# Smart Blood & Organ Donation CRM – Life Saving Support System

## Overview

The **Smart Blood & Organ Donation CRM** is a Salesforce-based solution designed to connect hospitals, NGOs, donors, and patients in critical situations.

It simplifies donor registration, automates emergency matching, and ensures life-saving blood or organ donations reach patients on time.

## Problem Statement

Patients in emergencies often struggle to find matching blood or organ donors quickly.

The existing process is slow, dependent on manual calls, and lacks a centralized donor database.

**This leads to:**

- Delay in finding matching donors
- Patients losing lives due to late response
- Donors not receiving timely notifications
- Lack of real-time tracking of donations and pledges

**Proposed Salesforce Solution**

- Centralized donor registration with blood group, organ pledge, and location
- Emergency request management by hospitals/NGOs
- Smart donor matching based on blood group, location, and availability
- Automated notifications (SMS/Email) to nearby donors in emergencies
- Appointment scheduling and donation tracking
- Donation history with reminders for next eligible donation date
- Real-time dashboards for hospitals, NGOs, and authorities

## Use Cases

1. **Donor Registration & Management**– Donors can register with details like blood group, organ pledge, and availability.

2. **Hospital Emergency Requests** – Hospitals can log urgent blood/organ requests.

3. **Smart Matching & Alerts**– System automatically matches and notifies nearby eligible donors.

4. **Appointment Scheduling** – Donors confirm donation slots, hospitals update status.

5. **Donation History & Follow-**ups – Tracks donor history and sends reminders when eligible again.

6. **Organ Pledge Tracking** – Maintains a registry of pledged organs for future needs.

7. **NGO & Volunteer Collaboration**– NGOs manage blood camps and donor mobilization.

8. **Dashboards & Reporting** – Real-time insights on donor availability, requests, and completed donations.

## Salesforce Features Used

**Data Modeling**: Custom objects for Donors, Requests, Appointments, Organ Pledges

**Automation**: Flow Builder, Email Alerts, Validation Rules, Approval Process

**Apex Development**: Triggers for donor matching, Scheduled Apex for reminders

**Lightning UI**: Record Pages, Tabs, LWC components for donor/hospital dashboards

**Integration**: SMS/Email API for donor alerts, possible NGO/hospital system integration

**Reports & Dashboards**: Donation trends, request fulfillment rate, NGO contribution

## Example Dashboard Metrics

- Active donors by blood group
- Pending vs Completed donation requests
- Organ pledges by type (Kidney, Liver, Cornea, etc.)
- Lives saved through successful matches

## Impact

This system ensures:

- Faster response to emergencies
- Increased donor participation through automation
- Transparent tracking of donations and pledges
- Social good by saving lives with technology

## Future Enhancements

- Mobile App Integration for donor check-in
- AI-based donor prediction & availability scoring
- Government/Red Cross integration for wider impact
- Multi-language support for accessibility

# Phase 1 Problem Understanding & Industry Analysis

---

## Goal of Phase 1

To understand the pain points in the healthcare donation process, analyze stakeholders, and design an improved workflow using Salesforce CRM. This phase focuses on identifying requirements, mapping current vs. proposed processes, and researching industry best practices.

---

## 1 Requirement Gathering

**What was done:**

- Studied pain points of **blood banks, hospitals, NGOs, patients, and donors**.
- Identified key challenges in the current system.

**Key Requirements Identified:**

- Difficulty in quickly finding a suitable donor during emergencies.
- No centralized system to track donor availability (blood group, organ compatibility, location).
- Manual communication between hospitals, patients, and donors causes delays.
- Lack of reminders for repeat donations.
- No analytics for monitoring donor trends and shortages.

□ **Output:** Requirements document listing problems to solve.

---

## 2 Stakeholder Analysis

**Stakeholders Identified:**

- **Donors** → Individuals willing to donate blood/organs.
- **Patients & Families** → Need urgent access to donations.
- **Hospitals/Clinics** → Require quick donor matching and patient history tracking.
- **Blood Banks & NGOs** → Manage donor lists and blood stock.

- **Government Health Agencies** → Require donation data for policy and public health planning.

**Goals of Stakeholders:**

- Donors: Easy scheduling & reminders.
- Patients: Quick access to compatible donors.
- Hospitals: Real-time updates on donor availability.
- NGOs/Blood Banks: Centralized donor management.
- Government: Data analytics for shortages and trends.

□ **Output:** Stakeholder matrix (stakeholder + their goals).

---

# 3 Business Process Mapping

### Current Process (Manual) – "As-Is":

1. Patient urgently needs blood.
2. Family calls multiple hospitals/blood banks.
3. If a donor is found → manual coordination begins.
4. Organ donations delayed due to poor inter-hospital communication.

### Proposed Process (With Salesforce CRM) – "To-Be":

1. Patient request logged into CRM (via web/app/call center).
2. CRM searches donor database (blood group, organ compatibility, location).
3. Nearest donor notified automatically (SMS/Email/App).
4. Hospital updates donation status in real-time.
5. Dashboards show donations, stock, and shortages for administrators.

□ **Output:** "As-Is" and "To-Be" process flow diagrams.

---

# 4 Industry-Specific Use Case Analysis

**Healthcare Industry Pain Points:**

- Time-sensitive emergency donations.
- Lack of real-time donor-patient matching.
- No follow-up system for repeat donors.

**Salesforce CRM Use Cases:**

- **Lead Management:** Donor = Lead, Patient Request = Case.
- **Service Console:** Hospital staff manage requests on a single screen.
- **Automation:** Automated reminders for donor's next donation date.
- **Dashboards:** Admins monitor donation trends, stock, and shortages.

□ **Output:** Documented healthcare-specific Salesforce use cases.

---

## 5 AppExchange Exploration

**Apps/Tools Explored:**

- **Salesforce Health Cloud** – Comprehensive healthcare solution.
- **Donor Management Apps** – Used by nonprofits for donor records.
- **Appointment Management Add-ons** – Enhance scheduling and follow-ups.

**Goal:** Identify what's already available and plan unique features like **smart donor matching** and **AI-powered reminders**.

□ **Output:** AppExchange research notes.

---

## End of Phase 1 Deliverables

- Problem statement & requirements document.
- Stakeholder analysis chart.
- Business process flow ("As-Is" vs "To-Be").
- Healthcare-specific Salesforce use cases.
- AppExchange research notes.

# Phase 2 – Org Setup & Configuration

## Goal of Phase 2

To configure the Salesforce Org with **company details, business hours, fiscal year, users, profiles, roles, OWD, and sharing rules** so that the system is secure and ready for custom object creation.

---

## Steps Followed

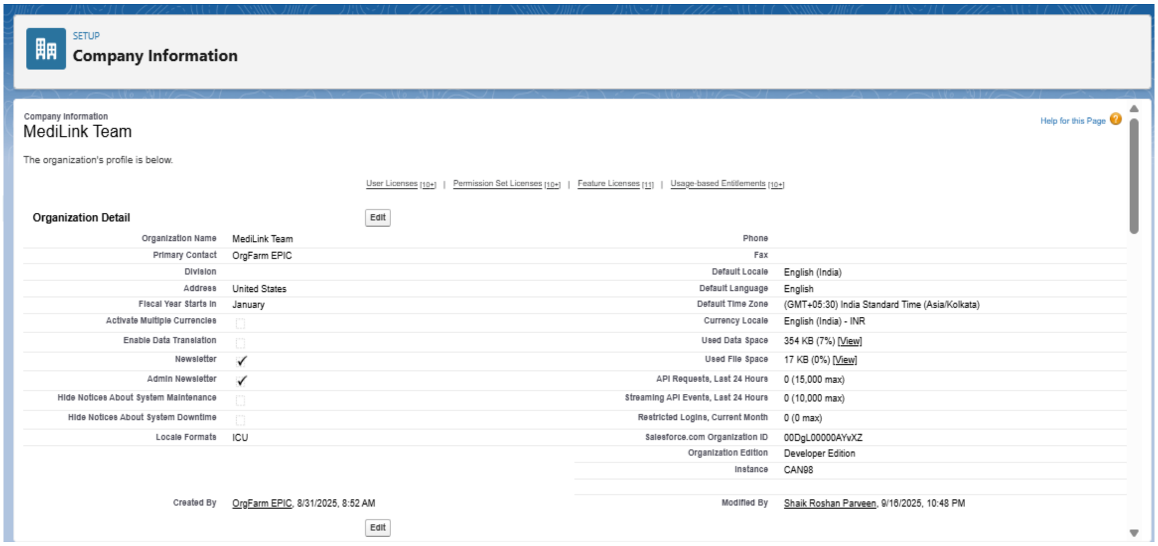
### 1. Salesforce Edition & Org Setup

- Logged into **Salesforce Developer Edition**.
- Went to **Setup → Quick Find → Company Information**.
- Verified edition = Developer Edition.

---

### 2. Company Profile Setup

- **Setup → Quick Find → Company Information**.
- Edited:
  - Time Zone → *Asia/Kolkata*
  - Default Currency → *INR*
  - Default Languages → *English, Telugu, Gujarati*
- Saved changes.


SETUP
Company Information
Help for this Page
User Licenses [10+] | Permission Set Licenses [10+] | Feature Licenses [11] | Usage-based Entitlements [10+]
Edit
Edit

Company Information

MediLink Team

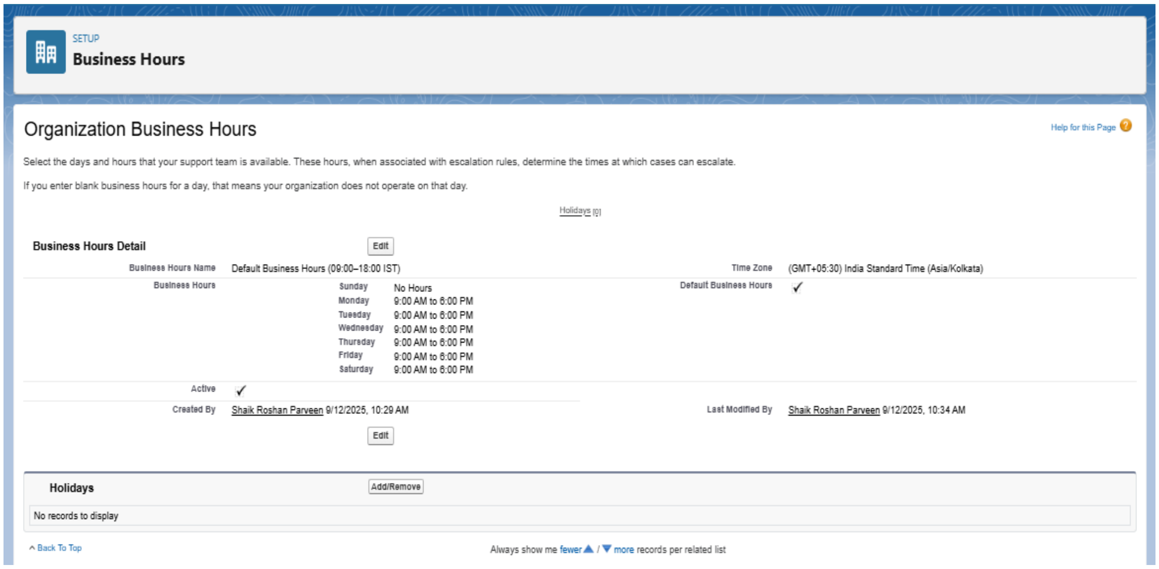
The organization's profile is below.

Organization Detail

| | | | |
|---|---|---|---|
| Organization Name | MediLink Team | Phone | |
| Primary Contact | OrgFarm EPIC | Fax | |
| Division | | Default Locale | English (India) |
| Address | United States | Default Language | English |
| Fiscal Year Starts In | January | Default Time Zone | (GMT+05:30) India Standard Time (Asia/Kolkata) |
| Activate Multiple Currencies | □ | Currency Locale | English (India) - INR |
| Enable Data Translation | □ | Used Data Space | 354 KB (7%) [View] |
| Newsletter | ✓ | Used File Space | 17 KB (0%) [View] |
| Admin Newsletter | ✓ | API Requests, Last 24 Hours | 0 (15,000 max) |
| Hide Notices About System Maintenance | □ | Streaming API Events, Last 24 Hours | 0 (10,000 max) |
| Hide Notices About System Downtime | □ | Restricted Logins, Current Month | 0 (0 max) |
| Locale Formats | ICU | Salesforce.com Organization ID | 00DgL00000AYvXZ |
| | | Organization Edition | Developer Edition |
| | | Instance | CAN98 |
| Created By | OrgFarm EPIC, 8/31/2025, 8:52 AM | Modified By | Shaik Roshan Parveen, 9/16/2025, 10:48 PM |

## 3. Business Hours & Holidays

- **Setup → Quick Find → Business Hours**.
- Clicked **New Business Hours**.
- Entered:
  - Label: *Hospital Business Hours*
  - Time Zone: Asia/Kolkata
- Checked **Active** and **Use these as default**.
- Set **Sunday → Closed**.
- Saved.



Organization Business Hours

Select the days and hours that your support team is available. These hours, when associated with escalation rules, determine the times at which cases can escalate.

If you enter blank business hours for a day, that means your organization does not operate on that day.

Business Hours Detail

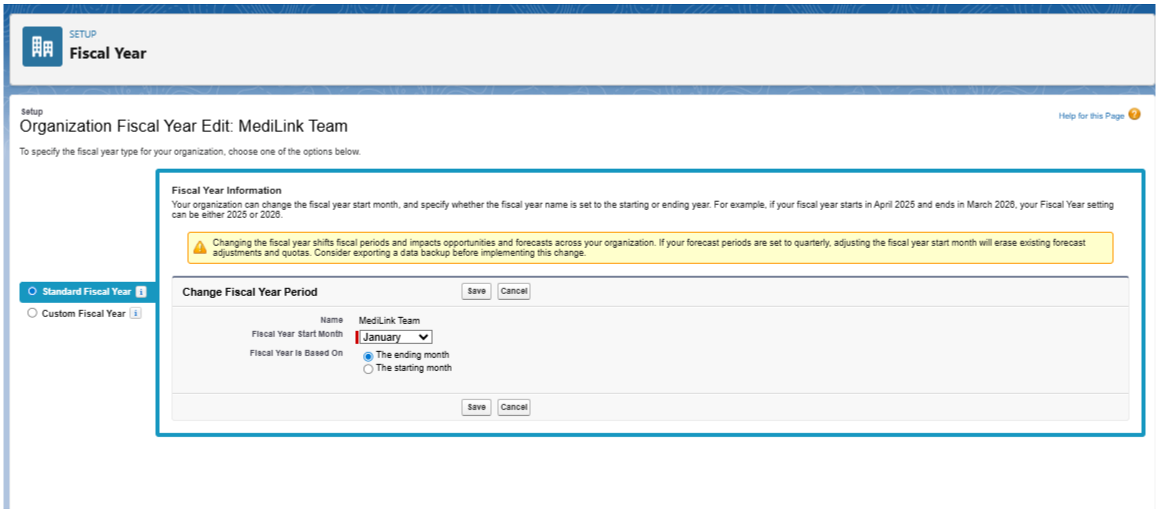
| | | | |
|---|---|---|---|
| Business Hours Name | Default Business Hours (09:00–18:00 IST) | Time Zone | (GMT+05:30) India Standard Time (Asia/Kolkata) |
| Business Hours | Sunday No Hours<br>Monday 9:00 AM to 6:00 PM<br>Tuesday 9:00 AM to 6:00 PM<br>Wednesday 9:00 AM to 6:00 PM<br>Thursday 9:00 AM to 6:00 PM<br>Friday 9:00 AM to 6:00 PM<br>Saturday 9:00 AM to 6:00 PM | Default Business Hours | ✓ |
| Active | ✓ | | |
| Created By | Shaik Roshan Parveen 9/12/2025, 10:29 AM | Last Modified By | Shaik Roshan Parveen 9/12/2025, 10:34 AM |

Holidays

No records to display

## 4. Fiscal Year Settings

- **Setup → Quick Find → Fiscal Year**.
- Verified that *Standard Fiscal Year* is enabled.
- Did not enable custom fiscal year.



SETUP

Fiscal Year

Setup

Organization Fiscal Year Edit: MediLink Team

Help for this Page

To specify the fiscal year type for your organization, choose one of the options below.

● Standard Fiscal Year

○ Custom Fiscal Year

**Fiscal Year Information**

Your organization can change the fiscal year start month, and specify whether the fiscal year name is set to the starting or ending year. For example, if your fiscal year starts in April 2025 and ends in March 2026, your Fiscal Year setting can be either 2025 or 2026.

Changing the fiscal year shifts fiscal periods and impacts opportunities and forecasts across your organization. If your forecast periods are set to quarterly, adjusting the fiscal year start month will erase existing forecast adjustments and quotas. Consider exporting a data backup before implementing this change.

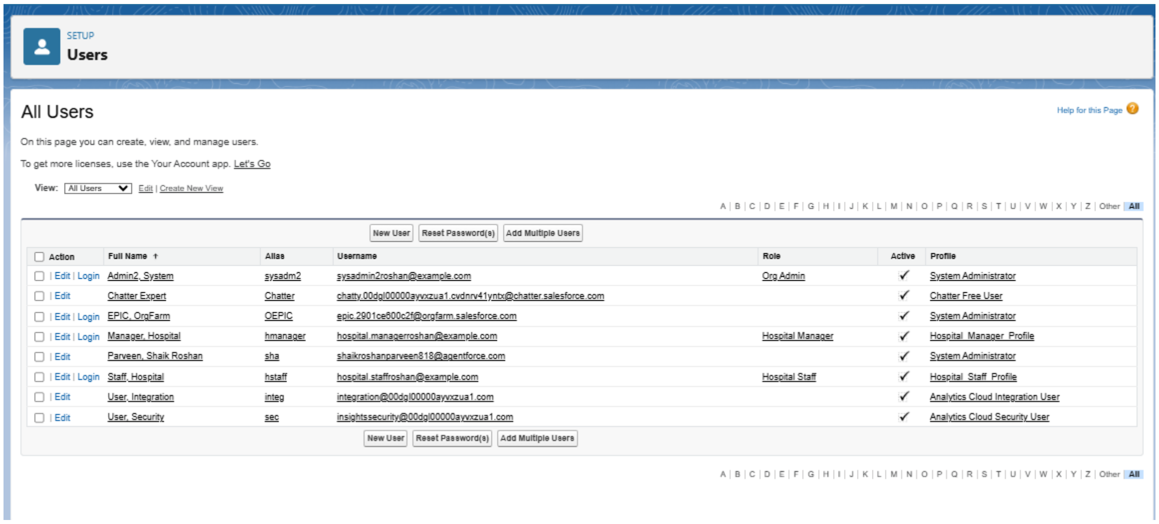
**Change Fiscal Year Period**

| | |
|---|---|
| Name | MediLink Team |
| Fiscal Year Start Month | January |
| Fiscal Year Is Based On | ● The ending month<br>○ The starting month |

---

## 5. User Setup & Licenses

- **Setup → Quick Find → Users → Users → New User**.
- Filled required fields:
    - First Name, Last Name, Email, Username.
    - User License: *Salesforce Platform*.
    - Profile: (selected custom profile after creation).
    - Role: (selected role from hierarchy).
- Checked **Generate password**.
- Saved.
- Repeated for each test user (Hospital Manager, Hospital Staff).



SETUP

Users

All Users

Help for this Page

On this page you can create, view, and manage users.

To get more licenses, use the Your Account app. Let's Go

View: All Users Edit | Create New View

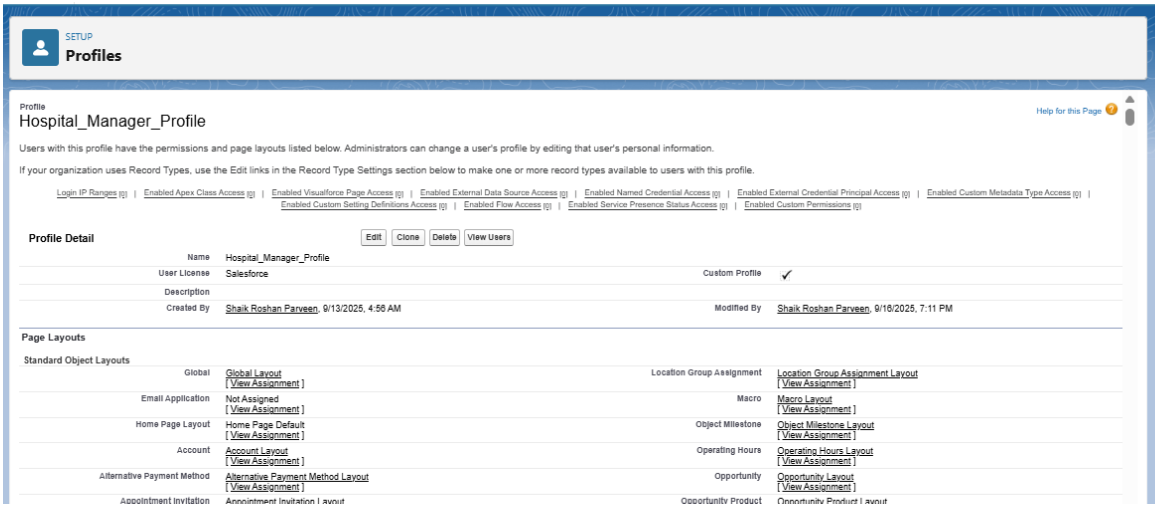
A | B | C | D | E | F | G | H | I | J | K | L | M | N | O | P | Q | R | S | T | U | V | W | X | Y | Z | Other | All
New User
Reset Password(s)
Add Multiple Users

| □ Action | Full Name ↑ | Alias | Username | Role | Active | Profile |
|---|---|---|---|---|---|---|
| □ \| Edit \| Login | Admin2, System | sysadm2 | sysadmin2roshan@example.com | Org Admin | ✓ | System Administrator |
| □ \| Edit | Chatter Expert | Chatter | chatty.00dgl00000ayvxzua1.cvdnrv41yntx@chatter.salesforce.com | | ✓ | Chatter Free User |
| □ \| Edit \| Login | EPIC, OrgFarm | OEPIC | epic.2901ce600c2f@orgfarm.salesforce.com | | ✓ | System Administrator |
| □ \| Edit \| Login | Manager, Hospital | hmanager | hospital.managerroshan@example.com | Hospital Manager | ✓ | Hospital_Manager_Profile |
| □ \| Edit | Parveen, Shaik Roshan | sha | shaikroshanparveen818@agentforce.com | | ✓ | System Administrator |
| □ \| Edit \| Login | Staff, Hospital | hstaff | hospital.staffroshan@example.com | Hospital Staff | ✓ | Hospital_Staff_Profile |
| □ \| Edit | User, Integration | integ | integration@00dgl00000ayvxzua1.com | | ✓ | Analytics Cloud Integration User |
| □ \| Edit | User, Security | sec | insightssecurity@00dgl00000ayvxzua1.com | | ✓ | Analytics Cloud Security User |

New User
Reset Password(s)
Add Multiple Users
A | B | C | D | E | F | G | H | I | J | K | L | M | N | O | P | Q | R | S | T | U | V | W | X | Y | Z | Other | All

## 6. Profiles

- **Setup → Quick Find → Profiles**.
- Selected **Standard User → Clone**.
- Named it **Hospital_Manager_Profile**.
- Edited **Object Settings**:
    - Donor__c → Read, Create, Edit, Delete.
    - Request__c → Read, Create, Edit, Delete.
    - Appointment__c → Read, Create, Edit.
- Set **Tabs → Default On** for Donor, Request, Appointment.
- Saved.
- Repeated to create **Hospital_Staff_Profile** (no Delete permission on Request).



SETUP

Profiles

Profile

Hospital_Manager_Profile

Help for this Page

Users with this profile have the permissions and page layouts listed below. Administrators can change a user's profile by editing that user's personal information.

If your organization uses Record Types, use the Edit links in the Record Type Settings section below to make one or more record types available to users with this profile.

Login IP Ranges [0] | Enabled Apex Class Access [0] | Enabled Visualforce Page Access [0] | Enabled External Data Source Access [0] | Enabled Named Credential Access [0] | Enabled External Credential Principal Access [0] | Enabled Custom Metadata Type Access [0] | Enabled Custom Setting Definitions Access [0] | Enabled Flow Access [0] | Enabled Service Presence Status Access [0] | Enabled Custom Permissions [0]
Edit
Clone
Delete
View Users

Profile Detail

| | | | |
|---|---|---|---|
| Name | Hospital_Manager_Profile | | |
| User License | Salesforce | Custom Profile | ✓ |
| Description | | | |
| Created By | Shaik Roshan Parveen, 9/13/2025, 4:56 AM | Modified By | Shaik Roshan Parveen, 9/16/2025, 7:11 PM |

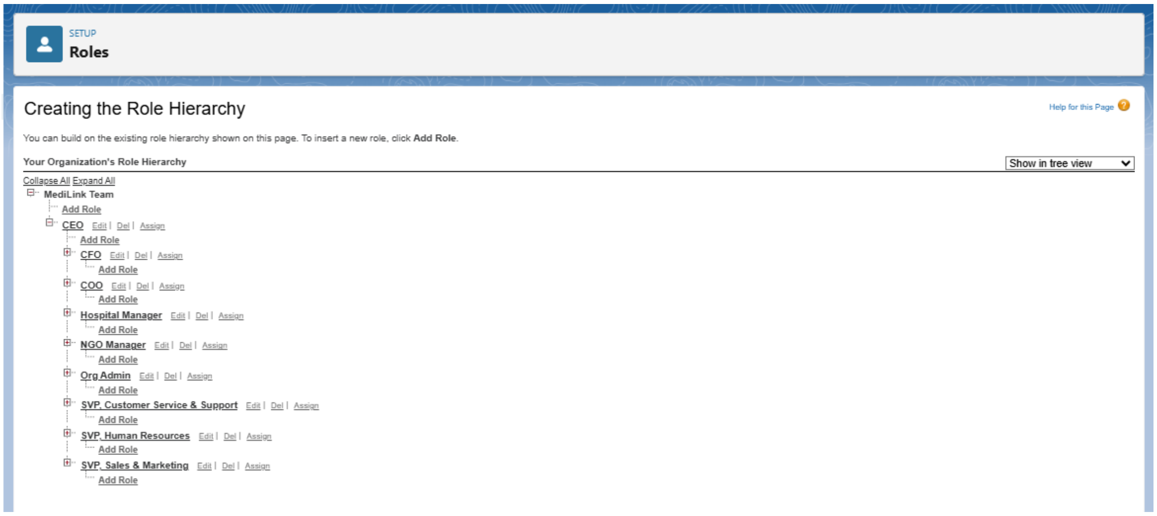
Page Layouts

Standard Object Layouts

| | | | |
|---|---|---|---|
| Global | Global Layout<br>[ View Assignment ] | Location Group Assignment | Location Group Assignment Layout<br>[ View Assignment ] |
| Email Application | Not Assigned<br>[ View Assignment ] | Macro | Macro Layout<br>[ View Assignment ] |
| Home Page Layout | Home Page Default<br>[ View Assignment ] | Object Milestone | Object Milestone Layout<br>[ View Assignment ] |
| Account | Account Layout<br>[ View Assignment ] | Operating Hours | Operating Hours Layout<br>[ View Assignment ] |
| Alternative Payment Method | Alternative Payment Method Layout<br>[ View Assignment ] | Opportunity | Opportunity Layout<br>[ View Assignment ] |
| Appointment Invitation | Appointment Invitation Layout | Opportunity Product | Opportunity Product Layout |

## 7. Roles & Role Hierarchy

- **Setup → Quick Find → Roles → Set Up Roles**.
- Clicked **Add Role**.
- Created roles in hierarchy:
  - System Admin (top)
  - Hospital Manager
  - Hospital Staff
- Saved each role.



Creating the Role Hierarchy

You can build on the existing role hierarchy shown on this page. To insert a new role, click **Add Role**.

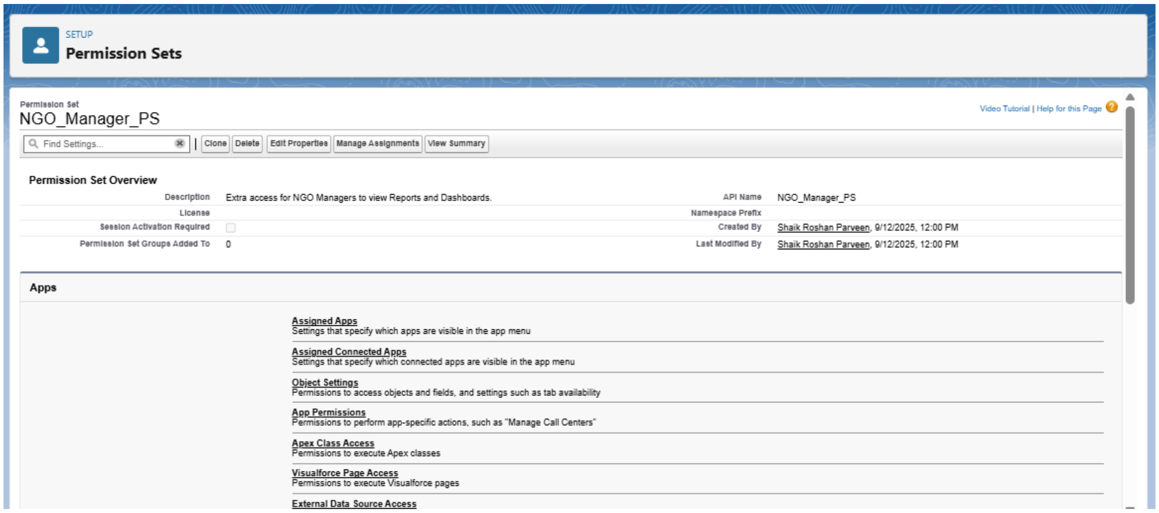
**Your Organization's Role Hierarchy**

- MediLink Team
  - Add Role
  - CEO Edit | Del | Assign
    - Add Role
    - CFO Edit | Del | Assign
      - Add Role
    - COO Edit | Del | Assign
      - Add Role
    - Hospital Manager Edit | Del | Assign
      - Add Role
    - NGO Manager Edit | Del | Assign
      - Add Role
    - Org Admin Edit | Del | Assign
      - Add Role
    - SVP, Customer Service & Support Edit | Del | Assign
      - Add Role
    - SVP, Human Resources Edit | Del | Assign
      - Add Role
    - SVP, Sales & Marketing Edit | Del | Assign
      - Add Role

---

## 8. Permission Sets (Optional)

- **Setup → Quick Find → Permission Sets → New**.
- Created:
  - NGO_Manager_PS
  - Donor_Portal_Access_PS
- Saved (assignments not done yet).



Permission Set

NGO_Manager_PS

Permission Set Overview

| | | | |
|---|---|---|---|
| Description | Extra access for NGO Managers to view Reports and Dashboards. | API Name | NGO_Manager_PS |
| License | | Namespace Prefix | |
| Session Activation Required | □ | Created By | Shaik Roshan Parveen, 9/12/2025, 12:00 PM |
| Permission Set Groups Added To | 0 | Last Modified By | Shaik Roshan Parveen, 9/12/2025, 12:00 PM |

Apps

Assigned Apps
Settings that specify which apps are visible in the app menu

Assigned Connected Apps
Settings that specify which connected apps are visible in the app menu

Object Settings
Permissions to access objects and fields, and settings such as tab availability

App Permissions
Permissions to perform app-specific actions, such as "Manage Call Centers"

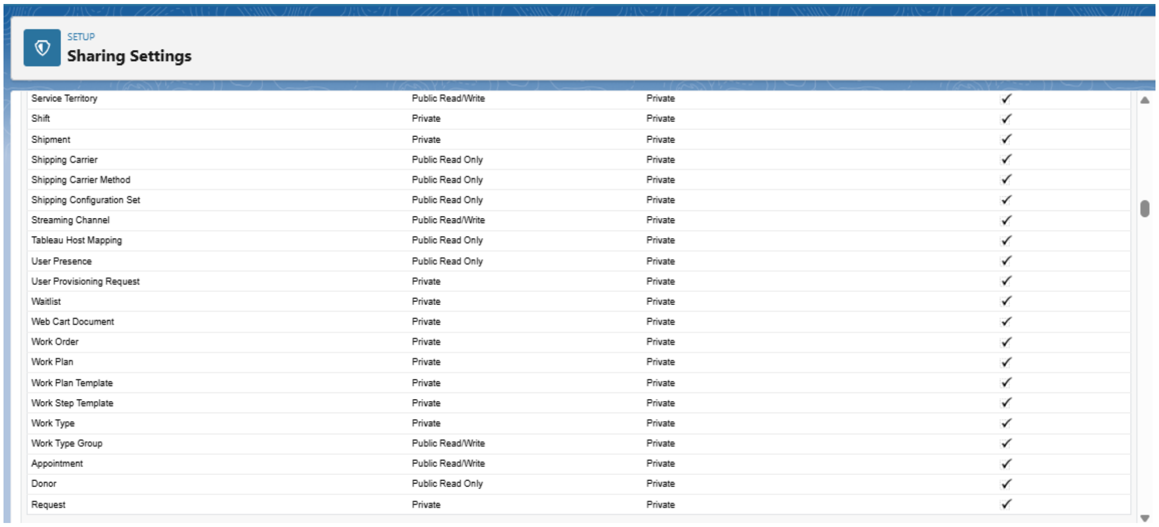
Apex Class Access
Permissions to execute Apex classes

Visualforce Page Access
Permissions to execute Visualforce pages

External Data Source Access

## 9. Org-Wide Defaults (OWD)

- **Setup → Quick Find → Sharing Settings**.
- Edited OWD defaults:
  - Donor__c → Public Read
  - Request__c → Private
  - Appointment__c → Controlled by Parent
- Save

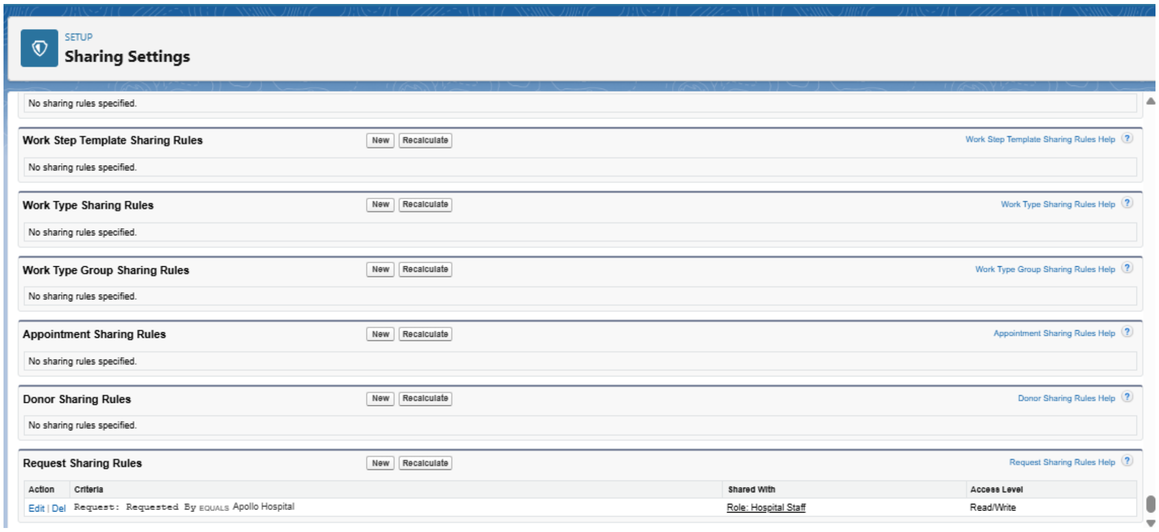


| | | | |
|---|---|---|---|
| Service Territory | Public Read/Write | Private | ✓ |
| Shift | Private | Private | ✓ |
| Shipment | Private | Private | ✓ |
| Shipping Carrier | Public Read Only | Private | ✓ |
| Shipping Carrier Method | Public Read Only | Private | ✓ |
| Shipping Configuration Set | Public Read Only | Private | ✓ |
| Streaming Channel | Public Read/Write | Private | ✓ |
| Tableau Host Mapping | Public Read Only | Private | ✓ |
| User Presence | Public Read Only | Private | ✓ |
| User Provisioning Request | Private | Private | ✓ |
| Waitlist | Private | Private | ✓ |
| Web Cart Document | Private | Private | ✓ |
| Work Order | Private | Private | ✓ |
| Work Plan | Private | Private | ✓ |
| Work Plan Template | Private | Private | ✓ |
| Work Step Template | Private | Private | ✓ |
| Work Type | Private | Private | ✓ |
| Work Type Group | Public Read/Write | Private | ✓ |
| Appointment | Public Read/Write | Private | ✓ |
| Donor | Public Read Only | Private | ✓ |
| Request | Private | Private | ✓ |

## 10. Sharing Rules

- **Setup → Quick Find → Sharing Settings** → Scroll to Request Sharing Rules.

- Clicked **New Sharing Rule**.
- Chose criteria: Requested by Apollo Hosipital
- Saved.


SETUP
New
Recalculate
Work Step Template Sharing Rules Help
Work Type Sharing Rules Help
Work Type Group Sharing Rules Help
Appointment Sharing Rules Help
Donor Sharing Rules Help
Request Sharing Rules Help

Sharing Settings

No sharing rules specified.

Work Step Template Sharing Rules

No sharing rules specified.

Work Type Sharing Rules

No sharing rules specified.

Work Type Group Sharing Rules

No sharing rules specified.

Appointment Sharing Rules

No sharing rules specified.

Donor Sharing Rules

No sharing rules specified.

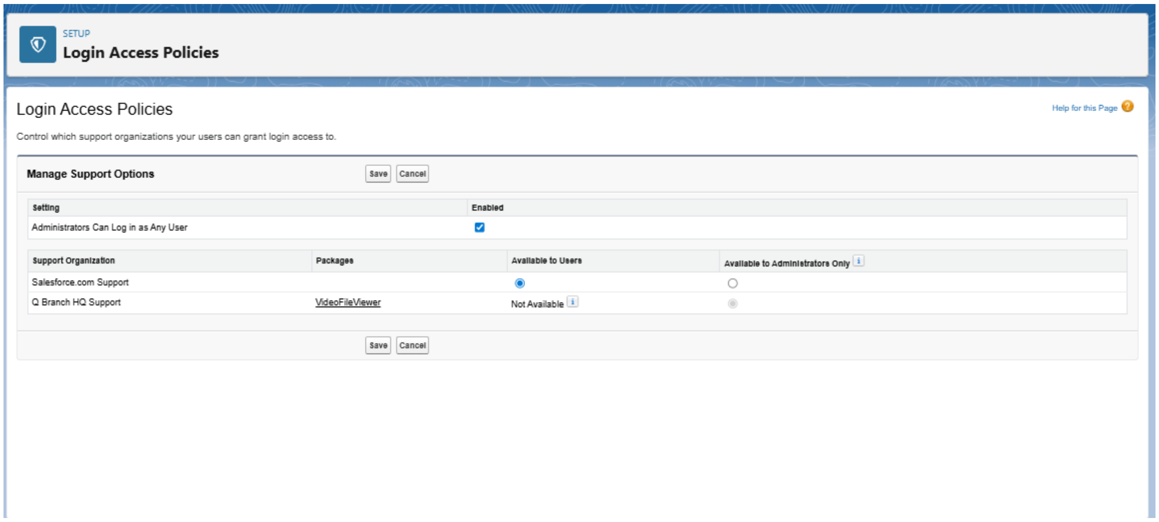
Request Sharing Rules

| Action | Criteria | Shared With | Access Level |
|---|---|---|---|
| Edit \| Del | Request: Requested By EQUALS Apollo Hospital | Role: Hospital Staff | Read/Write |

---

## 11. Login Access Policies

- **Setup → Quick Find → Login Access Policies**.
- Checked **Administrators Can Log in as Any User**.
- Saved.



Login Access Policies

Login Access Policies

Control which support organizations your users can grant login access to.

Manage Support Options

| Setting | Enabled |
|---|---|
| Administrators Can Log in as Any User | ☑ |

| Support Organization | Packages | Available to Users | Available to Administrators Only |
|---|---|---|---|
| Salesforce.com Support | | ● | ○ |
| Q Branch HQ Support | VideoFileViewer | Not Available | ● |

---

## □ Outcome

At the end of Phase 2, I successfully configured:

- Company details (timezone, currency, languages).
- Business hours & fiscal year.
- Users, profiles, and roles.
- Security model with OWD & sharing rules.
- Admin login access for user testing.

# Phase 3 Documentation – Data Modeling & Relationships

---

## □ Goal of Phase 3

To design and implement the data model for the Hospital Donor Management System using Salesforce objects, fields, record types, layouts, and relationships. This ensures that all donor, request, and appointment data is properly structured, accessible, and secure.

---

## 1 Standard & Custom Objects

**What was done:**

- Reviewed **Standard Objects** (like User, Contact).
- Created **Custom Objects** specific to the project:
  - **Donor__c** → Stores donor details.
  - **Request__c** → Stores patient donation requests (blood/organ).
  - **Appointment__c** → Stores scheduled donation appointments.

□ **Output:** Custom objects for Donor, Request, Appointment created.

---

# 2 Fields

### What was done:

- Added necessary **Custom Fields** to capture data:

### Donor__c Fields:

- Name, Gender, DOB, Phone, Email, Address, Blood Group, Organ Donor (Checkbox), Availability.

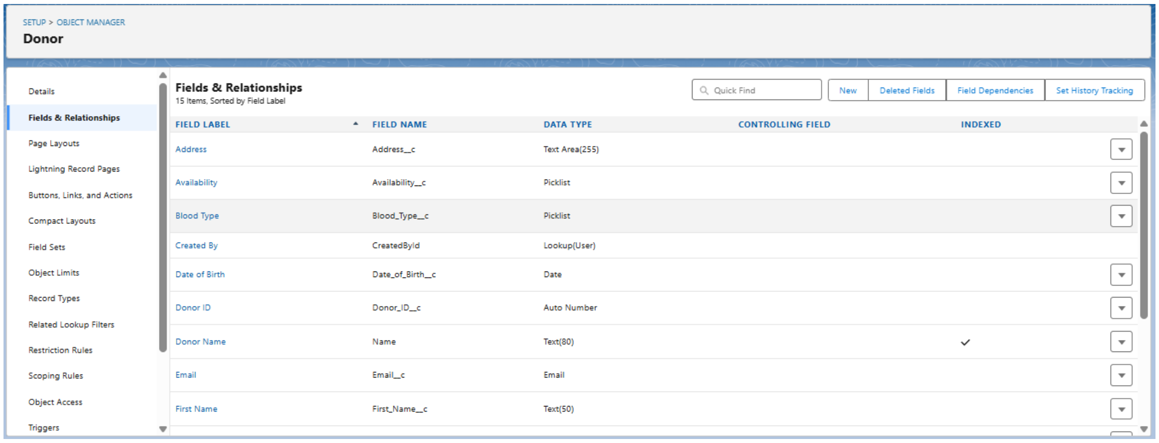
### Request__c Fields:

- Request Type (Blood/Organ), Urgency, Status, Related Donor Lookup.

### Appointment__c Fields:

- Appointment Date, Time, Notes, Donor Lookup, Request Lookup.

□ **Output:** Custom fields created per object.



Fields & Relationships

15 Items, Sorted by Field Label

| FIELD LABEL | FIELD NAME | DATA TYPE | CONTROLLING FIELD | INDEXED |
|---|---|---|---|---|
| Address | Address__c | Text Area(255) | | |
| Availability | Availability__c | Picklist | | |
| Blood Type | Blood_Type__c | Picklist | | |
| Created By | CreatedById | Lookup(User) | | |
| Date of Birth | Date_of_Birth__c | Date | | |
| Donor ID | Donor_ID__c | Auto Number | | |
| Donor Name | Name | Text(80) | | ✓ |
| Email | Email__c | Email | | |
| First Name | First_Name__c | Text(50) | | |

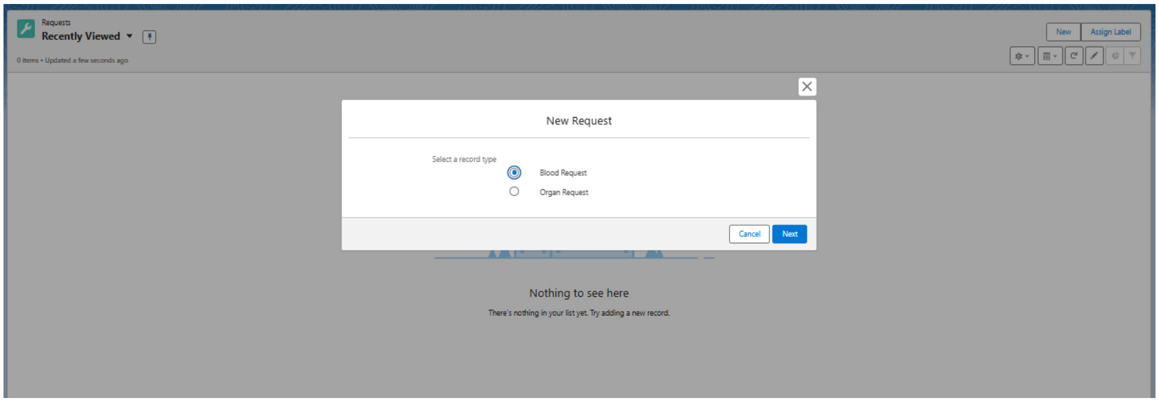
---

# 3 Record Types

### What was done:

- Applied **Record Types** only where needed:
  - **Donor__c:** Default record type only (all donors are similar).
  - **Request__c:** Two record types created → **Blood Request** & **Organ Request**.
  - **Appointment__c:** Default record type only.

□ **Output:** Record Types designed to separate blood vs. organ requests.



0 items • Updated a few seconds ago

New Request

Select a record type

● Blood Request

○ Organ Request

Nothing to see here

There's nothing in your list yet. Try adding a new record.

# 4 Page Layouts

### What was done:

- Created **different layouts for Manager & Staff** profiles.

### Donor__c Layouts:

- **Manager Layout:** Full access to all donor fields (Name, Gender, DOB, Phone, Email, Availability, Organ Donor).
- **Staff Layout:** Limited view (hides sensitive fields like DOB, Email, Phone).

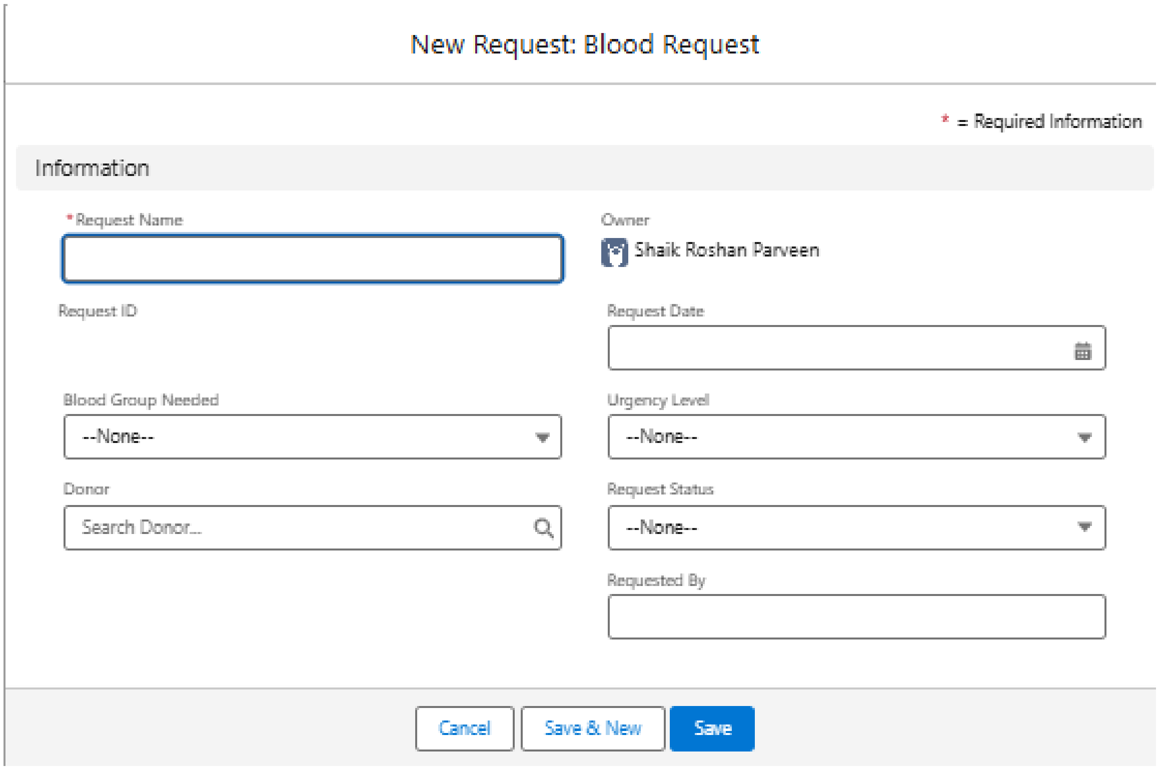
### Request__c Layouts:

- **Manager Layout:** Full details including urgency & requested by.
- **Staff Layout:** Can create/edit but restricted from viewing/deleting manager-only fields.

### Appointment__c Layouts:

- **Manager Layout:** Can see appointment notes.
- **Staff Layout:** Notes field hidden.

□ **Output:** Separate layouts for each profile with controlled visibility.



New Request: Blood Request

* = Required Information

Information

*Request Name

Owner
Shaik Roshan Parveen

Request ID

Request Date

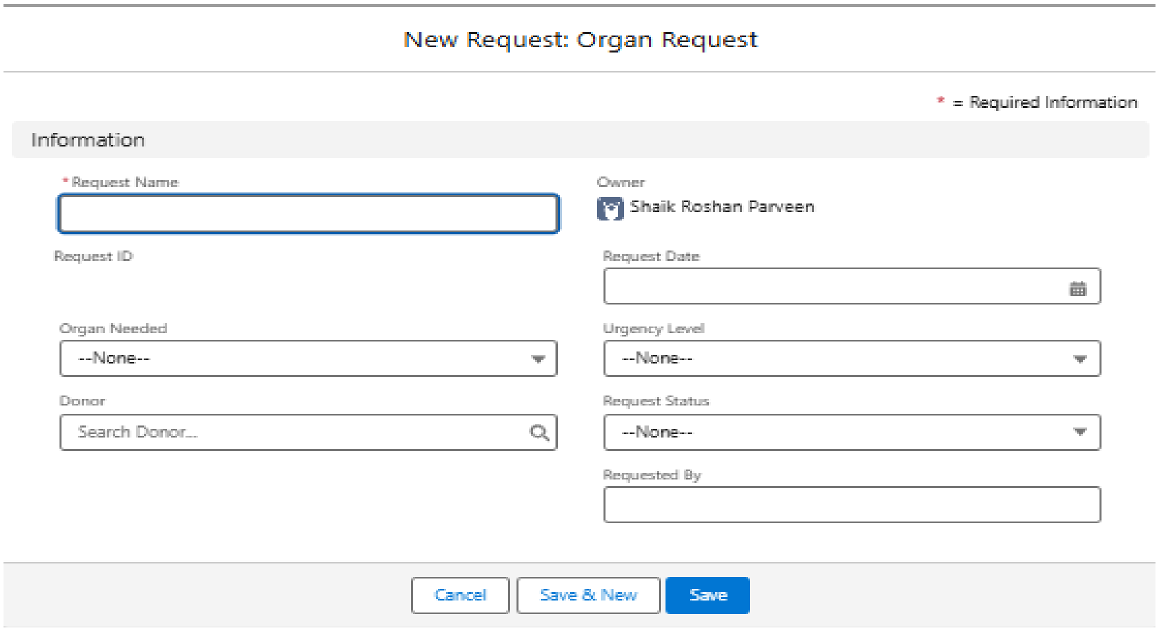
Blood Group Needed
--None--

Urgency Level
--None--

Donor
Search Donor...

Request Status
--None--

Requested By



New Request: Organ Request

* = Required Information

Information

*Request Name

Owner
Shaik Roshan Parveen

Request ID

Request Date

Organ Needed
--None--

Urgency Level
--None--

Donor
Search Donor...

Request Status
--None--

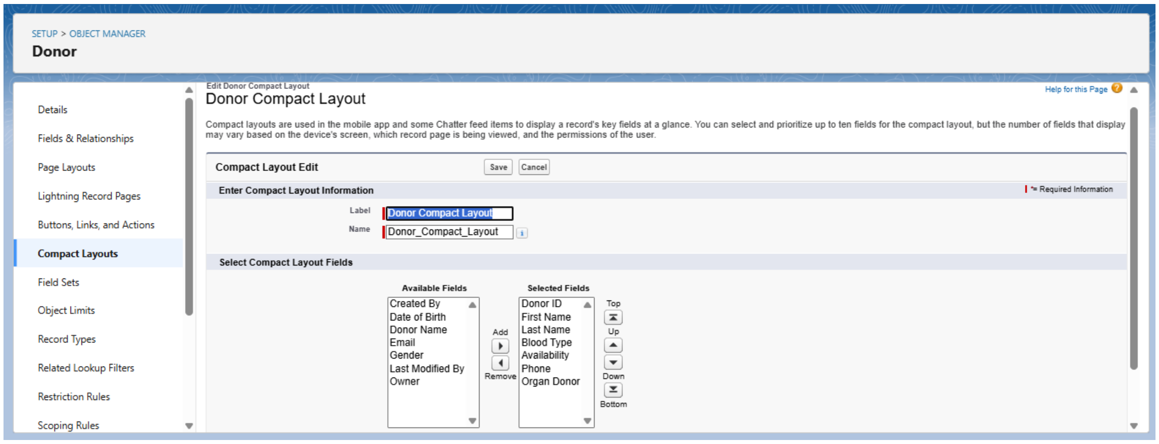
Requested By

# 5 Compact Layouts

### What was done:

- Created **Compact Layouts** for quick view in highlights panel.
- Example: Donor__c → Name, Blood Group, Availability, Phone.

□ **Output:** Compact layouts applied to display key fields in record previews.



Edit Donor Compact Layout

Donor Compact Layout

Compact layouts are used in the mobile app and some Chatter feed items to display a record's key fields at a glance. You can select and prioritize up to ten fields for the compact layout, but the number of fields that display may vary based on the device's screen, which record page is being viewed, and the permissions of the user.

Compact Layout Edit

Enter Compact Layout Information

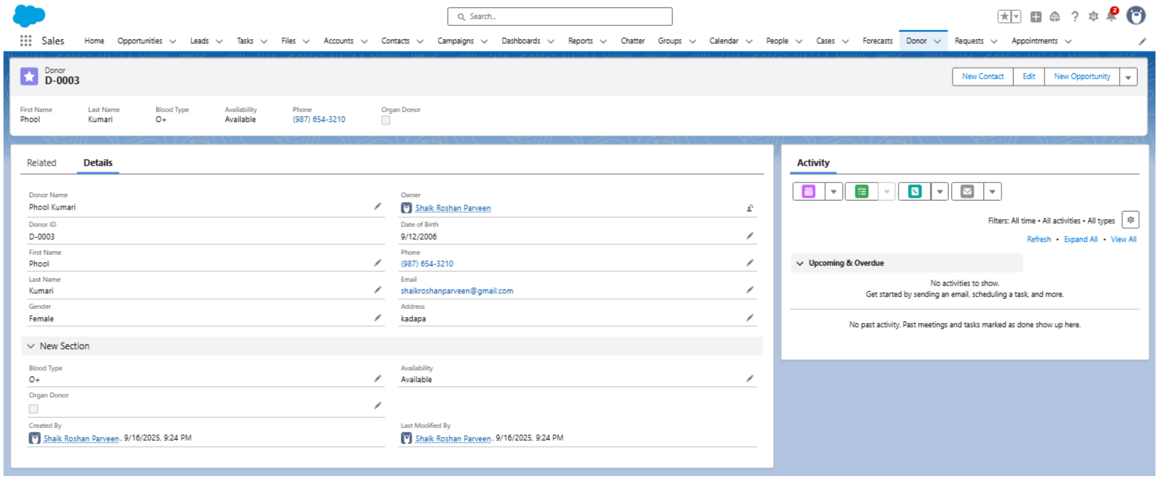
| = Required Information

Label: Donor Compact Layout

Name: Donor_Compact_Layout

Select Compact Layout Fields

| Available Fields | Selected Fields |
|---|---|
| Created By | Donor ID |
| Date of Birth | First Name |
| Donor Name | Last Name |
| Email | Blood Type |
| Gender | Availability |
| Last Modified By | Phone |
| Owner | Organ Donor |



Donor
D-0003

| First Name | Last Name | Blood Type | Availability | Phone | Organ Donor |
|---|---|---|---|---|---|
| Phool | Kumari | O+ | Available | (987) 654-3210 | □ |

Donor Name: Phool Kumari
Owner: Shaik Roshan Parveen
Donor ID: D-0003
Date of Birth: 9/12/2006
First Name: Phool
Phone: (987) 654-3210
Last Name: Kumari
Email: shaikroshanparveen@gmail.com
Gender: Female
Address: kadapa

New Section

Blood Type: O+
Availability: Available
Organ Donor: □
Created By: Shaik Roshan Parveen, 9/16/2025, 9:24 PM
Last Modified By: Shaik Roshan Parveen, 9/16/2025, 9:24 PM

Filters: All time • All activities • All types

Upcoming & Overdue

No activities to show.
Get started by sending an email, scheduling a task, and more.

No past activity. Past meetings and tasks marked as done show up here.

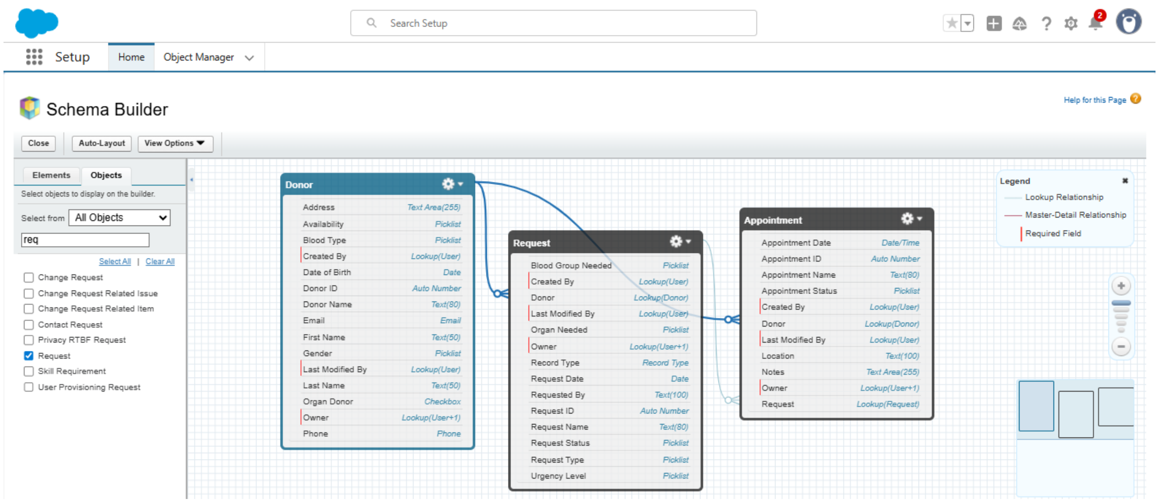
# 6 Schema Builder

### What was done:

- Used **Schema Builder** to visualize objects and their relationships.
- Verified **Donor** ↔ **Request** ↔ **Appointment** connections.

- Ensured all relationships (lookup/master-detail) are properly mapped.

□ **Output:** Graphical data model available for reference.



☐ Change Request
☐ Change Request Related Issue
☐ Change Request Related Item
☐ Contact Request
☐ Privacy RTBF Request
☑ Request
☐ Skill Requirement
☐ User Provisioning Request

| Donor | |
|---|---|
| Address | Text Area(255) |
| Availability | Picklist |
| Blood Type | Picklist |
| Created By | Lookup(User) |
| Date of Birth | Date |
| Donor ID | Auto Number |
| Donor Name | Text(80) |
| Email | Email |
| First Name | Text(50) |
| Gender | Picklist |
| Last Modified By | Lookup(User) |
| Last Name | Text(50) |
| Organ Donor | Checkbox |
| Owner | Lookup(User+1) |
| Phone | Phone |

| Request | |
|---|---|
| Blood Group Needed | Picklist |
| Created By | Lookup(User) |
| Donor | Lookup(Donor) |
| Last Modified By | Lookup(User) |
| Organ Needed | Picklist |
| Owner | Lookup(User+1) |
| Record Type | Record Type |
| Request Date | Date |
| Requested By | Text(100) |
| Request ID | Auto Number |
| Request Name | Text(80) |
| Request Status | Picklist |
| Request Type | Picklist |
| Urgency Level | Picklist |

| Appointment | |
|---|---|
| Appointment Date | Date/Time |
| Appointment ID | Auto Number |
| Appointment Name | Text(80) |
| Appointment Status | Picklist |
| Created By | Lookup(User) |
| Donor | Lookup(Donor) |
| Last Modified By | Lookup(User) |
| Location | Text(100) |
| Notes | Text Area(255) |
| Owner | Lookup(User+1) |
| Request | Lookup(Request) |

# 7 Relationships

### What was done:

- Applied correct relationship types:
- **Lookup Relationship:** Appointment__c → Donor__c (one donor can have many appointments).
- **Lookup Relationship:** Appointment__c → Request__c (one request can have many appointments).

□ **Output:** Clear relationship structure defined between objects.

# Phase 4: Process Automation

## Overview

In Phase 4, we implemented **Process Automation** in Salesforce to streamline operations for our *Blood & Organ Donation CRM*. This ensures data quality, reduces manual effort, and enables faster response during emergencies. We explored **Validation Rules, Flows, Email Alerts, Field Updates and Approval Processes**.

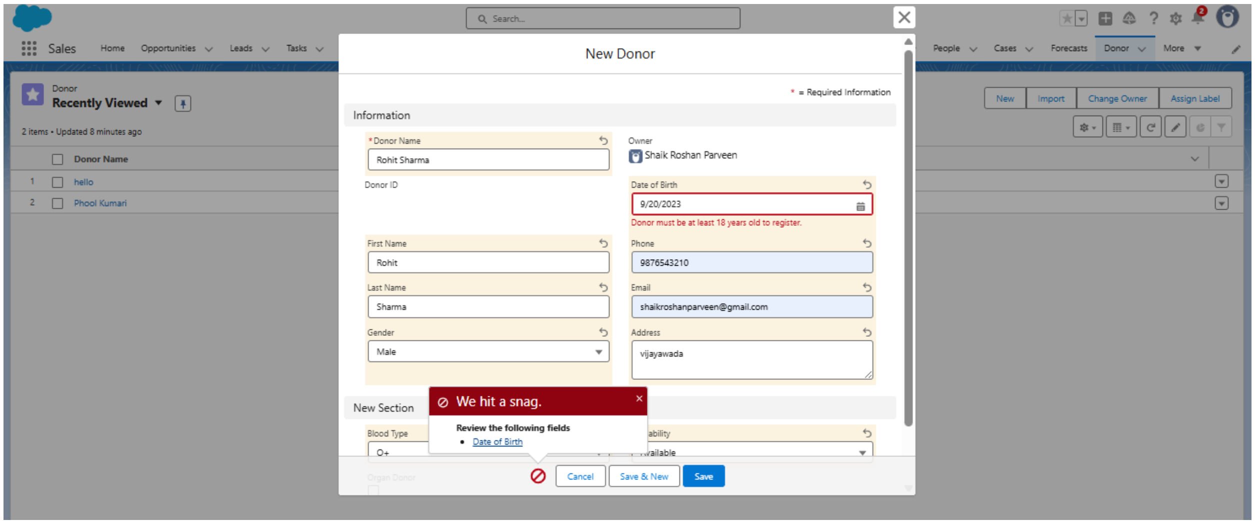
---

## 1. Validation Rules

**Purpose:** Ensure data accuracy and prevent invalid records.

**Rules Implemented:**

1. **Donor Minimum Age** — Donors must be at least 18 years old.
   - Formula: `TODAY() - Date_of_Birth__c < 6570`



Donor
Recently Viewed
2 items • Updated 8 minutes ago

| | Donor Name |
|---|---|
| 1 | hello |
| 2 | Phool Kumari |

New Donor

* = Required Information

Information

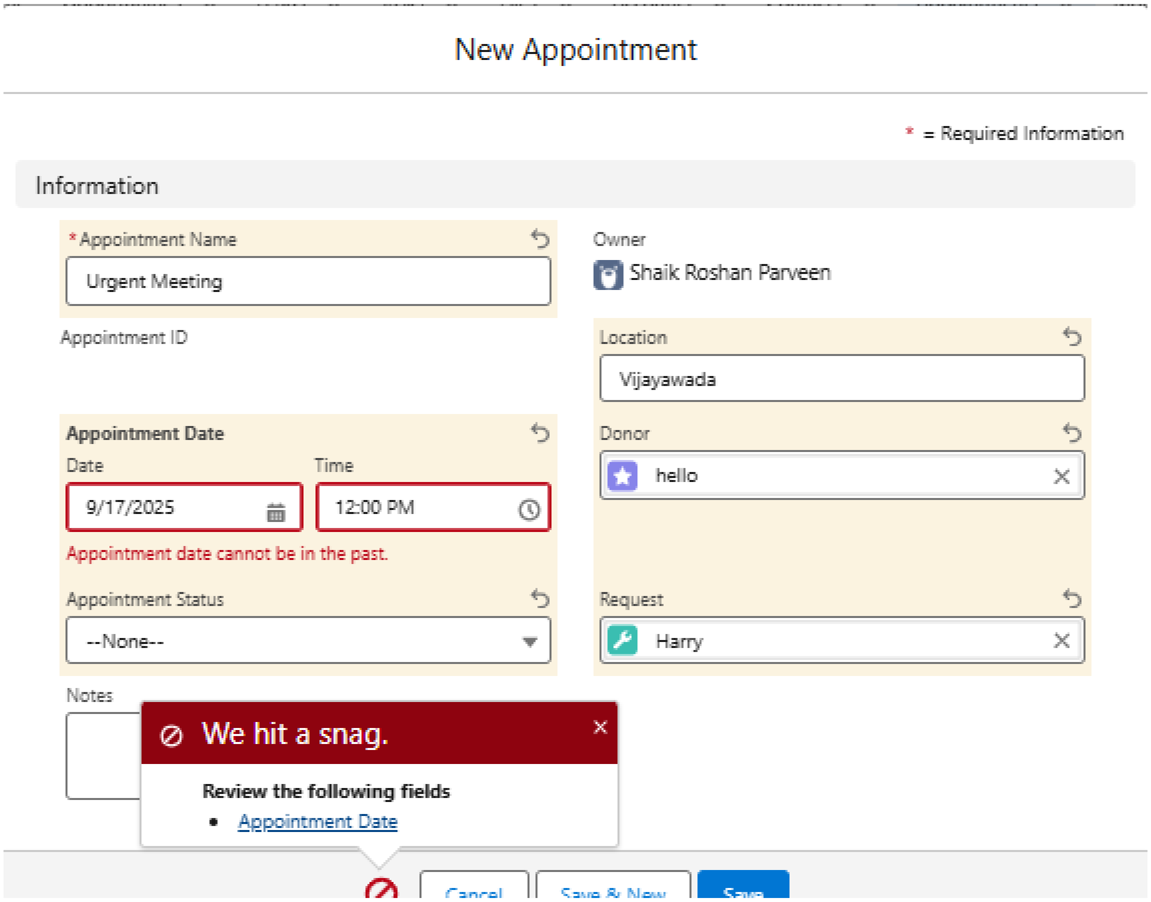
* Donor Name: Rohit Sharma
Owner: Shaik Roshan Parveen
Donor ID
Date of Birth: 9/20/2023
Donor must be at least 18 years old to register.
First Name: Rohit
Phone: 9876543210
Last Name: Sharma
Email: shaikroshanparveen@gmail.com
Gender: Male
Address: vijayawada

New Section

Blood Type: O+
Organ Donor

We hit a snag.
Review the following fields
- Date of Birth

2. **Appointment Date Check** — Appointment date/time cannot be in the past.
   - Formula: `Appointment_Date__c < NOW()`



New Appointment

* = Required Information

Information

*Appointment Name: Urgent Meeting

Owner: Shaik Roshan Parveen

Appointment ID

Location: Vijayawada

Appointment Date

Date: 9/17/2025

Time: 12:00 PM

Appointment date cannot be in the past.

Donor: hello

Appointment Status: --None--

Request: Harry

Notes

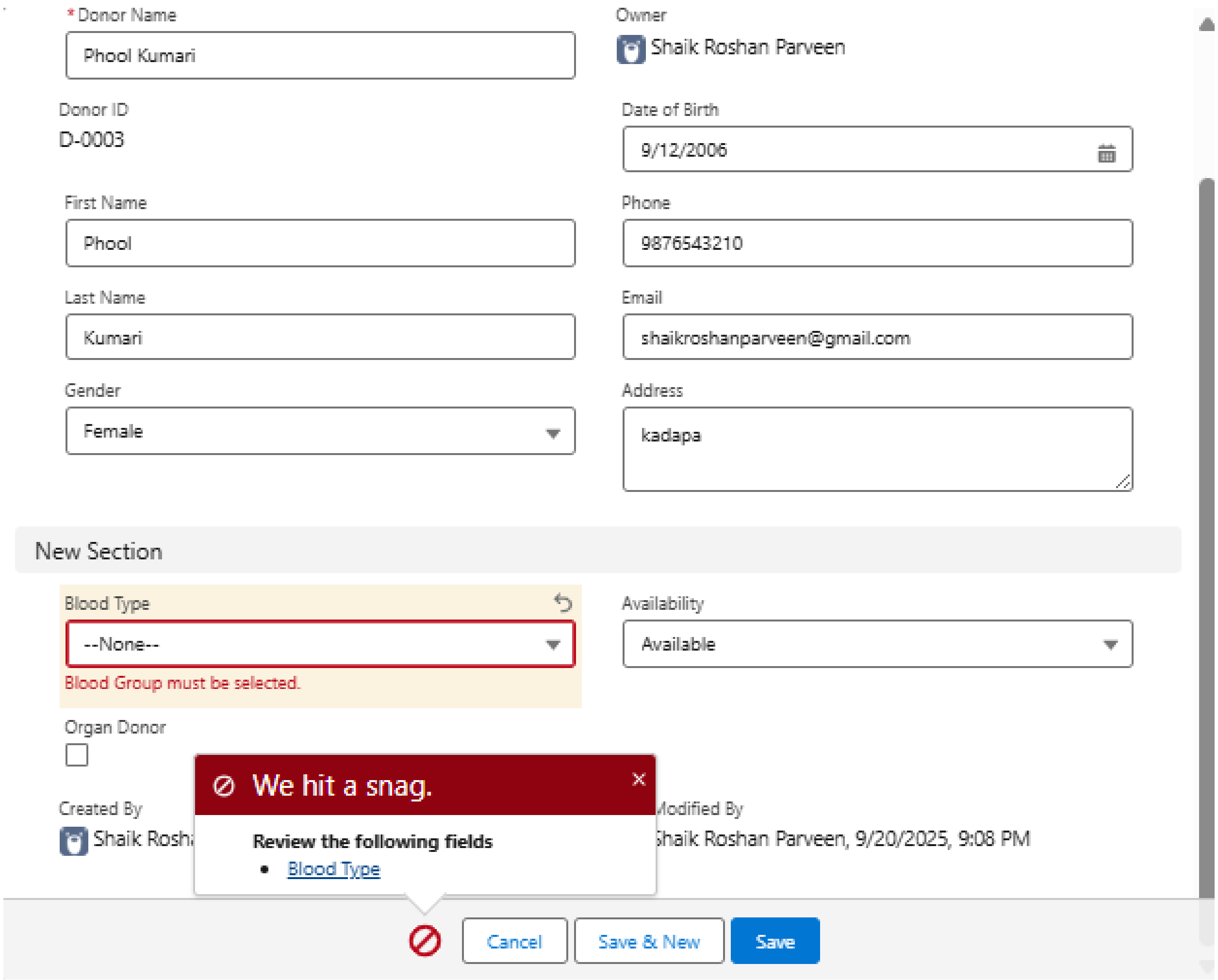
We hit a snag.

Review the following fields

- Appointment Date

3. **Blood Group Required** — Donor must have a blood group selected.
   - Formula: `ISBLANK(TEXT(Blood_Type__c))`

**Outcome:** Only valid donor, appointment, and request data can be saved in the system.



| | |
|---|---|
| * Donor Name<br>Phool Kumari | Owner<br>Shaik Roshan Parveen |
| Donor ID<br>D-0003 | Date of Birth<br>9/12/2006 |
| First Name<br>Phool | Phone<br>9876543210 |
| Last Name<br>Kumari | Email<br>shaikroshanparveen@gmail.com |
| Gender<br>Female | Address<br>kadapa |

New Section

| | |
|---|---|
| Blood Type<br>--None--<br>Blood Group must be selected. | Availability<br>Available |
| Organ Donor<br>□ | |
| Created By<br>Shaik Rosh | Modified By<br>Shaik Roshan Parveen, 9/20/2025, 9:08 PM |

We hit a snag.

Review the following fields

- Blood Type

## 2. Workflow Rules

Not implemented as standalone in Phase 4; replaced by more advanced Flow automations for better scalability and control.

## 3. Process Builder

Existing use cases replaced with Flow Builder for automating field updates and sending notifications.

# 4. Flow Builder

**Purpose:** Most powerful automation tool (Screen, Record-Triggered, Scheduled, Auto-launched).

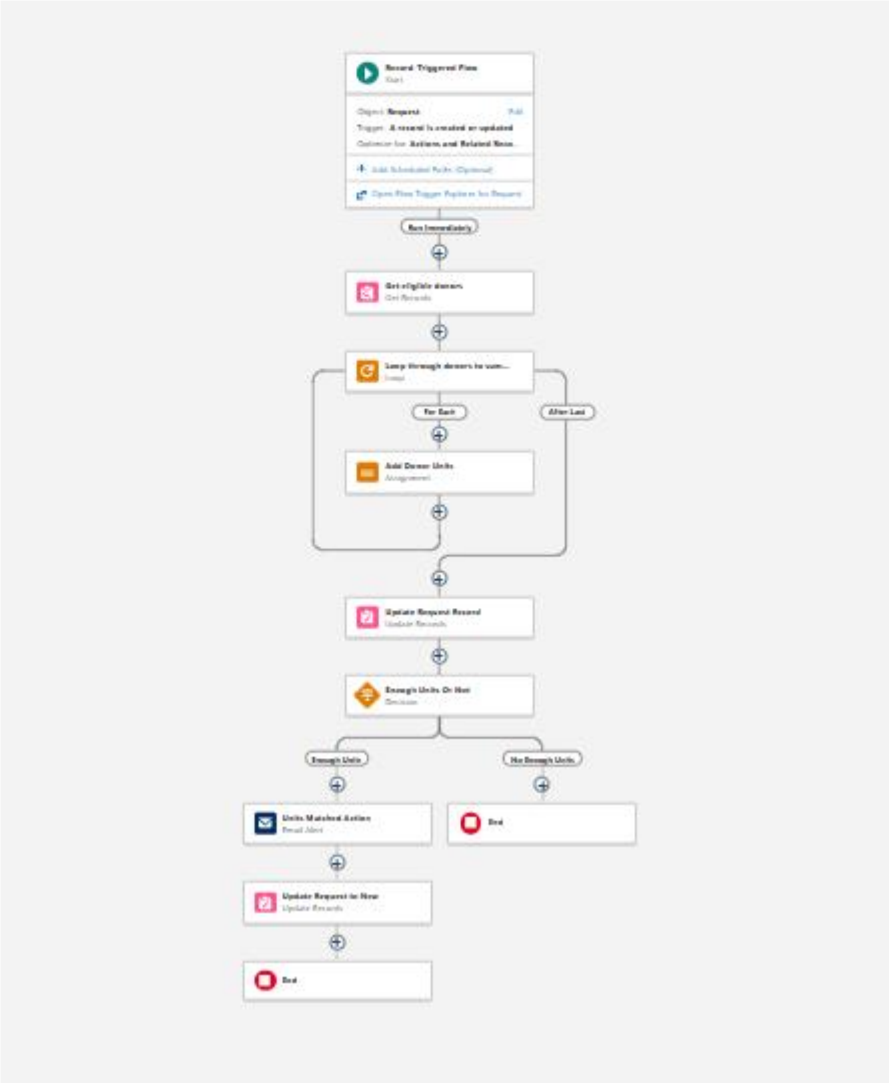
**Flows Implemented:**

1. **Record-Triggered Flow (Total Units Available)**
   - Trigger: On Blood_Request__c create/edit.
   - Logic:
     - Get all donors with matching blood type.
     - Sum their `Units_Available__c`.
     - Update Request field `Total_Units_Available__c`.
   - Condition: If Total Units ≥ Required Units → request is marked "Fulfillable."

**Outcome:** Automatically calculates donor availability for each request.

## 2. Record-Triggered Flow (Appointment Fulfillment)

- **Trigger:** On `Appointment__c` create/edit.
- **Logic:**
  - Get the related `Request__c` record.
  - Check if `Request_Status__c = In Progress` and `Donor__r.Availability__c = "Available"`.
  - If true →
    - Send **Email Alert** (appointment confirmation).
    - Update the related `Request__c.Request_Status__c = "Fulfilled"`.
- **Condition:** Only runs when Appointment is created or updated for a donor linked to an In Progress request.
- appointment is scheduled with an available donor.

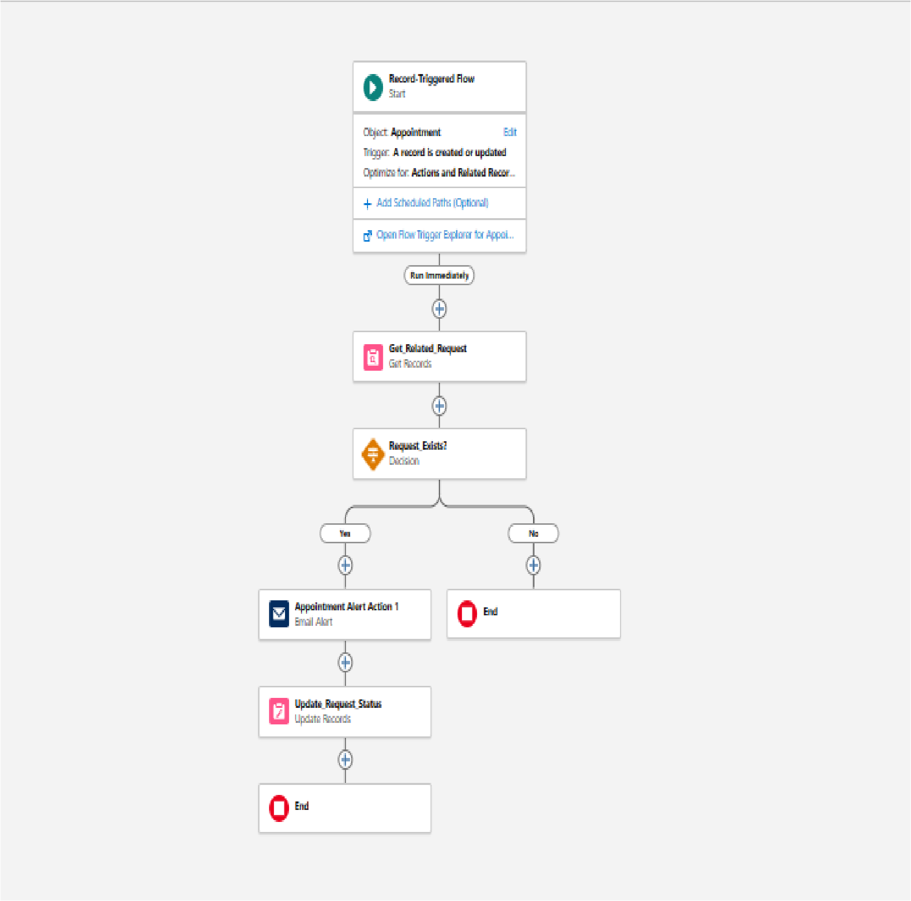


Object: Request

Trigger: A record is created or updated

Optimize for: Actions and Related Reco...

Fig: Total Units Available



Object: Appointment

Trigger: A record is created or updated

Optimize for: Actions and Related Recor...

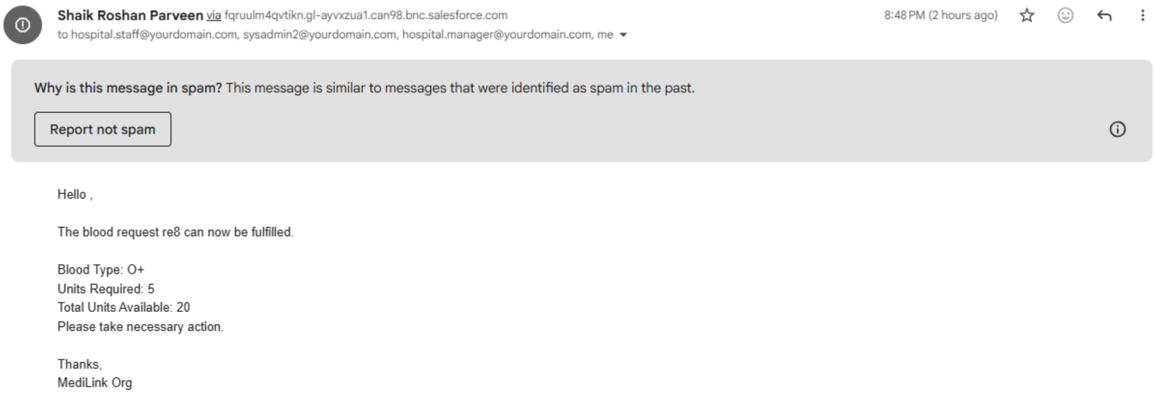
Fig: Appointment Fullfillment

---

# 5. Email Alerts

- Configured **Classic Email Templates** (New Request Notification).
- Appointment Confirmation Email, Blood Request Fulfillable Notification, Blood Request Approved.

**Outcome:** Consistent, automated communication.



Shaik Roshan Parveen via fqruulm4qvtikn.gl-ayvxzua1.can98.bnc.salesforce.com
to hospital.staff@yourdomain.com, sysadmin2@yourdomain.com, hospital.manager@yourdomain.com, me
8:48 PM (2 hours ago)

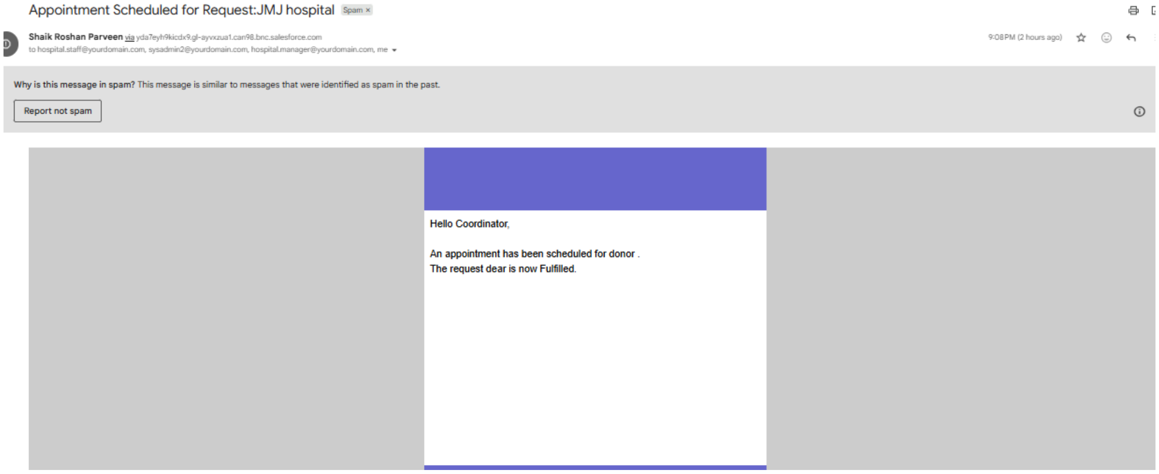
Why is this message in spam? This message is similar to messages that were identified as spam in the past.

Hello ,

The blood request re8 can now be fulfilled.

Blood Type: O+
Units Required: 5
Total Units Available: 20
Please take necessary action.

Thanks,
MediLink Org



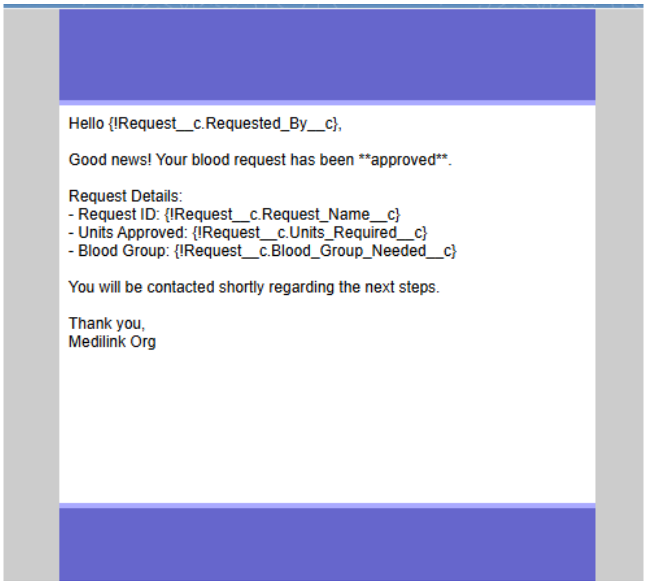
Appointment Scheduled for Request:JMJ hospital

Shaik Roshan Parveen via yda7eyh9kicdx9.gl-ayvxzua1.can98.bnc.salesforce.com
to hospital.staff@yourdomain.com, sysadmin2@yourdomain.com, hospital.manager@yourdomain.com, me
9:08 PM (2 hours ago)

Why is this message in spam? This message is similar to messages that were identified as spam in the past.

Hello Coordinator,

An appointment has been scheduled for donor .
The request dear is now Fulfilled.



Hello {!Request__c.Requested_By__c},

Good news! Your blood request has been **approved**.

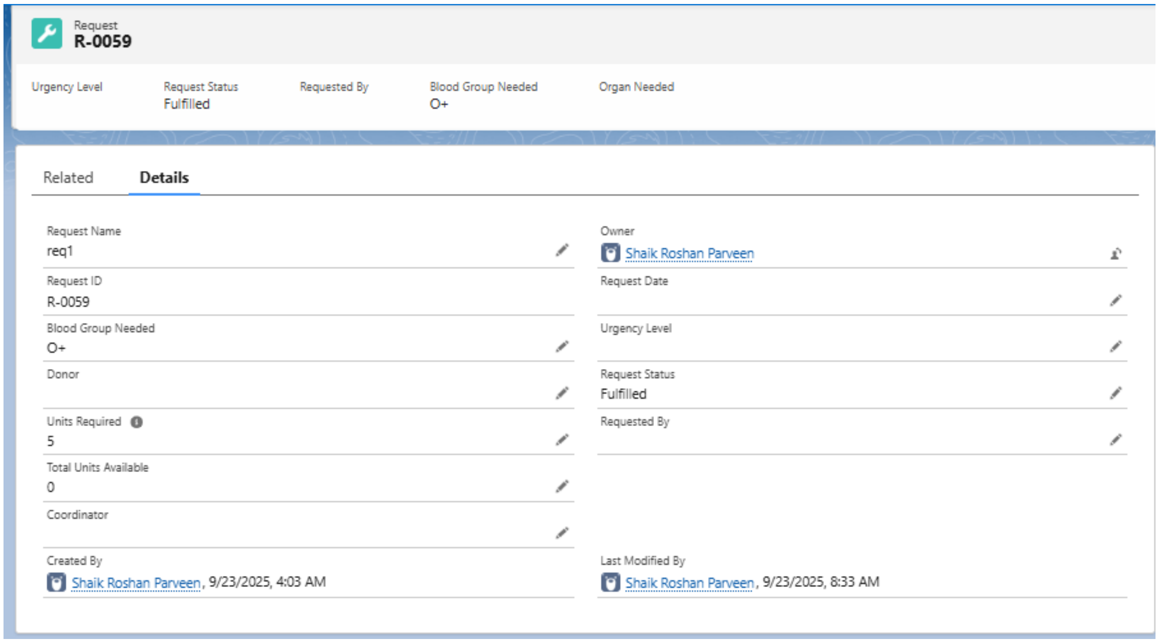
Request Details:
- Request ID: {!Request__c.Request_Name__c}
- Units Approved: {!Request__c.Units_Required__c}
- Blood Group: {!Request__c.Blood_Group_Needed__c}

You will be contacted shortly regarding the next steps.

Thank you,
Medilink Org

## 6. Field Updates

- In Process Builder, `Status__c` automatically updates to **Closed** once fulfilled.
- Helps maintain request lifecycle without manual intervention.



Request
**R-0059**

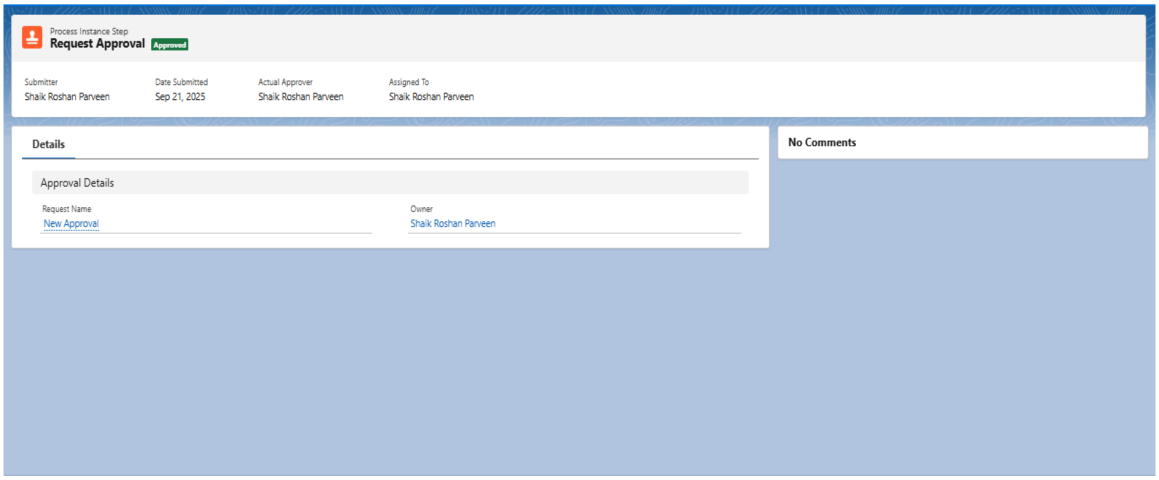
| Urgency Level | Request Status | Requested By | Blood Group Needed | Organ Needed |
|---|---|---|---|---|
|  | Fulfilled |  | O+ |  |

| | |
|---|---|
| Request Name<br>req1 | Owner<br>Shaik Roshan Parveen |
| Request ID<br>R-0059 | Request Date |
| Blood Group Needed<br>O+ | Urgency Level |
| Donor | Request Status<br>Fulfilled |
| Units Required<br>5 | Requested By |
| Total Units Available<br>0 | |
| Coordinator | |
| Created By<br>Shaik Roshan Parveen, 9/23/2025, 4:03 AM | Last Modified By<br>Shaik Roshan Parveen, 9/23/2025, 8:33 AM |

## 9. Approval Process

- Designed an **Approval Process** for **Blood Donation Requests**.
- Request goes through: Coordinator(or any user assigned).
- Once approved, request is marked as “Approved for Blood.”

**Outcome:** Sensitive requests follow compliance and multi-level approval.



Process Instance Step

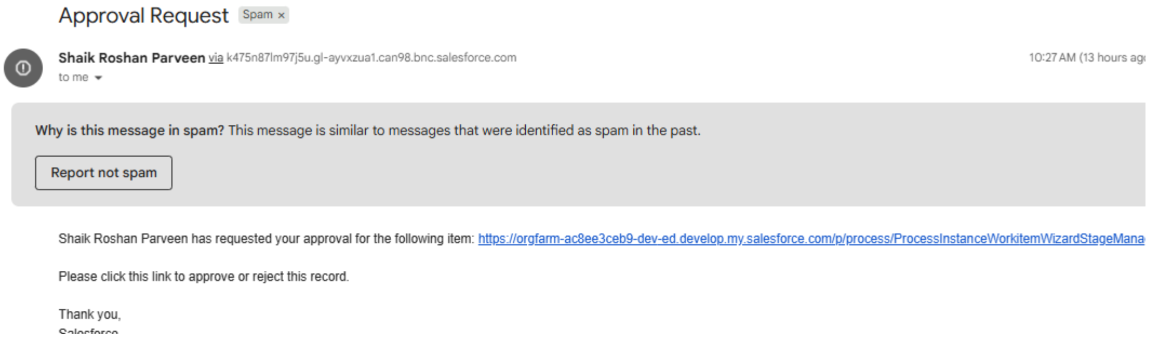
Request Approval

| Submitter | Date Submitted | Actual Approver | Assigned To |
|---|---|---|---|
| Shaik Roshan Parveen | Sep 21, 2025 | Shaik Roshan Parveen | Shaik Roshan Parveen |

Approval Details

| Request Name | Owner |
|---|---|
| New Approval | Shaik Roshan Parveen |

No Comments



Approval Request

Shaik Roshan Parveen via k475n87lm97j5u.gl-ayvxzua1.can98.bnc.salesforce.com 10:27 AM (13 hours ag

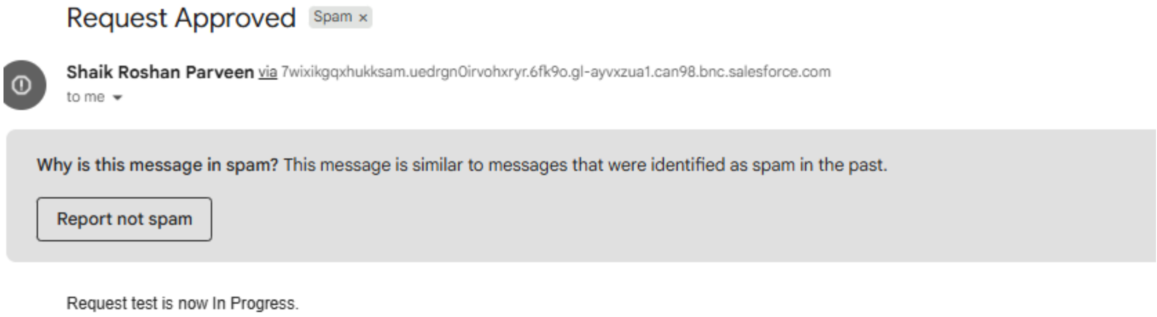
to me

Why is this message in spam? This message is similar to messages that were identified as spam in the past.

Shaik Roshan Parveen has requested your approval for the following item: https://orgfarm-ac8ee3ceb9-dev-ed.develop.my.salesforce.com/p/process/ProcessInstanceWorkitemWizardStageMana

Please click this link to approve or reject this record.

Thank you,
Salesforce



Request Approved

Shaik Roshan Parveen via 7wixikgqxhukksam.uedrgn0irvohxryr.6fk9o.gl-ayvxzua1.can98.bnc.salesforce.com

to me

Why is this message in spam? This message is similar to messages that were identified as spam in the past.

Request test is now In Progress.

# Conclusion

Phase 4 introduced **intelligent automation** into the CRM system:

- Data is validated before saving.
- Coordinators and volunteers are notified instantly.
- Status updates and donor availability calculations happen automatically.
- Sensitive requests follow a structured approval chain.

This automation reduces errors, saves time, and ensures **faster life-saving response** in critical donation scenarios.

# Phase 5: Apex Programming (Developer)

## Overview

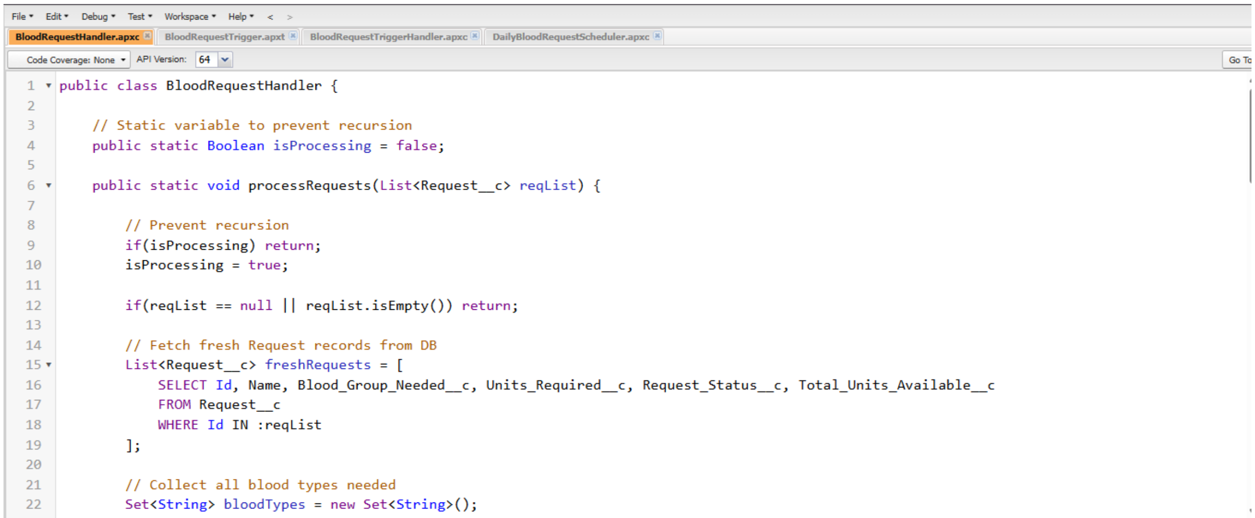
In Phase 5, we implemented **Apex Programming** to extend automation beyond declarative tools. Apex was used for custom logic to process blood requests, manage donor availability, and schedule automated jobs. This ensured **real-time updates, data consistency, and scalability**.

---

## 1. Apex Classes & Objects

**Implemented Classes:**

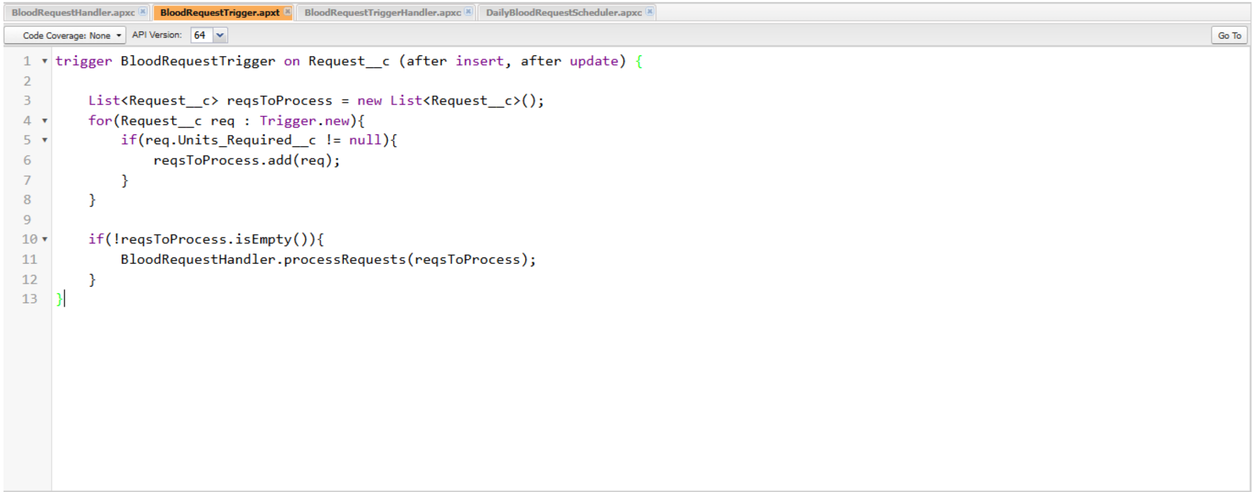
- **`BloodRequestHandler.cls`**
  - Contains core logic for processing blood requests.
  - Prevents recursion using a static flag.
  - Calculates total units available by blood type.
  - Deducts donor units when a request is fulfillable.
  - Updates request status (`New → In Progress`).
- **`DailyBloodRequestScheduler.cls`**
  - Implements `Schedulable` interface.
  - Runs daily to re-check pending requests.
  - Updates donor availability and request status in bulk.



```
public class BloodRequestHandler {

    // Static variable to prevent recursion
    public static Boolean isProcessing = false;

    public static void processRequests(List<Request__c> reqList) {

        // Prevent recursion
        if(isProcessing) return;
        isProcessing = true;

        if(reqList == null || reqList.isEmpty()) return;

        // Fetch fresh Request records from DB
        List<Request__c> freshRequests = [
            SELECT Id, Name, Blood_Group_Needed__c, Units_Required__c, Request_Status__c, Total_Units_Available__c
            FROM Request__c
            WHERE Id IN :reqList
        ];

        // Collect all blood types needed
        Set<String> bloodTypes = new Set<String>();
```

## 2. Apex Triggers (after insert, after update)

- **`BloodRequestTrigger.trigger`**
    - Fires on `Request__c` after insert and update.
    - Calls `BloodRequestHandler.processRequests()` to execute the business logic.
    - Ensures requests are processed immediately upon creation/update.



```
trigger BloodRequestTrigger on Request__c (after insert, after update) {

    List<Request__c> reqsToProcess = new List<Request__c>();
    for(Request__c req : Trigger.new){
        if(req.Units_Required__c != null){
            reqsToProcess.add(req);
        }
    }

    if(!reqsToProcess.isEmpty()){
        BloodRequestHandler.processRequests(reqsToProcess);
    }
}
```

## 3. Trigger Design Pattern

- Followed **one trigger per object** principle.
- Trigger kept lean by delegating logic to `BloodRequestHandler.cls`.
- Added a static Boolean variable to prevent recursive updates.

## 4. SOQL

- Used SOQL queries to fetch required records:
  - Fetch **Requests** with needed fields.
  - Fetch **Donors** filtered by blood type and availability.
- Example:
- `List<Donor__c> allDonors = [`
- `    SELECT Id, Name, Units_Available__c, Blood_Type__c`
- `    FROM Donor__c`
- `    WHERE Blood_Type__c IN :bloodTypes`
- `    AND Units_Available__c > 0`
- `    ORDER BY Units_Available__c DESC`
- `];`

## 5. Collections: List, Set, Map

- **List:** Stored donors to be updated in bulk.
- **Set:** Collected distinct blood types required by pending requests.
- **Map:** Grouped donors by blood type (`Map<String, List<Donor__c>>`).

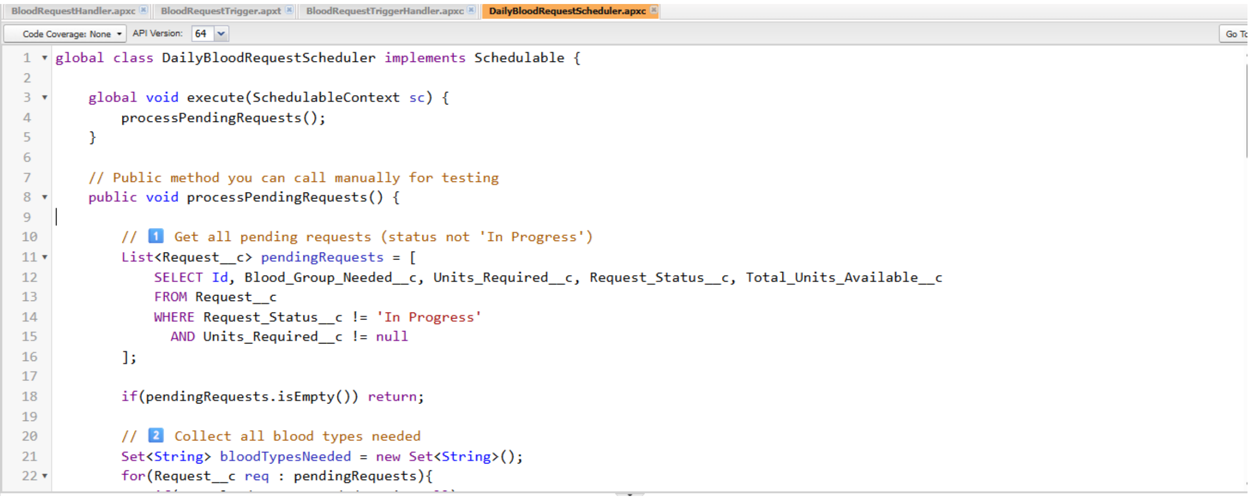
## 6. Control Statements

- **If-Else:** Checked whether requests had enough donors.
- **For Loops:** Iterated through donors to sum units and deduct availability.
- **Break Statements:** Stopped looping once required units were deducted.

## 7. Scheduled Apex

- **`DailyBloodRequestScheduler.cls`** implemented `Schedulable`.
- Runs daily to:
  - Find pending requests.
  - Recalculate donor availability.
  - Update request statuses accordingly.

**Outcome:** Automation runs in background without manual intervention.



```
global class DailyBloodRequestScheduler implements Schedulable {

    global void execute(SchedulableContext sc) {
        processPendingRequests();
    }

    // Public method you can call manually for testing
    public void processPendingRequests() {

        // 1️⃣ Get all pending requests (status not 'In Progress')
        List<Request__c> pendingRequests = [
            SELECT Id, Blood_Group_Needed__c, Units_Required__c, Request_Status__c, Total_Units_Available__c
            FROM Request__c
            WHERE Request_Status__c != 'In Progress'
              AND Units_Required__c != null
        ];

        if(pendingRequests.isEmpty()) return;

        // 2️⃣ Collect all blood types needed
        Set<String> bloodTypesNeeded = new Set<String>();
        for(Request__c req : pendingRequests){
```

# □ Conclusion

Phase 5 introduced **Apex-driven automation** into the project. With handler classes, triggers, SOQL queries, collections, and a scheduled job, the system now:

- Automatically matches donors with requests.
- Deducts donor units in real time.
- Updates request statuses (`New` → `In Progress`).
- Re-checks pending requests daily.

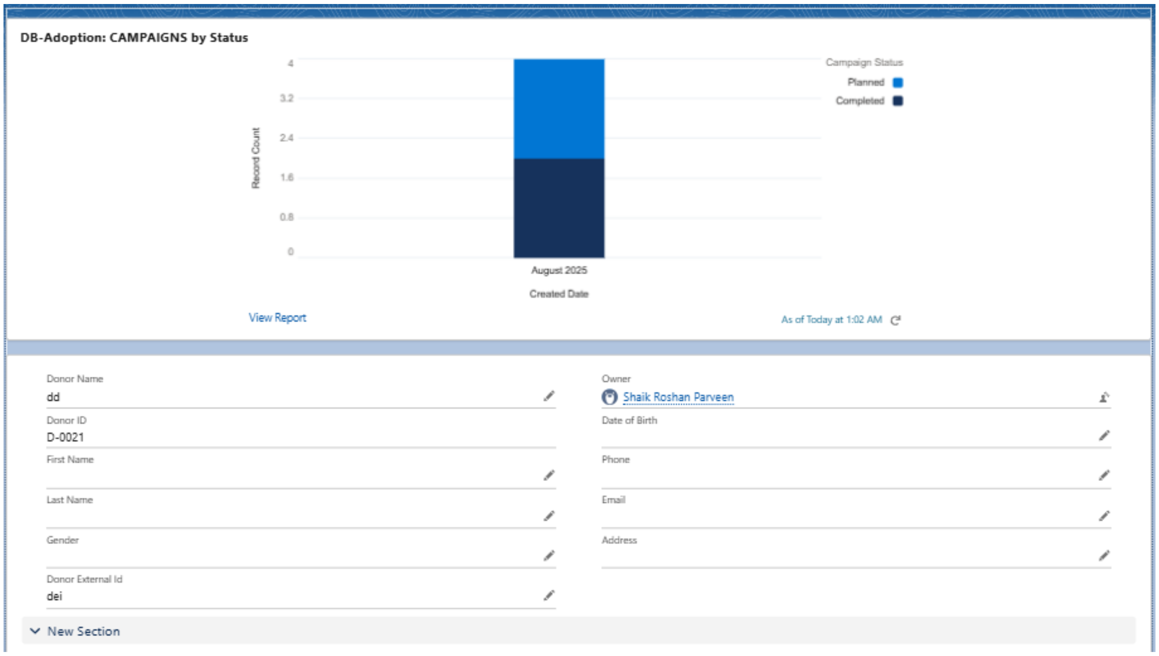
This ensures the CRM can **scale**, remain **data-consistent**, and handle **complex donor-request matching** beyond point-and-click automation.

# Phase 6: User Interface Development

In this phase, the focus was on designing and customizing the Salesforce user interface to make it intuitive, user-friendly, and aligned with the requirements of the Blood Request Management system. The following steps were undertaken:

## 1. Lightning App Builder

- Used **Lightning App Builder** to create custom pages and apps for the system.
- Developed a dedicated **Blood Request Management app** to provide users with quick access to blood requests, donors, and appointments.
- Configured **custom components** and **standard components** (like Lists, Charts, and Rich Text) to display relevant information dynamically.



Donor Name: dd

Donor ID: D-0021

First Name:

Last Name:

Gender:

Donor External Id: dei

Owner: Shaik Roshan Parveen

Date of Birth:

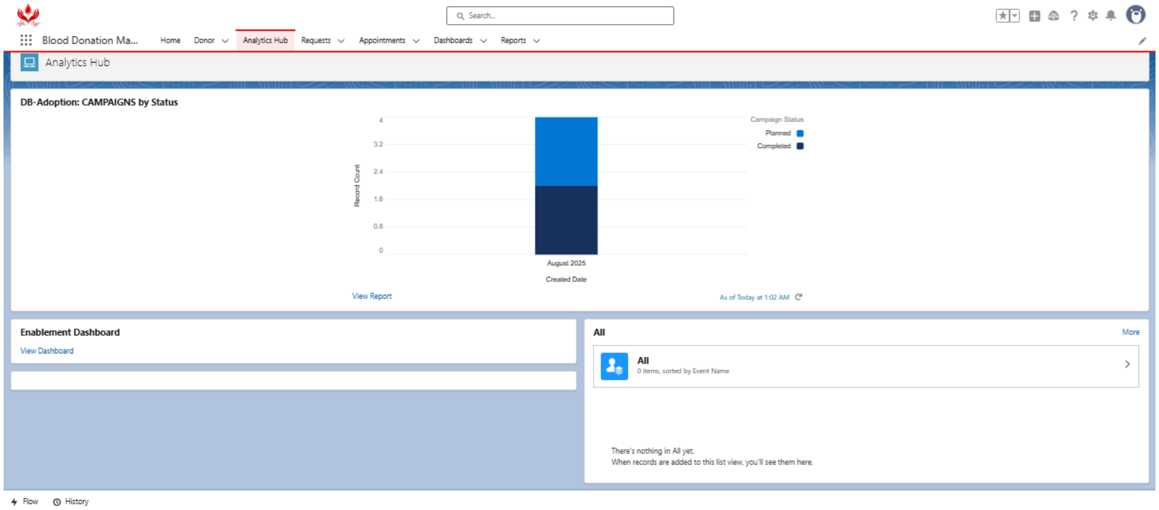
Phone:

Email:

Address:

## 2. Record Pages

- Customized **Record Pages** for `Request__c`, `Donor__c`, and `Appointment__c`.
- Ensured that fields, sections, and related lists are arranged logically for ease of data entry and review.
- Configured **conditional visibility** for components based on field values (e.g., only show appointment details if a request is in progress).



All

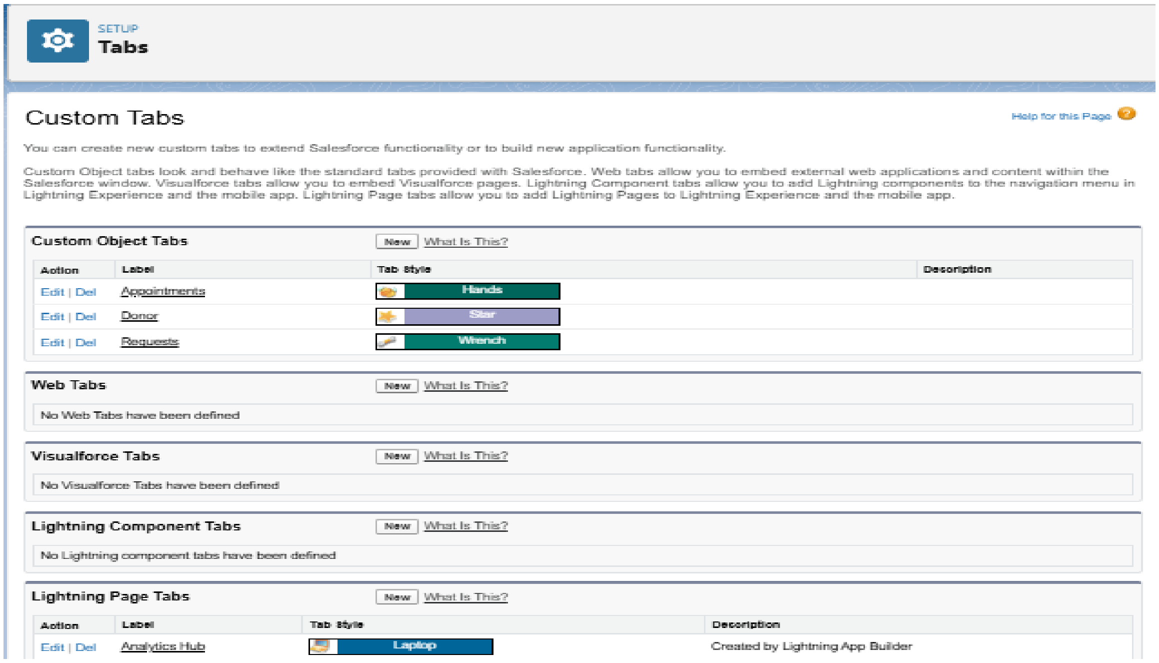
0 items, sorted by Event Name

There's nothing in All yet.

When records are added to this list view, you'll see them here.

## 3. Tabs

- Created custom **Tabs** for `Blood Requests`, `Donors`, and `Appointments` in the Lightning app.
- Enabled easy navigation for users, ensuring that frequently used objects were accessible from the app navigation bar.



SETUP
Tabs

Custom Tabs

Help for this Page

You can create new custom tabs to extend Salesforce functionality or to build new application functionality.

Custom Object tabs look and behave like the standard tabs provided with Salesforce. Web tabs allow you to embed external web applications and content within the Salesforce window. Visualforce tabs allow you to embed Visualforce pages. Lightning Component tabs allow you to add Lightning components to the navigation menu in Lightning Experience and the mobile app. Lightning Page tabs allow you to add Lightning Pages to Lightning Experience and the mobile app.

Custom Object Tabs — New — What Is This?

| Action | Label | Tab Style | Description |
|---|---|---|---|
| Edit \| Del | Appointments | Hands | |
| Edit \| Del | Donor | Star | |
| Edit \| Del | Requests | Wrench | |

Web Tabs — New — What Is This?

No Web Tabs have been defined

Visualforce Tabs — New — What Is This?

No Visualforce Tabs have been defined

Lightning Component Tabs — New — What Is This?

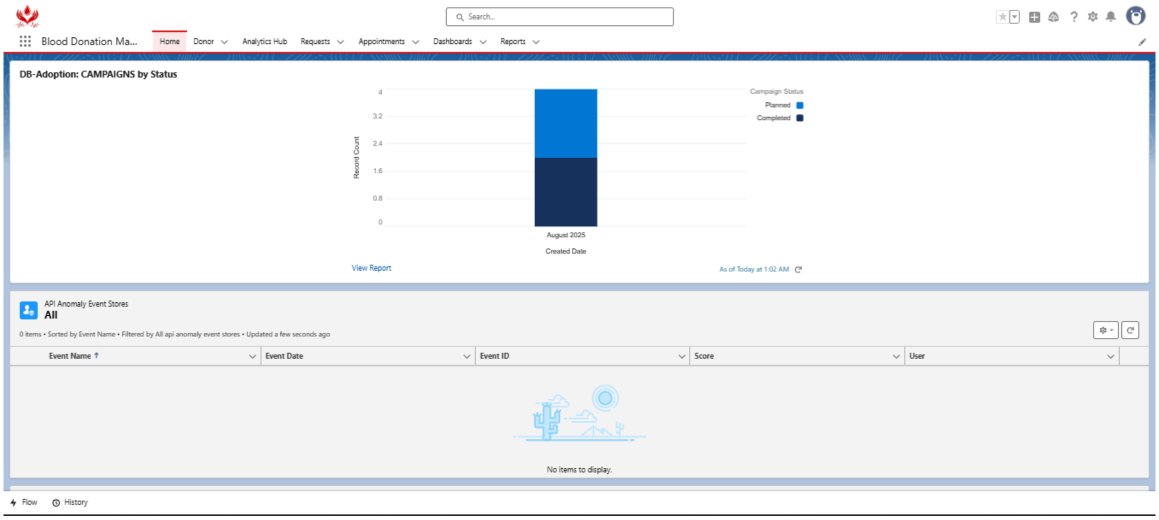
No Lightning component tabs have been defined

Lightning Page Tabs — New — What Is This?

| Action | Label | Tab Style | Description |
|---|---|---|---|
| Edit \| Del | Analytics Hub | Laptop | Created by Lightning App Builder |

# 4. Home Page Layouts

- Designed a **custom Home Page Layout** to provide users with an overview of important metrics.
- Added **dashboards, charts, and recent records lists** to display critical data such as total pending blood requests, donors with available units, and upcoming appointments.
- Ensured a responsive and user-friendly interface that supports quick decision-making.



DB-Adoption: CAMPAIGNS by Status

View Report

As of Today at 1:02 AM

API Anomaly Event Stores
All

0 items • Sorted by Event Name • Filtered by All api anomaly event stores • Updated a few seconds ago

| Event Name ↑ | Event Date | Event ID | Score | User |
|---|---|---|---|---|

No items to display.

# Phase 7: Integration & External Access

## 1. Named Credentials

- Store external system credentials (username/password, OAuth, etc.) securely.
- Simplifies authentication for API callouts.
- Avoids hardcoding credentials in Apex or flows.

---

## 2. External Services

- Connect Salesforce to external REST APIs declaratively.
- Automatically generate Apex actions from OpenAPI (Swagger) schemas.
- Can use in Flows, Process Builder, or Apex without manual coding.

---

## 3. Web Services (REST/SOAP)

- **REST/SOAP APIs** allow Salesforce to communicate with external systems.
- Use **Apex callouts** for sending/receiving data.
- Can expose Salesforce objects/data to external systems via **Apex REST/SOAP services**.

---

## 4. Callouts

- Make HTTP requests from Salesforce to external services (GET, POST, PUT, DELETE).
- Often used for real-time integrations with other apps or APIs.
- Can be synchronous (immediate response) or asynchronous (future methods, queues).

---

## 5. Platform Events

- Event-driven architecture within Salesforce.
- Publish and subscribe to events between Salesforce and external systems.
- Useful for real-time notifications, integrations, or workflow triggers.

---

## 6. Change Data Capture (CDC)

- Track changes (create, update, delete, undelete) in Salesforce objects.
- External systems or processes can subscribe to these changes.
- Enables real-time sync with external databases or apps.

---

## 7. Salesforce Connect

- Access external data in real-time without storing it in Salesforce.
- Use **External Objects** to map tables from other systems.
- Useful for large datasets or integrating with ERP/legacy systems.

---

## 8. API Limits

- Salesforce enforces limits on API calls (REST, SOAP, Bulk API, etc.).
- Important to monitor and optimize integration to avoid hitting limits.

---

## 9. OAuth & Authentication

- Securely authenticate external apps and users with Salesforce.
- Supports OAuth 2.0 flows, JWT, and connected apps.
- Often used with Named Credentials and External Services.

---

## 10. Remote Site Settings

- Add external URLs to allow Apex callouts to these sites.
- Prevents unauthorized access to external systems.
- Required for any callout to an endpoint outside Salesforce.

---

□ **Summary:**
This phase is all about **connecting Salesforce to other systems**, **sending/receiving data**, **tracking events**, and **maintaining secure authentication**. You can integrate in real-time, schedule data syncs, or expose Salesforce data externally.

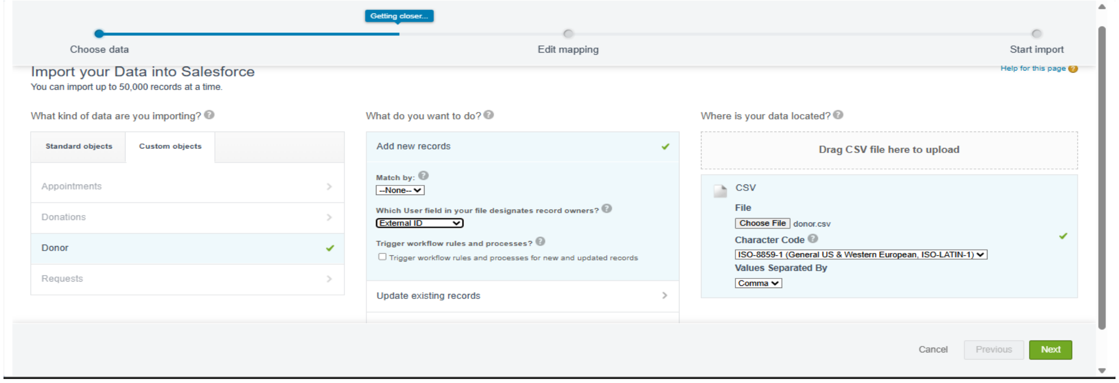
# Phase 8: Data Management & Deployment

In this phase, the focus was on managing data efficiently, ensuring data integrity, and deploying changes across environments. The following tools and functionalities were used:

---

## 1. Data Import Wizard

- **Purpose:** To import records such as Donor__c and Request__c into Salesforce.
- **Steps Taken:**
  1. Accessed **Setup → Data → Data Import Wizard**.
  2. Selected the object to import (e.g., Donor, Request).
  3. Uploaded the **prepared CSV file** containing donor and request data.
  4. Mapped CSV fields to Salesforce fields carefully to ensure accuracy.
  5. Reviewed and started the import process.
- **Outcome:** Successfully imported initial dataset for donors and requests, creating a foundation for testing and further operations.



Import your Data into Salesforce

You can import up to 50,000 records at a time.

What kind of data are you importing?

Donor ✓

What do you want to do?

Add new records ✓

Match by: --None--

Which User field in your file designates record owners? External ID

Trigger workflow rules and processes?

□ Trigger workflow rules and processes for new and updated records

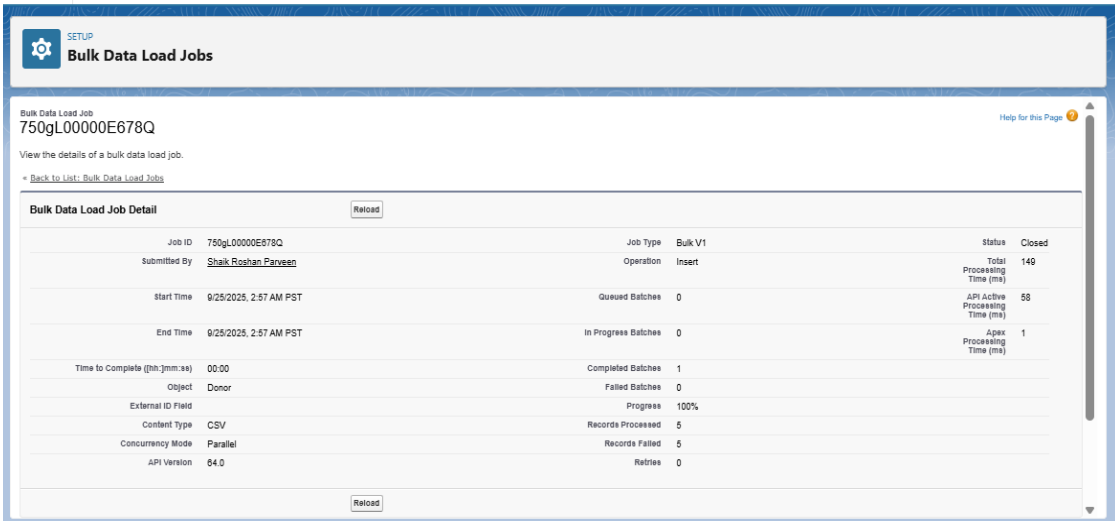
Where is your data located?

CSV ✓

File: donor.csv

Character Code: ISO-8859-1 (General US & Western European, ISO-LATIN-1)

Values Separated By: Comma



Bulk Data Load Jobs

Bulk Data Load Job

750gL00000E678Q

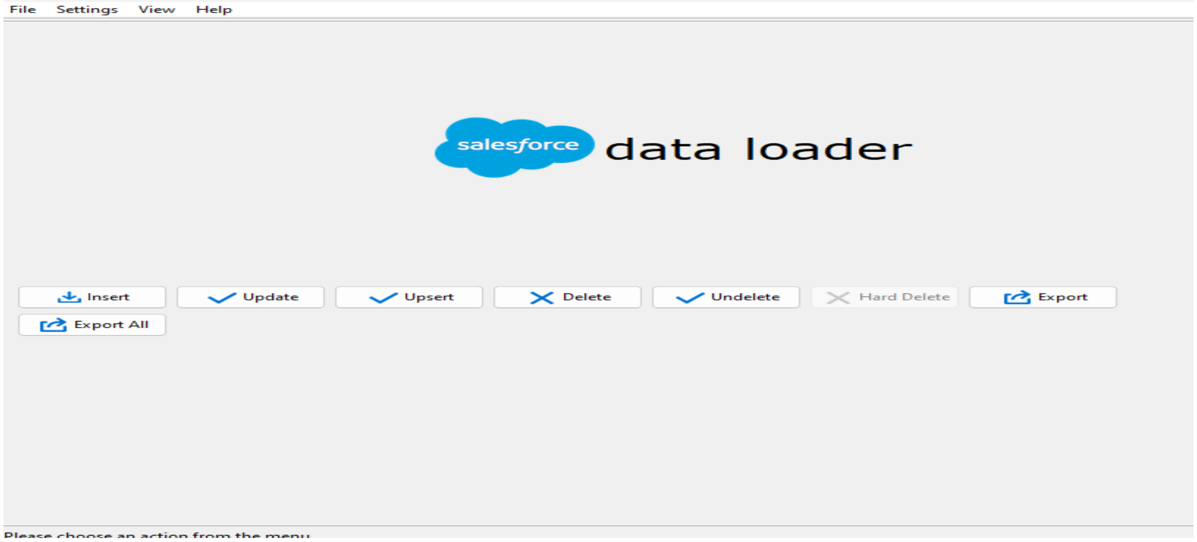
View the details of a bulk data load job.

Bulk Data Load Job Detail

| | | | | | |
|---|---|---|---|---|---|
| Job ID | 750gL00000E678Q | Job Type | Bulk V1 | Status | Closed |
| Submitted By | Shaik Roshan Parveen | Operation | Insert | Total Processing Time (ms) | 149 |
| Start Time | 9/25/2025, 2:57 AM PST | Queued Batches | 0 | API Active Processing Time (ms) | 58 |
| End Time | 9/25/2025, 2:57 AM PST | In Progress Batches | 0 | Apex Processing Time (ms) | 1 |
| Time to Complete ([hh:]mm:ss) | 00:00 | Completed Batches | 1 | | |
| Object | Donor | Failed Batches | 0 | | |
| External ID Field | | Progress | 100% | | |
| Content Type | CSV | Records Processed | 5 | | |
| Concurrency Mode | Parallel | Records Failed | 5 | | |
| API Version | 64.0 | Retries | 0 | | |

## 2. Data Loader

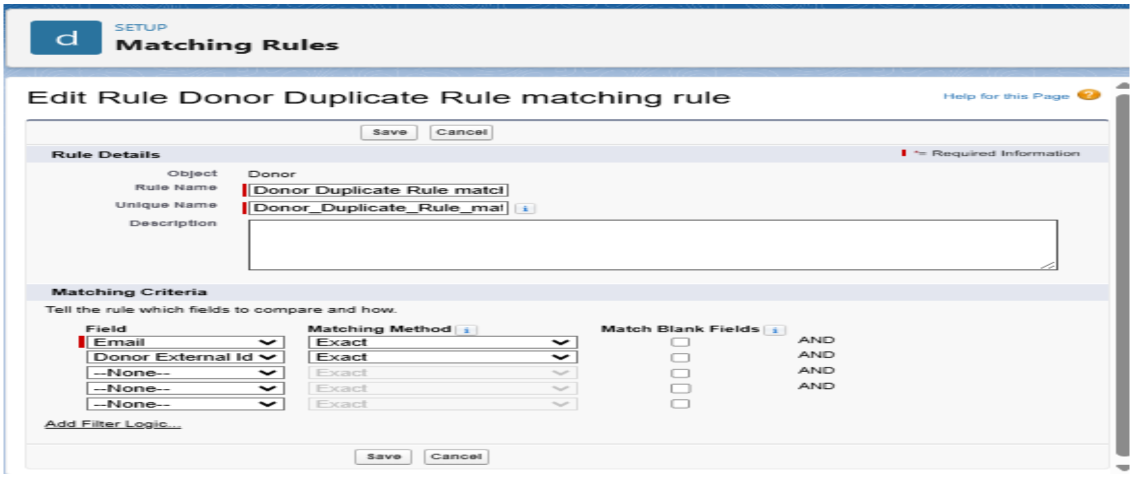
- **Purpose:** To perform large-scale data operations like insert, update, export, and delete.
- **Steps Taken:**
    1. Installed **Salesforce Data Loader** on the local machine.
    2. Logged in using Salesforce credentials.
    3. Selected the operation type (Insert, Update, Upsert).
    4. Chose the object (e.g., Donor__c, Request__c).
    5. Uploaded CSV file and mapped the fields.
    6. Executed the operation and reviewed success/failure files.
- **Outcome:** Efficiently handled bulk data operations, ensuring records were accurately updated and created.



Please choose an action from the menu

## 3. Duplicate Rules

- **Purpose:** To prevent duplicate records from being created and maintain data integrity.
- **Steps Taken:**
    1. Accessed **Setup** → **Duplicate Management** → **Duplicate Rules**.
    2. Created rules for `Donor__c` and `Request__c` objects.
    3. Configured **matching rules** based on unique identifiers like `Donor_External_Id__c` and `Request_External_Id__c`.
    4. Set the actions to **Alert or Block** duplicates during record creation or updates.
- **Outcome:** Ensured that duplicate donors and requests could not be created, maintaining clean and accurate data.



SETUP
Matching Rules

Edit Rule Donor Duplicate Rule matching rule

Help for this Page

Save Cancel

Rule Details — = Required Information

Object: Donor
Rule Name: Donor Duplicate Rule matcl
Unique Name: Donor_Duplicate_Rule_mal
Description:

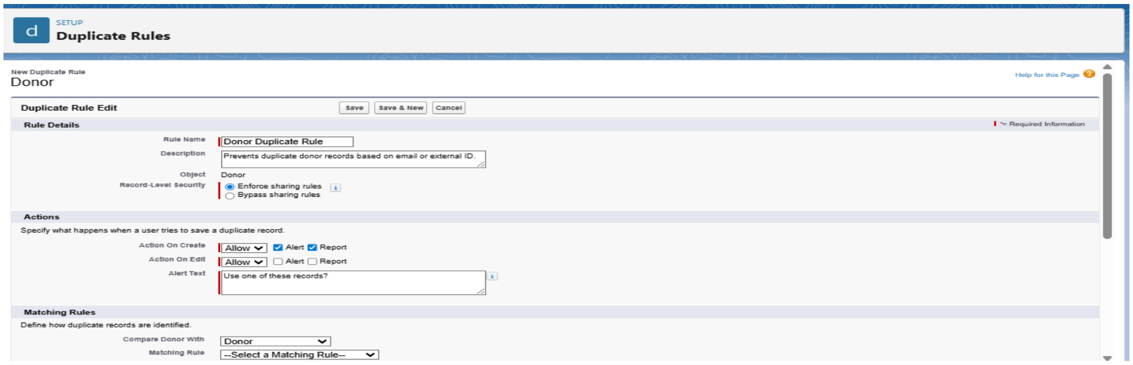
Matching Criteria

Tell the rule which fields to compare and how.

| Field | Matching Method | Match Blank Fields | |
|---|---|---|---|
| Email | Exact | □ | AND |
| Donor External Id | Exact | □ | AND |
| --None-- | Exact | □ | AND |
| --None-- | Exact | □ | AND |
| --None-- | Exact | □ | |

Add Filter Logic...

Save Cancel



SETUP
Duplicate Rules

Help for this Page

New Duplicate Rule
Donor

Duplicate Rule Edit — Save | Save & New | Cancel

Rule Details — = Required Information

Rule Name: Donor Duplicate Rule
Description: Prevents duplicate donor records based on email or external ID.
Object: Donor
Record-Level Security: ● Enforce sharing rules ○ Bypass sharing rules

Actions

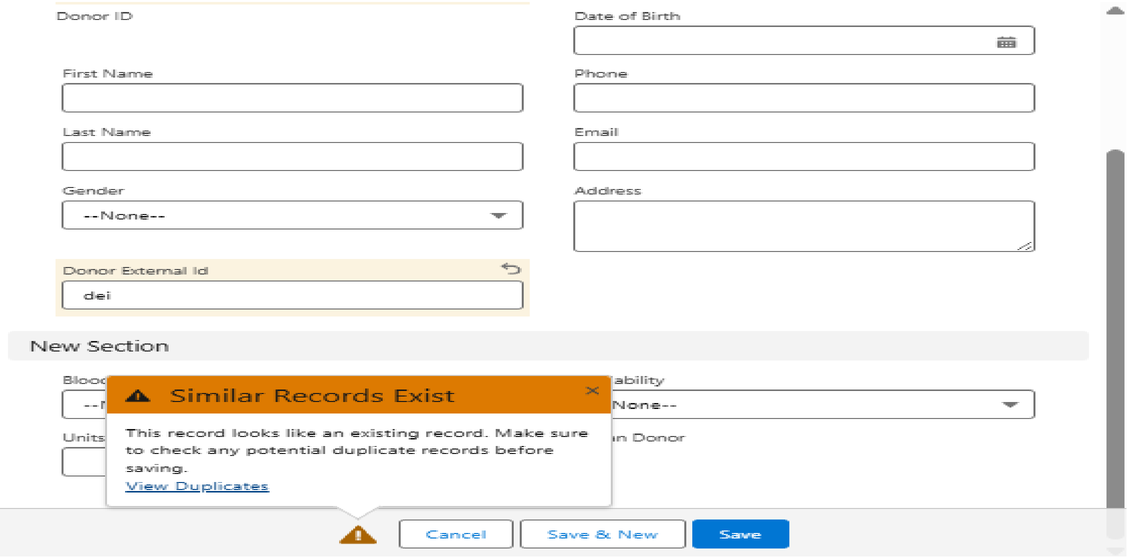
Specify what happens when a user tries to save a duplicate record.

Action On Create: Allow ☑ Alert ☑ Report
Action On Edit: Allow □ Alert □ Report
Alert Text: Use one of these records?

Matching Rules

Define how duplicate records are identified.

Compare Donor With: Donor
Matching Rule: --Select a Matching Rule--



Donor ID
Date of Birth
First Name
Phone
Last Name
Email
Gender: --None--
Address
Donor External Id: dei

New Section

Blood
Units
ability: None--
n Donor

Similar Records Exist

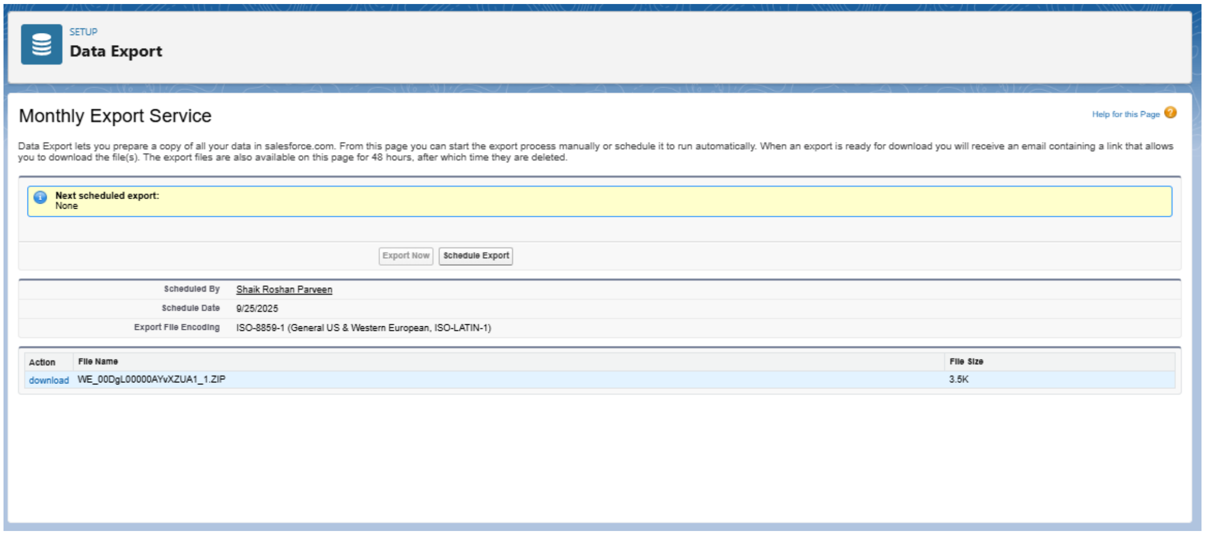
This record looks like an existing record. Make sure to check any potential duplicate records before saving.

View Duplicates

Cancel | Save & New | Save

# 4. Data Export & Backup

- **Purpose:** To back up Salesforce data for security, audit, and recovery purposes.
- **Steps Taken:**
    1. Accessed **Setup → Data → Data Export**.
    2. Selected objects to export (Donor__c, Request__c, Appointment__c).
    3. Chose **manual or scheduled export** (weekly or monthly).
    4. Downloaded the exported **.zip files** containing CSV files of all selected records.
- **Outcome:** Maintained regular backups of all Salesforce data to prevent loss and support future recovery needs.



Monthly Export Service

Data Export lets you prepare a copy of all your data in salesforce.com. From this page you can start the export process manually or schedule it to run automatically. When an export is ready for download you will receive an email containing a link that allows you to download the file(s). The export files are also available on this page for 48 hours, after which time they are deleted.

**Next scheduled export:**
None

| | |
|---|---|
| Scheduled By | Shaik Roshan Parveen |
| Schedule Date | 9/25/2025 |
| Export File Encoding | ISO-8859-1 (General US & Western European, ISO-LATIN-1) |

| Action | File Name | File Size |
|---|---|---|
| download | WE_00DgL00000AYvXZUA1_1.ZIP | 3.5K |

---

**Result of Phase 8:**
This phase ensured **accurate data import, bulk data handling, prevention of duplicates, and reliable backup mechanisms** for the Blood Request Management system. These processes enhanced data integrity, security, and operational efficiency.

# Phase 9: Reporting, Dashboards & Security Review

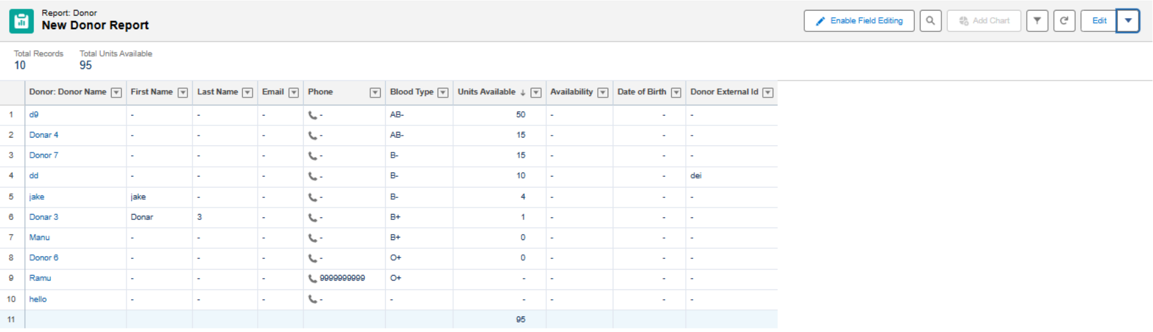
This phase focused on creating **reports, dashboards, and implementing security settings** in Salesforce for the Blood Request Management System.

---

## 1. Reports

Reports were created to track and analyze requests, donors, and appointments.

- **Tabular Report:**
  - **Steps:** Setup → Reports → New Report → Select Object → Add Columns & Filters → Save & Run.
  - **Use:** Simple list view of records like all active blood requests.

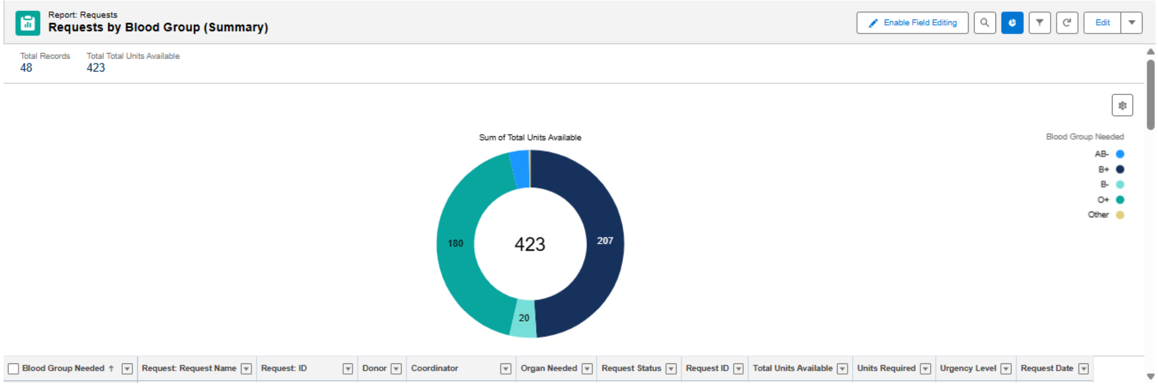


Report: Donor
**New Donor Report**

Total Records 10 Total Units Available 95

| | Donor: Donor Name | First Name | Last Name | Email | Phone | Blood Type | Units Available ↓ | Availability | Date of Birth | Donor External Id |
|---|---|---|---|---|---|---|---|---|---|---|
| 1 | d9 | - | - | - | ☎ - | AB- | 50 | - | - | - |
| 2 | Donar 4 | - | - | - | ☎ - | AB- | 15 | - | - | - |
| 3 | Donor 7 | - | - | - | ☎ - | B- | 15 | - | - | - |
| 4 | dd | - | - | - | ☎ - | B- | 10 | - | - | dei |
| 5 | jake | jake | - | - | ☎ - | B- | 4 | - | - | - |
| 6 | Donar 3 | Donar | 3 | - | ☎ - | B+ | 1 | - | - | - |
| 7 | Manu | - | - | - | ☎ - | B+ | 0 | - | - | - |
| 8 | Donor 6 | - | - | - | ☎ - | O+ | 0 | - | - | - |
| 9 | Ramu | - | - | - | ☎ 9999999999 | O+ | - | - | - | - |
| 10 | hello | - | - | - | ☎ - | - | - | - | - | - |
| 11 | | | | | | | 95 | | | |

- **Summary Report:**
  - **Steps:** Setup → Reports → New Report → Add Grouping (e.g., Blood Group) → Add Summary fields (Sum of Units).
  - **Use:** Show total units required per blood group.



Report: Requests
**Requests by Blood Group (Summary)**

Total Records 48 Total Total Units Available 423

| ☐ Blood Group Needed ↑ | Request: Request Name | Request: ID | Donor | Coordinator | Organ Needed | Request Status | Request ID | Total Units Available | Units Required | Urgency Level | Request Date |
|---|---|---|---|---|---|---|---|---|---|---|---|

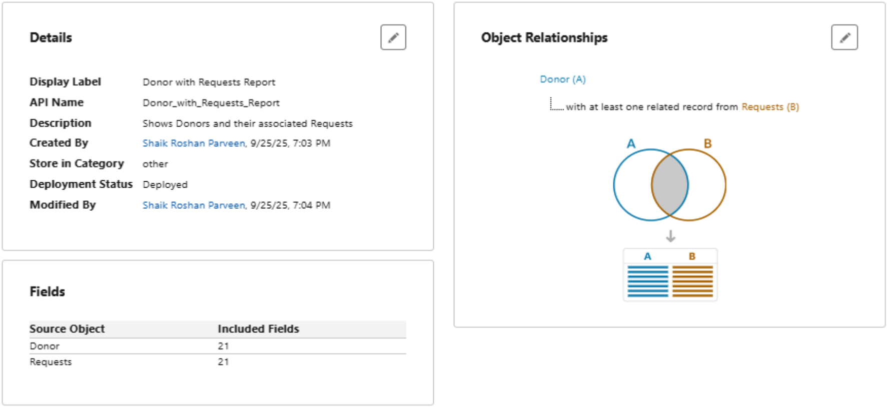
**Outcome:** Reports helped in monitoring blood request status and donor availability.

---

## 2. Report Types

Custom report types were created for combining data from multiple objects.

- **Steps:**
  Setup → Report Types → New Custom Report Type → Select Primary Object (Request__c) → Add Related Object (Donor__c) → Deploy → Save.



**Details**

| | |
|---|---|
| Display Label | Donor with Requests Report |
| API Name | Donor_with_Requests_Report |
| Description | Shows Donors and their associated Requests |
| Created By | Shaik Roshan Parveen, 9/25/25, 7:03 PM |
| Store in Category | other |
| Deployment Status | Deployed |
| Modified By | Shaik Roshan Parveen, 9/25/25, 7:04 PM |

**Fields**

| Source Object | Included Fields |
|---|---|
| Donor | 21 |
| Requests | 21 |

**Object Relationships**

Donor (A)

with at least one related record from Requests (B)

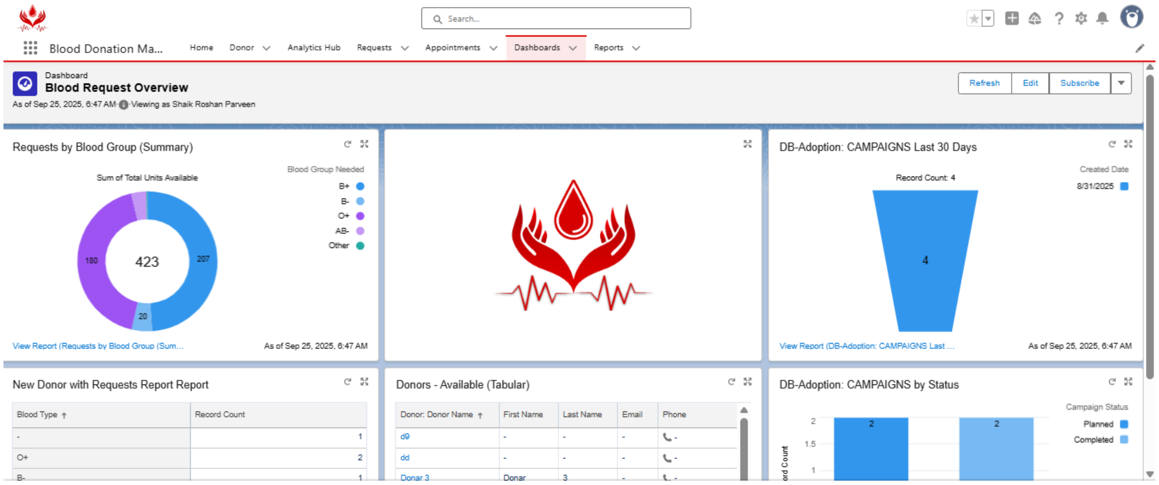
**Outcome:** Enabled cross-object reporting (e.g., Requests linked with Donors).

---

## 3. Dashboards

Dashboards were created to provide a visual overview of system data.

- **Steps:**
  Setup → Dashboards → New Dashboard → Add Components → Choose Source Report → Select Chart Type → Save.



Dashboard
**Blood Request Overview**
As of Sep 25, 2025, 6:47 AM · Viewing as Shaik Roshan Parveen

Requests by Blood Group (Summary)

View Report (Requests by Blood Group (Sum...
As of Sep 25, 2025, 6:47 AM

DB-Adoption: CAMPAIGNS Last 30 Days

View Report (DB-Adoption: CAMPAIGNS Last ...
As of Sep 25, 2025, 6:47 AM

New Donor with Requests Report Report

| Blood Type ↑ | Record Count |
|---|---|
| - | 1 |
| O+ | 2 |
| B- | 1 |

Donors - Available (Tabular)

| Donor: Donor Name ↑ | First Name | Last Name | Email | Phone |
|---|---|---|---|---|
| d9 | - | - | - | ☎ - |
| dd | - | - | - | ☎ - |
| Donar 3 | Donar | 3 | - | ☎ - |

DB-Adoption: CAMPAIGNS by Status

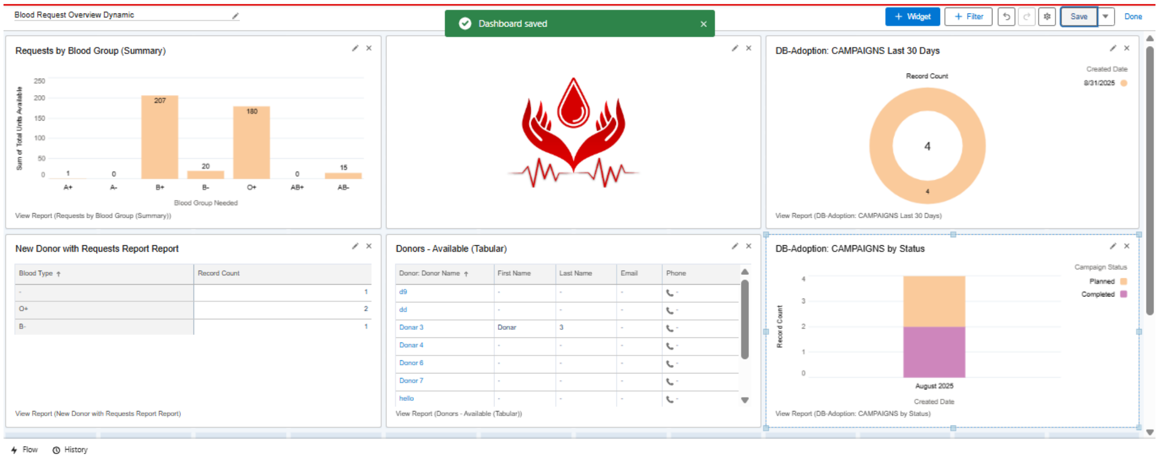
**Outcome:** Graphical view of requests by blood group, donor availability, and fulfilled requests.

---

# 4. Dynamic Dashboards

Dynamic dashboards were enabled to show data according to the logged-in user.

- **Steps:**
  In Dashboard Edit → View Dashboard As → Run as Logged-in User → Save.



Blood Request Overview Dynamic

Requests by Blood Group (Summary)

View Report (Requests by Blood Group (Summary))

DB-Adoption: CAMPAIGNS Last 30 Days

View Report (DB-Adoption: CAMPAIGNS Last 30 Days)

New Donor with Requests Report Report

| Blood Type ↑ | Record Count |
|---|---|
| - | 1 |
| O+ | 2 |
| B- | 1 |

View Report (New Donor with Requests Report Report)

Donors - Available (Tabular)

| Donor: Donor Name ↑ | First Name | Last Name | Email | Phone |
|---|---|---|---|---|
| d9 | - | - | - | ☎ - |
| dd | - | - | - | ☎ - |
| Donar 3 | Donar | 3 | - | ☎ - |
| Donar 4 | - | - | - | ☎ - |
| Donor 6 | - | - | - | ☎ - |
| Donor 7 | - | - | - | ☎ - |
| hello | - | - | - | ☎ - |

View Report (Donors - Available (Tabular))

DB-Adoption: CAMPAIGNS by Status

View Report (DB-Adoption: CAMPAIGNS by Status)

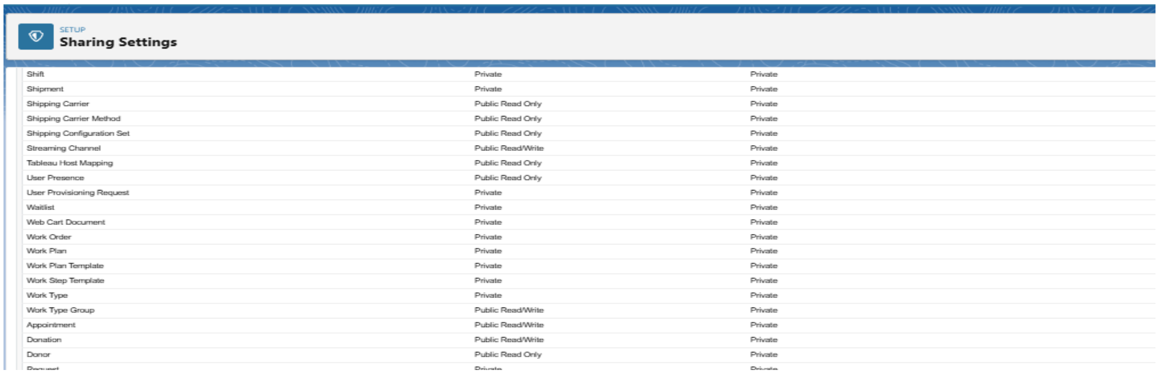
**Outcome:** Each coordinator/user sees only the data relevant to them.

---

## 5. Sharing Settings

Sharing rules and Organization-Wide Defaults (OWD) were set to manage record access.

- **Steps:**
  Setup → Sharing Settings → Configure OWD for each object → Add Sharing Rules → Save.



SETUP
Sharing Settings

| | | |
|---|---|---|
| Shift | Private | Private |
| Shipment | Private | Private |
| Shipping Carrier | Public Read Only | Private |
| Shipping Carrier Method | Public Read Only | Private |
| Shipping Configuration Set | Public Read Only | Private |
| Streaming Channel | Public Read/Write | Private |
| Tableau Host Mapping | Public Read Only | Private |
| User Presence | Public Read Only | Private |
| User Provisioning Request | Private | Private |
| Waitlist | Private | Private |
| Web Cart Document | Private | Private |
| Work Order | Private | Private |
| Work Plan | Private | Private |
| Work Plan Template | Private | Private |
| Work Step Template | Private | Private |
| Work Type | Private | Private |
| Work Type Group | Public Read/Write | Private |
| Appointment | Public Read/Write | Private |
| Donation | Public Read/Write | Private |
| Donor | Public Read Only | Private |
| Request | Private | Private |

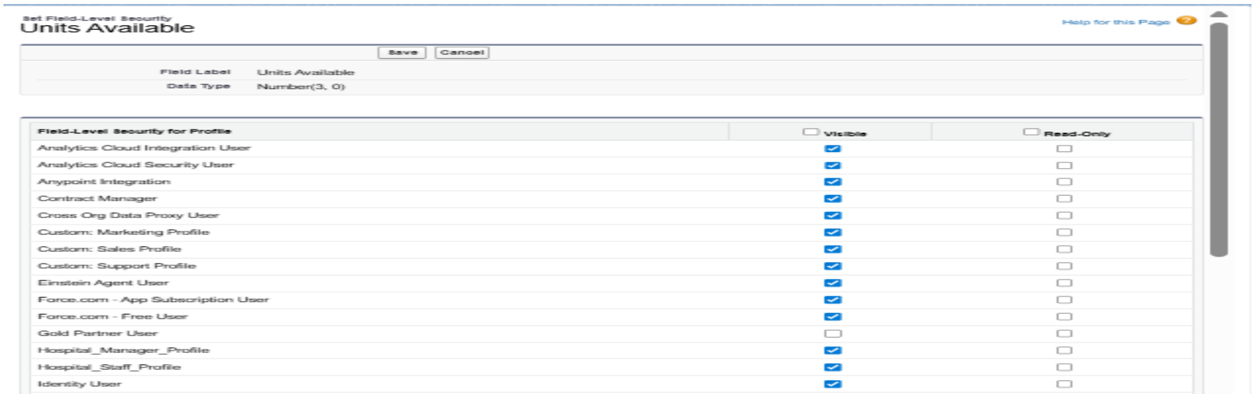
**Outcome:** Controlled data access while maintaining collaboration.

---

## 6. Field Level Security

Field visibility was controlled for different profiles.

- **Steps:**
  Setup → Object Manager → Select Object → Fields & Relationships → Select Field → Set Field-Level Security → Save.



Set Field-Level Security
Units Available

Field Label: Units Available
Data Type: Number(3, 0)

| Field-Level Security for Profile | □ Visible | □ Read-Only |
|---|---|---|
| Analytics Cloud Integration User | ☑ | □ |
| Analytics Cloud Security User | ☑ | □ |
| Anypoint Integration | ☑ | □ |
| Contract Manager | ☑ | □ |
| Cross Org Data Proxy User | ☑ | □ |
| Custom: Marketing Profile | ☑ | □ |
| Custom: Sales Profile | ☑ | □ |
| Custom: Support Profile | ☑ | □ |
| Einstein Agent User | ☑ | □ |
| Force.com - App Subscription User | ☑ | □ |
| Force.com - Free User | ☑ | □ |
| Gold Partner User | □ | □ |
| Hospital_Manager_Profile | ☑ | □ |
| Hospital_Staff_Profile | ☑ | □ |
| Identity User | ☑ | □ |

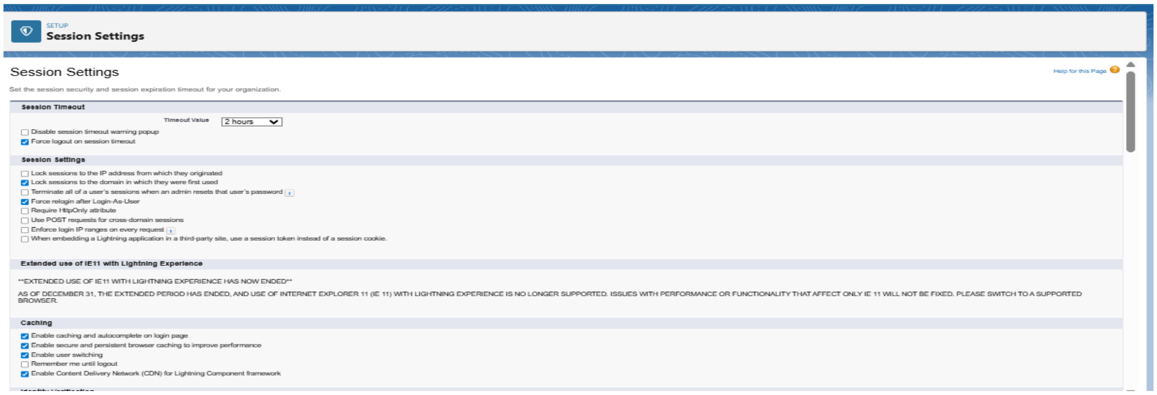
**Outcome:** Restricted sensitive fields (like Units_Available__c) to authorized users only.

---

## 7. Session Settings

Session security policies were applied.

- **Steps:**
  Setup → Session Settings → Configure Session Timeout, Security Levels → Save.



### Session Settings

Session Settings

Set the session security and session expiration timeout for your organization.

**Session Timeout**

Timeout Value: 2 hours

□ Disable session timeout warning popup
☑ Force logout on session timeout

**Session Settings**

□ Lock sessions to the IP address from which they originated
☑ Lock sessions to the domain in which they were first used
□ Terminate all of a user's sessions when an admin resets that user's password
☑ Force relogin after Login-As-User
□ Require HttpOnly attribute
□ Use POST requests for cross-domain sessions
□ Enforce login IP ranges on every request
□ When embedding a Lightning application in a third-party site, use a session token instead of a session cookie.

**Extended use of IE11 with Lightning Experience**

**EXTENDED USE OF IE11 WITH LIGHTNING EXPERIENCE HAS NOW ENDED**

AS OF DECEMBER 31, THE EXTENDED PERIOD HAS ENDED, AND USE OF INTERNET EXPLORER 11 (IE 11) WITH LIGHTNING EXPERIENCE IS NO LONGER SUPPORTED. ISSUES WITH PERFORMANCE OR FUNCTIONALITY THAT AFFECT ONLY IE 11 WILL NOT BE FIXED. PLEASE SWITCH TO A SUPPORTED BROWSER.

**Caching**

☑ Enable caching and autocomplete on login page
☑ Enable secure and persistent browser caching to improve performance
☑ Enable user switching
□ Remember me until logout
☑ Enable Content Delivery Network (CDN) for Lightning Component framework

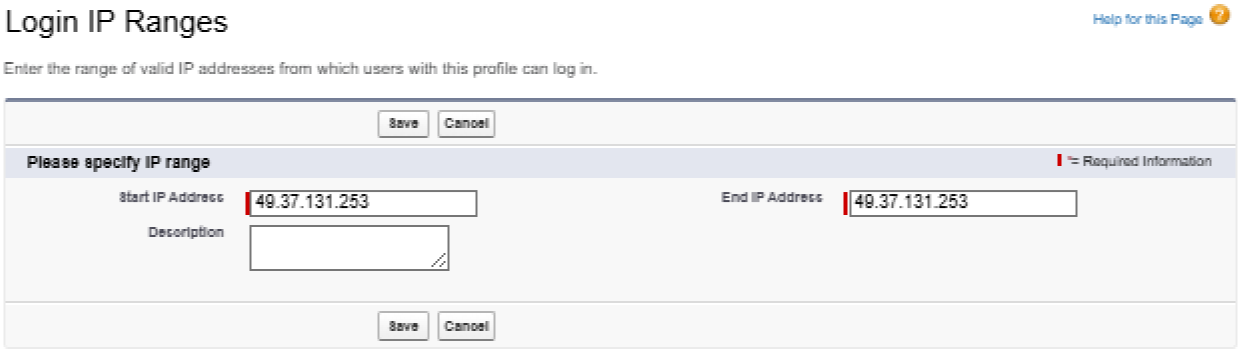
**Outcome:** Prevented unauthorized access and ensured session security.

---

## 8. Login IP Ranges

Login IP ranges were set to restrict access to trusted networks.

- **Steps:**
  Setup → Profiles → Select Profile → Login IP Ranges → Add New Range → Save.



Login IP Ranges

Enter the range of valid IP addresses from which users with this profile can log in.

**Please specify IP range**

Start IP Address: 49.37.131.253

End IP Address: 49.37.131.253

Description:

**Outcome:** Restricted users from logging in outside the defined IP ranges.

---

## □ Overall Result

- Reports & dashboards provide **clear visibility of blood requests and donor availability**.
- Dynamic dashboards allow **role-based insights**.
- Security settings (Sharing, FLS, Session, IP Ranges) ensure **data protection and controlled access**.