



SOFTWARE REQUIREMENTS SPECIFICATION

NGO Management System

Team: DJR

Dheeraj Parmar	20190802113
Jill Italiya	20190802028
Rohit Ramamorthy	20190802071

Table of Contents

1. Introduction	3
1.1 Purpose	
1.2 Project Scope	
1.3 References	
1.4 Overview	
 2. Overall Description	 3-6
2.1 Product Perspective	
2.2 Product Features	
2.3 User Classes and Characteristics	
2.4 Operating Environment	
2.5 Design and Implementation Constraints	
2.6 Assumptions and Dependencies	
2.7 User Documentation	
 3. Functional Requirements	 6-8
3.1 User Case Diagram	
3.2 Specific Requirements	
3.2.1 User Class – Admin	
3.2.2 User Class - Donor	
 4. External Interface Requirements	 8
4.1 User Interface	
4.2 Software Interface	

1 Introduction

1.1 Purpose

The purpose of this document is to give a detailed and complete description of the requirements for the NGO Management Software. It will illustrate the purpose and complete declaration for the development of the system. It will explain system constraints, interface and interactions with other external applications. It includes the functions and features of the software. Moreover, it gives an overview of the design implementation of the software. The expected audience of this document is the NGO management particularly the person responsible for running the NGO.

1.2 Project Scope

This project aims to computerize the administration of an NGO. There is a web portal for the registration of new donors and a blog of the NGO. It would help increase productivity of the NGO by reducing the time required in the operations of the NGO. It will also increase the accuracy of the work done.

1.3 References

- IEEE Software Engineering Standards Committee, IEEE Std 830-1998, IEEE Recommended Practice for Software Requirements Specifications, October 20, 1998.
- Class's Slides

1.4 Overview

This software system is a web NGO management system. It has different views for the admin as well as the donors. People can view the donations made by themselves and the admin can see all the donors' donations. It will be a platform where the needy can get their help during any emergency.

2 Overall Description

It has some attributes which includes donors, admin and the pledged donors. Those can contribute there on the portal.

The pledged donors can donate frequently by first registering. Then they can donate any amount of money or they can donate books, dress etc.

The admin can check the records, check the pledged funds and contact pledged donors.

2.1 Product Perspective

The origin of this problem is the need of automating and computerizing the operations of an NGO. All the data including the information of donors, admin is stored in the database. The user interface helps the management to extract information from the database. The software takes care of the communications between the databases and the user interface. The software also automates various operations of the NGO. The web portal allows new donors to register and view the activities of the NGO. Since this is a data-centric product it will need somewhere to store the data, a database will thus be used.

2.2 Product Features

The software will allow the admin to do the following tasks:

- Get records of those who have sought the NGOs help.
- Display records of donors who have pledged to help the NGO.
- Admin can block or unblock the donor/user from the web portal itself.
- Maintain the data of various donations to the NGO.

The web portal will serve the following purposes:

1. Contain information related to the activities of the NGO.
2. Have an option for users to register as donors.
3. Have an option for common users to donate money/items to the NGO.
4. Managing the donors all at once.
5. Having a function to fix the frequency for frequent donation.

2.3 User Classes and Characteristics

Types of user who interacts with the software:

- Admin: The person responsible for running the NGO. He can login and access all the functionalities available to the Admin.
- Donors: The users who use the internet portal are Donors and other users who want to learn about the NGO.

2.4 Operating Environment

Hardware: Central Server and Personal Workstations.

Operating System: Windows XP and above.

Internet Connection: Any suitable connection.

Performance: The software should function correctly always.

- The software has been developed in PHP, Angular and Python. The database Management task is done with the help of SQL.

2.5 Design and implementation constraints

- The information of all users and statistics related to the NGO must be stored in a database that is accessible by the Admin.
- The data is stored in a single database so it's prone to loss due to digital and physical calamities.
- The web portal requires an internet connection so Donor registration and donations can't be done offline.
- The software will be constrained by the capacity of the database. MS SQL Server will be used as SQL engine and database.
- The software is available only in English.
- PHP and Angular should be the languages preferred in case of updating the software.

2.6 Assumptions and Dependencies

- Each NGO member who is a part of the NGO management is expected to have an account of his own. We were trying to make their accounts as volunteers.
- The software assumes that an estimate for the amount of money needed for books, dress etc for each class is available to the software.
- The software assumes that a list of donors who have pledged support is available to the software.

2.7 User Documentation:

This is a user-friendly software with an easy to understand GUI. It is highly unlikely that anyone will have a problem using it. It only involves basic clicking and typing operations.

3 Functional Requirements

3.1 Use Case Diagram / Flow Chart



3.2 Specific Requirements

3.2.1 User Class -Admin

- **Register:**
→ **Process:**
Admin id can be automatically generated through the database. And that admin code will be the secret code which is the single unique admin id that cannot be generated by the portal.
- **Get estimate of funds to continue its operation:**
→ **Input:** The Admin can get an estimate of the amount of money needed for books, dress, fees etc for different classes which would be needed by the NGO to continue its

operation. This should be done manually by the admin. No functionality is added yet.

→ **Output:** The HOD gets the total amount of funds the NGO would require to continue its operation in the coming year.

→ **Process:** The NGO has an estimate of the amount of money needed for the books, dress, etc. This estimated data stored in the database is accessed to find out total money that would be needed for giving stuffs and the total cost of funding all needy ones are required is determined. Money required for other things is further added and the total funds required for continuing operations is thus obtained.

- **Check available donors:**

→ **Input:** The Admin can get the list of donors who have pledged to contribute and haven't contributed money in the last time along with the pledged amount.

→ **Output:** The Admin gets a list of donors who can contribute and the amount they are ready to contribute. And then he has to remind them of the donation.

→ **Process:** The Donor list is traversed and those donors who haven't contributed are added to the list. This list is finally returned.

- **Enter Donations received:**

→ **Input:** The Admin gets an option of updating the donations database, of adding the donations the NGO has received.

→ **Output:** Database updated: All new items received by the NGO as donation is added to the database.

→ **Process:** After the donation has been made the donation database is updated of all the donations made.

3.2.2 User Class – Donor

- **Register:**

→ **Input:** A user can register as a donor through the web portal.

→ **Output:** The user gets his Donor id as his email id.

→ **Process:** The email id which was given during registration will be assigned as his Donor id.

Update donation Information:

→ **Input:** Any donor having an account on the web portal can update his information and choose if they wish to donate annually or semi-annually.

→ **Output:** Donor information is updated.

→ **Process:** The NGO has the donor information stored in a database. When a donor updates his information on the web portal the changes are made in the database and the changes saved.

- **Donate items:**
 - **Input:** The donor selects items from a list, the item he wants to donate.
 - **Output:** The information about the donated item is stored in the database.
 - **Process:** The user gets a list of items each corresponding to some stuff . The user selects the particular item he is donating and the number of that item is entered in the corresponding field. The information is added to the database.
- **Enter Donations received:**
 - **Input:** The admin will update the donations database, by adding the donations the NGO has received. He needs to do it manually.
 - **Output:** Database updated: All new items received by the NGO as donation is added to the database.
 - **Process:** After the donation has been made the donation database is updated of all the donations made.

4 External Interface Requirements

This section provides a detailed description of all inputs into and outputs from the system. It gives a description of the software interfaces and provides basic prototypes of the user interface.

4.1 User Interfaces

Anyone who opens the portal can see a welcome page. There are options to log in.

4.2 Software Interfaces

The application communicates with the database in order to get the information. The communication between the database and the web portal consists of operations concerning both reading and modifying the data.

Thank You !!