

# 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.

The screenshot shows the 'Company Information' page in the Salesforce Setup interface. The page title is 'Company Information' with a sub-header 'MediLink Team'. Below the title, it states 'The organization's profile is below.' and provides links for 'User Licenses (114)', 'Permission Set Licenses (114)', 'Feature Licenses (11)', and 'Usage-based Entitlements (114)'. The 'Organization Detail' section includes an 'Edit' button and a table of organization settings. The table is divided into two columns: 'Organization Detail' and 'Phone'. The 'Organization Detail' column includes fields for Organization Name, Primary Contact, Division, Address, Fiscal Year Starts In, Activate Multiple Currencies, Enable Data Translation, Newsletter, Admin Newsletter, Hide Notices About System Maintenance, Hide Notices About System Downtime, and Locale Formats. The 'Phone' column includes fields for Phone, Fax, Default Locale, Default Language, Default Time Zone, Currency Locale, Used Data Space, Used File Space, API Requests, Last 24 Hours, Streaming API Events, Last 24 Hours, Restricted Logins, Current Month, Salesforce.com Organization ID, Organization Edition, Instance, and Created By. The 'Created By' field shows 'OrgFarm EPIC, 8/31/2025, 8:52 AM' and the 'Modified By' field shows 'Shah Roshan Parveen, 9/16/2025, 10:48 PM'.

| Organization Detail                   |                                     |
|---------------------------------------|-------------------------------------|
| Organization Name                     | MediLink Team                       |
| Primary Contact                       | OrgFarm EPIC                        |
| Division                              |                                     |
| Address                               | United States                       |
| Fiscal Year Starts In                 | January                             |
| Activate Multiple Currencies          | <input type="checkbox"/>            |
| Enable Data Translation               | <input type="checkbox"/>            |
| Newsletter                            | <input checked="" type="checkbox"/> |
| Admin Newsletter                      | <input checked="" type="checkbox"/> |
| Hide Notices About System Maintenance | <input type="checkbox"/>            |
| Hide Notices About System Downtime    | <input type="checkbox"/>            |
| Locale Formats                        | ICU                                 |

| Phone                               |  |
|-------------------------------------|--|
| Phone                               |  |
| Fax                                 |  |
| Default Locale                      | English (India)                                |
| Default Language                    | English  |
| Default Time Zone                   | (GMT+05:30) India Standard Time (Asia/Kolkata) |
| Currency Locale                     | English (India) - INR                          |
| Used Data Space                     | 354 KB (7%) <a href="#">View</a>               |
| Used File Space                     | 17 KB (0%) <a href="#">View</a>                |
| API Requests, Last 24 Hours         | 0 (15,000 max)                                 |
| Streaming API Events, Last 24 Hours | 0 (10,000 max)                                 |
| Restricted Logins, Current Month    | 0 (0 max)                                      |
| Salesforce.com Organization ID      | 00DgL00000AYVXZ                                |
| Organization Edition                | Developer Edition                              |
| Instance                            | CAN68  |
| Created By                          | OrgFarm EPIC, 8/31/2025, 8:52 AM               |
| Modified By                         | Shah 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.

**Business Hours**

**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.

[Holidays](#)

**Business Hours Detail** [Edit](#)

| Business Hours Name | Default Business Hours (09:00-18:00 IST)   | Time Zone |          |        |                    |         |                    |           |                    |          |                    |        |                    |          |                    |  |
|---------------------|--|-----------|----------|--------|--------------------|---------|--------------------|-----------|--------------------|----------|--------------------|--------|--------------------|----------|--------------------|--|
| Business Hours      | <table><tr><td>Sunday</td><td>No Hours</td></tr><tr><td>Monday</td><td>9:00 AM to 6:00 PM</td></tr><tr><td>Tuesday</td><td>9:00 AM to 6:00 PM</td></tr><tr><td>Wednesday</td><td>9:00 AM to 6:00 PM</td></tr><tr><td>Thursday</td><td>9:00 AM to 6:00 PM</td></tr><tr><td>Friday</td><td>9:00 AM to 6:00 PM</td></tr><tr><td>Saturday</td><td>9:00 AM to 6:00 PM</td></tr></table> | Sunday    | No Hours | Monday | 9:00 AM to 6:00 PM | Tuesday | 9:00 AM to 6:00 PM | Wednesday | 9:00 AM to 6:00 PM | Thursday | 9:00 AM to 6:00 PM | Friday | 9:00 AM to 6:00 PM | Saturday | 9:00 AM to 6:00 PM | (GMT+05:30) India Standard Time (Asia/Kolkata) |
| Sunday              | No Hours   |           |          |        |                    |         |                    |           |                    |          |                    |        |                    |          |                    |  |
| Monday              | 9:00 AM to 6:00 PM   |           |          |        |                    |         |                    |           |                    |          |                    |        |                    |          |                    |  |
| Tuesday             | 9:00 AM to 6:00 PM   |           |          |        |                    |         |                    |           |                    |          |                    |        |                    |          |                    |  |
| Wednesday           | 9:00 AM to 6:00 PM   |           |          |        |                    |         |                    |           |                    |          |                    |        |                    |          |                    |  |
| Thursday            | 9:00 AM to 6:00 PM   |           |          |        |                    |         |                    |           |                    |          |                    |        |                    |          |                    |  |
| Friday              | 9:00 AM to 6:00 PM   |           |          |        |                    |         |                    |           |                    |          |                    |        |                    |          |                    |  |
| 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

[Edit](#)

**Holidays** [Add/Remove](#)

No records to display

[Back To Top](#)

Always show me fewer [▲](#) / [▼](#) more records per related list

---

### 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.

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.

Standard Fiscal Year

Custom Fiscal Year

Change Fiscal Year Period

Save

Cancel

Name

MediLink Team

Fiscal Year Start Month

January

Fiscal Year Is Based On

The ending month

The starting month

Save

Cancel

## 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 User

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

New User

Reset Password(s)

Add Multiple Users

| Action  | Full Name             | Alias     | Username   | Role             | Active | Profile                          |
|---|-----------------------|-----------|--|------------------|--------|----------------------------------|
| <input type="checkbox"/> <a href="#">Edit</a> <a href="#">Login</a> | Admin2_System         | sysadmin2 | sysadmin2roshan@example.com                                  | Org Admin        | ✓      | System Administrator             |
| <input type="checkbox"/> <a href="#">Edit</a>                       | Chatter Expert        | Chatter   | chatter.00d900000ayvzua1.cudhny41yrbu@chatter.salesforce.com |                  | ✓      | Chatter Free User                |
| <input type="checkbox"/> <a href="#">Edit</a> <a href="#">Login</a> | EPIC_OrgFarm          | OEPIE     | epic.2901ce800c2d@orgfarm.salesforce.com                     |                  | ✓      | System Administrator             |
| <input type="checkbox"/> <a href="#">Edit</a> <a href="#">Login</a> | Manager_Hospital      | hmanager  | hospital.managerroshan@example.com                           | Hospital Manager | ✓      | Hospital Manager Profile         |
| <input type="checkbox"/> <a href="#">Edit</a>                       | Parveen, Shaik Roshan | sha       | shaikroshanparveen818@qaenforce.com                          |                  | ✓      | System Administrator             |
| <input type="checkbox"/> <a href="#">Edit</a> <a href="#">Login</a> | Staff_Hospital        | hstaff    | hospital.staffroshan@example.com                             | Hospital Staff   | ✓      | Hospital Staff Profile           |
| <input type="checkbox"/> <a href="#">Edit</a>                       | User Integration      | integ     | integration@00d900000ayvzua1.com                             |                  | ✓      | Analytics Cloud Integration User |
| <input type="checkbox"/> <a href="#">Edit</a>                       | User Security         | sec       | insightssecurity@00d900000ayvzua1.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

## 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).

The screenshot shows the Salesforce 'Profiles' page in the Setup menu. The profile being viewed is 'Hospital\_Manager\_Profile'. The page includes a 'Profile Detail' section with fields for Name, User License, Description, Created By, and Modified By. Below this is the 'Page Layouts' section, which displays a table of assignments for various standard object layouts. The table has two columns: 'Standard Object Layouts' and 'Page Layouts'. The 'Standard Object Layouts' column lists various layouts like Global, Email Application, Home Page Layout, Account, Alternative Payment Method, and Appointment Invitation. The 'Page Layouts' column lists the assigned layouts for each, such as 'Global Layout', 'Not Assigned', 'Home Page Default', 'Account Layout', 'Alternative Payment Method Layout', and 'Appointment Invitation Layout'. Each layout name is followed by a '(View Assignment)' link. The 'Global Layout' is assigned to the 'Global' standard object layout. The 'Not Assigned' status is shown for the 'Email Application' standard object layout. The 'Home Page Default' is assigned to the 'Home Page Layout'. The 'Account Layout' is assigned to the 'Account' standard object layout. The 'Alternative Payment Method Layout' is assigned to the 'Alternative Payment Method' standard object layout. The 'Appointment Invitation Layout' is assigned to the 'Appointment Invitation' standard object layout. The 'Location Group Assignment' is assigned to the 'Location Group Assignment Layout'. The 'Macro' is assigned to the 'Macro Layout'. The 'Object Milestone' is assigned to the 'Object Milestone Layout'. The 'Operating Hours' is assigned to the 'Operating Hours Layout'. The 'Opportunity' is assigned to the 'Opportunity Layout'. The 'Opportunity Product' is assigned to the 'Opportunity Product Layout'.

**Profile Detail**

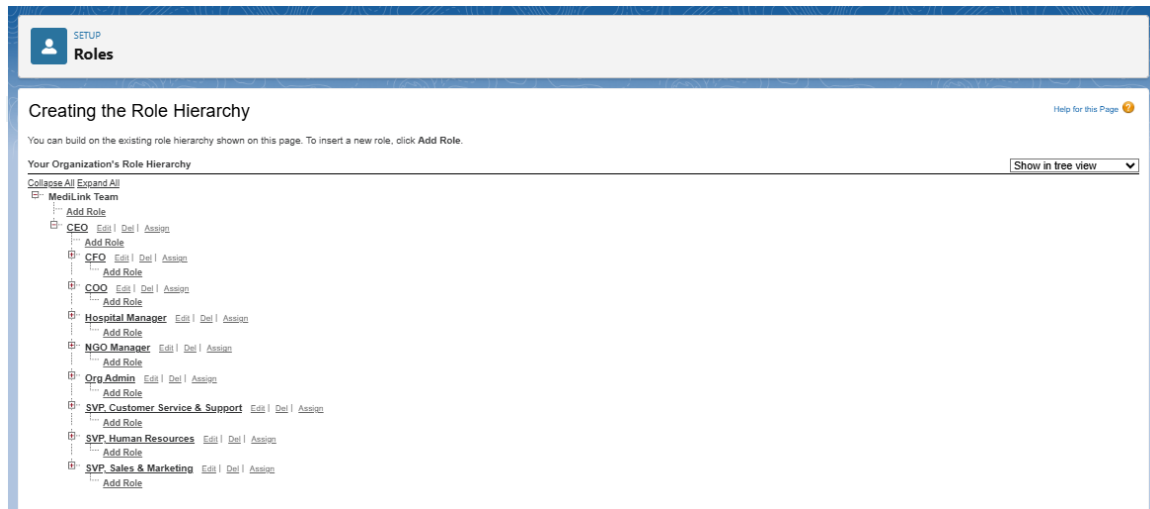
| Name                     | User License | Description | Created By                               | Modified By                              |
|--------------------------|--------------|-------------|--|--|
| Hospital_Manager_Profile | Salesforce   |             | Shaik Roshan Parveen, 9/13/2025, 4:56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

**Page Layouts**

| Standard Object Layouts    | Page Layouts  |
|----------------------------|---|
| Global                     | Global Layout (View Assignment)                     |
| Email Application          | Not Assigned (View Assignment)                      |
| Home Page Layout           | Home Page Default (View Assignment)                 |
| Account                    | Account Layout (View Assignment)                    |
| Alternative Payment Method | Alternative Payment Method Layout (View Assignment) |
| Appointment Invitation     | Appointment Invitation Layout (View Assignment)     |
| Location Group Assignment  | Location Group Assignment Layout (View Assignment)  |
| Macro                      | Macro Layout (View Assignment)                      |
| Object Milestone           | Object Milestone Layout (View Assignment)           |
| Operating Hours            | Operating Hours Layout (View Assignment)            |
| Opportunity                | Opportunity Layout (View Assignment)                |
| Opportunity Product        | Opportunity Product Layout (View Assignment)        |

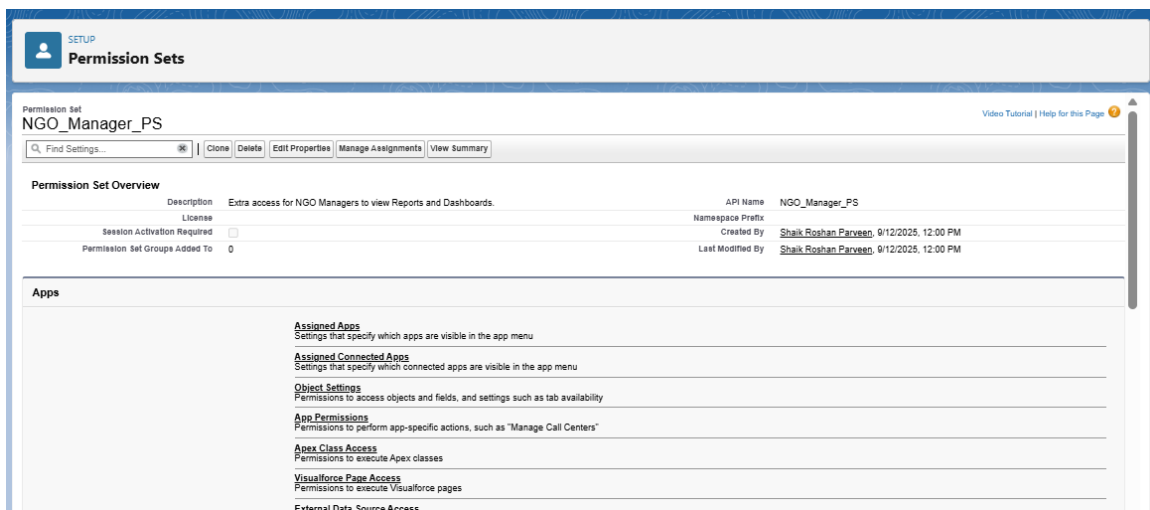
## 7. Roles & Role Hierarchy

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



## 8. Permission Sets (Optional)

- Setup → Quick Find → Permission Sets → New.
- Created:
  - NGO\_Manager\_PS
  - Donor\_Portal\_Access\_PS
- Saved (assignments not done yet).



## 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

| SETUP Sharing Settings     |                   |         |   |
|----------------------------|-------------------|---------|---|
| 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 Hospital
- Saved.

| SETUP Sharing Settings                  |  |                      |   |
|---|--|----------------------|---|
| No sharing rules specified.             |  |                      |   |
| <b>Work Step Template Sharing Rules</b> |  | New Recalculate      | Work Step Template Sharing Rules Help ? |
| No sharing rules specified.             |  |                      |   |
| <b>Work Type Sharing Rules</b>          |  | New Recalculate      | Work Type Sharing Rules Help ?          |
| No sharing rules specified.             |  |                      |   |
| <b>Work Type Group Sharing Rules</b>    |  | New Recalculate      | Work Type Group Sharing Rules Help ?    |
| No sharing rules specified.             |  |                      |   |
| <b>Appointment Sharing Rules</b>        |  | New Recalculate      | Appointment Sharing Rules Help ?        |
| No sharing rules specified.             |  |                      |   |
| <b>Donor Sharing Rules</b>              |  | New Recalculate      | Donor Sharing Rules Help ?              |
| No sharing rules specified.             |  |                      |   |
| <b>Request Sharing Rules</b>            |  | New Recalculate      | Request Sharing Rules Help ?            |
| 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.

The screenshot shows the 'Login Access Policies' setup page in Salesforce. The page title is 'Login Access Policies' with a 'Help for this Page' link. Below the title, it says 'Control which support organizations your users can grant login access to.' There is a 'Manage Support Options' section with 'Save' and 'Cancel' buttons. Under 'Manage Support Options', there is a 'Setting' section with 'Enabled' status and a checkbox for 'Administrators Can Log in as Any User' which is checked. Below this is a table with columns: 'Support Organization', 'Packages', 'Available to Users', and 'Available to Administrators Only'. The table has two rows: 'Salesforce.com Support' and 'Q Branch HQ Support'. The 'Salesforce.com Support' row has a radio button selected under 'Available to Users' and an empty radio button under 'Available to Administrators Only'. The 'Q Branch HQ Support' row has 'VideoFileViewer' in the 'Packages' column, 'Not Available' in the 'Available to Users' column, and a radio button selected under 'Available to Administrators Only'. There are 'Save' and 'Cancel' buttons at the bottom of the table.

| Support Organization   | Packages        | Available to Users               | Available to Administrators Only |
|------------------------|-----------------|----------------------------------|----------------------------------|
| Salesforce.com Support |                 | <input checked="" type="radio"/> | <input type="radio"/>            |
| Q Branch HQ Support    | VideoFileViewer | Not Available                    | <input checked="" type="radio"/> |

---

### □ 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.

SETUP > OBJECT MANAGER

**Donor**

Details

**Fields & Relationships**

15 items. Sorted by Field Label

Quick Find New Deleted Fields Field Dependencies Set History Tracking

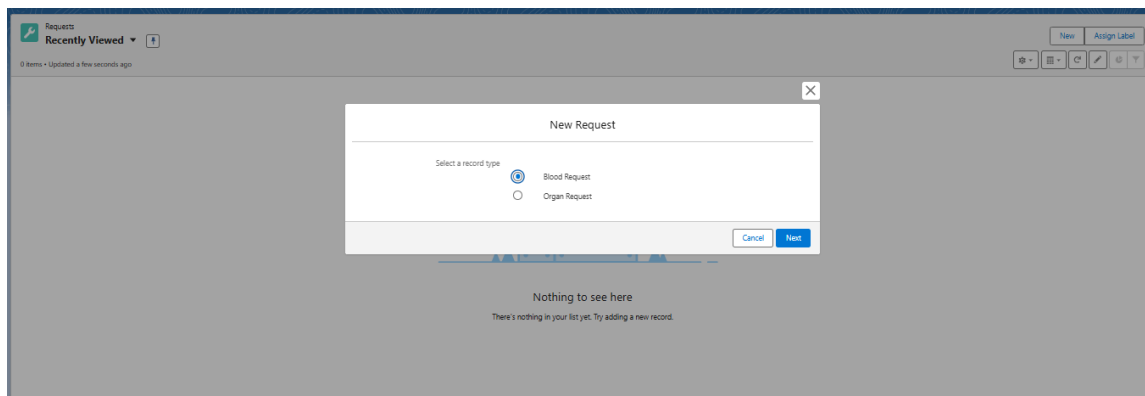
| FIELD LABEL   | FIELD NAME       | DATA TYPE      | CONTROLLING FIELD | INDEXED |
|---------------|------------------|----------------|-------------------|---------|
| Address       | Address__c       | Text Area(255) |                   |         |
| Availability  | Availability__c  | Picklist       |                   |         |
| 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.



## 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

Request ID

Blood Group Needed

--None--

Donor

Search Donor...

Owner

Shaik Roshan Parveen

Request Date

Urgency Level

--None--

Request Status

--None--

Requested By

Cancel

Save & New

Save

## New Request: Organ Request

\* = Required Information

### Information

\* Request Name

Request ID

Organ Needed

--None--

Donor

Search Donor...

Owner

 Shaik Roshan Parveen

Request Date

Urgency Level

--None--

Request Status

--None--

Requested By

Cancel

Save & New

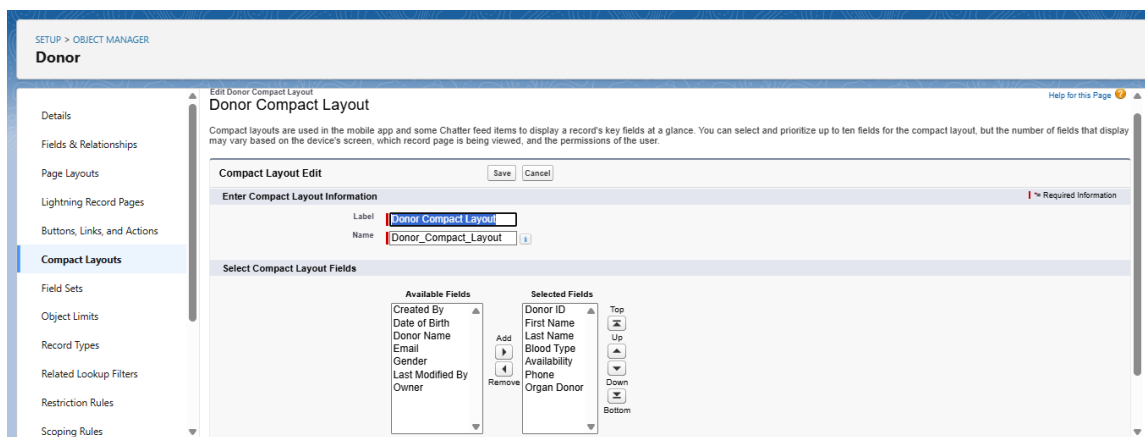
Save

## 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.



The screenshot shows the 'Edit Donor Compact Layout' interface in Salesforce. The left sidebar contains navigation links: Details, Fields & Relationships, Page Layouts, Lightning Record Pages, Buttons, Links, and Actions, Compact Layouts (selected), Field Sets, Object Limits, Record Types, Related Lookup Filters, Restriction Rules, and Scoping Rules. The main content area is titled 'Donor Compact Layout' and includes a description: '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.' Below this, there are two sections: 'Enter Compact Layout Information' and 'Select Compact Layout Fields'. The 'Enter Compact Layout Information' section has fields for 'Label' (Donor Compact Layout) and 'Name' (Donor\_Compact\_Layout). The 'Select Compact Layout Fields' section has two columns: 'Available Fields' and 'Selected Fields'. The 'Available Fields' column lists: Created By, Date of Birth, Donor Name, Email, Gender, Last Modified By, and Owner. The 'Selected Fields' column lists: Donor ID, First Name, Last Name, Blood Type, Availability, Phone, and Organ Donor. There are 'Add' and 'Remove' buttons between the columns, and 'Top', 'Up', 'Down', and 'Bottom' buttons on the right side of the 'Selected Fields' column.

Donor D-0003

First Name: Phool, Last Name: Kumari, Blood Type: O+, Availability: Available, Phone: (987) 654-3210, Organ Donor: [ ]

Related Details:

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

New Section:

|             |     |              |           |
|-------------|-----|--------------|-----------|
| Blood Type  | O+  | Availability | Available |
| Organ Donor | [ ] |              |           |

Created By: Sheik, Roshan Parveen, 9/16/2023, 9:24 PM

Last Modified By: Sheik, Roshan Parveen, 9/16/2023, 9:24 PM

Activity:

Filters: All time • All activities • All types

Upcoming & Overdue:

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

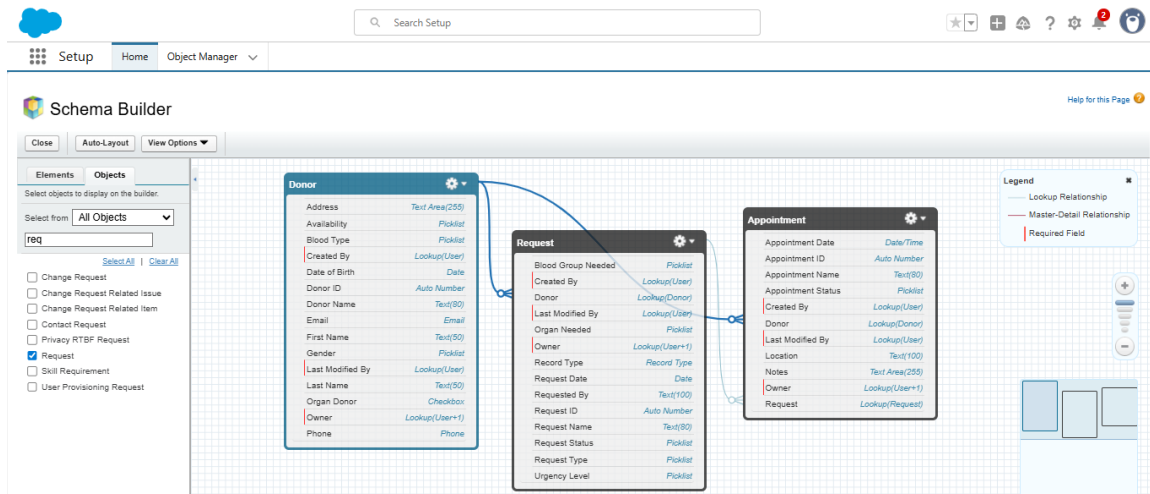
No past activity. Past meetings and tasks marked as done show up here.

## 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.



## 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`



**New Donor**

\* = Required Information

**Information**

\* Donor Name: Rohit Sharma

Donor ID: [Empty]

First Name: Rohit

Last Name: Sharma

Gender: Male

Date of Birth: 9/20/2023  
Donor must be at least 18 years old to register.

Phone: 9876543210

Email: shaikroshanparveen@gmail.com

Address: vijayawada

**New Section**

Blood Type: O+

**We hit a snag.**

Review the following fields

- Date of Birth

ability: available

Organ Donor: [Empty]

Cancel Save & New Save

## 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

Appointment ID: [Empty]

**Appointment Date**

Date: 9/17/2025  
Time: 12:00 PM  
Appointment date cannot be in the past.

Appointment Status: --None--

**Owner**

Shaik Roshan Parveen

**Location**

Vijayawada

**Donor**

hello

**Request**

Harry

**Notes**

**We hit a snag.**

Review the following fields

- Appointment Date

Cancel Save & New Save

### 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.

The screenshot shows a donor registration form with the following fields and values:

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

Below the form, there is a "New Section" containing:

- Blood Type:** A dropdown menu showing "--None--" with a red border and a red error message below it: "Blood Group must be selected."
- Availability:** A dropdown menu showing "Available".
- Organ Donor:** An unchecked checkbox.
- Created By:** Shaik Roshan Parveen
- Modified By:** Shaik Roshan Parveen, 9/20/2025, 9:08 PM

A red error dialog box is displayed in the center, stating "We hit a snag." and "Review the following fields" with a list containing "Blood Type". At the bottom of the form, there are three buttons: "Cancel", "Save & New", and "Save".

## 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  $\geq$  Required Units  $\rightarrow$  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  $\rightarrow$ 
    - 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.

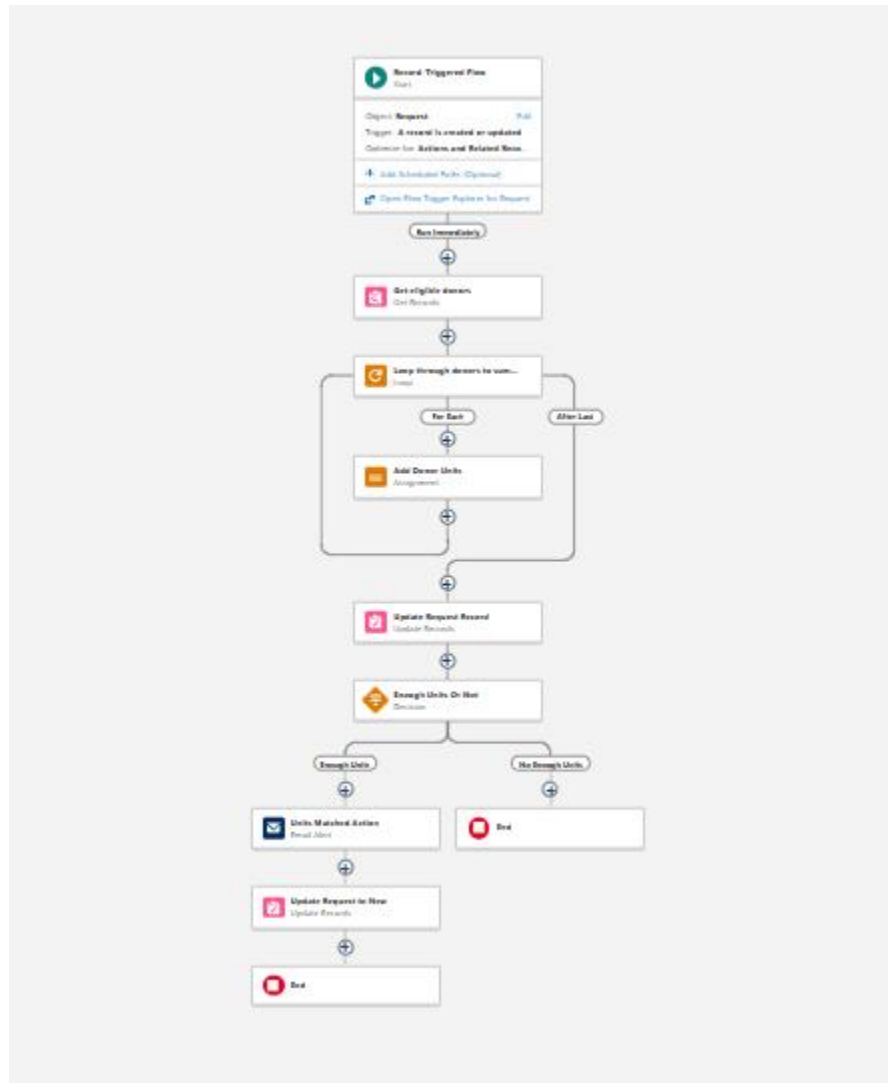


Fig: Total Units Available

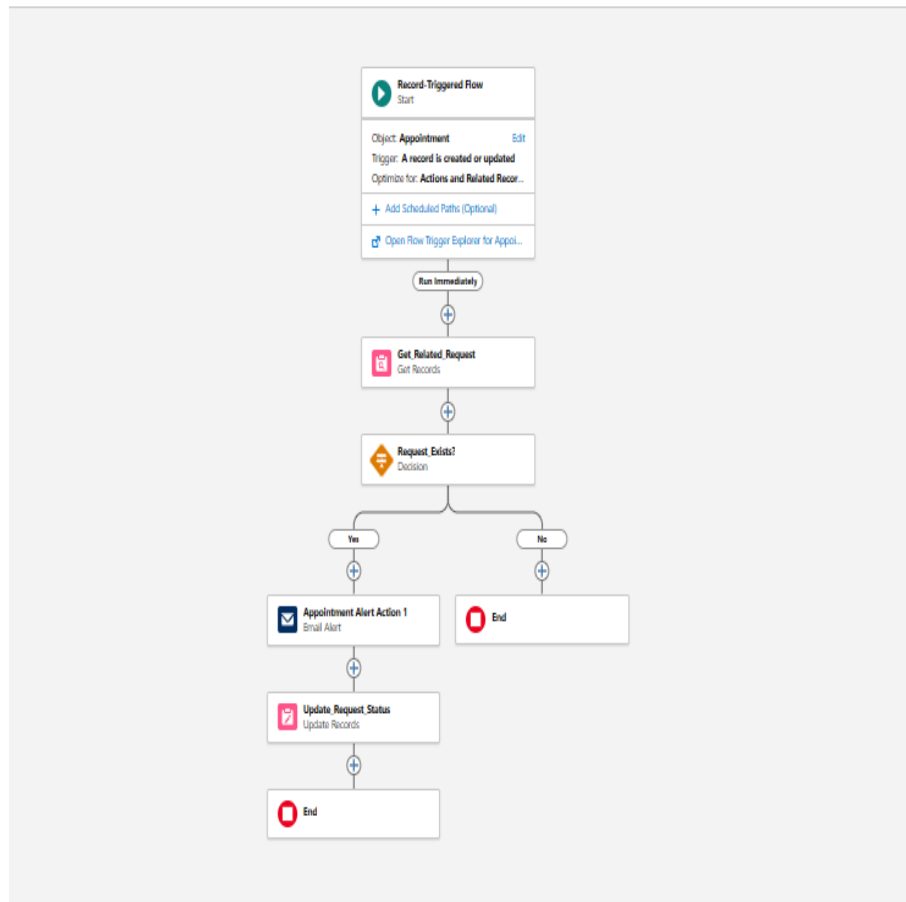


Fig: Appointment Fulfillment

---

## 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 fqrulm4qvtikn.gl-ayvxua1.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.

Report not spam



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 Spam x



**Shaik Roshan Parveen** via yda7yH9kcdv9.gl-ayvxua1.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.

Report not spam



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  
Fulfilled

Requested By

Blood Group Needed  
O+

Organ Needed

Related

Details

Request Name  
req1

Request ID  
R-0059


Blood Group Needed  
O+


Donor

Units Required ⓘ  
5

Total Units Available  
0

Coordinator

Created By  
 Shaik Roshan Parveen, 9/23/2025, 4:03 AM


Owner  
 Shaik Roshan Parveen

Request Date

Urgency Level

Request Status  
Fulfilled

Requested By

Last Modified By  
 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

Approved

Submitter

Shaik Roshan Parveen

Date Submitted

Sep 21, 2025

Actual Approver

Shaik Roshan Parveen

Assigned To

Shaik Roshan Parveen

Details

Approval Details

Request Name

[New Approval](#)

Owner

Shaik Roshan Parveen

No Comments

## Approval Request Spam x



**Shaik Roshan Parveen** via [k475n87lm97j5u.gl-ayvxua1.can98.bnc.salesforce.com](mailto:k475n87lm97j5u.gl-ayvxua1.can98.bnc.salesforce.com)  
to me ▾

10:27 AM (13 hours ago)

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

Report not spam

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

Please click this link to approve or reject this record.

Thank you,  
Salesforce

## Request Approved Spam x



**Shaik Roshan Parveen** via [7wixikgqxhukksam.uedrgn0irvohxryr.6fk9o.gl-ayvxua1.can98.bnc.salesforce.com](mailto:7wixikgqxhukksam.uedrgn0irvohxryr.6fk9o.gl-ayvxua1.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.

Report not spam

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.

```
File • Edit • Debug • Test • Workspace • Help • < • >
BloodRequestHandler.apxc • BloodRequestTrigger.apxt • BloodRequestTriggerHandler.apxc • DailyBloodRequestScheduler.apxc
Code Coverage: None • API Version: 64 • Go To

1 public class BloodRequestHandler {
2
3     // Static variable to prevent recursion
4     public static Boolean isProcessing = false;
5
6     public static void processRequests(List<Request__c> reqList) {
7
8         // Prevent recursion
9         if(isProcessing) return;
10        isProcessing = true;
11
12        if(reqList == null || reqList.isEmpty()) return;
13
14        // Fetch fresh Request records from DB
15        List<Request__c> freshRequests = [
16            SELECT Id, Name, Blood_Group_Needed__c, Units_Required__c, Request_Status__c, Total_Units_Available__c
17            FROM Request__c
18            WHERE Id IN :reqList
19        ];
20
21        // Collect all blood types needed
22        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.

```
BloodRequestHandler.apxc • BloodRequestTrigger.apxt • BloodRequestTriggerHandler.apxc • DailyBloodRequestScheduler.apxc
Code Coverage: None • API Version: 64 • Go To

1 trigger BloodRequestTrigger on Request__c (after insert, after update) {
2
3     List<Request__c> reqsToProcess = new List<Request__c>();
4     for(Request__c req : Trigger.new){
5         if(req.Units_Required__c != null){
6             reqsToProcess.add(req);
7         }
8     }
9
10    if(!reqsToProcess.isEmpty()){
11        BloodRequestHandler.processRequests(reqsToProcess);
12    }
13 }
```

## 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.

```
BloodRequestHandler.apxc | BloodRequestTrigger.apxt | BloodRequestTriggerHandler.apxc | DailyBloodRequestScheduler.apxc
Code Coverage: None | API Version: 64 | Go To

1 global class DailyBloodRequestScheduler implements Schedulable {
2
3     global void execute(SchedulableContext sc) {
4         processPendingRequests();
5     }
6
7     // Public method you can call manually for testing
8     public void processPendingRequests() {
9
10        // 1 Get all pending requests (status not 'In Progress')
11        List<Request__c> pendingRequests = [
12            SELECT Id, Blood_Group_Needed__c, Units_Required__c, Request_Status__c, Total_Units_Available__c
13            FROM Request__c
14            WHERE Request_Status__c != 'In Progress'
15            AND Units_Required__c != null
16        ];
17
18        if(pendingRequests.isEmpty()) return;
19
20        // 2 Collect all blood types needed
21        Set<String> bloodTypesNeeded = new Set<String>();
22        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.