ABSTRACT

BLOOD DONATION MANAGEMENT SYSTEM is a web based application developed to facilitate the identification of the nearest available blood donor volunteer and the communication with him/her in the emergency situations where the blood can't be supplied through the blood banks stocks. The blood management information system also offers functionality to quick access to donor records collected from various parts of the country. This system helps to bridge the gap between the donors and recipients and to reduce the efforts required to search for donors. The purpose of this system is to simplify and automate the process of searching for blood in case of emergency and maintain the records of blood donors, recipients, and blood stocks in the bank.

TABLE OF CONTENTS

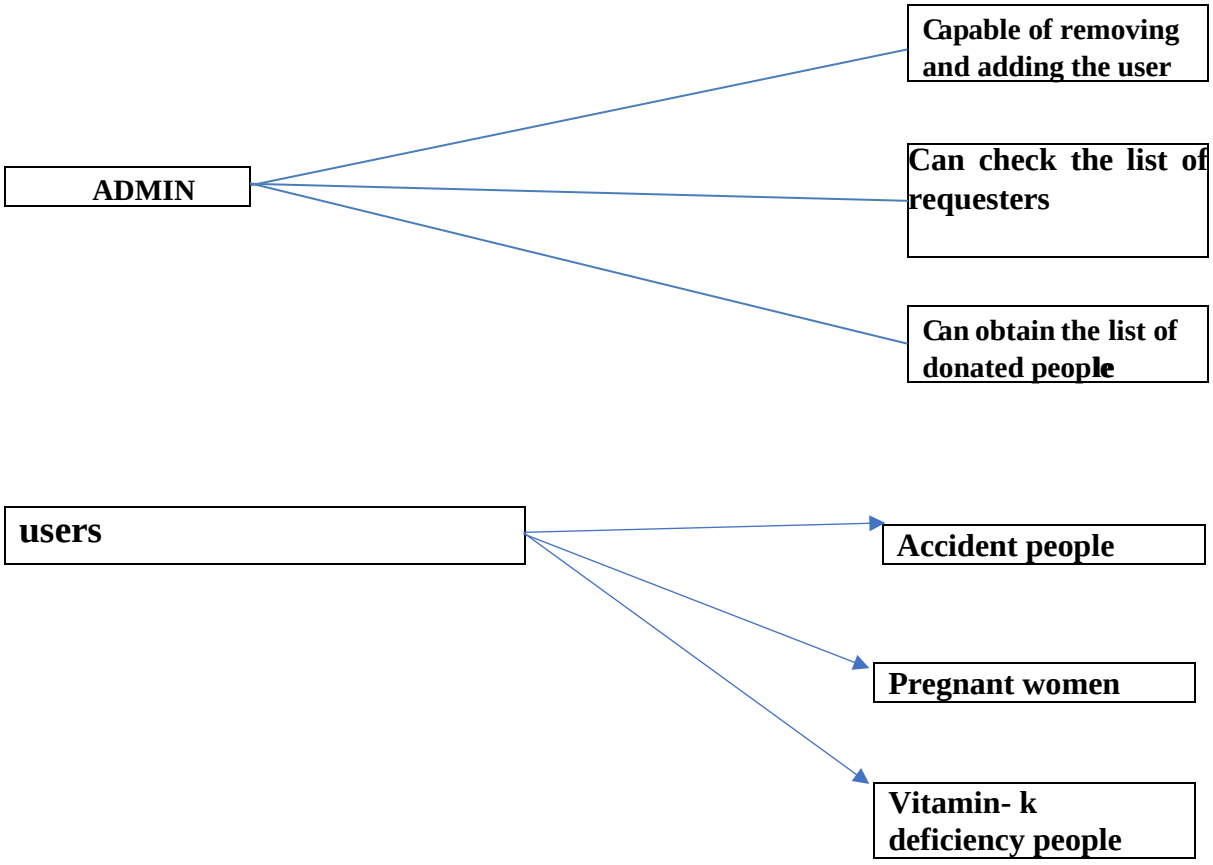
1. INTRODUCTION

The population of the world is increasing day by day, so the increase in health issues and new diseases are taking place. With the increase in the population and the increase in health issues, the need for blood is also growing. Due to the lack of communication between blood donors and the blood recipients, most of the patients in need of blood do not get blood on time and hence lose their lives. Lack of information and communication can sometimes lead to mediocre situations. The prevailing situations like covid have clearly shown the trouble many people faced in search of required blood and the struggles they faced on not finding the exact blood donor and need for blood is increasing rapidly in the day-to-day life of a human. These problems can be dealt with by automating a blood donation management system.

There is an expectation that the blood will always be there when it is really needed. Blood donor volunteers constitute the main supply source in an effective blood supply chain management. They feed blood stocks through their donation. In an emergency situation, if the stocks are insufficient, the only source of blood supply will be the people who come to the health center and donate the blood on a voluntary basis. It is certain that time is a very important component in such situation. For this reason, the health care center should call the nearest available donor in order to ensure to get the service as quickly as possible.

A web based application will be developed to facilitate the identification of the nearest available blood donor volunteer and the communication with him/her in the emergency situations where the blood can't be supplied through the blood banks stocks. In this paper this application will be presented.The blood management information system also offers functionality to quick access to donor records collected from various parts of the country. It enables monitoring of the results and performance of the blood donation activity such that relevant and measurable objectives of the organization can be checked. It provides to management timely, confidential and secure medical reports that facilitates planning and decision making and hence improved medical service delivery. The reports generated by the system give answers to most of the challenges management faces as far as blood donor records are concerned.

2.METHODOLOGY



In this project the user are capable of requesting the blood and will be waited to get the help as soon as possible.

As we talk about admin side of the project it was capable of adding, updating and deleting the donor and the probability of providing blood to the requester will be dependent on admin's active time such that they he will be checking the requests periodically which were updated by the user as of their needs

After Login can see the Number Of Donor, Number of blood requests
Can View, Update, Delete Donor.

Can see the history of blood requests

Tech stack:

- **SOFTWARE REQUIREMENTS :**

1.NetBeans idle

2.XML

3.JDBC drivers

NETBEANS:

NetBeans IDE is a free, open source, integrated development environment (IDE) that **enables you to develop desktop, mobile and web applications**. The IDE supports application development in various languages, including Java, HTML5, PHP and C++.

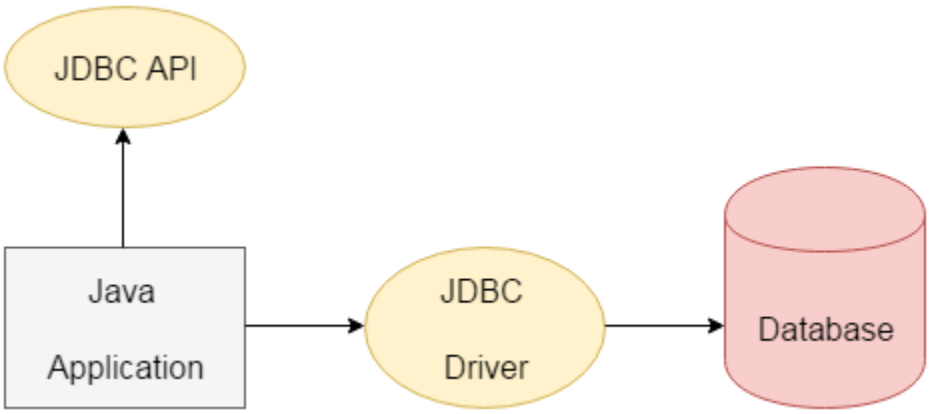
- Project management.
- Visual debugger.
- Static analysis tools.
- Code converters.
- NetBeans Profiler.
- Batch code analyzers.
- Cross-platform support.
- Multiple language support.

JDBC

JDBC stands for Java Database Connectivity. JDBC is a Java API to connect and execute the query with the database. It is a part of JavaSE (Java Standard Edition). JDBC API uses JDBC drivers to connect with the database. There are four types of JDBC drivers:

- JDBC-ODBC Bridge Driver,
- Native Driver,
- Network Protocol Driver, and
- Thin Driver

We can use JDBC API to access tabular data stored in any relational database. By the help of JDBC API, we can save, update, delete and fetch data from the database.



Through jdbc we can achieve the following

- To connect individual databases
- Implements the protocol for transferring the query and result between client and database.
- Send queries and update statements to the database
- Retrieve and process the results received from the database as answer to our question.
- To connect the databases using java programming

XML:

In JDBC applications, you use ResultSet.getXXX or ResultSet.getObject methods to retrieve data from XML columns.

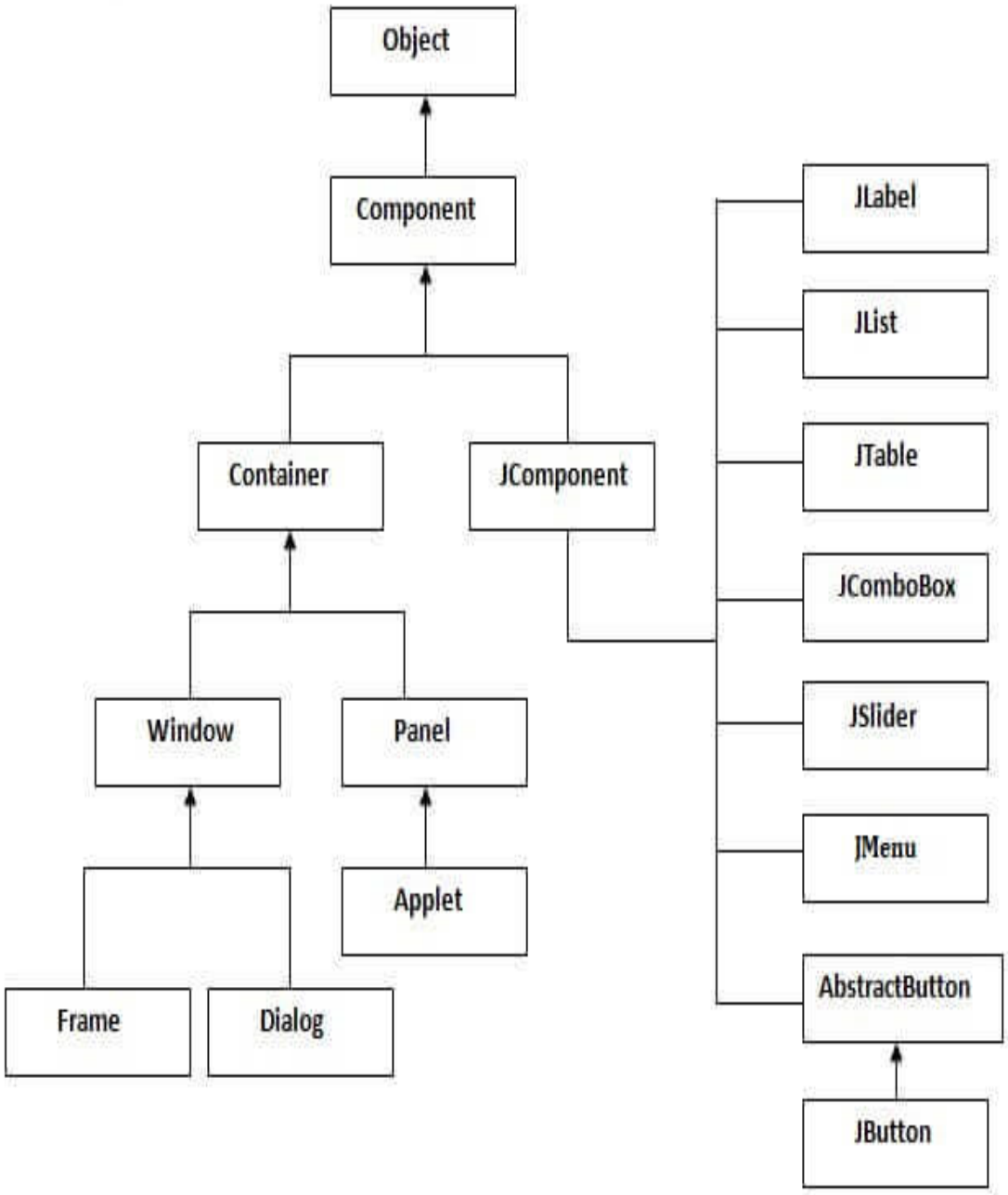
In a JDBC application, you can retrieve data from XML columns in a Db2® table as XML textual data. You can retrieve data from XML columns in a table as binary XML data (data that is in the Extensible Dynamic Binary XML DB2® Client/Server Binary XML Format), if the data server supports binary XML data.

We can use one of the following techniques to retrieve XML data:

- Use the ResultSet.getSQLXML method to retrieve the data. Then use a SQLXML.getXXX method to retrieve the data into a compatible output data type. This technique requires JDBC 4.0 or later.

For example, you can retrieve data by using the SQLXML.getBinaryStream method or the SQLXML.getSource method.

SWING



Swing API is a set of extensible GUI Components to ease the developer's life to create JAVA based Front End/GUI Applications. It is build on top of AWT API and acts as a replacement of AWT API, since it has almost every control corresponding to AWT controls. Swing component follows a Model-View-Controller architecture to fulfill the following criterias.

- A single API is to be sufficient to support multiple look and feel.
- API is to be model driven so that the highest level API is not required to have data.
- API is to use the Java Bean model so that Builder Tools and IDE can provide better services to the developers for use.

MVC Architecture

Swing API architecture follows loosely based MVC architecture in the following manner.

- Model represents component's data.
- View represents visual representation of the component's data.
- Controller takes the input from the user on the view and reflects the changes in Component's data.
- Swing component has Model as a seperate element, while the View and Controller part are clubbed in the User Interface elements. Because of which, Swing has a pluggable look-and-feel architecture.

Swing Features

- **Light Weight** − Swing components are independent of native Operating System's API as Swing API controls are rendered mostly using pure JAVA code instead of underlying operating system calls.
- **Rich Controls** − Swing provides a rich set of advanced controls like Tree, TabbedPane, slider, colorpicker, and table controls.
- **Highly Customizable** − Swing controls can be customized in a very easy way as visual apperance is independent of internal representation.
- **Pluggable look-and-feel** − SWING based GUI Application look and feel can be changed at run-time, based on available values.

3.IMPLEMENTATION

Blood bank management system will be implemented as a web application using the netbeans idle in java

- ✓ This NetBeans was extremely simple to configure on new workspace
- ✓ NetBeans being a best open Source IDE for Java.
- ✓ Can build High quality applications quickly and easily.
- ✓ Its part of back-end developers tool kit.

Major Roles in the Application:

1. **Admin**
2. **Recipient**
3. **Donor**

- This web application enter from a login form depending upon the type of credentials it check either they were users or admin moves to their required homepages and dashboards. Upon logging in all the tech stack will be based upon swing as it has enhanced featured compared to awt.
- The java swing has been used entirely as the components that were platform independent hence our application can become more scalable and accessible to the users.
- The jdbc diver's and xml were used to work with store and retrieve the data from the database starting from connecting mysql into it.

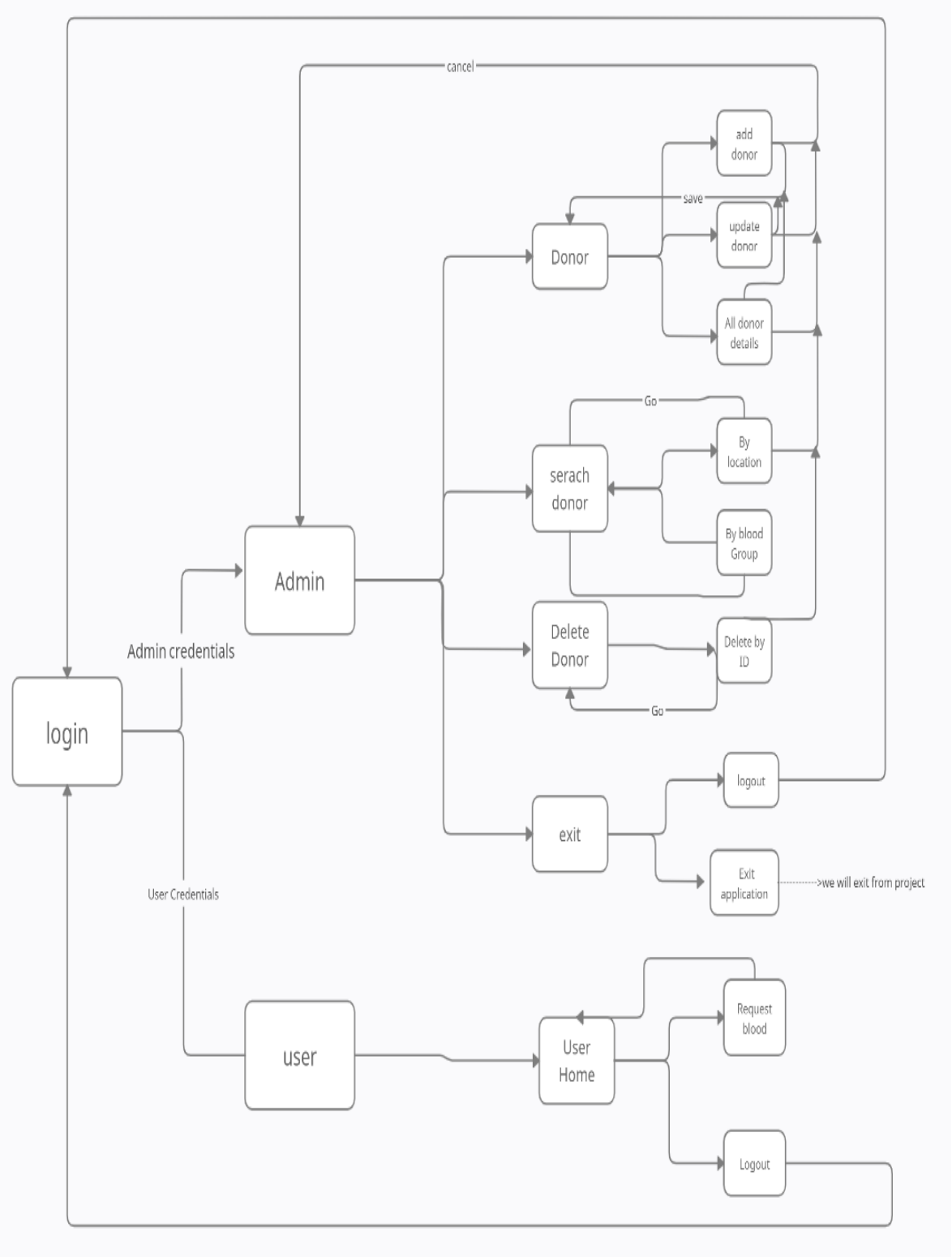
- The admin will add donor by taking necessary details from the donor and save in the database

- Like wise update and deletion operation can be done in which each operation has its own swing jframe container.

- While talking about the searching the donor we can actually filter based on the location and by blood group and a combination of both so that we will actually known how many different

blood group available in that location so that if a perfect match was not found we can give the contact of universal donor if available all this was because of the like clause in the sql language.

- After checking the required blood group the admin team will inform the requester about the blood donor with acceptance of the donor so donar donates the blood to that receiver of their location.



°Our Application can be Authorized by Admin or Users.

°If you Login with admin credentials then you will be able to :

1.View the Donar details

2.Search a donar

3.Delete a Donar

4.Exit from application

1.View the Donar Details:

Viewing the donar details refers to the list of donars details including name,gender,place,mail id,Pincode etc.., entire information related to donar will be displayed.

1.View the Donar Details:

Viewing the donar details refers to the list of donars details including name,gender,place,mail id,Pincode etc.., entire information related to donar will be displayed.

Along with this we can also update the donar details. It means that if any specific details of the donar have been changed., then we can also modify that and restore it.

2 .Search a donar

Searching a donar is done when someone requests the blood. while searching for blood distance should be considered as major criteria. So.,

we will search donar either by their location or by their blood group. When we are searching with specific Constraints if they are matched with the details present in donar details then those donar names will be displayed.

3. Delete a Donar

Deletion of donar is done when the donar is nor willing to donate blood or due to some other issues. In that case deletion of donar details is done by taking Donar ID as reference.

4.Exit from Application

We there is no task to perform then usually we exit or logout from application.

If we logout then,the page is redirected to login page ,if you choose to exit from the page, then we will be quiting the application.

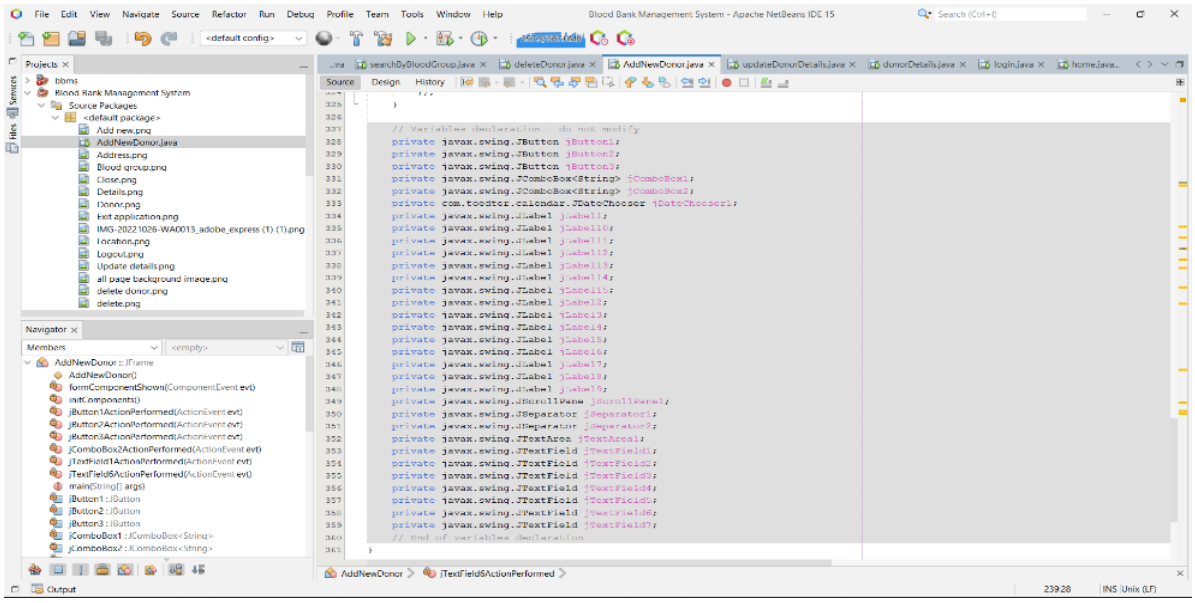
Along with Admin credentials we also have an another option for authorization, where there will be only partial view of application.

i.e.,On logging in with User credentials then we will get User Home page.,where we will be able to Request blood.

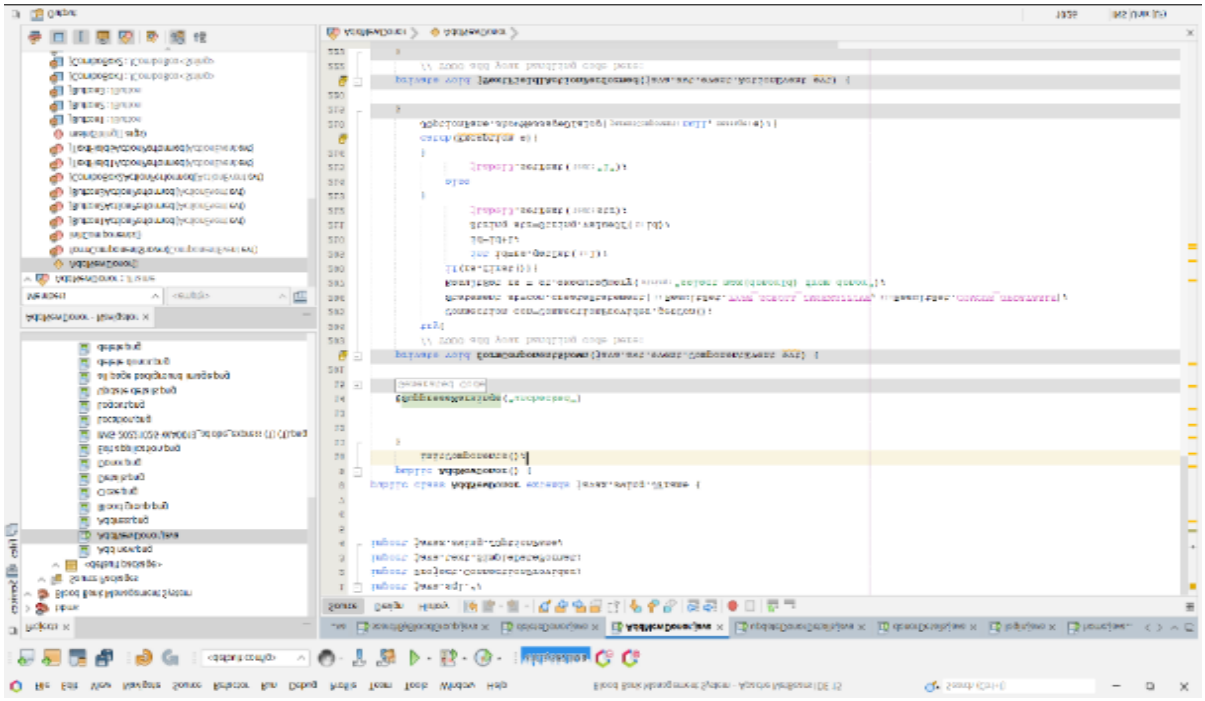
When blood is requested then we should fill the form of details asked i.e., like Which blood group is required,What is the name if the person ,their phone number,Which place they belong to, these are all mandatory.

After doing this, the request will be processed to Admin. Now from here after admin looks after that requirements.

Now after the request if the user logs out then again they will be back to Login page.



Some basic code snippets of an jframe container

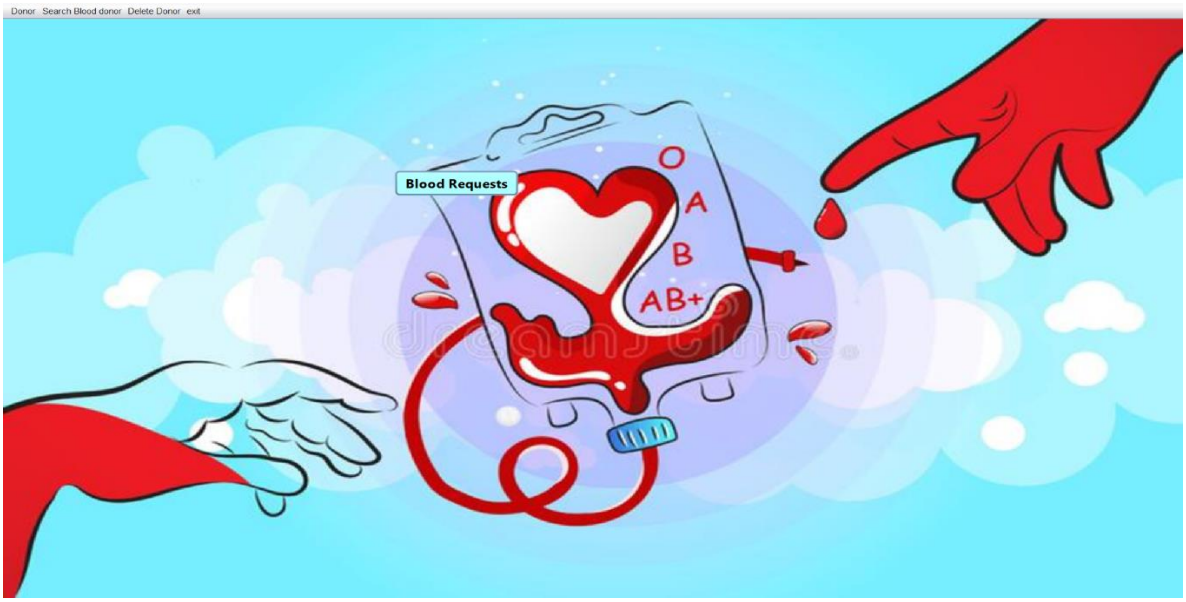


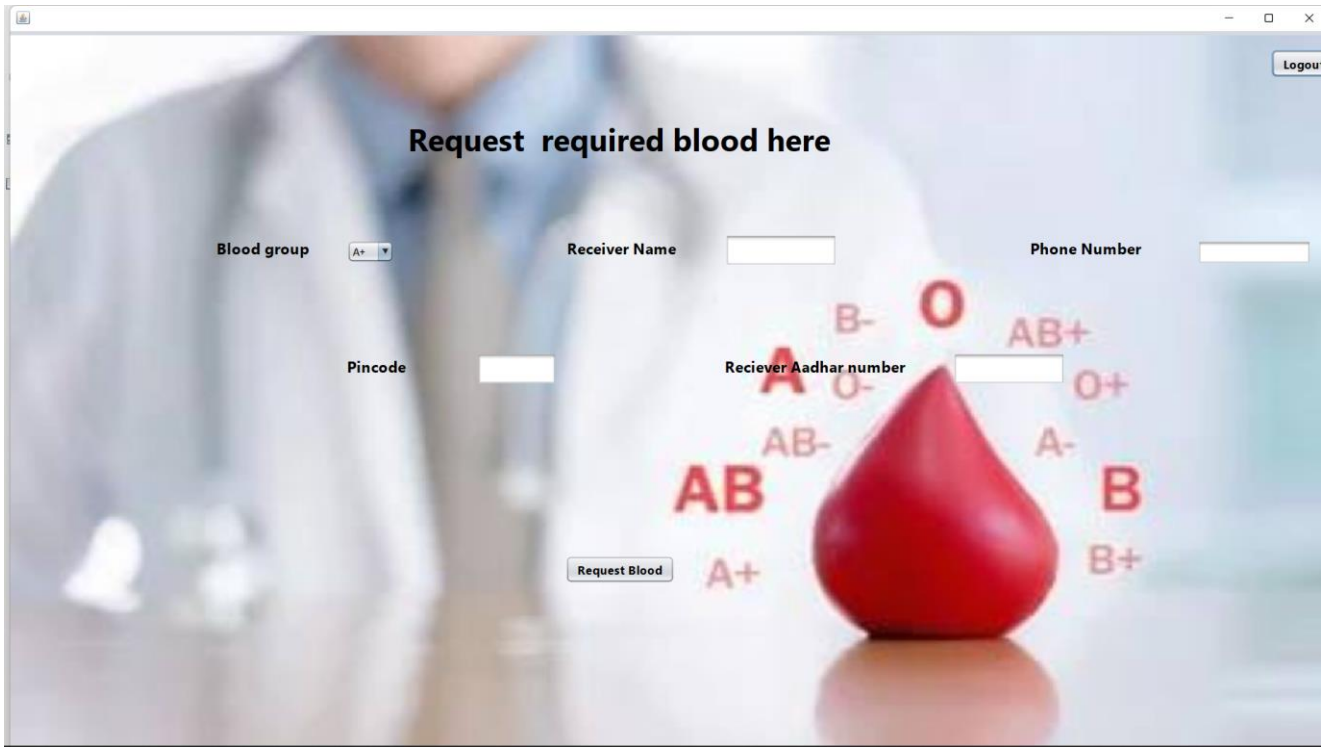
You can check the complete code for the project here-

https://drive.google.com/drive/u/0/folders/1WrFwZIm8tMy-Lkpfp8cE9fc-yBklcuWe

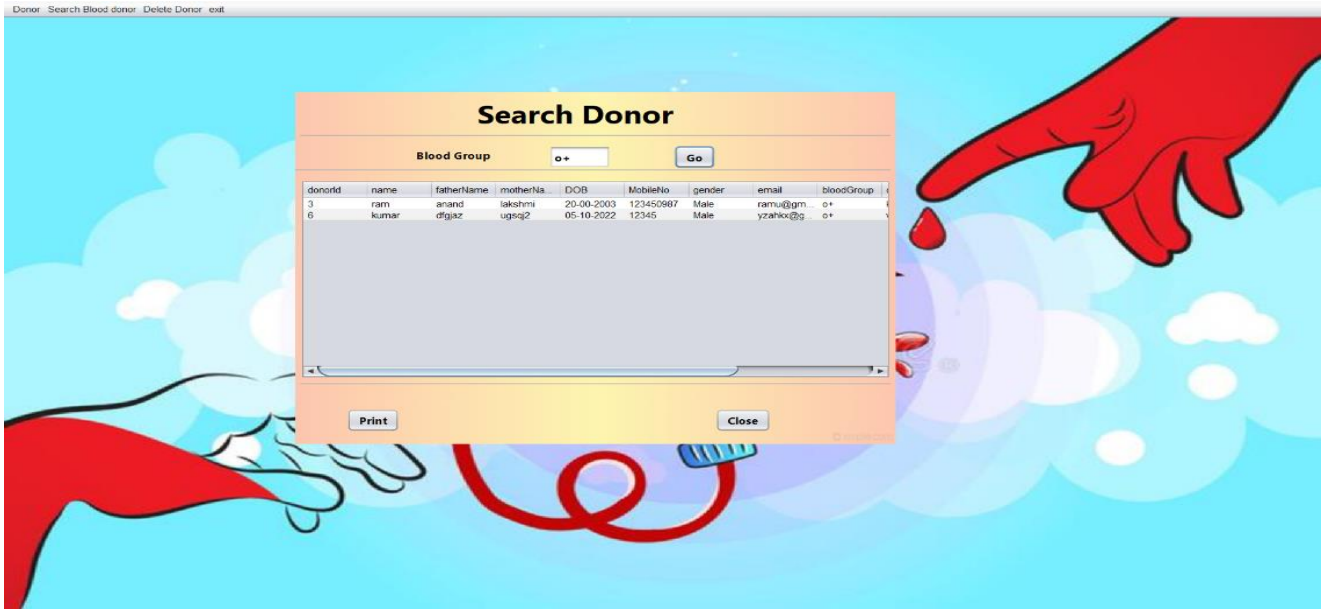
4.OUTPUTS

Login Page : This the staring point of the application where users will log in





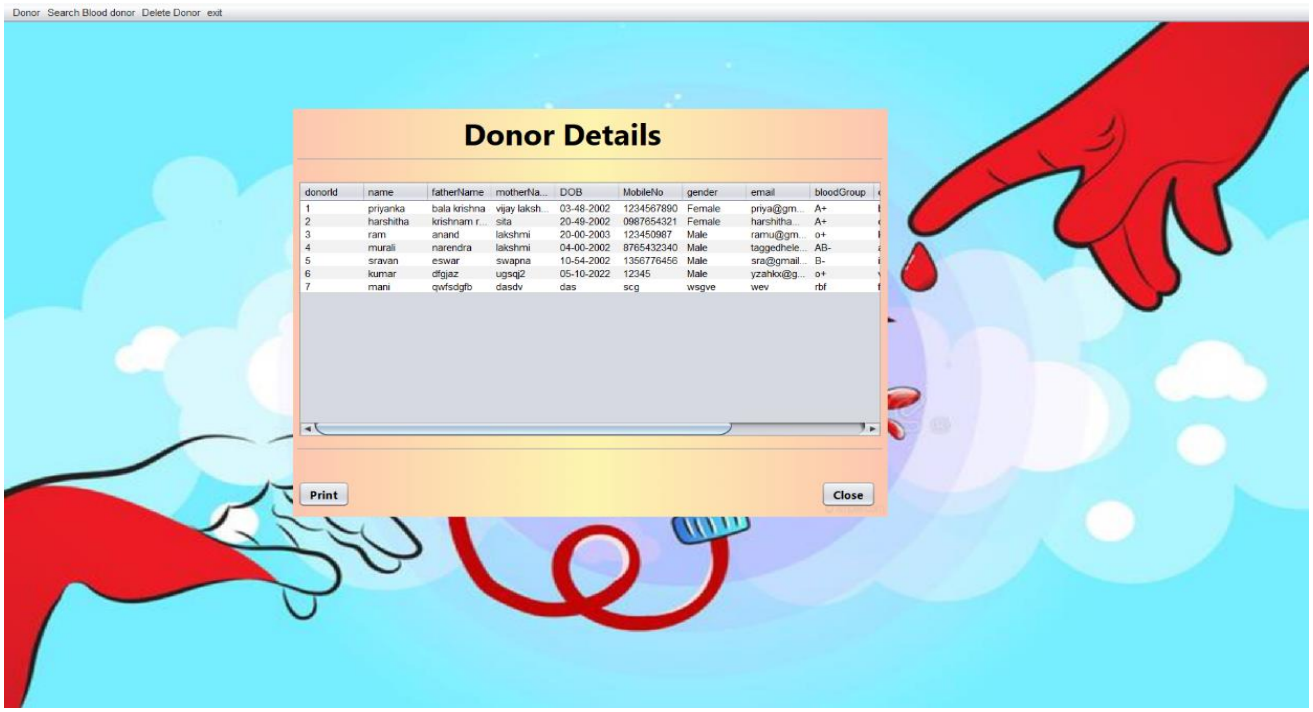
Home page: These are the home pages of users and admin



SEARCH PAGE

On this page, the user can search for the required blood group and select the place where the recipient needs the blood to be delivered /donated.

NEW DONOR DETAILS: Here the details of the donor will be entered.



DONOR DETAILS PAGE:All details of the donor will be displayed here

DELETE DONOR PAGE:we can able delete the donor record using donorId.

	Field	Type	Null	Key	Default	Extra
▶	donorId	int	NO	PRI	NULL	
	name	varchar(100)	YES		NULL	
	fatherName	varchar(100)	YES		NULL	
	motherName	varchar(100)	YES		NULL	
	DOB	varchar(20)	YES		NULL	
	MobileNo	varchar(10)	YES		NULL	
	gender	varchar(10)	YES		NULL	
	email	varchar(100)	YES		NULL	
	bloodGroup	varchar(10)	YES		NULL	
	city	varchar(100)	YES		NULL	
	pincode	varchar(10)	YES		NULL	
	address	varchar(500)	YES		NULL	

Donor table scheme

	Field	Type	Null	Key	Default	Extra
▶	bloodGroup	varchar(5)	YES		NULL	
	name	varchar(30)	YES		NULL	
	MobileNo	varchar(10)	YES		NULL	
	picode	varchar(10)	YES		NULL	
	aadharNo	varchar(12)	NO	PRI	NULL	

Receiver table schema

5. CONCLUSION

Through the "BLOOD BANK MANAGEMENT SYSTEM" project, anybody in need of blood will be able to locate the nearest donor. With the rising need for blood, this programme will undoubtedly assist individuals in society in obtaining the blood they require more conveniently and pleasantly. As a result, this project would undoubtedly benefit in the hunt for a donor with less manual labour; just by using this programme, obtaining information in an emergency would become easier.

6. PROJECT OUTCOME

From this project, we have learnt that how can we identify the problem from our surroundings and how can we help in tackling and giving a solution to this day-to-day problem using technology.

From this project, we came to know various domains and what problems each domain has and how to select a particular problem and derive a particular solution path to the problem we choose. There might be many solutions to a problem but choosing the optimum solution made it easier for our problem.

From this project, we have learnt how to improve our presentation skills, how to make an effective presentation and even learnt how to present our thoughts and implementation through this.

From this project, team management skills improvised a lot, each one of us enthusiastically took each part of theirs and worked on it. We learnt how coordination can help make a bigger problem into a smaller problem and each person was the leader of the team in their way.

From this project, communication skills have got better and how to represent ourselves in front of a crowd improved, communicating among ourselves also helped in making our work easier.

From this project, we have learnt a new software called Netbeans which is used to implement web applications using a java programming language.

Finally, this project helped us in learning new skills, helped us to think out of the box, improve our thoughts towards society and how to take up a problem from our daily life and put some work on it to give a minimal solution which would positively benefit the society.

REFERENCES

1. https://www.researchgate.net/publication/329519303_Blood_Bank_System
2. https://itsourcecode.com/fyp/blood-bank-management-system-project-reportdocumentation-pdf/
3. https://sites.google.com/site/ignoubcafinalyearprojects/project-report/blood-bankmanagement-system-project-report
4. https://projectgurukul.org/python-project-blood-bank-management-system/