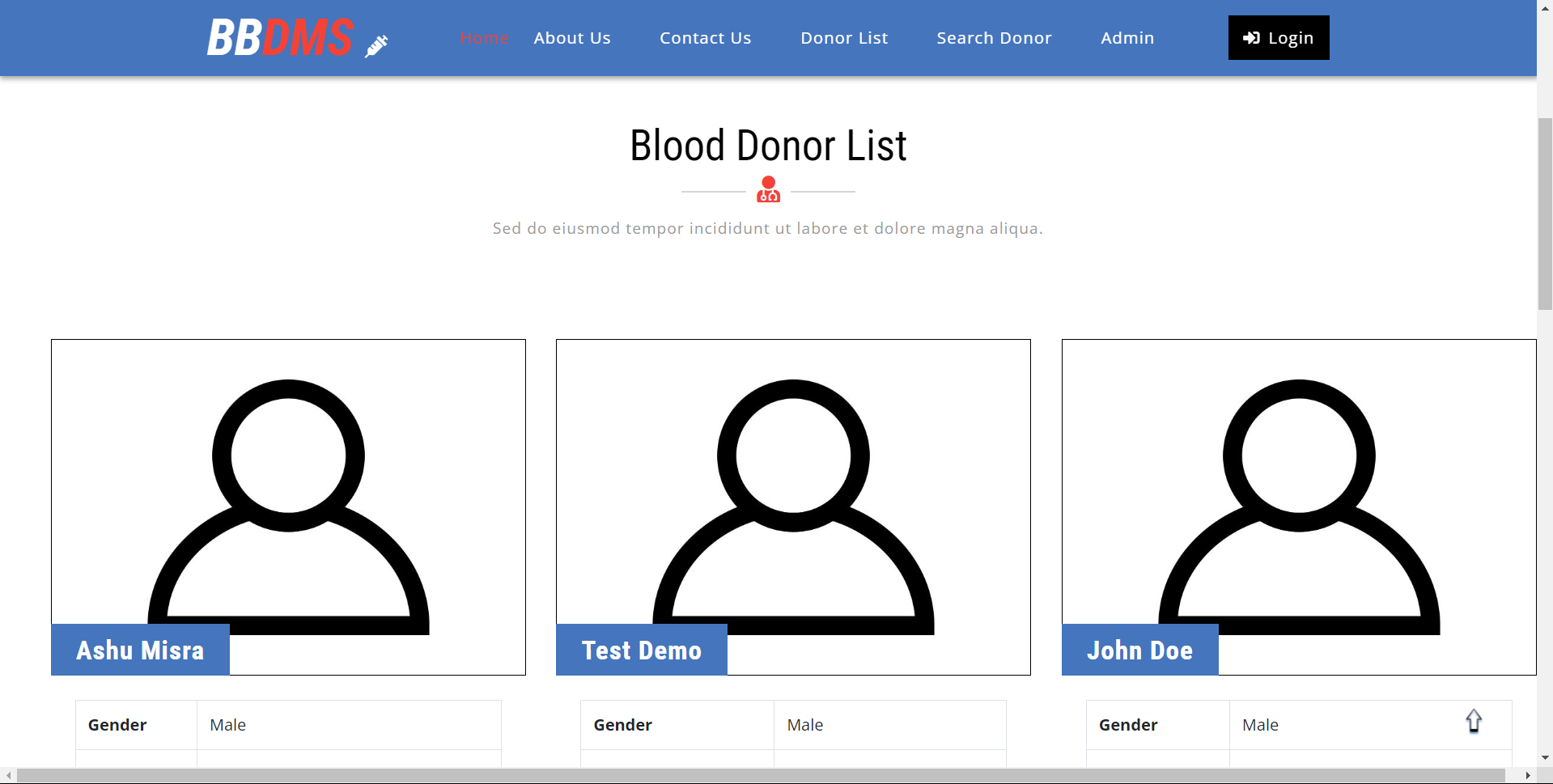
|  |  |  |  |
| --- | --- | --- | --- |
| **FIELD NAME** | **DATA TYPE** | **LENGTH** | **DESCRIPTION** |
| id | int | 11 | Unique ID |
| name | varchar | 255 | First name |
| dob | date |  | Date of birth |
| blood | varchar | 255 | Blood group |
| address | varchar | 255 | Donor address |
| area | varchar | 255 | Area of Donor |
| dist | varchar | 255 | District of donor |
| gender | varchar | 255 | Gender |
| doc | varchar | 255 | ID Proof |
| email | varchar | 255 | Email ID of user |
| no | int | 11 | Mobile number |

**CONCLUSION**

This project is developed to increase the availability of donors for people in times of emergencies. The donors can easily register using their personal details and upload their ID proof for verification. The admin can approve their profile after verification and they become active donors in the website. Recipients can search for donors simply by their blood group, town/ area, or district and can look at the available donors and can call them with just a tap. The fast and simple process helps in getting donors fast and hassle free by themselves. The donor numbers keep increasing as the website gets more popular and more donors become available for donating blood compared to conventional records in hospitals which are old and outdated.

**SCREENSHOT**

****

****

**SAMPLE CODE**

<?php

session\_start();

?>

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="utf-8">

<title>Admin</title>

<meta content="width=device-width, initial-scale=1.0" name="viewport">

<meta content="" name="keywords">

<meta content="" name="description">

<!-- Favicon -->

<link href="img/favicon.ico" rel="icon">

<!-- Google Web Fonts -->

<link rel="preconnect" href="https://fonts.googleapis.com">

<link rel="preconnect" href="https://fonts.gstatic.com" crossorigin>

<link href="https://fonts.googleapis.com/css2?family=Heebo:wght@400;500;600;700&display=swap" rel="stylesheet">

<!-- Icon Font Stylesheet -->

<link href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/5.10.0/css/all.min.css" rel="stylesheet">

<link href="https://cdn.jsdelivr.net/npm/bootstrap-icons@1.4.1/font/bootstrap-icons.css" rel="stylesheet">

<!-- Libraries Stylesheet -->

<link href="lib/owlcarousel/assets/owl.carousel.min.css" rel="stylesheet">

<link href="lib/tempusdominus/css/tempusdominus-bootstrap-4.min.css" rel="stylesheet" />

<!-- Customized Bootstrap Stylesheet -->

<link href="css/bootstrap.min.css" rel="stylesheet">

<!-- Template Stylesheet -->

<link href="css/style.css" rel="stylesheet">

</head>

<body>

<div class="container-xxl position-relative bg-white d-flex p-0">

<!-- Spinner Start -->

<div id="spinner" class="show bg-white position-fixed translate-middle w-100 vh-100 top-50 start-50 d-flex align-items-center justify-content-center">

<div class="spinner-border text-primary" style="width: 3rem; height: 3rem;" role="status">

<span class="sr-only">Loading...</span>

</div>

</div>

<!-- Spinner End -->

<!-- Sidebar Start -->

<div class="sidebar pe-4 pb-3">

<nav class="navbar bg-light navbar-light">

<a href="index.html" class="navbar-brand mx-4 mb-3">

<h3 class="text-primary">ADMIN</h3>

</a>

<div class="d-flex align-items-center ms-4 mb-4">

<div class="position-relative">

<img src="img/user.png" height="50px" width="50px">

</div>

<div class="ms-3">

<h6 class="mb-0"><?php echo $\_SESSION['uname'];?></h6>

<span>Admin</span>

</div>

</div>

<div class="navbar-nav w-100">

<a href="mainpage.php" class="nav-item nav-link"><i class="fa fa-tachometer-alt me-2"></i>New Registrations</a>

<a href="donors.php" class="nav-item nav-link"><i class="fa fa-th me-2"></i>Search Donors</a>

</div>

</nav>

</div>

<!-- Sidebar End -->

<!-- Content Start -->

<div class="content">

<!-- Navbar Start -->

<nav class="navbar navbar-expand bg-light navbar-light sticky-top px-4 py-0">

<a href="#" class="sidebar-toggler flex-shrink-1">

<i class="fa fa-bars"></i>

</a>

<div class="navbar-nav align-items-center ms-auto">

<div class="nav-item dropdown">

<a href="#" class="nav-link dropdown-toggle" data-bs-toggle="dropdown">

<img class="rounded-circle me-lg-2" src="img/user.png" alt="" style="width: 40px; height: 40px;">

<?php

if(isset($\_SESSION['uname']))

{

?>

<span class="d-none d-lg-inline-flex"><?php echo $\_SESSION['uname'];?></span>

</a>

<div class="dropdown-menu dropdown-menu-end bg-light border-0 rounded-0 rounded-bottom m-0">

<a href="signout.php" class="dropdown-item">Log Out</a>

</div>

</div>

</div>

</nav>

<?php

}

else{

?><script>

alert("Username or Password Incorrect");

window.location.href = "index.php";

</script>

<?php

}?>

<!-- Navbar End -->

<!-- ################# Slider Starts Here#######################--->

<div class="slider-detail">

<div id="carouselExampleIndicators" class="carousel slide" data-ride="carousel">

<ol class="carousel-indicators">

<li data-target="#carouselExampleIndicators" data-slide-to="0" class="active"></li>

<li data-target="#carouselExampleIndicators" data-slide-to="1"></li>

</ol>

<div class="carousel-inner">

<div class="carousel-item active">

<img class="d-block w-100" src="assets/images/slider/slide-02.jpg" alt="First slide">

<div class="carousel-caption d-none d-md-block">

<h5 class=" bounceInDown">Donate Blood & Save a Life</h5>

<p class=" bounceInLeft">Blood donation is a practice where people donate their blood to help others with health problems. Blood is a vital fluid that helps the body function properly. When the body loses too much blood, people can get deadly diseases or even die.

Blood donation is a critical part of healthcare worldwide. It ensures a stable supply of blood for patients who need transfusions. These patients include those with thalassemia, cancer, and victims of accidents or natural disasters.</p>

<div class=" vbh">

<div class="btn btn-success bounceInUp">Donate Now</div>

<a href="#contact"><div class="btn btn-success bounceInUp"> Contact US </div></a>

</div>

</div>

</div>

<div class="carousel-item">

<img class="d-block w-100" src="assets/images/slider/slide-03.jpg" alt="Third slide">

<div class="carousel-caption vdg-cur d-none d-md-block">

<h5 class=" bounceInDown">Donate Blood & Save a Life</h5>

<p class=" bounceInLeft">Blood donation is a practice where people donate their blood to help others with health problems. Blood is a vital fluid that helps the body function properly. When the body loses too much blood, people can get deadly diseases or even die.

Blood donation is a critical part of healthcare worldwide. It ensures a stable supply of blood for patients who need transfusions. These patients include those with thalassemia, cancer, and victims of accidents or natural disasters.</p>

<div class=" vbh">

<div class="btn btn-danger bounceInUp"> Donate Now </div>

<a href="#contact"><div class="btn btn-success bounceInUp"> Contact US </div></a>

</div>

</div>

</div>

</div>

<a class="carousel-control-prev" href="#carouselExampleIndicators" role="button" data-slide="prev">

<span class="carousel-control-prev-icon" aria-hidden="true"></span>

<span class="sr-only">Previous</span>

</a>

<a class="carousel-control-next" href="#carouselExampleIndicators" role="button" data-slide="next">

<span class="carousel-control-next-icon" aria-hidden="true"></span>

<span class="sr-only">Next</span>

</a>

</div>

</div>

<!--\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* About Us Starts Here \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*-->

<section id="about" class="contianer-fluid about-us">

<div class="container">

<div class="row session-title">

<h2>About Us</h2>

<p>Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has</p>

</div>

<div class="row">

<div class="col-md-6 text">

<h2>About Blood Doners</h2>

<p>Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book. It has survived not only five centuries, but also the leap into electronic typesetting, remaining essentially unchanged.</p>

<p> It was popularised in the 1960s with the release of Letraset sheets containing Lorem Ipsum passages, and more recently with desktop publishing software like Aldus PageMaker including versions of Lorem Ipsum.</p>

<p>There are many variations of passages of Lorem Ipsum available, but the majority have suffered alteration in some formhumour, or randomised words which don't look even slightly believable. If you are going to use a passage. industry's standard dummy has survived not only five centuries, but also the leap into electronic typesetting, remaining essentially unchanged.</p>

<p>Industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book. It has survived not only five centuries, but also the leap into electronic typesetting, remaining essentially unchanged.</p>

</div>

<div class="col-md-6 image">

<img src="assets/images/about.jpg" alt="">

</div>

</div>

</div>

</section>

<!-- ################# Gallery Start Here #######################--->

<!-- ################# Donation Process Start Here #######################--->

<section id="process" class="donation-care">

<div class="container">

<div class="row session-title">

<h2>Donation Process</h2>

<p>The donation process from the time you arrive center until the time you leave</p>

</div>

<div class="row">

<div class="col-md-3 col-sm-6 vd">

<div class="bkjiu">

<img src="assets/images/gallery/g1.jpg" alt="">

<h4><b>1 - </b>Registration</h4>

<p>Ut wisi enim ad minim veniam, quis laore nostrud exerci tation ulm hedi corper turet suscipit lobortis</p>

</div>

</div>

<div class="col-md-3 col-sm-6 vd">

<div class="bkjiu">

<img src="assets/images/gallery/g2.jpg" alt="">

<h4><b>2 - </b>Seeing</h4>

<p>Ut wisi enim ad minim veniam, quis laore nostrud exerci tation ulm hedi corper turet suscipit lobortis</p>

</div>

</div>

<div class="col-md-3 col-sm-6 vd">

<div class="bkjiu">

<img src="assets/images/gallery/g3.jpg" alt="">

<h4><b>3 - </b>Donation</h4>

<p>Ut wisi enim ad minim veniam, quis laore nostrud exerci tation ulm hedi corper turet suscipit lobortis</p>

</div>

</div>

<div class="col-md-3 col-sm-6 vd">

<div class="bkjiu">

<img src="assets/images/gallery/g4.jpg" alt="">

<h4><b>4 - </b>Save Life</h4>

<p>Ut wisi enim ad minim veniam, quis laore nostrud exerci tation ulm hedi corper turet suscipit lobortis</p>

</div>

</div>

</div>

</div>

</section>

FUTURE SCOPE

1. **Mobile App Development**: To enhance accessibility and reach a wider audience, future developments could involve the creation of a dedicated mobile app for the blood donation website. The app can provide features such as location-based alerts for nearby blood donation drives, push notifications for urgent blood donation needs, and integration with health tracking apps to encourage regular blood donors.
2. **Integration with Health Records**: Future advancements could involve integrating the blood donation website with electronic health record (EHR) systems. This integration would allow donors to securely share their health information, medical history, and blood type with blood banks and healthcare providers, streamlining the donation process and ensuring compatibility with recipient needs.
3. **Artificial Intelligence (AI) for Donor Matching**: Implementing AI algorithms can optimize the donor matching process by analyzing donor profiles, medical histories, and recipient needs to identify suitable matches. AI-powered recommendation engines can suggest potential donors based on factors such as blood type, location, availability, and eligibility criteria, improving efficiency and reducing wait times for patients in need of blood transfusions.
4. **Blockchain for Transparency and Traceability**: Leveraging blockchain technology can enhance transparency, traceability, and trust in the blood donation process. Blockchain-based solutions can create immutable records of blood donations, verify donor identities, track blood units from donation to transfusion, and ensure compliance with regulatory requirements. This increases confidence among donors, recipients, and healthcare providers while reducing the risk of fraud or errors.
5. **Gamification and Incentive Programs**: To encourage regular blood donation and reward donors for their contributions, future blood donation websites can implement gamification elements and incentive programs. Donors can earn points, badges, or rewards for each donation, participate in challenges or competitions, and receive recognition for their altruism. Incentive programs can include perks such as priority access to appointments, discounts at partner stores, or exclusive events for donors.
6. **Community Engagement and Education**: Blood donation websites can serve as platforms for community engagement and education on blood donation-related topics. Future developments could include interactive forums, webinars, and educational resources to raise awareness about the importance of blood donation, dispel myths and misconceptions, and promote a culture of voluntary blood donation.
7. **Real-time Inventory Management**: Implementing real-time inventory management systems can optimize blood supply chain logistics and ensure timely availability of blood products. Integration with blood bank databases, hospitals, and donation centers can provide up-to-date information on blood inventory levels, expiration dates, and demand forecasts, enabling efficient allocation and distribution of blood products to meet patient needs.
8. **Telehealth and Remote Donation Services**: With advancements in telehealth technology, future blood donation websites could offer remote donation services, allowing donors to participate in virtual donation consultations, medical screenings, and pre-donation assessments from the comfort of their homes. Telehealth platforms can facilitate remote blood donation drives, donor recruitment, and follow-up care, expanding access to blood donation opportunities for individuals in remote or underserved areas.

By embracing these future trends and innovations, blood donation websites can evolve into comprehensive platforms that leverage technology to enhance the efficiency, accessibility, and impact of blood donation efforts, ultimately saving more lives and improving patient outcomes.

Top of Form

REFFERENCES

BIBLIOGRAPHY

Comprehensive Web Development Textbook References

General Web Development:

• MDN Web Docs: (https://developer.mozilla.org/) - The authoritative source from Mozilla, offering in-depth documentation, tutorials, and references for various web technologies.

• W3Schools: (https://www.w3schools.com/) - A well-established website with interactive tutorials, references, and examples for a wide range of web development topics.

• The Odin Project: (https://theodinproject.com/) - A free, full-stack web development curriculum with a strong focus on practical projects.

• FreeCodeCamp: (https://www.freecodecamp.org/) - Another free coding platform with interactive lessons, projects, and a supportive community for web development learning.

Specific Technologies:

• HTML:

o HTML Dog: (https://www.htmldog.com/) - Offers interactive tutorials, challenges, and references for learning HTML.

• CSS:

o CSS-Tricks: (https://css-tricks.com/) - A popular website with articles, tips, tricks, and best practices for mastering CSS.

• JavaScript:

o JavaScript30: (https://javascript30.com/) - Provides 30 JavaScript coding challenges for practicing and improving your JavaScript skills.

o Eloquent JavaScript Website: (https://eloquentjavascript.net/) - Interactive tutorials and resources aligned with the book "Eloquent JavaScript" by Marijn Haverbeke.

• Bootstrap:

o Get Bootstrap: (https://getbootstrap.com/) - The official Bootstrap website with comprehensive documentation, examples, and tutorials for learning and using the framework.

o Start Bootstrap: (https://startbootstrap.com/) - Offers free Bootstrap templates to use as a starting point for your projects.

• PHP:

o PHP.net: (https://www.php.net/manual/en/index.php) - The official PHP website with comprehensive documentation, tutorials, and a reference manual.

o Laracasts: (https://laracasts.com/) - Features video tutorials and screencasts for learning PHP, Laravel (a popular PHP framework), and other web development topics.

• MySQL:

o MySQL Documentation: (https://dev.mysql.com/doc/) - Official documentation from MySQL, providing detailed information on using the database management system.

o SQLBolt: (https://sqlbolt.com/) - An interactive platform where you can practice writing and running SQL queries, the language used with MySQL.

Community Resources:

• Stack Overflow: (https://stackoverflow.com/) - A question-and-answer website for programmers, where you can search for solutions to your coding problems or ask questions related to web development.

• GitHub: (https://github.com/) - A version control system for code hosting and collaboration. GitHub also offers a wealth of open-source web development projects that you can explore and learn from.

Comprehensive Web Development Online References

General Web Development:

• MDN Web Docs: (https://developer.mozilla.org/) - The authoritative source from Mozilla, offering in-depth documentation, tutorials, and references for various web technologies.

• W3Schools: (https://www.w3schools.com/) - A well-established website with interactive tutorials, references, and examples for a wide range of web development topics.

• The Odin Project: (https://theodinproject.com/) - A free, full-stack web development curriculum with a strong focus on practical projects.

• FreeCodeCamp: (https://www.freecodecamp.org/) - Another free coding platform with interactive lessons, projects, and a supportive community for web development learning.

Specific Technologies:

• HTML:

o HTML Dog: (https://www.htmldog.com/) - Offers interactive tutorials, challenges, and references for learning HTML.

• CSS:

o CSS-Tricks: (https://css-tricks.com/) - A popular website with articles, tips, tricks, and best practices for mastering CSS.

• JavaScript:

o JavaScript30: (https://javascript30.com/) - Provides 30 JavaScript coding challenges for practicing and improving your JavaScript skills.

o Eloquent JavaScript Website: (https://eloquentjavascript.net/) - Interactive tutorials and resources aligned with the book "Eloquent JavaScript" by Marijn Haverbeke.

• Bootstrap:

o Get Bootstrap: (https://getbootstrap.com/) - The official Bootstrap website with comprehensive documentation, examples, and tutorials for learning and using the framework.

o Start Bootstrap: (https://startbootstrap.com/) - Offers free Bootstrap templates to use as a starting point for your projects.

• PHP:

o PHP.net: (https://www.php.net/manual/en/index.php) - The official PHP website with comprehensive documentation, tutorials, and a reference manual.

o Laracasts: (https://laracasts.com/) - Features video tutorials and screencasts for learning PHP, Laravel (a popular PHP framework), and other web development topics.

• MySQL:

o MySQL Documentation: (https://dev.mysql.com/doc/) - Official documentation from MySQL, providing detailed information on using the database management system.

o SQLBolt: (https://sqlbolt.com/) - An interactive platform where you can practice writing and running SQL queries, the language used with MySQL.

Community Resources:

• Stack Overflow: (https://stackoverflow.com/) - A question-and-answer website for programmers, where you can search for solutions to your coding problems or ask questions related to web development.

• GitHub: (https://github.com/) - A version control system for code hosting and collaboration. GitHub also offers a wealth of open-source web development projects that you can explore and learn from.