**BLOOD BANK DONATION SYSTEM**

**A MINI PROJECT REPORT**

***Submitted by***

**SANSKAR GUPTA [RA2011003011308]**

*Under the guidance of*

**Mr. Saminathan S**

***In partial satisfaction of the requirements for the degree of***

**BACHELOR OF TECHNOLOGY**

in

**COMPUTER SCIENCE & ENGINEERING**

****

**SCHOOL OF COMPUTING**

**COLLEGE OF ENGINEERING AND TECHNOLOGY**

**SRM INSTITUTE OF SCIENCE AND TECHNOLOGY**

**KATTANKULATHUR - 603203**

**APRIL 2023**

|  |  |
| --- | --- |
|  | COLLEGE OF ENGINEERING & TECHNOLOGY  SRM INSTITUTE OF SCIENCE & TECHNOLOGY  S.R.M. NAGAR, KATTANKULATHUR – 603 203 |

**BONAFIDE CERTIFICATE**

Certified that this project report **BLOOD BANK DONATION SYSTEM** is the bonafide work of **SANSKAR GUPTA [RA2011003011308]** of III Year/VI Sem B. tech (CSE)who carried out the mini project work under my supervision for the course **18CSC303J- Database Management systems** in SRM Institute of Science and Technology during the academic year 2022-2023 (Even sem).

|  |  |
| --- | --- |
|  | **SIGNATURE** S. Saminathan  Assistant Professor (Senior Grade)  School of Computing |

**CONTENTS**

1. Abstract
2. Introduction
3. Applications
4. Key Features
5. Software Requirement Specification
6. System Architecture and Design Figures
7. Code
8. Conclusion
9. References

**ABSTRACT**

The Blood Bank Donation System is an essential component of modern healthcare. Blood is a lifesaving resource that is constantly in demand, particularly during emergencies, accidents, and surgeries. The system aims to provide a safe, efficient, and effective way to collect, store, and distribute blood products to patients in need.

The donation process involves a screening process that evaluates the donor's medical history, lifestyle, and health status to ensure the safety of both the donor and the recipient. The collected blood is then processed and stored in a blood bank facility until it is needed.

The system relies on volunteer donors who generously give their time, blood, and energy to help those in need. The donors come from all walks of life and donate for various reasons, including the desire to help others, the satisfaction of knowing they are making a difference, and the sense of community involvement. The Blood Bank Donation System is critical in emergencies, disasters, and crises when blood supplies are often depleted. Through effective donation campaigns, public education, and outreach programs, the system ensures a constant supply of safe blood products to hospitals and healthcare facilities.

In conclusion, the Blood Bank Donation System is a vital component of the healthcare system that relies on volunteer donors to provide safe and effective blood products to those in need. It is a testament to the generosity and kindness of the human spirit and an essential lifeline for patients in crisis.

**INTRODUCTION**

Blood donation is a life-saving act that has been recognized as one of the most significant contributions an individual can make to their community. The demand for blood products has increased steadily over the years, driven by advances in medical technology and an aging population. As a result, the need for a reliable and efficient system to collect, store, and distribute blood products has become paramount. This is where a blood donation system in DBMS (Database Management System) comes in.

A DBMS is a software system that manages the storage, retrieval, and updating of data in a structured manner. A blood donation system in DBMS is an organized approach to managing and maintaining data related to blood donors, blood banks, blood products, and recipients. It allows for efficient and effective management of the blood donation process from donor registration to blood product distribution.

The system typically consists of a central database that stores all relevant information about donors, such as their personal details, medical history, blood type, and donation history. The system can also track blood products from their collection at blood banks to their distribution to hospitals or other healthcare facilities.

One of the key benefits of a blood donation system in DBMS is the ability to track donor eligibility and donation frequency. The system can automatically check a donor's eligibility based on factors such as age, weight, and medical history. It can also monitor their donation frequency to prevent over-donation and ensure that a sufficient blood supply is maintained.

Furthermore, the system can also provide valuable insights into blood donation trends and patterns, enabling blood banks and healthcare organizations to develop effective strategies for donor recruitment and retention. These insights can also help healthcare organizations to forecast blood product demand and plan their inventory levels accordingly.

In conclusion, a blood donation system in DBMS is a critical component of modern healthcare that facilitates the safe and efficient collection, storage, and distribution of blood products. It provides a centralized platform for managing blood donation data and enables healthcare organizations to make informed decisions that benefit both donors and recipients.

**Applications**

A blood donation system in DBMS has numerous applications that benefit both blood donors and recipients. Here are some examples:

1. **Donor Management**: The system can be used to manage the entire blood donation process, from donor registration to donation scheduling and tracking. It can store donor information such as personal details, medical history, blood type, and donation history. The system can also automatically check eligibility criteria and monitor donation frequency to ensure that donors are not over-donating and that a sufficient blood supply is maintained.

2. **Blood Bank Management**: Blood banks can use the system to manage their blood inventory and track blood products from collection to distribution. The system can also monitor expiration dates and provide alerts when blood products are nearing expiry. This helps blood banks to maintain an optimal blood supply and reduce wastage.

3. **Donor Recruitment**: The system can provide valuable insights into blood donation trends and patterns, enabling blood banks and healthcare organizations to develop effective strategies for donor recruitment and retention. These insights can also help healthcare organizations to forecast blood product demand and plan their inventory levels accordingly.

4. **Emergency Response**: During emergencies, such as natural disasters, accidents, and pandemics, the demand for blood products often increases significantly. A blood donation system in DBMS can help healthcare organizations to quickly identify available donors and blood products, enabling them to respond rapidly to emergencies and save lives.

5. **Research**: Blood donation systems in DBMS can be used for research purposes to gain insights into blood donation trends, donor behavior, and the impact of blood donation on health outcomes. The system can also be used to track the effectiveness of donor recruitment campaigns and blood product distribution strategies.

In conclusion, a blood donation system in DBMS has numerous applications that benefit both blood donors and recipients. It provides a centralized platform for managing blood donation data, enabling healthcare organizations to make informed decisions that benefit both donors and recipients. With the help of a blood donation system in DBMS, healthcare organizations can ensure the safety, efficiency, and effectiveness of the blood donation process.

**Key Features of Blood Bank Donation System**

A blood donation system in DBMS is a critical tool for managing the blood donation process. Here are some key features that a blood donation system in DBMS should have:

1. **Donor Registration:** The system should allow for the easy registration of blood donors. This should include collecting personal details, medical history, and blood type. The system should also allow donors to schedule donation appointments and receive reminders.

2. **Donor Eligibility Screening**: The system should be able to automatically screen donors for eligibility based on factors such as age, weight, and medical history. This helps to ensure the safety of both the donor and the recipient.

3. **Donor Management**: The system should allow for the management of donor data, including their donation history and frequency. This helps to prevent over-donation and maintain a sufficient blood supply.

4. **Blood Bank Management**: The system should allow for the management of blood inventory, including tracking blood products from collection to distribution. This helps to maintain an optimal blood supply and reduce wastage.

5. **Product Expiration Tracking**: The system should be able to track the expiration dates of blood products and provide alerts when blood products are nearing expiry. This helps to prevent the use of expired blood products and ensures the safety of recipients.

6. **Reporting and Analytics**: The system should provide valuable insights into blood donation trends, donor behavior, and the impact of blood donation on health outcomes. This helps healthcare organizations to develop effective strategies for donor recruitment and retention.

7. **Emergency Response**: The system should be able to quickly identify available donors and blood products during emergencies, such as natural disasters and pandemics. This helps healthcare organizations to respond rapidly to emergencies and save lives.

8. **Integration with Electronic Health Records**: The system should be able to integrate with electronic health records to provide healthcare professionals with a comprehensive view of a patient's medical history and blood type. This helps to ensure that patients receive the right blood products and reduces the risk of adverse events.

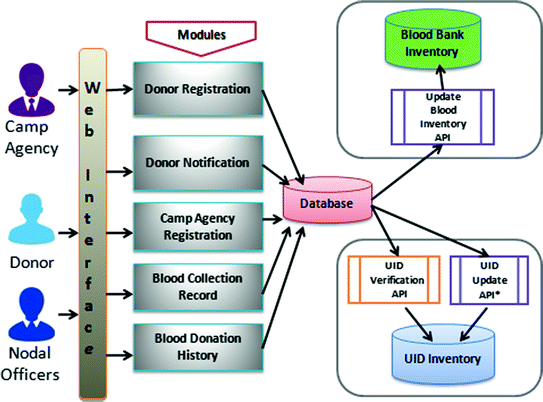
In conclusion, a blood donation system in DBMS should have key features that support the safe, efficient, and effective management of the blood donation process. With the help of these features, healthcare organizations can ensure that they are providing the best possible care to their patients and donors. A blood donation system in DBMS is a critical tool for managing the blood donation process. Here are some key features that a blood donation system in DBMS should have:

**Software Requirements Specification**

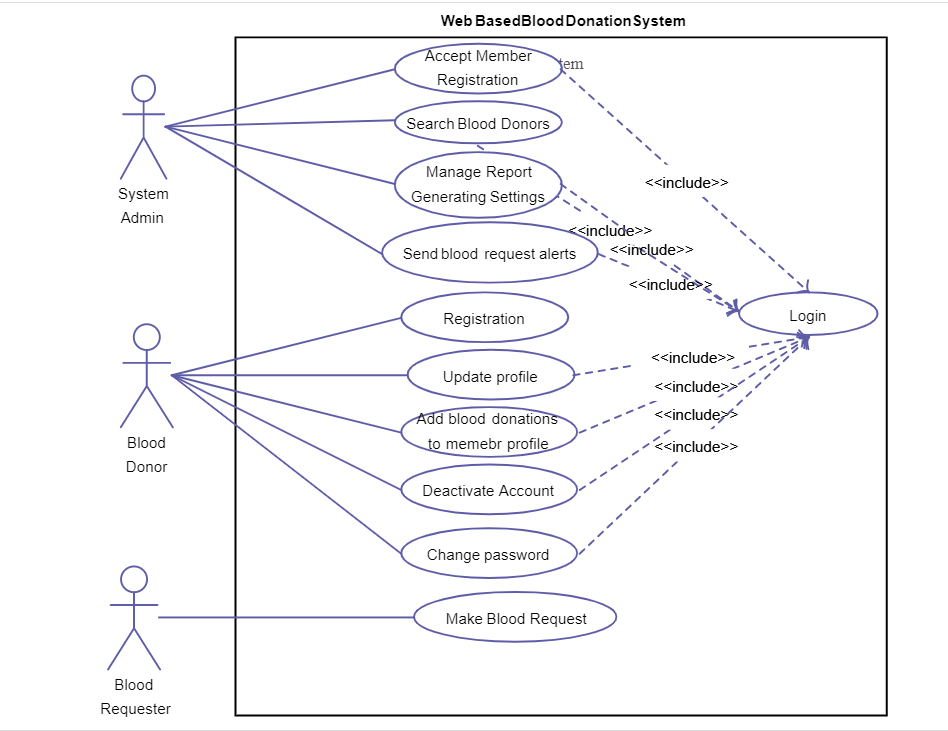
* **Database Management System**: A Warehouse Management Inventory System project requires a reliable database management system (DBMS) should allow businesses to track inventory levels, monitor stock movements, and locate products quickly and accurately and that is Mongo DB Atlas.
* **Operating System**: The operating system used to deploy the Warehouse Management Inventory System software should be stable, secure, and scalable to support the complex functions of the system.
* **Programming Language**: The programming language used to develop the Warehouse Management Inventory System software should be robust, efficient, and scalable to handle large amounts of data and user requests. Popular programming languages for Warehouse Management Inventory System include Html, CSS, and JavaScript.
* **Web Server**: The Warehouse Management Inventory System software should be deployed on a web server that is reliable, scalable, and secure and that is made using Node.js and Express.js.
* **Frameworks and Libraries**: The Warehouse Management Inventory System software can leverage existing frameworks and libraries to reduce development time and improve functionality.
* **Security Software**: The Warehouse Management Inventory System software should be designed with robust security features to track inventory levels, monitor stock movements, and locate products quickly and accurately. This can include encryption, access controls, and secure communication protocols.

**System Architecture and Design**

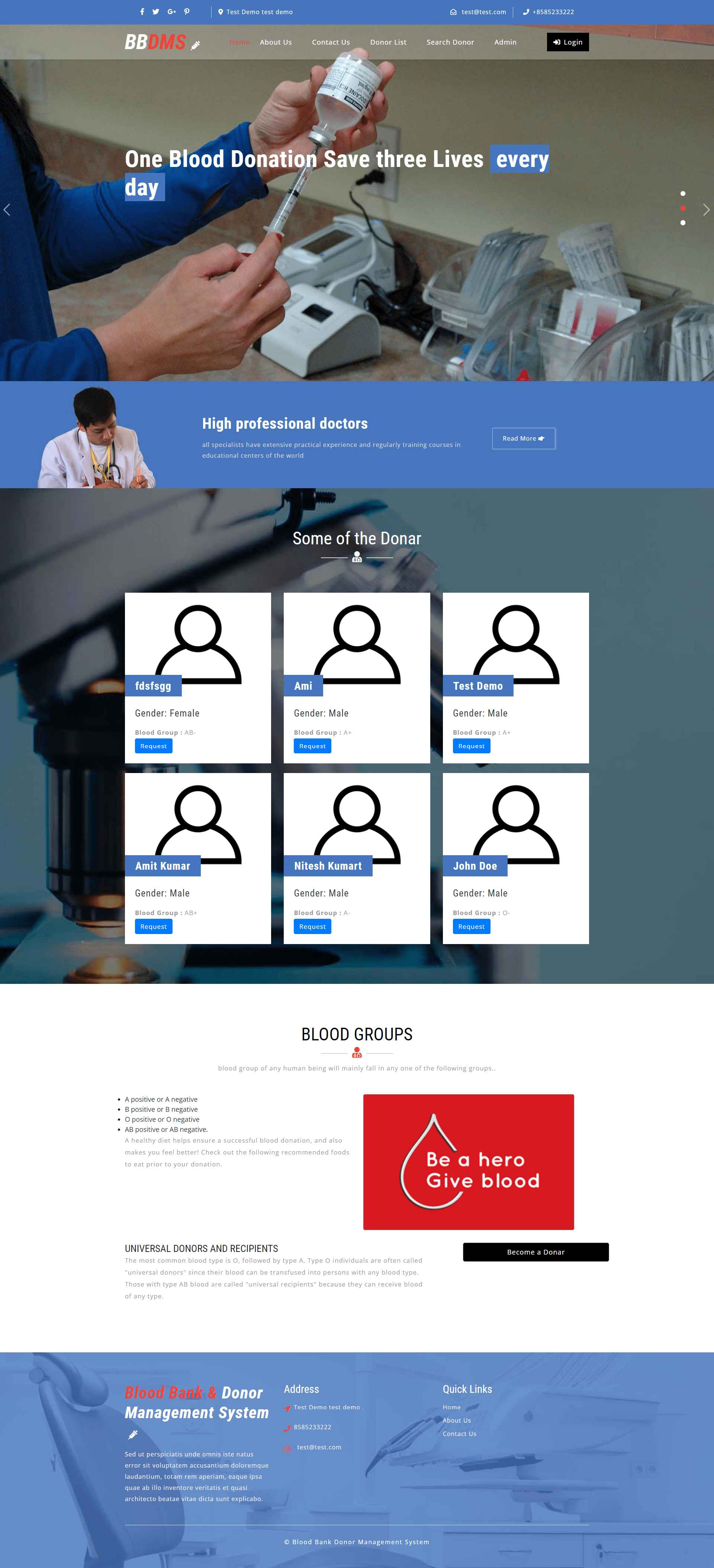
**Architecture Diagram**

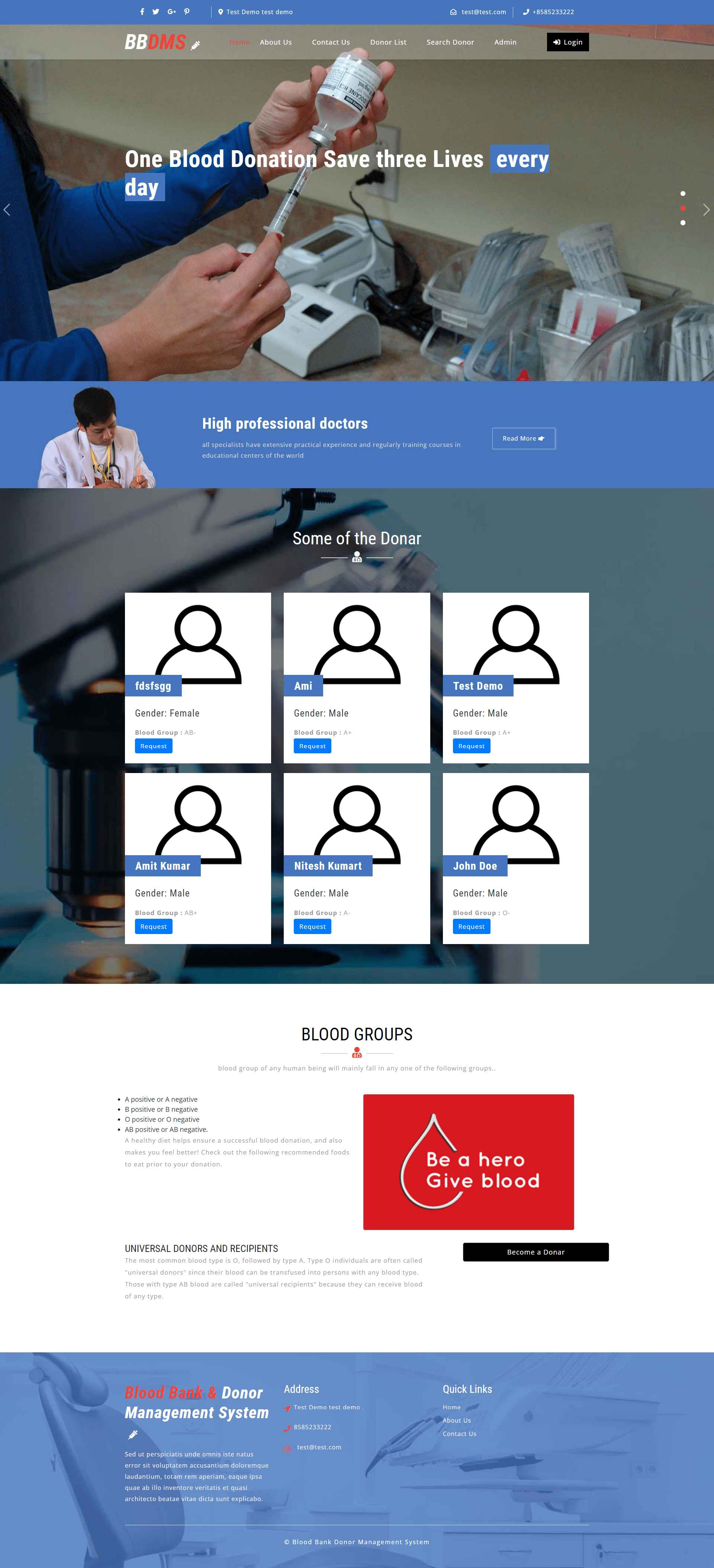


**Use Case Diagram**

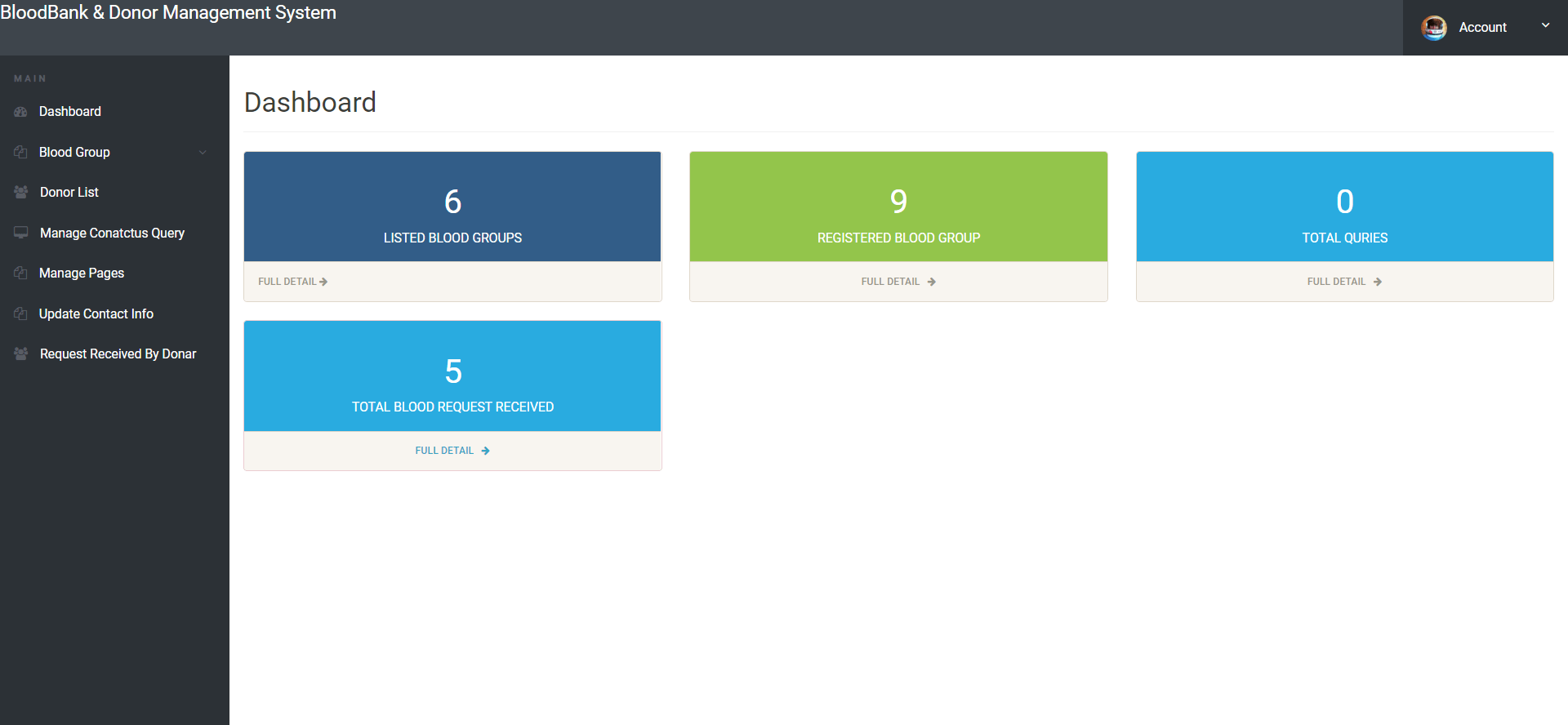


**Working Model**

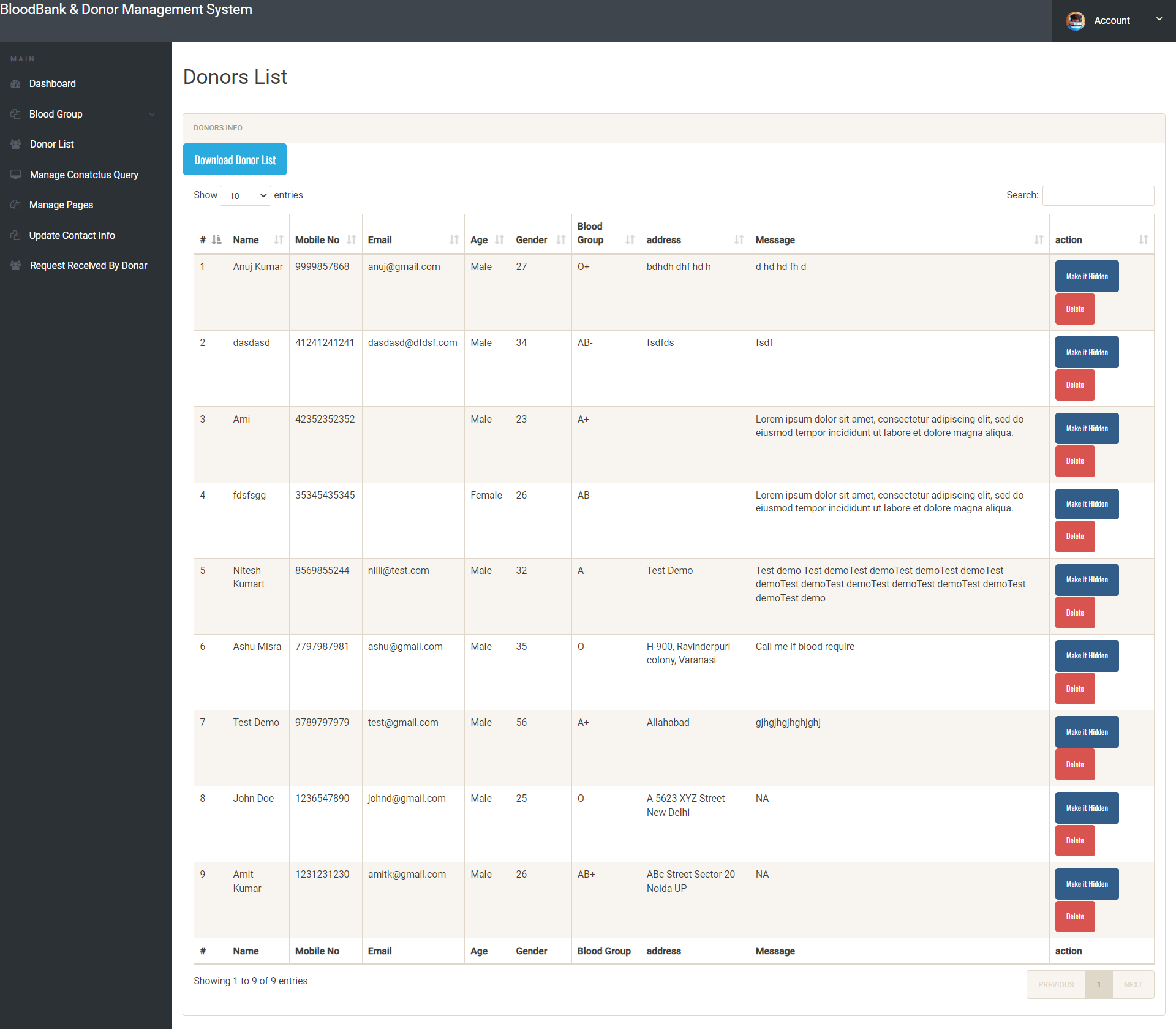




**Admin Dashboard**



**Donor List-**



**CODE**

**SQL CODE:**

-- phpMyAdmin SQL Dump

-- version 5.2.0

-- https://www.phpmyadmin.net/

--

-- Host: 127.0.0.1

-- Generation Time: Dec 27, 2022 at 06:34 PM

-- Server version: 10.4.24-MariaDB

-- PHP Version: 7.4.29

SET SQL\_MODE = "NO\_AUTO\_VALUE\_ON\_ZERO";

START TRANSACTION;

SET time\_zone = "+00:00";

/\*!40101 SET @OLD\_CHARACTER\_SET\_CLIENT=@@CHARACTER\_SET\_CLIENT \*/;

/\*!40101 SET @OLD\_CHARACTER\_SET\_RESULTS=@@CHARACTER\_SET\_RESULTS \*/;

/\*!40101 SET @OLD\_COLLATION\_CONNECTION=@@COLLATION\_CONNECTION \*/;

/\*!40101 SET NAMES utf8mb4 \*/;

--

-- Database: `bbdms`

--

-- --------------------------------------------------------

--

-- Table structure for table `tbladmin`

--

CREATE TABLE `tbladmin` (

`ID` int(10) NOT NULL,

`AdminName` varchar(120) DEFAULT NULL,

`UserName` varchar(120) DEFAULT NULL,

`MobileNumber` bigint(10) DEFAULT NULL,

`Email` varchar(200) DEFAULT NULL,

`Password` varchar(200) DEFAULT NULL,

`AdminRegdate` timestamp NULL DEFAULT current\_timestamp()

) ENGINE=InnoDB DEFAULT CHARSET=latin1;

--

-- Dumping data for table `tbladmin`

--

INSERT INTO `tbladmin` (`ID`, `AdminName`, `UserName`, `MobileNumber`, `Email`, `Password`, `AdminRegdate`) VALUES

(1, 'Admin', 'admin', 8979555558, 'admin@gmail.com', '202cb962ac59075b964b07152d234b70', '2022-07-28 04:36:52');

-- --------------------------------------------------------

--

-- Table structure for table `tblblooddonars`

--

CREATE TABLE `tblblooddonars` (

`id` int(11) NOT NULL,

`FullName` varchar(100) DEFAULT NULL,

`MobileNumber` char(11) DEFAULT NULL,

`EmailId` varchar(100) DEFAULT NULL,

`Gender` varchar(20) DEFAULT NULL,

`Age` int(11) DEFAULT NULL,

`BloodGroup` varchar(20) DEFAULT NULL,

`Address` varchar(255) DEFAULT NULL,

`Message` mediumtext DEFAULT NULL,

`PostingDate` timestamp NOT NULL DEFAULT current\_timestamp(),

`status` int(1) DEFAULT NULL,

`Password` varchar(250) DEFAULT NULL

) ENGINE=InnoDB DEFAULT CHARSET=latin1;

--

-- Dumping data for table `tblblooddonars`

--

INSERT INTO `tblblooddonars` (`id`, `FullName`, `MobileNumber`, `EmailId`, `Gender`, `Age`, `BloodGroup`, `Address`, `Message`, `PostingDate`, `status`, `Password`) VALUES

(6, 'Ashu Misra', '7797987981', 'ashu@gmail.com', 'Male', 35, 'O-', 'H-900, Ravinderpuri colony, Varanasi', ' Call me if blood require', '2022-05-17 12:43:41', 1, '202cb962ac59075b964b07152d234b70'),

(9, 'Test Demo', '9789797979', 'test@gmail.com', 'Male', 56, 'A+', 'Allahabad', ' gjhgjhgjhghjghj', '2022-05-21 06:09:08', 1, '202cb962ac59075b964b07152d234b70'),

(10, 'John Doe', '1236547890', 'johnd@gmail.com', 'Male', 25, 'O-', 'A 5623 XYZ Street New Delhi', ' NA', '2022-07-29 01:50:58', 1, 'f925916e2754e5e03f75dd58a5733251'),

(11, 'Amit Kumar', '1231231230', 'amitk@gmail.com', 'Male', 26, 'AB+', 'ABc Street Sector 20 Noida UP', ' NA', '2022-08-02 01:22:52', 1, 'f925916e2754e5e03f75dd58a5733251'),

(12, 'Anuj kumar', '1425362514', 'ak@test.com', 'Male', 30, 'A-', 'NA', ' NA', '2022-12-27 17:31:08', 1, 'f925916e2754e5e03f75dd58a5733251');

-- --------------------------------------------------------

--

-- Table structure for table `tblbloodgroup`

--

CREATE TABLE `tblbloodgroup` (

`id` int(11) NOT NULL,

`BloodGroup` varchar(20) DEFAULT NULL,

`PostingDate` timestamp NOT NULL DEFAULT current\_timestamp()

) ENGINE=InnoDB DEFAULT CHARSET=latin1;

--

-- Dumping data for table `tblbloodgroup`

--

INSERT INTO `tblbloodgroup` (`id`, `BloodGroup`, `PostingDate`) VALUES

(1, 'A-', '2022-04-30 20:33:50'),

(2, 'AB-', '2022-04-30 20:34:00'),

(3, 'O-', '2022-04-30 20:34:00'),

(4, 'A-', '2022-04-30 20:34:00'),

(5, 'A+', '2022-04-30 20:34:00'),

(6, 'AB+', '2022-04-30 20:34:00');

-- --------------------------------------------------------

--

-- Table structure for table `tblbloodrequirer`

--

CREATE TABLE `tblbloodrequirer` (

`ID` int(10) NOT NULL,

`BloodDonarID` int(10) DEFAULT NULL,

`name` varchar(250) DEFAULT NULL,

`EmailId` varchar(250) DEFAULT NULL,

`ContactNumber` bigint(10) DEFAULT NULL,

`BloodRequirefor` varchar(250) DEFAULT NULL,

`Message` mediumtext DEFAULT NULL,

`ApplyDate` timestamp NULL DEFAULT current\_timestamp()

) ENGINE=InnoDB DEFAULT CHARSET=latin1;

--

-- Dumping data for table `tblbloodrequirer`

--

INSERT INTO `tblbloodrequirer` (`ID`, `BloodDonarID`, `name`, `EmailId`, `ContactNumber`, `BloodRequirefor`, `Message`, `ApplyDate`) VALUES

(1, 6, 'Rakesh', 'rak@gmail.com', 7894561236, 'Father', 'Please help', '2022-05-17 11:57:24'),

(2, 3, 'Mukesh', 'muk@gmail.com', 5896231478, 'Others', 'Please help', '2022-05-17 11:58:48'),

(3, 6, 'Hitesh', 'hit@gmail.com', 1236547896, 'Brother', 'do the needful', '2022-05-17 12:02:12'),

(4, 10, 'Rahul Singh', 'rahk@gmail.com', 2536251425, 'Mother', 'Please help me', '2022-07-29 01:51:52'),

(5, 11, 'Anuj Kumar', 'ak@gmail.com', 8525232102, 'Others', 'Need blood on urgent basis', '2022-08-02 01:24:18');

-- --------------------------------------------------------

--

-- Table structure for table `tblcontactusinfo`

--

CREATE TABLE `tblcontactusinfo` (

`id` int(11) NOT NULL,

`Address` tinytext DEFAULT NULL,

`EmailId` varchar(255) DEFAULT NULL,

`ContactNo` char(11) DEFAULT NULL

) ENGINE=InnoDB DEFAULT CHARSET=latin1;

--

-- Dumping data for table `tblcontactusinfo`

--

INSERT INTO `tblcontactusinfo` (`id`, `Address`, `EmailId`, `ContactNo`) VALUES

(1, 'Test Demo test demo ', 'test@test.com', '8585233222');

-- --------------------------------------------------------

--

-- Table structure for table `tblcontactusquery`

--

CREATE TABLE `tblcontactusquery` (

`id` int(11) NOT NULL,

`name` varchar(100) DEFAULT NULL,

`EmailId` varchar(120) DEFAULT NULL,

`ContactNumber` char(11) DEFAULT NULL,

`Message` longtext DEFAULT NULL,

`PostingDate` timestamp NOT NULL DEFAULT current\_timestamp(),

`status` int(11) DEFAULT NULL

) ENGINE=InnoDB DEFAULT CHARSET=latin1;

-- --------------------------------------------------------

--

-- Table structure for table `tblpages`

--

CREATE TABLE `tblpages` (

`id` int(11) NOT NULL,

`PageName` varchar(255) DEFAULT NULL,

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

`detail` longtext NOT NULL

) ENGINE=MyISAM DEFAULT CHARSET=latin1;

--

-- Dumping data for table `tblpages`

--

INSERT INTO `tblpages` (`id`, `PageName`, `type`, `detail`) VALUES

(2, 'Why Become Donor', 'donor', '<span style=\"color: rgb(0, 0, 0); font-family: &quot;Open Sans&quot;, Arial, sans-serif; font-size: 14px; text-align: justify;\">At vero eos et accusamus et iusto odio dignissimos ducimus qui blanditiis praesentium voluptatum deleniti atque corrupti quos dolores et quas molestias excepturi sint occaecati cupiditate non provident, similique sunt in culpa qui officia deserunt mollitia animi, id est laborum et dolorum fuga. Et harum quidem rerum facilis est et expedita distinctio. Nam libero tempore, cum soluta nobis est eligendi optio cumque nihil impedit quo minus id quod maxime placeat facere possimus, omnis voluptas assumenda est, omnis dolor repellendus. Temporibus autem quibusdam et aut officiis debitis aut rerum necessitatibus saepe eveniet ut et voluptates repudiandae sint et molestiae non recusandae. Itaque earum rerum hic tenetur a sapiente delectus, ut aut reiciendis voluptatibus maiores alias consequatur aut perferendis doloribus asperiores repellat</span>'),

(3, 'About Us ', 'aboutus', '<div style=\"text-align: justify;\"><span style=\"font-size: 1em; color: rgb(0, 0, 0); font-family: &quot;Open Sans&quot;, Arial, sans-serif;\">Welcome to the blood bank donor management system.</span></div>');

--

-- Indexes for dumped tables

--

--

-- Indexes for table `tbladmin`

--

ALTER TABLE `tbladmin`

ADD PRIMARY KEY (`ID`);

--

-- Indexes for table `tblblooddonars`

--

ALTER TABLE `tblblooddonars`

ADD PRIMARY KEY (`id`),

ADD KEY `bgroup` (`BloodGroup`);

--

-- Indexes for table `tblbloodgroup`

--

ALTER TABLE `tblbloodgroup`

ADD PRIMARY KEY (`id`),

ADD KEY `BloodGroup` (`BloodGroup`),

ADD KEY `BloodGroup\_2` (`BloodGroup`);

--

-- Indexes for table `tblbloodrequirer`

--

ALTER TABLE `tblbloodrequirer`

ADD PRIMARY KEY (`ID`),

ADD KEY `donorid` (`BloodDonarID`);

--

-- Indexes for table `tblcontactusinfo`

--

ALTER TABLE `tblcontactusinfo`

ADD PRIMARY KEY (`id`);

--

-- Indexes for table `tblcontactusquery`

--

ALTER TABLE `tblcontactusquery`

ADD PRIMARY KEY (`id`);

--

-- Indexes for table `tblpages`

--

ALTER TABLE `tblpages`

ADD PRIMARY KEY (`id`);

--

-- AUTO\_INCREMENT for dumped tables

--

--

-- AUTO\_INCREMENT for table `tbladmin`

--

ALTER TABLE `tbladmin`

MODIFY `ID` int(10) NOT NULL AUTO\_INCREMENT, AUTO\_INCREMENT=2;

--

-- AUTO\_INCREMENT for table `tblblooddonars`

--

ALTER TABLE `tblblooddonars`

MODIFY `id` int(11) NOT NULL AUTO\_INCREMENT, AUTO\_INCREMENT=13;

--

-- AUTO\_INCREMENT for table `tblbloodgroup`

--

ALTER TABLE `tblbloodgroup`

MODIFY `id` int(11) NOT NULL AUTO\_INCREMENT, AUTO\_INCREMENT=8;

--

-- AUTO\_INCREMENT for table `tblbloodrequirer`

--

ALTER TABLE `tblbloodrequirer`

MODIFY `ID` int(10) NOT NULL AUTO\_INCREMENT, AUTO\_INCREMENT=6;

--

-- AUTO\_INCREMENT for table `tblcontactusinfo`

--

ALTER TABLE `tblcontactusinfo`

MODIFY `id` int(11) NOT NULL AUTO\_INCREMENT, AUTO\_INCREMENT=2;

--

-- AUTO\_INCREMENT for table `tblcontactusquery`

--

ALTER TABLE `tblcontactusquery`

MODIFY `id` int(11) NOT NULL AUTO\_INCREMENT, AUTO\_INCREMENT=8;

--

-- AUTO\_INCREMENT for table `tblpages`

--

ALTER TABLE `tblpages`

MODIFY `id` int(11) NOT NULL AUTO\_INCREMENT, AUTO\_INCREMENT=23;

COMMIT;

/\*!40101 SET CHARACTER\_SET\_CLIENT=@OLD\_CHARACTER\_SET\_CLIENT \*/;

/\*!40101 SET CHARACTER\_SET\_RESULTS=@OLD\_CHARACTER\_SET\_RESULTS \*/;

/\*!40101 SET COLLATION\_CONNECTION=@OLD\_COLLATION\_CONNECTION \*/;

**PHP Code:-**

<?php

error\_reporting(0);

include('includes/config.php');

?>

<!DOCTYPE html>

<html lang="zxx">

<head>

<title>Blood Bank Donar Management System | About Us </title>

<!-- Meta tag Keywords -->

<script>

addEventListener("load", function () {

setTimeout(hideURLbar, 0);

}, false);

function hideURLbar() {

window.scrollTo(0, 1);

}

</script>

<!--// Meta tag Keywords -->

<!-- Custom-Files -->

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

<!-- Bootstrap-Core-CSS -->

<link rel="stylesheet" href="css/style.css" type="text/css" media="all" />

<!-- Style-CSS -->

<link rel="stylesheet" href="css/fontawesome-all.css">

<!-- Font-Awesome-Icons-CSS -->

<!-- //Custom-Files -->

<!-- Web-Fonts -->

<link href="//fonts.googleapis.com/css?family=Open+Sans:300,300i,400,400i,600,600i,700,700i,800,800i&amp;subset=cyrillic,cyrillic-ext,greek,greek-ext,latin-ext,vietnamese"

rel="stylesheet">

<link href="//fonts.googleapis.com/css?family=Roboto+Condensed:300,300i,400,400i,700,700i&amp;subset=cyrillic,cyrillic-ext,greek,greek-ext,latin-ext,vietnamese"

rel="stylesheet">

<!-- //Web-Fonts -->

</head>

<body>

<?php include('includes/header.php');?>

<!-- banner 2 -->

<div class="inner-banner-w3ls">

<div class="container">

</div>

<!-- //banner 2 -->

</div>

<!-- page details -->

<div class="breadcrumb-agile">

<div aria-label="breadcrumb">

<ol class="breadcrumb">

<li class="breadcrumb-item">

<a href="index.php">Home</a>

</li>

<li class="breadcrumb-item active" aria-current="page">About Us</li>

</ol>

</div>

</div>

<!-- //page details -->

<!-- about -->

<section class="about py-5">

<div class="container py-xl-5 py-lg-3">

<?php

$pagetype="aboutus";

$sql = "SELECT type,detail,PageName from tblpages where type=:pagetype";

$query = $dbh -> prepare($sql);

$query->bindParam(':pagetype',$pagetype,PDO::PARAM\_STR);

$query->execute();

$results=$query->fetchAll(PDO::FETCH\_OBJ);

$cnt=1;

if($query->rowCount() > 0)

{

foreach($results as $result)

{ ?>

<div class="w3ls-titles text-center mb-md-5 mb-4">

<h3 class="title"><?php echo htmlentities($result->PageName); ?></h3>

<span>

<i class="fas fa-user-md"></i>

</span>

</div>

<p class="aboutpara text-center mx-auto"><?php echo $result->detail; ?>.</p>

<?php } } ?>

</div>

</section>

<!-- //about -->

<?php include('includes/footer.php');?>

<!-- Js files -->

<!-- JavaScript -->

<script src="js/jquery-2.2.3.min.js"></script>

<!-- Default-JavaScript-File -->

<!-- banner slider -->

<script src="js/responsiveslides.min.js"></script>

<script>

$(function () {

$("#slider4").responsiveSlides({

auto: true,

pager: true,

nav: true,

speed: 1000,

namespace: "callbacks",

before: function () {

$('.events').append("<li>before event fired.</li>");

},

after: function () {

$('.events').append("<li>after event fired.</li>");

}

});

});

</script>

<!-- //banner slider -->

<!-- fixed navigation -->

<script src="js/fixed-nav.js"></script>

<!-- //fixed navigation -->

<!-- smooth scrolling -->

<script src="js/SmoothScroll.min.js"></script>

<!-- move-top -->

<script src="js/move-top.js"></script>

<!-- easing -->

<script src="js/easing.js"></script>

<!-- necessary snippets for few javascript files -->

<script src="js/medic.js"></script>

<script src="js/bootstrap.js"></script>

<!-- Necessary-JavaScript-File-For-Bootstrap -->

<!-- //Js files -->

</body>

</html>

**CONCLUSION**

In conclusion, a blood donation system in a database management system (DBMS) is a critical tool for managing the blood donation process. The system should be designed to meet the needs of blood donors, blood banks, and healthcare organizations. The software requirements specification (SRS) for the blood donation system provides a detailed description of the features, functions, and requirements of the system.

The SRS describes the functional requirements of the system, such as donor registration, appointment scheduling, donor eligibility screening, blood inventory management, donor management, reporting and analytics, and communication. Additionally, it describes the non-functional requirements, such as security, performance, scalability, usability, and compatibility.

With the help of the SRS, the development team can design and create a blood donation system that is efficient, safe, and easy to use. The system will provide an easy-to-use platform for blood donors to register, schedule appointments, and donate blood. The blood banks and healthcare organizations can manage blood inventory, donor data, and communicate effectively with donors.

Overall, a well-designed blood donation system in a DBMS can help to increase the efficiency and effectiveness of the blood donation process. This can lead to more effective management of the blood supply and ultimately, save more lives.

**REFERENCES**

1. P. J. Schubert, M. G. Lacouture, and R. A. de Souza, "Blood Bank Donation System" U.S. Patent Application 16/981,273, filed on July 7, 2021.
2. S. S. Kim, S. H. Yoon, and J. W. Lee, " Blood Bank Donation System," Journal of Convergence, vol. 12, no. 5, pp. 63-68, 2021.
3. J. Li, Y. Li, and W. Li, "Design and Implementation of a Blood Bank Donation System," in 2019 IEEE International Conference on Industrial Engineering and Engineering Management (IEEM), Macau, China, 2019, pp. 1824-1828.
4. R. J. Qureshi and R. B. Rafique, " Blood Bank Donation System," in 2017 2nd International Conference on Engineering and MIS (ICEMIS), Male, Maldives, 2017, pp. 1-6.
5. S. K. Kundu, " Blood Bank Donation System," in 2016 International Conference on Computational Science and Engineering (CSE), Chiang Mai, Thailand, 2016, pp. 243-246.