Software Requirement Specification

Table of Contents

1.Introduction

As we known importance of blood for human being. To get particular type of blood on time is important to save the life of patient. When patient need blood the problem may occur that particular blood is not available in that hospital. So, we need to search that particular type of blood in the blood bank. This blood banks are available different location in the city so we should go to check for that blood, if that blood not available in that then there is waste of time and we should go for other blood bank.

This Blood Drive Engine web software will help to get the blood to the user for particular blood group if available as early as possible.

1.1 Purpose:-

The purpose of these software is to get the blood for patient on time and easily. No need to check or contact with particular blood bank.

The user will simply login and get information about availability of blood within short time. User can get that information on their registered number through the SMS. This software will improve the blood management system operations of the blood bank.

1.2 Scope:-

The scope of these software is to centralized repository of blood or particular blood bank. Basically the aim of the software to manage the particular type of blood within time and easily so, there will not be any panic situation at that movement.

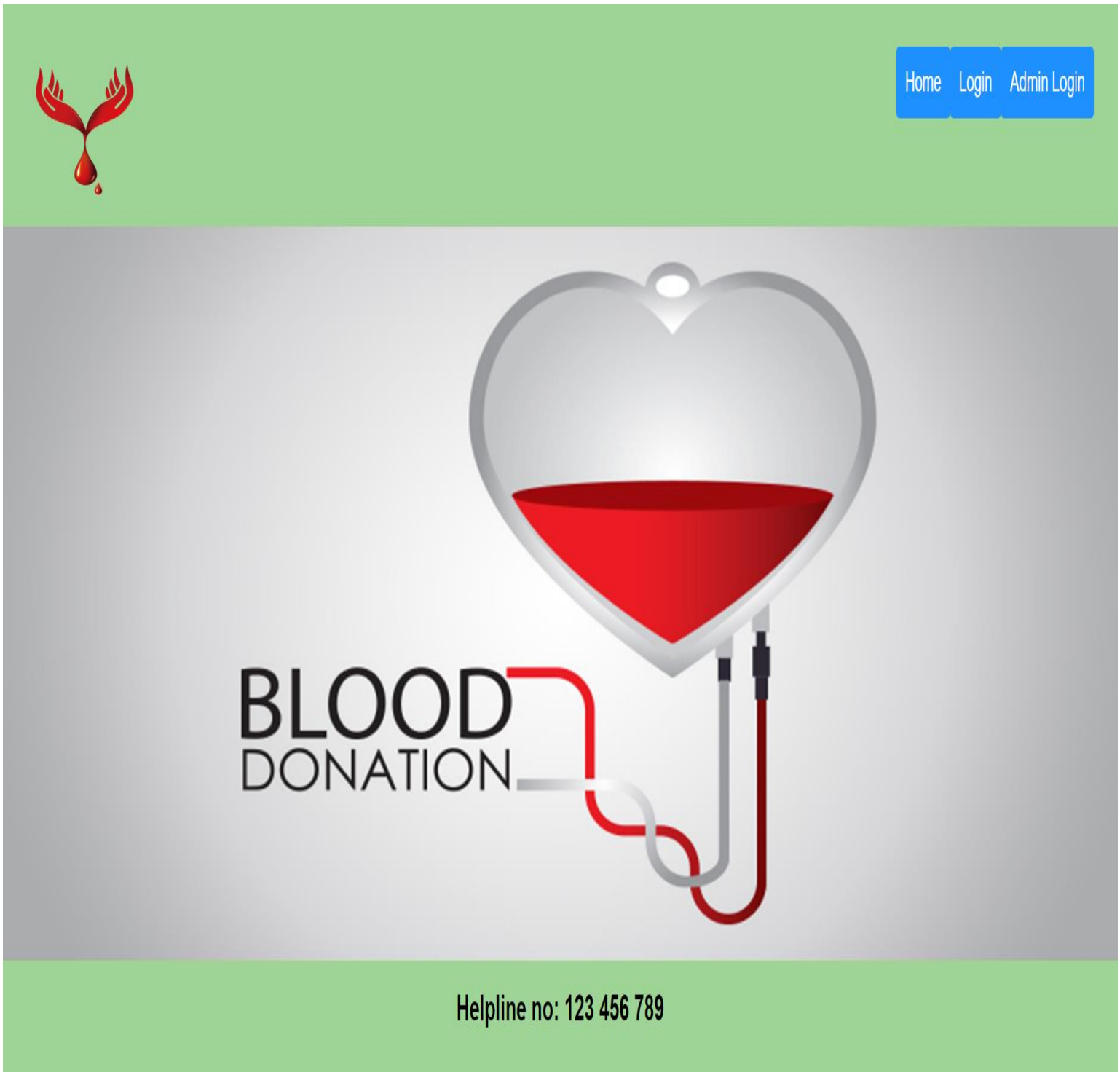
1.3 Product Perspective:-

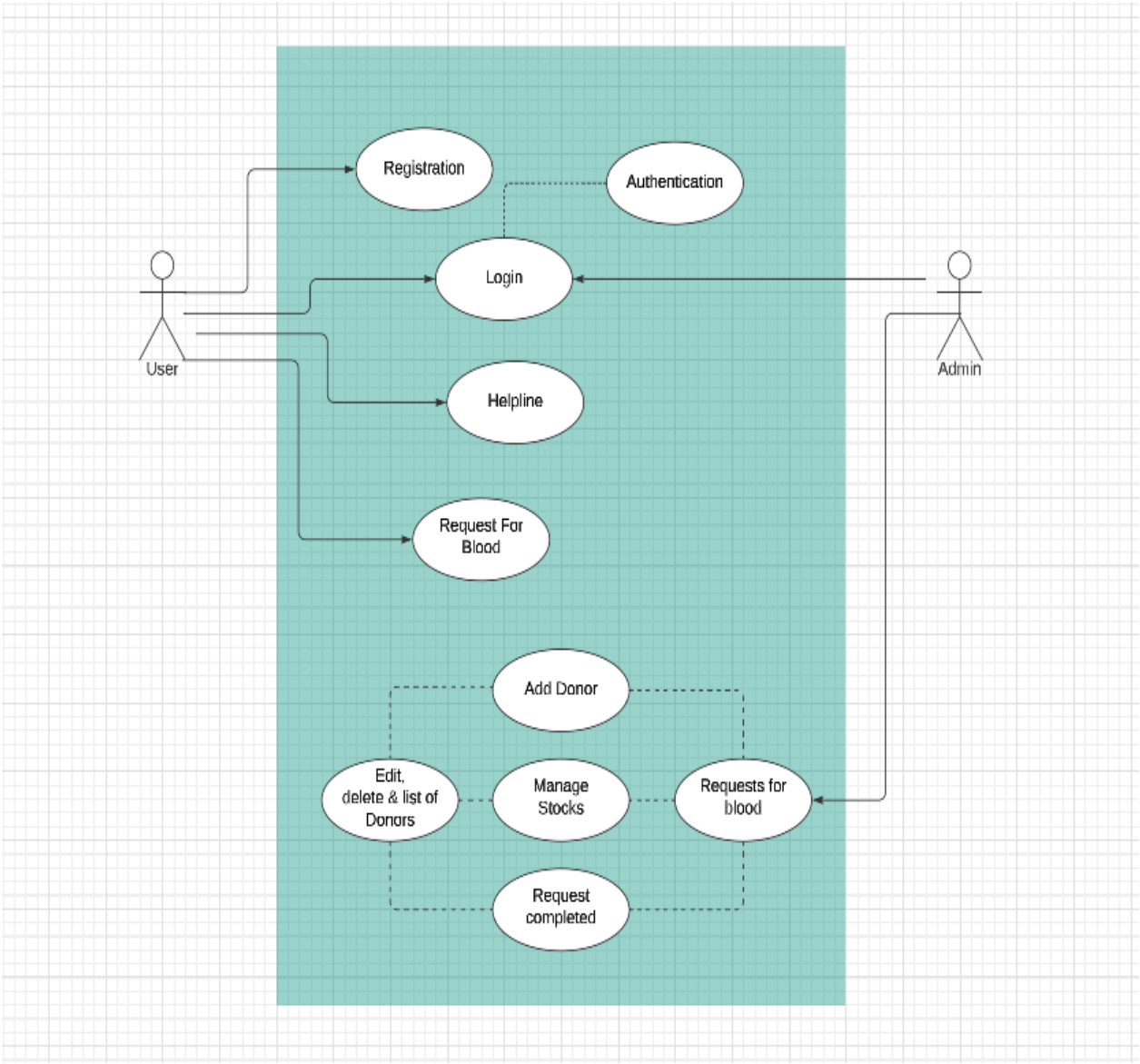
This web application designed in such a way that it will help you to get the required blood group easily. User will simply check the availability of blood which is required. This system also keeps track of number of request and also specific operations done.

The web application provides following features:-

- Information about availability of the blood of different blood groups.
- User can easily request for blood.
- User also get notification through SMS on registered mobile number.
- Admin can add new donor or remove donor.
- Admin can manage the stock of blood.
- Admin can easily manage the requests for blood.
- Through this application admin can easily manage all the operations.

1.4 Product Functions:-





Unified Modelling Language

1.5 User Characteristics

1.5.1 Registration:-

A new user can register can register himself or herself into the system and will get unique user ID and password.
User can fill the basic information at the time of registration like name, address, email ID, phone number etc.

1.5.2 To Log -In into system:-

If the user registered into the system then he/she can login into the system using their unique ID and password.

1.5.3 Request for blood:-

User can easily requested for blood after login. User can fill the requirements like blook group and number of units.

1.5.4 Notification:-

User can get notification regarding availability of blood of specified blood group through SMS on registered mobile number.

1.5.5 Helpline Number:-

User can also get the helpline number of blood bank on home page. By using this helpline number user can easily contact with blood bank and get additional information.

1.5.6 Response from Admin:-

Admin can check the number of request for blood and accordingly admin can manages all the operations. Also admin can make decision on status of request.
That status notify to the user through SMS on registered mobile number.

1.5.7 Donation of blood:-

Admin can add donor or remove from the list of donor.

2. Requirements

2.1 External interfaces

2.1.1 System Interfaces

The application runs in the latest version of chrome, Firefox or Microsoft edge browser on Windows, Linux and Mac.

2.1.2 User Interfaces

The web page shall provide very intuitive and simple interface to the user. So, that one can easily navigate through pages, user's registration page, login process etc.

2.1.2.1 Graphical Interfaces:

That enables person communicate with a computer through the use of symbols, visual metaphors and pointing devices.

2.1.2.2 Menu Driven Interfaces:

our web application provides with range of options in the from of list or menu displayed

2.1.2.3 Form-based Interfaces:

Used to get data from users and save data into database and also requests for blood by the user.

2.1.3 Hardware Interfaces

2.1.3.1 Server side:

The web application is hosted on a web server which is listing on standard web port****.

2.1.3.2 Client side:

Monitor screen - the website shall display information to the via monitor screen.
Mouse – The website shall interact with the movement of the mouse and the mouse buttons.
The mouse shall activate areas for data input, command buttons and select options from menus.
Keyboard – The website shall interact with the keystroke of the keyboard.
The keyboard will input data into active area of the database.

2.1.4 Hardware Interfaces

2.1.4.1 Server side:

An Apache server will accept all request from client and forward it accordingly. A database will hosted centrally using MySQL.

2.1.4.2 Client side:

An OS which is capable of running a modern web browser which supports JavaScript and HTML5. The web page is downloaded from web server and user can interact with this content in the web browser, which acts as a client.

2.1.5 Web Based Interfaces

2.1.5.1 Web browser:

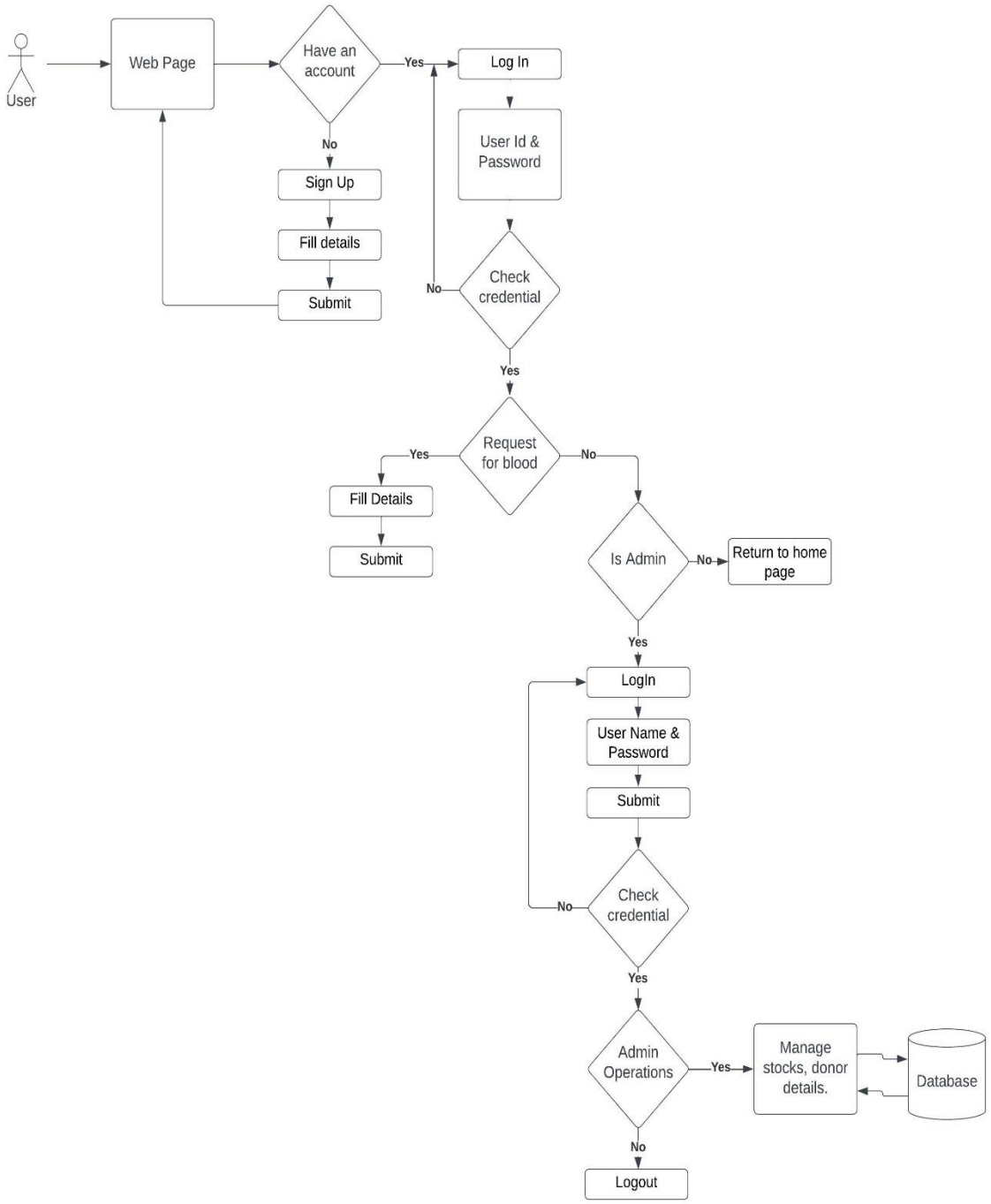
Our website most popularly works on web browser i.e, Google Chrome, Microsoft Edge, Mozilla Firefox and Apple's safari and works ok with all versions and new one.

2.1.5.2 Communication standards and Network standard communication protocols used:

The HTTP or HTTPS protocol(s) will be used to facilitate communications between the client and server.

2.1.5.3 Electronic form:

HTML Forms for user registration and also get data from user which is requested for different blood.



Control Flow Diagram

2.2 Functions

2.2.1 Functional Requirements:

These are the requirements that end user specifically demands as basic facilities that the system should offer. All these functionality need to be necessarily incorporated into system as a part of contract. These are represented or stated in the form of input to be given to the system, operation performed and the output expected. They are basically the requirements stated by the user which one can see directly in the final product, unlike the non-functional requirements.

1) To Log – In into system:

Input: Click on log-in button.
Output: User logged into its account.

2) Sign Up – To create an account:

Input: User should fill up the form with personal details and click on register button.
Output: A pop up will come with confirmation of account created.

3) 24*7 Helpline service:

User can able to see the helpline number on home page.

4) Notifications:

Input: User should fill up the form with details and click on submit button.
Output: User can see the notification on screen your “Form submitted successfully. You will get notify shortly”.

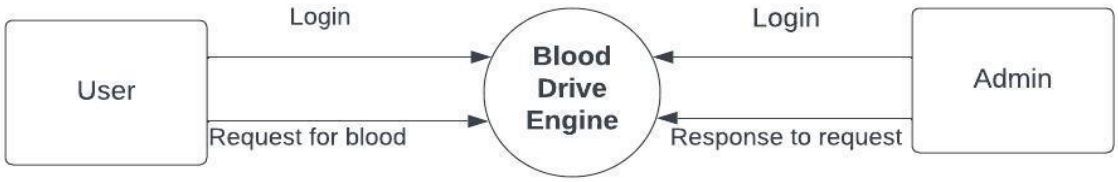
5) SMS service:

Once user fill form and submit form after that he will get message on register mobile number shortly.

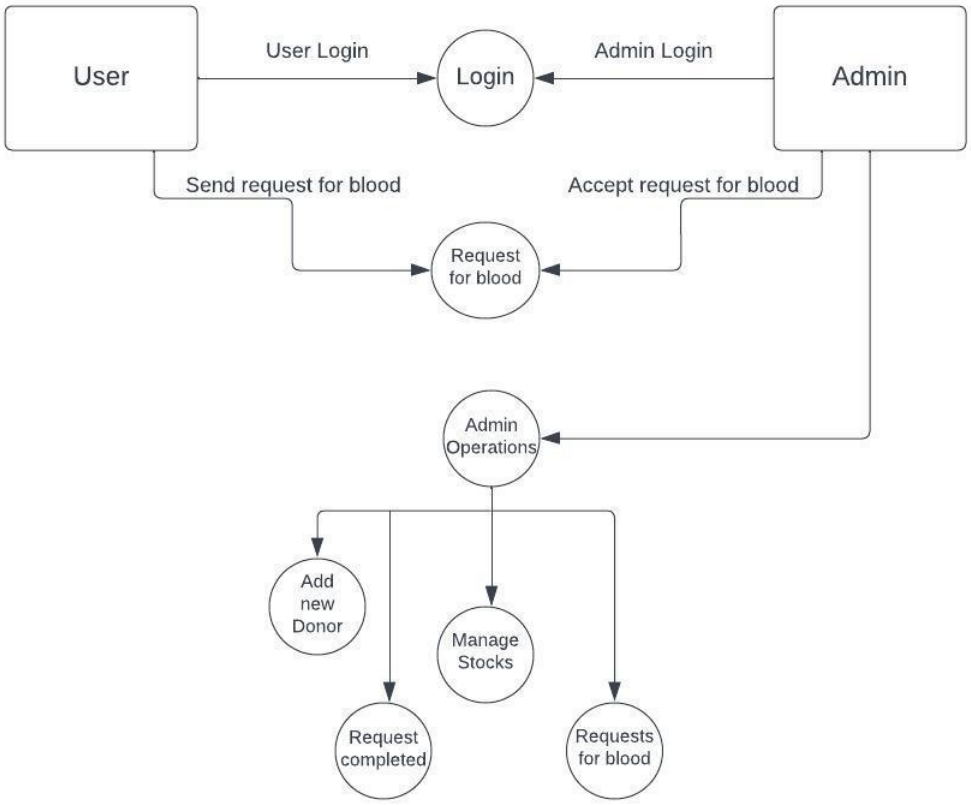
2.2.2 Non-Functional Requirements:

There are basically quality constraints that the system must satisfy according to the project contract. They are also called as non-behavioural requirement.

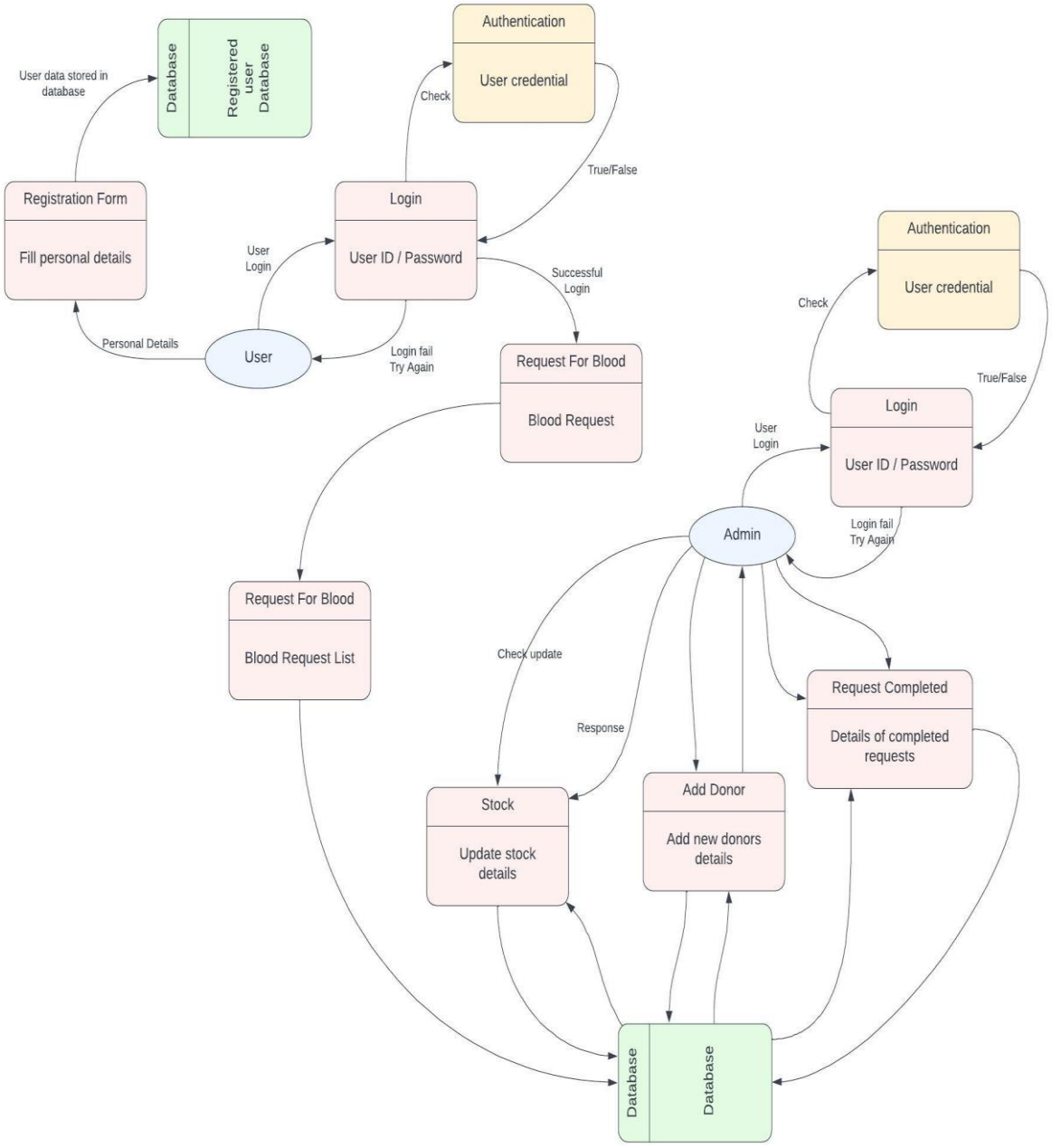
1. Authentication of user whenever he/she logs into the system.
2. Notification for each request should be sent with a latency of not greater than 12 hours from such an activity.
3. The site should load even when the number of simultaneous users is high.
4. This application should run in any Windows, MacOS and Linux platform having browser of any version. It is also available in mini browsers for mobile.



Data Flow Diagram
(Level 0)



Data Flow Diagram
(Level 1)



Data Flow Diagram
(Level 2)

2.3 Usability requirements

The design should support the following requirements for its primary users:

- **Learnability -** Easy-to-learn interface with simple navigation. All the headings, Buttons, massages and errors are easy to understand.
- The most important things is to be place at right place on web page.
- Give correct choice to the user is very obvious way.
- **Navigability** – User will able to perform operations without having multiple pages/links. No operations requires more than 5 to 8 clicks.
- **Familiarity -** The system's interfaces and navigations will be based on other system that the users are familiar with.
- The system will not require any administration tasks at user level.
- **Help –** The system will equipped with computer based tutorial in English and other languages for user to "self-learn" and self "self-solve" any navigability or operational doubts.
- **Memorability –** When users return to the website after a period of not using it, he/she can easily re-establish proficiency.

2.4 Performance requirements

- **Start-up Time** -The application should display the opened document within 10s after it is started.

- **Edit Response Time** - The application should display updated values within 1s after user triggers the edit operation.

- **Smooth Scrolling** - While a user scrolls the requirements table, the application should not display scrolling jerks longer than 200ms.

- Since this software is going to web based, it does require a powerful server machine with high band internet access so that it can handle multiple users at the same time.

- The web application should be developed as a lightweight web app so that it can work on almost any platform even with slower internet connections.

- To make the web application lightweight, simple libraries and tools should be used at developing phase.

- System should be able to deal with multiple users at the same time. Also, database of the system should handle at least a thousand of users at any periods.

- To improve portability, software should run on variety of platforms and variety of connection speeds.

- Portability also means running on different platforms without an additional effort. To achieve this, web application should be developed by using the common technologies and tools which are provided by all common web browsers and operating system.

2.5 Logical database requirements

A Logical Database is a special type of ABAP (Advance Business Application and Programming) that is used to retrieve data from various tables and the data is interrelated to each other. Logical Database we will use joins instead of multiple SELECT statements, which will improve response time and this will increase the Performance of Logical Database.

2.5.1 Below is some important task of Logical Database:

- With the help of the Logical database, we can read the same data from varies programs.
- A database provides the same user interface for multiple programs.
- Database ensures the Authorization checks for the centralized sensitive database.
- When the structure of data is large it is convenient to store it in database.
- We can easily retrieve, modify, save, delete the data using logical database.
- Different functional operation can be performed to retrieve required data from database. Like select, join, group by etc.

2.5.2 Designing Database:

ER model helps to systematically analyse data requirements to produce a well-designed database. **ER Diagram** stands for Entity Relationship Diagram, also known as ERD is a diagram that displays

the relationship of entity sets stored in a database. In other words, ER diagrams help to explain the logical structure of databases. ER diagrams are created based on three basic concepts: entities, attributes and relationships.ER Diagrams contain different symbols that use rectangles to represent entities, ovals to define attributes and diamond shapes to represent relationships.

2.5.3 Components of ER Diagram:

- **Entities :** A real-world thing either living or non-living that is easily recognizable and nonrecognizable. An entity can be place, person, object, event or a concept, which stores data in the database. The characteristics of entities are must have an attribute, and a unique key. Every entity is made up of some 'attributes' which represent that entity.

- **Attributes :** It is a single-valued property of either an entity-type or a relationship-type.

- **Relationships:** Relationship is nothing but an association among two or more entities. E.g., Tom works in the Chemistry department.

- **Cardinality:** Defines the numerical attributes of the relationship between two entities or entity sets. Different types of cardinal relationships are One-to-One Relationships, One-to-Many Relationships, May to One Relationships & Many-to-Many Relationships

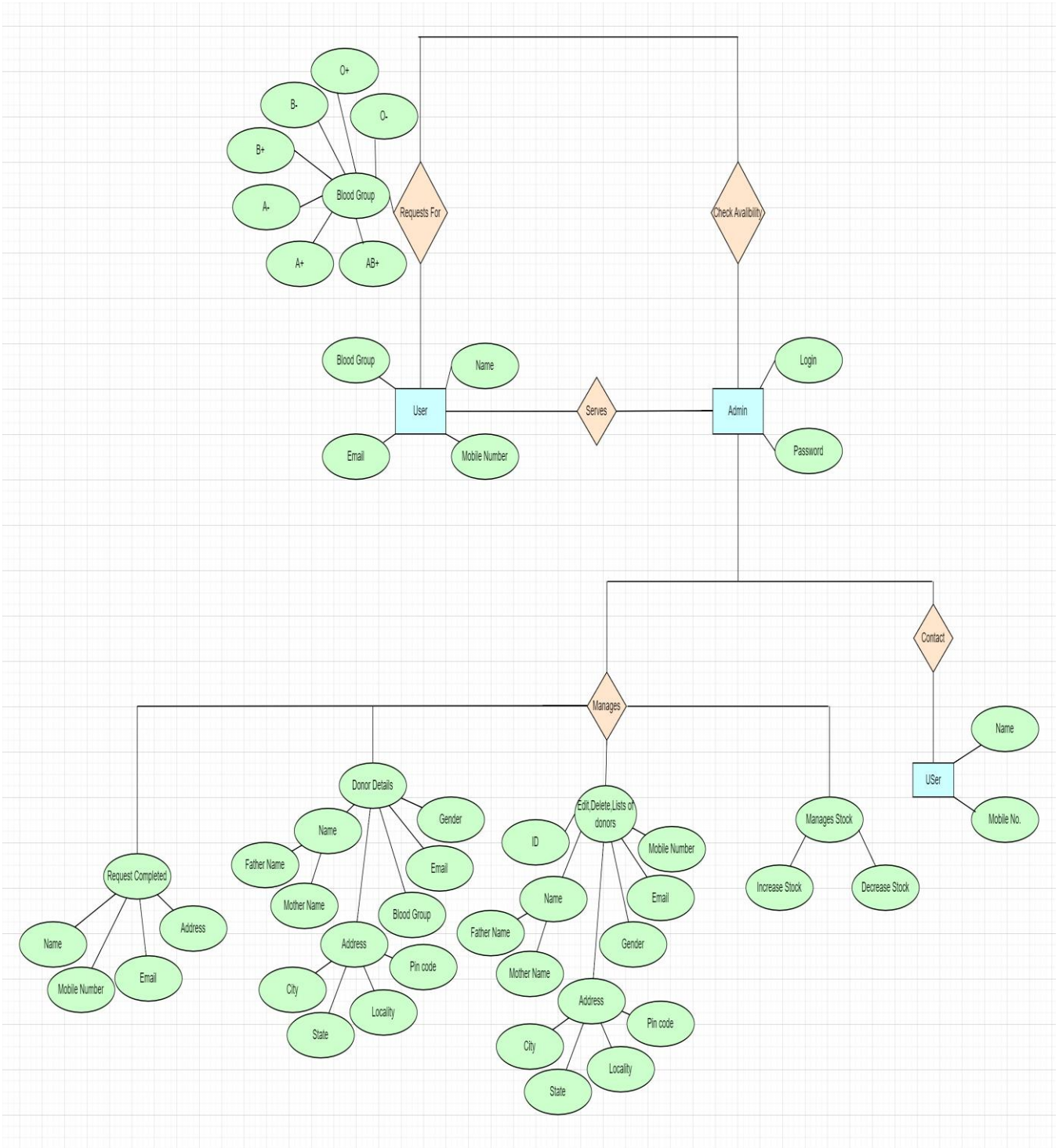
Following are the main components and its symbols in ER Diagrams:

- **Rectangles:** This Entity Relationship Diagram symbol represents entity types.

- **Ellipses :** Symbol represent attributes.

- **Diamonds:** This symbol represents relationship types.

- **Lines:** It links attributes to entity types and entity types with other relationship types.
- **Primary key:** attributes are underlined.
- **Double Ellipses:** Represent multi-valued attributes.


O+
B-
O-
B+
A-
Blood Group
A+
AB+
Requests For
Check Availibility
Blood Group
Name
User
Serves
Admin
Login
Email
Mobile Number
Password
Contact
Manages
Name
USer
Mobile No.
Donor Details
Gender
Name
Father Name
Email
Mother Name
Blood Group
Address
Pin code
City
Locality
State
Request Completed
Address
Name
Mobile Number
Email
Edit,Delete,Lists of donors
ID
Mobile Number
Name
Email
Father Name
Gender
Mother Name
Address
Pin code
City
Locality
State
Manages Stock
Increase Stock
Decrease Stock

E-R Diagram

2.6 Design Constraint:

- **User Interface Constraints:** Using this system is fairly simple and intuitive. A user familiar with basic browser navigation skills should be able to understand all functionality provided by the system.

- **Hardware Constraints:** The system should work on most home desktop and laptop computers which support Java and HTML5.

- **Software Constraints:** The system will be intended to run on Firefox, Google Chrome and Internet Explorer.

- **Data Management Constraints:** System shall be able to interface with other components according to their specifications.

- **Operational Constraints:** The system is limited by its operating server in terms of the maximum number of users it can support at a given time.

- **Server Side:** The disk space on Server should always be at least 10GB available.

- **Client Side:** The client's browser must be a modern browser and support HTML5 standards.

2.7 Standard Compliance

2.8 Software system attributes

The software must consist following attributes.

- **Reliability**

  The reliability of the overall system depends on the reliability of the separate components. A powerful server should be able to handle multiple requests. The server also should be able to process each request and serve back by sending a response.

- **Availability**

  The system should be available at all times, meaning the user can access it using a web browser, only restricted by the down time of the server on which the system runs. In case of a of a hardware failure or database corruption, a replacement page will be shown. Also, in case of a hardware failure or database corruption, backups of the database should be retrieved from server and saved by the client.

- **Security**

  Passwords will be saved encrypted in the database in order to ensure the user's privacy and user's IP will be logged. The system will be protected against vulnerabilities like injection attacks.

- **Maintainability**

  Database is used for maintaining the user’s data records and the server takes care of the site. In case of a failure, a re-initialization of the program is recommended.

3 Verification

4 Supportive Information

5 References