**SENG 244 - Object Oriented Software Engineering**

**Software Analysis Report**

**NGO Aid Operations Management System:**

**NGO-AOMSYS**

**Group Members:**

**Doğa Ömrüuzun**: 210201027

**İrem Sena Alpak**: 210201030

**Melis Gedik**: 220201027

**Ayşe Asude Doğan**: 220201044

**Ayşe Aybala Girginol**: 220204014

* **1.Introduction**
* **2.Overview(Domain Analysis)**

2.1 Main Task 1

2.2 Main Task 2

* **3.Functional Requirements**
* **4.Nonfunctional Requirements**
* **5.System Models**

5.1 Use Case Model

5.2 Class Model

5.3 Sequence Model

5.4 State Chart Model

5.5 User Interface

5.5.1 Login Interface

5.5.2 Donor Registration Interface

5.5.3 Volunteer Registration Interface

5.5.4 Aid Apllication Interface

5.5.5 Donation Interface

5.5.6 Admin Panel-User Management Interface

5.5.7 Admin Dashboard Interface

* **6. Glossary**

**1.INTRODUCTION**

Non-Governmental Organizations (NGOs) has an importance of management of aid, and reaching the indigents that need the aid the most.

NGO Aid Operations Management System (NGO-AOMSYS) is developed to meet these needs of NGOs. This project will facilitate the management of aid.

**2.OVERVEW(DOMAIN ANALYSIS)**

**2.1 MAIN TASK 1**

The proposed system, NGO Aid Operations Management System (NGO-AOMSYS), is a software project designed to facilitate the management of aid by non-governmental organizations (NGOs).

**2.2 MAIN TASK 2**

This system will manage donations, register aid recipients and plan aid. Additionally, the system will facilitate communication between donors, volunteers, recipients and coordinators

**3. FUNCTIONAL REQUIREMENTS**

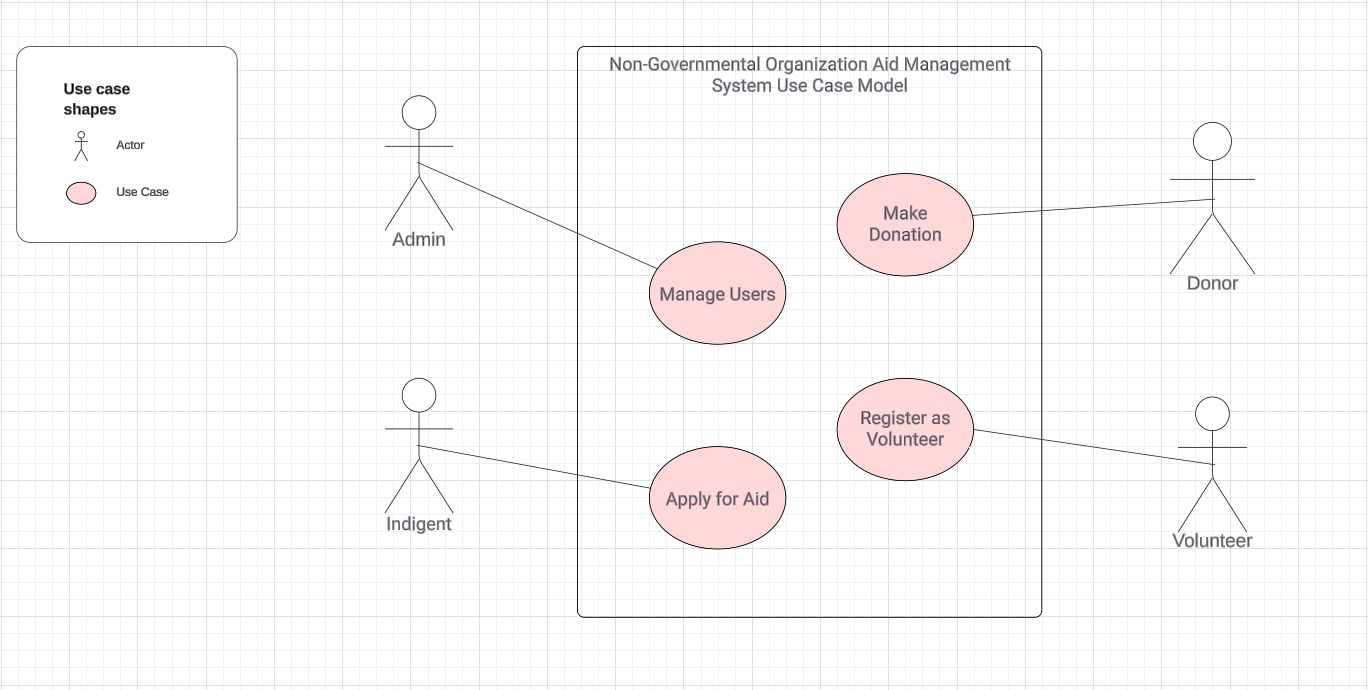
* Donors shall be able to register with the system and create accounts.
* Indigents shall be able to register with the system and create accounts.
* Volunteers shall be assigned tasks based on their profiles.
* Indigents shall be able to apply for aid.
* Donors shall be able to log in to their accounts to view donation history.
* Donors shall be able to select the area, project and amount and type of donation.
* Volunteers shall provide personal information, professional details, availability, and willingness to provide transportation.
* Volunteers shall be assigned tasks based on their profiles.
* Administrators shall review and approve volunteer registrations.
* Administrators shall have functionalities to manage user accounts and system settings.
* Operation coordinators shall have access to a dashboard to review aid requests and manage operations.
* Operation coordinators shall categorize aid operations into collecting donations, distributing aid materials.
* The system shall provide communication between donors, volunteers,indigents, and operation coordinators through messaging or notifications.
* The system shall generate reports on donation history, volunteer activities, aid distribution.

**4. NONFUNCTIONAL REQUIREMENTS**

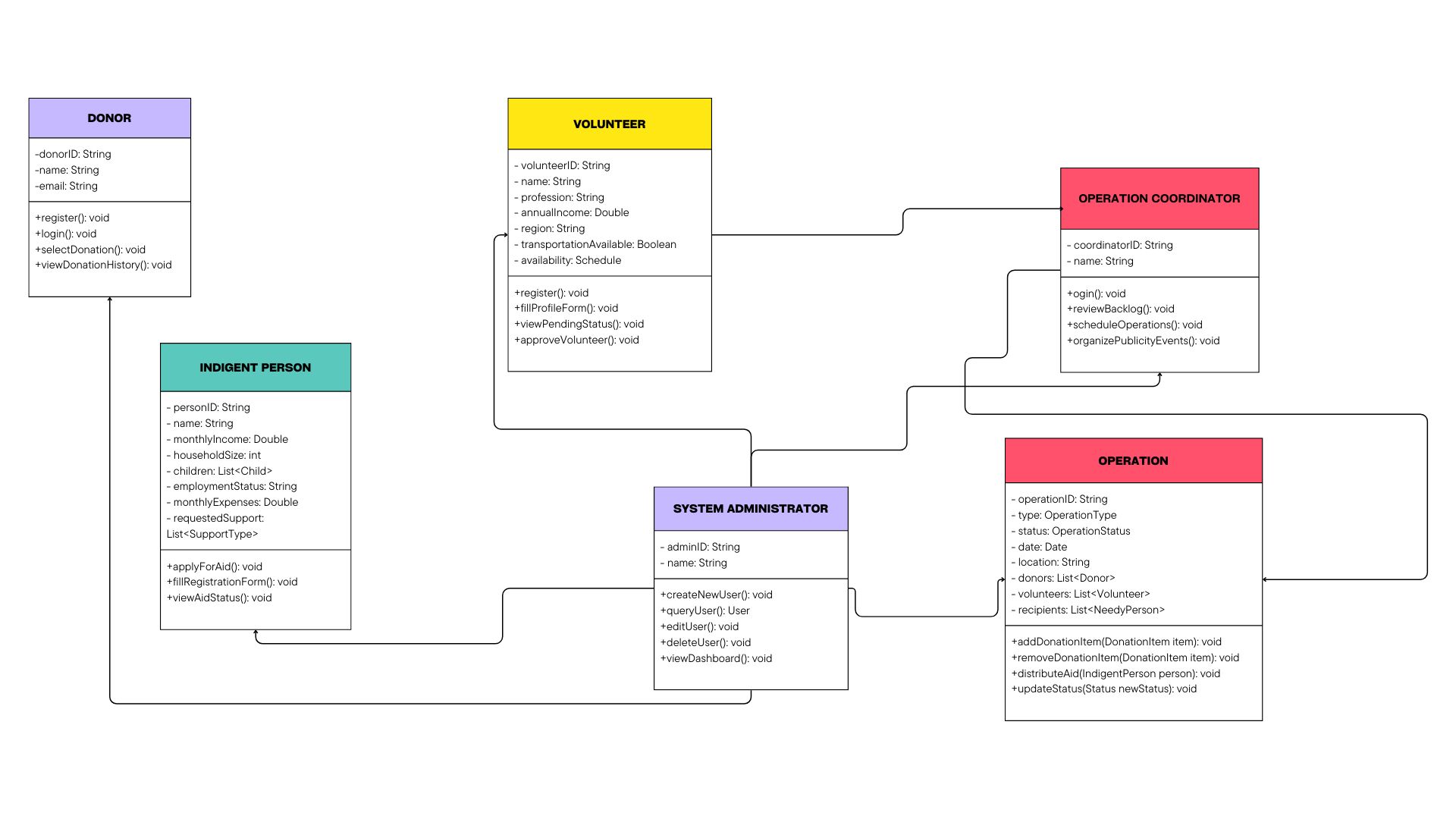
* The system shall be highly reliable, with minimal downtime.
* The system shall be compatible with commonly used web browsers (e.g., Chrome, Firefox, Safari) and operating systems (e.g., Windows, macOS, Linux).
* Response time for user actions, such as donation submissions and volunteer registrations, shall be within acceptable limits.
* System creates custom font size for user that may have vision impairment

**5. SYSTEM MODELS**

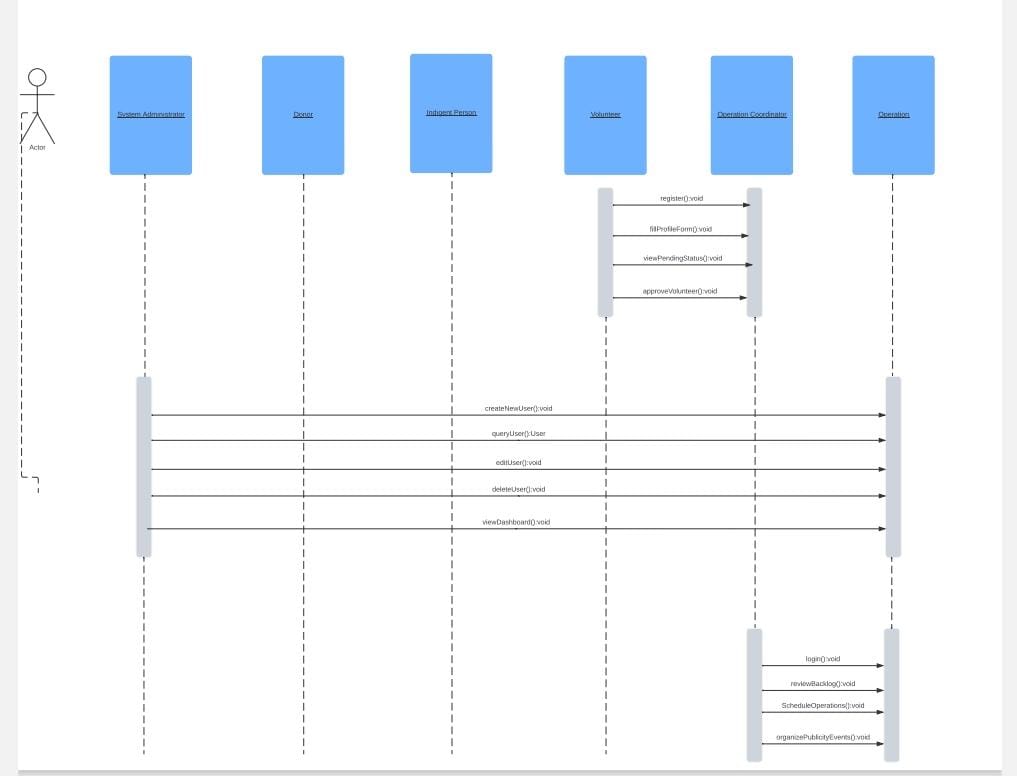
**5.1 Use Case Model**



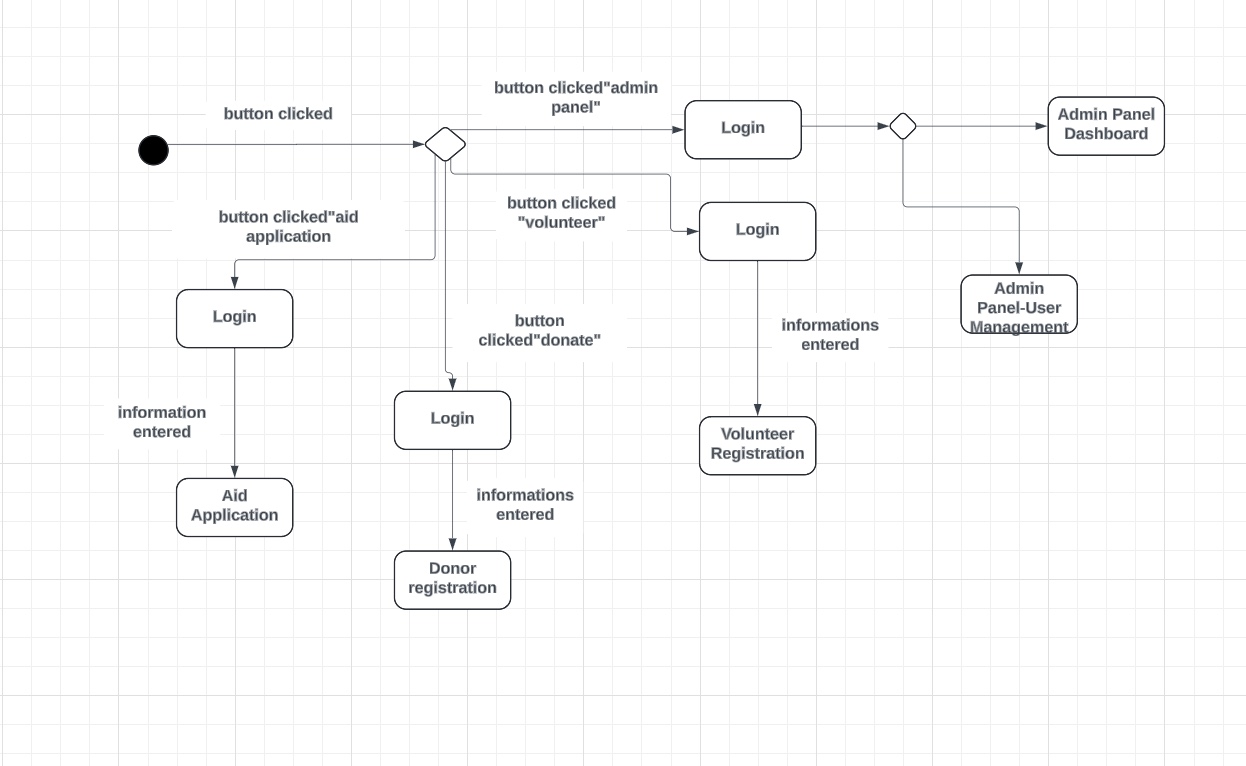
**5.2 Class Model**



**5.3 Sequence Model**

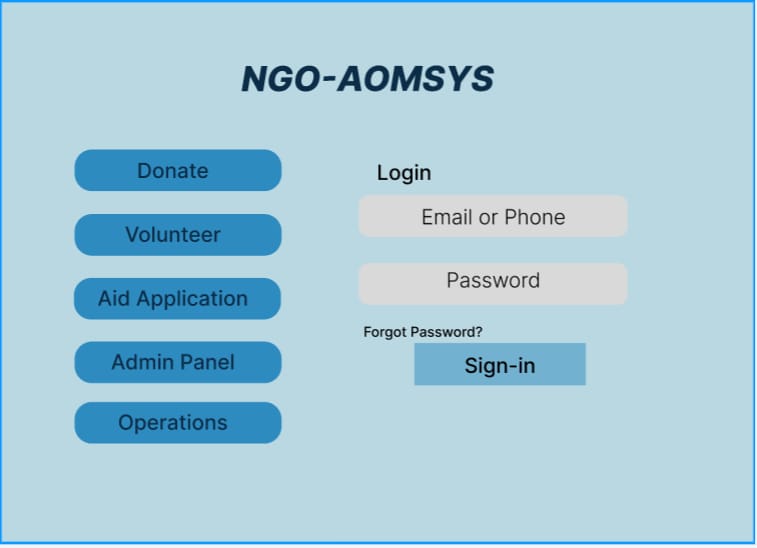


**5.4 State Chart Model**

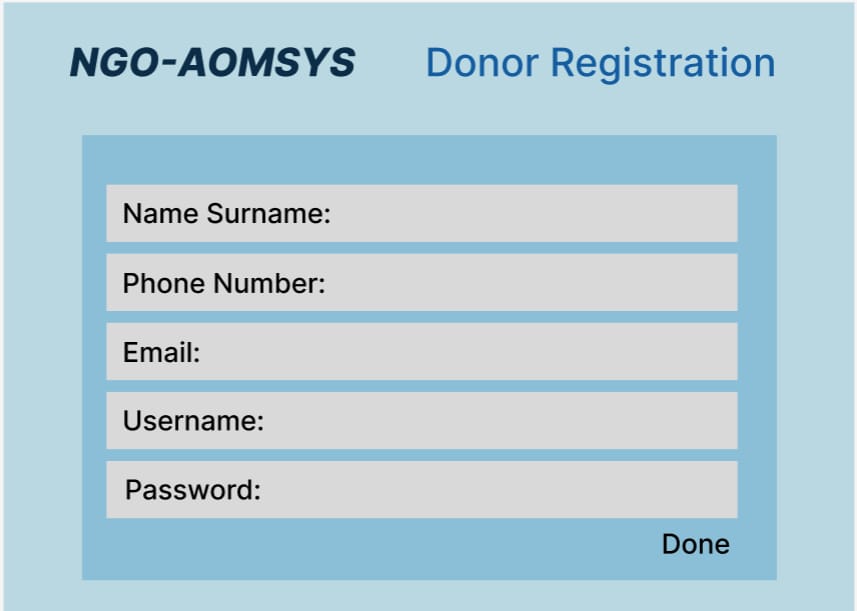


**5.5 USER INTERFACE MODEL**

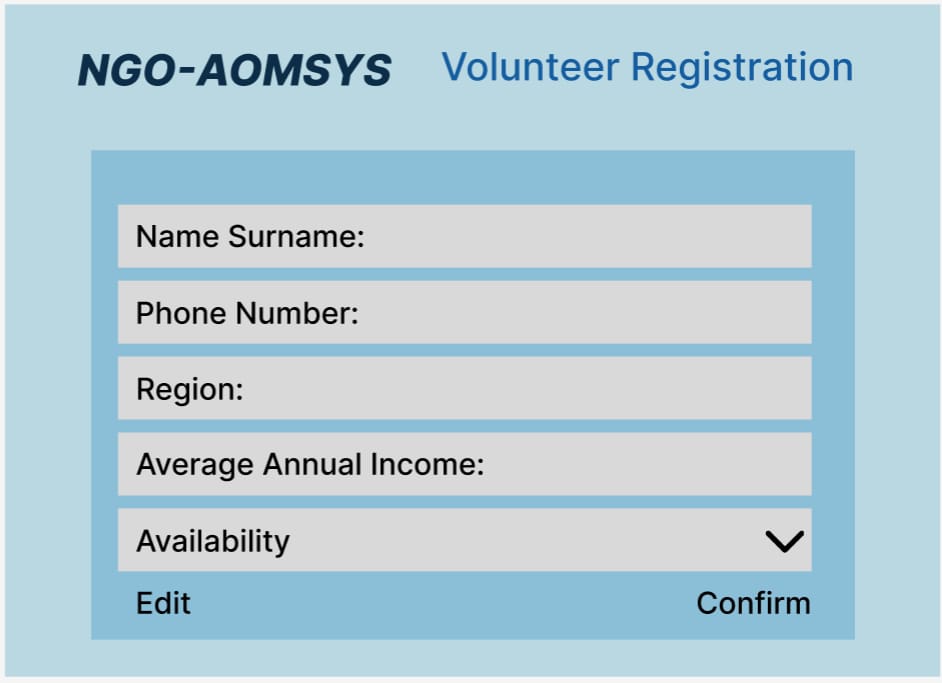
**5.5.1 LOGIN INTERFACE**



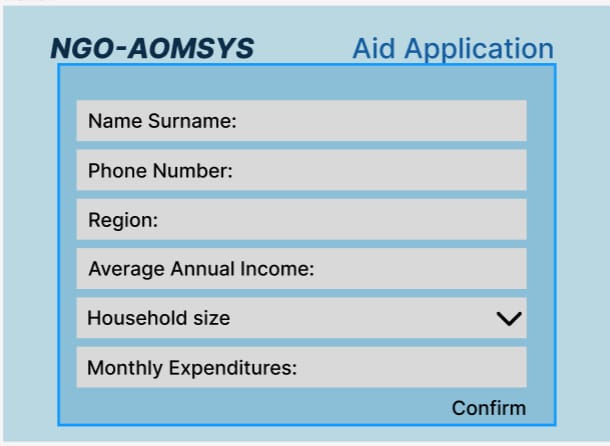
**5.5.2 DONOR REGISTRATION INTERFACE**



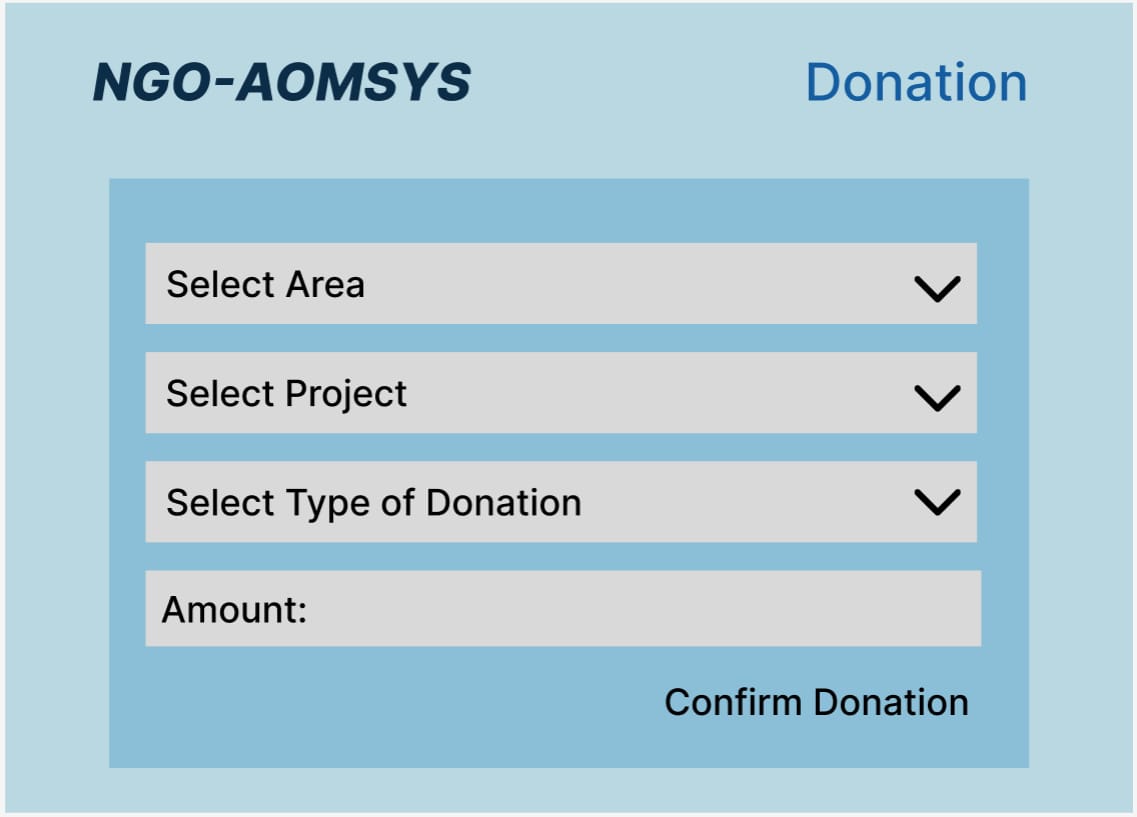
**5.5.3 VOLUNTEER REGISTRATION INTERFACE**



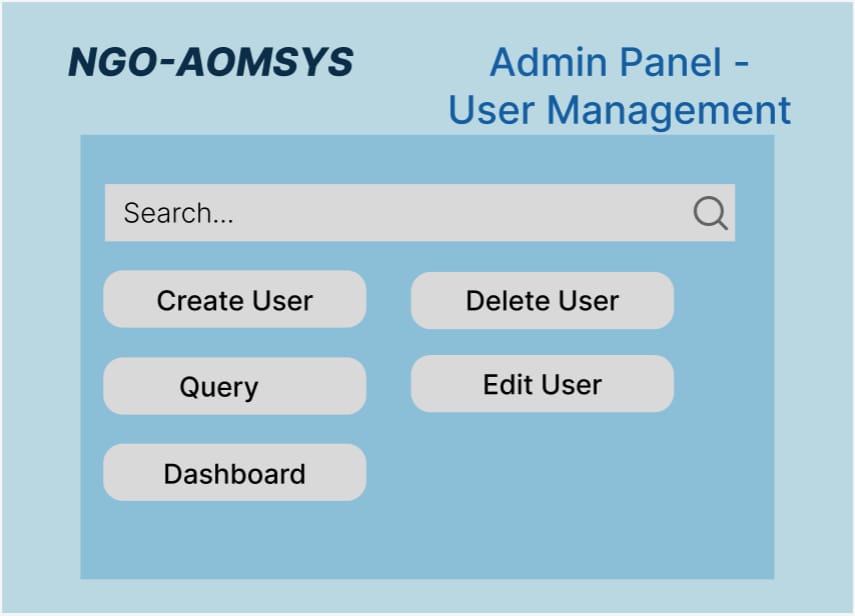
**5.5.4 AID APPLICATION INTERFACE**



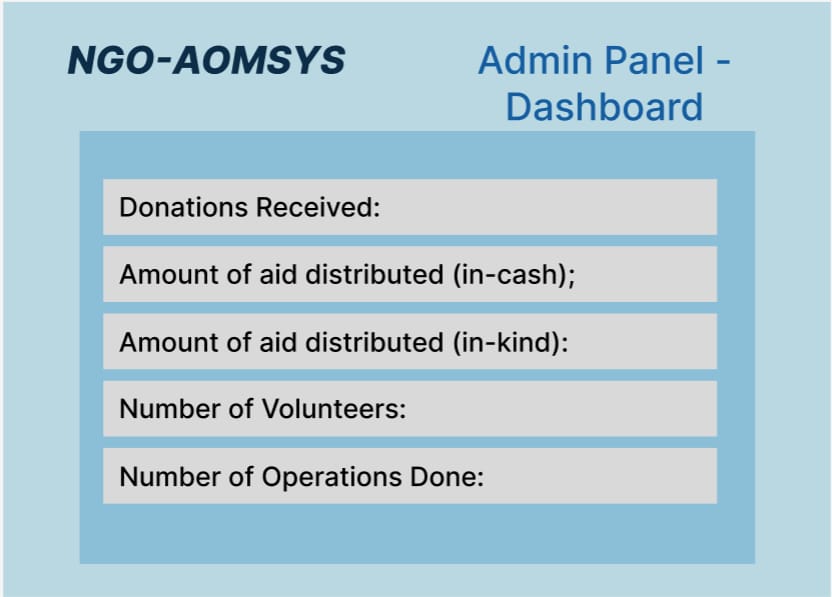
**5.5.5 DONATION INTERFACE**



**5.5.6 ADMIN PANEL-USER MANAGEMENT INTERFACE**



**5.5.7 ADMIN DASHBOARD INTERFACE**



**6. GLOSSARY**

* Donor: Benefactor
* Volunteer: Aid volunteer
* Indigent: Needy person
* Administrator: Person who has the ability to access and control the system
* GIS: Geographic Information System
* Shall: Must be implemented, and its implementation verified.
* Functional Requirements: Product features or functions that developers must implement to enable users to accomplish their tasks.
* Nonfunctional Requirements: Intended to specify 'system qualities,' various systems attributes that are not directly related to their functionality.