

TEAM NAME:	Kindred
DOMAIN	Non-profit and NGO Operations
PROBLEM STATEMENT	Volunteer and Resource Coordination

TEAM MEMBERS	ROLL NUMBERS
JAIDEEP	S20240010054
SAJJA VENKATA NAVEEN	S20240010208
SAI PRANEETH	S20240010188
MYTHRI	S20240010168
YASASWINI	S20240010047

TABLE OF CONTENTS

1. PREFACE	3
2. INTRODUCTION	3
2.1 Need for the System	3
2.2 Objective	3
3. GLOSSARY	3
4. USER REQUIREMENTS	4
5. SYSTEM REQUIREMENTS	4
5.1 Functional Requirements	4
5.2 Non-Functional Requirements	6
5.2.1 Product Requirements.....	7
5.2.2 Organizational Requirements.....	7
5.2.3 Security Requirements	8
6. SYSTEM DESIGN	9
6.1 Use Case Diagrams	9
6.2 Activity Diagrams	9
6.2.1 Volunteer Management	10
6.2.2 Register and apply for assistance	11
6.2.3 Volunteer communication and network	12
6.2.4 Search and apply for opportunities	13
6.2.5 Manage Users	14
6.3 Sequence Diagrams	15
7. APPENDICES	16
8. FUTURE SCOPE	16
9. CONCLUSION	17

1. Preface:

- This document defines the requirements for the Volunteer and Resource Coordination System.
- Intended for developers, testers, admins, and end users.
- It explains what the system should do and how it should behave.

2.Introduction:

2.1 Need for the System:

NGOs face problems like:

Poor volunteer management

Lack of donation transparency

Difficulty in tracking resources

Manual and inefficient processes

2.2 Objective:

To develop a centralized platform that connects Volunteers, Donors, Beneficiaries, NGO administrators

3. Glossary

- **Volunteer:** Person providing service
- **Donor:** Person donating money/resources
- **Beneficiary:** Person receiving help
- **Admin:** System controller
- **Resource:** Funds or materials donated

4. User Requirements

The system should allow users to:

- Access the platform through a web browser
- Register and login securely
- View information based on their role
- Receive notifications and updates
- Use the system easily without technical knowledge

5. System Requirements:

Functional requirements:

Functional requirements describe what a system is supposed to do, including the services it provides and the actions it performs in response to user input.

A. User Management:

- Volunteers can **register, login, and update profiles**
- NGOs/Admins can **verify volunteers**
- Different roles: **Volunteer, NGO/Admin, Government Authority**

B. Volunteer Management:

- Register volunteer skills (medical, logistics, rescue, food supply)
- Assign volunteers to tasks based on **location & skills**
- Track volunteer availability (active/inactive)

C. Resource Management:

- Add resources (food, medicine, shelters, vehicles)
- Track resource quantity and location
- Allocate resources to affected areas

D. Task & Activity Coordination:

- Create tasks (rescue, distribution, medical help)
- Assign tasks to volunteers
- Update task status (pending, in-progress, completed)

E. Location & Mapping:

- Identify disaster-affected locations
- Show nearby volunteers and resources
- Route planning for delivery

F. Communication & Alerts:

- Send notifications to volunteers
- Emergency alerts to all users
- Messaging between volunteers and coordinators

G. Reporting & Monitoring:

- Generate reports (volunteers deployed, resources used)
- View real-time dashboards
- Audit logs for actions

Non-Functional Requirements:

Non- Functional requirements specify quality attributes and constraints of a system, such as performance, security and reliability.

A. Product Requirements:

1. Efficiency Requirements:

- System should handle **large number of users simultaneously**
- Resource allocation should be fast

2. Performance Requirements:

- Page response time \leq **2 seconds**
- Notifications delivered in **real time**

3. Space Requirements:

- Efficient database storage for volunteer & resource data
- Optimize storage for images/documents

4. Usability Requirements:

- Simple UI for non-technical users
- Mobile-friendly interface
- Supports multiple languages

5. Dependability Requirements:

- System should be available **24/7**
- Data backup and recovery
- Minimal downtime during disasters

6. Security Requirements:

- Secure login and authentication
- Role-based access control
- Encrypt sensitive data (personal info, locations)

B. Organizational Requirements:

1. Operational Requirements:

- System should work on web and mobile
- Must run on cloud infrastructure

2. Environmental Requirements:

- Operates under low bandwidth conditions
- Supports use during power/network instability

3. Development Requirements:

- Developed using standard frameworks
- Easy to maintain and upgrade
- Modular design

C. External Requirements:

1. Regulatory Requirements:

- Follow government disaster-management guidelines
- Data protection compliance

2. Ethical Requirements:

- No misuse of volunteer personal data
- Fair task assignment without bias.

3. Legislative Requirements:

- Comply with IT laws and data privacy acts
- Maintain legal audit logs

4. Accounting Requirements:

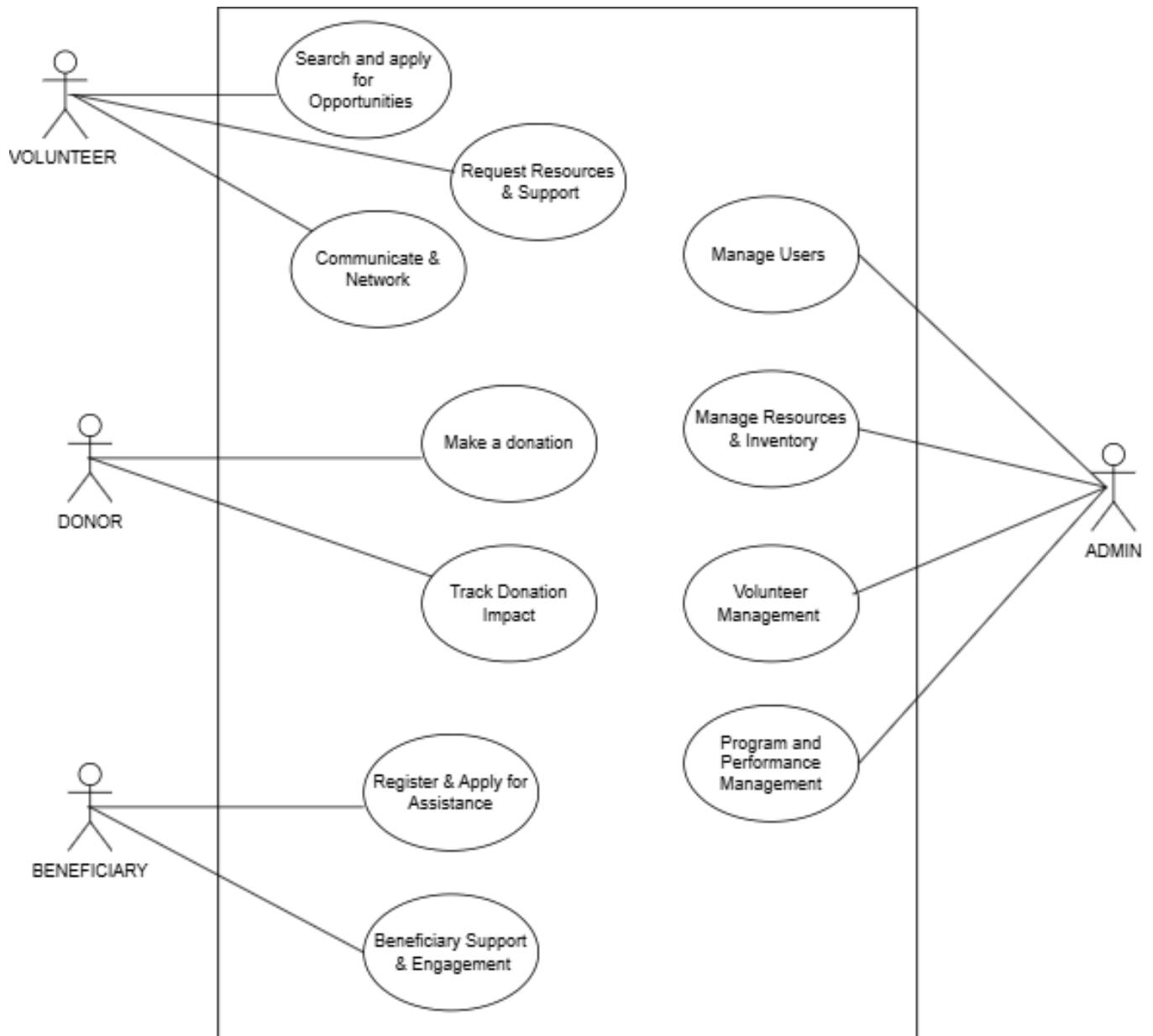
- Track donations and resource funding
- Generate financial usage reports

5. Safety / Security Requirements:

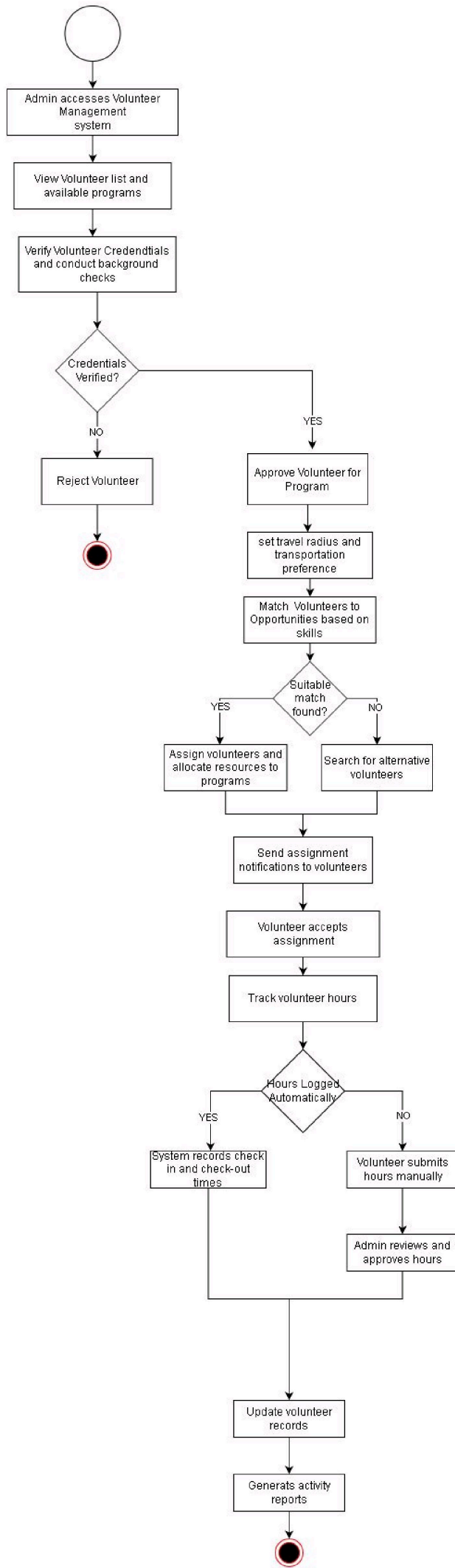
- Protect system from cyber attacks
- Prevent false emergency requests

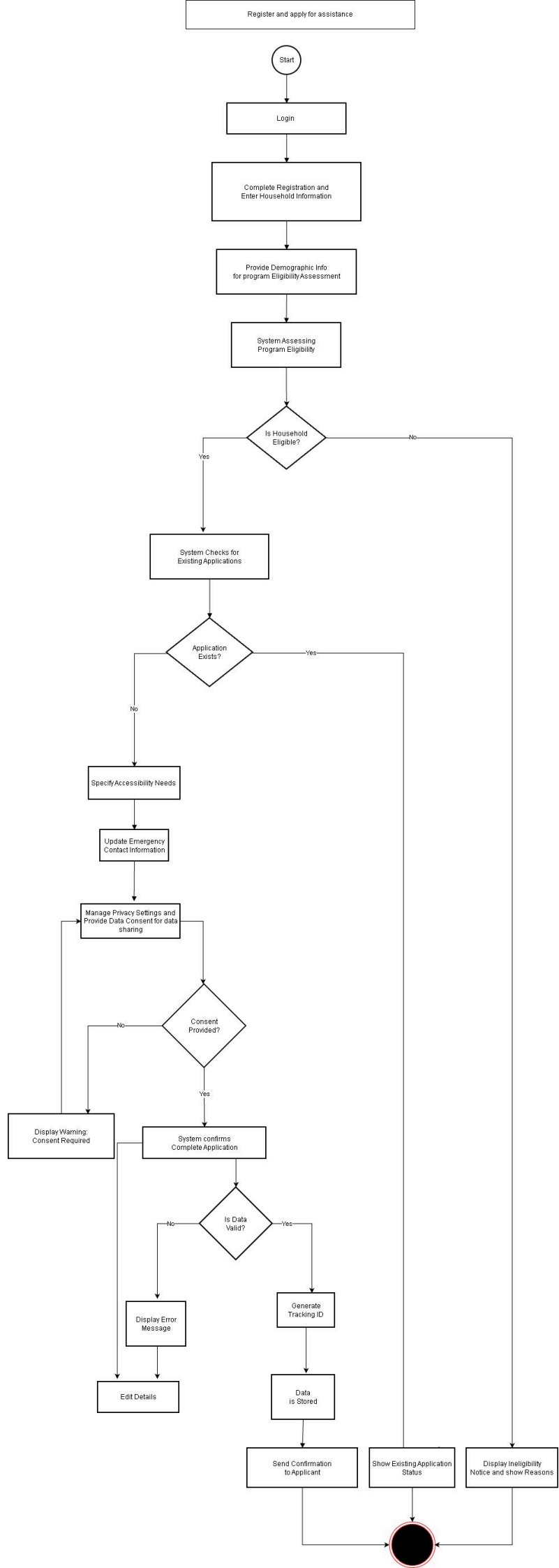
6.System Models

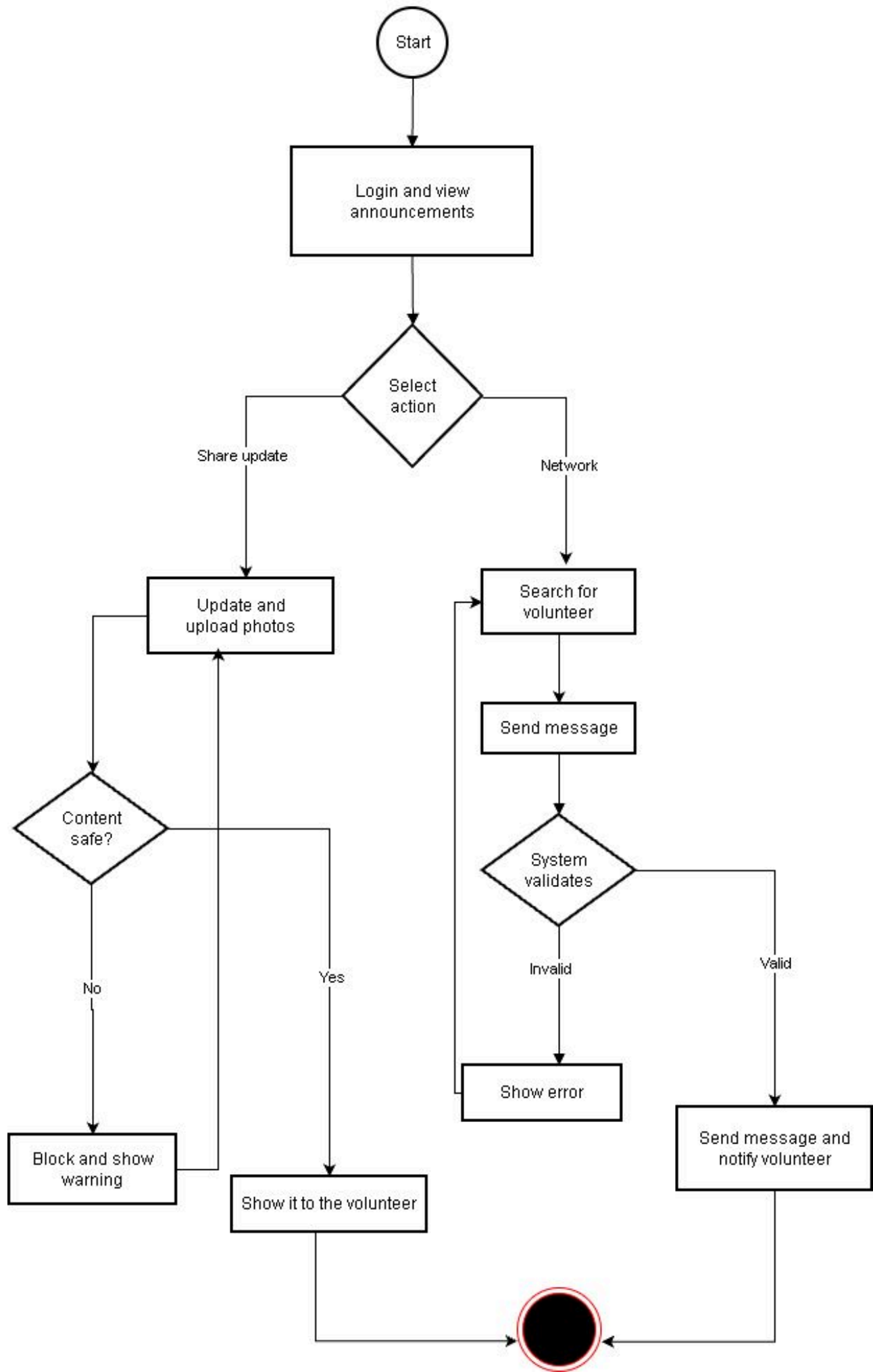
6.1 USE CASE:

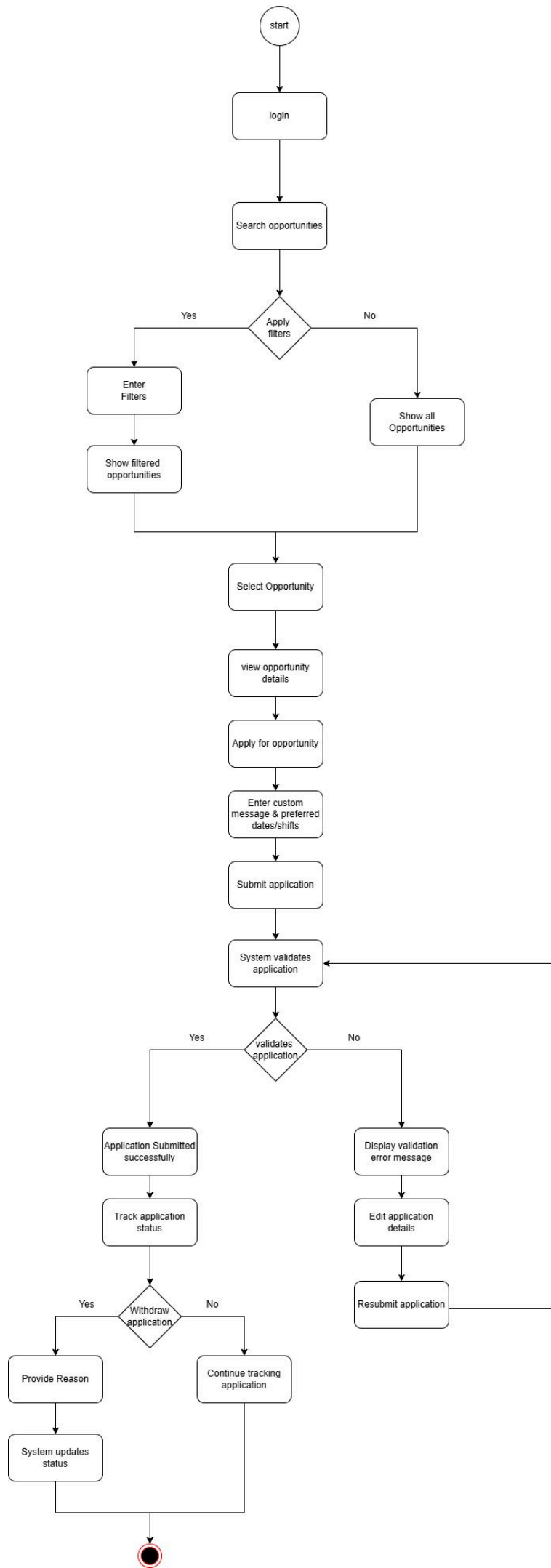


6.2 ACTIVITY DIAGRAMS:

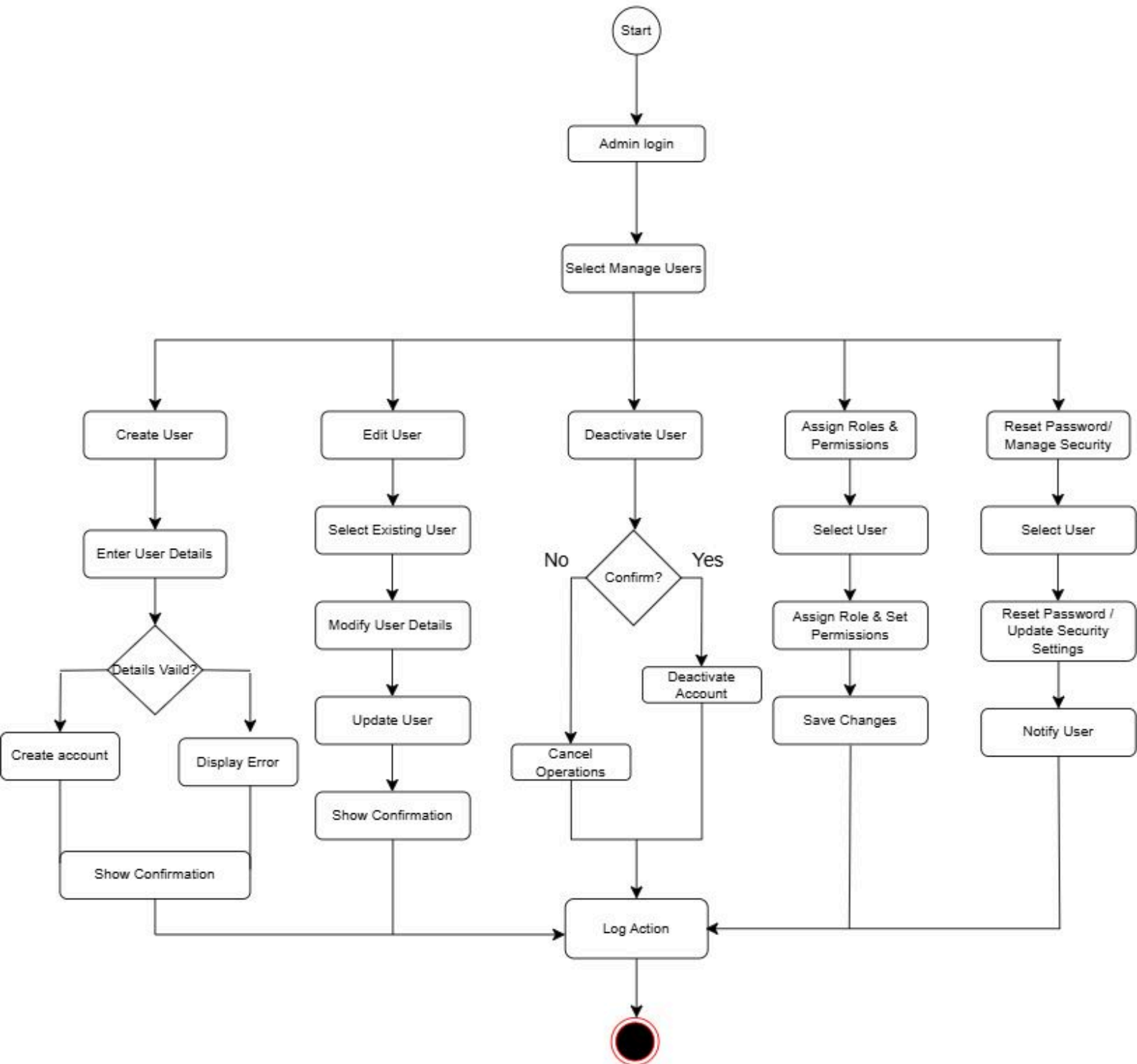




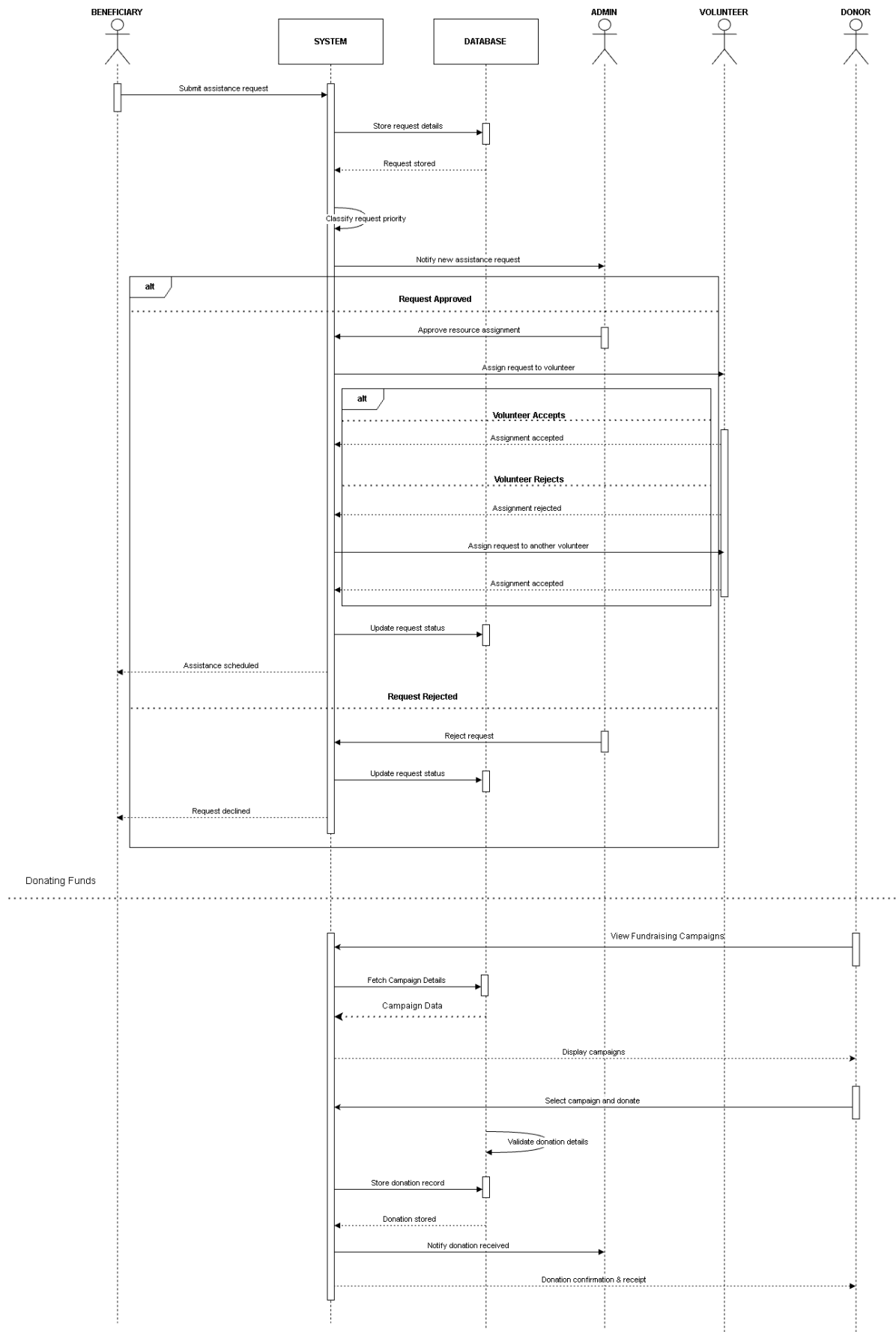




Activity diagram : Manage Users



6.3 SEQUENCE DIAGRAMS



7. APPENDICES

This section provides supplementary information related to the Volunteer and Resource Coordination System that supports the overall functionality but does not form part of the core system requirements.

Appendix A: Hardware Requirements

- A device such as a computer, laptop, tablet, or smartphone with internet connectivity.
- A reliable server with sufficient processing power, memory, and storage to handle user data.
- Secure backup hardware or cloud storage to prevent data loss.

Appendix B: Software Requirements

- A modern web browser (Chrome, Edge, Firefox, etc.) for accessing the system.
- Database management system to store volunteer, donor, and beneficiary information.
- Secure authentication system to protect user accounts and sensitive data.
- Basic operating system support (Windows, Linux, or macOS) for deployment and maintenance.

8. FUTURE SCOPE:

- Add automated notifications for volunteer assignments, resource updates, and beneficiary requests.
- Develop a mobile application to improve accessibility for volunteers, donors, and administrators.
- Implement intelligent volunteer matching based on skills,

availability, and location.

- Integrate secure online donation and real-time tracking features for better transparency.

9. CONCLUSION:

The Volunteer and Resource Coordination System helps NGOs manage volunteers, resources, donations, and beneficiary requests in an organized way. It reduces manual work, saves time, and improves communication between all users.

By using this system, NGOs can allocate resources properly and ensure that help reaches the right people at the right time.

Overall, the system provides a simple and effective solution for better NGO management and supports their mission to serve society.