

# Requirements Specification Document

(Group : 6)

Annamitra

## 1. Functional & Non-Functional Requirements

### Functional Requirements (What the system will do)

These directly come from your described flows — the **donation lifecycle** + dashboards.

#### 1.1 User Registration & Authentication

- Users can **sign up** and **log in** using email/OTP.
- Users choose their **role** (Donor, NGO, Volunteer, Admin).
- NGO accounts require **document verification** by Admin.
- Auth implemented via **Passport local strategy**.

#### 1.2 Donation Management (Donor Side)

- Donor can **create donation posts** with:
  - Food items, quantity, best-before time.
  - Pickup location and time window.
  - Notes/special instructions.
- Donor can **view their donations list** with statuses:
  - Open, Claimed, In Pickup, Delivered, Expired/Cancelled.
- Donor receives **real-time notifications** on claim, pickup, delivery.
- Donor can **rate NGOs/Volunteers** after successful delivery.

#### 1.3 NGO Management

- NGO can **view nearby open donations** based on location.
- NGO can **request/claim donations**.
- NGO can **assign volunteers** for pickup.
- NGO confirms **final delivery** once food is received.
- NGO can view compliance checklist and past donations.

#### 1.4 Volunteer Management

- Volunteers can **view assigned donations** or accept open pickups.
- Stepper flow for pickup process:
  - Start → At Donor → Picked → At NGO → Delivered.
- Volunteers must **upload proof of delivery** (photo + OTP/signature).

#### 1.5 Admin Dashboard

- Admin can **verify NGOs and users**.
- Can **monitor donations** and handle disputes.
- Can create other admin accounts.
- Can view system metrics (donations completed, users active, etc.)

#### 1.6 Notification System

- Real-time alerts for:
  - NGO when donation is posted.
  - Donor when claim/pickup/delivery happens.
- Supports Email + Firebase push notifications.

#### 1.7 Proof of Delivery

- Volunteer uploads **photo** and **OTP/signature** from recipient.
- System automatically **marks donation as delivered** after proof.

#### 1.8 Feedback & Rating

- Donor ↔ NGO can give star ratings and comments.
- Ratings contribute to NGO/Volunteer reputation.

## Non-Functional Requirements (How the system will *behave*)

Category	Requirement
<b>Performance</b>	Publish → notify latency < 5s; system handles 100+ concurrent users.
<b>Availability</b>	Minimum uptime 99.5% during peak donation hours.
<b>Security</b>	Passport local strategy-based authentication, hashed passwords, encrypted user data.
<b>Privacy</b>	Donors can hide contact info; phone masking for communication.
<b>Usability</b>	UI simple enough for semi-tech users (NGOs/Volunteers).
<b>Localization</b>	English, Hindi, Gujarati supported.
<b>Accessibility</b>	Large buttons, readable contrast, mobile-friendly layout.
<b>Scalability</b>	Modular backend (Node + Express + DB) for easy future expansion.
<b>Maintainability</b>	Clean code structure with separated frontend/backend repos.

## 2. Detailed User Stories / Use Case Diagrams

### User Stories (Agile format)

#### Donor

1. As a *Donor*, I want to create a donation post so that nearby NGOs can request it.
2. As a *Donor*, I want to see my donation status (Open, Claimed, Delivered) so I can track progress.
3. As a *Donor*, I want notifications when an NGO claims or delivers my food.
4. As a *Donor*, I want to rate the NGO/Volunteer after successful delivery.

#### NGO

1. As an *NGO*, I want to view nearby open donations so I can claim them.
2. As an *NGO*, I want to assign volunteers for pickup and delivery.
3. As an *NGO*, I want to confirm delivery and upload proof to maintain transparency.

#### Volunteer

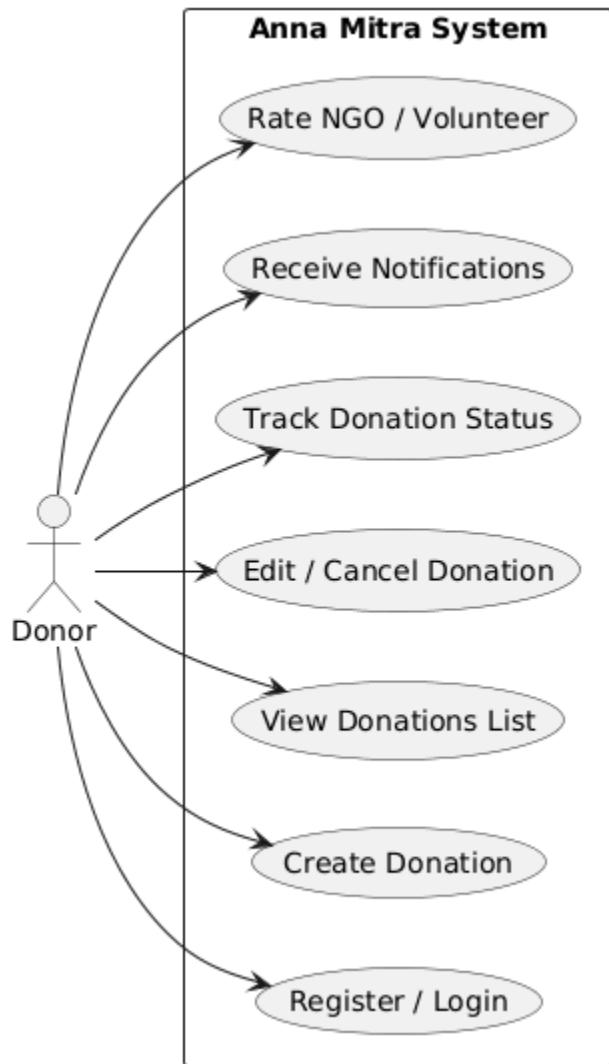
1. As a *Volunteer*, I want to see my assigned pickups and drop-off points on map.
2. As a *Volunteer*, I want to update my status step-by-step (Picked, Delivered).
3. As a *Volunteer*, I want to upload delivery proof (photo/OTP) for record keeping.

#### Admin

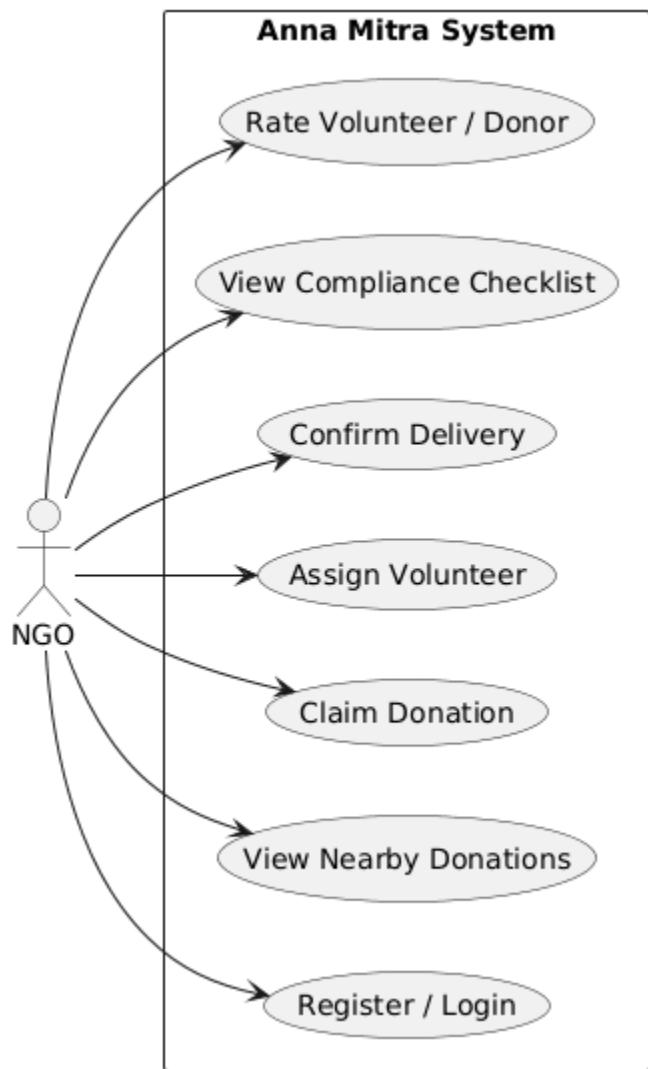
1. As an *Admin*, I want to verify NGOs and manage users.
2. As an *Admin*, I want to monitor all donations and metrics.
3. As an *Admin*, I want to handle complaints or expired donations.

## USE CASES:

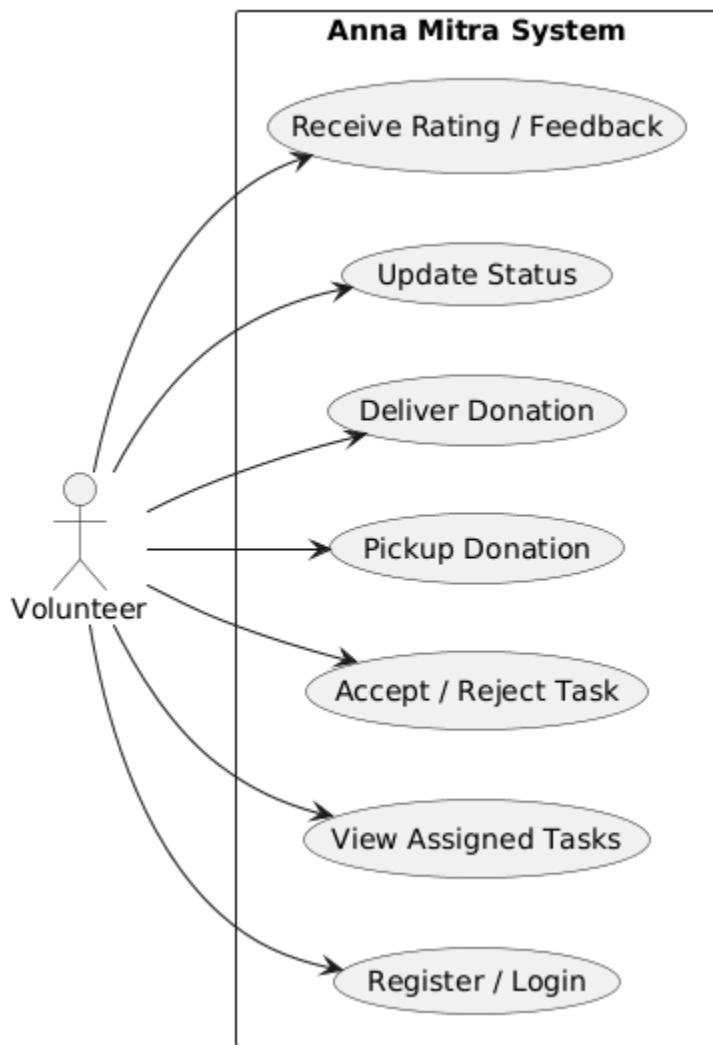
DONOR



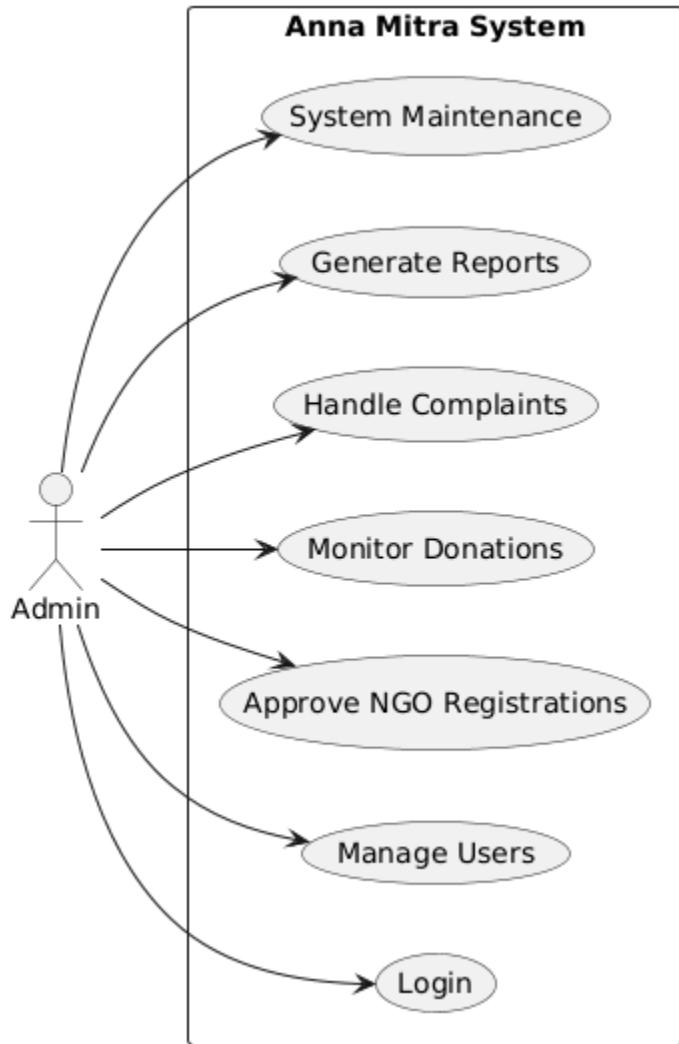
NGO:



VOLUNTEER:



ADMIN:



### 3. Interface Requirements

Interface	Description
<b>Login/Signup Screen</b>	Role-based registration (Donor, NGO, Volunteer, Admin). OTP/email verification.
<b>Home Page</b>	Public view showing mission, stats, testimonials, CTA to “Donate Food”.
<b>Create Donation Form</b>	Donor enters food details, quantity, pickup location, best-before time, and contact info.
<b>Donation List Page</b>	Displays all donations with status filters (Open, Claimed, Delivered).
<b>Donation Details Page</b>	Full info, map, and action buttons (Claim, Pickup, Deliver).
<b>NGO Dashboard</b>	List of nearby donations; option to assign volunteer.
<b>Volunteer Dashboard</b>	Shows assigned pickups with map and timeline.
<b>Admin Dashboard</b>	User verification, donations overview, and system metrics.
<b>Notifications Panel</b>	Real-time updates for donation lifecycle.

## 4. Wireframes / UI-UX Mockups (Initial Draft)

(Since actual images aren't available, description given.)

Screen	Key UI Elements
<b>Home Page</b>	Header (Logo, About, Contact, Login); Main banner “Donate Food, Feed Hope”; Stats cards (Donations saved, Meals served); CTA button.
<b>Login Page</b>	Email + Password fields; Role selection dropdown; Submit button.
<b>Donor Dashboard</b>	Button “+ Create Donation”; Card list of donations (status color-coded); Sidebar: Profile, History, Logout.
<b>Create Donation Page</b>	Form fields: Item name, quantity, best-before, packaging type, location map picker; Save/Publish buttons.
<b>NGO Dashboard</b>	Table/List of available donations; “Claim” button; status indicators; Assign volunteer modal.
<b>Volunteer Dashboard</b>	List of tasks with status (Pending, Picked, Delivered); “View Map” and “Upload Proof” buttons.
<b>Admin Dashboard</b>	Tabs for Users, Donations, NGOs, Reports, inquiries;

UI style is **clean, minimal**, with **green and white color palette** (eco/sustainability theme).

## 5. Acceptance Criteria (For Key Features)

Feature	Acceptance Criteria
<b>User Registration</b>	User can successfully register with valid email/OTP and role. Invalid OTP → error shown.
<b>NGO Verification</b>	Admin approves/rejects NGO within dashboard; unverified NGO cannot claim donations.
<b>Create Donation</b>	Donor can create post with at least 1 item, location, and pickup window.
<b>Claim Donation</b>	NGO can claim open donation only once; donation status changes to “Claimed”.
<b>Volunteer Assignment</b>	NGO can assign a volunteer from list; volunteer gets notification.
<b>Pickup &amp; Delivery</b>	Volunteer updates each stage; proof upload required before “Delivered” status.
<b>Notifications</b>	Donor gets instant alerts on each lifecycle event (Claim, Pickup, Delivered).
<b>Proof Upload</b>	Delivery photo must be submitted; otherwise cannot close donation.
<b>Rating System</b>	After delivery, donor/NGO/volunteer can rate each other once per transaction.
<b>Admin Dashboard</b>	Admin can view all active donations and verified NGOs without app errors.