**SOFTWARE DESIGN DOCUMENT**

**PROJECT TITLE:** ONLINE DONATION SYSTEM

**1. Purpose:**

The purpose of this document is to define the software requirements for the Online Donation System. This document will serve as a guideline for the design, development, testing, and maintenance of the system. It is intended for developers, testers, project managers, and stakeholders.

**2. Problem Statement:**

In today’s digital world, charitable organizations and social causes need a streamlined and secure way to collect donations from a broad range of donors. Traditional donation systems often rely on manual processes, limited payment options, and lack transparency and donor engagement features. This creates inefficiencies in fundraising and discourages repeat contributions.

There is a need for a web-based Online Donation System that:

* Enables individuals to contribute quickly and securely to various campaigns.
* Offers modern features like recurring donations, anonymous giving, and progress tracking.
* Provides a robust backend for campaign management and reporting.

This system will bridge the gap between donors and organizations by making fundraising more accessible, transparent, and automated.

**3.System Overview:**

The Online Donation System is a web-based platform that connects donors, campaign managers, and beneficiaries through a secure and user-friendly interface. It facilitates charitable donations via campaigns, ensures transparency in fund allocation, and provides real-time reports and tracking for

all stakeholders. The system supports user registration, campaign creation, secure payments, and administrative monitoring**.**

**4. Design Considerations:**

1. Scalability: The system must handle increasing users and donation traffic without performance loss.

2. Security: All sensitive data (user credentials, payment info) must be encrypted (e.g., HTTPS, TLS).

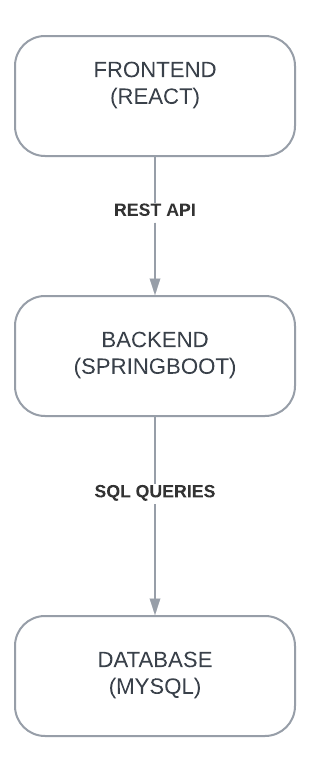
3. Usability: Simple and intuitive UI for donors and campaign managers.

4. Modularity: The system should be developed in modules for maintainability and reusability.

5. Availability: Ensure system uptime of at least 99.9%.

6. Accessibility: Should follow accessibility standards (WCAG 2.1) for broader user reach.

**5. System Architecture:**



**6. DATA DESIGN:**

Key Tables:

* Users: id, name, email, password\_hash, role
* Campaigns: id, title, description, target\_amount, current\_amount, start\_date, end\_date, manager\_id, status
* Donations: id, user\_id, campaign\_id, amount, date, payment\_status
* Beneficiaries: id, name, campaign\_id, allocated\_fund
* Reports: id, campaign\_id, report\_details, created\_at

**7. COMPONENT DESIGN:**

Frontend (React):

* Pages: Home, Login, Register, Dashboard, Campaign Details, Donate, Reports
* Components: Navbar, CampaignCard, DonationForm, Profile, ReportChart

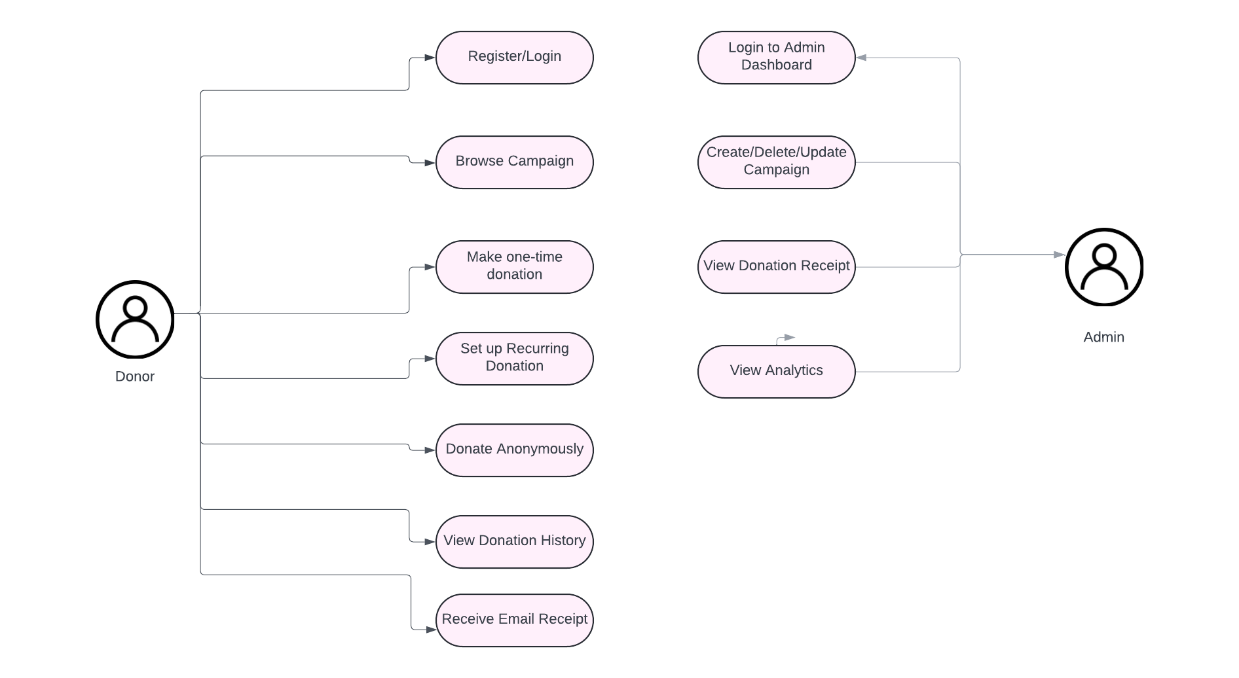
Backend (Node.js or Django):

* APIs:
  + /api/register, /api/login, /api/campaigns, /api/donate, /api/reports
* Controllers:
  + AuthController, CampaignController, DonationController, ReportController
* Middleware: Authentication, Role-Based Access Control, Input Validation

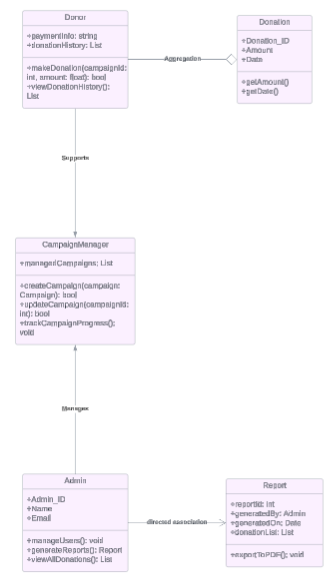
Database (MySQL/PostgreSQL):

* Designed with normalized schema for efficiency.
* Uses foreign keys for relationship integrity.

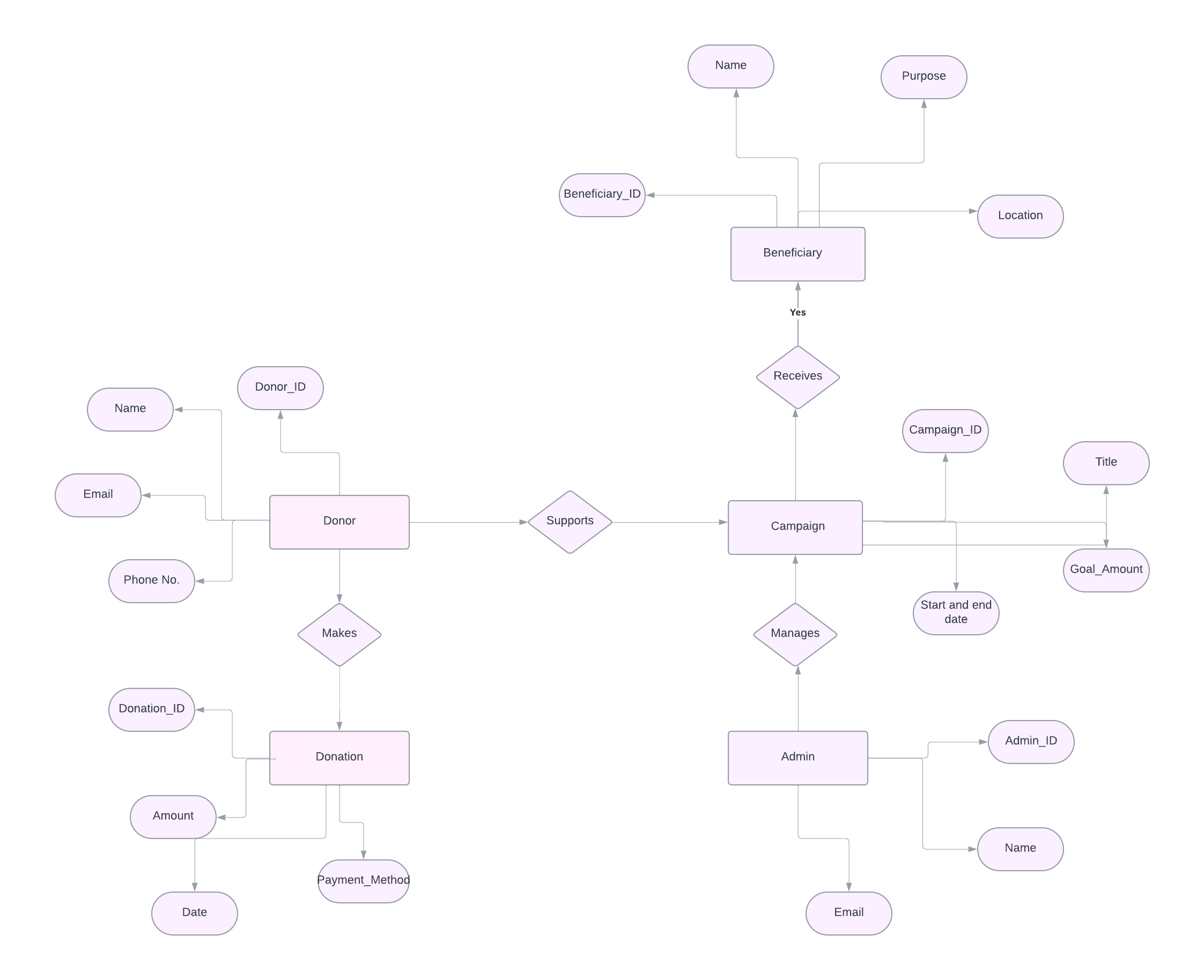
**8. USE CASE DIAGRAM:**



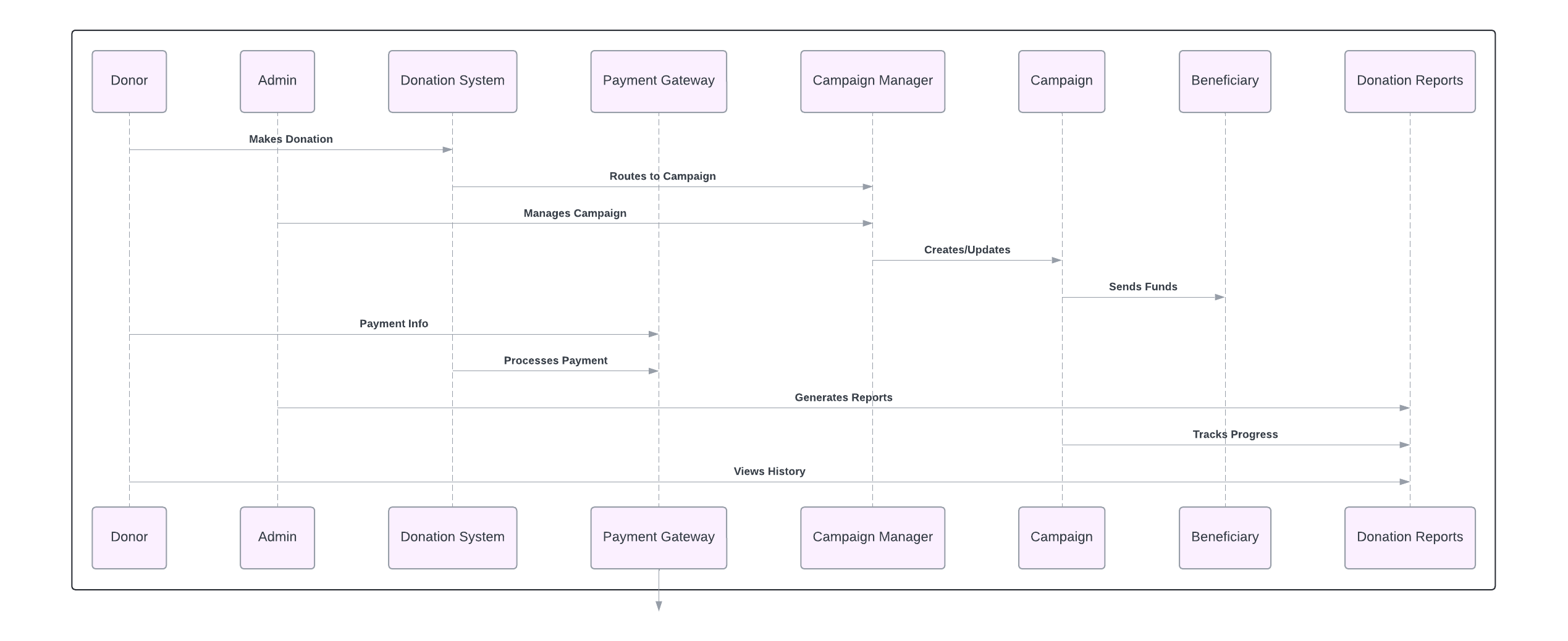
**9. CLASS DIAGRAM:**

****

**10. E-R DIAGRAM:**



**11.Sequential Diagram:**



**12. Conclusion:**

The Online Donation System is designed to streamline the process of charitable giving by providing a secure, transparent, and user-friendly platform for donors, campaign managers, and beneficiaries. Through its modular architecture and well-defined components, the system ensures scalability, maintainability, and ease of integration with third-party services like payment gateways**.**