**INTRODUCTION**

The Blood Donation Agent is to create an e-Information about the donor and organization that are related to donating the blood. Through this application any person who is interested in donating the blood can register himself. If any general consumer wants to make request blood online he can also take the help of this site. Admin is the main authority who can do addition, deletion, and modification if required.

**Problem Definition**

**Existing System**

* Cannot Upload and Download the latest updates.
* No use of Web Services.
* No proper coordination between different Applications and Users.
* Fewer Users – Friendly
* .

**Scope Of a Proposed System**

To debug the existing system, remove procedures those cause data redundancy, make navigational sequence proper. Can provide information about audits on different level and also to reflect the current work status. This system has strong password mechanism. New system is more user friendly. There is a proper coordination between the different application and the user. We can easily update the latest updates. This system is used to manage the blood bank of a certain organization.

**Objectives Of a proposed system:**

The objective of this system is for the blood bank management. Donor registration , Managing the donor database ,recording their physical and medical statistics. Registration of the new employee, editing the information of the employee and deleting the information of the employee of an organization. List of donor who are eligible for donation on a particular date with the contact number. Checking the availability of certain blood group in a blood bank;

**Requirement gathering :**

For requirement gathering of a system design different techniques are used such as interview, record review, questionnaire and observation. For the online blood bank management system we have studied the existing system in the detail. The information for this system is gathered from record review like books, manuals, existing system documents.

Requirements are gathered from interviewing the users of the system the system and the observation.

**Identify End User of The System :**

An end user is the person that a software program or hardware device is design for. The end user is the person who use the software or a hardware after it has been fully developed, marketed and installed. It is also the person who keeps calling the “IT Guy” with questions about why the product is not working correctly. Generally, the terms “user” and “end user” means the same thing. The end user of the this system are admin who manages the whole system. Users who can register themselves for the blood donation and employee of the organization.

**Input Data to The System :**

Very careful attention had to be given to input design which is major part of

the overall system design. In order to make the data entry as easy, logical and error free as possible, specific standard has been followed ,validation checks provided in the system prevented the user in entering incorrect, erroneous data. This made sure that only valid data had been available for data processing. If valid data was entered, then meaningful errors massages had been prompted to enter correct data.

1. Admin gives his login id and password for the login.

2. User while joining the gives the input information such as name, date of birth, city, address, blood group and emails.

3. Input information of the blood donor is entered by the employee of the system.

**Output information from the system:**

Output, as you probably know, generally refers to the results and information that are generated by the system. Output of the system this system is members who joined for the blood donation. Blood donation information of the blood donor. Availability of blood in the blood bank. Details of Employee of the organization.

**Platform**

* WINDOWS OS
* Visual Studio Code
* XAMPP Secrver/Mysqli

**Tables**

**Donor**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Sno** | **Columnname** | **Datatype** | **Constraint** | **description** |
| 1 | id | Int(11) | Primary Key | Doner id |
| 2 | FirstName | varchar(45) | Not Null | First name |
| 3 | MiddleName | varchar(30) | Not Null | Middle name |
| 4 | LastName | varchar(40) | Not Null | Last name |
| 5 | Sex | Var(10) | Not Null | sex |
| 6 | h\_address | Varchar(50) | Not Null | Donor address |
| 7 | city | Varchar(30) | Not Null | city |
| 8 | don\_date | date | Not Null | Donation date |
| 9 | stat | text | Not Null | stat |
| 10 | temp | Varchar(10) | Not Null | Temperature in degree |
| 11 | pulse | Varchar(10) | Not Null | Pulse rate  60-100 |
| 12 | bp | Varchar(10) | Not Null | Blood pressure  80-120 |
| 13 | weight | varchar(20) | Not Null | Weight in kg |
| 14 | heamoglobin | varchar(25) | Not Null | Heamoglobin level  16-18 |
| 15 | hbsag | varchar(10) | Not Null | Hepatitis b  Negative/positive |
| 16 | aids | Varchar(15) | Not Null | Negative/positive |
| 17 | malaria\_smear | Varchar(20) | Not Null | Negative/positive |
| 18 | hematocrit | Varchar(15) | Not Null | Red blood cell % by volume  40.7%-50.3% |
| 19 | phone | Varchar(10) | Not Null | Phone number |
| 20 | mobile | Varchar(11) | Not Null | Mobile number |

**Employee**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Sno** | **Columnname** | **Datatype** | **Constraint** | **description** |
| 1 | Id | Int (11) | Not Null | Employee if |
| 2 | f\_name | Varchar(35) | Not Null | First name |
| 3 | l\_name | Varchar(30) | Not Null | Last name |
| 4 | username | Varchar(15)  Unique,primary key | Not Null | username |
| 5 | password | Varchar(15) | Not Null | password |
| 6 | b\_day | date | Not null | Birth date |
| 7 | Pre\_nr | Int(25)  Unique  Primary key | Not Null | Employee no |
| 8 | desig | Varchar(35) | Not Null | designation |
| 9 | landline | Varchar(10) | Not Null | Landline number |
| 10 | mobile | Varchar(11) | Not Null | Mobile number |

User

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Sno** | **Columnname** | **Datatype** | **Constraint** | **description** |
| 1 | id | Int(11) | Not null | User id  Primary key |
| 2 | First\_name | Varchar(11) | Not null | First name |
| 3 | Last\_name | Varchar(100) | Not Null | Last name |
| 4 | email | Varchar(100) | Not Null | Email address |
| 5 | dob | date | Not Null | Date of birth |
| 6 | gender | Varchar(10) | Not null | gender |
| 7 | b\_type | Varchar(10) | Not Null | Blood group |
| 8 | address | Varchar(500) | Not Null | Address of user |
| 9 | city | Varchar(100) | Not null | City of user |
| 10 | mobile | Varchar(15) | Not null | Mobile number of user |

**Data flow Diagram**

**Context level DFD**

user

Employee login

User Employee

Admin's features and tasks

Registration log in details

Details

Admin login

Donor

Details

**0.0** donor

Donors Details

Admin's login

Blood Donor

Details

Employee

details Member

Employee Details

Member details

Blood Donor

Details

**User dataflow diagram:**

Reg

details form Details member

details

**Data flow diagram Admin:**

Admin details

user

**login**

Employee details

View

details donor

**Employee Details**

verification details donor

Employee details donor

View

details donor

**Donor Details**

verification details donor

**Member details**

View donor

Donor details

Employee

ER- Diagram :

M

M

M

1

1

1

Member

Has

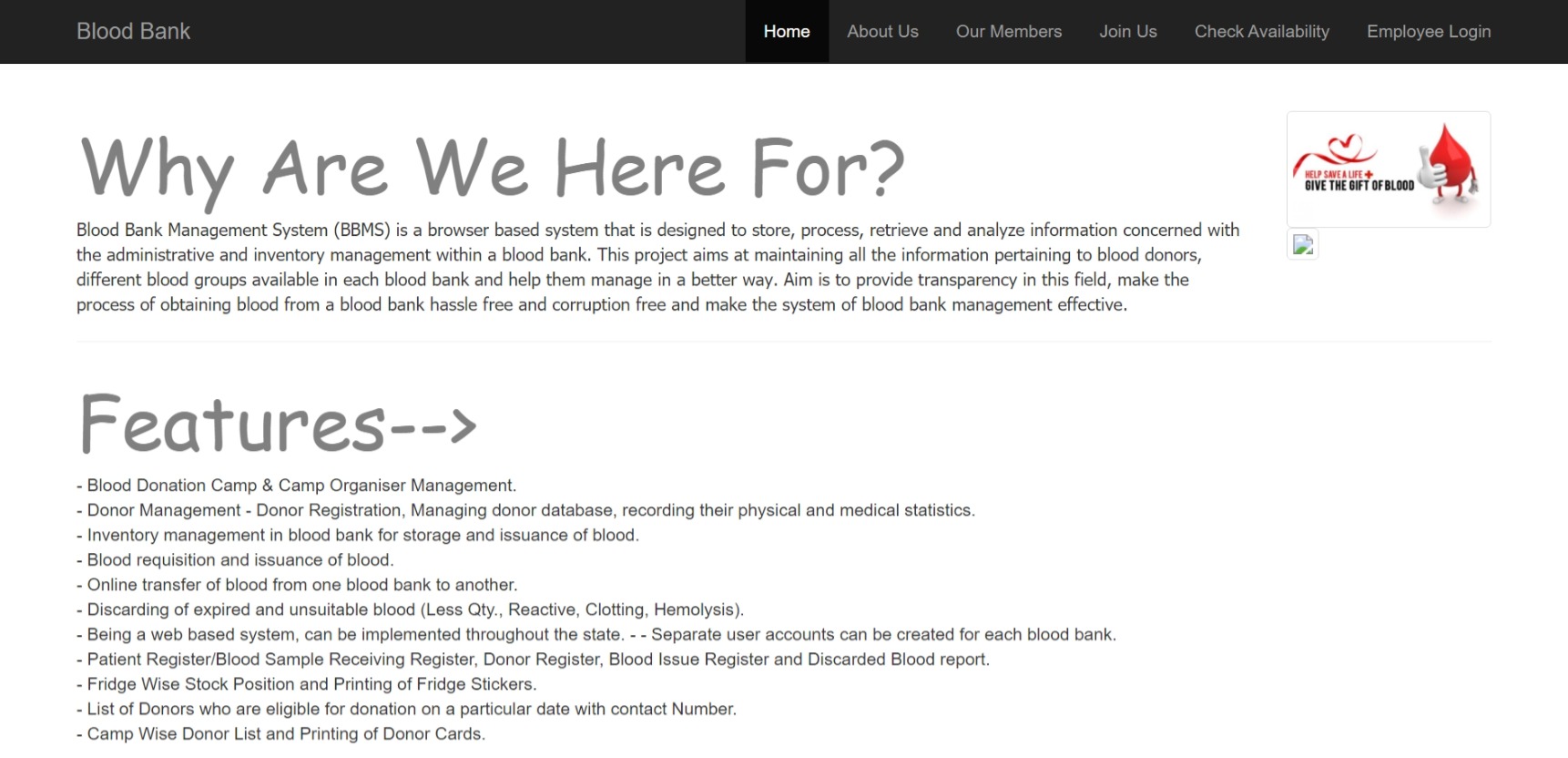
Has

Employee

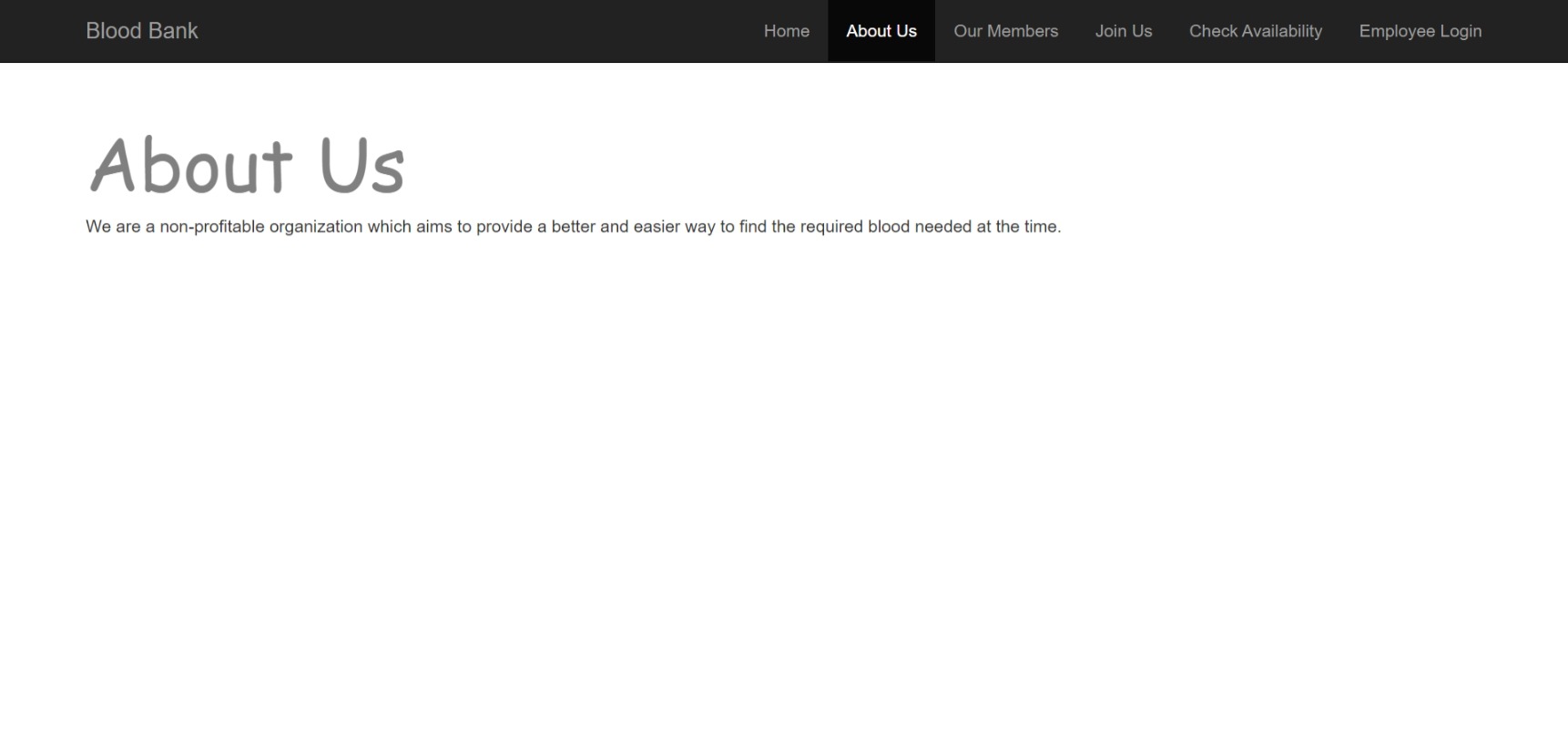
Donor

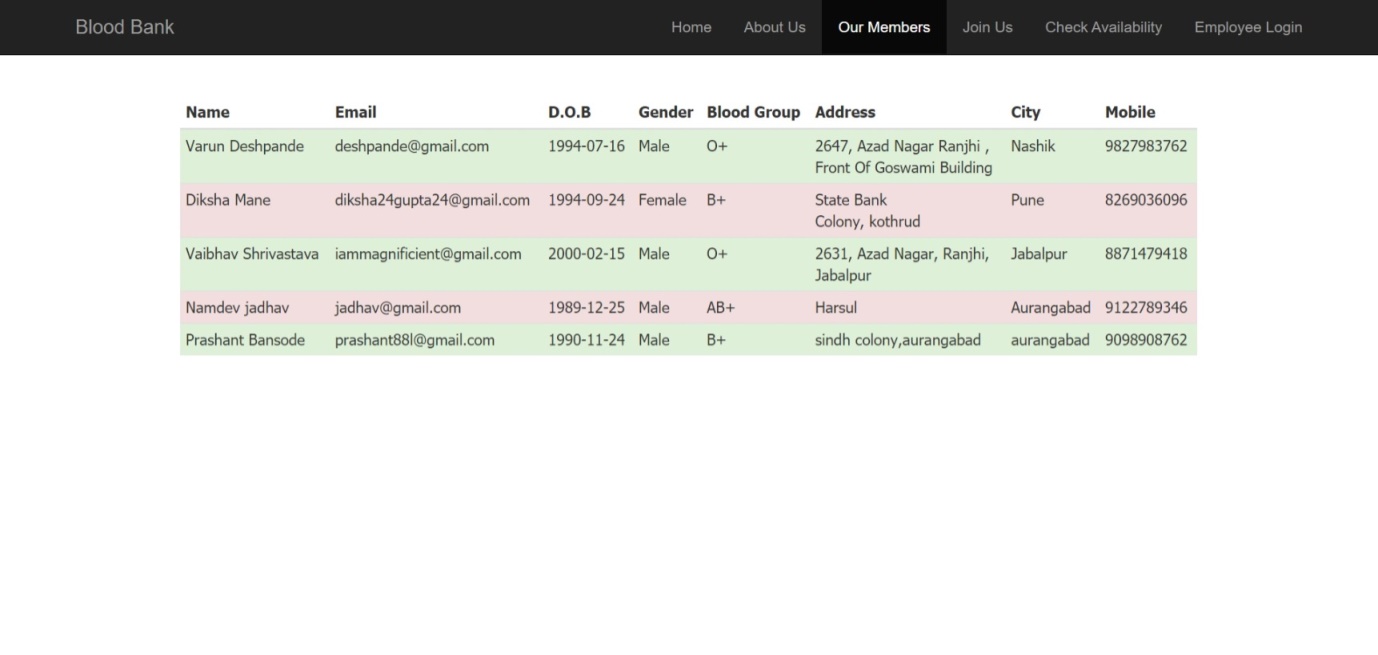
Has

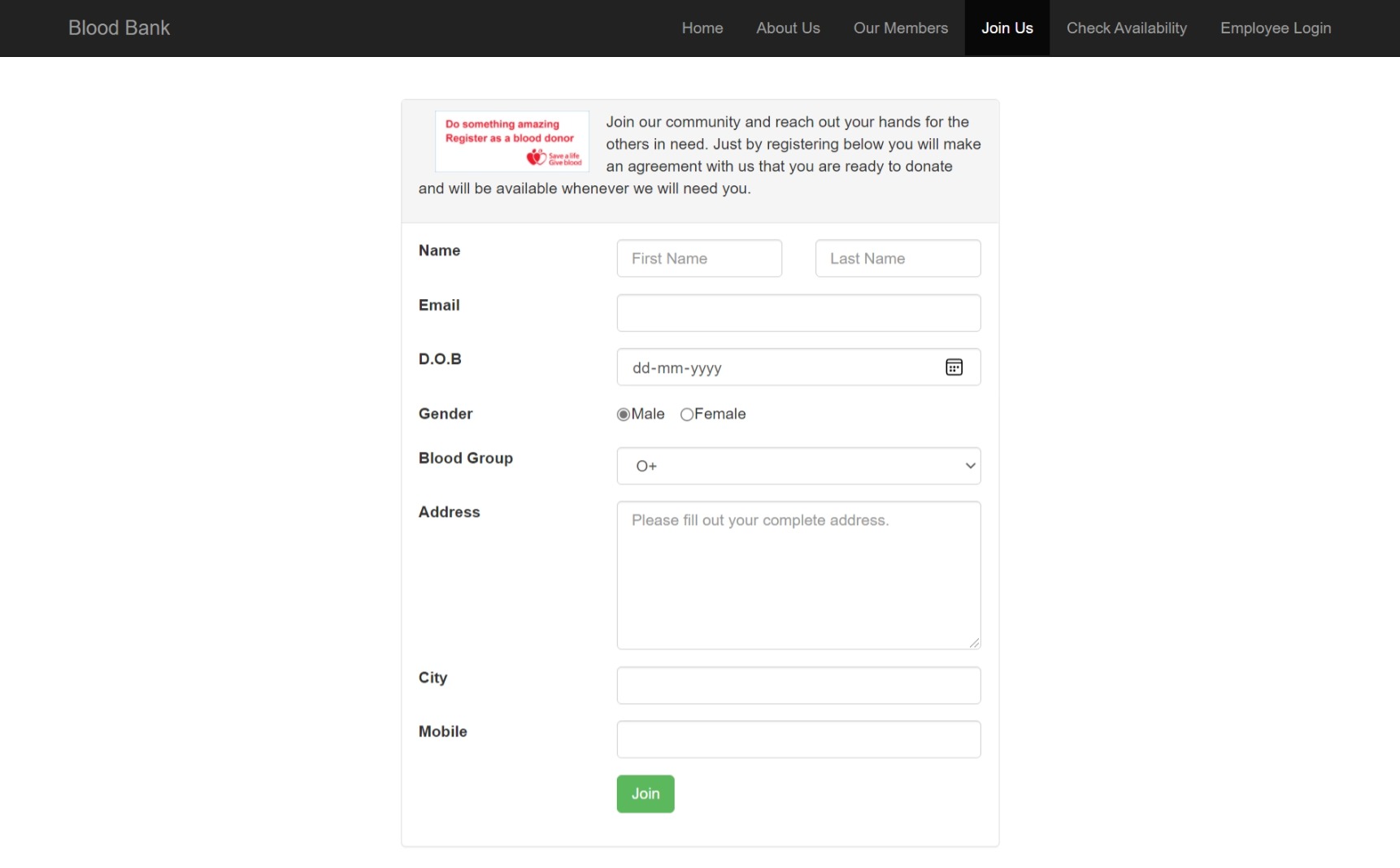
Organization

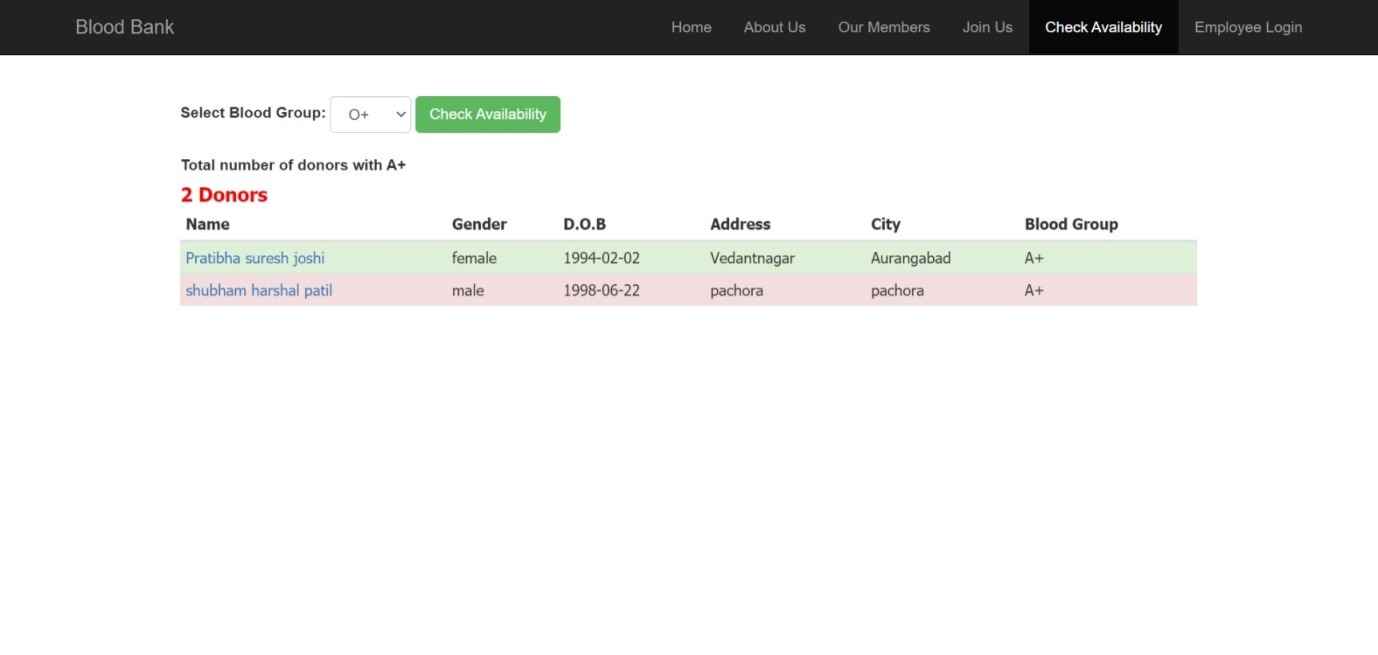
 Input and output screen

**Home screen**

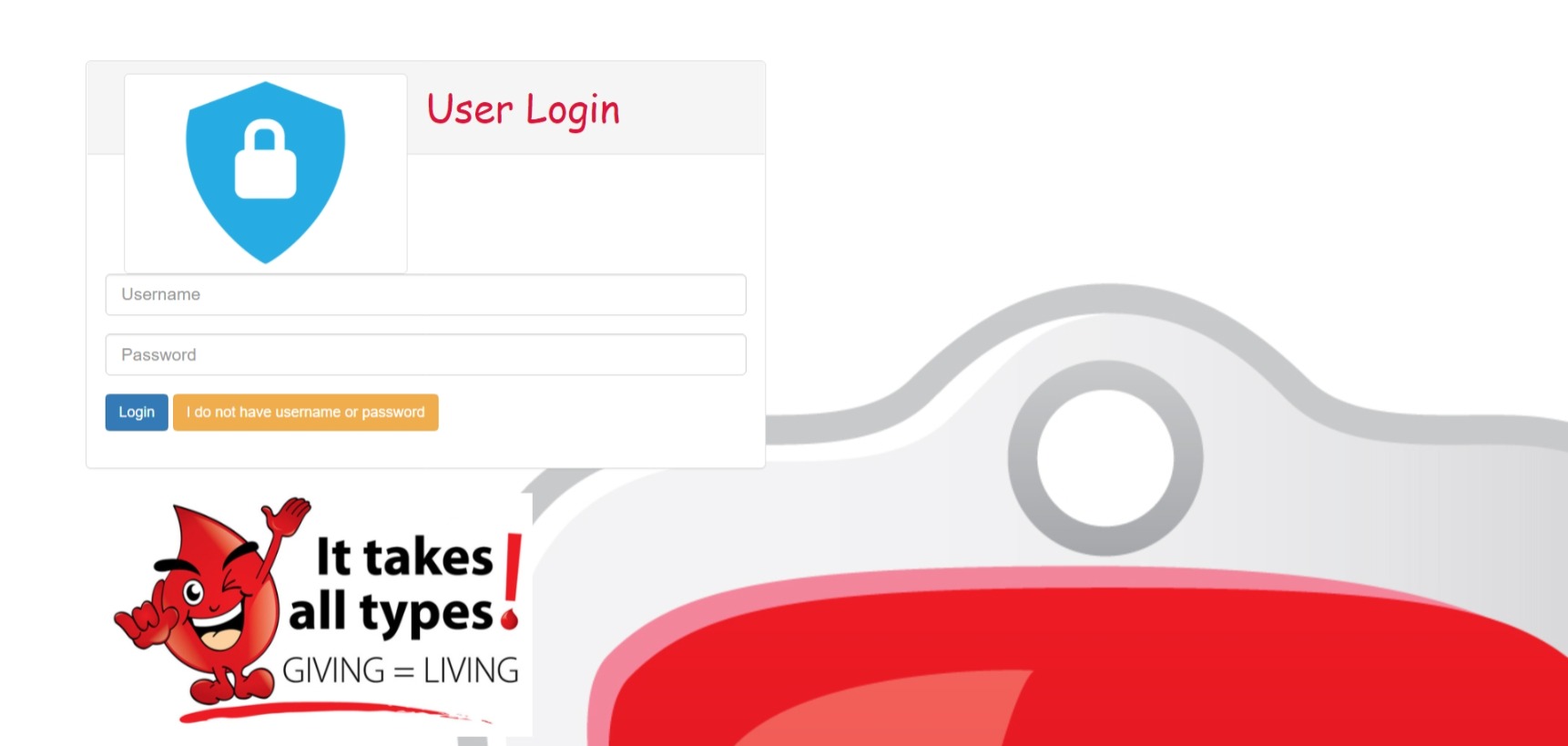
About us

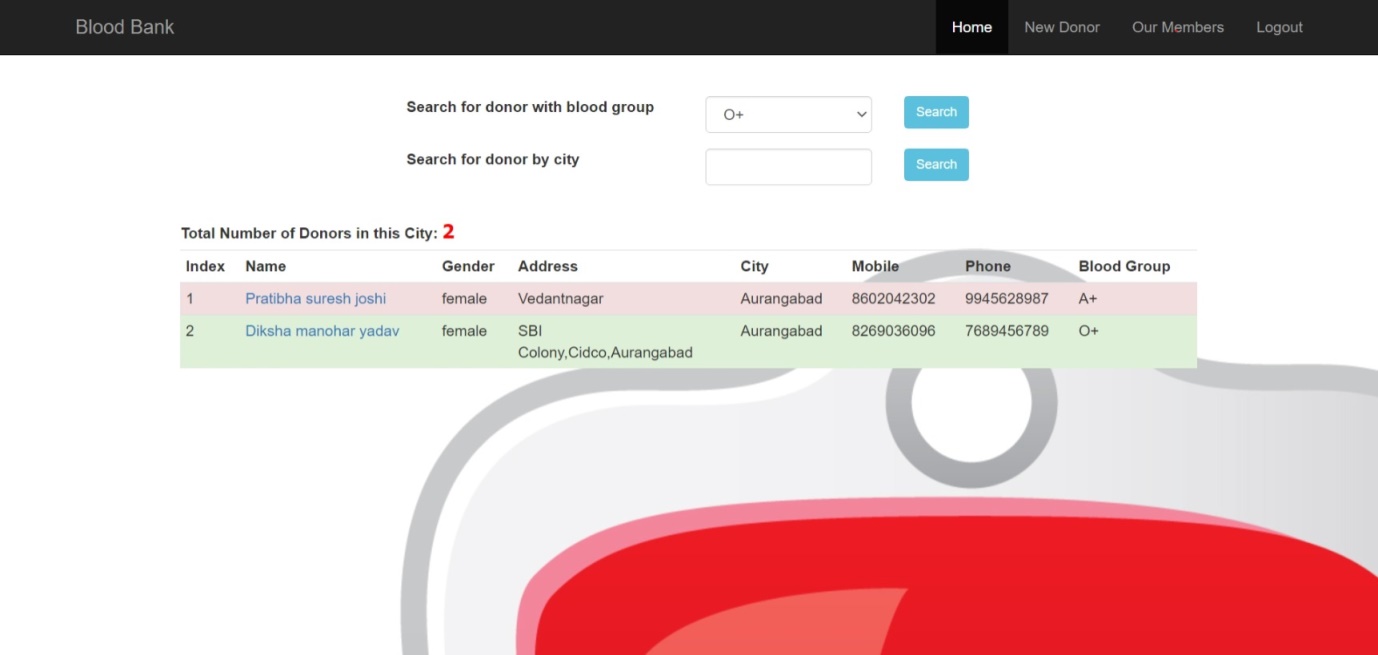
Our members output page

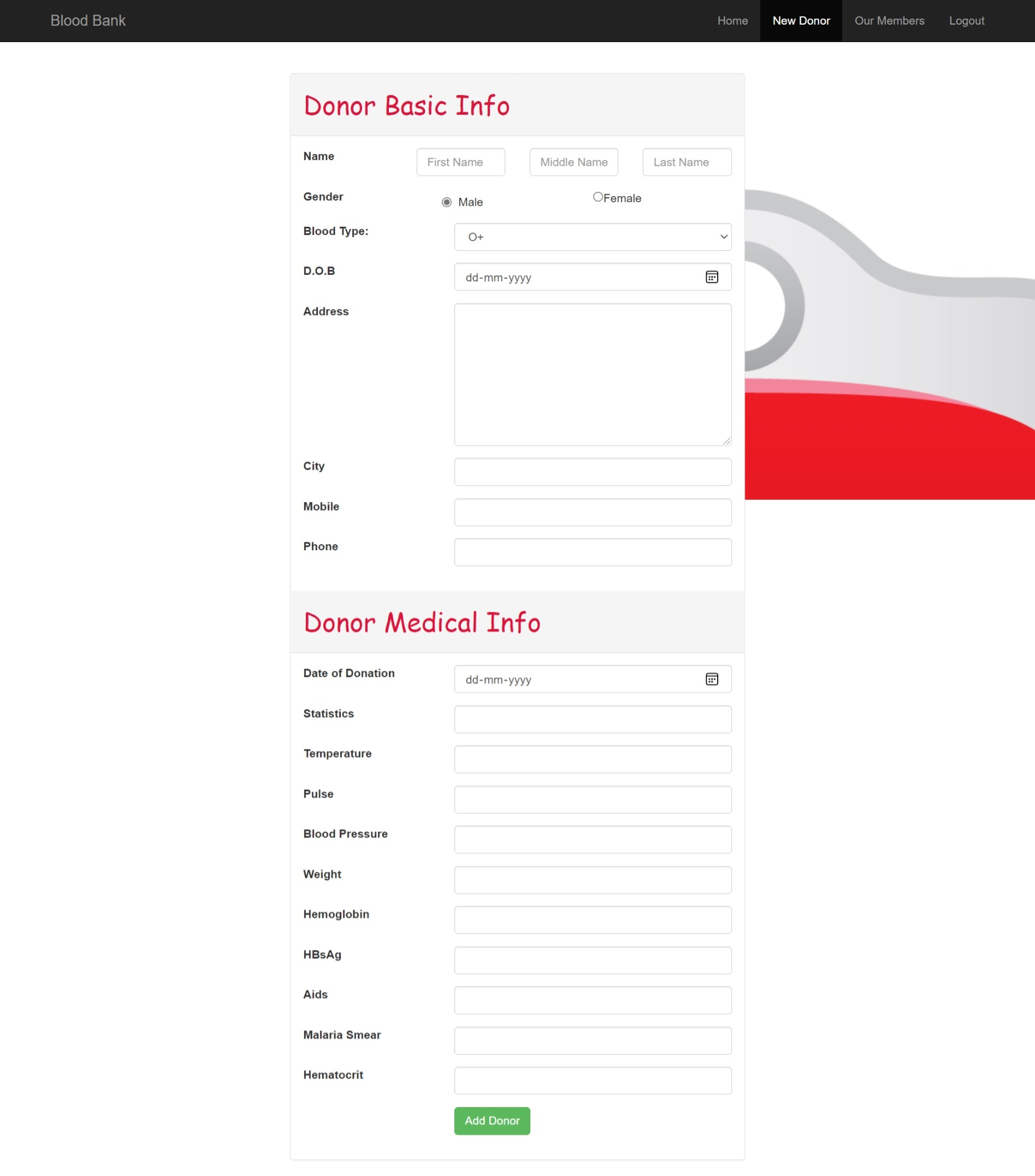
 Join us /User Input Page

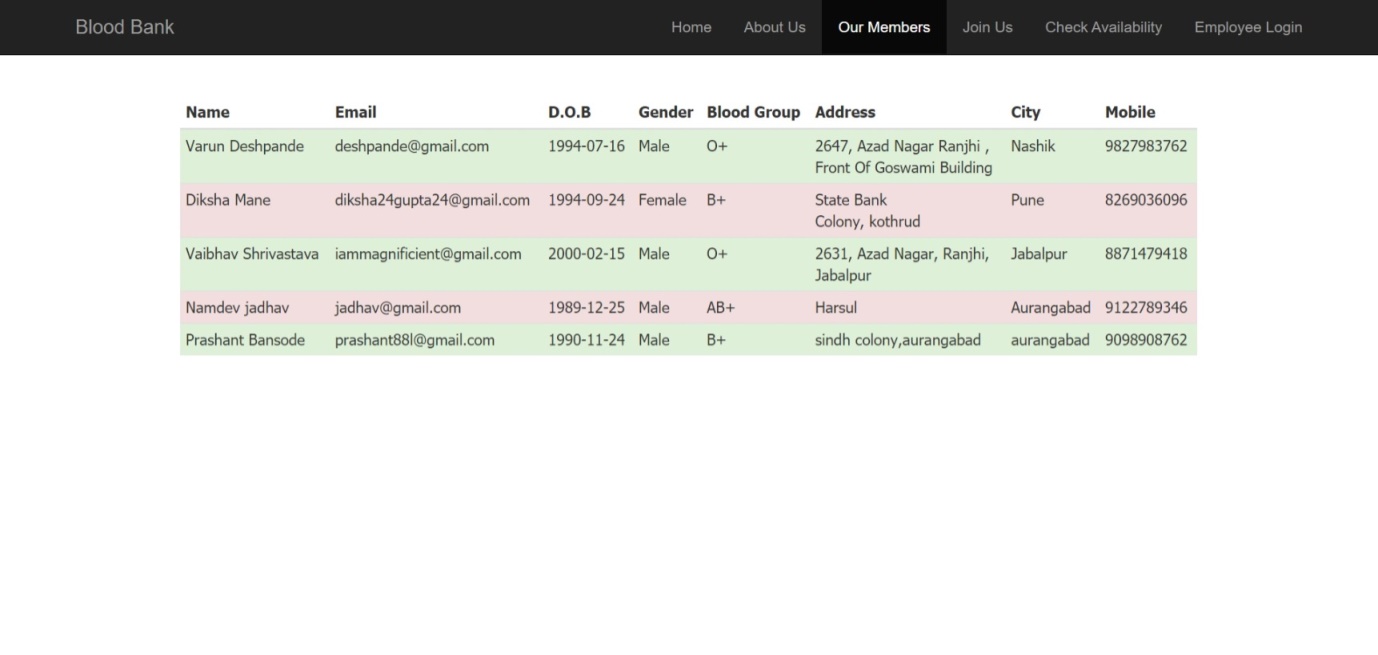
Check Availability/Output Page

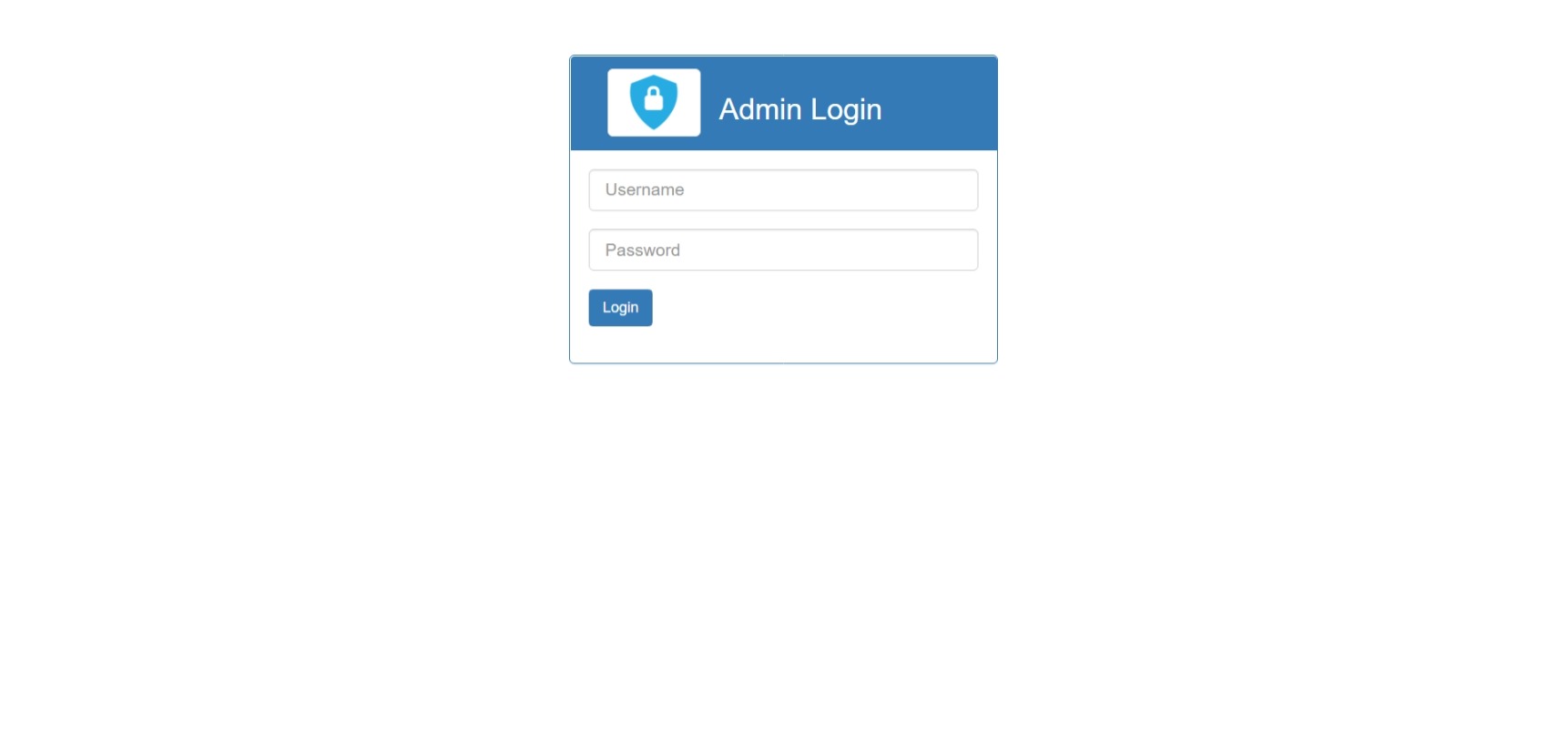
Employee login Page

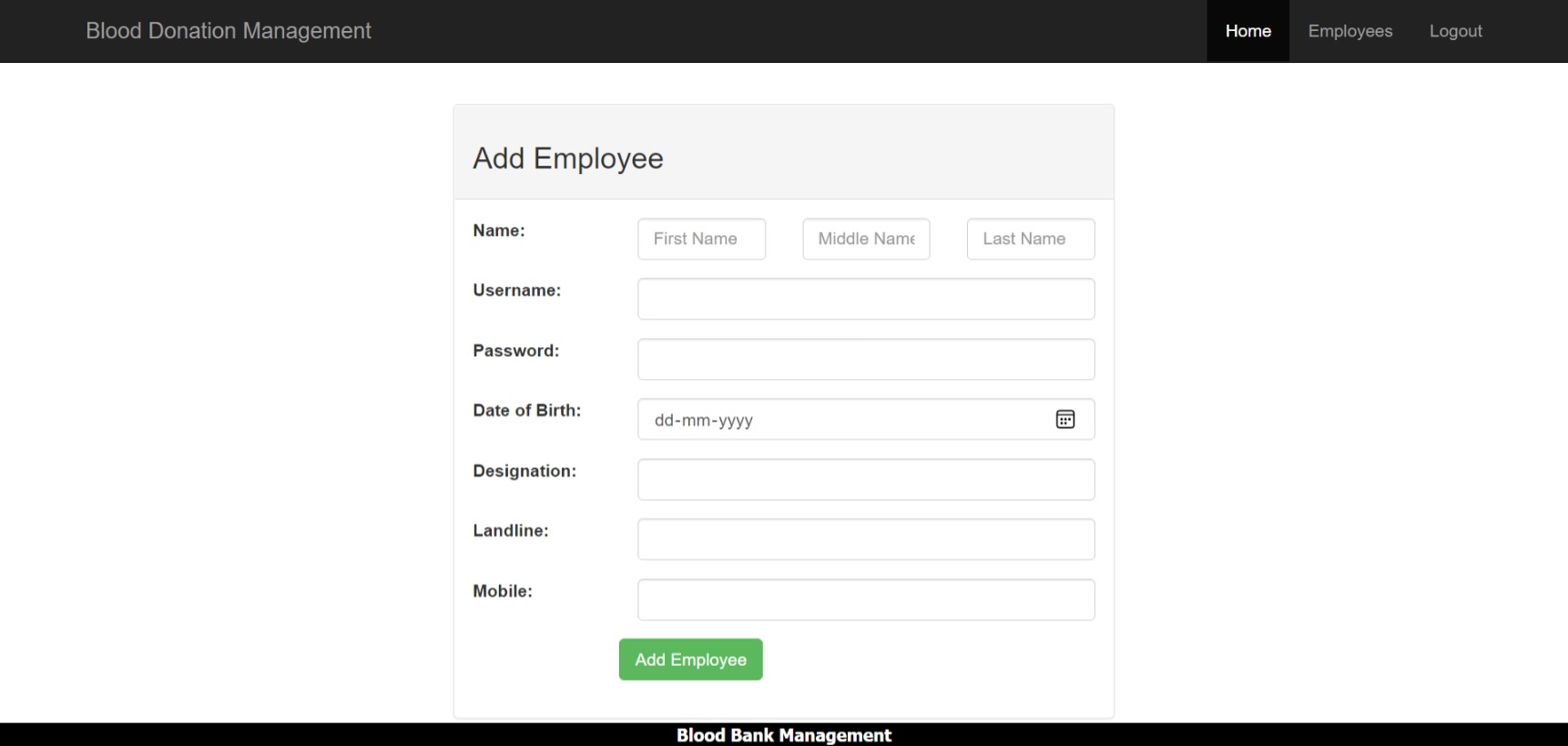


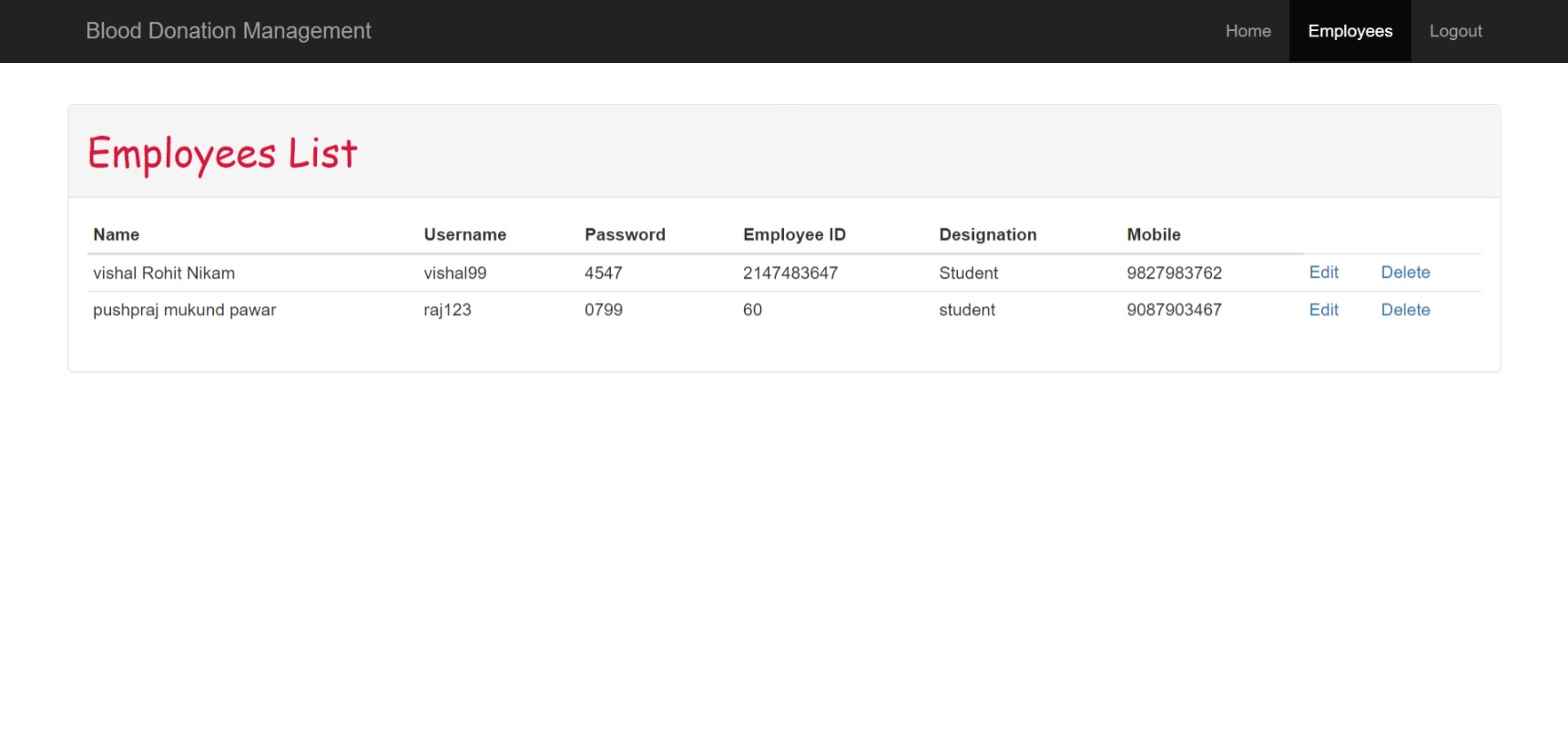
Employee Home Page-Search Donor

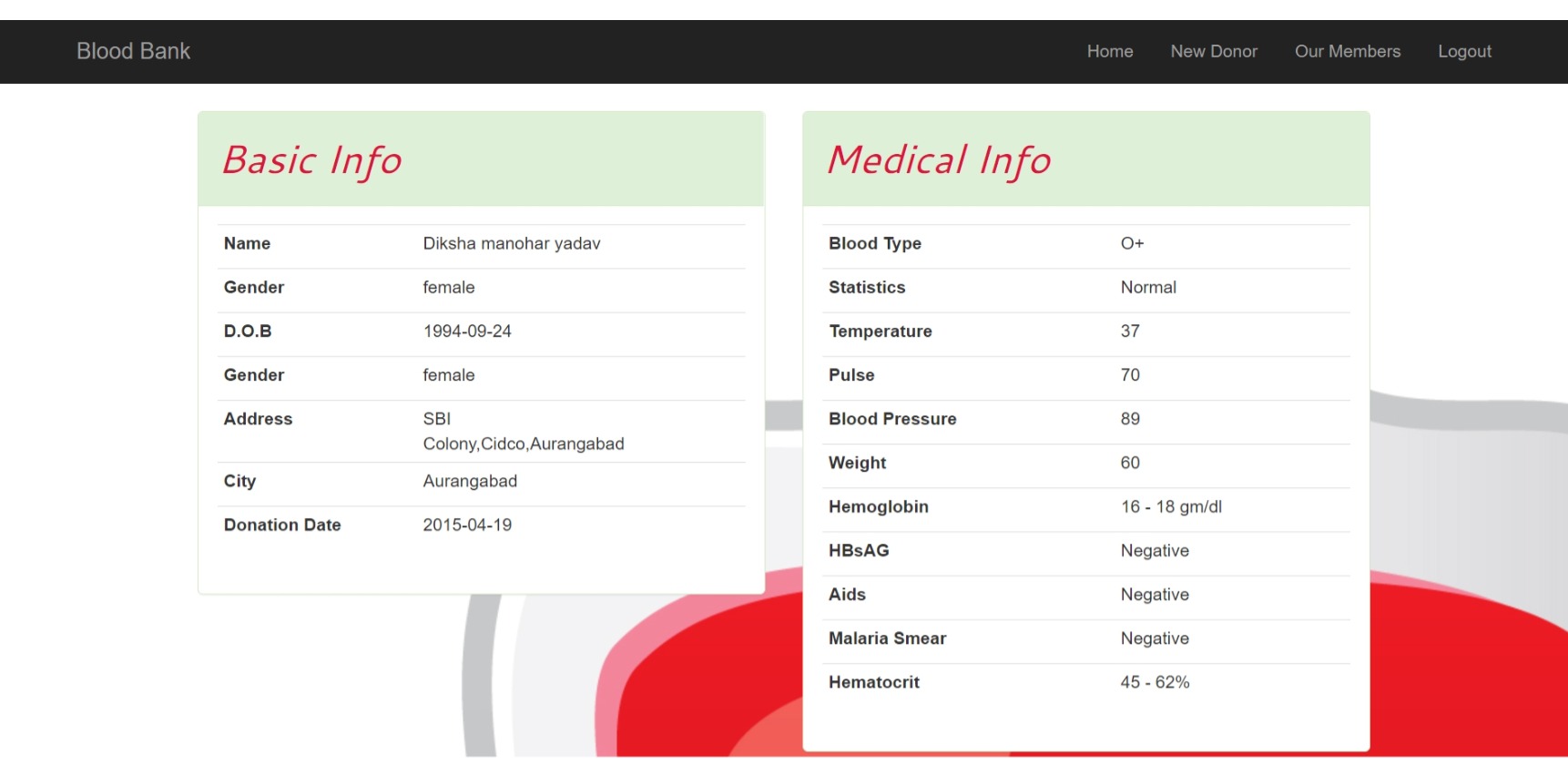
Add New Donor-Input Page

Our Memebers-Output Page

 Admin Login Page-Input Page

 Add Employee-Input Page

Employee List-Output Page

 Donor Information-Output Page

UML Diagram:Use case Diagram

*Use case Diagrams* represent the functionality of the system from a user’s point of view. Use cases are used during requirements elicitation and analysis to represent the functionality of the system. Use cases focus on the behavior of the system from external point of view.

*Actors*are external entities that interact with the system. Examples of actors include users like administrator, user,employee …etc., or another system like central database.

Admin Interface

Home Page



Login Page



Add Employee



Edit Employee Details



Delete Employee details



**Admin**

**Employee Interface**

Login Page

Login Page

Search donor by blood group and city

Add new Donor details

Search Blood Donor

See Member Details

See donor details

Log out

**Employee**

**User interface**

**User**

Check availability

Join Us

View Members

About us

Home Page

Testing

|  |  |  |
| --- | --- | --- |
| **Test case name** | **Test Done** | **Output of the test** |
| 1)Verify Employee(login\_id, password) | The login and password cannot be null. | If login and password matches the employee logs into his account. |
| 2)Verify admin(login id,password) | The login and password cannot be null. | If login and password matches the employee logs into his account. |
| 3)Registration | Most of the fields are compulsory. Email is validated for correct syntax. Mobile number is validated for the numbers. | Customer Registration is successful and now he can log into his account. |
|  |  |  |

**LIMITATIONS:**

* The size of the database increases day-by-day, increasing the load on the database back up and data maintenance activity.
* Training for simple computer operations is necessary for the users working on the system

**CONCLUSION:**

* It’s a web-enabled project.
* This project offers user to enter the data through simple and interactive forms. This is very helpful for the client to enter the desired information through so much simplicity.
* The user is mainly more concerned about the validity of the data, whatever he is entering. There are checks on every stages of any new creation, data entry or updating so that the user cannot enter the invalid data, which can create problems at later date.
* User is provided the option of monitoring the records he entered earlier. He can see the desired records with the variety of options provided by him.
* From every part of the project the user is provided with the links through framing so that he can go from one option of the project to other as per the requirement. This is bound to be simple and very friendly as per the user is concerned.
* Data storage and retrieval will become faster and easier to maintain because data is stored in a single database.
* Decision making process would be greatly enhanced because of faster processing of information since data collection from information available on computer takes much less time then manual system.