Blood Donation Management CSCE 5350 Fundamentals of Database Systems

PROJECT REPORT

Vamshi Kalyan Yerramilli - 11655488
Abhiram Yashwanth Pusarla - 11712109
Amani Peddolla - 11743144
Harshith Kumar Pappula - 11646762
Vijayalakshmi Talla - 11711703
Aamani Chamarthi - 11703676

About System

The purpose of the **Blood Donation Management System** is to make blood donation easier by handling donor data, setting up appointments, and ensuring successful blood donation events. It makes it easier for hospitals, blood banks, and organizations to track appointments, handle donor data, and provide a seamless blood donation experience. The system makes sure that giving blood is safe, well-organized, and convenient for people who need it.
Among the system's primary characteristics are:

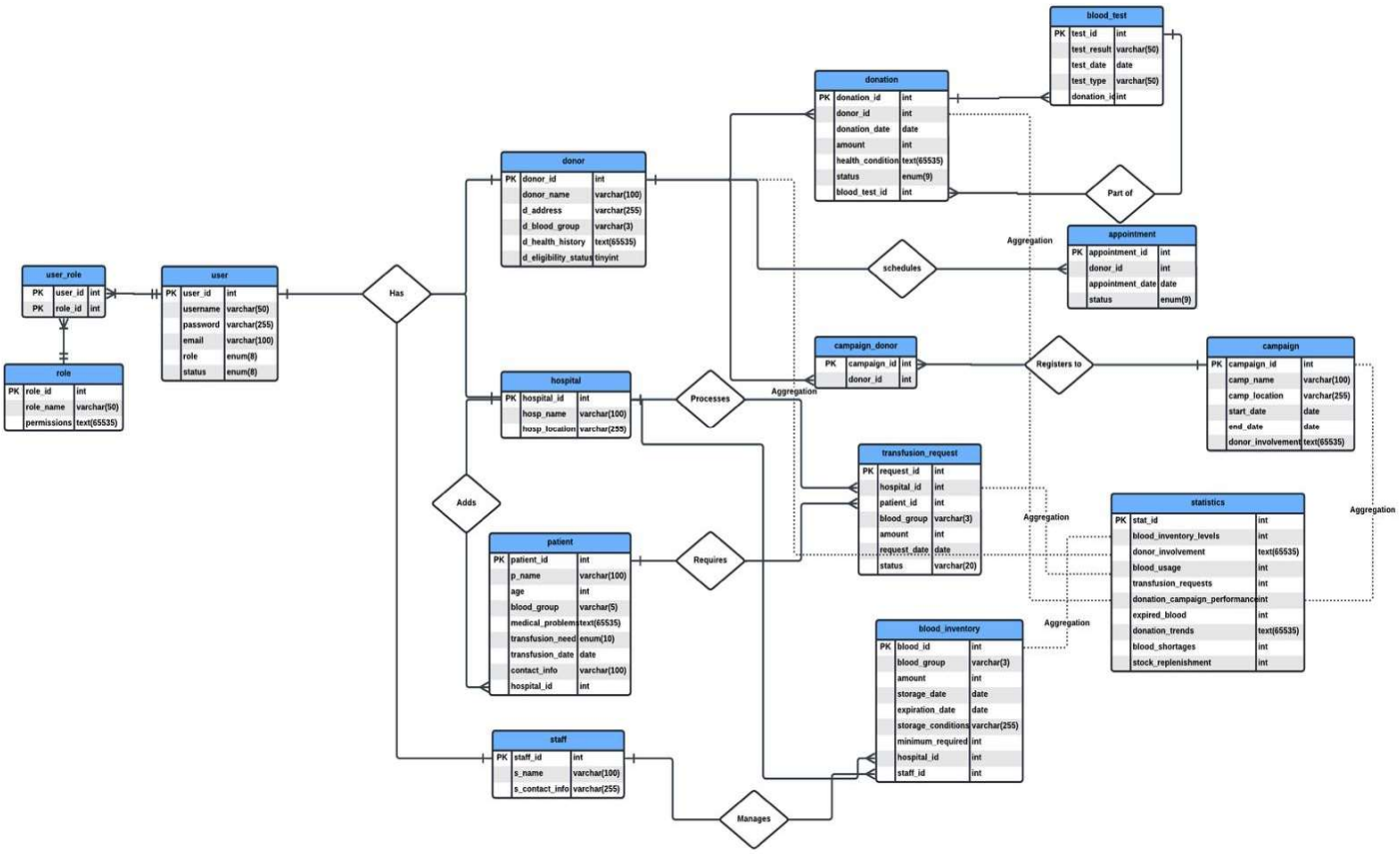
- **Donor Management and Registration:** Donors can register and modify their personal information, including their name, donor ID, contact details, and donation history.
- **Appointment Scheduling:** Donors can make appointments to donate blood, guaranteeing a prompt and organized donation procedure.
- **Appointment Tracking:** The system keeps track of each appointment's progress and makes sure that staff and donors are informed on time about appointment dates and availability.
- **Security of Donor Information:** All donor data, including medical histories and donation records, is safely kept and only authorized staff can access it.
- **Blood Donation Events:** By monitoring donor involvement, available blood types, and the volume of blood donated, the system enables organizations to plan and oversee blood donation events.
- **Feedback and Confirmation:** Contributors can offer comments on their experience making a gift as well as obtain confirmation of their contribution.

Assumptions

- **Donor Accounts:** To schedule appointments and monitor donation history, each donor must have their own account.
- **Appointment Scheduling:** To prevent donation dates from overlapping, donors are only permitted to make one appointment at a time.

- **Blood Donation Events:** The system assumes that donors will be informed of forthcoming events, or donation drives and that blood donation events will be planned.
- **Appointment Availability:** The system makes sure that no appointments are overbooked, however donor appointments are contingent upon availability.
- **Donor Consent:** To guarantee that the procedure complies with health and safety laws, donors must give their consent for blood donation at the time of appointment scheduling.
- **Appointment Confirmation:** Donors receive confirmation and reminder reminders regarding their appointments via the system.
- **Blood Type Tracking:** The system keeps track of the blood types that are donated and makes sure the blood bank has enough supplies on hand in case of emergencies.
- **Medical Eligibility:** Based on predetermined health criteria, only donors who are medically eligible may make appointments.

Entity Relationship Diagram



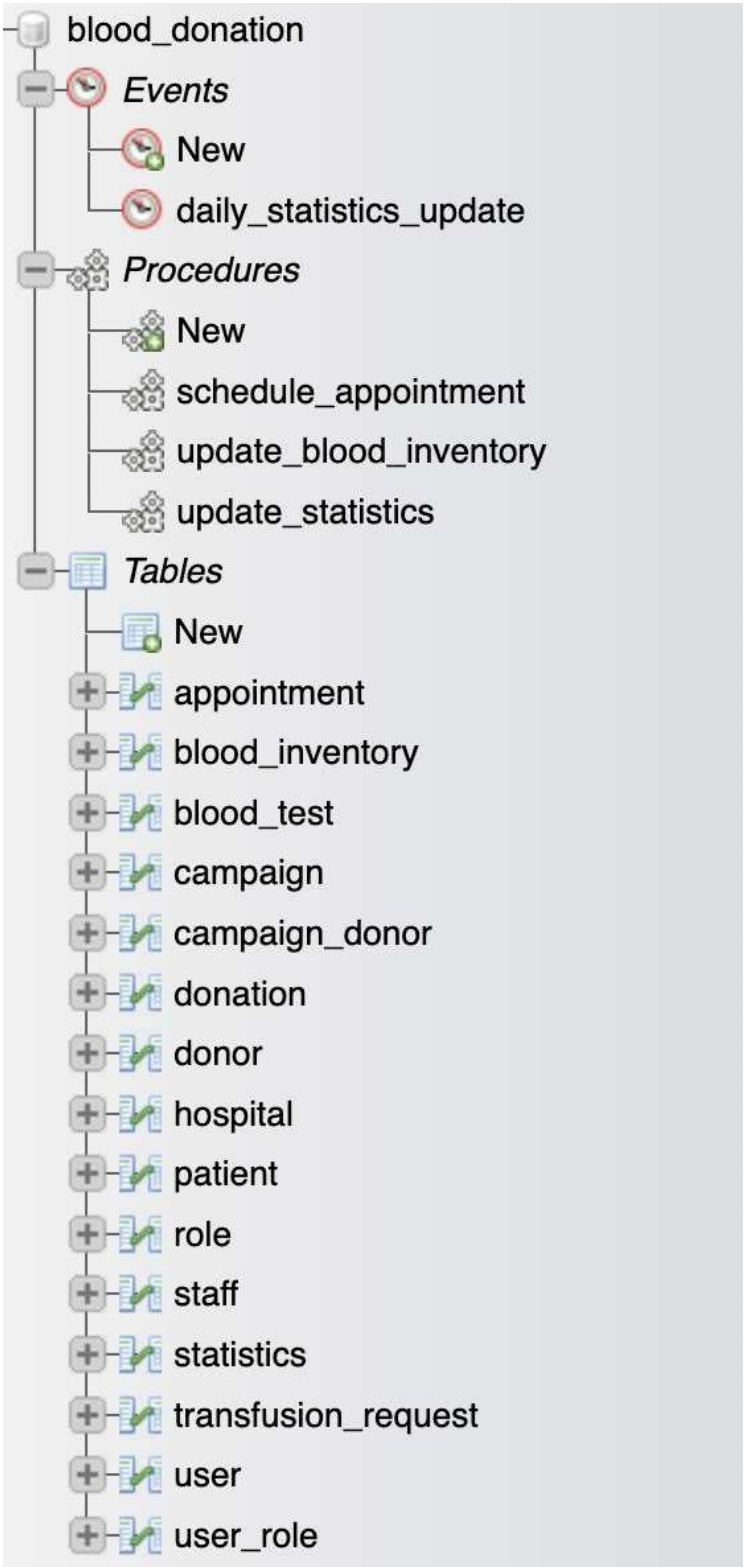
Relational Schema

The Blood Donation Management System has a total of 15 entities. They are listed below.

- appointment
- blood_inventory
- blood_test
- campaign
- campaign_donor
- donation
- donor
- hospital
- patient
- role
- staff
- statistics
- transfusion_request
- user
- user_role

also, we have few stored procedures and one event to aggregate all the statistics on daily basis, below ones are listed here.

- schedule_appointment
- update_blood_inventory
- update_statistics
- daily_statistics_update


blood_donation
Events
New
daily_statistics_update
Procedures
New
schedule_appointment
update_blood_inventory
update_statistics
Tables
New
appointment
blood_inventory
blood_test
campaign
campaign_donor
donation
donor
hospital
patient
role
staff
statistics
transfusion_request
user
user_role

appointment table:

	appointment_id	donor_id	appointment_date	status
Edit Copy Delete	1	9	2024-11-13	Cancelled
Edit Copy Delete	2	9	2024-11-15	Cancelled
Edit Copy Delete	3	9	2024-11-20	Cancelled
Edit Copy Delete	4	20	2024-11-13	Cancelled
Edit Copy Delete	5	9	2024-11-13	Cancelled
Edit Copy Delete	6	9	2024-11-13	Cancelled
Edit Copy Delete	7	9	2024-12-13	Scheduled

```
CREATE TABLE `appointment` (
`appointment_id` int(11) NOT NULL,
`donor_id` int(11) DEFAULT NULL,
`appointment_date` date NOT NULL,
`status` enum('Scheduled','Completed','Cancelled') NOT NULL
)
```

blood_Inventory table:

	blood_id	blood_group	amount	storage_date	expiration_date	storage_conditions	minimum_required
Edit Copy Delete	1	A+	20	2024-11-11	2024-11-14	*NULL*	5
Edit Copy Delete	2	O+	10	2024-12-02	2024-12-28	*NULL*	5

```
CREATE TABLE `blood_inventory` (
  `blood_id` int(11) NOT NULL,
  `blood_group` varchar(3) NOT NULL,
  `amount` int(11) DEFAULT NULL,
  `storage_date` date NOT NULL,
  `expiration_date` date NOT NULL,
  `storage_conditions` varchar(255) DEFAULT NULL,
  `minimum_required` int(11) DEFAULT '5'
)
```

blood_test table:

	test_id	test_result	test_date	test_type	donation_id
Edit Copy Delete	1	Pass	2024-11-12	*NULL*	2
Edit Copy Delete	2	Pass	2024-12-01	*NULL*	3

```
CREATE TABLE `blood_test` (
  `test_id` int(11) NOT NULL,
  `test_result` varchar(50) DEFAULT NULL,
  `test_date` date DEFAULT NULL,
  `test_type` varchar(50) DEFAULT NULL,
  `donation_id` int(11) NOT NULL
)
```

campaign table:

	campaign_id	camp_name	camp_location	start_date	end_date	donor_involvement
Edit Copy Delete	1	Christmas Blood Donation	Denton	2024-11-12	2024-11-15	*NULL*
Edit Copy Delete	2	New Year Campaign	Dallas	2024-12-02	2024-12-28	*NULL*

```
CREATE TABLE `campaign` (
  `campaign_id` int(11) NOT NULL,
  `camp_name` varchar(100) NOT NULL,
  `camp_location` varchar(255) DEFAULT NULL,
  `start_date` date NOT NULL,
  `end_date` date NOT NULL,
  `donor_involvement` text
)
```

campaign_donor table:

	campaign_id	donor_id
Edit Copy Delete	1	9

```
CREATE TABLE `campaign_donor` (
  `campaign_id` int(11) NOT NULL,
  `donor_id` int(11) NOT NULL
)
```

donation table:

	donation_id	donor_id	donation_date	amount	health_condition	status	blood_test_id
Edit Copy Delete	1	9	2024-11-12	NULL	NA	Completed	NULL
Edit Copy Delete	2	9	2024-11-12	NULL	NA	Completed	NULL
Edit Copy Delete	3	20	2024-12-01	NULL	NA	Completed	NULL

```
CREATE TABLE `donation` (
  `donation_id` int(11) NOT NULL,
  `donor_id` int(11) DEFAULT NULL,
  `donation_date` date NOT NULL,
  `amount` int(11) DEFAULT NULL,
  `health_condition` text,
  `status` enum('Pending','Completed','Failed') DEFAULT 'Pending',
  `blood_test_id` int(11) DEFAULT NULL
)
```

donor table:

	donor_id	donor_name	d_address	d_blood_group	d_health_history	d_eligibility_status
Edit Copy Delete	9	tst_donor	1101 avenue C	O+	Good Standing.	1
Edit Copy Delete	15	tst_donor1	1101 Ave C	A+	NA	1
Edit Copy Delete	18	tst2_donor	Union Circle	O+	NA	1
Edit Copy Delete	20	Harshith	1101 Ave C	O+	NA	1

```
CREATE TABLE `donor` (
  `donor_id` int(11) NOT NULL,
  `donor_name` varchar(100) NOT NULL,
  `d_address` varchar(255) DEFAULT NULL,
  `d_blood_group` varchar(3) NOT NULL,
  `d_health_history` text,
```

```
  `d_eligibility_status` tinyint(1) NOT NULL
)
```

hospital table:

	hospital_id	hosp_name	hosp_location
Edit Copy Delete	11	Texas Health	Denton
Edit Copy Delete	16	United health	Texas

```
CREATE TABLE `hospital` (
  `hospital_id` int(11) NOT NULL,
  `hosp_name` varchar(100) NOT NULL,
  `hosp_location` varchar(255) DEFAULT NULL
)
```

patient table:

	patient_id	p_name	age	blood_group	medical_problems	transfusion_need	transfusion_date	contact_info	hospital_id
Edit Copy Delete	1	Krishna	NULL	O+ve	NA	Urgent	NULL	9803311133	NULL
Edit Copy Delete	2	Harshith	NULL	A+ve	Hepatits	Urgent	NULL	9803311133	NULL
Edit Copy Delete	3	rohith	28	O+	NA	Non-Urgent	NULL	1234567890	16
Edit Copy Delete	4	Vamshi	27	B+	na	Urgent	NULL	2345678910	11
Edit Copy Delete	5	test	43	B-	na	Urgent	NULL	8766655559	11

```
CREATE TABLE `patient` (
  `patient_id` int(11) NOT NULL,
  `p_name` varchar(100) NOT NULL,
  `age` int(11) DEFAULT NULL,
  `blood_group` varchar(5) DEFAULT NULL,
  `medical_problems` text,
  `transfusion_need` enum('Urgent','Non-Urgent') NOT NULL,
  `transfusion_date` date DEFAULT NULL,
  `contact_info` varchar(100) DEFAULT NULL,
  `hospital_id` int(11) DEFAULT NULL
)
```

role table:

	role_id	role_name	permissions
Edit Copy Delete	1	Donor	Can register as a donor and donate blood
Edit Copy Delete	2	Hospital	Can request blood and manage hospital data
Edit Copy Delete	3	Staff	Can manage appointments and inventory
Edit Copy Delete	4	Admin	Full access to manage users and system settings

```
CREATE TABLE `role` (
  `role_id` int(11) NOT NULL,
  `role_name` varchar(50) NOT NULL,
  `permissions` text
)
```

staff table:

	staff_id	s_name	s_contact_info
Edit Copy Delete	10	tst_staff	9989279124
Edit Copy Delete	14	tst_staff1	21321313322
Edit Copy Delete	17	tst_staff2	test

```
CREATE TABLE `staff` (
  `staff_id` int(11) NOT NULL,
  `s_name` varchar(100) NOT NULL,
  `s_contact_info` varchar(255) DEFAULT NULL
)
```

statistics table:

	stat_id	blood_inventory_levels	donor_involvement	blood_usage	transfusion_requests	donation_campaign_performance	expired_blood
Edit Copy Delete	1	20	3	NULL	6	1	0

```
CREATE TABLE `statistics` (
  `stat_id` int(11) NOT NULL,
  `blood_inventory_levels` int(11) DEFAULT NULL,
  `donor_involvement` text,
  `blood_usage` int(11) DEFAULT NULL,
  `transfusion_requests` int(11) DEFAULT NULL,
  `donation_campaign_performance` int(11) DEFAULT NULL,
  `expired_blood` int(11) DEFAULT NULL,
  `donation_trends` text,
  `blood_shortages` int(11) DEFAULT NULL,
  `stock_replenishment` int(11) DEFAULT NULL
)
```

transfusion_request table:

	request_id	hospital_id	patient_id	blood_group	amount	request_date	status
Edit Copy Delete	1	11	NULL	A+	1	2024-11-11	Pending
Edit Copy Delete	2	11	NULL	B+	2	2024-11-11	Pending
Edit Copy Delete	3	11	NULL	O+	22	2024-11-11	Pending
Edit Copy Delete	4	11	NULL	O+	20	2024-11-11	Pending
Edit Copy Delete	5	11	NULL	O+	20	2024-11-11	Pending
Edit Copy Delete	6	11	NULL	B+	11	2024-11-11	Pending

```
CREATE TABLE `transfusion_request` (
  `request_id` int(11) NOT NULL,
  `hospital_id` int(11) DEFAULT NULL,
  `patient_id` int(11) DEFAULT NULL,
  `blood_group` varchar(3) NOT NULL,
  `amount` int(11) DEFAULT NULL,
  `request_date` date NOT NULL,
  `status` varchar(20) NOT NULL
)
```

user table:

	user_id	username	password	email	role	status
Edit Copy Delete	9	tst_donor	$2y$10$S5t3EXPcYZn79jaOHXQHGuC9XRi6.cWWmub1hiETJoP...	donor@test.com	Donor	Active
Edit Copy Delete	10	tst_staff	$2y$10$vt0G5lgoVmVrzrz7Kgj8.uYDKp8i4yggUW.TQ4ZfbKp...	staff@test.com	Staff	Active
Edit Copy Delete	11	tst_hosp	$2y$10$f2eG/FADGhKX7O8v6kHKt.K3DjXnANw/Beld9ykAql1...	th@hospital.com	Hospital	Active
Edit Copy Delete	14	tst_staff1	$2y$10$jmZladiljbUaXwmS/pTlyuUlm00PMK4p7CVtVte.122...	satff1@test.com	Staff	Active
Edit Copy Delete	15	tst_donor1	$2y$10$u1FDdfHXBYFNMRLTPpzeNeecd.kCKpYS7WglsyCSx3h...	donor1@test.com	Donor	Active
Edit Copy Delete	16	tst_hosp1	$2y$10$G4q6zvAcq8Zw/USrRU8z7.xMVX4pPSckpqzPqPA2Wee...	hosp1@test.com	Hospital	Active
Edit Copy Delete	17	tst_staff2	$2y$10$jj1/bBM9ug0WCskqlQyCB.n7ksibDiAdiW4hpWuDuze...	staff2@test.com	Staff	Active
Edit Copy Delete	18	tst2_donor	$2y$10$MKpMjvirXMO3Ex6DD5SfVudtsrduZlrg3gpBoD5axDQ...	donor2@test.com	Donor	Active
Edit Copy Delete	19	admin	$2y$10$ufgBn.pmoGD5nyQReXTMuODLTfHWb3dKD5Oi1JblMDr...	admin@test.com	Admin	Active
Edit Copy Delete	20	Harshith	$2y$10$fE2YUqBCFvOEygPD0JaOVuWk6NSV1/w1e9ct22mB6wL...	harshith@gmail.com	Donor	Active

```
CREATE TABLE `user` (
  `user_id` int(11) NOT NULL,
  `username` varchar(50) NOT NULL,
  `password` varchar(255) NOT NULL,
  `email` varchar(100) NOT NULL,
  `role` enum('Donor','Staff','Hospital','Admin') NOT NULL,
  `status` enum('Active','Inactive') NOT NULL
)
```

user_role table:

	user_id	role_id
Edit Copy Delete	9	1
Edit Copy Delete	15	1
Edit Copy Delete	18	1
Edit Copy Delete	20	1
Edit Copy Delete	11	2
Edit Copy Delete	16	2
Edit Copy Delete	10	3
Edit Copy Delete	14	3
Edit Copy Delete	17	3
Edit Copy Delete	19	4

```
CREATE TABLE `user_role` (
  `user_id` int(11) NOT NULL,
  `role_id` int(11) NOT NULL )
```

Normalization:

Regarding the normalization process:

- **First Normal Form (1NF):** Every table in our relational schema includes a distinct primary key property that makes it possible to identify every record in a unique way.
- Since none of the data in our tables can be further subdivided, they are all atomic. As a result, our tables are already in their first standard format.

- All non-prime properties (those that are not a part of the primary key) are totally dependent on the primary key in the **Second Normal Form (2NF).**
- Every non-prime attribute now has a direct relationship to the primary key as we removed partial dependencies from our relations. Our tables are therefore already in second normal form.

- **Third Normal Form (3NF):** There are no non-prime qualities in our relations that rely on other non-prime attributes, which rely on the primary key.
- All non-prime properties are totally dependent on the primary key; transitive dependencies do not exist. Our relationship is thus already in its third regular state.

To sum up, our database design satisfies the 1NF, 2NF, and 3NF criteria, suggesting a high degree of normalization; no additional normalization is thought to be required.

Technologies:

- **Frontend**: PHP, HTML, CSS, JavaScript (for form handling and UI)
- **Backend**: PHP with MySQL for database interaction
- **Database**: MySQL for storing patient and transfusion-related data
- **Authentication**: PHP sessions with role-based access control (Admin, Hospital)
- **Security**: SQL injection prevention through data sanitization and validation

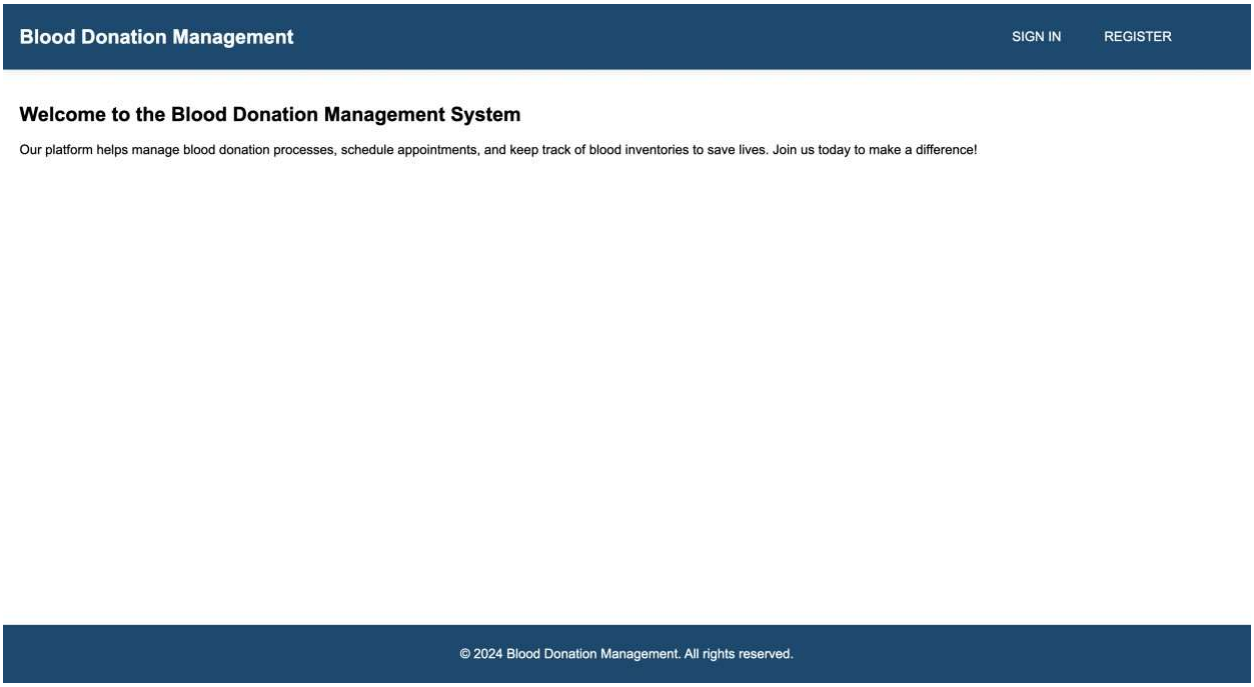
Use these steps to operate the blood donation management system:

- **Unzip the Project File:** Find your XAMPP directory and unzip the project file.
  Copy to "htdocs" Directory: Locate the "htdocs" folder inside the XAMPP directory. In this directory, paste the project folder that was extracted.
- **Launch Google Chrome:** Start the Google Chrome browser.
  Reach out to phpMyAdmin: Open your browser and navigate to "http://localhost/phpmyadmin".
- **Create Database:** Use the script provided to create the database and insert sample data as below
- **Import Database File:** In phpMyAdmin, select the "Import" tab. Select the supplied database file
- **Configure Database:** Follow the import procedure to set up the database.
- **Launch the Application:** In your browser, type http://localhost:8888/app/public/index.html to access the system.
- **Retrieve Login Information:** To log in and access the system, consult the phpMyAdmin user and admin tables.
- **Login and Use:** To access and utilize the Supply Chain Management System, enter the login credentials you were given.

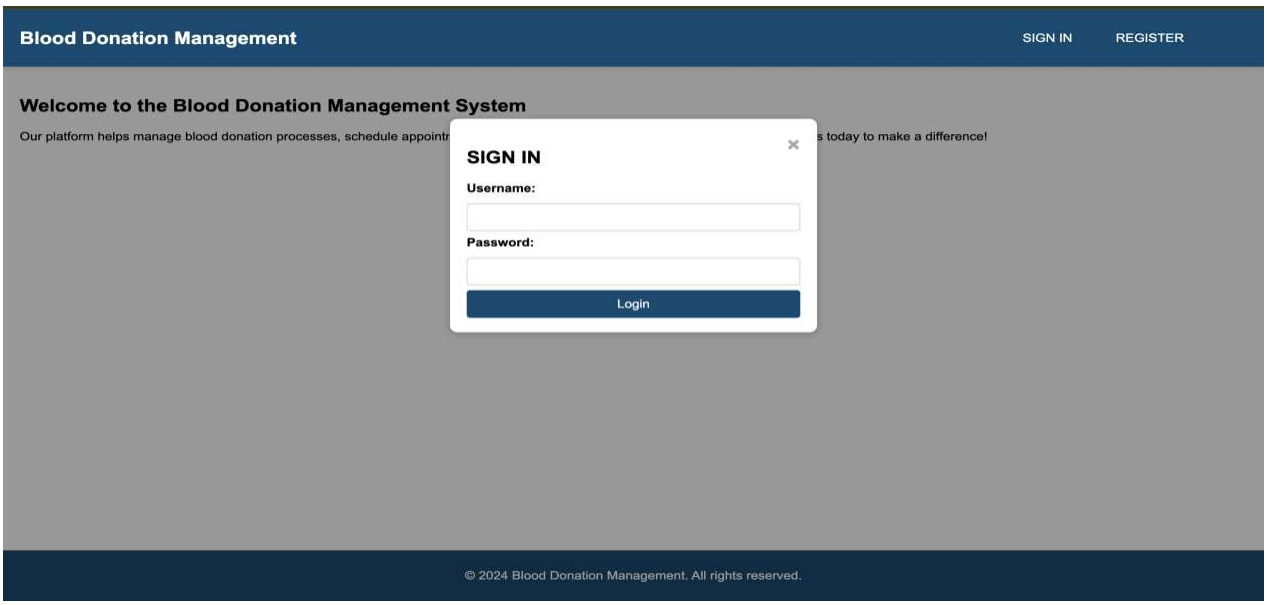
To guarantee a seamless setup and functioning of the blood donation management application, carefully follow these procedures.

Walkthrough of the Application:

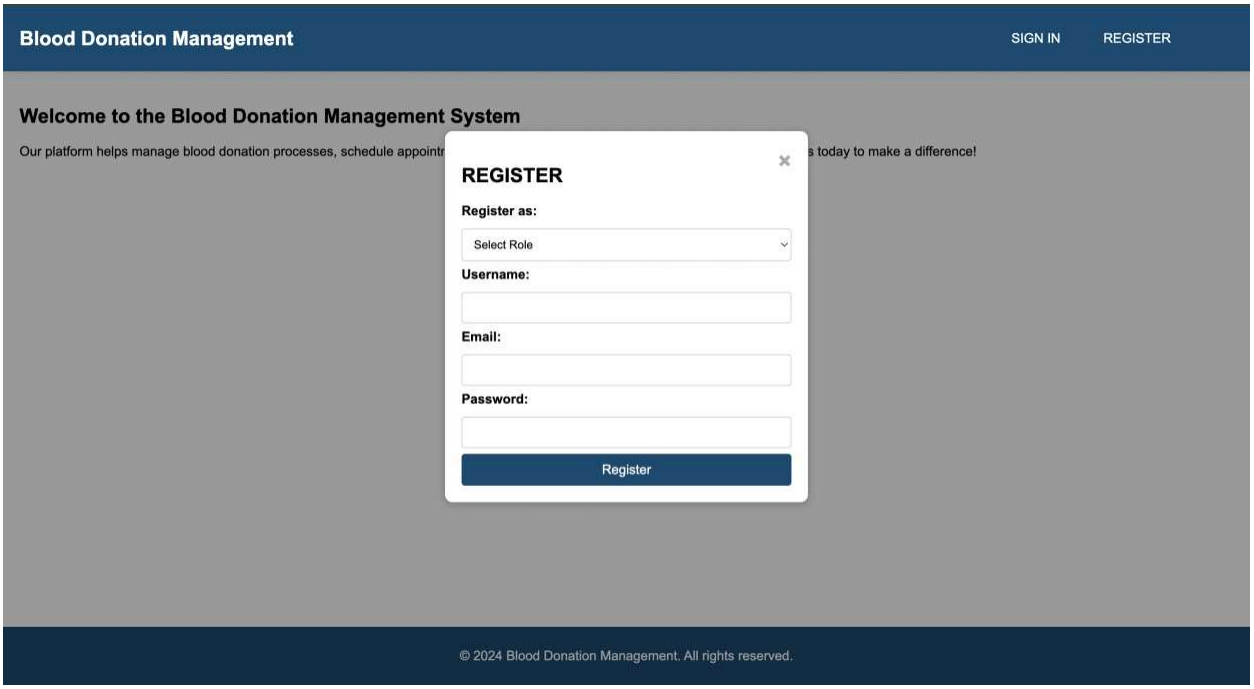
Below is the home page after you successfully launched the application URL.



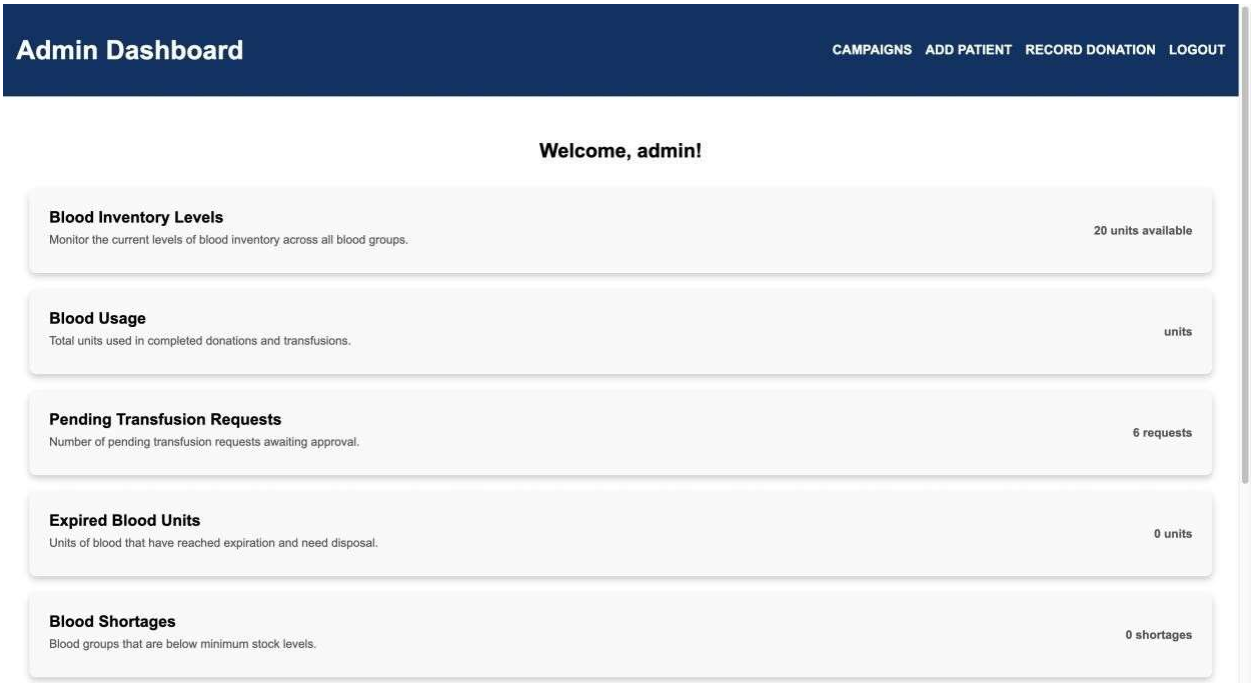
When user clicks on the sign in button, user will see the sign in popup on the home page as below:



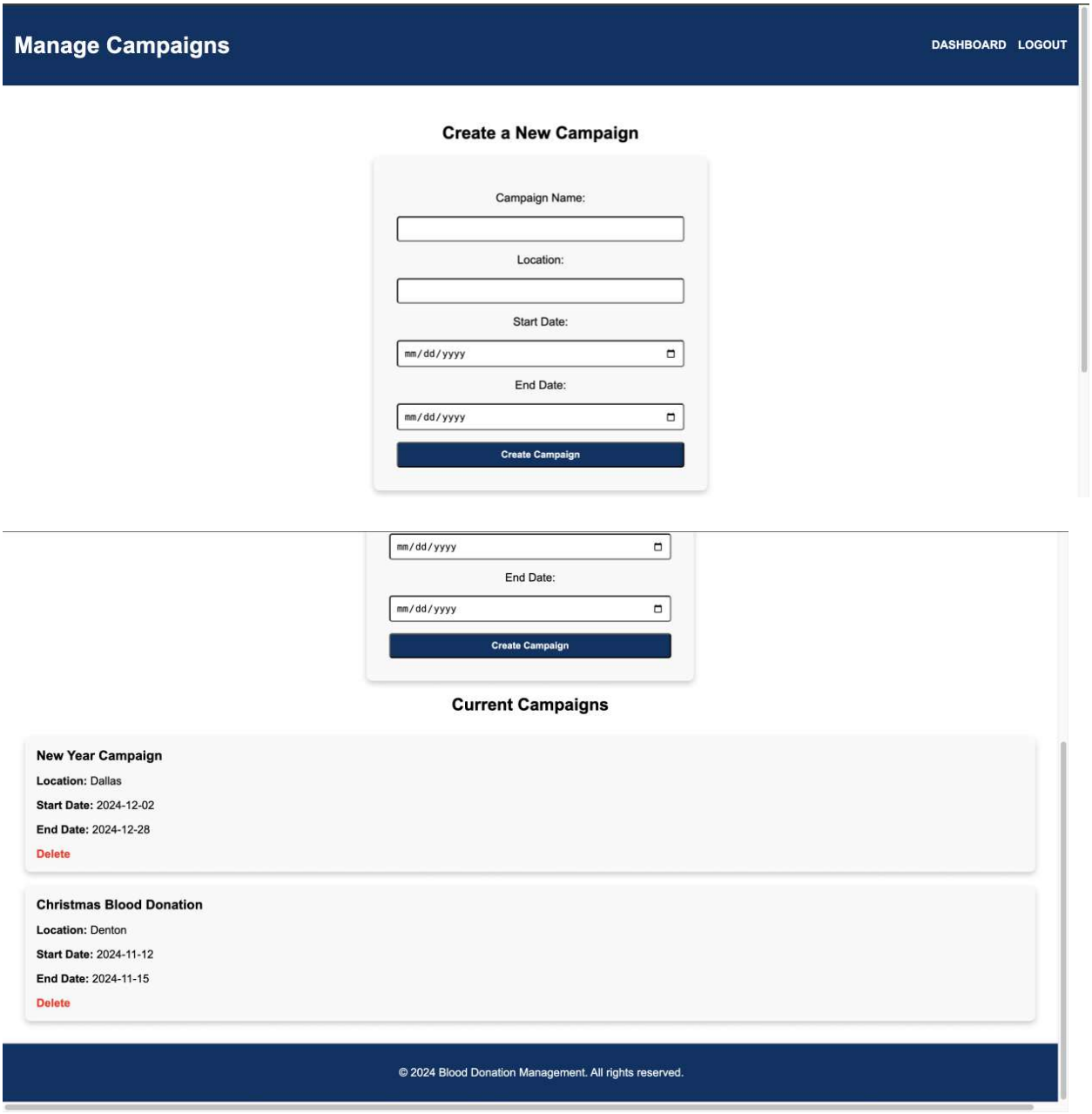
When user clicks on the register button, user will see the register popup on the home page as below:



User upon successful login with admin credentials, will be in dashboard page.

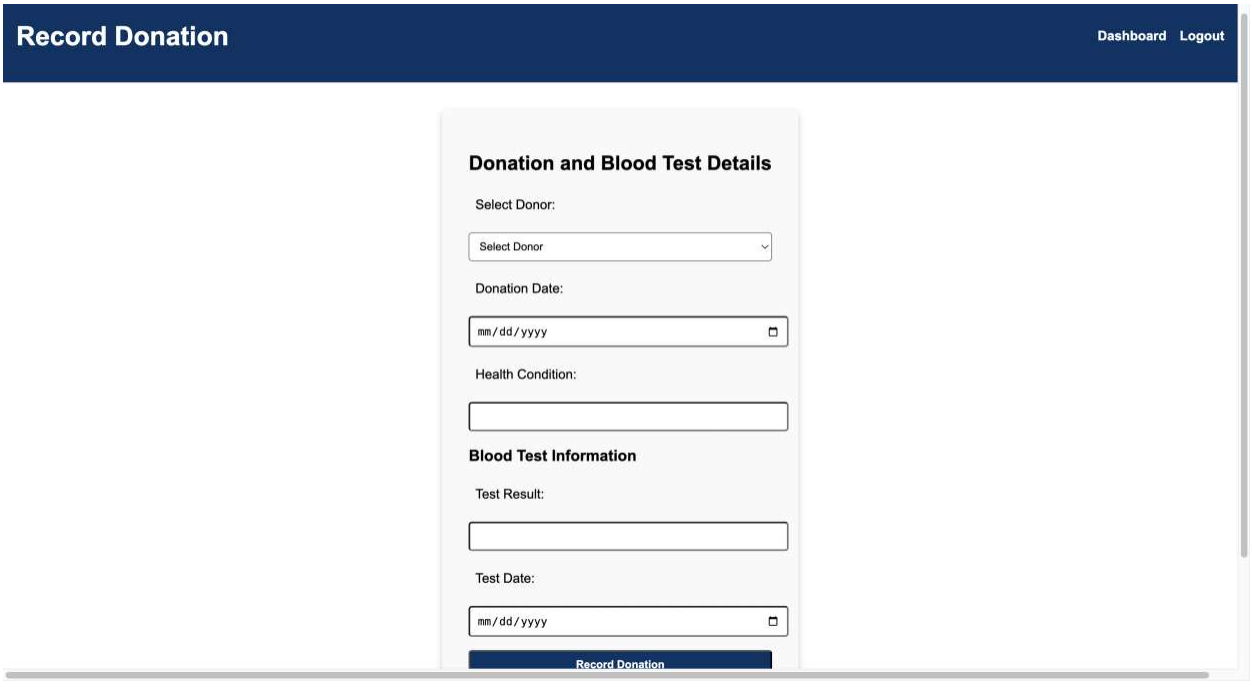


Below screen is the from where admin create a new campaign and view current campaigns.

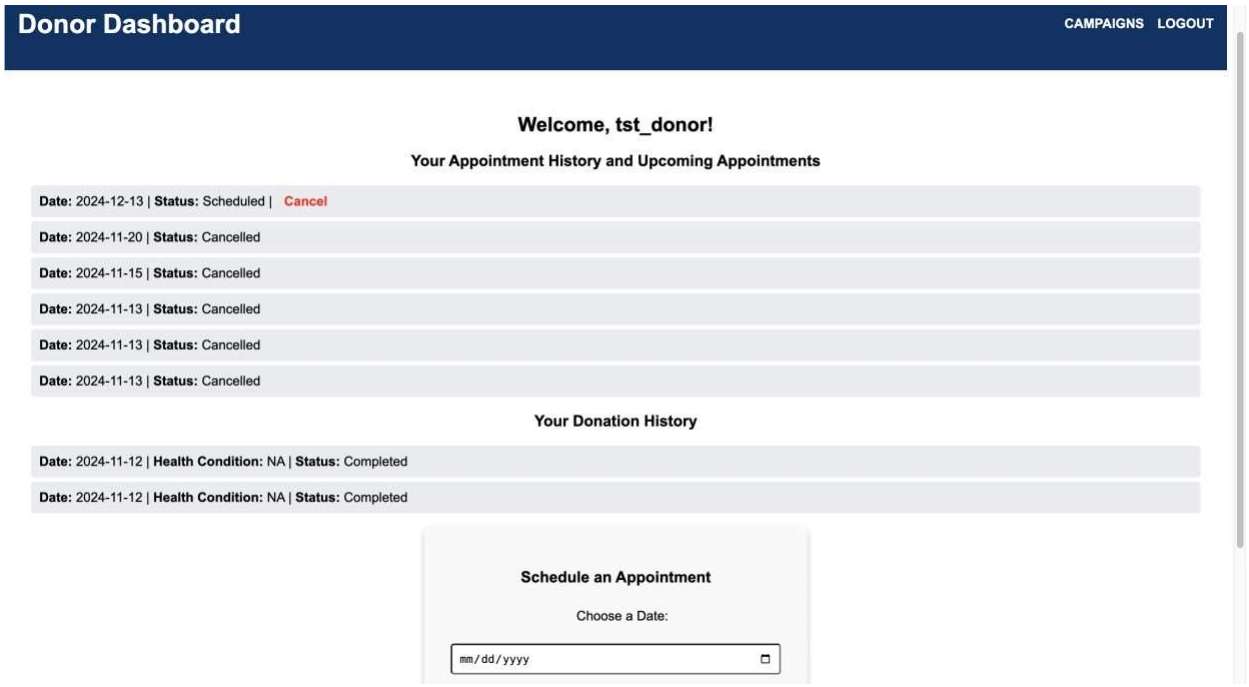


Below screen shows from where new patient will be added from admin side.

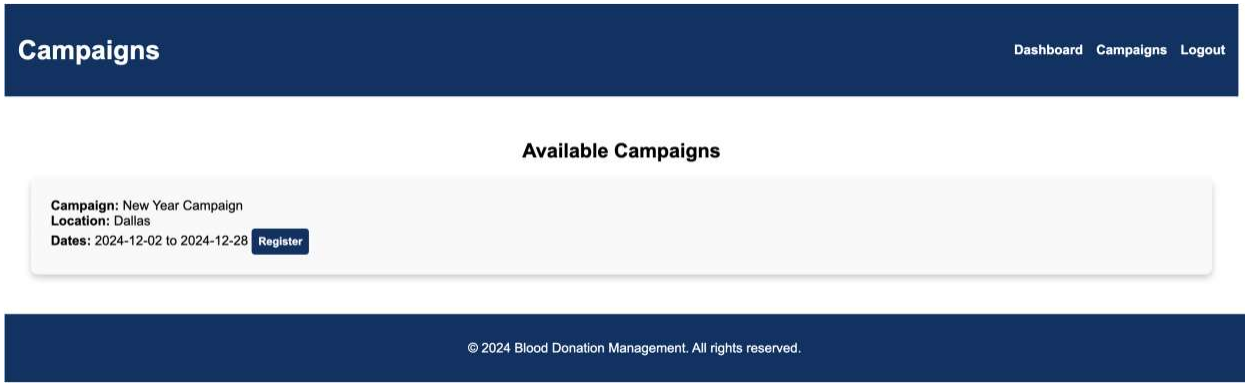
Below screen shows from where admin records the completed donation and the blood test results.



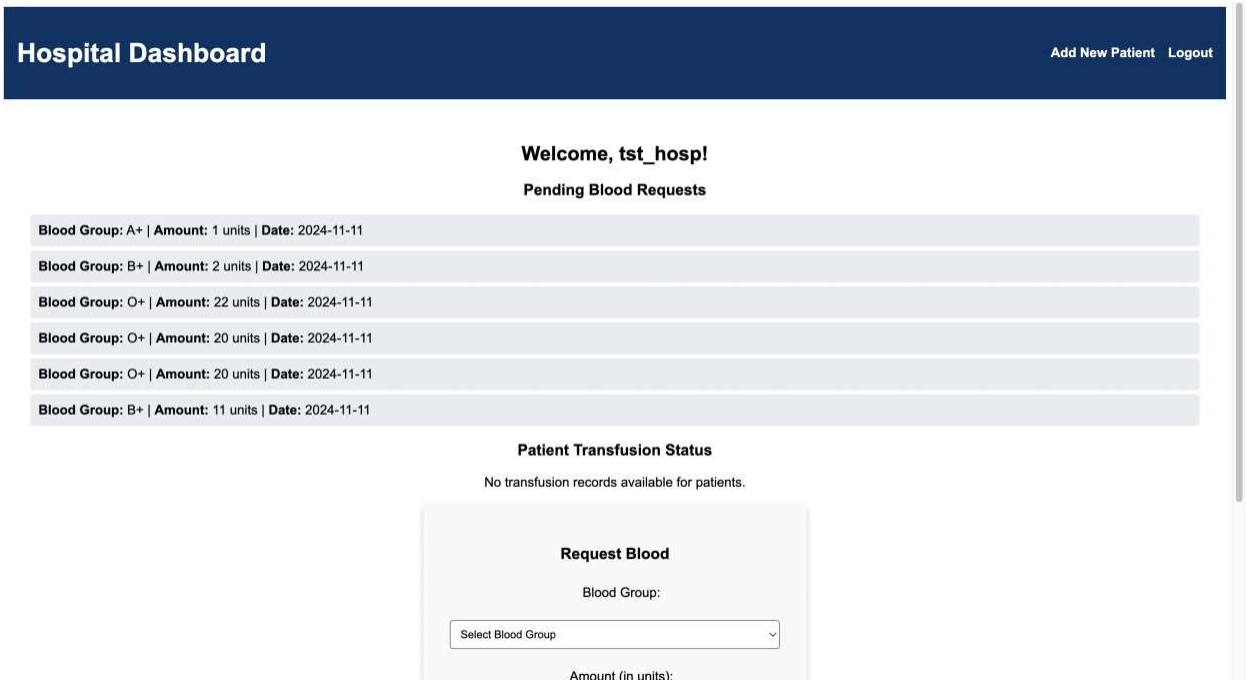
User upon successful login with donor credentials, will be in donor dashboard page and from where donor can book appointment and view history and upcoming appointments.



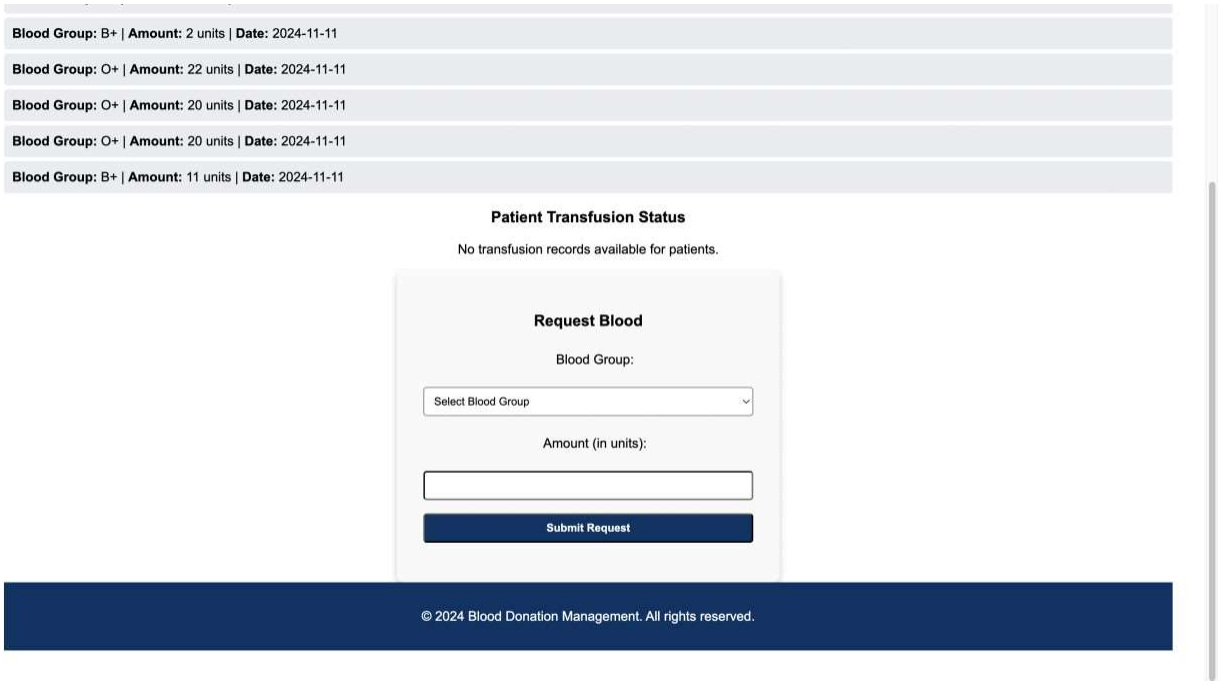
Below is the screen from where donor should be able to register for available campaigns



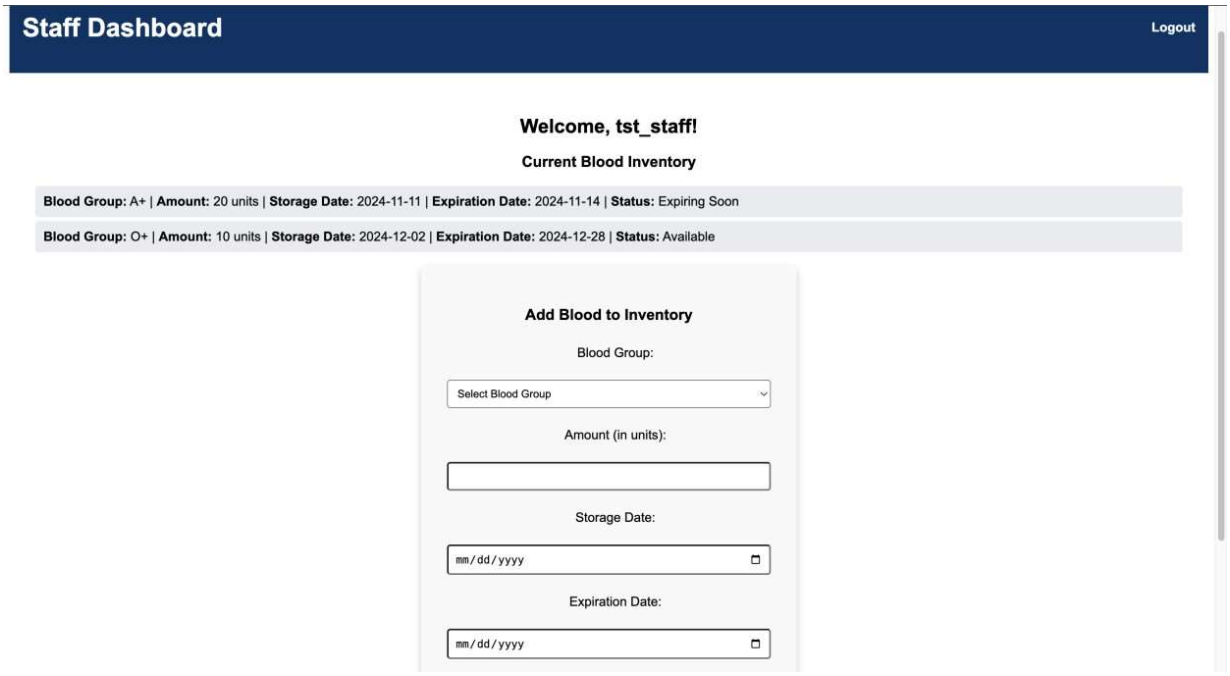
User upon successful login with hospital credentials, will be in hospital dashboard page from where hospital can view pending blood requests, can request blood.



Below screen depicts that hospital can process the transfusion request and also can add patients

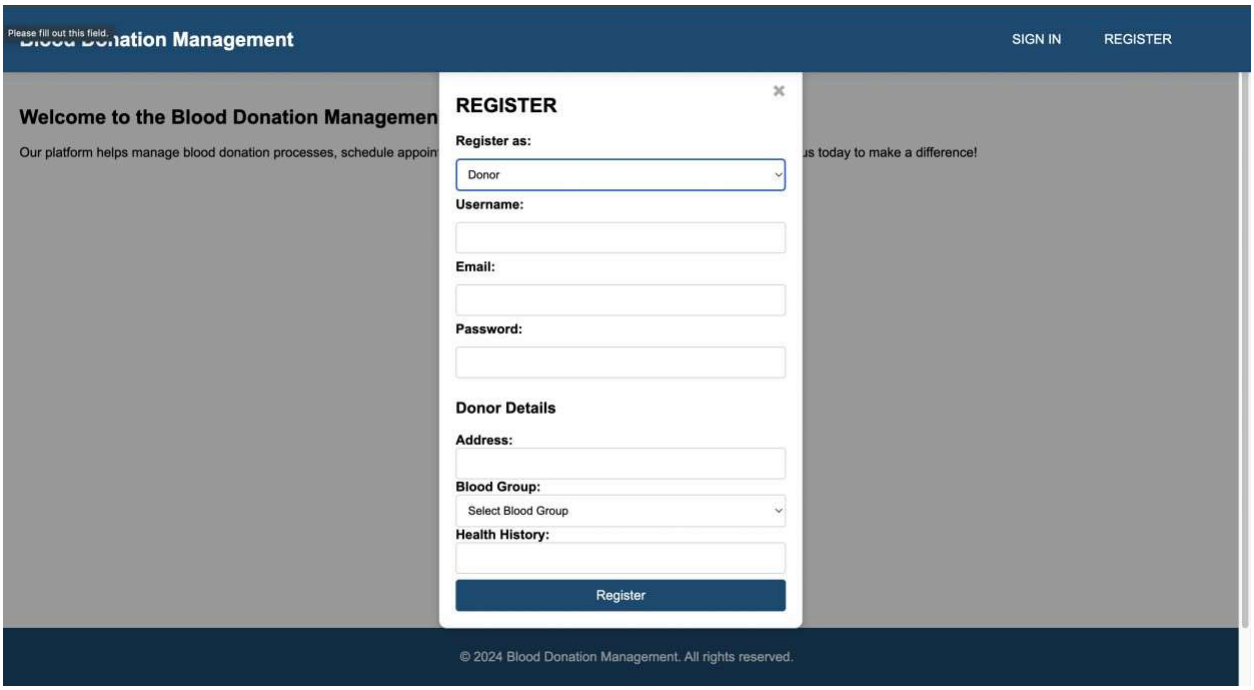


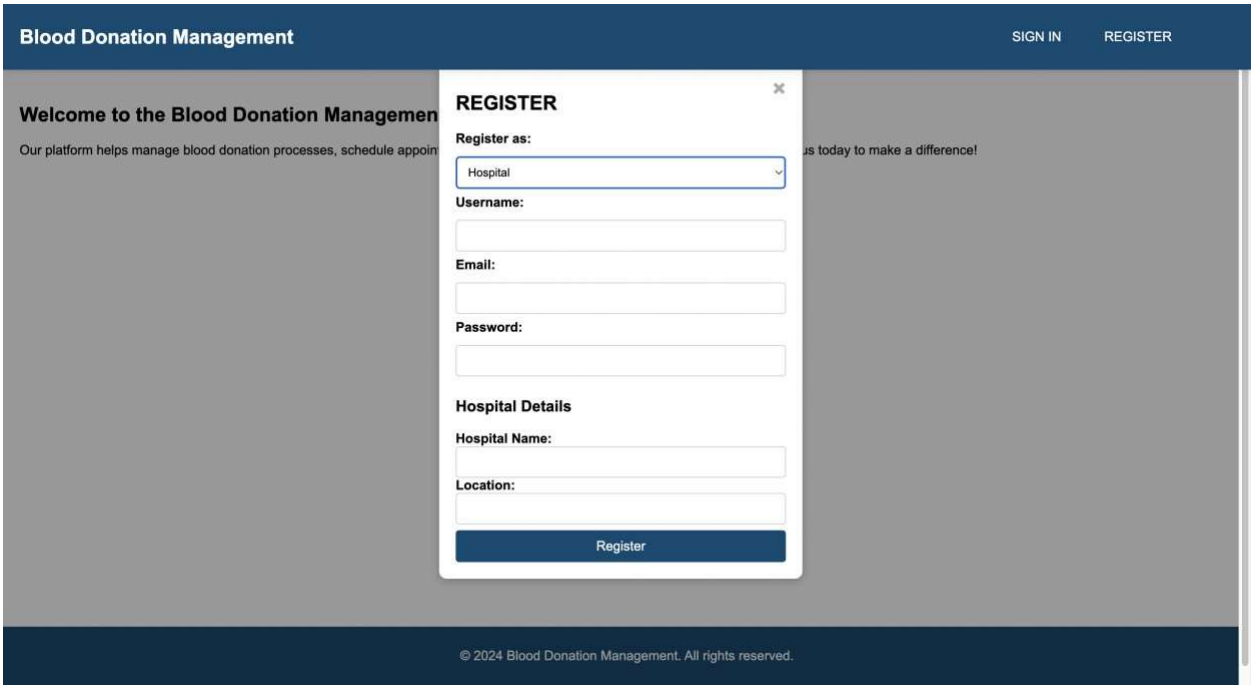
Here is the staff dashboard from where staff can manage the blood inventory details.

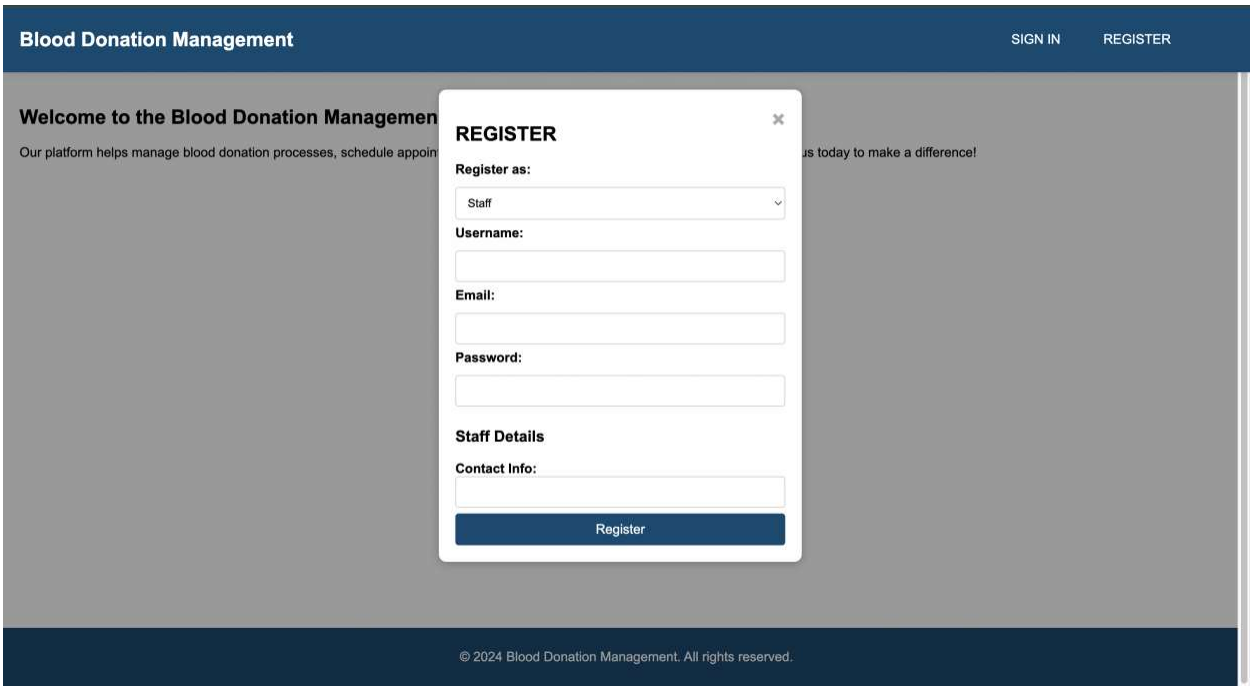


Functioning of the blood donation management system :

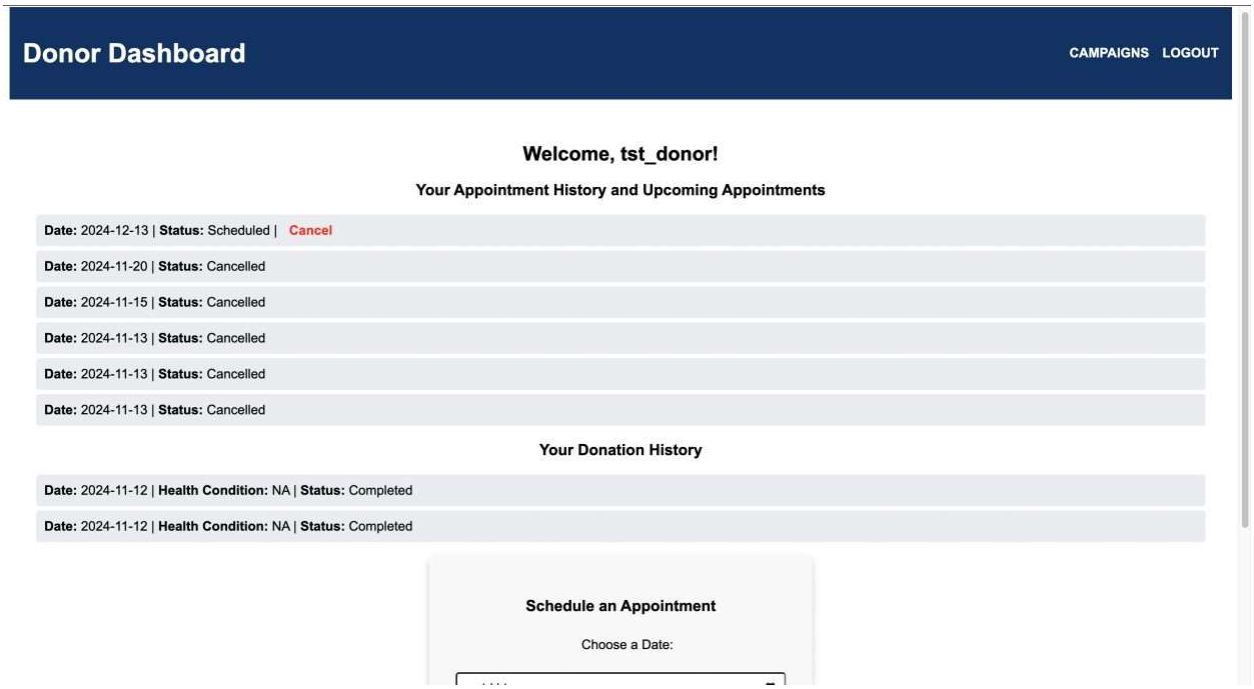
The register functionality is created in such a way that it should function more dynamically instead of creating three different register pages it is designed in a way to function for all the different roles available in the system such as donor, hospital and staff. Below are the screenshots for the same:



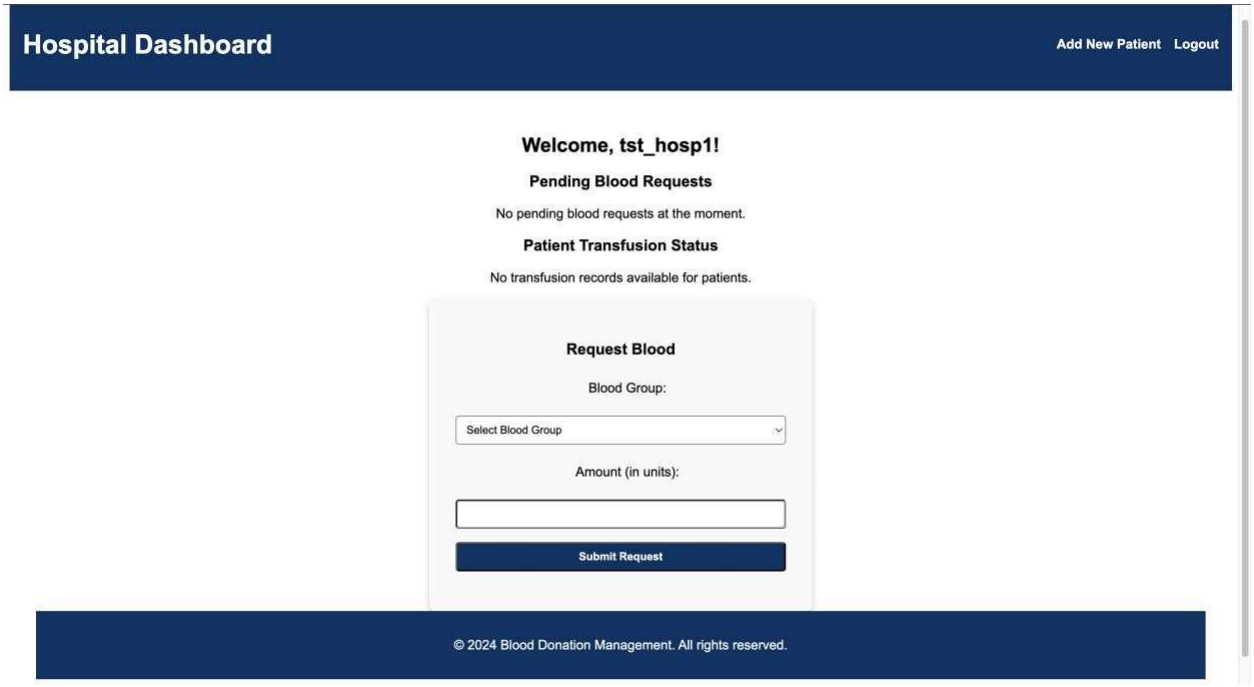




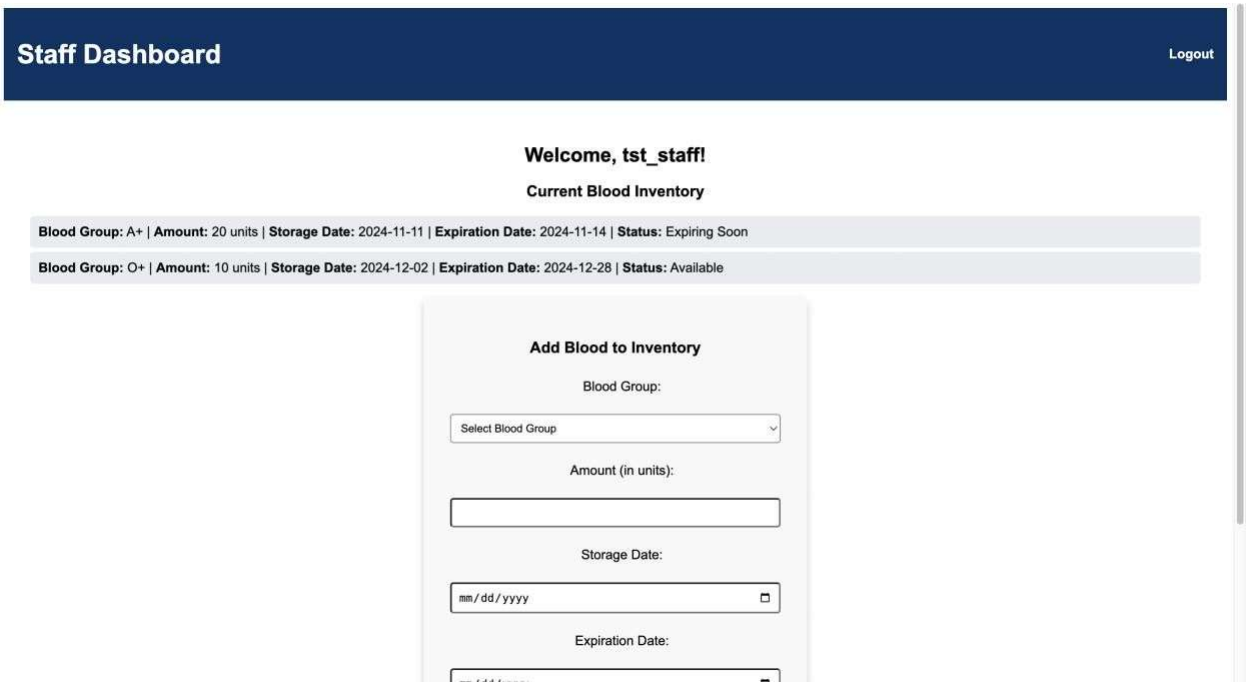
Upon successful logins, each role will have specially designed dashboards based on their roles, below is the donor dashboard with various options like schedule an appointment, register for a campaign and logout.



Below is the hospital dashboard with different options like add patient, request blood and view pending blood and transfusion requests.



Below is the staff dashboard from where staff can easily update the blood inventory details.



Challenges Faced

- **SQL Injection Prevention:** Making certain that every user input is appropriately cleaned to stop illegal access or corrupted data.
- **Data Validation**: Before enabling the data to be saved, make sure that all required fields (such as hospital ID and transfusion need) are filled out and legitimate.

Future Improvements

- **Mobile Compatibility:** Hospital administrators and employees can use mobile devices to access the system if it is made mobile-responsive.
- **Reporting:** Including tools for creating reports on hospital performance, patient data, and transfusion requirements.
- **Email Notifications:** Including email notifications to inform hospitals or patients of updates or needs related to transfusions.

Conclusion

The main difficulties in overseeing the blood donation and transfusion procedures have been effectively addressed by the development of the Blood Donation Management System. The system simplifies the process for administrators and hospitals by utilizing secure data entry, role-based access control, and effective patient management capabilities. It guarantees that vital patient data, including contact information, medical conditions, and transfusion requirements, are precisely recorded and readily available, improving responsiveness and decision-making throughout blood transfusion procedures. Sensitive patient information is protected by the system's security and dependability, which are guaranteed by the application of SQL injection prevention strategies, data validation, and error handling.

Even if the project has achieved its goals, it may still be made even more flexible and user-friendly for medical professionals with future additions like email notifications, mobile compatibility, and sophisticated reporting functions.

To sum up, the Blood Donation Management System is an invaluable resource for administrators and hospitals, guaranteeing the safe and effective administration of blood transfusions. The system might be extended to accommodate more sophisticated features and larger-scale operations with additional improvements, which would enhance patient care and operational effectiveness in healthcare environments.