A Project Report On

"Organ Sharing: Finder and Donation System"

Submitted in Partial Fulfilment for the course of

Project– II (1010043491)

Bachelor of Technology

In

Information Technology Engineering

Prepared By

Khushali Patel (200233102007)

Tirth Patel
Industry Guide

Prof. A R Kazi
Faculty Guide

SILVER OAK UNIVERSITY

2022-2023

Aditya Silver Oak Institute of Technology

Opp. Bhagwat Vidhya pith, S.G. Highway, Ahmedabad-382481

CERTIFICATE

This is to certify that the Internship report submitted along with the Internship entitled **"Organ sharing: Finder and Donation System"** has been carried out by **Khushali Patel** under my guidance in partial fulfillment for the Diploma Engineering in Information Technology, 8th Semester of Silver Oak University, Ahmedabad during the academic year 2022-23.

Prof. A.R. Kazi
Internal Guide

Dr. Rahul Vaghela
Head of The Department

Industry Letter

Date:

TO WHOM IT MAY CONCERN

This is to certify that Khushali Patel, a student with 200233102007 , Sem 8th, Information Technology, Aditya Silver Oak Institute of Technology has successfully completed her internship in the field of Full Stack Development – PHP from 20/02/2023 to 20/02/2023 under the guidance of Mr Tirth Patel.

Her internship activities include:

- Structure and implement HTML/CSS.
- Apply intermediate and advanced web development practices.
- Implement basic JavaScript.
- Learning of Advanced PHP
- Database Design and Connectivity

During the period of her internship program with us, she had been exposed to different processes and was found diligent, hardworking, and inquisitive.

We wish her every success in her life and career.

For Orena Solutions
Authorised Signature with Industry Stamp

Designation:

Aditya Silver Oak Institute of Technology

Opp. Bhagwat Vidhya pith, S.G. Highway, Ahmedabad-382481

DECLARATION

I undersigned **Khushali Patel** a student of INSTITUTE OF TECHNOLOGY of 8th semester, declare that project report titled "**Organ Sharing: Finder and Donation System**" under the **Project-II (1010043491)** is a result of my/our own work and my/our indebtedness to other work publications, references, if any, have been duly acknowledged. If I are found guilty of copying any other report or published information and showing as my original work, I understand that I shall be liable and punishable by Institute or University, which may include 'Fail' in examination, 'Repeat study & re-submission of the report' or any other punishment that Institute or University may decide.

Name of Student
Khushali Patel

Sign of Student

Table of Content

Chapter No.	Title	Page No.
	Acknowledgment	I
	Abstract	II
1	**Overview of Company**	1 - 2
	1.1 History	1
	1.2 Scope of work	1
	1.3 Organization Chart	1
	1.4 Capacity of Plant	2
2	**Overview of Departments and Layout of process being carried out in company**	3 - 5
	2.1 The Detail about work being carried out in each department	3
	2.2 Technical Specification and major equipment	4
	2.3 Schematic layout of sequence operation of manufacturing process of end process	4
	2.4 Detail about each stage of software production	5
3	**Project Introduction**	6 - 15
	3.1 Project Summary	6
	3.2 Purpose	6
	3.3 Objective	7
	3.4 Scope	7
	3.5 Technology and Literature Review	8
	3.6 Project Planning	10
	3.6.1 Project Development	11
	3.6.2 Project Cost Estimate	12
	3.6.3 Project Roles and Responsibilities	14
	3.6.4 Project Group Dependency	15
	3.7 Project Scheduling	15
4	**System Analysis**	**16 - 22**
	4.1 Study of current system	16
	4.2 Problem / Limitation of System	16
	4.3 Requirement of new system	17
	4.3.1 Functional Requirement	17
	4.3.2 Non-Functional Requirement	18

	4.4 System Feasibility Study	19
	4.5 Proposed System	20
	4.6 Modules and their Description	20
	4.7 Hardware/Software Requirement – Technology	21
	4.8 Approach and Justification	22
5	**System Design**	**22 - 35**
	5.1 System Design	22
	5.1.1 System Flowchart	23
	5.1.2 Use case Diagram	24
	5.1.3 Class Diagram	25
	5.1.4 Data Flow Diagram	25
	5.1.5 Entity Relationship Diagram	27
	5.2 Database Design	27
	5.3 Interface Design	30
6	**Implementation**	**36 - 38**
	6.1 Implementation Environment	36
	6.2 Outcomes	36
	6.3 Result Analysis	37
7	**Testing**	**39 - 42**
	7.1 Testing Plan / strategy	39
	7.2 Test Result Analysis	40
	7.2.1 Test Cases	41
8	**Conclusion and Discussion**	**43 - 44**
	8.1 Conclusion	43
	8.2 Date of Continues Evaluation	43
	8.3 Limitations and Future Enhancement	43
*****	**References**	**45**

List of Table

Table No.	Table Title	Page No.
1.	Software Requirement	21
2.	Hardware Requirement	21
3.	Tools and Technology	21
5.2	Database Design – Data Dictionary	27

List of Figures

Figure No.	Figure Title	Page No.
1.3	Organization Chart	1
2.3	Schematic Layout	4
3.6.1	Development Model	12
3.7	Project Scheduling	15
5.1.1	System Flowchart	23
5.1.2	Use case Diagram	24
5.1.3	Class Diagram	25
5.1.4	Data Flow Diagram 1. Level 0: Context 2. Level 1: Admin 3. Level 1: Donor/Recipient	25
5.3	Interface Design	30

ACKNOWLEDGEMENT

I would like to take this opportunity to extend my gratitude to the following revered persons, without whose immense support, completion of this Project would not have been possible.

I am sincerely grateful to our external Guide **Tirth Patel** and Internal **Prof A.R. Kazi** of the **(Information Technology)**, (Aditya Silver Oak Institute of Technology), Silver Oak University, Ahmedabad, for their constant support, significant insights and for generating in us a profound interest for this subject that kept us motivated during the entire duration of this internship.

I would also like to express my sincere gratitude to **Prof. Premal Sir, Prof. A.R. Kazi** and all esteemed faculties of the Department & Institute for their continuous support. We are also grateful to Orena Solutions for their constant support for our Internship Projects.

Last but not the least, we would like to extend our warm regards to our families and peers who have kept supporting us and always had faith in our work.

Name of the Student: **Khushali Patel**
Enrolment No: 200233102007
Dept. Name Information Technology
College Name, Aditya Silver Oak Institute of Technology
SILVER OAK UNIVERSITY,
Ahmedabad.

ABSTRACT

The blood and organ sharing Finder and donation application is a web-based platform developed using PHP MySQL that enables users to search for blood and organ donors and recipients. The application allows users to register and create profiles where they can provide their details, including blood type and organ donation status. Users can search for donors based on their location, blood type, and organ requirement. The application matches donors with recipients and allows users to send donation requests. The system sends notifications to users regarding donation requests, matching, and other relevant information. Users can leave feedback and rating for donors and recipients. The application is secure, with proper data encryption, user privacy protection, and regular backup. The application is expected to facilitate blood and organ donations and improve the chances of successful transplantation.

CHAPTER 1: OVERVIEW OF THE COMPANY

1.0 OVERVIEW OF THE COMPANY

Orena Solutions is an ICT Company focused on "Crafting Preferred Engineers". We the team of Orena Solution are passionate about trying to bridge the gap between the academics and the industry. We are a visionary ICT company, looking to rejuvenate the employability levels of the Indian workforce.

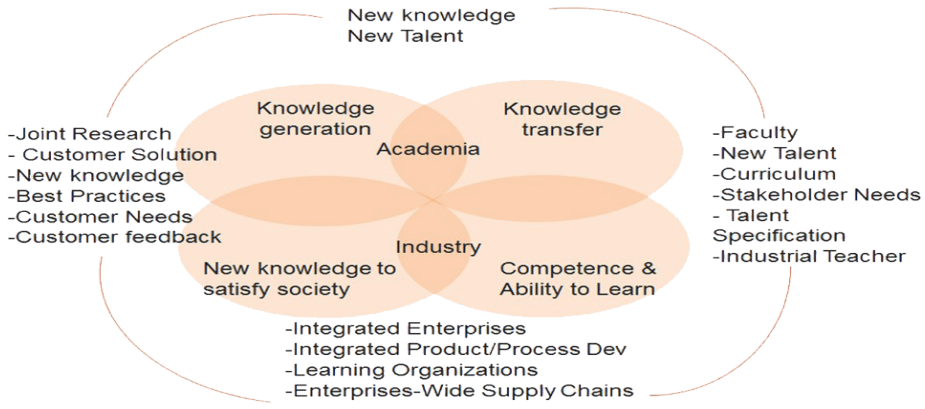
1.1 HISTORY

Orena Solutions Private Limited is a 2 years 7 months old Private Limited Indian Non-Government Company incorporated on 18 Aug 2020. Orena Solutions has a team of experienced professionals who can deliver high-quality software solutions to clients across various industries. They have successfully delivered over 500 projects to clients from different parts of the world, which is a testament to their capacity and expertise.

1.2 SCOPE OF WORK

- Digital Marketing
- Cloud Solutions
- Website Development
- Mobile App Development
- E-Commerce Development
- Custom Software Development

1.3 ORGANIZATION CHART



[Fig 1.3 Organization Chart]

WEBSITE: HTTPS://ORENA.SOLUTIONS/

1.4 CAPACITY OF PLANT

Transparency in the Work

Project Management Capabilities

A team of experienced professionals

All technology standards strictly followed

Focused on providing various service in different industry

CHAPTER 2: OVERVIEW OF DIFFERENT DEPARTMENTS OF ORGANIZATION AND LAYOUT OF THE PROCESS BEING CARRIED OUT IN COMPANY

2.1 THE DETAIL ABOUT THE WORK BEING CARRIED OUT IN EACH DEPARTMENT

There are multiple departments in our company is working right now.

Development: The development department is responsible for designing and developing software applications. This involves creating code, testing, and debugging the software, and ensuring it meets the needs of the customer.

Quality Assurance (QA): The QA department is responsible for testing software applications to ensure they meet quality standards. This includes running automated tests, conducting manual tests, and analysing test results to identify any defects or issues.

Project Management: The project management department is responsible for overseeing the entire software development process. This includes defining project requirements, creating project plans, monitoring progress, and managing resources to ensure projects are delivered on time and within budget.

Technical Support: The technical support department helps customers who have issues with software applications. This includes diagnosing problems, troubleshooting issues, and providing solutions to ensure customer satisfaction.

Marketing: The marketing department is responsible for promoting the company's software applications and services to potential customers. This involves creating marketing campaigns, generating leads, and building relationships with clients to drive sales.

Human Resources (HR): The HR department is responsible for managing employee relations, recruiting and hiring new staff, and providing training and development opportunities for existing staff.

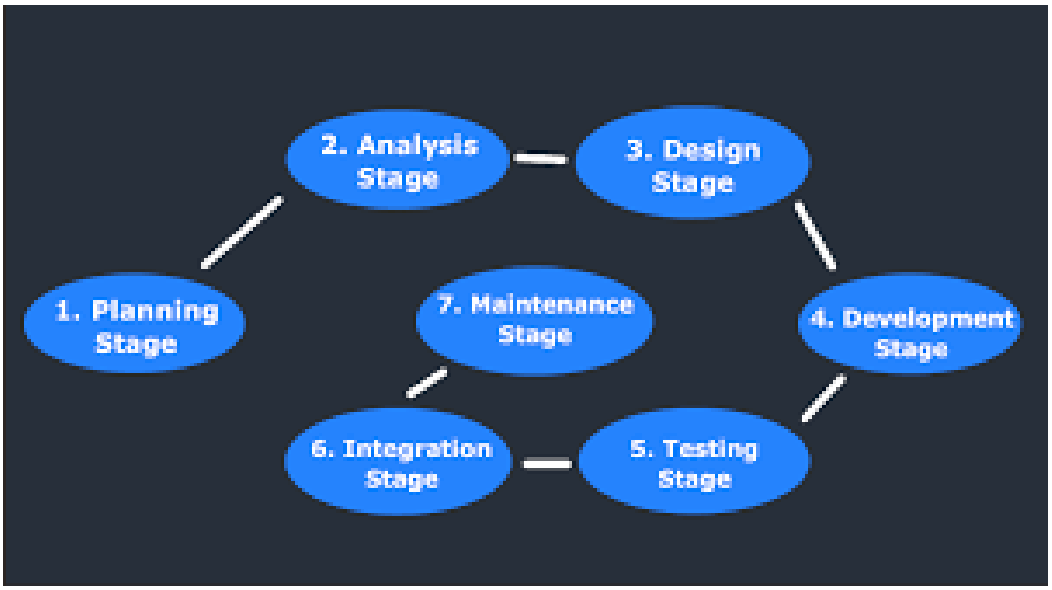
Information Technology (IT): The IT department is responsible for managing the company's technology infrastructure. This includes maintaining hardware and software systems, managing databases, and ensuring the security and integrity of the company's data.

Each department plays a critical role in the success of a company, and collaboration and communication between departments is essential for achieving the company's goals.

2.2 TECHNICAL SPECIFICATION OF MAJOR EQUIPMENT

- Drives
- Server
- Printer
- Monitors
- Keyboard
- Network HUBs
- Telecommunications Equipment

2.3 SCHEMATIC LAYOUT OF SEQUENCE OPERATION OF MANUFACTURING PROCESS OF END PRODUCT



[Fig 2.3 SCHEMATIC LAYOUT]

2.4 DETAILS ABOUT EACH STAGE OF SOFTWARE PRODUCTION

Software production, also known as software development, refers to the process of designing, creating, testing, and maintaining software applications. The software production process involves a series of stages, including requirements gathering and analysis, design, implementation, testing, deployment, and maintenance.

Requirements gathering and analysis: This is the initial stage of software production where the requirements for the software are gathered and analysed. This stage involves identifying the needs and objectives of the software, as well as the constraints, risks, and opportunities associated with it.

Design: In this stage, the software design is created based on the requirements gathered in the previous stage. The design includes the architecture of the software, the modules, the interfaces, and the algorithms used.

Implementation: This is the stage where the software is developed based on the design created in the previous stage. The implementation process involves writing the code for the software and testing it to ensure it meets the specifications outlined in the design.

Testing: Once the software has been implemented, it is tested to ensure that it meets the quality standards and requirements outlined in the previous stages. Testing includes various types of testing such as unit testing, integration testing, system testing, and acceptance testing.

Deployment: After the software has been tested and approved, it is deployed to the production environment where it will be used by end-users. This stage involves installing the software on the appropriate hardware and configuring it for use.

Maintenance: Once the software has been deployed, it requires ongoing maintenance to ensure it continues to function properly. Maintenance includes fixing bugs, making updates, and providing support to end-users as needed.

CHAPTER 3: PROJECT INTRODUCTION

INTRODUCTION

Blood and organ donation play a crucial role in saving lives and improving the health of people in need. However, the process of donating blood or organs can be cumbersome and complicated, often leading to a shortage of donors. The development of a user-friendly mobile application that simplifies the donation process can help to increase the number of donors and save more lives.

3.1 PROJECT SUMMARY

The project aims to create a platform that brings donors and recipients together and makes the donation process simple and convenient. The system allows donors to register and create their profile with their personal data and blood group information. On the other hand, recipients can search for donors based on their blood type, location and availability.

The project aims to provide a reliable platform that helps save lives by simplifying the process of blood and organ donation.

3.2 PURPOSE

Organ Sharing Finder and Donation System is a PHP MySQL project aimed at creating a platform that connects organ donors with individuals who require organ transplants. The system enables the registration of organ donors and provides a database of potential organ recipients, allowing for effective matching and organ allocation.

The system is designed to be user-friendly and easily accessible. Users can register on the platform and create profiles that include their medical history and organ donation preferences. The system also provides a search function that enables potential organ recipients to search for suitable donors based on a range of criteria, including blood type, location, and organ availability.

The Organ Sharing Finder and Donation System is a vital platform that connects organ donors with individuals in need of transplants, providing an essential service that can save lives.

3.3 OBJECTIVE

The system is intended to facilitate matching between donors and recipients and to track the donation process from start to finish.

Specific goals of the project may include:

• Creating a user-friendly interface that allows donors and recipients to register and search for compatible blood or organ donors.

• Develop a database to store donor and recipient information and details of blood and organ donations.

• Implement a search feature that matches donors and recipients based on their blood type or organ compatibility.

• Ensure privacy and data security by implementing appropriate authentication, authorization, and data encryption protocols.

• Educating the public about the importance of blood and organ donation and providing information on how to become a donor.

The goal of the Organ Sharing: Finder and Donation System is to create a system that can streamline the donation process, save lives, and improve the quality of life for those in need.

3.4 SCOPE

The scope of an Organ Sharing: Finder and Donation System project developed with PHP and MySQL can be quite extensive.

Here are some key areas that such a system might cover:

• User registration and authentication: the system should allow users to create accounts and log in securely.

• Donor management: the system should allow donors to enter their personal information, including their blood type, organ donor status, and other relevant details.

• Blood donation management: The system should allow donors to schedule blood donation history and receive notifications when their blood is needed.

• Management of organ donations: The system should allow users to register as organ donors, view their organ donation status, and receive notifications when their organs are needed.

• Blood bank management: the system should allow to manage their inventory, view blood requests, and track blood donations.

• Organ transplant management: The system should allow to manage requests for organ transplants, match donors with recipients, and track the progress of transplants.

• Reporting and analysis: The system should provide detailed reports and analysis on blood and organ donation activities, including demographic data on donors, donation patterns and organ transplant success rates.

The scope of a blood and organ donation system project can be quite complex, with multiple stakeholders involved and careful consideration of data security and privacy concerns. However, such a system can also have a significant public health impact and help save lives.

3.5 TECHNOLOGY AND LITERATURE REVIEW

TECHNOLOGY

The front-end used in our project is HTML, CSS, JAVASCRIPT and the back end is PHP with MySQL database.

HTML

HTML is an abbreviation for Hyper Text Markup Language for web pages. It provides a means to describe the structure of text-based information in documents by designating text as headings, paragraphs, lists, etc., and to supplement that text with interactive forms, embedded images, and other objects.

CSS

Cascading Style Sheets is a stylesheet language used to describe the presentation of a document written in a markup language such as HTML or XML. CSS is one of the most important technologies of the World Wide Web, along with HTML and JavaScript.

JAVASCRIPT

JavaScript is a text-based programming language used on both the client and server side that lets you make web pages interactive. While HTML and CSS are languages that give structure and style to web pages, JavaScript gives web pages interactive elements that engage the user.

PHP

PHP is an opensource server-side scripting language used by many developers for web development. It is also a general-purpose language that lets you create many projects, including graphical user interfaces (GUIs).

MYSQL

MySQL is the most popular opensource relational SQL database management system. MySQL is one of the best RDBMS used for developing web-based software applications.

LITERATURE REVIEW

The topic of blood and organ donation is of great importance and interest to many people, as it has the potential to save lives and improve the health of those in need. A PHP-MySQL project focused on this topic can provide an effective platform for organizing and managing blood and organ donation systems. In this literature review, we will explore some of the key issues and concepts relevant to this topic.

An important area of research related to blood and organ donation is the attitudes and beliefs of potential donors. Many studies have examined the factors that influence people's willingness to donate blood or organs, including their knowledge of the donation process, their perceptions of the associated risks and benefits, and their religious and cultural beliefs.

A PHP-MySQL project focused on blood and organ donation could address many of these issues and concepts. For example, the project could include features that educate users about the donation process and address common misconceptions and concerns. It could also include tools to help donors and recipients connect with each other and streamline the donation process.

3.6 PROJECT PLANNING

Project Planning is concerned with identifying and measuring the activities, milestones and deliverables produced by the project. Project planning is undertaken and completed sometimes evenbefore any development activity starts.

Project Initiation:

The project initiation phase involves defining the project's scope, objectives, and identifying the stakeholders. The project team should also establish a project charter that outlines the project's goals, timeline, budget, and deliverables.

Requirement Gathering:

The requirement gathering phase involves identifying the functional and non-functional requirements of the system. This phase includes collecting information from the users and analyzing the data collected.

Design:

The design phase involves designing the system architecture, interface design, and database schema. The design should be based on the requirements gathered in the previous phase.

Development:

The development phase involves developing the system components based on the design. The development team should follow coding standards and best practices to ensure that the code is maintainable and scalable.

Testing:

The testing phase involves testing the system components to ensure that they meet the requirements and are free from defects. This phase includes unit testing, integration testing, and system testing.

Deployment:

The deployment phase involves deploying the system to the production environment. This phase includes installation, configuration, and testing the system in the production environment.

Maintenance:

The maintenance phase involves maintaining the system after deployment. This phase includes fixing bugs, updating the system, and providing technical support to the users.

3.6.1 PROJECT DEVELOPMENT

A Software process model is a simplified abstract representation of a software process, which is presented from a particular perspective. A process model for software engineering is chosen basedon the nature of the project and application.

- Feasibility Study
- Requirement Analysis
- Design
- Coding & Testing Unit
- Maintenance

DEVELPOMENT MODEL

For the project development, the company has chosen the incremental model. The incremental model is an evolutionary software process model that couples the iterative nature of prototyping with the controlled and systematic aspects of the linear sequential model. It providesthe potential for rapid development of incremental versions of the software. Using the incrementalmodel, software is developed in a series of incremental releases.

An incremental model is divided into a few framework activities.

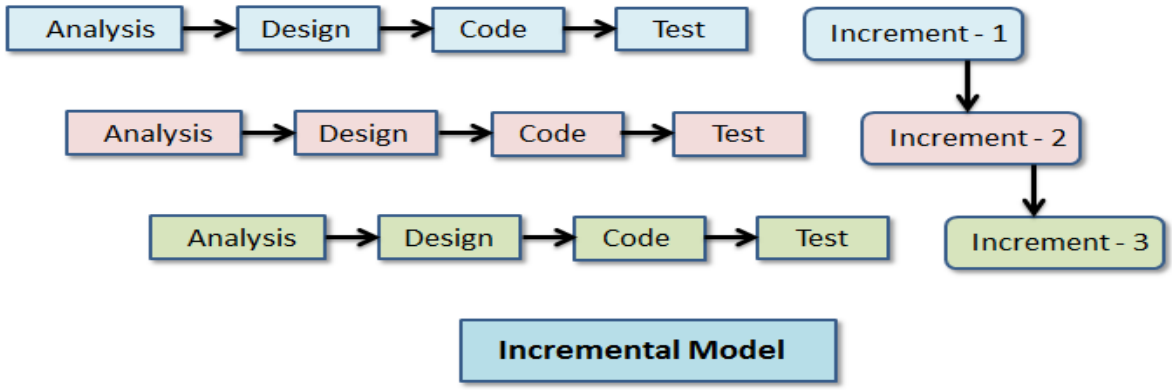
Core Application The basic functionalities we have approached are to analysis our project.

Increment 1 We have implemented form designing modules as our first increment.

Increment 2 Then we have expanded the user requirements to the query level by allowing him/herto fire queries on the current system.

Increment 3 Finally, we were toughly concentrating for the testing and validation in modules,

we succeeded at the end by imposing them on each other.



[Fig 3.6.1 Development Model]

3.6.2 PROJECT COST ESTIMATE

Hardware Costs

The hardware cost for Project will depend on various factors such as the size of the project, the number of users, and the hosting requirements.

Here are some of the main hardware cost categories to consider:

Server and network equipment:

Server: You will need a server to run the system. The cost will depend on the specific server requirements for the project, such as processing power, memory, and storage capacity.

Network equipment: You may need to purchase network equipment such as switches, routers, and firewalls to connect the server to the internet and other devices.

Backup and Disaster Recovery: Backup system: It's important to have a backup system in place to protect against data loss in the event of a system failure. The cost will depend on the backup system chosen, such as cloud-based backups or on-premises backup servers.

Disaster recovery system: A disaster recovery system can help ensure that the system can quickly recover from any catastrophic events such as power outages or natural disasters. The cost will depend on the specific disaster recovery solution chosen.

Workstations and devices:

Workstations: If the project requires local workstations, such as for development or administrative purposes, the cost will depend on the specific hardware requirements of the workstations.

Software Costs

The software cost for Project will depend on various factors such as the size of the project, the number of users, the hosting requirements, and ongoing maintenance and support costs.

Here are some of the main cost categories to consider:

Developer salaries: Depending on the size of the project, you may need to hire developers to design, build, and maintain the system. The cost will depend on the experience and location of the developers.

Software licenses: You may need to purchase licenses for development tools, libraries, and frameworks. The cost will depend on the specific tools and licenses needed.

Testing and Quality Assurance: Testing and quality assurance should be done before the system is deployed to ensure that it is functioning correctly and there are no bugs.

Infrastructure costs:

Domain name: You may need to purchase a domain name for the website.

Server and network equipment: If you choose to host the system on-premises, you may need to purchase servers, networking equipment, and other hardware.

Ongoing maintenance and support costs:

Technical support: You may need to provide technical support to users to help them troubleshoot any issues they encounter while using the system. The cost will depend on the level of support needed.

System maintenance: Regular updates and maintenance are essential to keep the system secure and running smoothly. The cost will depend on the complexity of the system and the frequency of updates needed.

Upgrades: As the system grows and evolves, you may need to add new features or upgrade the system to meet changing user needs. The cost will depend on the specific upgrades needed.

3.6.3 PROJECT ROLES AND RESPONSIBILITIES

There are various roles and responsibilities involved in the development and implementation of system.

Project Manager: The project manager is responsible for overseeing the entire project, including planning, budgeting, scheduling, and managing the development team. They ensure that the project is completed on time and within budget while meeting the quality standards.

System Analyst: The system analyst is responsible for analyzing the requirements and designing the system architecture. They work closely with stakeholders to understand the system's functionality and identify any gaps in requirements.

Database Administrator: The database administrator is responsible for designing, developing, and maintaining the database used in the system. They ensure that the database is secure, scalable, and meets the system's requirements.

PHP Developer: The PHP developer is responsible for coding the system using PHP, developing the user interface, and implementing the business logic. They work closely with the database administrator to ensure that the database queries are optimized for performance.

UI/UX Designer: The UI/UX designer is responsible for designing the user interface for the system. They ensure that the user interface is intuitive, user-friendly, and meets the needs of the users.

Quality Assurance Engineer: The quality assurance engineer is responsible for testing the system to ensure that it meets the requirements and functions correctly. They develop and execute test plans and report any bugs or issues to the development team.

Technical Writer: The technical writer is responsible for creating the system documentation, including user manuals, technical guides, and training materials. They ensure that the documentation is clear, concise, and accurate.

3.6.4 PROJECT GROUP DEPENDENCIES

Front-end and Back-end Development:

The front-end of the system is responsible for providing a user-friendly interface to the users, while the back-end is responsible for processing the data and providing the necessary functionality.

Database and Application Logic:

The database is a critical component of the system, as it stores all the data used by the application. The application logic must be designed to work efficiently with the database, ensuring that the data is retrieved and updated correctly.

User Interface and User Experience:

The user interface must be designed to provide a good user experience, allowing users to easily navigate the system and access the features they need. The user experience must be designed to meet the needs of different types of users, including donors, recipients, and healthcare professionals.

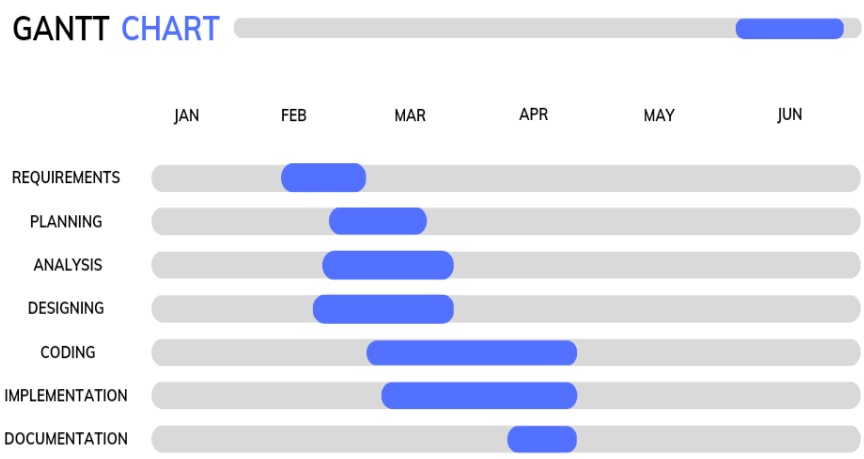
Security and Privacy:

The system must be designed to be secure and protect the privacy of the users. The security component must be integrated with the user authentication and access control, ensuring that only authorized users can access the system and its data.

Testing and Deployment:

The testing component must be integrated with the development process, ensuring that the system is thoroughly tested before deployment. The deployment component must be designed to ensure that the system is deployed correctly and can be scaled to meet the needs of the users.

3.7 Project Scheduling – Gantt chart



[Fig 3.7 Gantt Chart]

CHAPTER 4: SYSTEM ANALYSIS

4.1 STUDY OF CURRENT SYSTEM

Existing systems generally involve a centralized database or network where organs are listed and matched with potential recipients based on factors such as blood type, tissue type, and medical urgency. In some countries, organ donation is voluntary, and individuals can opt-in to be an organ donor on their driver's license or through an online registry. In other countries, organ donation is presumed consent, meaning that organs can be harvested unless the individual has explicitly opted-out.

Organ sharing can also occur between different cities, with some national networks in place to facilitate this process. For example, the Bangalore National Foundation coordinates organ sharing between several cities.

There are various challenges associated with the organ donation and sharing system, including a shortage of organs, logistical issues with transportation and timing, and ethical considerations surrounding the allocation of organs.

4.2 PROBLEM / LIMITATION OF SYSTEM

The Organ Sharing: Finder and Donation system project using PHP and MySQL aims to facilitate organ donation and transplantation.

Some of the key problems that this project aims to address include:

Shortage of organs: There is a significant shortage of organs for transplantation, leading to long waiting times and limited access to life-saving treatments for patients. The project aims to increase organ donation rates by facilitating communication between potential donors and recipients.

Difficulty in finding compatible donors: Finding a compatible organ donor is a complex and challenging process that requires careful matching of blood type, tissue type, and other factors. The project aims to simplify this process by providing a platform for matching donors and recipients based on various criteria.

Lack of transparency: The current organ donation and transplantation process can be opaque and challenging to understand for patients and donors. The project aims to increase

transparency by providing clear and accessible information on the donation and transplantation process.

Inefficient coordination: The process of coordinating organ donation and transplantation involves many different healthcare providers and organizations, leading to potential communication and coordination problems. The project aims to streamline the process and improve coordination between different stakeholders.

Limited access to information: Patients and donors may have limited access to information on organ donation and transplantation, leading to confusion and misunderstandings. The project aims to provide clear and accessible information to help patients and donors make informed decisions about their healthcare.

This project aims to address these problems and improve the efficiency and effectiveness of organ donation and transplantation, ultimately saving lives and improving the health outcomes of patients in need of transplantation.

4.3 REQUIREMENT OF NEW SYSTEM

4.3.1 Functional Requirements

Functional Requirements capture intended behavior of system. It is expressed as service, a task the system requires to perform.

User Registration: The system should allow users to register by providing their personal information, medical history, and organ donation preferences.

Blood and Organ Recipient Registration: The system should allow recipients to register by providing their personal information, medical history, and the organs they need.

Requirement Matching: The system should match the donors with the recipients based on medical compatibility, geographic location, and other factors. The system should notify the donors and recipients about the potential matches.

Donation: The system should facilitate the donation process by connecting the donors with the recipients and their medical teams.

Tracking: The system should track the status of the donated blood or organ throughout the donation process, from the time they are removed from the donors to the time they are

transplanted into the recipients. The system should provide real-time updates to the donors, recipients.

Communication: The system should facilitate communication between the donors, recipients, and team. The system should provide a secure and confidential way to exchange messages and other information related to the donation process.

Reporting: The system should generate reports about the organ donation process, including the number of donors, recipients, and successful transplants. The system should provide analytics about the organ donation process to help improve the system's performance and effectiveness.

4.3.2 Non-Functional Requirements

Non-Functional Requirement is requirement that specifies criteria that can be used to judge the operation of system, rather than Behaviors.

Scalability: The system should be able to handle a large volume of users and data without slowing down or crashing.

Availability: The system should be available 24/7 without any downtime for maintenance or upgrades.

Security: The system should be designed to protect the privacy of users' personal information and medical records.

Reliability: The system should be reliable, and it should not fail in the middle of an important operation.

Performance: The system should be able to respond to user requests quickly and efficiently.

Usability: The system should be user-friendly and easy to use.

Compatibility: The system should be compatible with different web browsers and devices, including desktop computers, laptops, tablets, and smartphones.

Maintainability: The system should be easy to maintain and upgrade. The system should have a modular architecture that allows developers to add new features and fix bugs without affecting the existing code.

4.4 SYSTEM FEASIBILITY STUDY

A feasibility study is conducted once the problem is clearly understood. Feasibility study is a high-level capsule version of the entire system analysis and design process.

Technical feasibility

The technical feasibility of the system refers to its ability to be developed and implemented using PHP and MySQL. The development team should have expertise in these technologies and be able to design and develop a scalable and secure system.

Operational feasibility

The operational feasibility of the system refers to its ability to meet the needs of the users. The system should be designed to streamline the organ donation process and improve communication among the users and their medical teams. The system should also be user-friendly and accessible from different devices.

Economic feasibility

The economic feasibility of the system refers to its ability to generate revenue and be financially viable. The system can be funded by the government, non-profit organizations, or through donations from the users and the public. The costs of developing, implementing, and maintaining the system should be evaluated against the potential benefits, including increased efficiency and effectiveness of the donation process.

Legal feasibility

The legal feasibility of the system refers to its ability to comply with the legal and regulatory requirements related to donation and transplantation. The system should comply with the laws related to patient privacy, data protection, and donation and transplantation. The legal and regulatory requirements should be thoroughly researched and incorporated into the system design.

4.5 PROPOSED SYSTEM

This is a Web application which have a user-friendly interface that allows users to easily navigate the system. It uses an advanced code to match the donors with the recipients based on medical compatibility, geographic location, and other factors. The software can handle a large volume of requests and generate accurate matches and provide an enhanced communication system that allows donors, recipients, and team to communicate securely and efficiently. The system provides real-time updates to the donors, recipients. It's integrated with the electronic health records (EHR) of the organ donors and recipients. It allows user to rate the system's performance and provide feedback on how to improve the system.

4.6 MODULES AND THEIR DESCRIPTION

1. **Admin** 2. **Donor / Recipient**

1. Admin

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

Blood Group: In this section, admin can manage blood group (Add/ Delete).

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 website pages.

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

Request Received by Donor: In this section, admin can view the request that is received by the donor.

Admin can also update his profile, change the password and recover the password.

2. Donor/Recipient

Home**:** It is a welcome page for users and donors. If any users want to donate their blood, they must register with us.

About Us: Users can view the about us page.

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

Donor List: Users can view and contact with donor.

Hospital List: User can view suggested hospitals.

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

4.7 HARDWARE/SOFTWARE REQUIERMENT – TECHNOLOGY

Software Requirement

Front End	HTML, CSS, JAVASCRIPT
Back End	PHP Programming
Project Type	Web Application
Other Tools	MS office 2007, Browser

[Table 1: Software Requirements]

Hardware Requirement

Supported OS	Windows 10
RAM	4 GB is Enough.
Display	Color Monitor
Hardware	Keyboard, Mouse, Internet Connectivity

[Table 2: Hardware Requirements]

Tools and Technology used

Project Name	**Organ Sharing: Finder and Donation**
Project Version	PHP 5.8
IDE Tools	Sublime, Visual Studio
Database	MYSQL

[Table 3: Tools & Technology]

4.8 APPROACH AND JUSTIFICATION

Approach

Organ sharing and donation are important aspects of healthcare systems around the world, and there are many different approaches to managing these processes. A Finder and Donation System using PHP and MySQL could help facilitate the process of organ donation and make it more efficient.

Organ sharing is a system where donors register to donate their organs after their death, and their organs are matched with patients who need them. This matching process could be based on several factors, including the compatibility of the donor and recipient, the urgency of the recipient's need, and the distance between the donor and recipient.

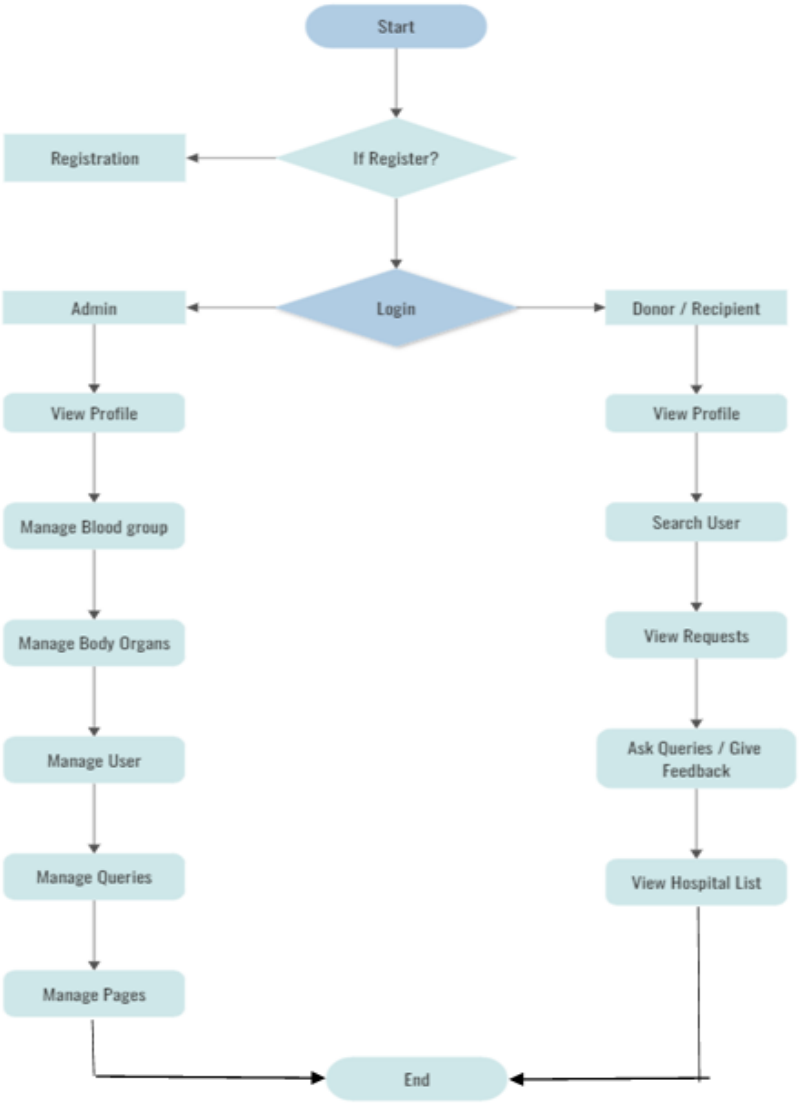
Justification

An organ donation is a selfless act that can have a profound impact on the lives of others. By donating their organs, individuals can help to save lives and improve the health and well-being of people in need. Additionally, organ donation is supported by many religious and ethical principles, which recognize the importance of helping others and promoting the common good. A Finder and Donation System using PHP and MySQL could help to facilitate the process of organ sharing and make it more efficient, while also promoting the benefits of organ donation and encouraging more people to participate in this important process.

CHAPTER 5: SYSTEM DESIGN

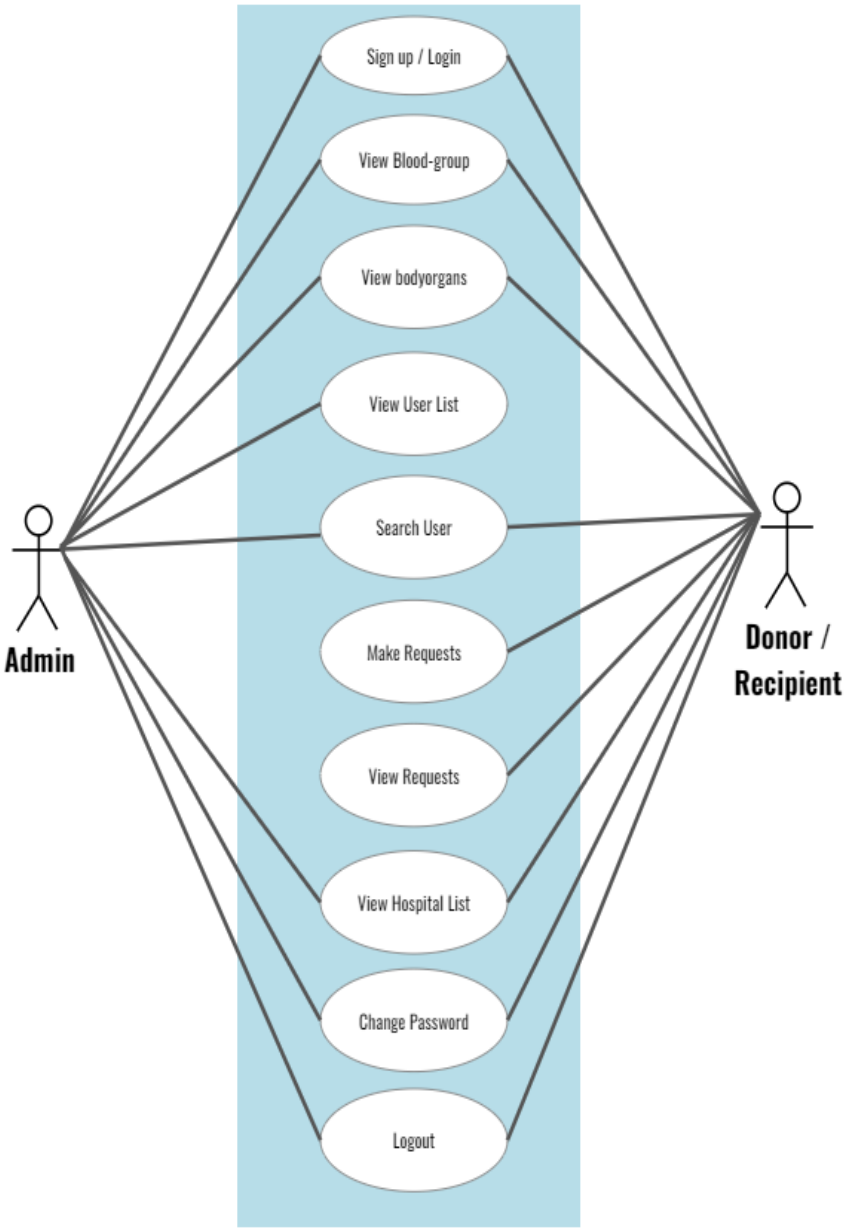
5.1 SYSTEM DESIGN

5.1.1 SYSTEM FLOWCHART



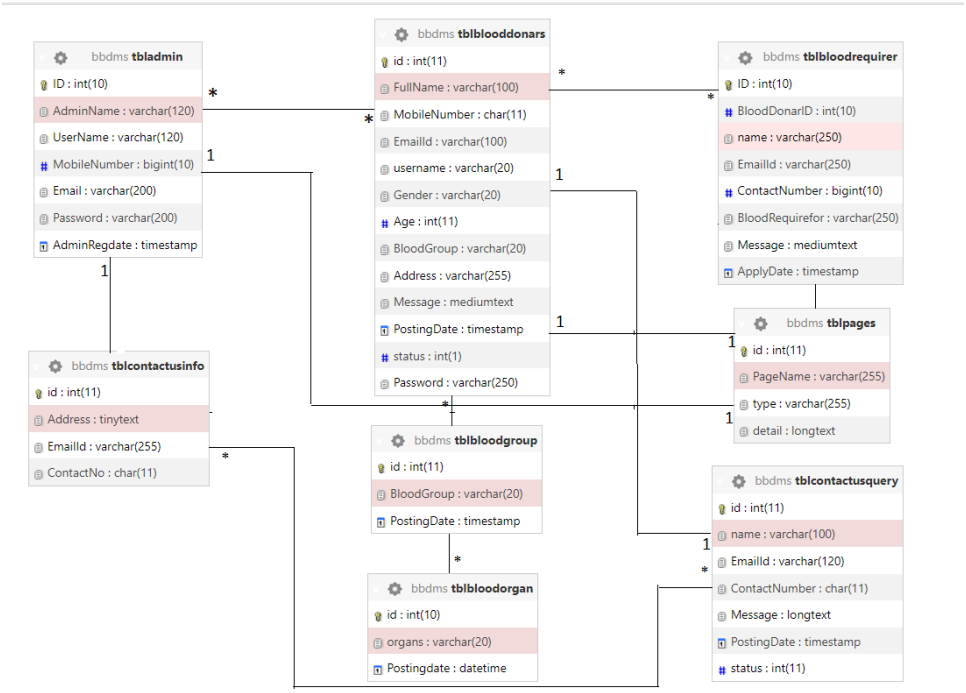
[Fig 5.1.1: System Flowchart]

5.1.2 USECASE DIAGRAM



[Fig 5.1.2: Use case Diagram]

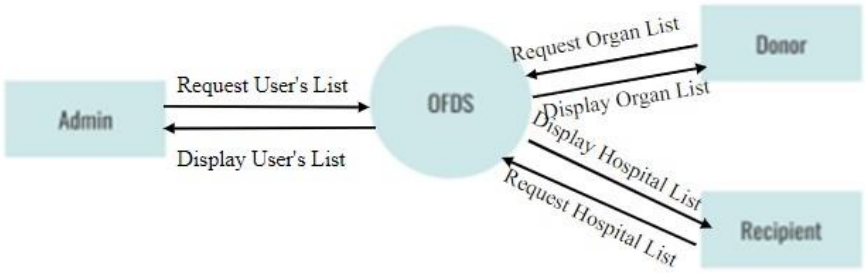
5.1.3 CLASS DIAGRAM



[Fig 5.1.3: Class Diagram]

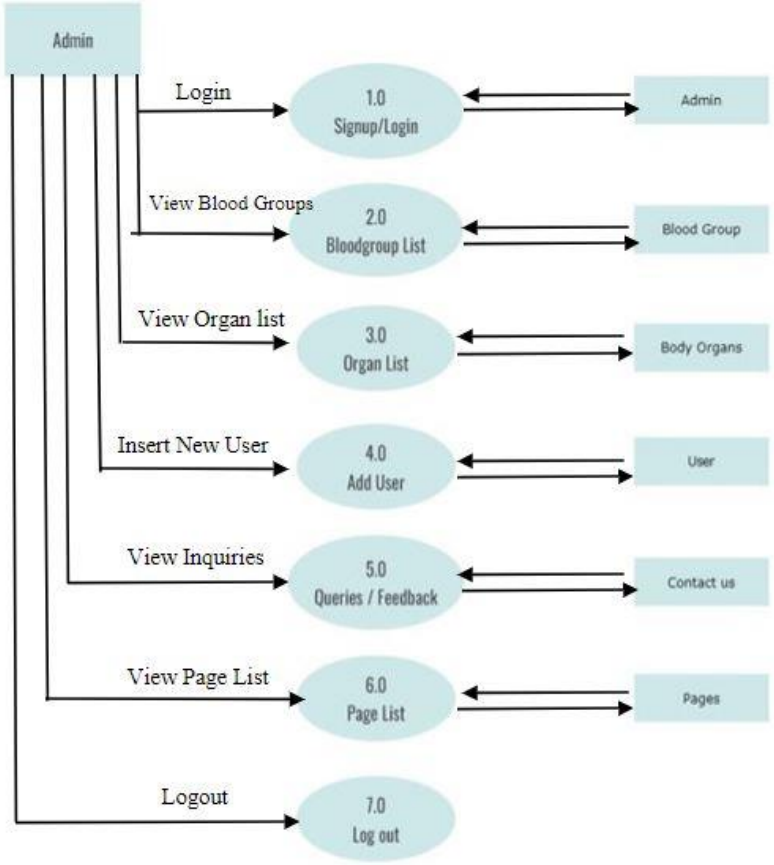
5.1.4 DATA FLOW DIAGRAM

1) Level 0: Context



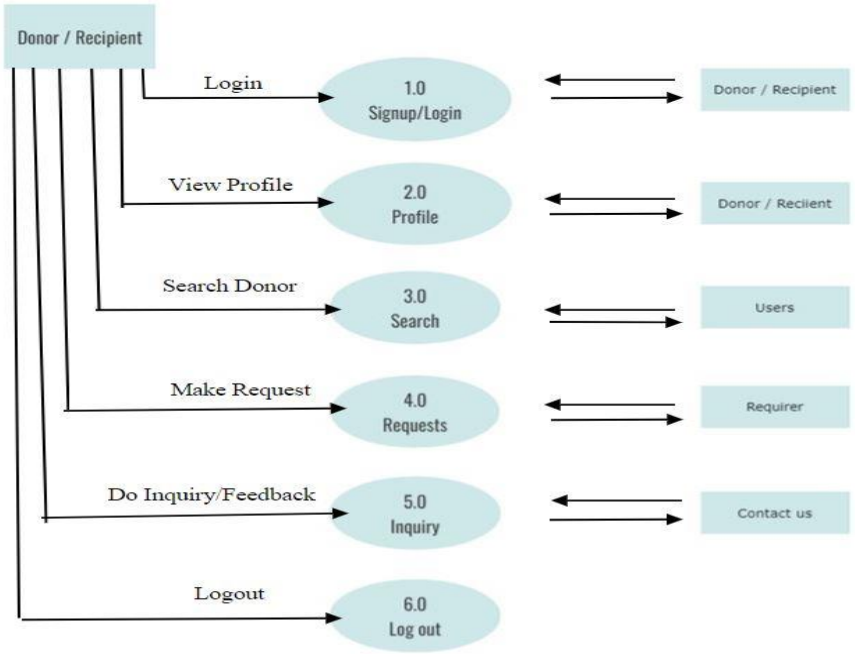
[Fig 1: DFD LEVEL 0]

2) Level 1: Admin



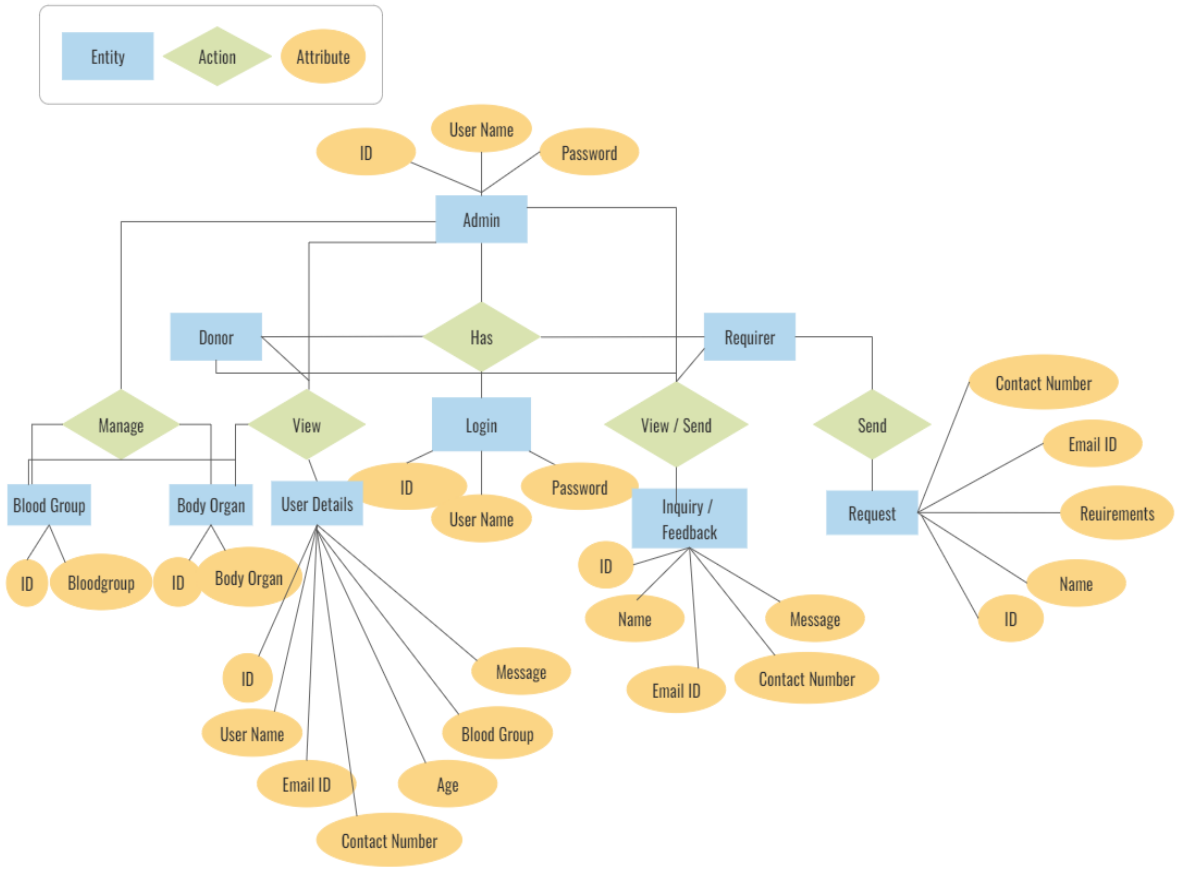
[Fig 2: DFD LEVEL 1 - Admin]

3) Level 1: Donor / Recipient



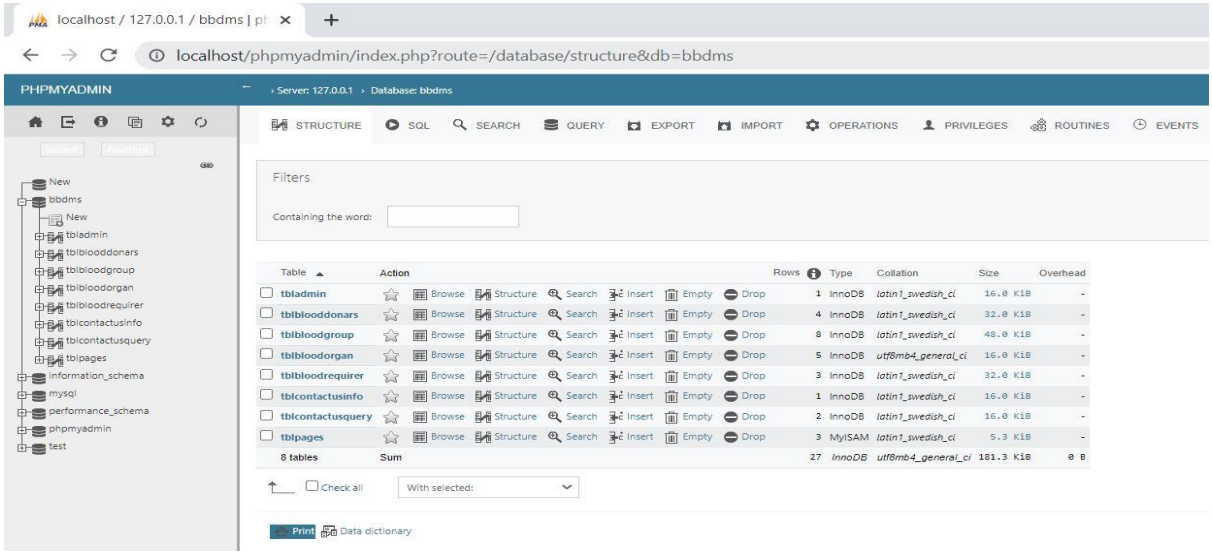
[Fig 3: DFD LEVEL 1 – Donor/Recipient]

5.1.5 ER DIAGRAM



[Fig 5.1.5: Entity Relationship Diagram]

5.2 DATABASE DESIGN



[Fig 5.2: Database Design]

DATA DICTIONARY

1. **tbladmin**

Column	Type	Null	Default	Comments
ID *(Primary)*	int (10)	No		Admin's ID
AdminName	varchar (120)	No		Admin Name
UserName	varchar (120)	No		Admin's Username
MobileNumber	bigint(10)	No		Admin's Mobile Number
Email	varchar (200)	No		Admin's Email Id
Password	varchar (200)	No		Admin's Password
AdminRegdate	Datetime	Yes	current_timestamp()	Present Date and Time

2. **tblblooddonars**

Column	Type	Null	Default	Comments
id *(Primary)*	int (11)	No		Identification Code
FullName	varchar (100)	Yes	*NULL*	User's Fullname
username	varchar(20)	Yes	*NULL*	Unique Username
MobileNumber	char(11)	Yes	*NULL*	User's Contact No.
EmailId	varchar(100)	Yes	*NULL*	User's Emailid
Gender	varchar(20)	Yes	*NULL*	User's Gender
Age	int(11)	Yes	*NULL*	User's Age
BloodGroup	varchar(20)	Yes	*NULL*	User's Bloodgroup
Address	varchar(255)	Yes	*NULL*	Home Address
Message	mediumtext	Yes	*NULL*	User's Message
Password	varchar(250)	Yes	*NULL*	User's Password
PostingDate	timestamp	No	current_timestamp()	Present Date and Time
status	int(1)	Yes	*NULL*	

3. **tblbloodgroup**

Column	Type	Null	Default	Comments
id *(Primary)*	int(11)	No		Blood Group Id
BloodGroup	varchar(20)	Yes	*NULL*	Blodd Group Type
PostingDate	timestamp	No	current_timestamp()	Present Date and Time

4. **tblbloodorgan**

Column	Type	Null	Default	Comments
id *(Primary)*	int(10)	No		Organ's Id
organs	varchar(20)	No		Organ Name
Postingdate	Datetime	No		Preset Date

5. **tblbloodrequirer**

Column	Type	Null	Default	Comments
ID *(Primary)*	int(10)	No		Requires Id
BloodDonarID	int(10)	Yes	*NULL*	Donor's Id
name	varchar(250)	Yes	*NULL*	Requires Name
EmailId	varchar(250)	Yes	*NULL*	Requirer's Email id
ContactNumber	bigint(10)	Yes	*NULL*	Requirer's Contact No
BloodRequirefor	varchar(250)	Yes	*NULL*	Reason
Message	mediumtext	Yes	*NULL*	Inquiry Message
ApplyDate	timestamp	Yes	current_timestamp()	Date and Time

6. **tblcontactusinfo**

Column	Type	Null	Default	Comments
id *(Primary)*	int(11)	No		
Address	tinytext	Yes	*NULL*	Inquiry Spot
EmailId	varchar(255)	Yes	*NULL*	Email id for Inquiry
ContactNo	char(11)	Yes	*NULL*	COntact No

7. **tblcontactusquery**

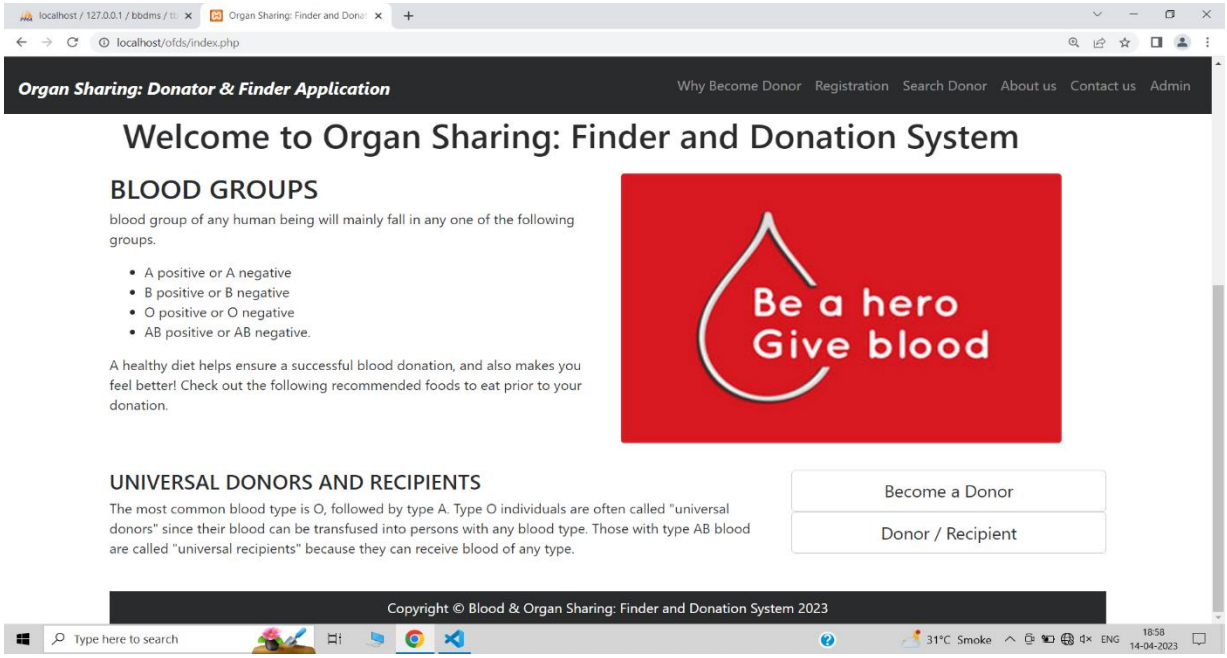
Column	Type	Null	Default	Comments
id *(Primary)*	int(11)	No		Identification Code
Name	varchar(100)	Yes	*NULL*	Inquirer's Name
EmailId	varchar(120)	Yes	*NULL*	Inquirer's Email Id
ContactNumber	char(11)	Yes	*NULL*	Inquirer's Contact No
Message	longtext	Yes	*NULL*	Inquiry Reason
PostingDate	timestamp	No	current_timestamp()	Date and Time
Status	int(11)	Yes	*NULL*	

8. **tblpages**

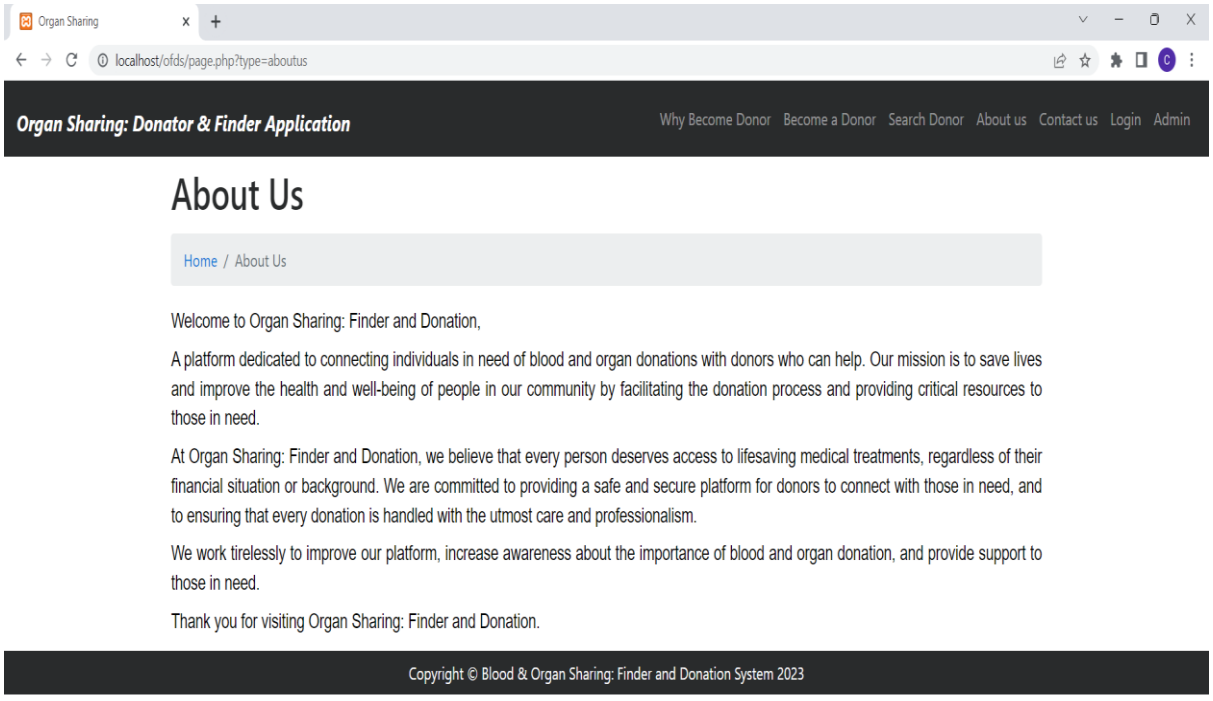
Column	Type	Null	Default	Comments
id *(Primary)*	int(11)	No		Identification Code
PageName	varchar(255)	Yes	*NULL*	Page Name
type	varchar(255)	No		Selected Page Name

5.3 INTERFACE DESIGN

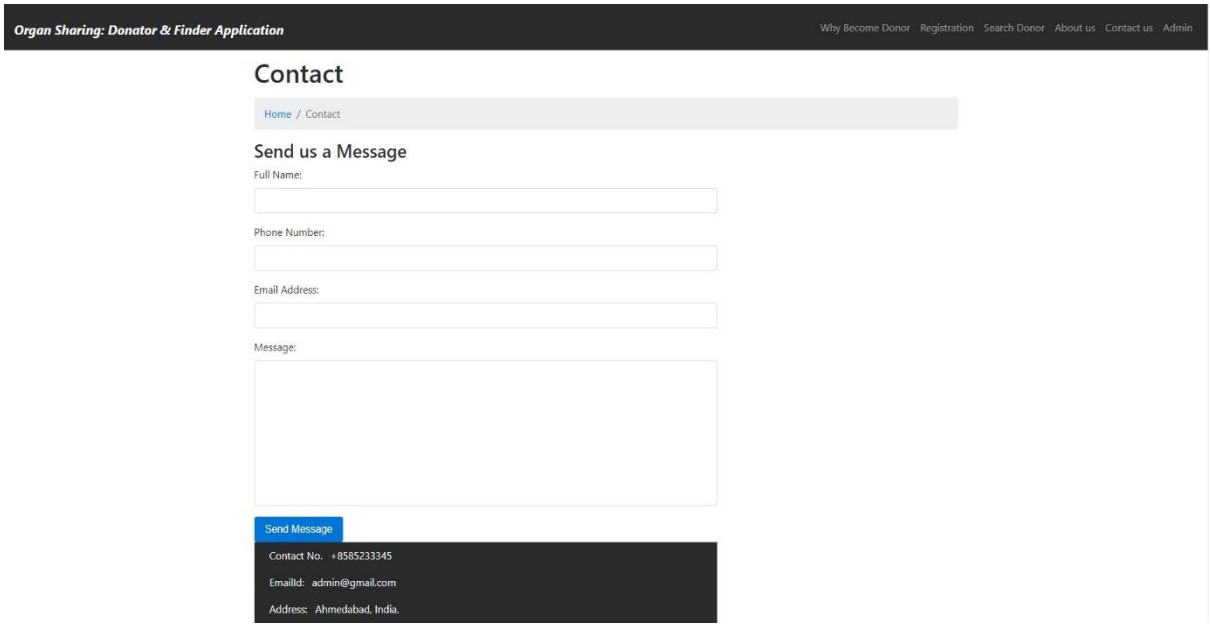
1. Home Page



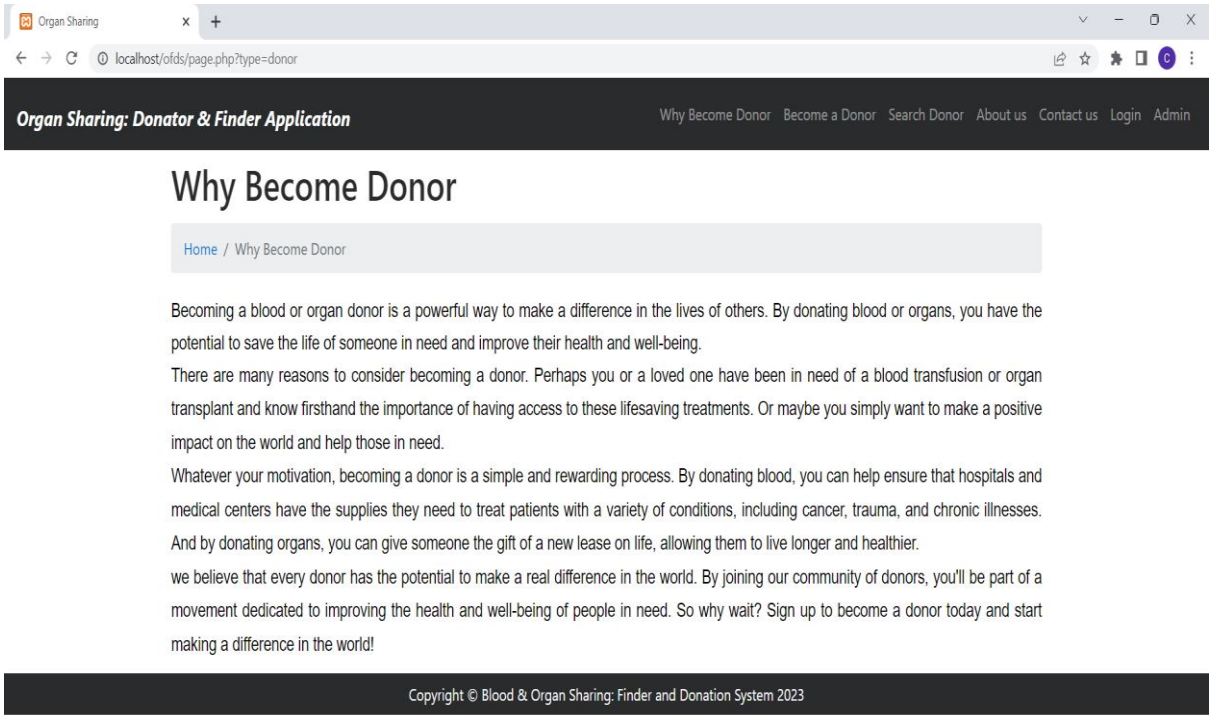
2. About us



3. Contact us



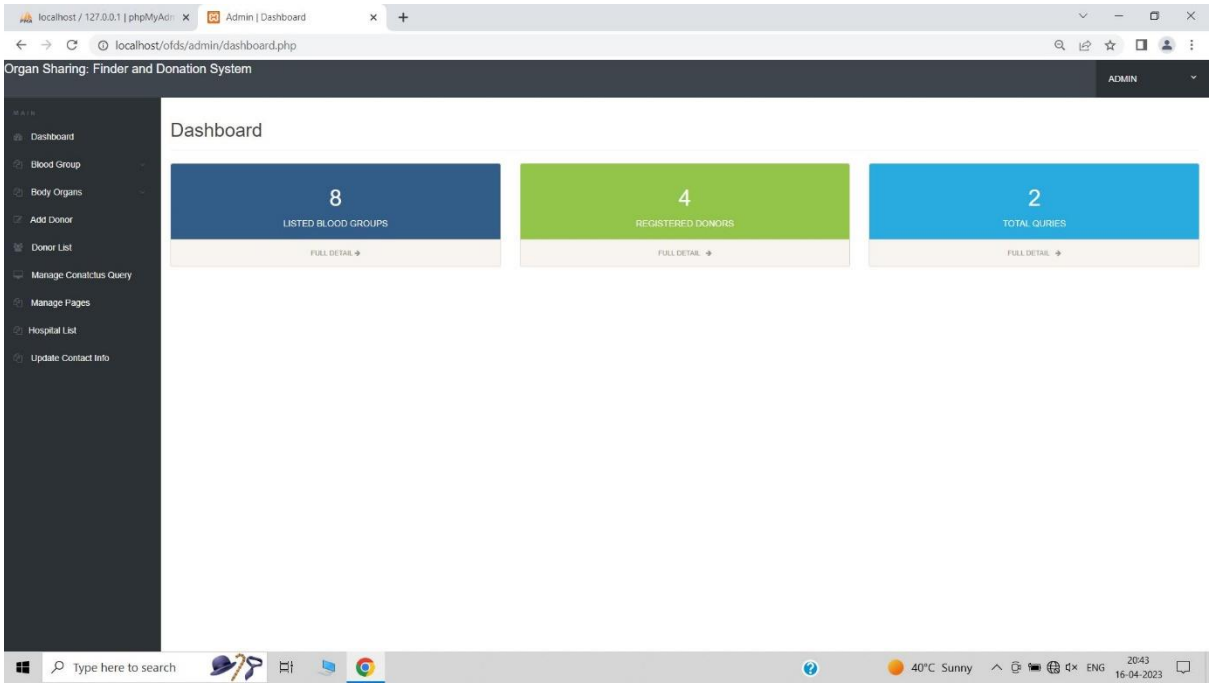
4. Why Become Donor – Awareness Message



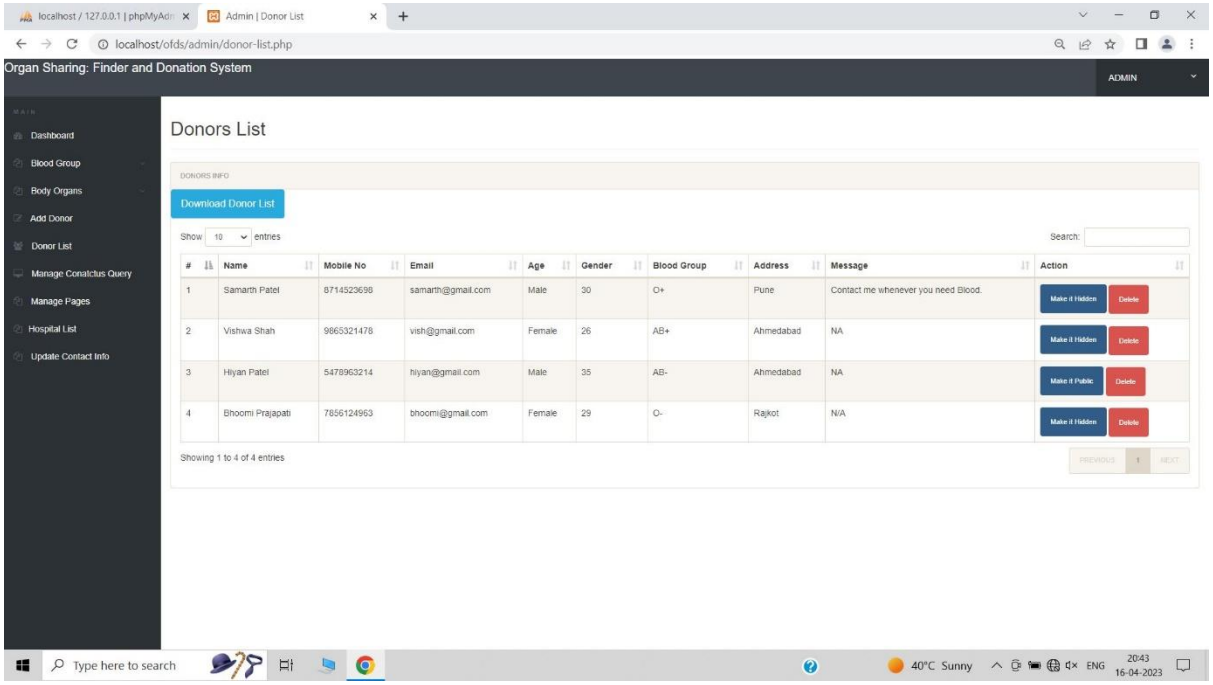
5. Login Page

6. Admin

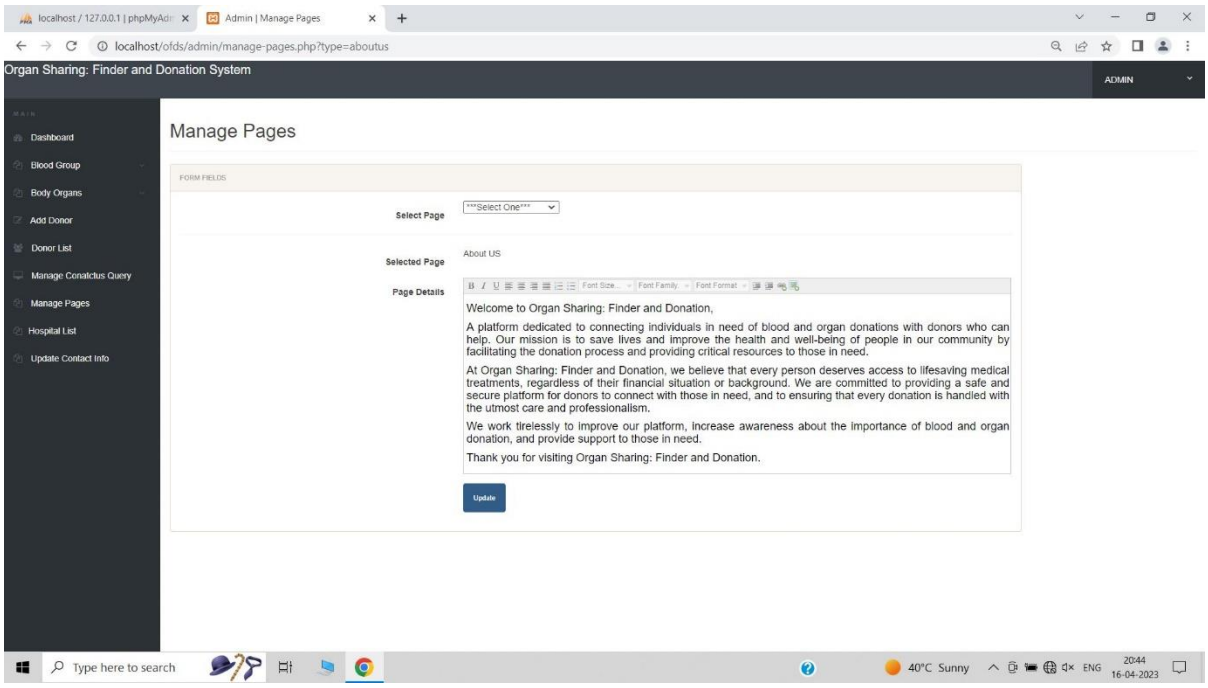
6.1 Dashboard



6.2 User's list

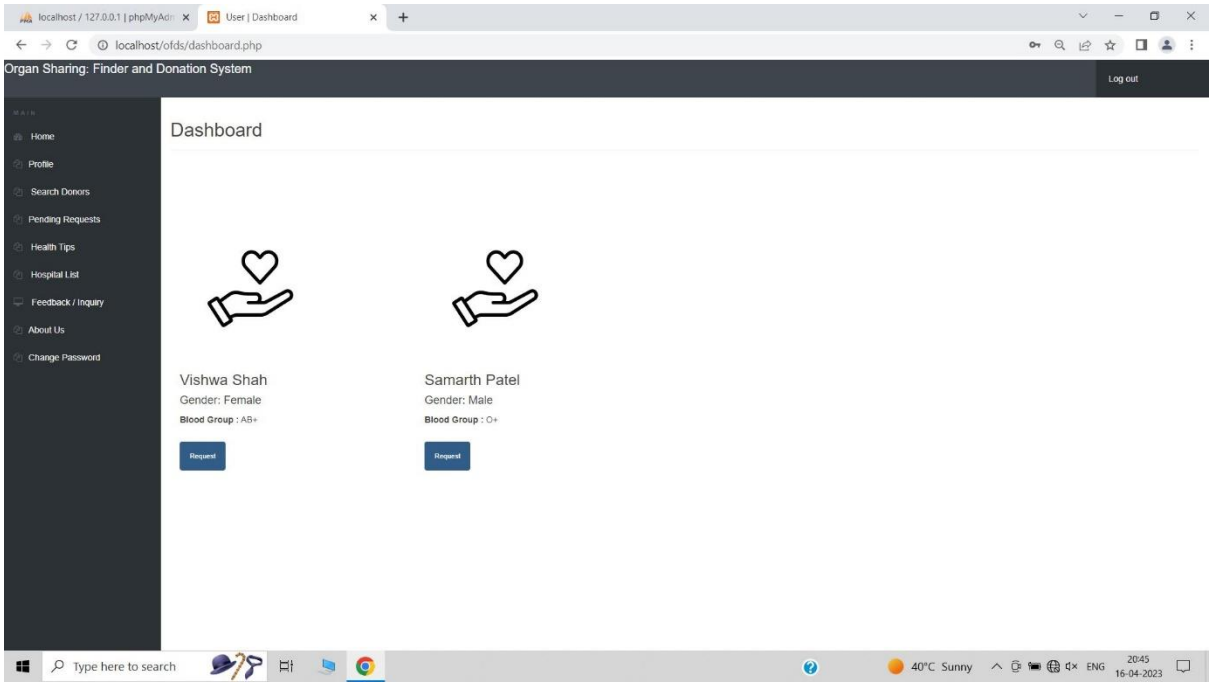


6.3 Manage Page

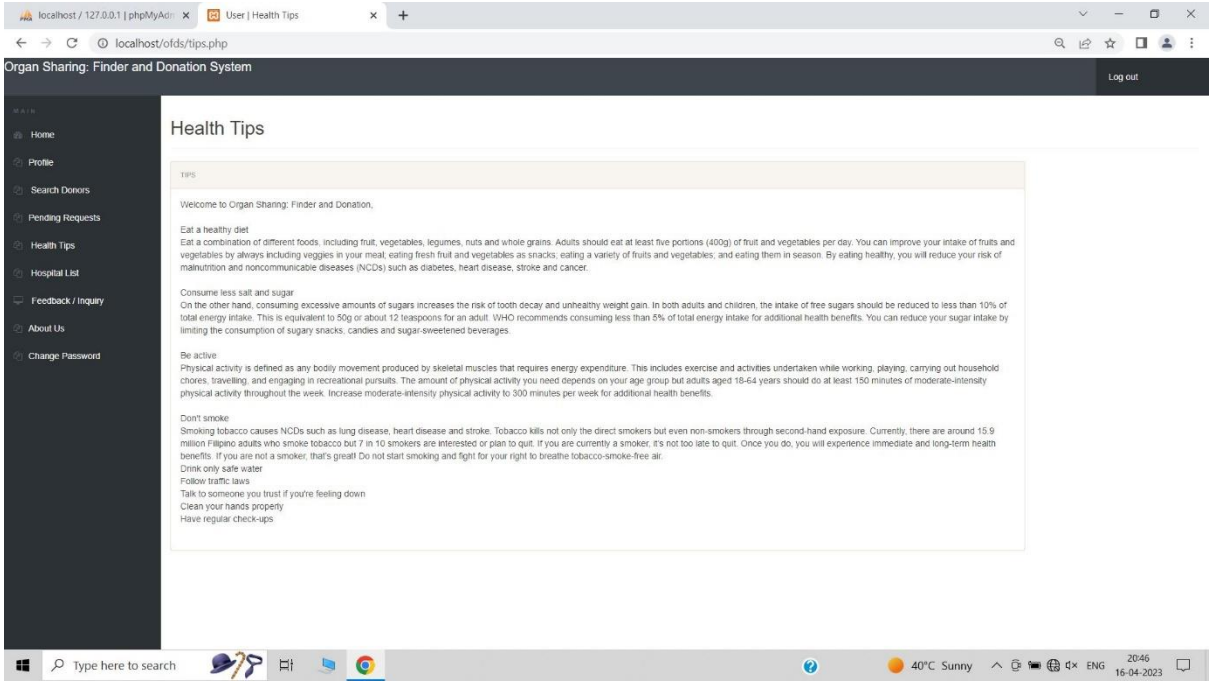


7. User

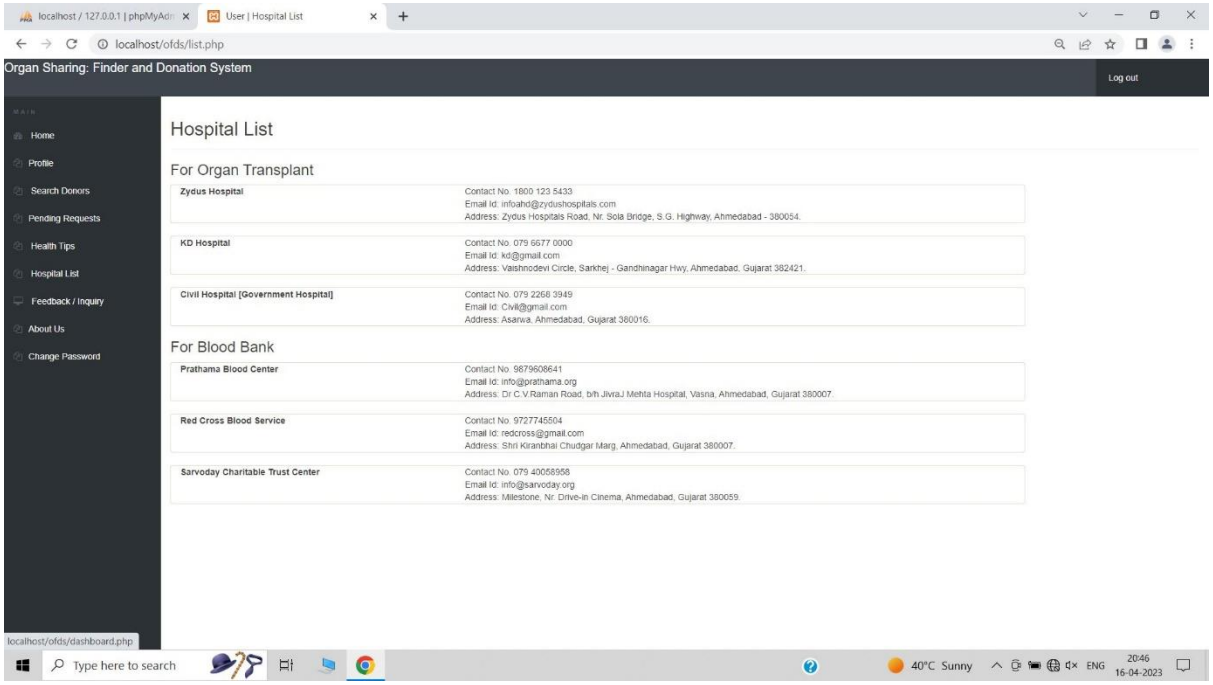
7.1 Dashboard



7.2 Health Tips



7.3 Hospital list



CHAPTER 6: IMPLEMENTATION

6.1 IMPLEMENTATION ENVIRONMENT

To implement an organ sharing finder and donation system PHP MySQL project, you would need to have a web server environment set up on your computer or on a remote server. This environment should include PHP and MySQL installed.

Here are the steps you can follow to set up the environment:

- Choose a web server: You can choose Apache or Nginx as your web server.
- Install PHP: You can install PHP using your package manager or download it from the official PHP website.
- Install MySQL: You can install MySQL using your package manager or download it from the official MySQL website.
- Create a database: Use the MySQL command-line interface or a tool like PHPMyAdmin to create a database for your project.
- Set up the project files: You can use a text editor or an integrated development environment (IDE) to create the PHP files for your project.
- Connect to the database: Use PHP's MySQLi or PDO extension to connect to the database and perform database operations.
- Implement the organ sharing finder and donation system functionality: You will need to create PHP scripts that allow users to search for organ donors, register as a donor, and request an organ donation.
- Test the project: You can test the project on your local machine or on a remote server.
- Deploy the project: You can deploy the project on a web server so that it can be accessed by users over the internet.

6.2 OUTCOMES

The outcomes of an organ sharing: finder and donation system PHP MySQL project can be significant, both for the individuals involved and the broader community.

- Increased Availability of Organs: The system can increase the availability of organs for transplantation by connecting donors with recipients who need organs.

- Faster and More Efficient Matching: The system can use algorithms to match donors with recipients based on criteria such as blood type, age, and medical history. This can result in faster and more efficient matching, reducing waiting times for patients in need of organs.
- Improved Communication: The communication platform provided by the system can facilitate communication between donors and recipients, allowing them to discuss any queries or details related to the organ donation process.
- Better Management of Organ Donation: The system can help manage organ donation more effectively by providing a centralized platform for organ donation management.
- Greater Awareness and Education: The system can help raise awareness about organ donation and the importance of organ donation by providing educational resources and information on the topic.
- An organ sharing: finder and donation system PHP MySQL project can have a positive impact on individuals in need of organs, their families, and the broader community.

6.3 RESUL ANALYSIS

To analyze the results of an organ sharing: finder and donation system PHP MySQL project, it is important to evaluate its performance against specific objectives and metrics.

- Number of successful transplants: The number of successful transplants facilitated by the system can be used as a measure of its effectiveness in connecting donors with recipients.
- Reduction in waiting times: The waiting times for organ transplants can be measured before and after the implementation of the system. A reduction in waiting times can indicate that the system is helping to improve the efficiency of the organ donation process.
- User satisfaction: User satisfaction can be measured through surveys or other feedback mechanisms. This can provide insight into how the system is being received by donors, recipients, and other stakeholders.
- Accuracy and reliability of the system: The accuracy and reliability of the system can be measured by tracking the number of errors, system downtime, and other technical issues that may affect the performance of the system.

- Compliance with ethical and legal requirements: The compliance of the system with ethical and legal requirements can be assessed through audits and reviews by relevant regulatory bodies.
- Number of registered donors and recipients: The number of registered donors and recipients using the system can be used as a measure of its adoption and usage.

A successful organ sharing finder and donation system PHP MySQL project should demonstrate improved outcomes in terms of successful transplants, reduced waiting times, high user satisfaction, high accuracy and reliability, compliance with ethical and legal requirements, and increased adoption and usage. By measuring these metrics and analyzing the results, the project team can identify areas for improvement and make necessary adjustments to ensure the success of the project.

CHATER 7: TESTING

7.1 TESTING PLAN / STRATEGY

A testing plan for an organ sharing: finder and donation system PHP MySQL project should include a comprehensive set of tests that ensure the system is functioning as intended, and that it meets the requirements and objectives of the project. Here are some key elements of a testing plan:

Functional Testing: This type of testing checks whether the system is functioning as per the specified requirements. It includes testing of all the features and functionalities of the system, including registration, search, communication, and payment processing.

Performance Testing: This type of testing evaluates the performance of the system under different conditions such as user load, network bandwidth, and response time. It includes load testing, stress testing, and scalability testing.

Security Testing: This type of testing evaluates the system's ability to protect against unauthorized access, data breaches, and other security threats. It includes testing of authentication and authorization, encryption, and data protection.

Compatibility Testing: This type of testing checks whether the system is compatible with different operating systems, web browsers, and devices. It includes testing the system on different platforms and devices to ensure it works as intended.

Usability Testing: This type of testing evaluates the system's ease of use and user interface design. It includes testing of the user interface, navigation, and user experience to ensure that the system is user-friendly.

Regression Testing: This type of testing checks whether the changes made to the system do not have any negative impact on its existing functionalities. It includes testing of all the functionalities of the system after a change or update is made.

User Acceptance Testing: This type of testing involves real-world users who test the system and provide feedback on their experience. It includes testing of the system in a simulated environment to ensure that it meets user expectations.

The testing plan should be executed at different stages of the project to ensure that the system is tested thoroughly before it is launched.

7.2 Test Result and Analysis

The test results and analysis of an organ sharing: finder and donation system PHP MySQL project provide valuable insights into the performance and functionality of the system.

Test Cases: The test results should include the list of test cases executed, including the test case name, description, and status (pass/fail). This helps to track the progress of the testing and identify areas that need improvement.

Test Coverage: The test results should also include the percentage of test coverage achieved, indicating how much of the system has been tested. A high percentage of test coverage indicates that the system has been tested thoroughly.

Defects: The test results should include a list of defects identified during testing, including their severity and priority. This helps to prioritize the defects and fix them in the order of importance.

Test Environment: The test results should also include details about the test environment, including the operating system, web server, database server, and browser used for testing. This helps to identify any compatibility issues.

Performance Metrics: The test results should include performance metrics such as response time, throughput, and resource utilization. This helps to identify performance bottlenecks and improve the system's performance.

User Feedback: The test results should include feedback from users who participated in user acceptance testing. This provides valuable insights into the user experience and helps to

improve the system's usability and user-friendliness.

The analysis should include an assessment of the system's strengths and weaknesses and recommendations for improvement. By analyzing the test results, the project team can identify areas that need improvement and make necessary adjustments to ensure the success of the project.

7.2.1 TEST CASES

Test case of organ sharing finder and donation system php mysql project:

1. **Registration Test Case:**

Test case name: User registration

Description: Verify that a user can register an account on the system

Steps:

Go to the registration page

Fill in the registration form with valid information

Submit the registration form

Expected result: The user should be registered and redirected to the login page

2. **Search Test Case:**

Test case name: Organ search

Description: Verify that a user can search for organs on the system

Steps:

Go to the search page

Enter the organ type and location

Click the search button

Expected result: The system should display a list of available organs that match the search criteria

3. **Communication Test Case:**

Test case name: Contact organ donor

Description: Verify that a user can contact the organ donor

Steps:

Find an available organ from the search results

Click the "contact donor" button

Fill in the contact form with valid information

Submit the contact form

Expected result: The system should send a message to the organ donor with the user's contact information

4. Security Test Case:

Test case name: Unauthorized access

Description: Verify that the system prevents unauthorized access to user information

Steps:

Log out of the system

Try to access a user's profile or other protected information

Expected result: The system should redirect the user to the login page and display an error message.

CHAPTER 8: CONCLUSION AND DISCUSS

8.1 CONCLUSION

A blood and organ donation PHP MySQL project is a vital system that can help save lives by facilitating the donation process. The system will provide a platform for donors and recipients to connect and donate blood or organs. The project will be divided into various modules, each with specific functionalities such as user management, search and matching, donation management, notification, reporting, admin panel, security, and feedback and reviews.

The feasibility study of the project has shown that it is technically, economically, operationally, legally, and socially feasible. The project will be financially viable, with a clear ROI and will be user-friendly, easy to navigate, and require minimal training. The system will be designed to comply with all relevant laws and regulations and promote awareness and education about blood and organ donation.

In conclusion, a blood and organ donation PHP MySQL project has the potential to make a significant impact on society and improve patient outcomes. It is crucial to conduct a thorough feasibility study and develop a well-designed system with robust features to ensure its success.

8.2 DATES OF CONTINUES EVALUTION

Internal Review 1: 27/02/2023

Internal Review 2: 08/04/2023

Internal Review 3: 29/04/2023

8.3 LIMITATION AND FUTURE ENHANCEMENT

Like any other system, A blood and organ donation PHP MySQL project has some limitations and areas that can be improved upon in the future.
Here are some of the limitations and potential enhancements:

Limitations:

Donor and recipient participation: The success of the system depends on the number of donors and recipients registered on the platform. However, some potential donors or recipients may not be aware of the system or may be hesitant to register due to privacy or security concerns.

Eligibility Criteria: The system relies on accurate information provided by users during registration. Inaccurate or incomplete information can lead to mismatches or disqualify some donors or recipients, limiting the system's effectiveness.

Data Privacy and Security: The system will require users to provide personal information, such as their name, contact information, and medical history. Ensuring the security and privacy of this information is crucial to the success of the system.

Future Enhancements:

Integration with Healthcare Providers: Integrating the system with healthcare providers' systems will enable a more efficient exchange of information between donors, recipients, and healthcare providers. It will also ensure that the donation process is well-coordinated and streamlined.

Mobile Application: Developing a mobile application for the system can make it more accessible and convenient for users to register, search, and donate blood or organs.

Social Media Integration: Integrating the system with social media platforms can help increase awareness and promote the donation process. Users can share their donation experiences, encourage others to donate, and raise awareness about the importance of blood and organ donation.

Incentives for Donors: Offering incentives, such as discounts or loyalty points, can encourage more people to donate and increase the system's success rate.

A blood and organ donation PHP MySQL project has the potential to make a significant impact on society, but it also has some limitations that need to be addressed. Future enhancements can help overcome these limitations and improve the system's effectiveness and reach.

REFERENCES

Website Referred:

There are many websites and resources available online that can provide information and guidance on developing a blood and organ donation PHP MySQL project.

Here are some websites that can be referred to:

Website References

W3Schools - https://www.w3schools.com/

W3Schools provides a comprehensive tutorial on web development, including PHP and MySQL. This website can be useful for developers who are just starting with web development.

PHP.net - https://www.php.net/

PHP.net is the official website for the PHP programming language. It provides a comprehensive documentation and reference guide for PHP developers.

MySQL Documentation - https://dev.mysql.com/doc/

The MySQL documentation provides a comprehensive guide to the MySQL database, including installation, configuration, and usage.

Book References

"Learning PHP, MySQL & JavaScript: With jQuery, CSS & HTML5" by Robin Nixon

"MySQL Workbench: Data Modeling & Development" by Michael McLaughlin

"PHP PROGRAMMING" by Dr. Shyam Chawda