Software Requirements Specification

for

NGO Management System

Version 1.0 approved

Prepared by
Dadi Sasank Kumar(22CS10020)
Nerella Trilochan(22CS10048)
Abhinav Meda(22CS10043)

INDIAN INSTITUTE OF TECHNOLOGY KHARAGPUR

23 MARCH, 2024

Table of Contents

Revision History

Name	Date	Reason For Changes	Version

1. Introduction

1.1 Purpose

This Software Requirements Specification (SRS) document outlines the specifications for the NGO Management System, version 1.0. It details the functional and non-functional requirements necessary for the development of the system. The scope of this document encompasses the core features required for managing donors, beneficiaries, volunteers, fundraising activities, financial transactions, and administrative tasks within the NGO dedicated to helping poor school children.

1.2 Document Conventions

This document follows standard conventions for software requirements specifications. Each requirement statement is accompanied by a priority level (e.g., high, medium, low) indicating its importance for system functionality. Additionally, any special terminology or abbreviations used throughout the document are defined in the Glossary section.

1.3 Intended Audience and Reading Suggestions

This document is intended for developers, project managers, testers, and stakeholders involved in the development and implementation of the NGO Management System. Developers should focus on understanding the technical specifications and requirements outlined in the Functional Requirements section. Project managers can use this document to plan project timelines and resource allocation. Testers should refer to the requirements for creating test cases and validating system functionality. Stakeholders can gain insights into the features and capabilities of the system to align with organizational goals and objectives. Readers are encouraged to start with the overview sections and progress through the document based on their specific roles and interests.

1.4 Project Scope

The NGO Management System is designed to streamline the operations of a non-governmental organization dedicated to assisting underprivileged school children. It aims to centralize donor management, beneficiary support, volunteer coordination, fundraising activities, financial management, and administrative tasks within a single platform. The system will improve efficiency, transparency, and accountability in delivering aid to the target beneficiaries. Future releases may include additional features such as integration with external platforms, enhanced reporting capabilities, and mobile application support.

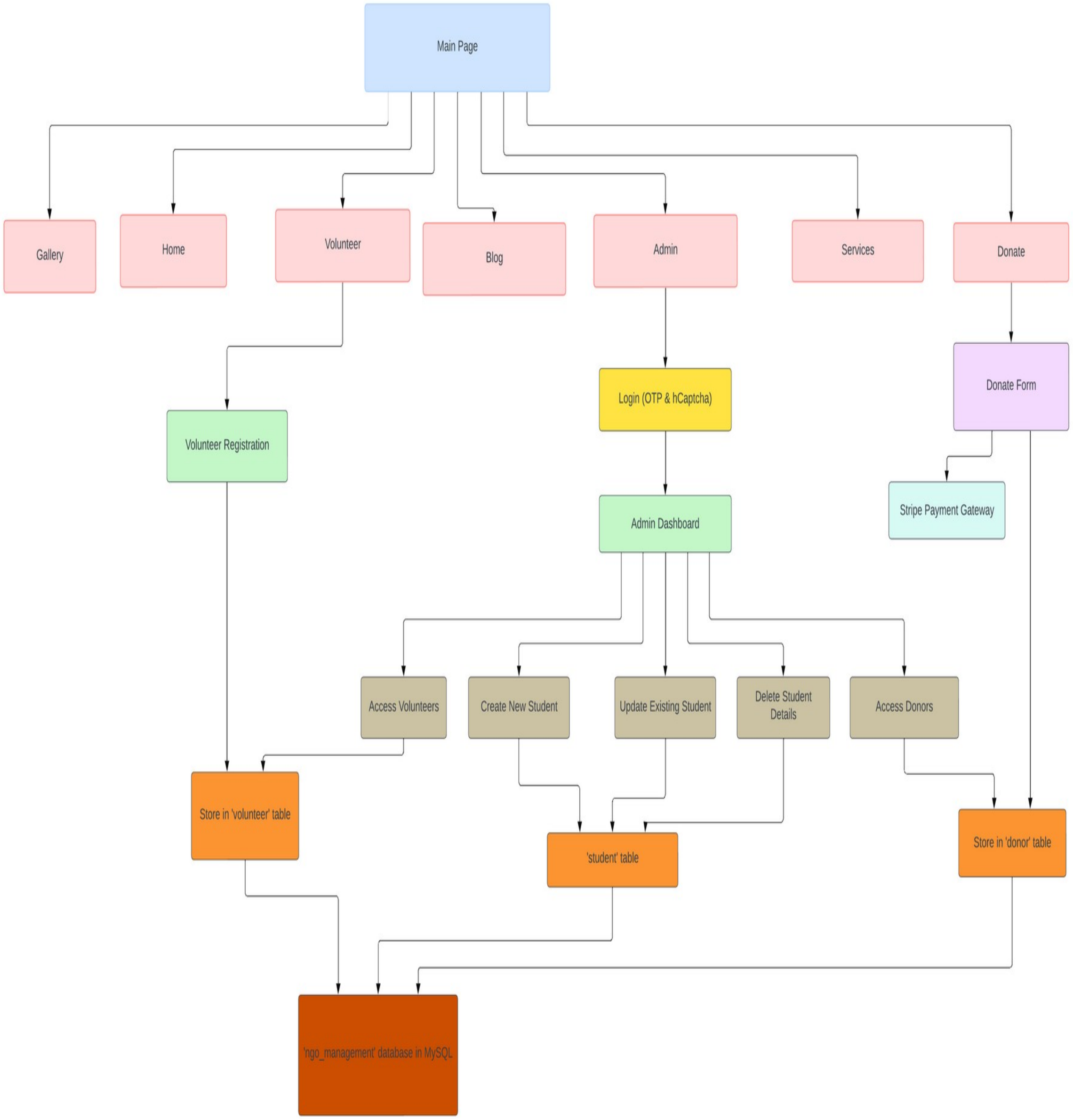
1.5 References

1) IEEE Std 1074-1997, IEEE Standard for Developing Software Life Cycle Processes
2) Links: https://rb.gy/wu2v9z
3) Class slides

2. Overall Description

2.1 Product Perspective

The NGO Management System (NMS) serves as a self-contained product providing comprehensive management capabilities for the NGO. It replaces manual or paper-based processes with an automated system. The NMS is developed as an extension of previous manual methods, aiming to streamline tasks for both helpers and managers within the NGO. The software enhances efficiency, accuracy, and convenience in managing student records, donations, and expenditure.



2.2 Product Features

The key features of the NMS include:

- *Registration and management of students data by admins*
- *New self registration of a student to NGO, subject to verification of the admin*
- *Acceptance and tracking of donations from pledged donors*
- *Managerial oversight including record checking, fund management, donor communication, and expense tracking.*
- *Automatic calculations and bookkeeping, reducing manual effort and minimizing errors.*
- *User-friendly interface requiring minimal experience for operation*

2.3 User Classes and Characteristics

2.3.1 Admin

- *Accesses the system with designated login credentials*
- *Has complete access to all records including students, donors, and volunteers.*
- *Responsible for donor communication, financial management, and overseeing volunteer activities.*
- *Authorized to add or remove or update student details from the system.*

2.3.2 Student

- *Can register themselves, upload documents(ID Proof)*
- *Has access to check their status of getting accepted by the NGO*

2.2.3 Donor

- *Can register as new donors*
- *Able to pledge donations and adjust frequency*
- *Can pay directly through Stripe payment gateway on donor side*

2.3.4 Volunteer

- *Can register themselves as volunteers for the NGO*
- *edit their details, remove their details from the system*

2.4 Operating Environment

The NGO Management System (NMS) is compatible with both Windows and Linux operating systems. It is designed to run on standard personal computers with an internet connection. The system is built using Python Flask for web development and MySQL for database management.

System Requirements:

- *Operating System: Windows or Linux*
- *Processor: Intel Core i3 or above*
- *RAM: 4GB minimum*
- *Hard Disk Space: 100MB for installation*
- *Internet Connection: Required for accessing the system and database connectivity*

Software Dependencies:

- *Python flask: Version 3.x or later*
- *MySQL: Version 5.x or later*
- *MySQL Connector for Python: Required for database connectivity*
- *Web Browser: Any modern web browser (e.g Google Chrome, Mozilla Firefox)*

Hardware Requirements:

- *Standard personal computer or laptop*
- *No special hardware components are necessary for installation or operation. The system utilizes existing hardware resources efficiently.*

***Note:** It's recommended to ensure that the system meets the specified requirements to ensure optimal performance and compatibility. Additionally, appropriate firewall and network configurations may be necessary to facilitate communication between the NMS application and the MySQL database server.*

2.5 Design and Implementation Constraints

The NGO Management System (NMS) is designed to operate on both Windows and Linux platforms, ensuring flexibility and compatibility with a wide range of systems. However, the system is currently available only in English, which may limit accessibility for users who prefer other languages.

Constraints:

- *Operating Platforms: The NMS is constrained to run on Windows and Linux environments. While these platforms cover a significant portion of users, it may not be accessible on other operating systems such as macOS.*
- *Language Limitation: The system is available only in English, potentially excluding users who require support in other languages. This constraint may impact the system's usability and adoption among non-English-speaking users.*

Considerations:

- *Localization: Future versions of the NMS may consider incorporating localization features to support multiple languages, thereby enhancing accessibility and usability for a broader user base.*
- *Platform Compatibility: Continuous development efforts should ensure ongoing compatibility with evolving versions of Windows and Linux, as well as potential expansion to other operating systems to accommodate diverse user preferences*

***Note:** While the current version of the NMS may have these constraints, addressing them in future updates can enhance the system's usability and reach.*

2.6 User Documentation

- *Online Help: Interactive help resources accessible within the NGO interface, providing contextual assistance and guidance to users as they interact with different features and functionalities.*
- *"Contact Us" Portal: Users will have access to a "contact us" portal on the NGO website, allowing them to submit inquiries, feedback, or support requests directly to the administrator or support team.*

Delivery Formats and Standards: *The user documentation will be provided in digital formats accessible through the NMS interface and online portals. It will adhere to standard documentation conventions, including clear language, structured organization, and visual aids for enhanced understanding.*

2.7 Assumptions and Dependencies

Assumptions:

- *User Familiarity: It is assumed that users have basic computer literacy and familiarity with standard web browsing and data entry operations.*
- *Email Support: The availability of an email support system is assumed for users to reach out for further assistance or clarification regarding the NMS functionalities.*

Dependencies:

- *Admin Access: The NMS relies on a hierarchical structure with administrators having comprehensive access to all information and functionalities within the system.*
- *Payment Verification Alerts: The system depends on the timely delivery of payment verification alerts to managers for ensuring transparency and accountability in donor transactions.*

3. System Features

In this section, the system features of the NGO Management System (NMS) are outlined, detailing the major services provided by the software.

3.1 Student data and profile management

3.1.1 Description and Priority

This feature allows the registration and management of poor students within the system.

Priority: High

3.1.2 Stimulus/Response Sequences

- *Stimulus: Admin accesses the student registration page.*
  Response: System displays the student registration form.
- *Stimulus: Admin submits student registration form.*
  Response: System validates the information and creates a student profile.
- *Stimulus: Admin submits update/delete requests*
  Response: Student data gets updates/deleted from the system

3.1.3 Functional Requirements

- *REQ-1: The system shall provide a registration form for entering student information, including name, age, grade level, school, and family background.*
- *REQ-2: Upon submission of the registration form, the system shall validate the entered information for completeness and accuracy.*
- *REQ-3: If any validation errors occur, the system shall display error messages prompting the helper to correct the invalid fields.*

- *REQ-4: Once validated, the system shall create a student profile with a unique identifier and store the student's information securely in the database.*
- *REQ-5: The system shall allow helpers to view and update student profiles, including adding new students or modifying existing information.*

3.2 Donor Management and accepting donations

3.2.1 Description and Priority

This feature allows donors to register their profiles within the system.
This feature also enables the recording of donations made by donors and maintains a history of donation transactions.

Priority: High

3.2.2 Stimulus/Response Sequences

- *Stimulus: Donor accesses the registration page.*
  Response: System displays the registration form.
- *Stimulus: Donor submits registration form.*
  Response: System validates the information and creates a donor profile
- *Stimulus: Donor accesses the donation page.*
  Response: System displays the donation form.
- *Stimulus: Donor submits donation form.*
  Response: System records the donation and updates the donor's donation history.

3.2.3 Functional Requirements

- *REQ-6: The system shall provide a donation form with fields for donation amount, frequency, and payment method.*
- *REQ-7: Upon submission of the donation form, the system shall record the donation transaction, associating it with the respective donor profile.*
- *REQ-8: The system shall maintain a donation history for each donor, including details such as donation amount, date and time.*
- *REQ-9: Donors shall be able to view their donation history through their profile dashboard.*

3.3 Communication tools

3.3.1 Description and Priority

This feature facilitates communication between the NGO and donors for engagement purposes.

Priority: Medium

3.3.2 Stimulus/Response Sequences

- *Stimulus: Donor accesses the communication portal ("Contact us portal").*
  Response: System displays form for communicating with the admin
- *Stimulus: Donor selects communication method.*
  Response: System initiates communication with the donor

3.3.3 Functional Requirements

- *REQ-10: The system shall provide communication portal for engaging with everyone using the website portal*
- *REQ-12: The system shall send automated email notifications to donors for acknowledging donations, providing updates, and expressing gratitude.*
- *REQ-13: Donors shall have the option to unsubscribe from email communications if desired.*
- *REQ-14: Admin gets the mail directly into their inbox which user had sent through the "contact us" portal*

3.4 Volunteer management

3.4.1 Description and Priority

This feature facilitates volunteers to register themselves on the website, so that admin can contact them when volunteers are needed for any activity NGO organizes/ will organize in future

Priority: Medium

3.4.2 Stimulus/Response Sequences

- *Stimulus: Volunteer accesses the form to register themselves for the NGO on the website*
  Response: System displays form for the volunteers, which has options like subscribe to newsletters/updates,etc
- *Stimulus: Volunteer submits registration form*
  Response: System validates the information and creates a volunteer profile.

- *Stimulus: Volunteer checks the box to subscribe for the newsletters/updates*
  Response: Database gets updates accordingly in a new column

3.4.3 Functional Requirements

- *REQ-15: The system shall provide a registration form with fields for volunteer information, including name, email, contact number, and skills/interests.*
- *REQ-16: Upon submission of the registration form, the system shall validate the entered information for completeness and accuracy.*
- *REQ-17: If any validation errors occur, the system shall display error messages prompting the volunteer to correct the invalid fields.*
- *REQ-18: Once validated, the system shall create a volunteer profile with a unique identifier and store the volunteer's information securely in the database.*
- *REQ-19: The system shall allow volunteers to log in to their accounts using their registered email and password for profile management.*

3.5 Admin login with hCaptcha and OTP

3.5.1 Description

This feature provides a secure login system for administrators by integrating both hCaptcha and OTP authentication methods to prevent automated bot attacks and enhance security.

3.5.2 Stimulus/Response Sequences

- *Stimulus: Admin accesses the login page*
  Response: System displays the login form with hCaptcha and OTP authentication.
- *Stimulus: Admin selects the preferred authentication method (hCaptcha and OTP).*
  Response: System prompts the admin to complete the chosen authentication method.
- *Stimulus: Admin completes the authentication processes*
  Response: System verifies the authentication response and if correct, will allow to login else will ask to login again
- *Stimulus: Admin if entered incorrect login credentials consequently for 3 times in a span of 1 min*
  Response: Login form disables for 30 secs

3.5.3 Functional Requirements

- *REQ-20: The system shall integrate hCaptcha with the admin login form to verify human interaction and prevent automated bot attacks.*
- *REQ-21: The system shall provide administrators for hCaptcha and OTP authentication methods.*
- *REQ-22: The system shall validate the hCaptcha response from the login form and then checks for username and password*
- *REQ-23: The system shall validate the OTP entered by the admin during the login process.*
- *REQ-24: If the hCaptcha or OTP validation fails or expires, the system shall prompt the admin to retry the verification process*
- *Upon successful hCaptcha or OTP validation, the system shall authenticate the admin credentials and grant access to the admin dashboard for system management tasks.*

***Note:** Adding OTP authentication enhances the security of the admin login process by requiring an additional verification step. This feature helps prevent unauthorized access to sensitive administrative functions within the system.*

***Note:** These are preliminary features for student management within the NGO Management System. Additional features and requirements may be identified during further analysis and stakeholder consultation*

4. External Interface Requirements

4.1 User Interfaces

The user interface of the NGO Management System (NMS) is designed to be intuitive, user-friendly, and accessible. It includes the following logical characteristics:

- ***Admin Dashboard:** Provides a centralized interface for administrators to manage the system, including features such as donor management, student management, volunteer management, and reporting tools.*
- ***Donor Registration:** Allows donors to register for donations*
- ***Student Registration and Support Services:** Enables admins to register and manage poor students and provide assistance as needed.*
- ***Volunteer Registration and Profile Management:** Allows individuals to register as volunteers, indicate their skills and availability, and manage their profiles.*

- ***Login Forms:*** *Secure login forms for administrators, incorporating hCaptcha and OTP authentication methods for enhanced security*
- ***Navigation Menu:*** *Consistent navigation menu for easy access to various features and functionalities.*
- ***Error Messages:*** *Standard error message display for notifying users of any validation errors or system issues.*
- ***Help Functionality:*** *Provision of help functionality to assist users in understanding the system features and processes.*

4.2 Hardware Interfaces

The NGO Management System (NMS) operates on standard personal computers with an internet connection. It does not have any specific hardware dependencies. The system interacts with common hardware components such as:

- *Laptops,*
- *Desktops,*
- *Mobile(Future Releases)*

4.3 Software Interfaces

The NMS interacts with various software components, including:

- *Database: Utilizes MySQL for data storage and retrieval.*
- *Operating Systems: Compatible with both Windows and Linux operating systems.*
- *Programming Languages and Frameworks: Developed using Python Flask framework for backend development and HTML/CSS/JavaScript for frontend development.*
- *Third-party services: Utilizes libraries for implementing features such as hCaptcha(hCaptcha,Open source and free) integration, OTP generation, and email sending functionality(FormCarry- Email communications), stripe payment portal(for accepting payments(only test mode used here for developers and project use cases)*
- *Integrated Development Environment (IDE): Developers may use IDEs such as PyCharm, Visual Studio Code, or Sublime Text for coding and debugging.*

4.4 Communications Interfaces

The NGO Management System (NMS) requires communication interfaces for various functions, including:

- ***Email:*** *Used for sending notifications, alerts, and communication with users.*
- ***Web Browser:*** *Users interact with the system through web browsers such as Chrome, Firefox, Safari, or Edge.*
- ***HTTP/HTTPS:*** *Utilized for communication between the client-side and server-side components of the system.*
- ***Encryption:*** *Utilizes encryption protocols such as SSL/TLS to ensure secure communication over the internet.*

- ***Data Transfer Rates:*** *The system aims for efficient data transfer rates to ensure optimal performance and responsiveness.*

These interfaces are crucial for the proper functioning and interaction of the NGO Management System with users, hardware components, software components, and communication channels.

5. Other Nonfunctional Requirements

5.1 Performance Requirements

- *The system should respond to user interactions within 2 seconds under normal load conditions to ensure a smooth user experience.*
- *The system should be capable of handling concurrent user sessions without significant degradation in performance. It should support at least 100 concurrent users without noticeable slowdowns.*
- *Database queries should execute within 1 second for standard operations such as fetching student or donor information.*
- *System backups should not impact normal system operations, and backup processes should complete within 1 hour.*
- *The system should be scalable to accommodate future growth in data volume and user traffic.*

5.2 Safety Requirements

- *Measures should be implemented to prevent unauthorized access to sensitive information, ensuring data confidentiality and integrity.*
- *Adequate data backup mechanisms should be in place to prevent data loss in case of system failures or disasters.*

5.3 Security Requirements

- *User authentication should be performed securely using industry-standard encryption protocols.*
- *Role-based access control (RBAC) should be enforced to restrict access to sensitive functionalities based on user roles.*
- *All communications between the client and server should be encrypted using secure protocols such as HTTPS.*
- *Regular security audits and vulnerability assessments should be conducted*
- *The system should maintain audit logs to track user activities.*

5.4 Software Quality Attributes

- *Usability: The system should have an intuitive user interface.*
- *Reliability: The system should have minimal downtime and consistent performance.'*
- *Maintainability: The system should be designed with modular components and well-documented code.*
- *Interoperability: The system should be compatible with common web browsers and operating systems.*

5.5 Other Requirements

- *The system should support multi-language functionality.*
- *Compliance with international accessibility standards, such as WCAG, should be ensured.*
- *Legal requirements related to data protection, copyright, and intellectual property rights should be adhered to.*
- *The system should provide options for data export and reporting.*

Appendix A: Glossary

- ***Admin:*** *Refers to the administrative user who has full access to the system's management features.*
- ***NMS:*** *Acronym for NGO Management System, the software developed for managing operations within the NGO.*
- ***OTP:*** *Acronym for One-Time Password, a unique password generated for a single login session.*
- ***RBAC:*** *Acronym for Role-Based Access Control, a method of restricting system access based on the roles of individual users.*
- ***hCaptcha:*** *A type of CAPTCHA system designed to distinguish between humans and automated bots.*
- ***HTTPS:*** *Hypertext Transfer Protocol Secure, a secure version of HTTP used for encrypted communication over a computer network.*

Appendix B: Analysis Models

Appendix C: Issues List

TBD: Clarification needed on the user authentication process for admin login.
Pending Decision: Determination required on the frequency of system backups.
Information Needed: Additional details needed on the specific data export formats required by stakeholders.
Conflicts Awaiting Resolution: Conflict between stakeholders regarding the prioritization of certain features.