

BLOOD DONATION AND DONOR MANAGEMNET SYSTEM

A MINI-PROJECT REPORT Submitted by

KAMALESH V 240701224

JOEL ALEX RAJ D 240701218

in partial fulfillment of the award of the degree

of

BACHELOR OF ENGINEERING

IN

COMPUTER SCIENCE AND ENGINEERING



RAJALAKSHMI ENGINEERING COLLEGE, CHENNAI

An Autonomous Institute

CHENNAI

NOVEMBER 2025

BONAFIDE CERTIFICATE

Certified that this project "BLOOD DONATION AND DONOR MANAGEMENT SYSTEM" is the bonafide work of "KAMALESH V, JOEL ALEX RAJ D" who carried out the project work under my supervision.

SIGNATURE

Dr. S. SATHIYAVATHI

PROFESSOR

Dept. of Computer Science and Engineering,
Rajalakshmi Engineering College

Chennai

This mini project report is submitted for the viva voce examination to be held on

INTERNAL EXAMINER EXTERNAL EXAMINER

ABSTRACT

The Blood Donation and Donor Management System is a desktop-based application developed using Java Swing and MySQL to streamline the operations of a blood bank. The system enables easy management of donor records, blood inventory, and hospital blood requests by providing an integrated and user-friendly admin dashboard. Donors can be registered, and their donation history is tracked with automatic updates to the inventory. Hospitals can request blood units, and the admin can approve or reject requests based on availability. The system reduces manual errors, improves data accuracy, and enhances the efficiency of tracking blood stock levels. Overall, this project aims to provide a reliable and efficient digital solution for

managing blood donation activities and ensuring timely availability of blood units for medical need.

ACKNOWLEDGEMENT

We express our sincere thanks to our beloved and honorable chairman MR. S. MEGANATHAN and the chairperson DR. M.THANGAM MEGANATHAN for their timely support and encouragement.

We are greatly indebted to our respected and honorable principal

Dr. S.N. MURUGESAN for his able support and guidance.

No words of gratitude will suffice for the unquestioning support extended to us by our Head Of The Department Dr. E.M. MALATHY and our Deputy Head Of The Department Dr. J. MANORANJINI for being ever supporting force during our project work

We also extend our sincere and hearty thanks to our internal guide Dr. S.SATHIYAVATHI, for her valuable guidance and motivation during the completion of this project.

Our sincere thanks to our family members, friends and other staff members of computer science engineering.

1. KAMALESH V

2. JOEL ALEX RAJ D

TABLE OF CONTENTS		
CHAPTER NO.	TITLE	PAGE NO.
ABSTRACT		IV
1	INTRODUCTION	1
1.1	INTRODUCTION	2
1.2	SCOPE OF THE WORK	3
1.3	PROBLEM STATEMENT	4
1.4	AIM AND OBJECTIVES OF THE PROJECT	5

CHAPTER NO.	TITLE	PAGE NO.
2	SYSTEM SPECIFICATIONS	6
2.1	HARDWARE SPECIFICATIONS	6
2.2	SOFTWARE SPECIFICATIONS	7
3	MODULE DESCRIPTION	8
4	CODING	10
5	SCREENSHOTS	25
6	CONCLUSION AND FUTURE ENHANCEMENT	35
—	REFERENCES	36

LIST of figures

FIGURE NO.	TITLE	PAGE NO.
5.1	ADMIN LOGIN PAGE	15
5.2	ADMIN DASHBOARD	15
5.3	DONOR REGISTRATION FORM	16
5.4	BLOOD REQUEST FORM	16

5.5	DONORS TAB	17
5.6	INVENTORY TAB	17

CHAPTER 1

INTRODUCTION

1.1INTRODUCTION

This chapter provides an overview of the Blood Donation and Donor Management System, its purpose, and the need for digital blood management in hospitals and blood banks. It introduces the concept of donor tracking and inventory monitoring using an automated software solution. Blood donation plays a vital role in saving lives. Managing donors, blood stock, and hospital requests manually often leads to errors and inefficiency. To address these issues, this project implements a computerized system using Java Swing and MySQL to handle all core operations of a blood bank in a structured, user-friendly manner.

1.2 SCOPE OF THE WORK

The project covers donor registration, recording donations, maintaining blood inventory, managing blood requests from hospitals, and providing admin-based dashboard features. It ensures proper monitoring of blood units across different blood groups and supports decision-making with real-time data updates.

1.3 PROBLEM STATEMENT

Traditional blood bank management relies heavily on manual record-keeping, which leads to data loss, delays, and inaccurate inventory tracking. There is no efficient way to check available blood units, verify donor histories, or handle multiple requests systematically. A digital, automated solution is required to overcome these limitations.

1.4 AIM AND OBJECTIVES OF THE PROJECT

The aim of this project is to develop a computerized Blood Donation and Donor Management System using Java Swing and MySQL that efficiently manages donor information, automates blood inventory updates, processes hospital blood requests, reduces manual errors, and provides a user-friendly dashboard for streamlined blood bank operations.

CHAPTER 2

SYSTEM SPECIFICATIONS

2.1 HARDWARE SPECIFICATIONS

Processor	:	Intel Core i3 or above
Memory Size	:	8GB (Minimum)
HDD	:	1 TB (Minimum)

2.2 SOFTWARE SPECIFICATIONS

- ❑ Operating System: Windows 10/11
- ❑ Programming Language: Java (JDK 17)
- ❑ IDE: IntelliJ IDEA
- ❑ Database: MySQL Server
- ❑ Connector: MySQL Connector/J
- ❑ Other Libraries: Swing, JDBC, ReportLab (for PDF generation)

CHAPTER 3

MODULE DESCRIPTION

This chapter describes the main modules of the system:

1. **Admin Login Module:** Allows secure login into the system.
2. **Donor Management Module:** Handles donor registration, viewing donors, updating donor information, and recording donations.
3. **Blood Inventory Module:** Automatically updates inventory based on donations and approved requests.
4. **Blood Request Module:** Allows hospitals/organizations to request blood units; admin can approve or reject requests.
5. **Dashboard Module:** Provides an overview of system operations and allows easy navigation between features.

CHAPTER 4

SAMPLE CODING

```
package com.bloodbank;  
  
import javax.swing.*.*;  
  
import  
  
javax.swing.table.DefaultTabl  
  
eModel;  
  
import java.awt.*.*;
```

```
import java.sql.Connection;

import

java.sql.PreparedStatement;

import java.sql.ResultSet;

import java.sql.Statement;

import java.util.List;

public class

AdminDashboardForm

extends JFrame {

    private JTable

requestTable;

    private DefaultTableModel

requestTableModel;

    private JButton

btnApprove, btnReject,

btnAddRequest,

btnRefreshRequests;
```

```
        private JTable donorTable;

        private DefaultTableModel
donorTableModel;

        private JButton

btnAddDonor, btnEditDonor,

btnDeleteDonor,

btnRecordDonation,

btnRefreshDonors;

        private JTextArea

inventoryArea;

        private JButton

btnRefreshInventory,

btnAddUnits;

public AdminDashboardForm() {

    setTitle("Admin Dashboard");

    setSize(950, 600);

    setDefaultCloseOperation(JFrame.
```

```
EXIT_ON_CLOSE);

    setLocationRelativeTo(null);

    JTabbedPane tabbedPane = new

JTabbedPane();

    // Requests Tab

    JPanel requestsPanel = new

JPanel(new BorderLayout());

    requestTableModel = new

DefaultTableModel(new

Object[]{"Request ID", "Hospital",

"Blood Type", "Units", "Status"},

0);

    requestTable = new

JTable(requestTableModel);

    requestsPanel.add(new

JScrollPane(requestTable),
```

```
BorderLayout.CENTER);

    JPanel requestBtnPanel = new

JPanel();

    btnApprove = new

JButton("Approve");

    btnReject = new

JButton("Reject");

    btnAddRequest = new

JButton("Submit Request");

    btnRefreshRequests = new

JButton("Refresh");

requestBtnPanel.add(btnApprove);

requestBtnPanel.add(btnReject);


requestBtnPanel.add(btnAddRequ

est);

requestBtnPanel.add(btnRefreshRe
```

```

quests);

requestsPanel.add(requestBtnPane

l, BorderLayout.SOUTH);

tabbedPane.add("Blood

Requests", requestsPanel);

JPanel donorsPanel = new JPanel(new BorderLayout());
donorTableModel = new DefaultTableModel(new Object[]{"Donor
ID","Name","Blood Type","Age","Gender","Contact","Last Donation","Units
Donated"},0);
donorTable = new JTable(donorTableModel);
donorsPanel.add(new JScrollPane(donorTable), BorderLayout.CENTER);
JPanel donorBtnPanel = new JPanel();
btnAddDonor = new JButton("Add Donor");
btnEditDonor = new JButton("Edit Donor");
btnDeleteDonor = new JButton("Delete Donor");
btnRecordDonation = new JButton("Record Donation");
btnRefreshDonors = new JButton("Refresh");
donorBtnPanel.add(btnAddDonor); donorBtnPanel.add(btnEditDonor);
donorBtnPanel.add(btnDeleteDonor); donorBtnPanel.add(btnRecordDonation);
donorBtnPanel.add(btnRefreshDonors);
donorsPanel.add(donorBtnPanel, BorderLayout.SOUTH);
tabbedPane.add("Donors", donorsPanel);

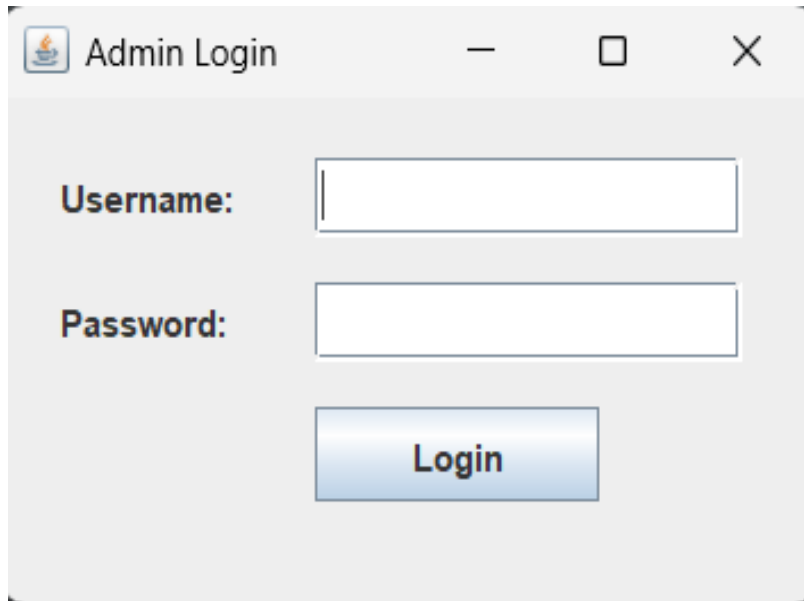
```

Sample 2

Sample 2 depicts the booking part of the code, where it displays booking details and enter user data and store it in database

CHAPTER 5

SCREEN SHOTS



A screenshot of a web application window titled "Admin Login". The window has a standard title bar with a minimize button, a maximize button, and a close button. The main content area is light gray and contains two text input fields. The first field is labeled "Username:" and the second is labeled "Password:". Below these fields is a blue button with the text "Login".

Admin Login

Username:

Password:

Login

FIG 5.1 ADMIN LOGIN PAGE

Admin Dashboard

Blood Requests

Donors

Inventory

Request ID	Hospital	Blood Type	Units	Status
1	apollo	b+	2	Rejected
2	apollo	B+	2	Pending
3	Srinivasa	A+	2	Approved
4	City Hospital	A+	2	Pending
5	Green Cross	O-	1	Pending
6	Sunrise Clinic	B+	3	Approved
7	Mercy Hospital	AB-	1	Rejected

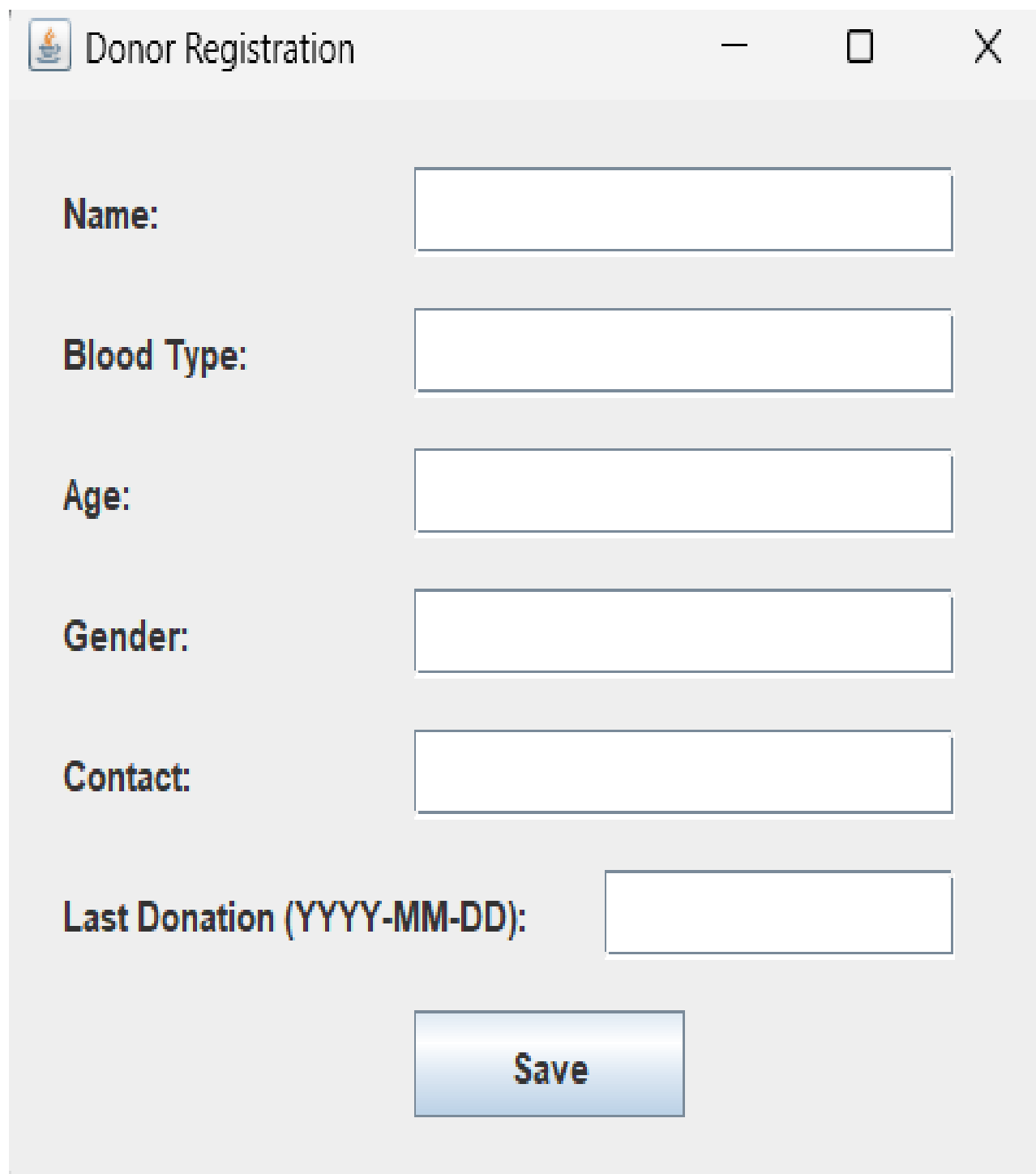
Approve

Reject

Submit Request

Refresh

FIG 5.2 ADMIN DASHBOARD



The image shows a software window titled "Donor Registration". It contains a form with the following fields: "Name:", "Blood Type:", "Age:", "Gender:", "Contact:", and "Last Donation (YYYY-MM-DD):". Each field has a corresponding text input box. At the bottom center of the form is a blue "Save" button. The window has a standard title bar with a minimize button, a maximize button, and a close button.

Name:	<input type="text"/>
Blood Type:	<input type="text"/>
Age:	<input type="text"/>
Gender:	<input type="text"/>
Contact:	<input type="text"/>
Last Donation (YYYY-MM-DD):	<input type="text"/>

Save

FIG 5.3 DONOR REGISTRATION FORM

Blood Request Form

Hospital Name:

Blood Type:

Units Needed:

Request

FIG 5.4 BLOOD REQUEST FORM

FIG 5.5 DONOR TAB

Blood Requests	Donors	Inventory					
Donor ID	Name	Blood Type	Age	Gender	Contact	Last Donation	Units Donated
3	Kamalesh	A+	19	Male	9874561230	2024-02-14	3
4	John Doe	A+	30	Male	9876543210	2025-09-01	2
5	Jane Smith	O-	25	Female	9876543211	2025-08-15	1
6	Robert Brown	B+	40	Male	9876543212	2025-09-10	3
7	Emily Davis	AB-	35	Female	9876543213	2025-09-05	2
8	adam	o	22	male	4512368574	2002-02-02	2

Blood Requests	Donors	Inventory
A- : 2		
A+ : 5		
AB- : 1		
AB+ : 2		
B- : 1		
B+ : 4		
O- : 3		
O+ : 6		

Refresh Inventory

Add Units

FIG 5.6 INVENTORY TAB

6. CONCLUSION AND FUTURE ENHANCEMENT

The Blood Donation and Donor Management System successfully digitalizes key blood bank operations, enabling reliable donor tracking and real-time inventory management.

Future enhancements may include:

- SMS/Email notification system
- Multi-user login (Admin, Staff, Hospital)
- Cloud database hosting
- Advanced analytics and reporting
- Web or mobile app extension

REFERENCES

1. [Java Documentation \(Oracle\)](#)
2. [MySQL Documentation](#)
3. [JDBC Programming Tutorials](#)
4. [Swing GUI Developer Guide](#)
5. [*Learning MySQL: A Beginner's Guide* — Seyed M.M. & Hugh E. Williams, O'Reilly Media.](#)