**CharityConnect CRM – Donor Transparency & Engagement Platform**

**Phase 1: Problem Understanding & Industry Analysis**

**Industry:** Non-Profit / Charity (Crowdfunding & Donations)  
**Project Type:** Salesforce CRM Implementation (Admin + Developer)  
**Target Users:** Donors, NGO Staff/Admins, Beneficiaries

**Problem Statement**

Non-profits struggle to build trust and engage donors due to:

* Scattered donation records in Excel/Sheets.
* Limited visibility into how donations are utilized.
* No reward system for recurring or loyal donors.
* Manual effort in sending updates, receipts, and impact reports.

This leads to **low donor retention, inefficiency, and declining trust**.

**Goal**

Develop a Salesforce-based platform that:

* Centralizes donors, donations, campaigns, and beneficiaries.
* Provides **real-time donation tracking and impact reports**.
* Introduces **innovative engagement features** (Smart Cart, Loyalty Points, Badges, Forecasting).
* Automates thank-you emails, festive donation drives, and reminders.
* Enhances transparency through dashboards and donor profiles.

**Requirement Gathering**

**Business Needs**

* Centralized platform for NGO staff and donors.
* Automated donation tracking + communication.
* Gamification features to retain donor interest.
* Impact-driven dashboards for staff and donors.

**Functional Requirements**

* **Donor Object** → details, donation history, loyalty points.
* **Campaign Object** → fundraising goals, progress.
* **Beneficiary Object** → supported communities/individuals.
* **Donation Object** → links Donor → Campaign → Beneficiary.
* **Smart Donor Cart** → select multiple campaigns, auto-total.
* **Donor Loyalty Points & Badges** → points for each donation, auto-badges via automation.
* **Impact Forecasting** → formulas & dashboard to show future impact.
* **Festive Campaign Triggers** → seasonal auto-reminders.

**Non-Functional Requirements**

* Mobile-friendly via Salesforce app.
* Role-based access (Donor, NGO Staff, Admin).
* Scalable for thousands of donors.
* Easy UI using Lightning App Builder + LWCs.

**Stakeholder Analysis**

| **Stakeholder** | **Needs / Responsibilities** |
| --- | --- |
| Donors | Donate easily, see real-time impact, earn rewards (points & badges), forecast future impact. |
| NGO Staff | Create/manage campaigns, track donations, run festive drives, update beneficiaries. |
| Beneficiaries | Receive support, provide updates, connect transparently with donors. |
| Admin | Manage setup, security, permissions, and ensure compliance. |

**Business Process Mapping**

**Current Manual Process**

* Donations recorded in Excel.
* Donors informed via manual email/WhatsApp.
* Reports created monthly with delays.
* No systematic loyalty/reward tracking.

**Proposed Salesforce Process**

1. Donor selects multiple campaigns → added to **Impact Cart**.
2. Total donation auto-calculates → stored as Donation record(s).
3. **Thank-you email** sent automatically.
4. Recurring reminders + festive campaign appeals triggered by Flow.
5. Donor earns **Loyalty Points** → when thresholds met, **Badges auto-assigned**.
6. **Impact Forecasting** shows how future pledges translate into meals/benefits.
7. Dashboards auto-refresh to show real-time donor & campaign performance.

**Industry-Specific Use Cases**

* **Smart Donor Cart** → e-commerce-like donation experience.
* **Donation Transparency Dashboard** → live usage of funds.
* **Gamification with Loyalty Points & Badges** → motivates donors.
* **Impact Forecasting** → future impact visualization.
* **Festive Campaign Triggers** → seasonal drives like “This Diwali, sponsor a meal.”

**AppExchange Exploration**

* **Nonprofit Success Pack (NPSP):** Base framework for NGO donation tracking.
* **Conga Composer:** Auto-generate donation receipts.
* **FormAssembly:** Custom donor registration forms.
* **SurveyMonkey:** Collect donor feedback post-campaign.

**Phase 2: Org Setup & Configuration**

### Salesforce Edition

* Sign up for a **Developer Edition Org**..
* Log in with your login credentials.

### Company Profile Setup

* Company Name: CharityConnect NGO
* Currency: INR
* Time Zone: Asia/Kolkata (IST)
* Address and contact details configured.

### Business Hours & Holidays

* Standard business hours: Monday–Friday, 9 AM – 6 PM IST;   
   Saturday–Sunday, 10 AM – 3 PM IST.
* Holidays configured: Diwali, Christmas, Independence Day.

### Fiscal Year Settings

* Standard Fiscal Year: April – March.

### User Setup & Licenses

* System Administrator (Admin User)
* NGO Staff User
* Donor User

### Profiles

* System Administrator (default)
* NGO Staff Profile (customized from Standard User)
* Donor Profile (customized with limited access)

### Roles

* Admin
* NGO Manager
* NGO Staff
* Donor

### Permission Sets

* Donor Dashboard Access
* NGO Advanced Reporting

### Organization-Wide Defaults (OWD)

* Donations: Private
* Donors: Private
* Campaigns: Public Read Only
* Beneficiaries: Public Read Only

### Sharing Rules

* NGO Staff access to team donations
* Donors restricted to their own donations only

### Login Access Policies

* Enabled for troubleshooting and admin support

### Dev Org Setup

* Developer Org created and configured for the project

### Sandbox Usage

* Not applicable (Developer Edition)
* Simulated via secondary Developer Org if required

### Deployment Basics

* Change Sets identified as the primary deployment method
* Awareness of ANT Migration Tool and SFDX for advanced deployments

### Screenshots:

### Developer Org Setup

### 

### Company Profile Setup

### 

### Business Hours & Holidays

### 

### 

### Fiscal Year Settings

### 

### Profiles

### 

### Roles

### 

### Permission Sets

### 

**Phase 3: Data Modeling & Relationships**

**Standard Objects Used**

* **User** → Represents NGO staff, donors (if given portal access), and admins.
* **Campaign** → Used to manage fundraising initiatives (e.g., Education Drive, Healthcare Relief).
* **Contact (extended for Donors)** → Acts as the base for donor information, extended with custom fields.

**Custom Objects Created**

1. **Donor**
   * Stores additional donor details beyond Contact.
   * Key Fields: Donor ID (Auto-Number), Loyalty Points, Preferred Cause, Total Contributions, Badge Status.
2. **Donation**
   * Tracks all individual donations.
   * Key Fields: Donation ID, Amount, Date, Mode of Payment, Status (Pending/Completed/Failed).
   * Relationships: Linked to Donor, Campaign, and Beneficiary.
3. **Beneficiary**
   * Represents individuals or communities supported.
   * Key Fields: Beneficiary ID, Name, Type (Individual/Community), Support Details, Success Story.
4. **Donor Badge**
   * Tracks gamification & recognition.
   * Key Fields: Badge ID, Badge Type (Bronze/Silver/Gold), Earned Date.
5. **Impact Cart**
   * Mimics a shopping cart for multi-campaign donations.
   * Key Fields: Cart ID, Selected Campaigns, Total Donation Value, Forecasted Impact.

**Fields Breakdown (Detailed)**

**Donor**

* Donor ID (Auto-Number)
* First Name (Text)
* Last Name (Text)
* Email (Email)
* Phone (Phone)
* Address (Text Area)
* Loyalty Points (Number)
* Total Contributions (Currency, Roll-up from Donations)
* Preferred Cause (Picklist: Education, Health, Disaster Relief, Other)
* Badge Status (Formula/Lookup from Donor Badge)

**Donation**

* Donation ID (Auto-Number)
* Donation Amount (Currency)
* Date (Date/Time)
* Mode of Payment (Picklist: UPI, Card, Bank Transfer, Cash, Others)
* Status (Picklist: Pending, Completed, Failed, Refunded)
* Donor (Lookup to Donor)
* Campaign (Master-Detail to Campaign)
* Beneficiary (Lookup to Beneficiary)

**Beneficiary**

* Beneficiary ID (Auto-Number)
* Name (Text)
* Type (Picklist: Individual, Community)
* Age (Number, optional if Individual)
* Location (Text)
* Support Details (Long Text Area)
* Success Story (Rich Text Area)

**Campaign (Extended)**

* Campaign Goal (Currency)
* Amount Raised (Currency, Roll-up from Donations)
* Start Date (Date)
* End Date (Date)
* Status (Picklist: Active, Completed, Cancelled)

**Donor Badge**

* Badge ID (Auto-Number)
* Badge Type (Picklist: Bronze, Silver, Gold, Platinum)
* Earned Date (Date)
* Donor (Master-Detail to Donor)

**Impact Cart**

* Cart ID (Auto-Number)
* Donor (Lookup to Donor)
* Selected Campaigns (Multi-Select Picklist or Junction Relationship)
* Total Donation Value (Currency, Calculated)
* Forecasted Impact (Formula/Rich Text Area)

**Record Types**

* **Donation** → One-Time vs Recurring
* **Campaign** → Education, Healthcare, Disaster Relief, CSR
* **Beneficiary** → Individual vs Community

**Page Layouts**

* **Donor Layout** → Personal info, donation history (related list), loyalty points, earned badges.
* **Donation Layout** → Donor, Campaign, Beneficiary, Amount, Status.
* **Campaign Layout** → Goal, Raised, Progress bar, Associated Donations.
* **Beneficiary Layout** → Details, Linked Donations, Linked Campaigns, Success Stories.
* **Donor Badge Layout** → Badge Type, Earned Date, Linked Donor.
* **Impact Cart Layout** → Selected Campaigns, Total Value, Forecasted Impact.

### Compact Layouts

* **Donor** → Name, Loyalty Points, Total Contributions, Badge Status.
* **Donation** → Amount, Status, Date.
* **Campaign** → Name, Goal, Raised, End Date.
* **Beneficiary** → Name, Type, Location, Support Status.

### Relationships (Detailed)

* **Donor → Donation** → Lookup (One donor can have many donations).
* **Campaign → Donation** → Master-Detail (Each donation must belong to a campaign).
* **Beneficiary → Donation** → Lookup (Donation linked to one beneficiary).
* **Donor → Donor Badge** → Master-Detail (Badges tied to donor lifecycle).
* **Donor → Impact Cart** → Lookup (Cart is linked to donor, can have multiple campaigns).

### Junction Objects

* **Donation** → Junction between Donor and Campaign.
* **Impact Cart** → Junction between Donor and multiple Campaigns.

### Schema Builder

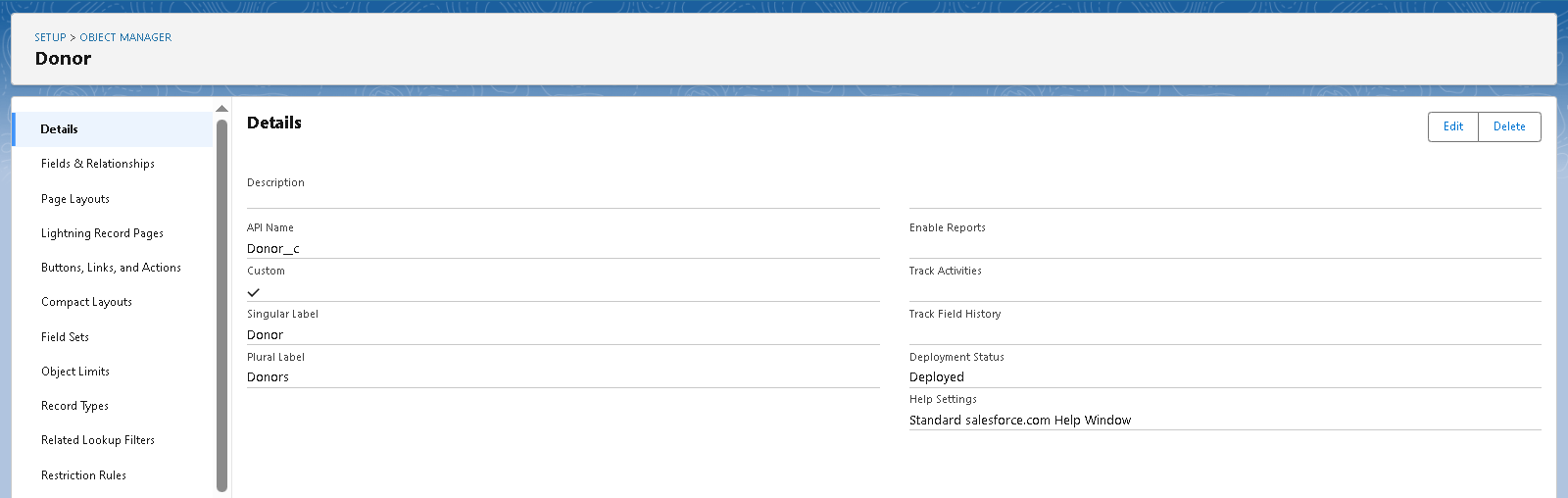
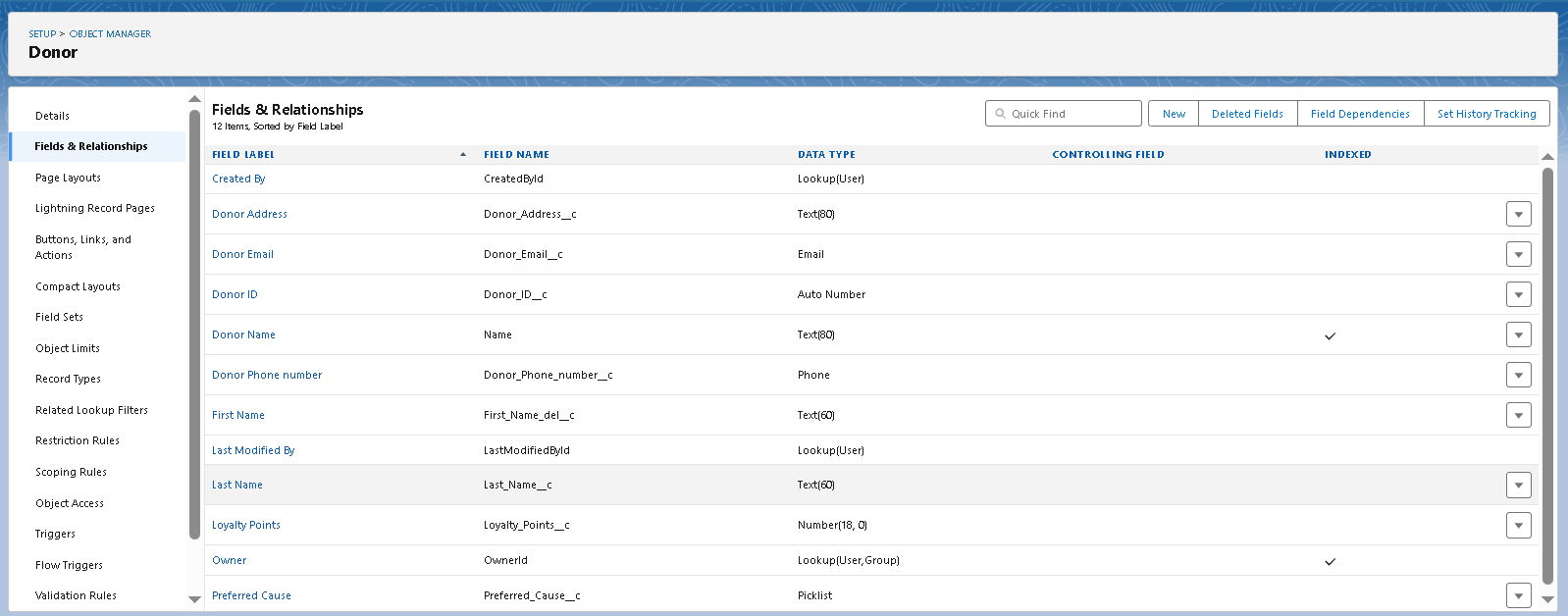
* Used to visually design & verify relationships.
* Highlights:
  + Central object → **Donation**.
  + Donor ↔ Donation ↔ Campaign → core fundraising flow.
  + Beneficiary connected via lookup to Donation.
  + Gamification objects (Donor Badge, Impact Cart) extend donor engagement model.

### External Objects (Future Scope)

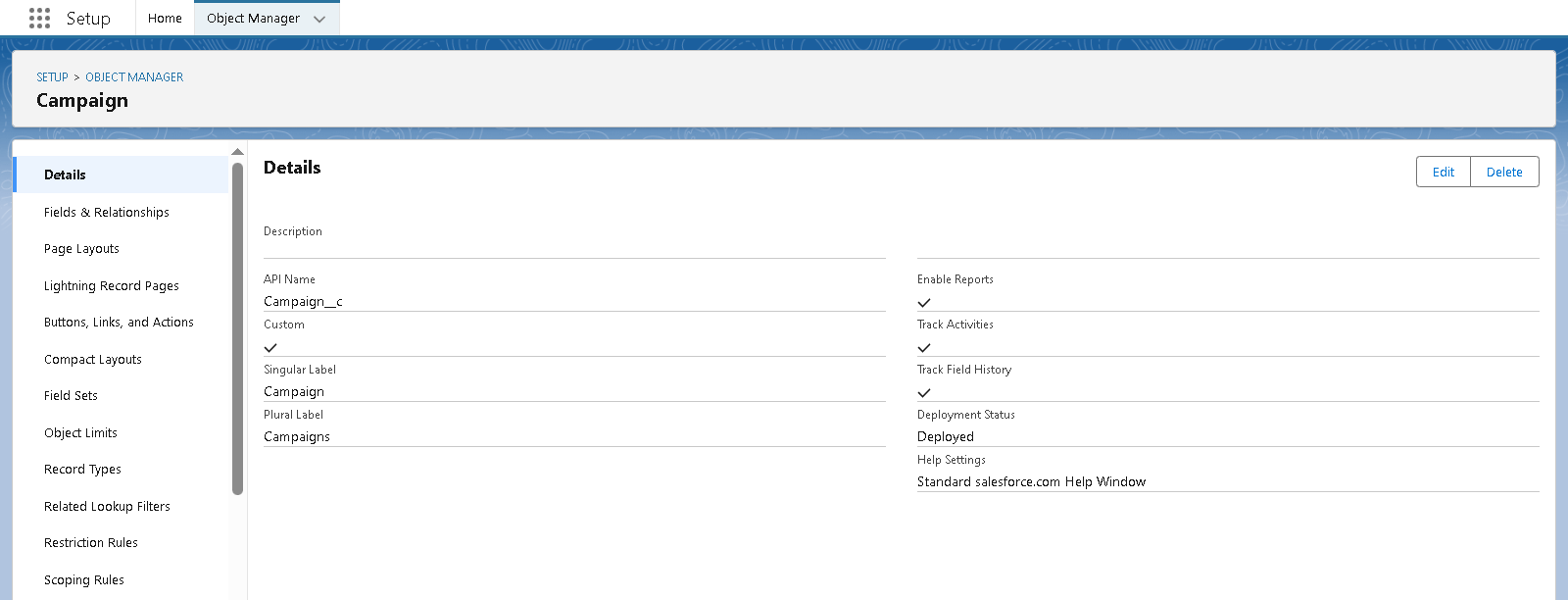
* **Payment Gateway Integration** → Stripe/PayPal/UPI external objects to pull transaction details.
* **Corporate CSR Systems** → Sync corporate donations to Salesforce campaigns.
* **Impact Story Repository** → External content (images, videos, testimonials) stored outside Salesforce but linked via External Objects.

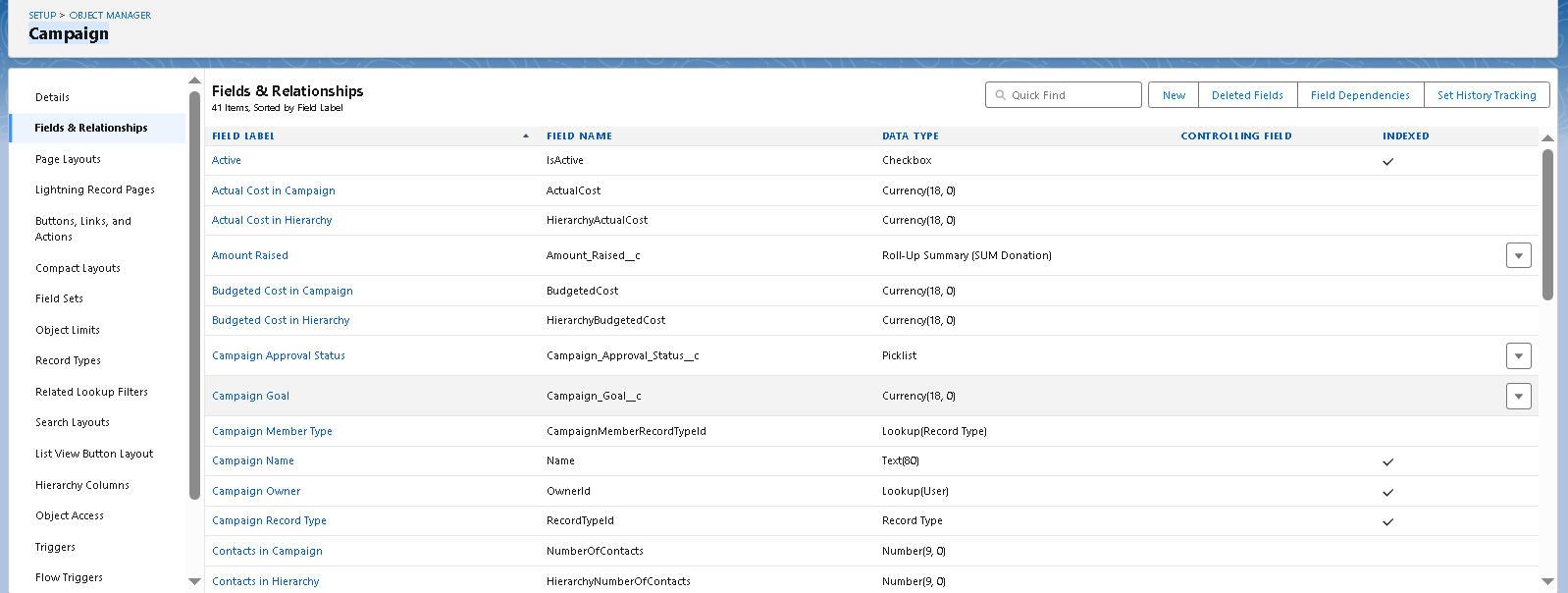
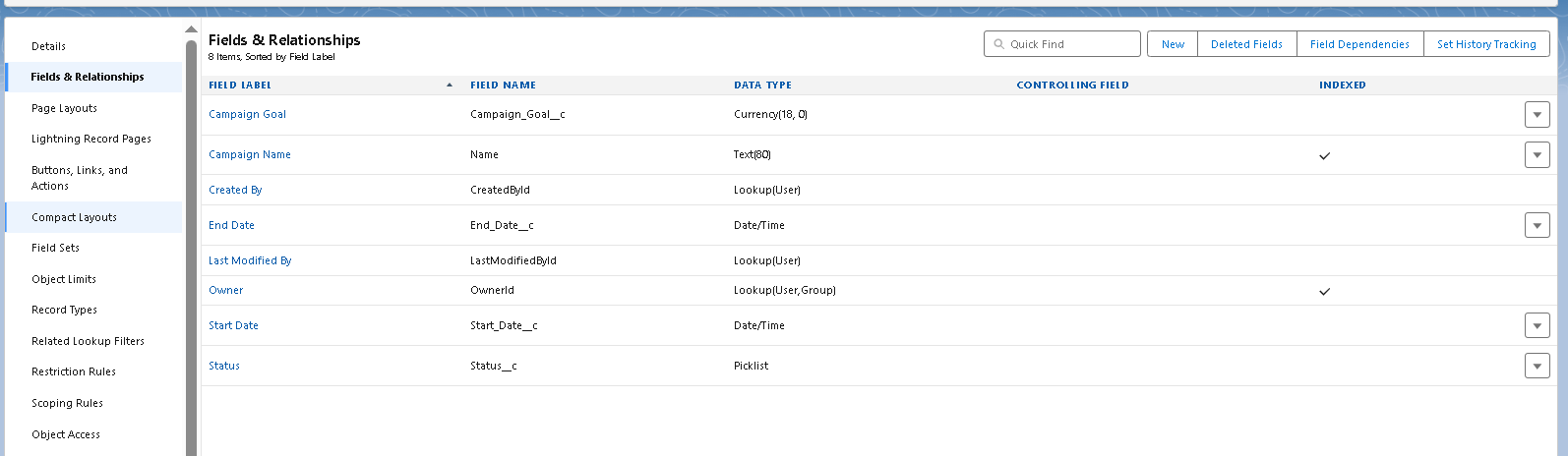
### Screenshots:

### Donor Object

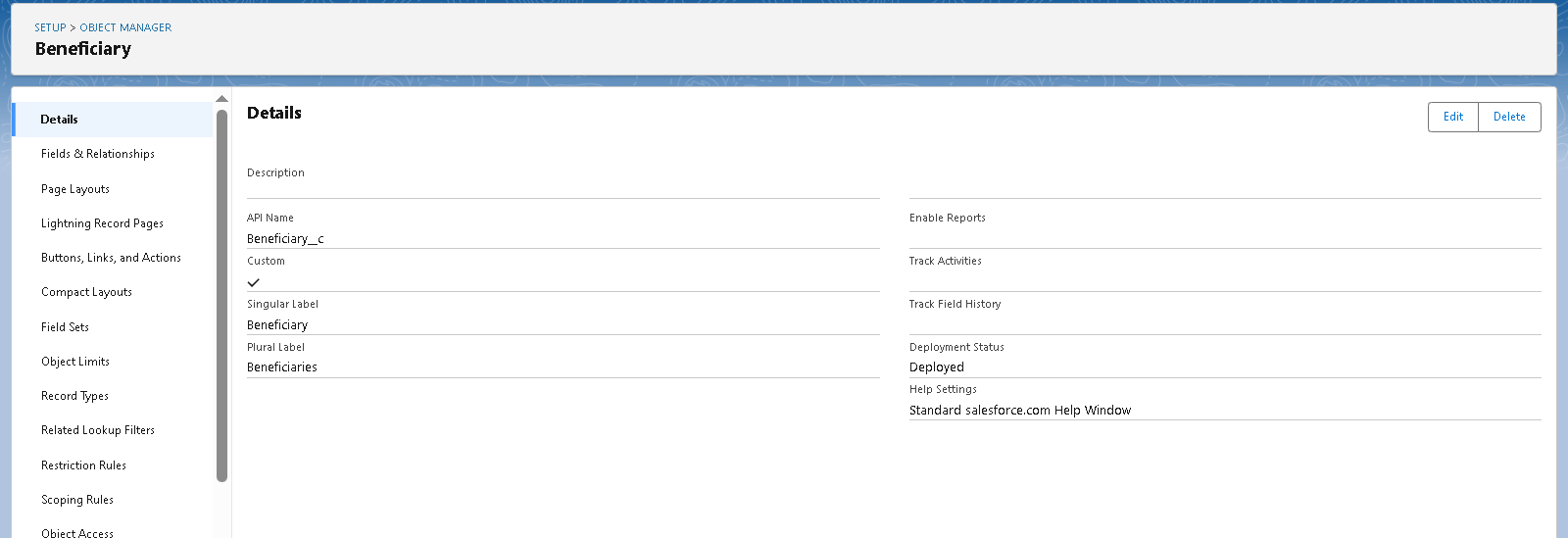
 

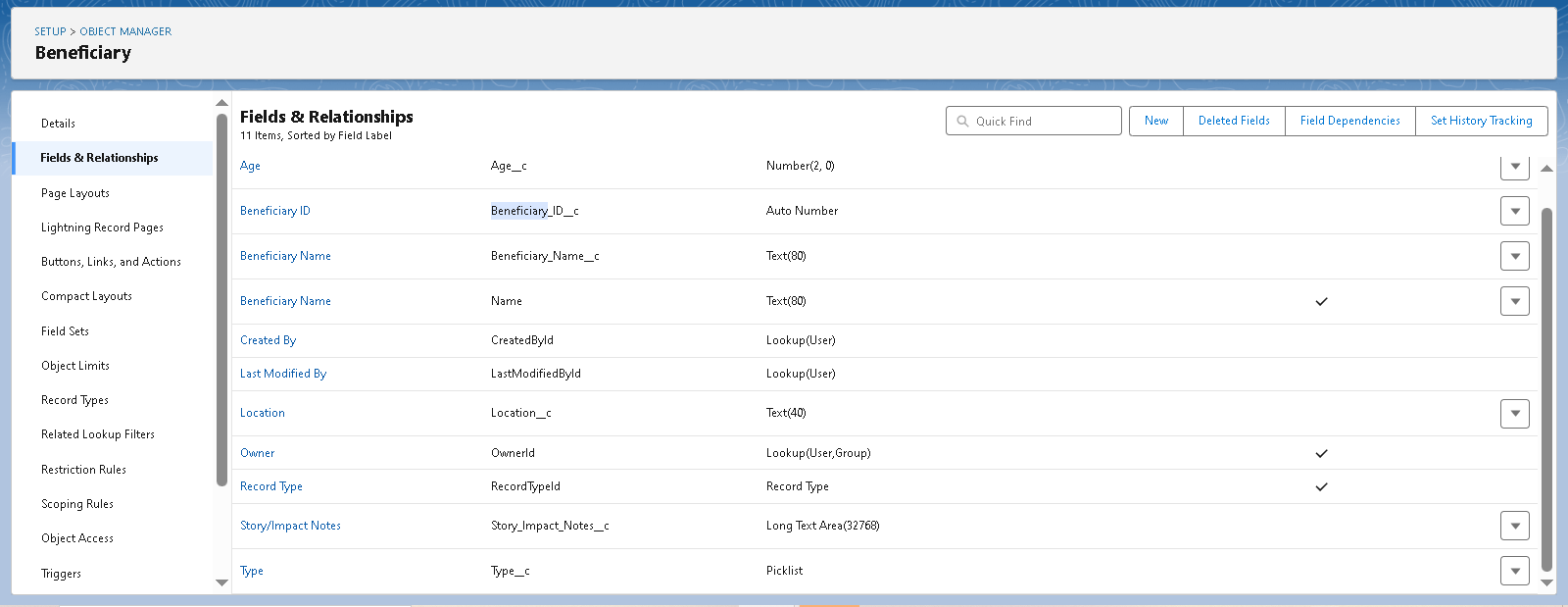
### Campaign Object



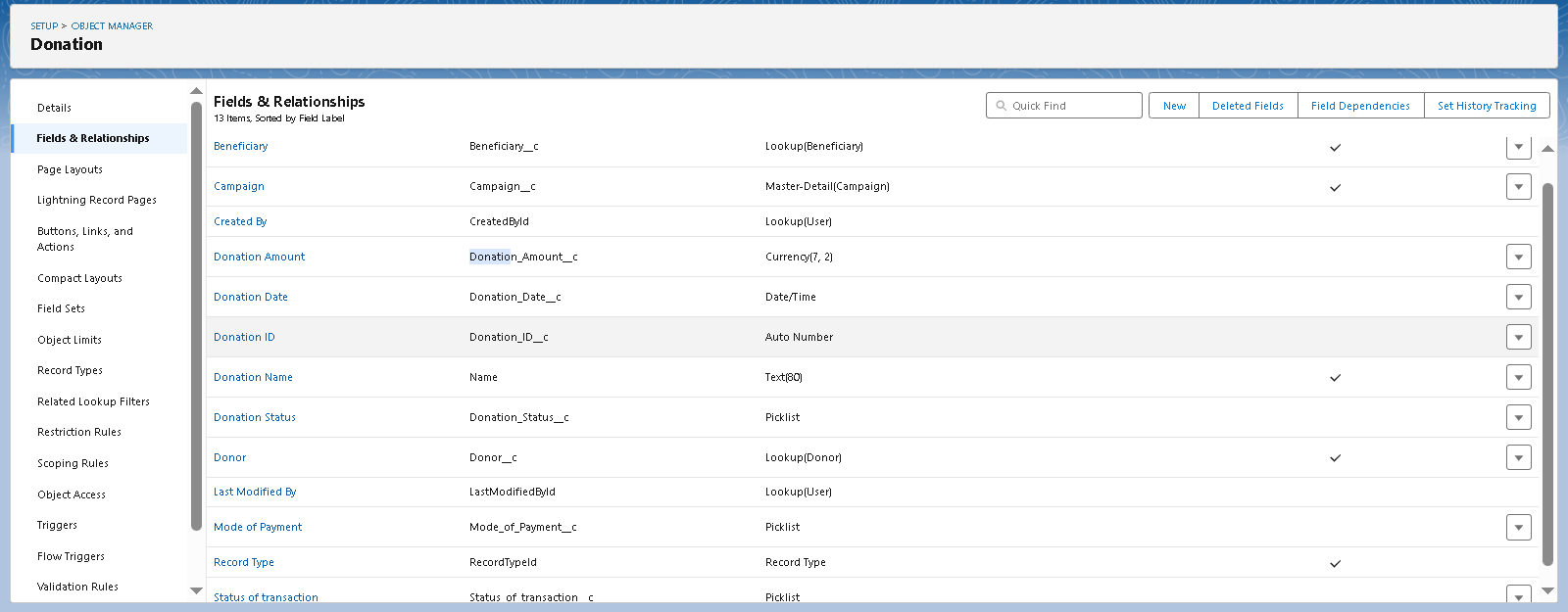
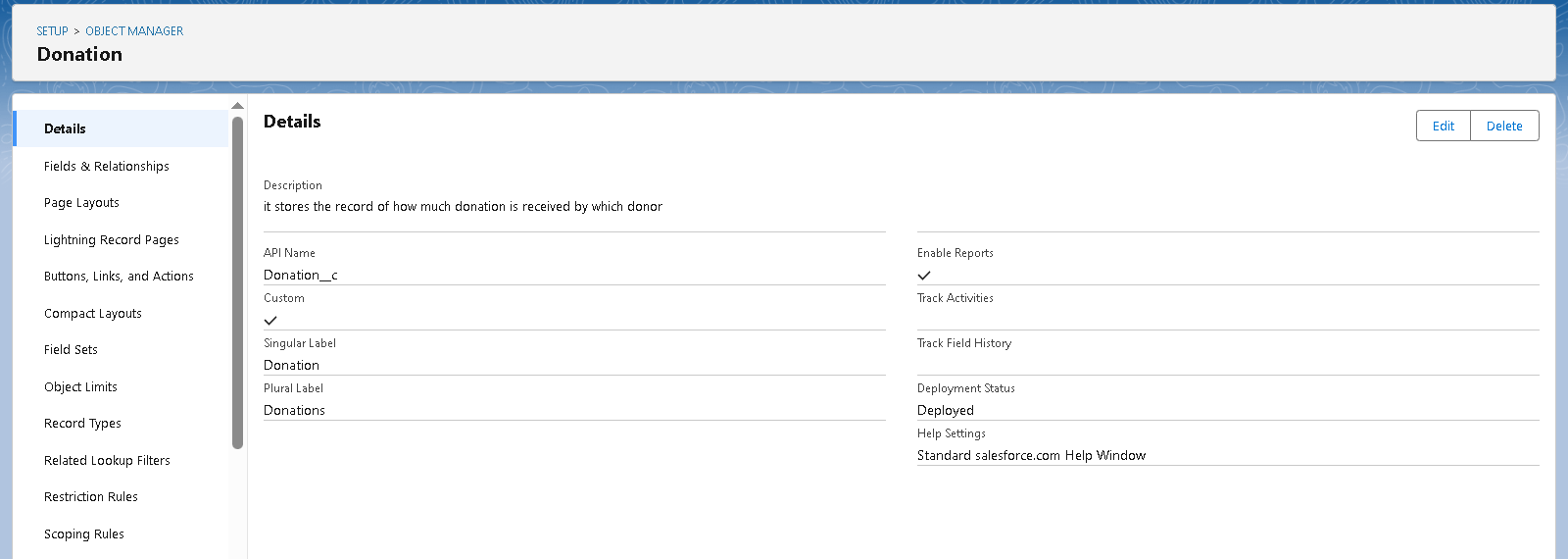
 

### Beneficiary Object

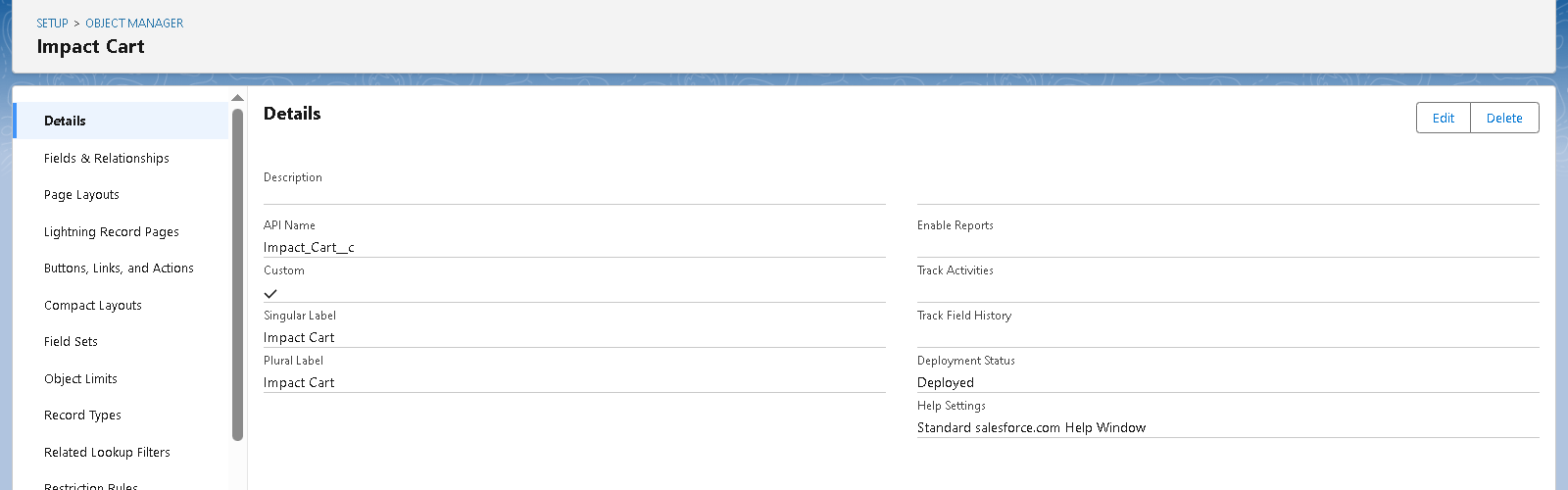
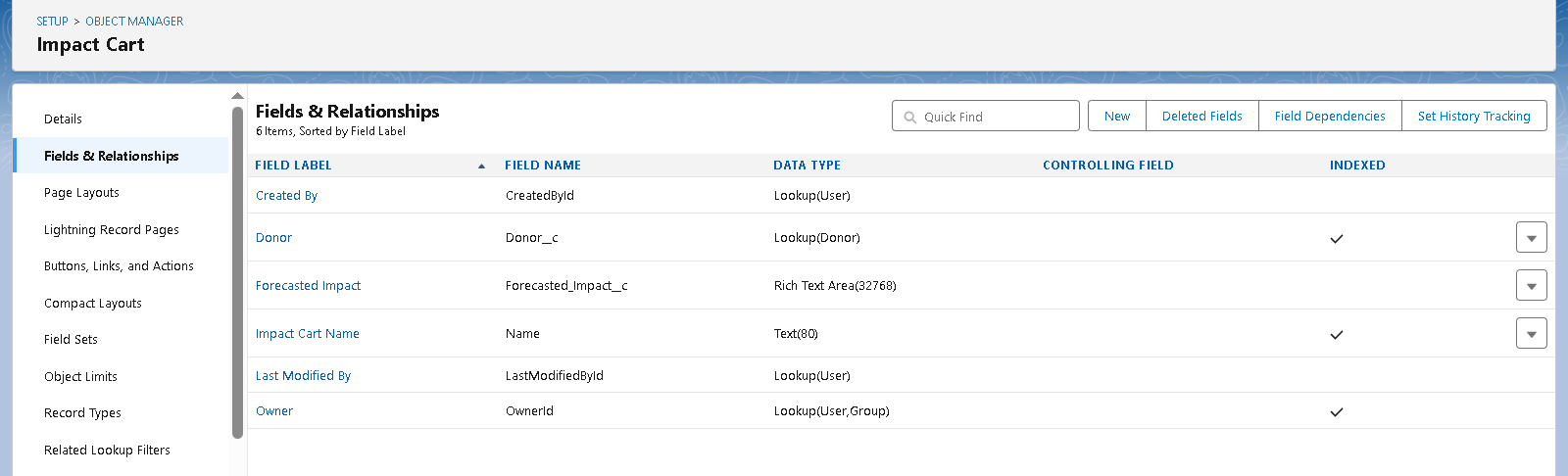




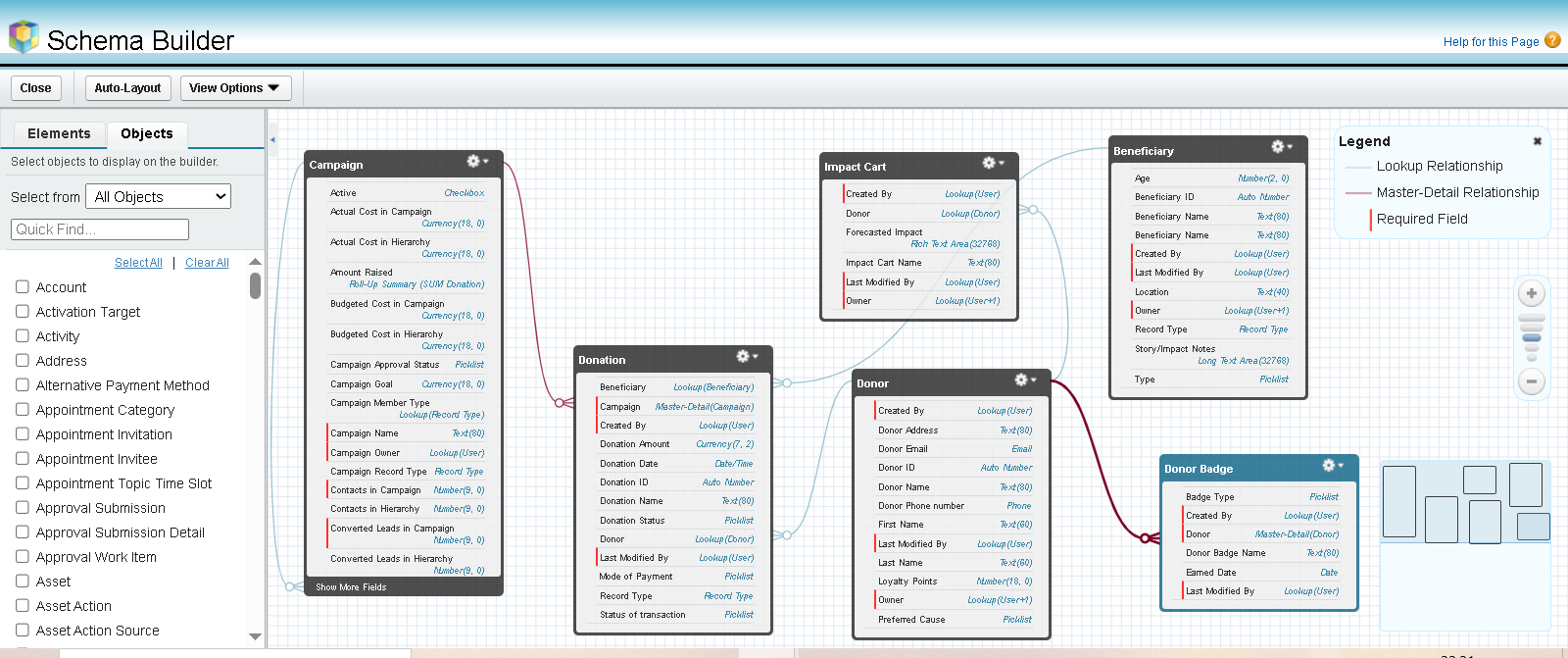
### Donation Object



### Impact Cart Object

### Schema Builder



**Phase 4: Process Automation (Admin)**

**Validation Rules**

Ensure data accuracy and prevent bad entries.

* **Donation Amount > 0** → Block negative or zero donations.
* **Email Required for Donor** → Donor record must have a valid email.
* **Campaign End Date > Start Date** → Prevent invalid campaign setup.
* **Beneficiary Age > 0** (if Type = Individual).

**Workflow Rules**

Automated simple field updates & alerts.

* **Donation Status = Completed** → Send **email alert** to donor with thank-you note.
* **Campaign Goal Reached** → Update Campaign Status to “Completed”.
* **Badge Earned** → Notify NGO staff when a donor reaches a new badge level.

**Process Builder**

Complex, conditional automation (but keep light since Flow is future-proof).

* **Recurring Donations** → Auto-generate next month’s donation record if donor opted for recurring.
* **Donor Loyalty Points** → Add points automatically based on donation amount (e.g., ₹1000 = 10 points).
* **High Value Donation (₹50,000+)** → Notify NGO Manager + create follow-up task for personal acknowledgment.

**Approval Process**

Formal approvals are used where human decision is required.

* **Refund Requests** → Donor requests refund → goes to NGO Manager for approval.
* **Large Campaign Proposals (> ₹5,00,000 goal)** → Requires Admin approval before going live.

**Flow Builder**

Most powerful automation tool (replacing Workflow/Process Builder gradually).  
Types of flows used:

1. **Record-Triggered Flow**
   * When **Donation is created with Status = Completed** →
     + Update Donor’s Total Contributions.
     + Update Campaign’s Amount Raised.
     + Check if Donor qualifies for a new Badge → Assign automatically.
2. **Screen Flow**
   * **Donor Portal Flow** → Donors can select campaigns, beneficiaries, and add to their **Impact Cart** before confirming donation.
   * **NGO Staff Flow** → Simplified UI for quickly adding new Beneficiaries.
3. **Scheduled Flow**
   * Weekly → Send donors a **consolidated donation summary**.
   * Monthly → Send NGO staff a report of campaigns nearing deadlines.
4. **Auto-Launched Flow**
   * Triggered by other processes (e.g., badge assignment).

**Email Alerts**

* **Donation Receipt** → Sent when Donation = Completed.
* **Campaign Update** → Sent to all campaign donors when milestone reached (e.g., 50% funded).
* **Badge Earned** → Congratulatory email with digital badge.

**Field Updates**

* Update **Donor Loyalty Points** on donation save.
* Update **Campaign Status** automatically when goals are reached.
* Set **Refund Status** to “Pending Manager Approval” when donor initiates request.

**Tasks**

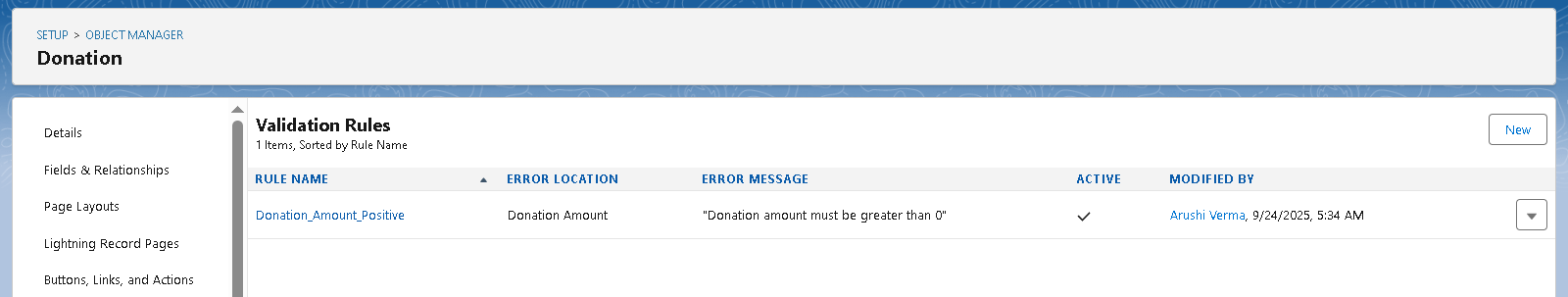
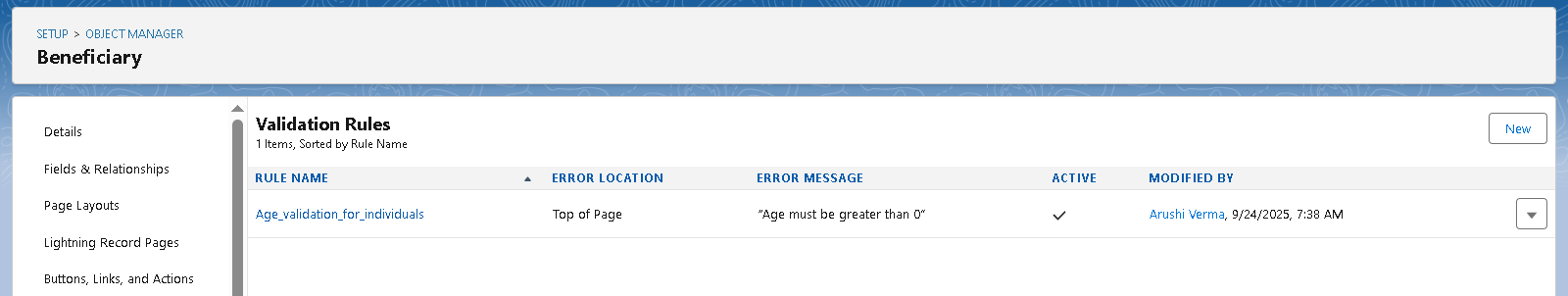
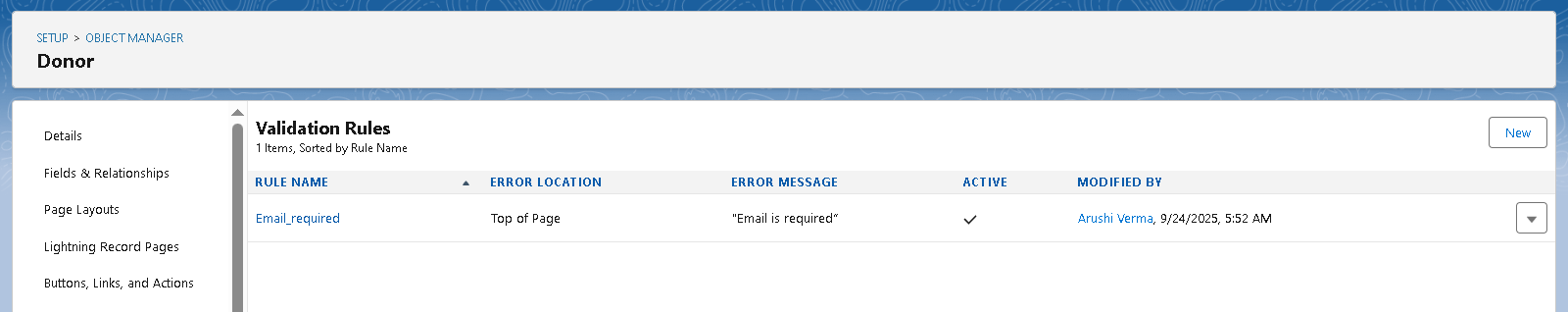
* Assign follow-up task to NGO Staff when a new Beneficiary is added (verify details).
* Assign donor relationship manager task when a **major donor (>₹50k)** contributes.

**Custom Notifications**

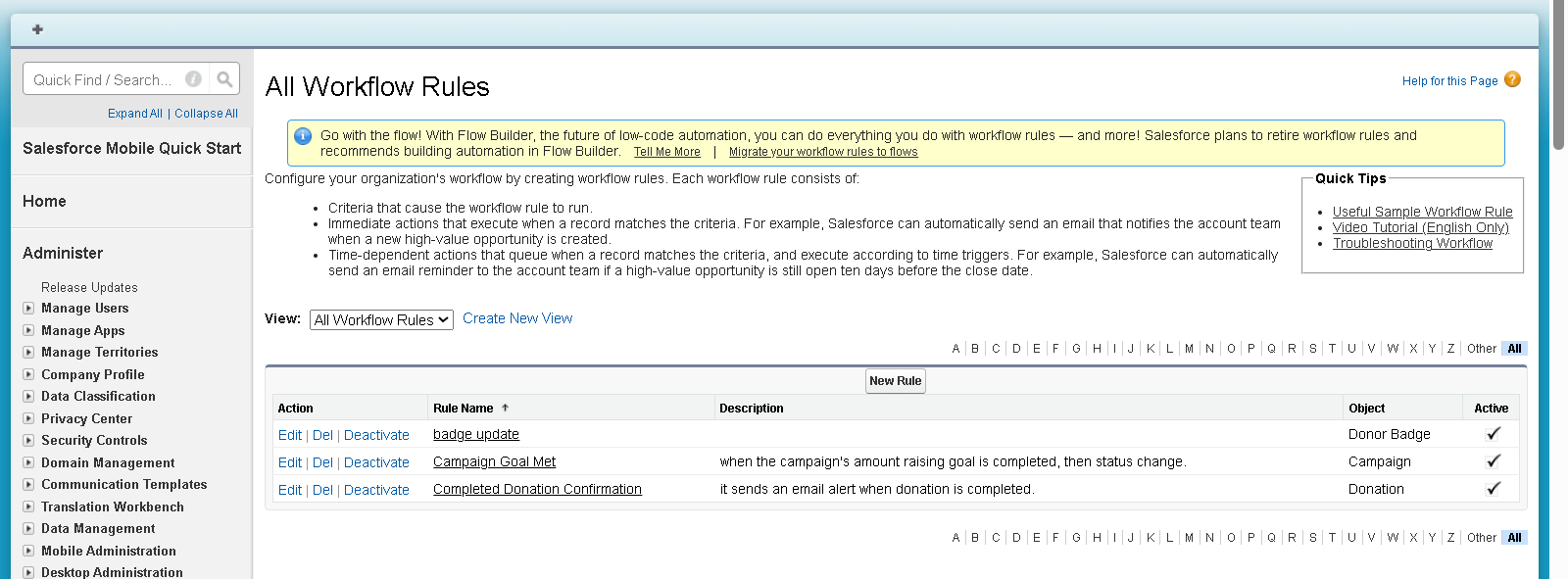
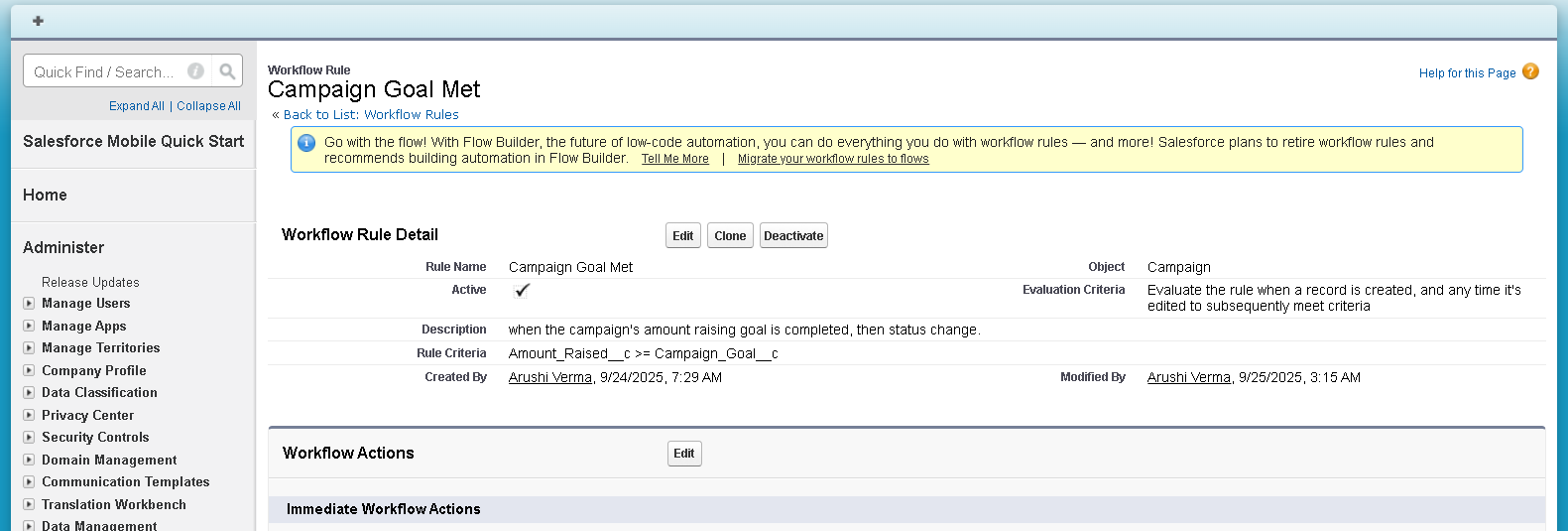
* **Push Notification to NGO Staff** when a new large donation is made.
* **Push Notification to Donor** (via Experience Cloud app) when their donation is successfully received.
* **Internal Alert** for Admin when suspicious donation patterns (multiple failed attempts) occur.

**Screenshots:**

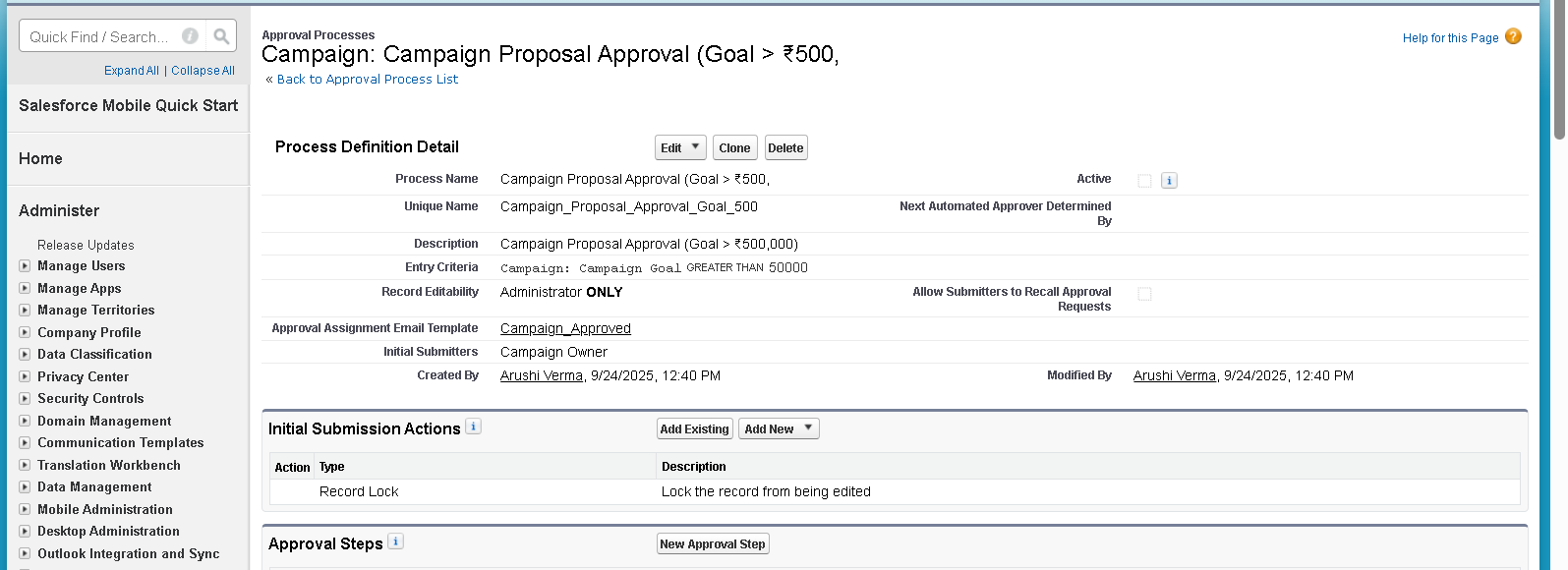
**Validation rules**

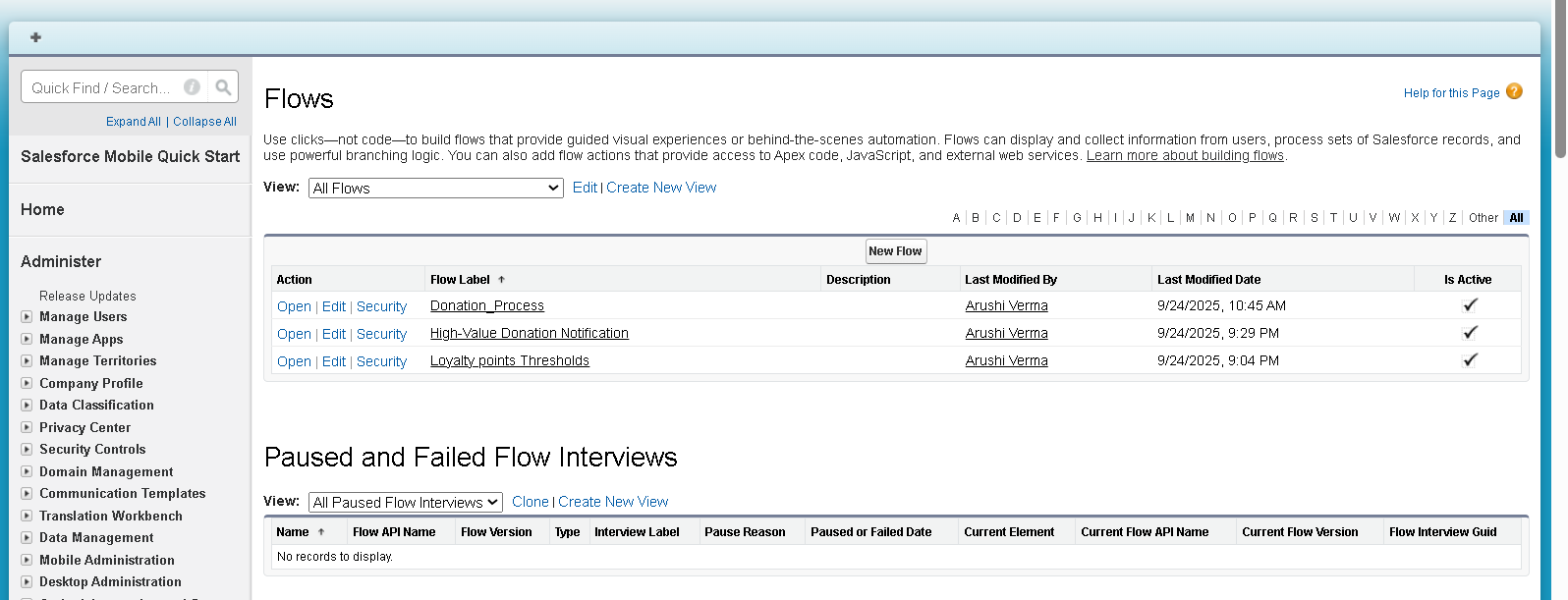
**Work Flows**

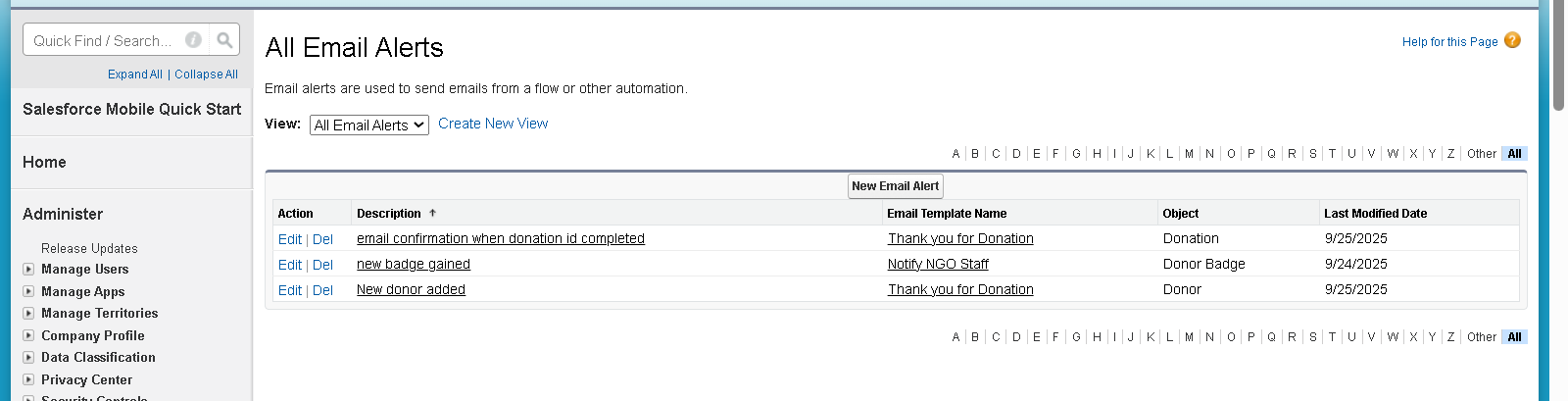
**Approval Process**

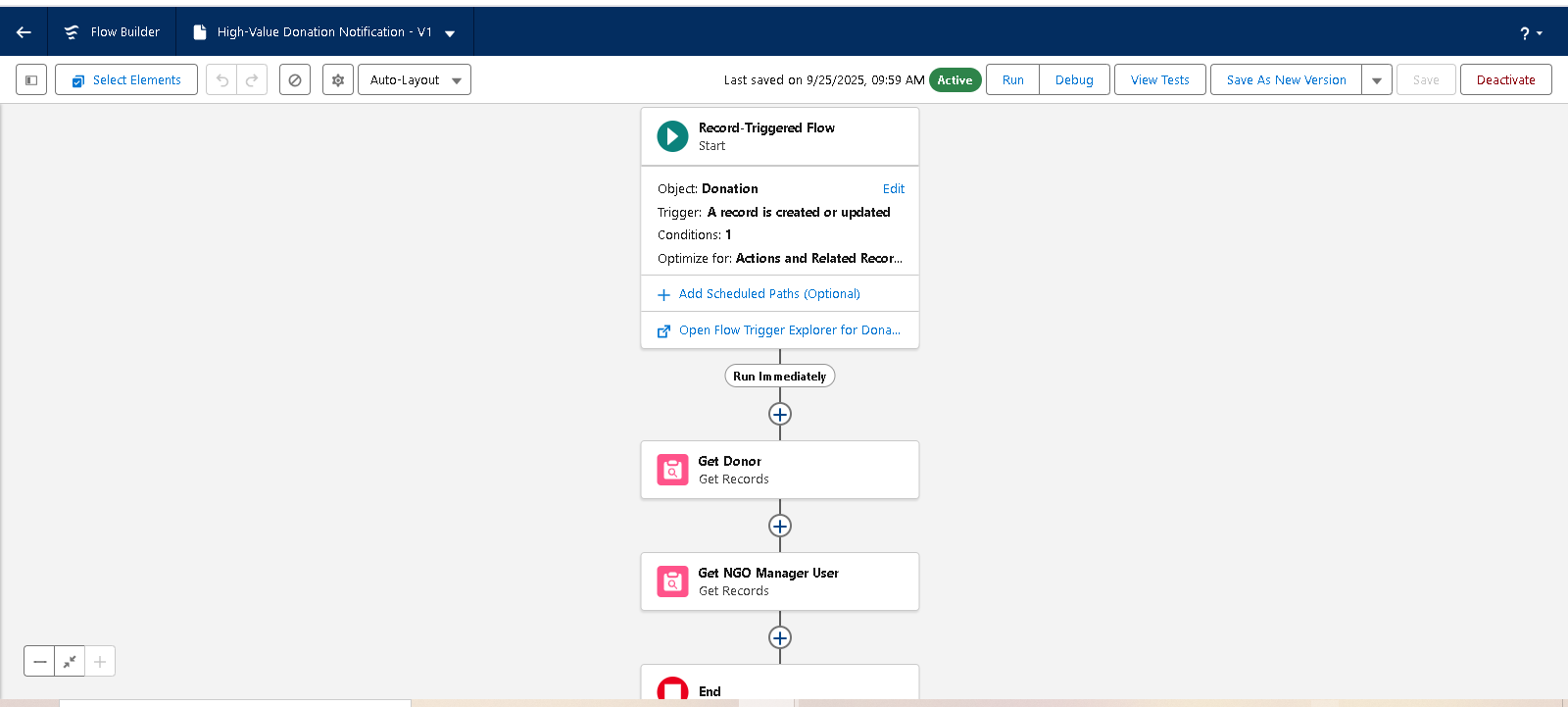
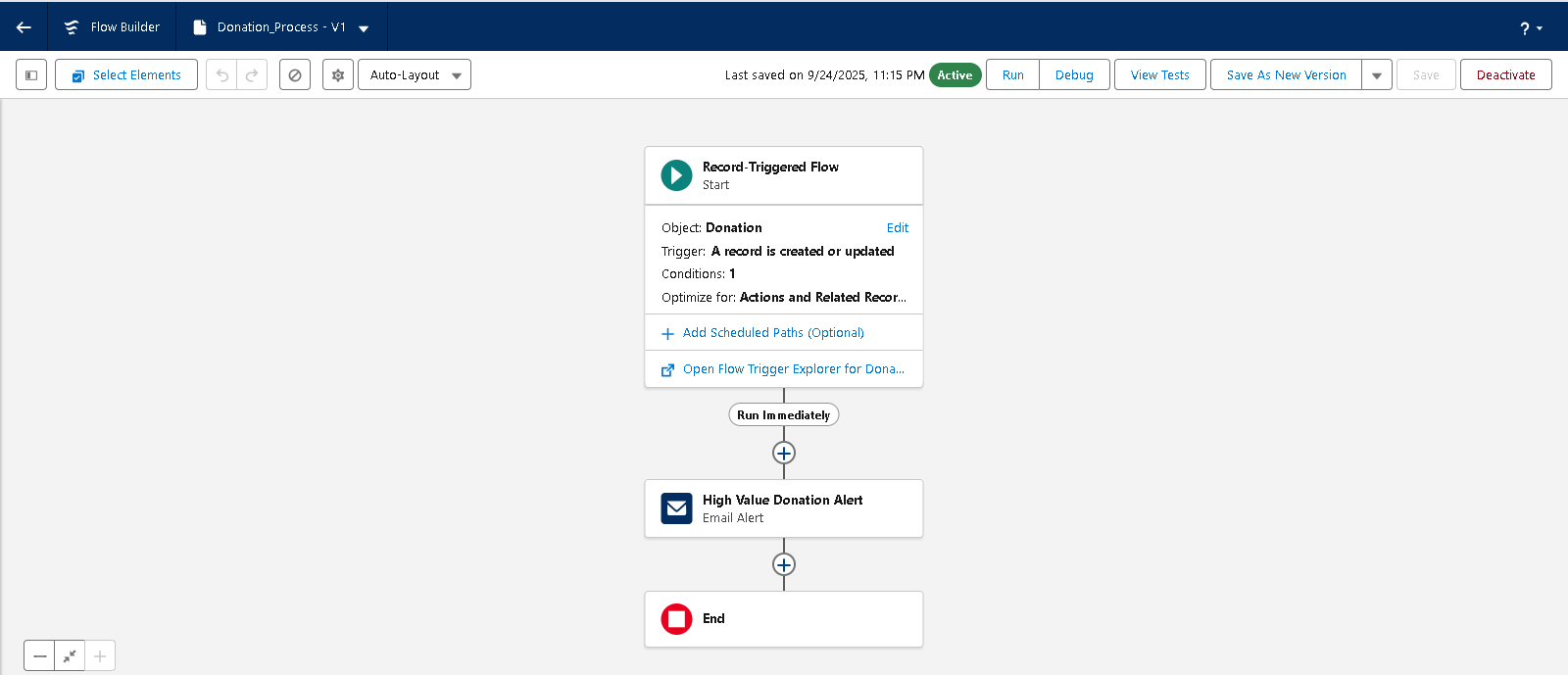
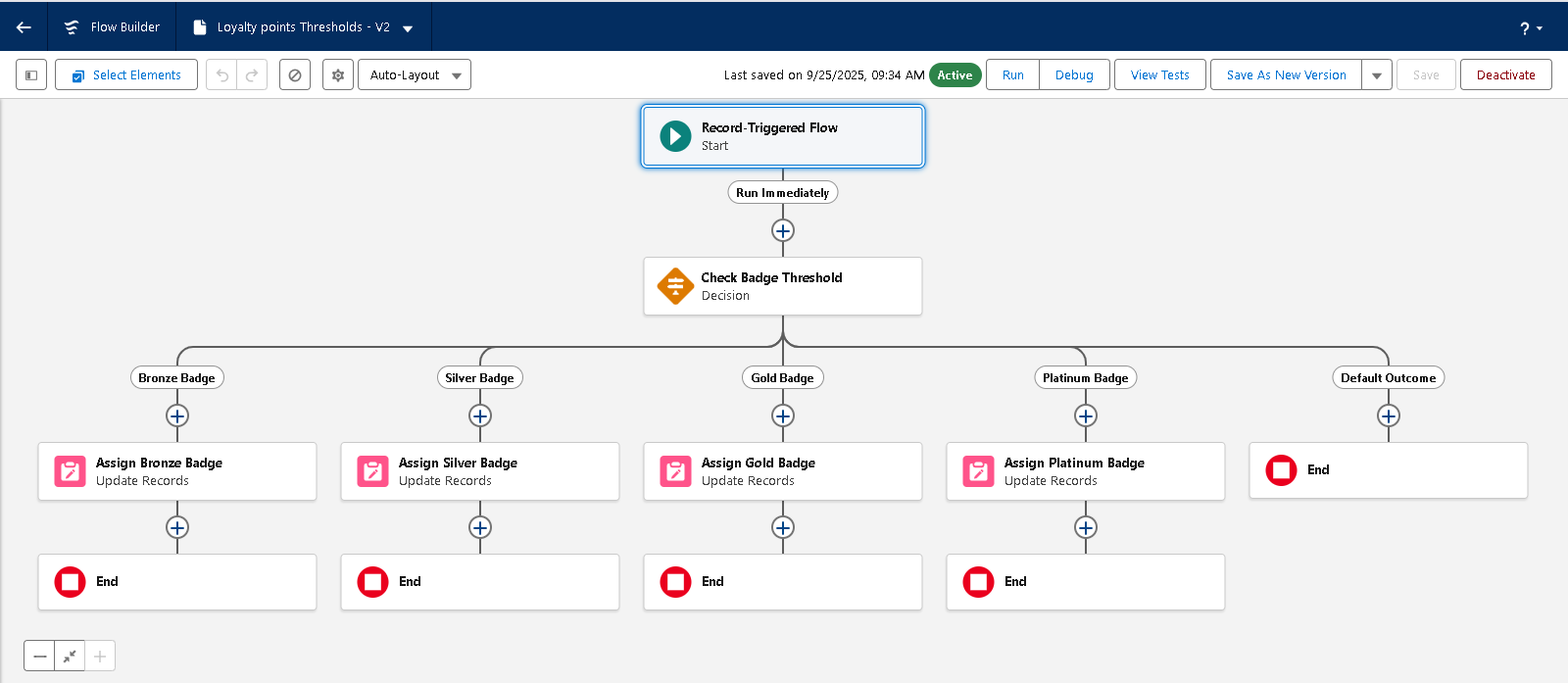
****

**Flow**



**Email Alerts**



**Flow builder**

**Phase 5: Apex Programming (Developer)**

**Classes & Objects**

We created custom **Apex Classes** to encapsulate business logic.

**Example:**

* **DonationHandler.cls** → Handles logic for donation insert/update.
* **BadgeService.cls** → Calculates and assigns badges based on loyalty points.

**Benefit:** Separating logic into classes ensures **reusability and clean architecture**.

**Apex Triggers (Before/After Insert/Update/Delete)**

**Key Triggers Implemented:**

1. **DonationTrigger**
   * *After Insert:* Update Donor’s total contributions and loyalty points.
   * *After Update:* If donation status changes to *Completed*, recalculate totals.
2. **CampaignTrigger**
   * *After Insert/Update:* Check if Amount Raised >= Goal → update status to *Completed*.

**Benefit:** Maintains **real-time accuracy** of donation totals and campaign progress.

**Trigger Design Pattern**

To avoid bulky triggers, we followed the **Trigger Handler Pattern**:

* One trigger per object → delegates logic to a handler class.
* Example: DonationTrigger → calls DonationHandler.onAfterInsert().

**Benefit:** Improves **readability, scalability, and testability**.

**SOQL & SOSL**

* **SOQL (Salesforce Object Query Language):**  
  Used in DonationHandler to fetch donor’s existing contributions.
* Donor d = [SELECT Id, Total\_Contributions\_\_c FROM Donor\_\_c WHERE Id = :donorId];
* **SOSL (Salesforce Object Search Language):**  
  Used for donor search by email/phone.
* List<List<sObject>> results = [FIND 'john@example.com' IN ALL FIELDS RETURNING Contact(Id, Name)];

**Benefit:** Efficiently fetches records, supporting **donor and campaign management**.

**Collections (List, Set, Map)**

* **List:** Store multiple donations for bulk updates.
* **Set:** Store unique donor IDs to avoid duplicate queries.
* **Map:** Map Donor ID → Total Contributions for quick lookups.

**Benefit:** Optimizes performance by handling bulk records gracefully.

**Control Statements**

Applied **if-else, loops, and switch cases** in handler classes.

**Example:**

if(donation.Status\_\_c == 'Completed') {

updateLoyaltyPoints(donation.Donor\_\_c, donation.Amount\_\_c);

}

**Benefit:** Provides **flexible decision-making logic** for real scenarios.

**Batch Apex**

Used for **large-scale data recalculations**.

**Example:**

* Batch job to **recalculate loyalty points** for all donors at the end of the fiscal year.

**Benefit:** Ensures **data consistency** even with thousands of records.

**Queueable Apex**

Used for **background operations that need chaining**.

**Example:**

* When a donation is completed, queueable job sends donor details to an external CSR portal.

**Benefit:** Handles **asynchronous tasks** without blocking user actions.

**Scheduled Apex**

**Use Case:** Monthly scheduled job to generate **Campaign Summary Reports** for staff.

**Benefit:** Automates recurring tasks without manual intervention.

**Future Methods**

**Use Case:**

* On donation completion, send an **email acknowledgment** using a future method (non-blocking).

**Benefit:** Improves **performance and user experience**.

**Exception Handling**

Wrapped logic in **try-catch blocks** to handle errors gracefully.

**Example:**

try {

update donor;

} catch (DmlException e) {

System.debug('Error updating donor: ' + e.getMessage());

}

**Benefit:** Prevents failures from breaking entire transactions.

**Test Classes**

Created test classes for **all Apex code**:

* **DonationHandlerTest** → validates loyalty point updates.
* **CampaignHandlerTest** → validates campaign status updates.

**Best Practices Followed:**

* Test coverage > 75% (mandatory).
* Positive and negative test cases.
* Bulk data testing.

**Benefit:** Guarantees **quality and deployability** of code.

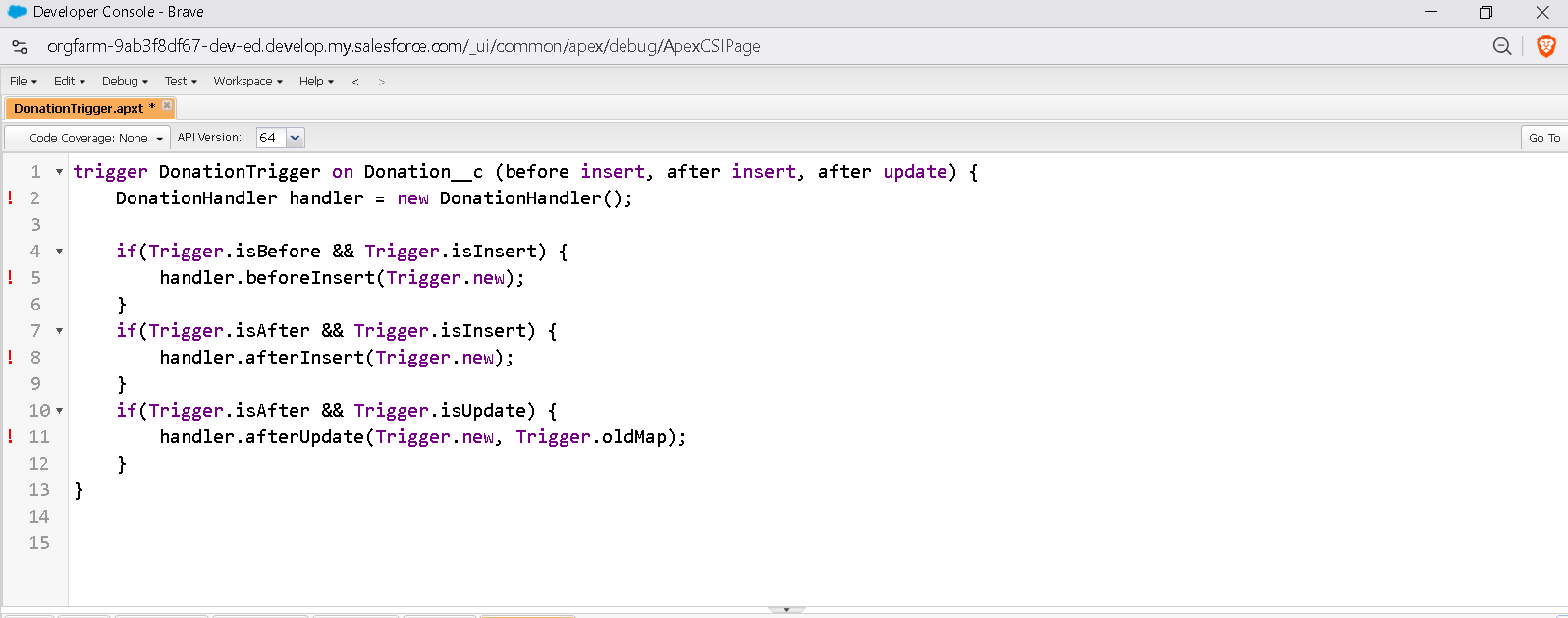
**Asynchronous Processing**

We leveraged **Batch Apex, Queueable, Scheduled, and Future methods** to handle:

* Bulk updates (Batch Apex).
* External system integration (Queueable).
* Periodic reports (Scheduled).
* Quick email sending (Future).

**Benefit:** Keeps the system **scalable, fast, and user-friendly**.

**Screenshots:**

****

**Phase 6: User Interface Development**

**Lightning App Builder**

**Purpose:** Create customized pages and apps for different user profiles without coding.

**Implementation:**

* **CharityConnect App:** A unified app containing all objects: Donors, Donations, Campaigns, Beneficiaries, Donor Badges, and Impact Cart.
* **Donor Home Page:** Highlights total contributions, loyalty points, and badges earned. Quick access to available campaigns.
* **NGO Staff Home Page:** Provides dashboards with campaign progress, donation totals, and tasks.

**Benefits:** Tailors the experience to user roles, allowing donors and staff to access relevant information quickly. It simplifies navigation and enhances user engagement.

**Record Pages**

**Purpose:** Customize how individual object records are displayed for clarity and usability.

**Implementation:**

* **Donor Record Page:** Displays personal details, donation history, loyalty points, and badges earned.
* **Campaign Record Page:** Shows campaign goal, amount raised, and associated donations.
* **Donation Record Page:** Contains donor, campaign, beneficiary, and amount details.

**Benefits:** Users can view all relevant information at a glance, reducing manual effort and errors.

**Tabs**

**Purpose:** Organize access to objects and functionality in the app.

**Implementation:**

* Tabs for **Donors, Donations, Campaigns, Beneficiaries, and Donor Badges**.
* Optional **Impact Cart tab** for donors to select multiple campaigns.

**Benefits:** Makes navigation intuitive and allows quick access to the most frequently used features.

**Home Page Layouts**

**Purpose:** Provide role-based dashboards and quick action areas.

**Implementation:**

* **Donor Dashboard:** Highlights loyalty points, badges, and recent donations.
* **Staff Dashboard:** Displays campaign goals, donations received, and beneficiary tracking.

**Benefits:** Ensures that both donors and staff can see **key information immediately** without searching through records.

**Utility Bar**

**Purpose:** Offer quick access tools visible across pages.

**Implementation:**

* Added tools for **Create Donation, Add Beneficiary, and View Campaign Performance**.

**Benefits:** Reduces clicks and improves productivity, allowing staff to perform frequent actions from any page.

**Lightning Web Components (LWC)**

**Purpose:** Create dynamic, interactive components that improve engagement.

**Implementation:**

* **Smart Donor Cart LWC:** Donors can select campaigns and see the total donation instantly.
* **Donor Badge Display LWC:** Shows earned badges and loyalty points.

**Benefits:** Provides a **modern, interactive user experience** for donors and staff. Components update dynamically without page reloads.

**Apex with LWC**

**Purpose:** Handle server-side logic and calculations for dynamic UI components.

**Implementation:**

* Apex Controllers fetch donor history for LWC display and calculate badge eligibility.

**Benefits:** Enables **real-time updates** and ensures complex calculations (like badges or loyalty points) are handled efficiently.

**Events in LWC**

**Purpose:** Allow communication between LWC components for interactive updates.

**Implementation:**

* When a donor adds campaigns to the Impact Cart, the **Badge Display component updates automatically** to reflect potential points earned.

**Benefits:** Provides a seamless, interactive experience and ensures the UI reflects changes immediately.

**Wire Adapters & Imperative Apex Calls**

**Purpose:** Retrieve Salesforce data either reactively (wire) or on-demand (imperative).

**Implementation:**

* **Wire Adapter:** Fetches donor’s past donations and displays in dashboard.
* **Imperative Apex Call:** Calculates total donation value when multiple campaigns are selected in the Impact Cart.

**Benefits:** Optimizes performance and provides accurate, real-time data to the user.

**Navigation Service**

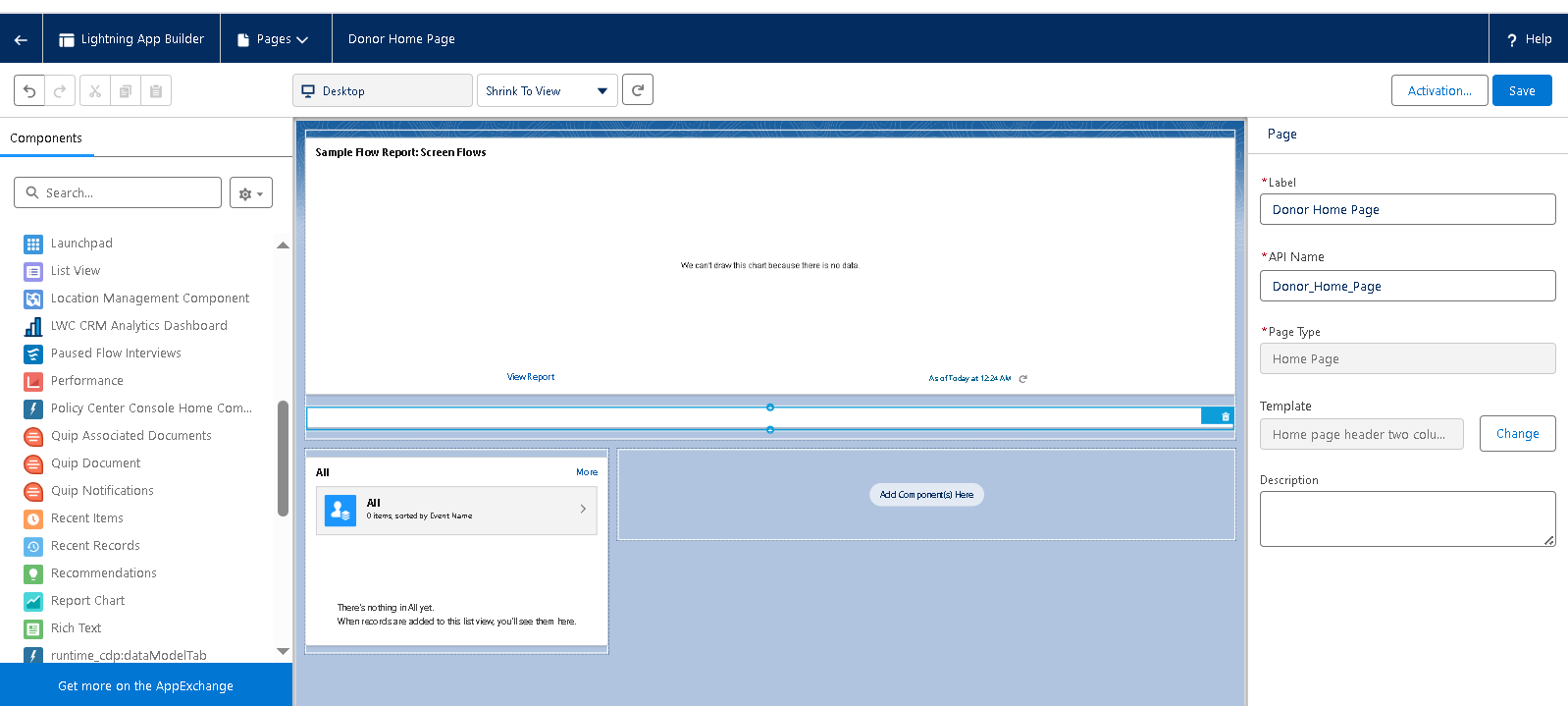
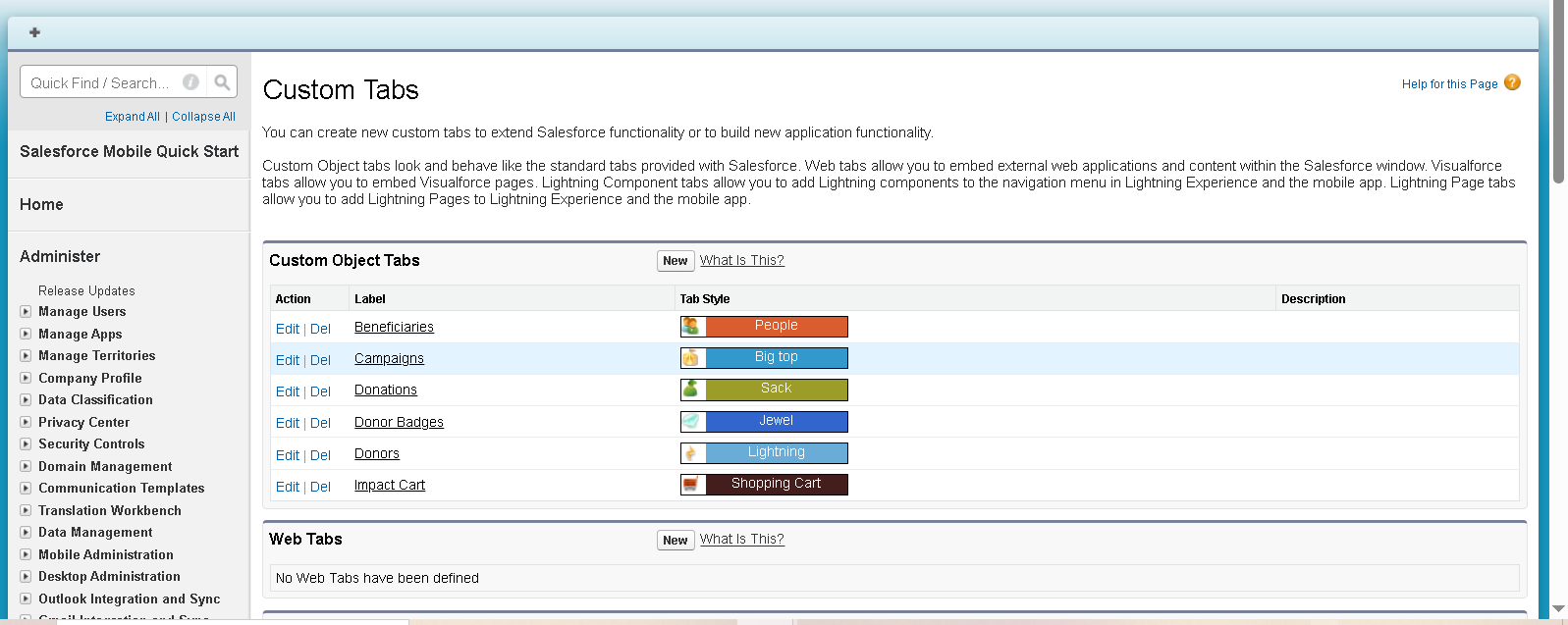
**Purpose:** Programmatically control navigation between Salesforce pages.

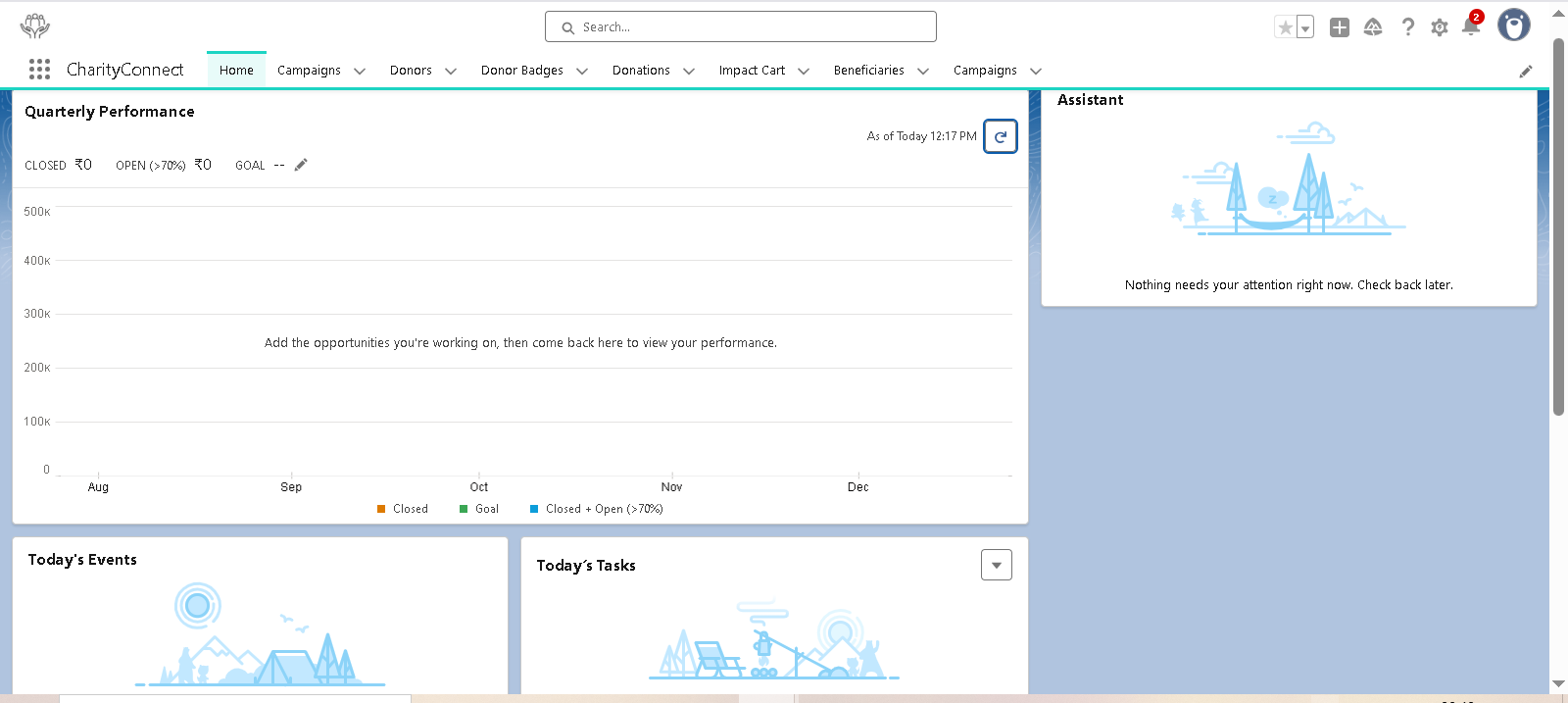
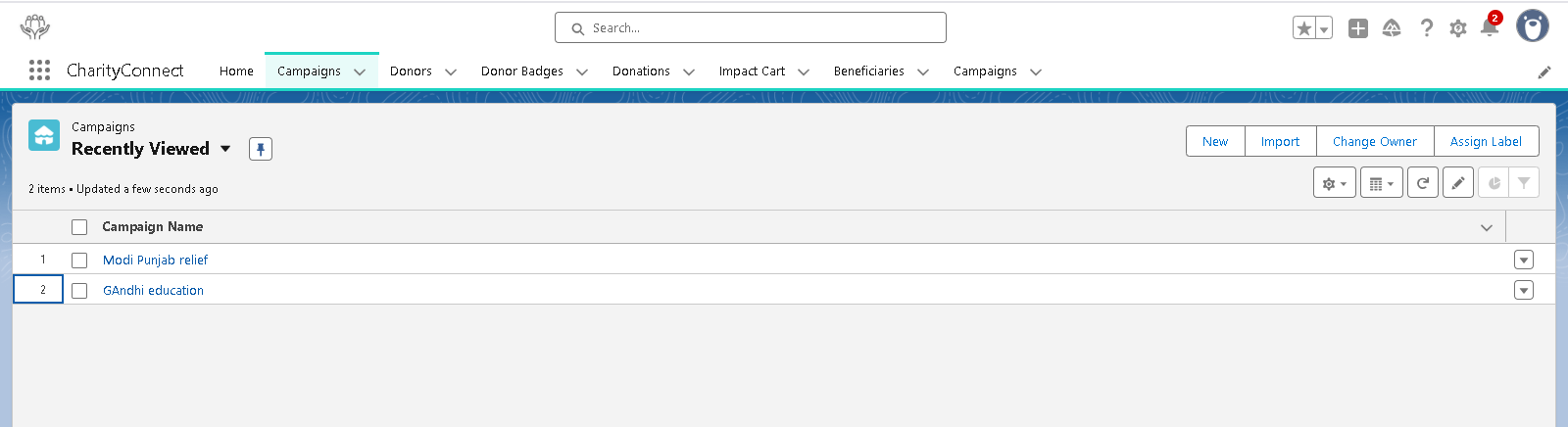
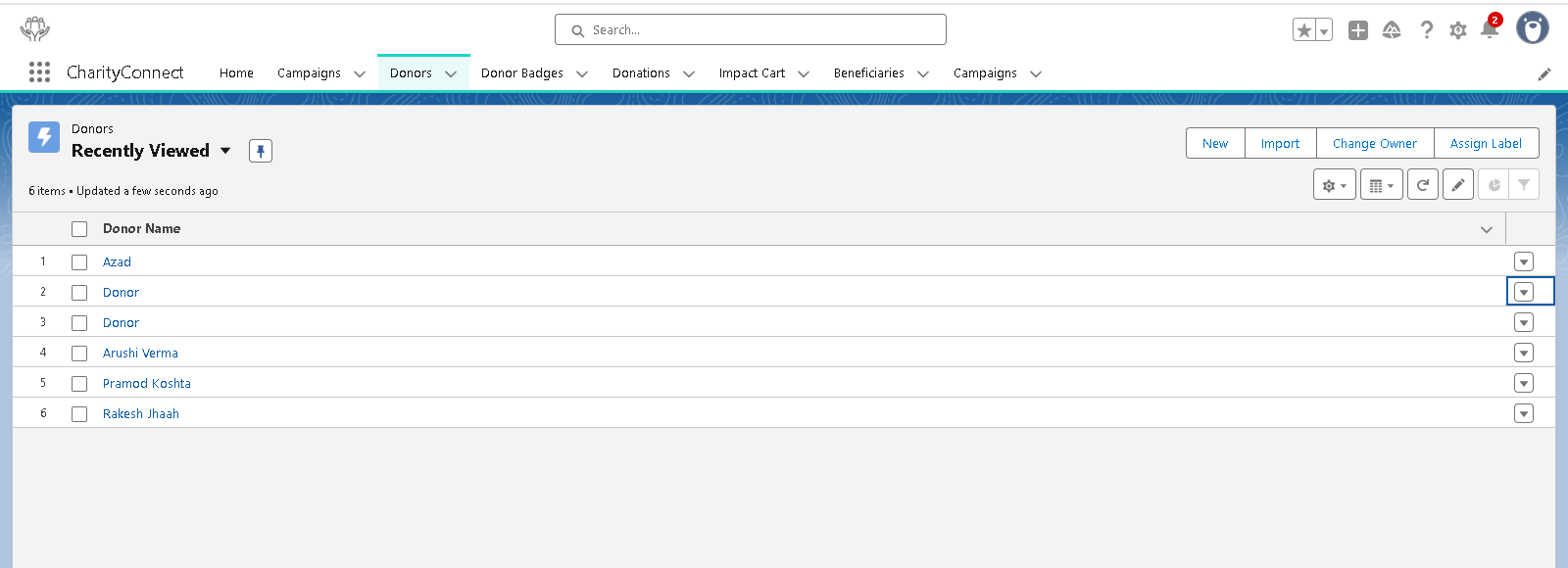
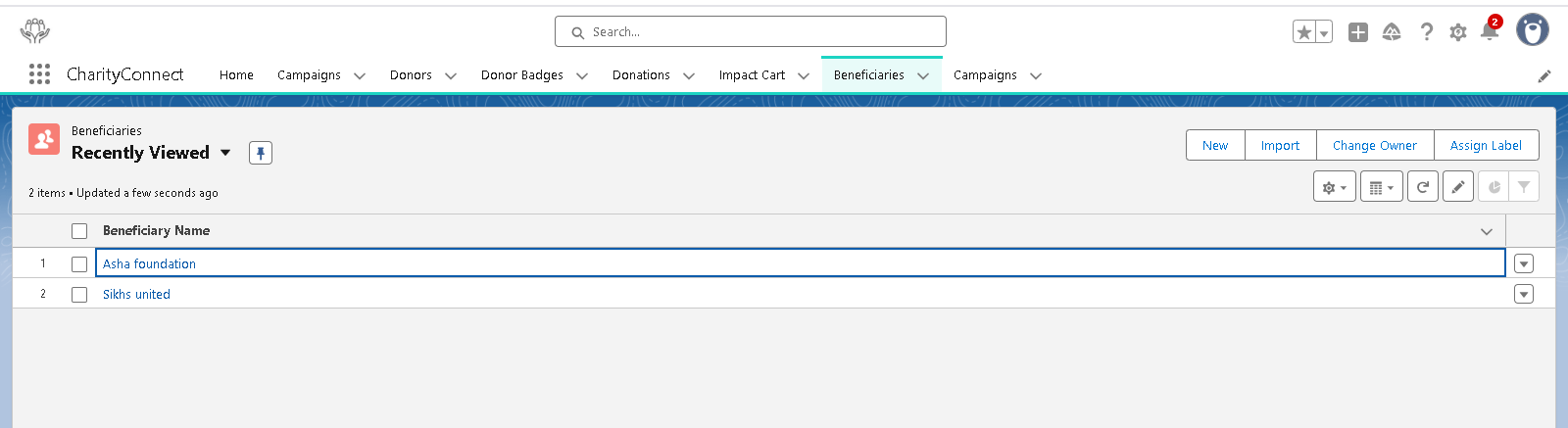
**Implementation:**

* Donor completes Impact Cart → auto-navigate to **Confirmation Page**.
* Staff clicks “Top Donors” widget → navigate directly to Donor Record Page.

**Benefits:** Reduces unnecessary clicks, improves workflow efficiency, and enhances user experience.

**Screenshots:**

**Lightening appTabs**

**App layout  
   **

**Phase 8: Data Management & Deployment**

**Data Import Wizard**

* A **user-friendly, declarative tool** in Salesforce to import small to medium data volumes (up to 50,000 records) via CSV files.
* **Alignment with CharityConnect:**
  + Initial import of donor base with details such as name, email, contact, preferred causes, and historical loyalty points.
  + Import of pre-existing campaigns (Education Drive, Healthcare Relief, Disaster Relief) to set up the org quickly.
* **Benefit:** No coding required; ideal for NGO staff/admins to quickly populate the system without developer intervention.

**Data Loader**

* A **client-based tool** for bulk operations (insert, update, delete, export) capable of handling **millions of records**.
* **Use Case in CharityConnect:**
  + Import historical donation data spanning multiple years.
  + Update existing donation statuses in bulk after payment gateway reconciliation.
  + Export large datasets for audits or reporting.
* **Alignment:** Ensures that large-scale donor and donation information is migrated efficiently and accurately.

**Duplicate Rules**

* Prevents **duplicate records** for Donors, Donations, and Beneficiaries.
* **Mechanism:** Matching criteria like email, phone number, and donor ID ensures data integrity.
* **Use Case:** If multiple NGO staff attempt to import donor data, duplicates are automatically flagged or blocked.
* **Benefit:** Maintains a **clean, trustworthy donor database**, crucial for transparency.

**Data Export & Backup**

* Salesforce provides options for **manual and scheduled data exports** for compliance, auditing, and disaster recovery.
* **Use Case in CharityConnect:**
  + Weekly or monthly export of Donation and Campaign records for **CSR audit and regulatory reporting**.
  + Backup of donor loyalty points and badges to prevent accidental data loss.
* **Benefit:** Provides NGOs with **secure historical records** and peace of mind about data safety.

**Change Sets**

* Declarative method to deploy metadata between Salesforce orgs (Sandbox → Production).
* **Use Case in CharityConnect:**
  + Deploy newly configured Flows for **Badge Assignment** and **Campaign Reporting** from UAT sandbox to live org.
  + Deploy updates to Custom Objects (Donor, Donation, Donor Badge) and Page Layouts without manual recreation.
* **Benefit:** Ensures **consistent configuration and automation** across environments.

**Unmanaged vs Managed Packages**

* **Unmanaged Packages:** Internal deployments; metadata editable after installation.
* **Managed Packages:** For distribution via AppExchange; metadata mostly locked.
* **Use Case in CharityConnect:**
  + Unmanaged Package to deploy custom objects, Flows, and LWC components to another NGO org if needed.
* **Benefit:** Facilitates **reuse and modular deployment** while maintaining flexibility.

**ANT Migration Tool**

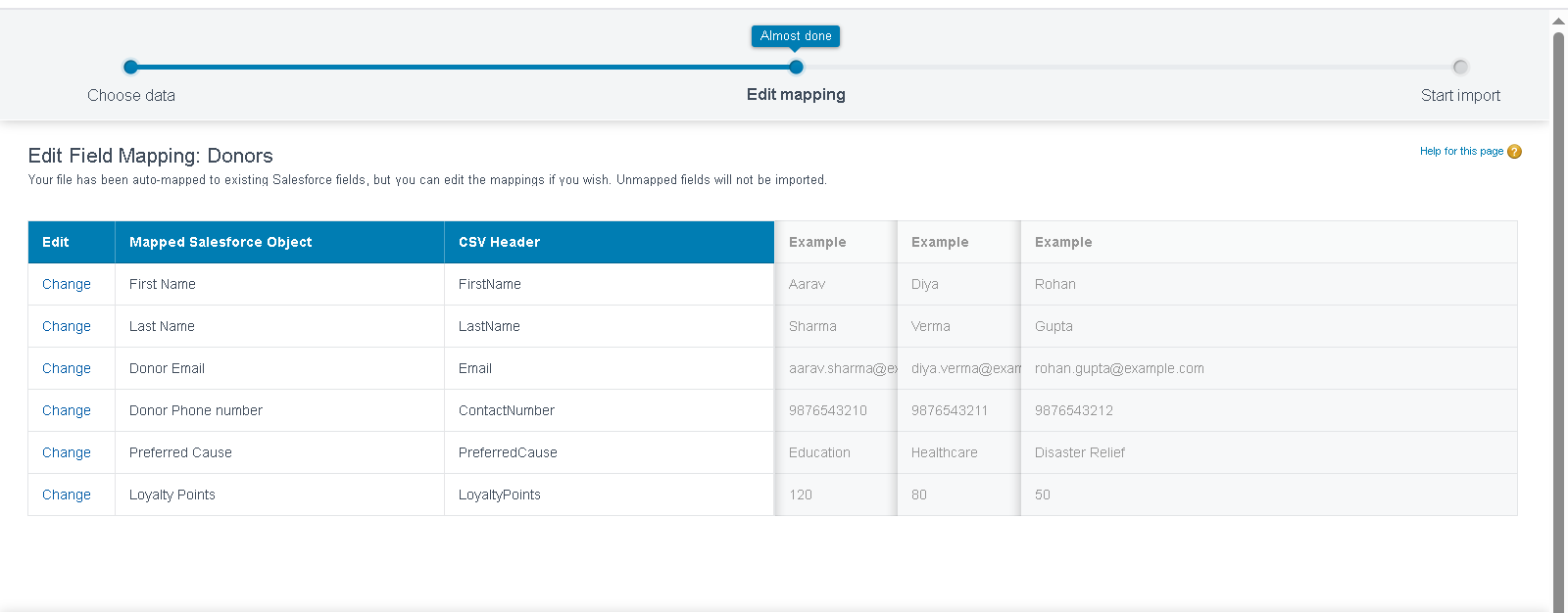
* A **command-line tool** for deploying metadata in a scripted manner, ideal for **advanced deployments or CI/CD pipelines**.
* **Use Case in CharityConnect:**
  + Automate migration of complex metadata like **Donation Flows, Badge Assignment logic, and Apex triggers** from Sandbox to Production.
* **Benefit:** Reduces human errors and speeds up deployment of large-scale projects.

**VS Code & Salesforce DX (SFDX)**

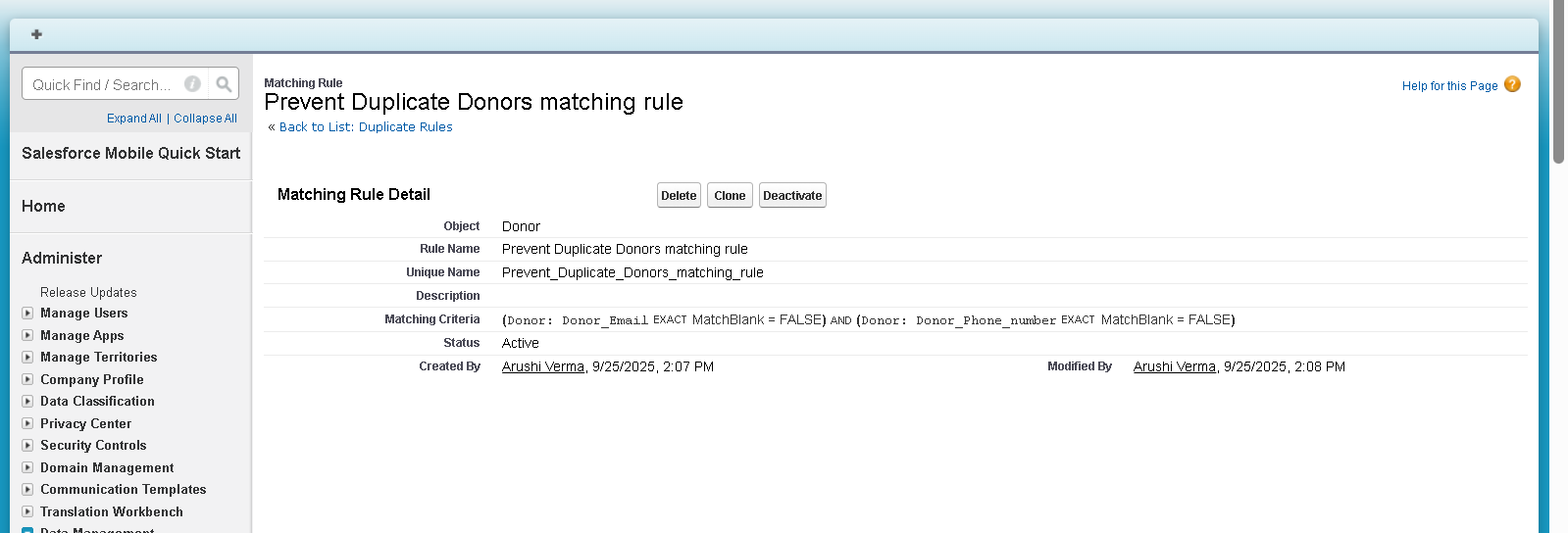
* Provides a **developer-centric environment** for building, version controlling, and deploying Salesforce components.
* **Use Case in CharityConnect:**
  + Developers create Apex triggers for Donation and Badge logic in **scratch orgs**, test locally, then deploy using SFDX.
  + Version control via GitHub ensures changes are tracked, collaborative, and reversible.
* **Benefit:** Supports **agile development**, safe testing, and continuous deployment for NGO projects.

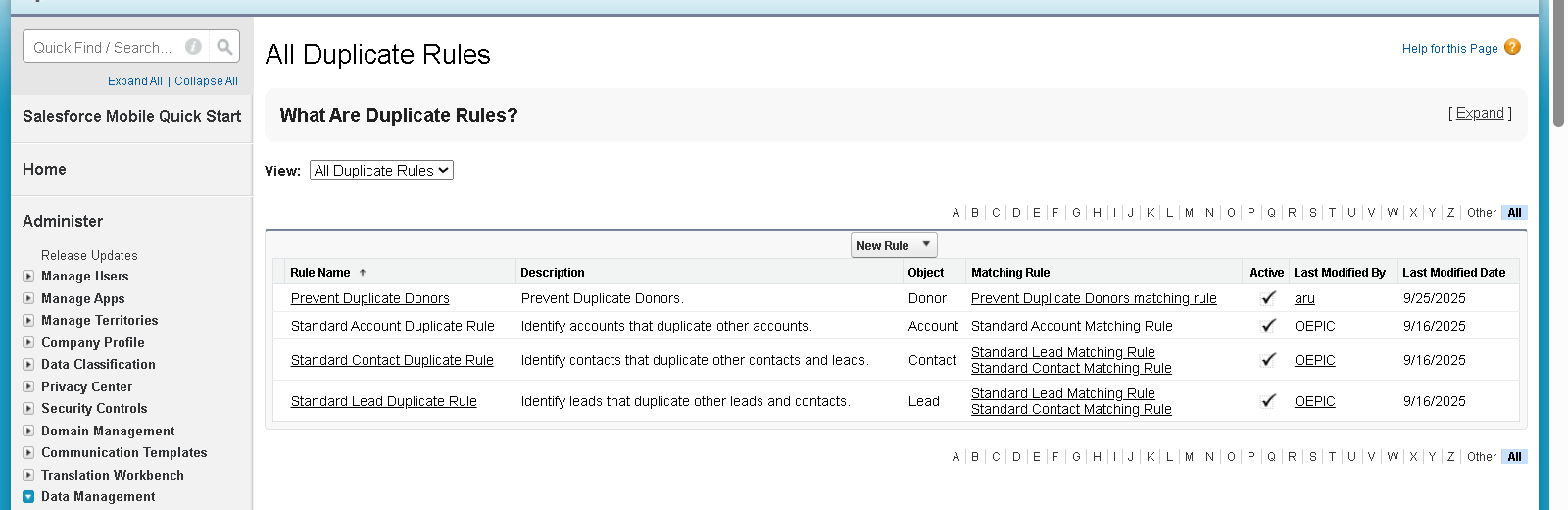
**Screenshots:**

**Data wizard**

****

**duplicate rule**





**Phase 9: Reporting, Dashboards & Security Review**

**1. Reports (Tabular, Summary, Matrix, Joined)**

* Salesforce reports allow **analyzing and visualizing data** across objects.
* **Types & Use Cases:**
  + **Tabular:** Simple list of donations → used for donor mailing lists or campaign audits.
  + **Summary:** Group donations by Campaign → shows total amount raised per campaign.
  + **Matrix:** Donations by Donor vs Campaign → visualizes contribution patterns.
  + **Joined:** Combines multiple objects → e.g., Donor + Donation + Beneficiary for transparency reports.
* **Benefit:** Helps NGOs **track donation trends, monitor campaign performance, and ensure transparency to donors**.

**2. Report Types**

* Define **relationships between objects** for reporting purposes.
* **Use Case in CharityConnect:**
  + Custom Report Type: Donation → Campaign → Beneficiary → Donor.
  + Enables staff to generate **impact reports** showing exactly how donor contributions are allocated.
* **Benefit:** Provides **flexible reporting options** for internal audits and donor updates.

**3. Dashboards**

* Visual representation of key metrics using **charts, tables, and gauges**.
* **Use Case:**
  + Campaign Progress Dashboard: Total amount raised vs goal, active campaigns, top donors.
  + Donor Engagement Dashboard: Loyalty points, badges earned, recurring donations.
* **Benefit:** Allows NGO staff to **monitor real-time performance** and make informed decisions.

**4. Dynamic Dashboards**

* Dashboards display **data personalized to the logged-in user**.
* **Use Case:**
  + Donor Portal: Donors see **only their own contributions and badges**.
  + NGO Manager: Sees **all campaigns, donations, and beneficiary details**.
* **Benefit:** Ensures **data privacy** while providing tailored insights.

**5. Sharing Settings**

* Control **who can see or edit records** in Salesforce.
* **Use Case:**
  + Donations: Private by default → only donor & NGO Staff can view.
  + Campaigns: Public Read-Only for all NGO staff → allows transparency without risk of modification.
* **Benefit:** Protects sensitive donor data while enabling collaboration among staff.

**6. Field Level Security (FLS)**

* Controls visibility and edit access to specific fields for different profiles.
* **Use Case:**
  + Donor email & payment information → visible to NGO Staff, hidden from Donor Users.
  + Loyalty Points & Badge Status → visible to both staff and donor portal.
* **Benefit:** Ensures **data confidentiality and regulatory compliance**.

**7. Session Settings**

* Configure **session timeouts, login history, and security policies**.
* **Use Case:**
  + Donor portal session timeout set to 30 minutes of inactivity.
  + Staff portal session timeout: 1 hour with automatic logout.
* **Benefit:** Enhances **data security**, reduces risk of unauthorized access.

**8. Login IP Ranges**

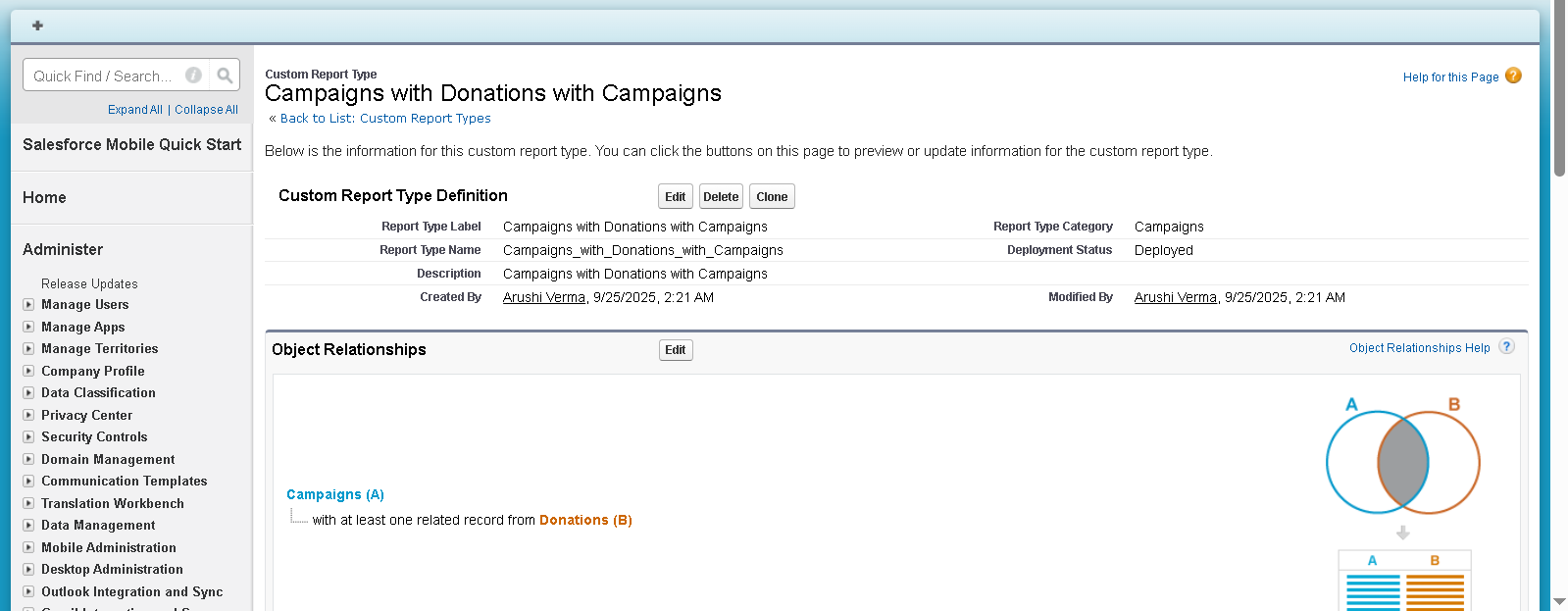
Restricts access to Salesforce org to **specific IP addresses**.

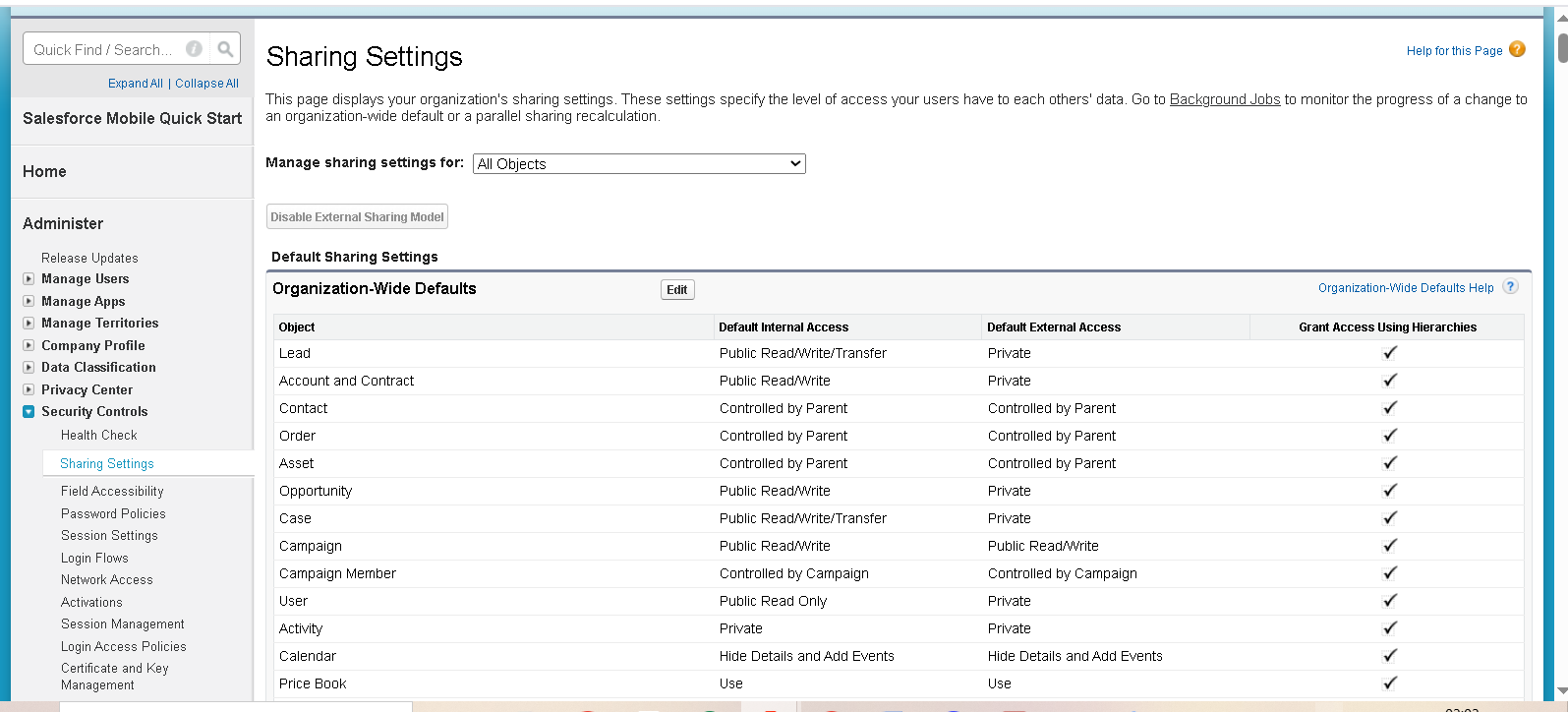
* **Use Case:**
  + NGO staff can log in only from office or verified home IP addresses.
* **Benefit:** Adds **network-level security** to prevent unauthorized logins.

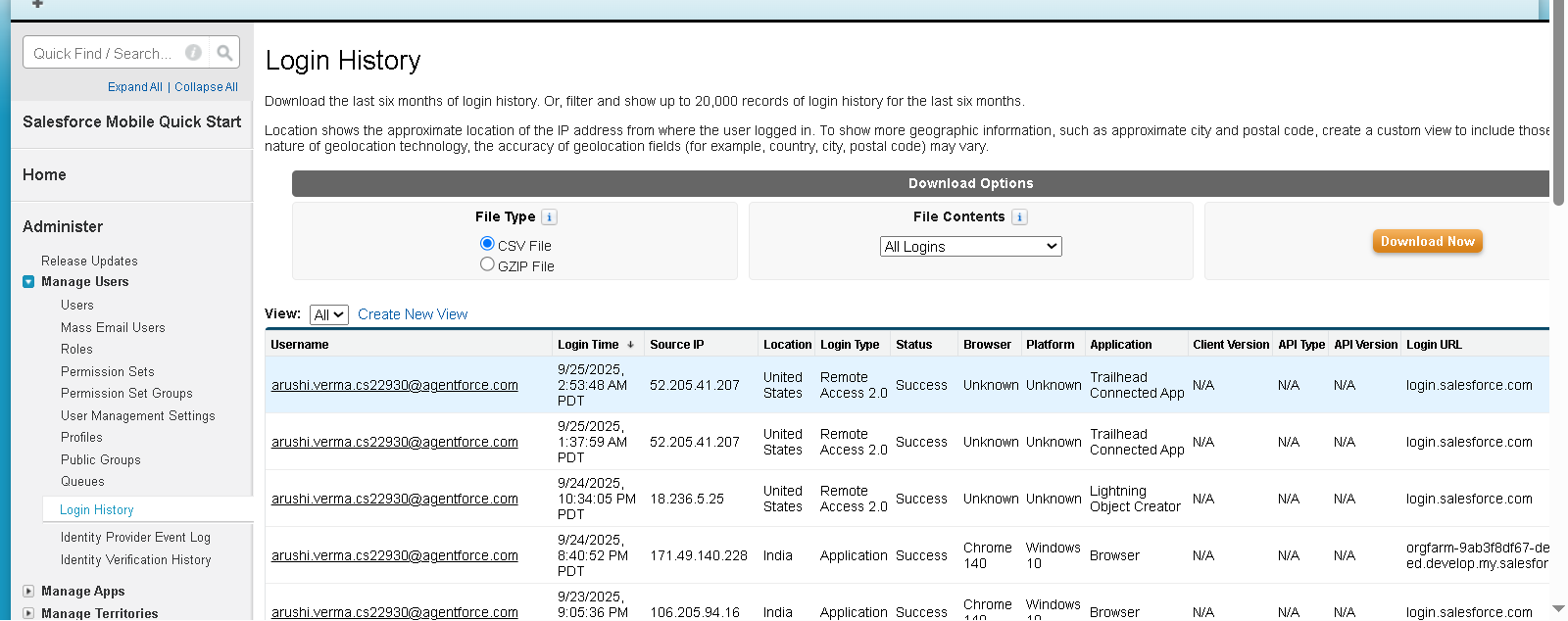
**9. Audit Trail**

* Tracks all **configuration changes and system modifications** in Salesforce.
* **Use Case:**
  + Track updates to Campaigns, Donations, or Beneficiary records.
  + Identify who modified a donor record and when.
* **Benefit:** Provides **accountability, transparency, and compliance** for NGO reporting.

**Screenshots:**

**Custom report types** 

**Sharing settings**

**Login history** 

**Phase 10: Final Presentation & Demo Day**

**1. Pitch Presentation**

* **Slide Content:**
  + **Problem:** Low donor trust, scattered donation records, no reward system.
  + **Solution:** CharityConnect CRM
    - **Smart Donor Cart:** Select multiple campaigns; total auto-calculated.
    - **Loyalty Points & Badges:** Rewards gamify donations.
    - **Transparency Dashboards:** Donors see how contributions are utilized.
    - **Email Alerts:** Automated thank-you & large donation notifications.
  + **Implementation Phases:** Phase 1–9 summary, showing Admin + Developer tasks.
  + **Impact:** Improved donor engagement, efficiency in campaign management.
* **Visuals:**
  + Screenshot of donor portal showing cart and badges.
  + Dashboard showing campaign progress.
  + Flow diagram of donation → donor → campaign → beneficiary process.

**2. Demo Walkthrough**

* + - The demo video link is as follows:  
      https://youtu.be/x3Ka60TnkY4

**3. Feedback Collection**

* **Methods:** Google Form + verbal interview.
* **Questions:**
  + Did donor portal provide clear transparency on fund usage?
  + Was the Smart Cart experience intuitive?
  + Are loyalty points and badges motivating?
  + Are reports and dashboards easy to interpret?
* **Outcome:** Feedback used to refine Flows, dashboard layout, and UI.

**4. Handoff Documentation**

* All the documentation of all the phases is shared on the github link

**5. LinkedIn / Portfolio Project Showcase**

**https://github.com/Aarushi2772/TCS\_Salesforce\_CharityConnect\_CRM\_project/tree/main**