# 1. Introduction

A Church donation system, Church Financial Record Keeping, is being developed. The application development is done with special attention given to user satisfaction. Design of the application's interface is given high priority. The interface is the user interaction screen with the application and its functionalities. New interaction features are provided to support the needs of the application user.

**2. User Interface Design**

The main purpose of the user interface design is to make the user interaction as simple as possible and efficient, in terms of goal accomplishment. The usability influences how the user perform the interaction and helps to improve the aesthetic appeal of the design. The system is created following the design process to balance the technical functionalities and the visual elements.

**3. Description of the User Interface**

The interface planned and to be developed is a Web interface system for the project. The web usage has made people very demanding for web interfaces. The interface gives a friendly appeal to the user. There are several interaction screens for interaction with the system. The different screen interface developed try to cover all the functionalities of the user needs.

In the starting, there is a login page interface. The login page displays the company logo and different input fields, buttons to validate and verify the correct users and enable the user for usage of the system. After correct validation and verification the screen navigates to the main page of the application from this page interface then the user can perform her task operations. The main page includes links to different pages where each page has different functionalities implemented as per the user's needs. There are different tabs where each tab has a functionality to be performed and is selected for a particular operation.

The Member tab is selected to add a member to the church financial recordkeeping system. It can also be used to edit or delete any existing member from the database of the system. The Fund is selected to perform operation on the type of funds of the church, like to add new type of fund to the system, to delete the existing type of fund which is of no use and to modify the existing fund as per the needs.

The Donation is chosen to add a record of the financial donation given to the church. The system user can add different records by choosing the member who has donated and the type of fund. The system is designed to add record with different type of fund by dividing the donation record among different funds. User flexibility is kept in mind while designing the system so that the user can add number of records with ease.

**3.1 UML Class**

The application functionalities are identified that constitute in forming the classes of the application.

The figure 1 is the class diagram. It contains the major classes of the application.

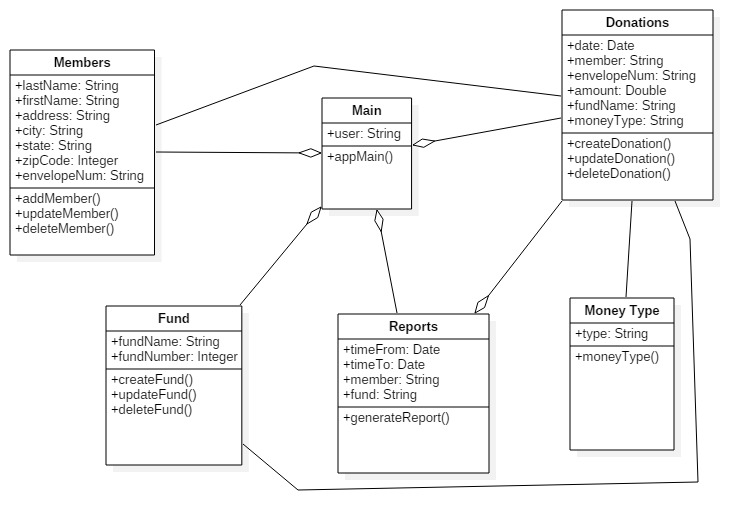


Figure 1: Class diagram

**3.2 Screen Images**

The system consists of objects and actions associated with each of them. The actions perform the function which perform one of the needs of the user.

The screen interface designs are made by taking different factors into consideration. The screen represents the user's point of view. The pages contains, text boxes, labels, which can be used to describe the text boxes or to provide additional information on the screens, check boxes, radio buttons, and user inputs and outputs display. The content in the fields is evaluated and verified as necessary. Message boxes would be provided to give the user information of what the application is doing and also to give additional information about errors raised or about input expected from the user. Tool tips would be provided for every button on the screen. This would give the user an idea of what the button or a particular component in the screen does. This would help make the user interface more user friendly. Buttons on the screen would be activated only as the need arises and buttons that are no longer required would be disabled.

The figure show the first screen of the application, the Login Page.

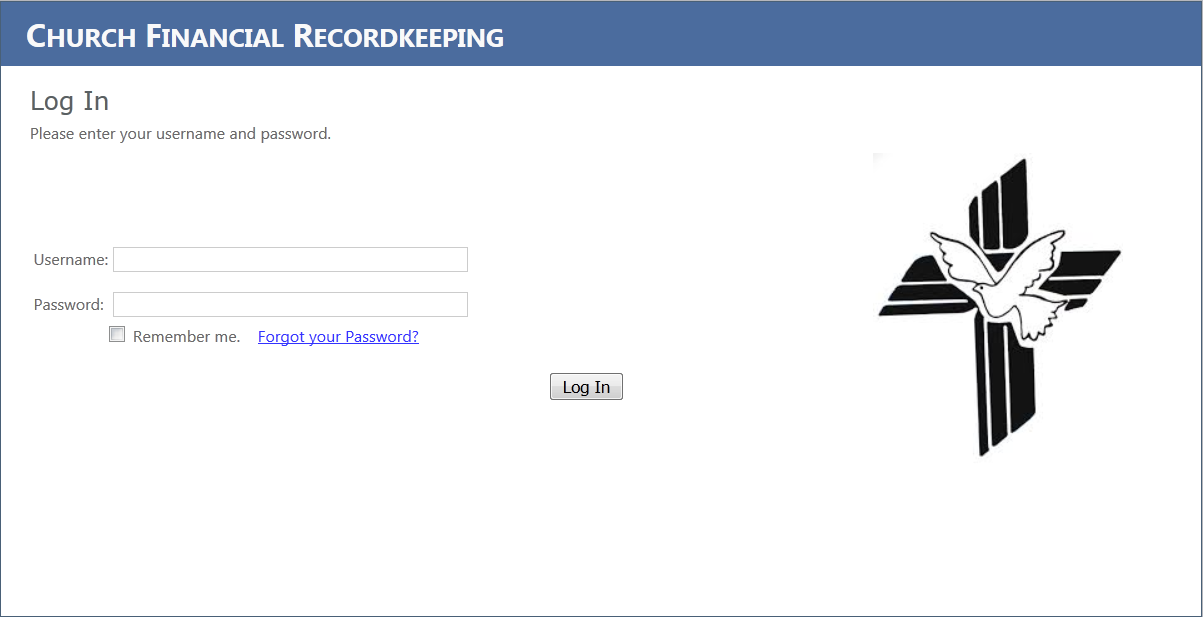


Figure 2: Login Page

The user enters Username and Password in the text boxes provided and use the button for login. The text boxes content will then be evaluated and verified for the correct user for authorization. The cursor will be placed in that field after the user is prompted about the unrecognized or unexpected input. This will help the user locate the field in which he is expected to enter values.

Logout button:

All the rest of the screens have logout button at the top right corner of every page after user logs in to the application.

The figure 3 shows the member screen for adding the member data.

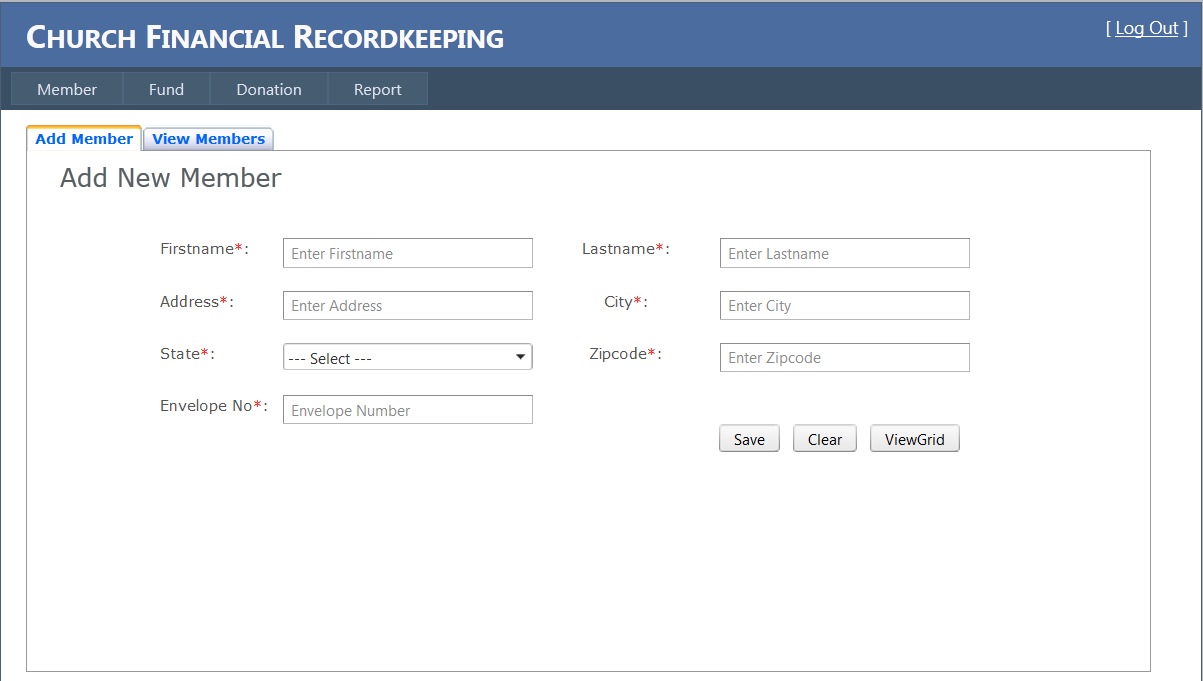


Figure 3. Add Member

The figure 4 shows the member screen for updating and modifying the member data.

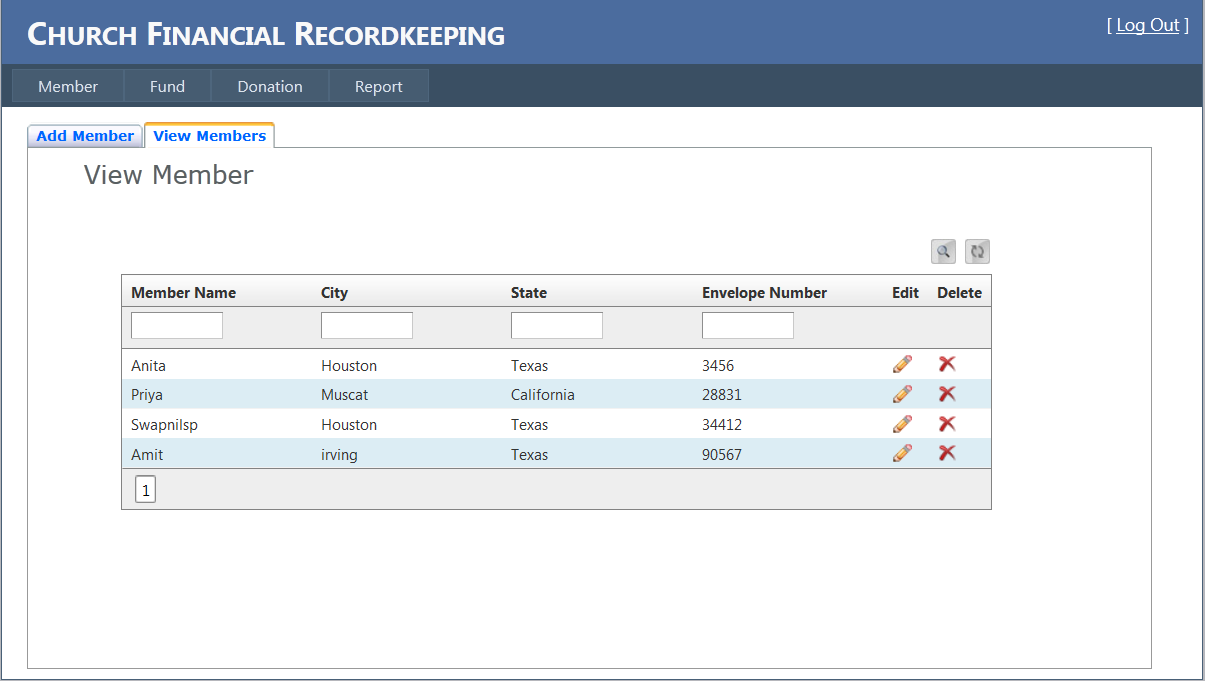


Figure 4. View Member

The member page is used by the user to store the member’s record in the system. The member information can be modified and deleted as needed.

Member screen page objects:

* Text boxes are used for storing member’s information.
* Buttons: save and clear are two buttons. The save button, when clicked, execute function to store the member information present in the text boxes in the database.
* Clear button, when used, clean the text from the text boxes.
* Grid display is used to display member’s information. It contains button options to delete the row and edit the row.
* Search button is used to search the record of a particular member.
* Refresh button is used to display the recent members data.

The below figure is the fund screen. The type of fund the members can donate are maintained using this screen.

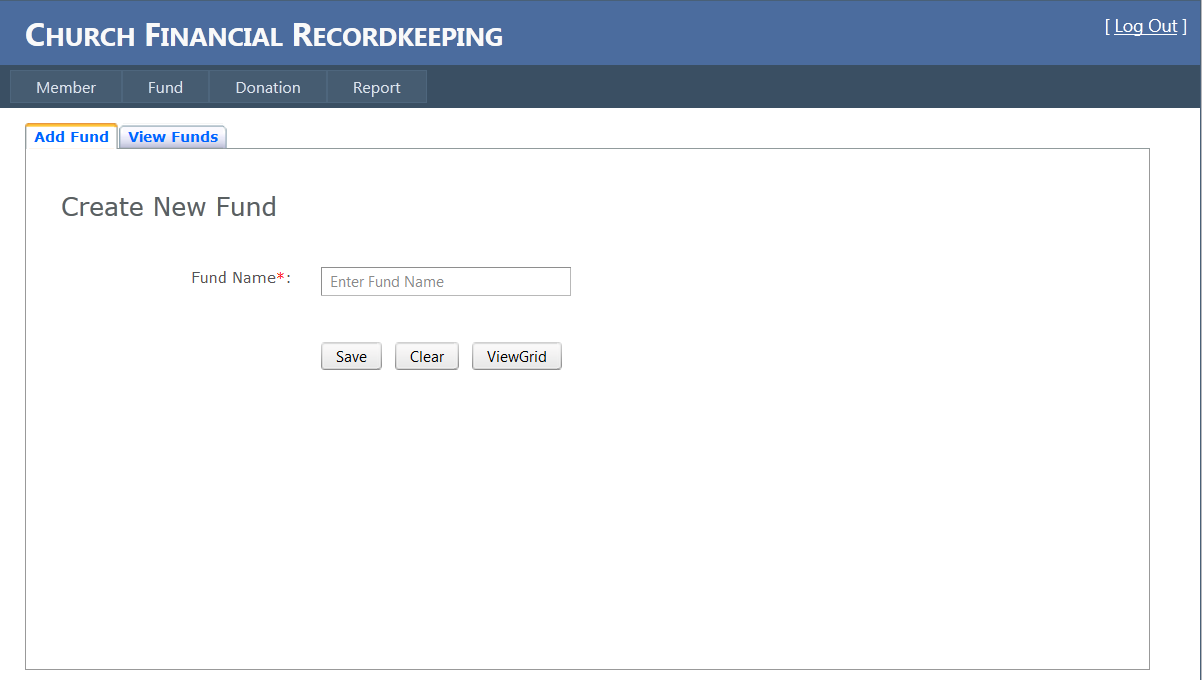


Figure 5: Add Fund

The below figure is the fund screen. The type of fund the members can donate are maintained to edit and delete using this screen.

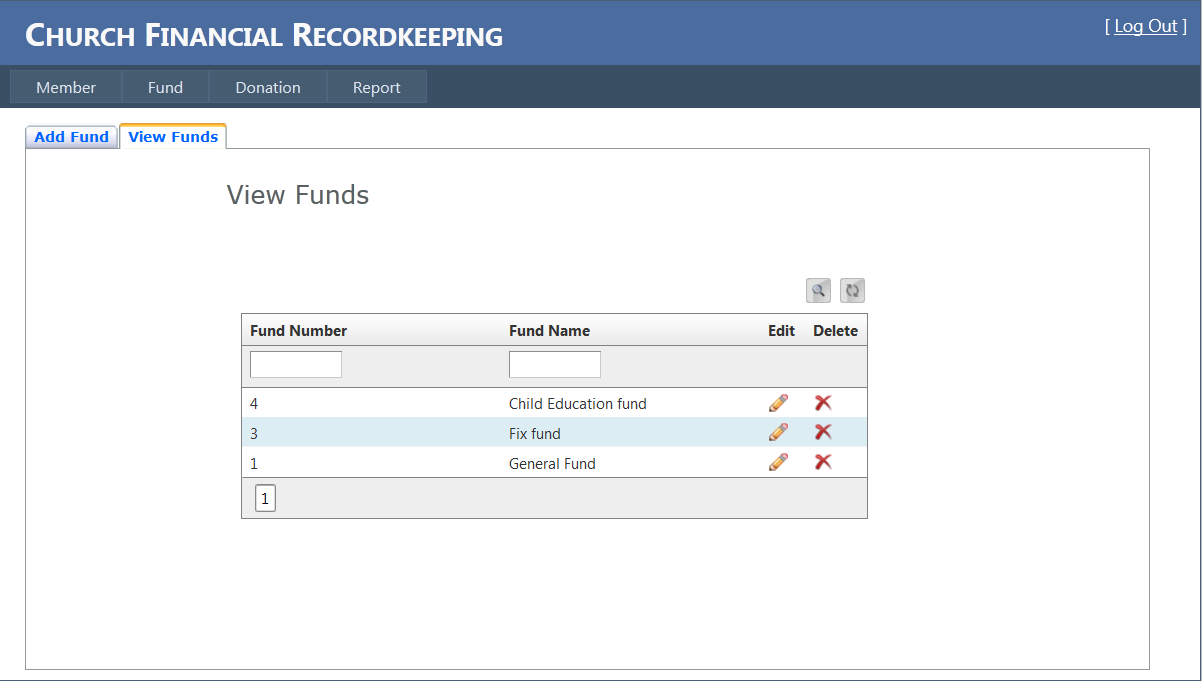


Figure 6: View Fund

The type of funds of the church are stored and used in the system. Different type of funds can be created, modified and deleted as per user needs.

Fund page contains:

* Text boxes for storing fund information like name and number.
* Grid display is used for displaying the stored fund records.
* The records can be edited and printed using buttons present in the grid.
* The objects like member pages can be printed, can be used to search and refresh the data.

The figure 7 is the donation screen. The donation screen is used to record the financial data of each donation made to church.

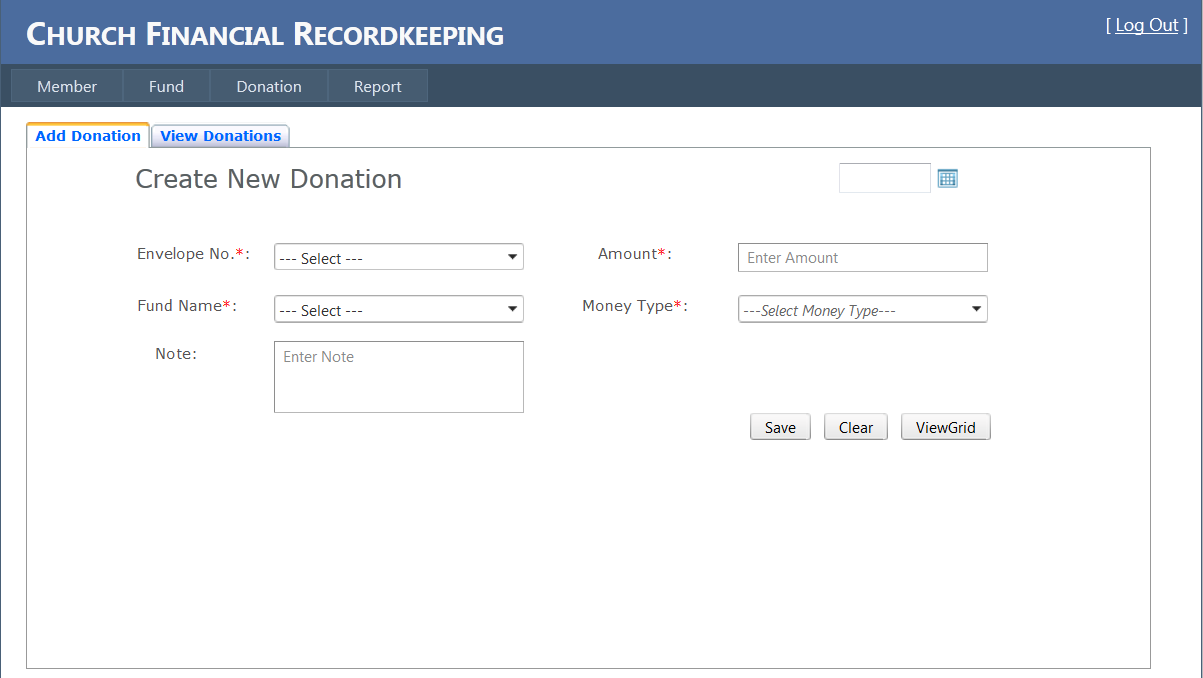


Figure 7: Add Donation

The figure 8 is the donation screen used to edit or delete the financial data of each donation made to church.

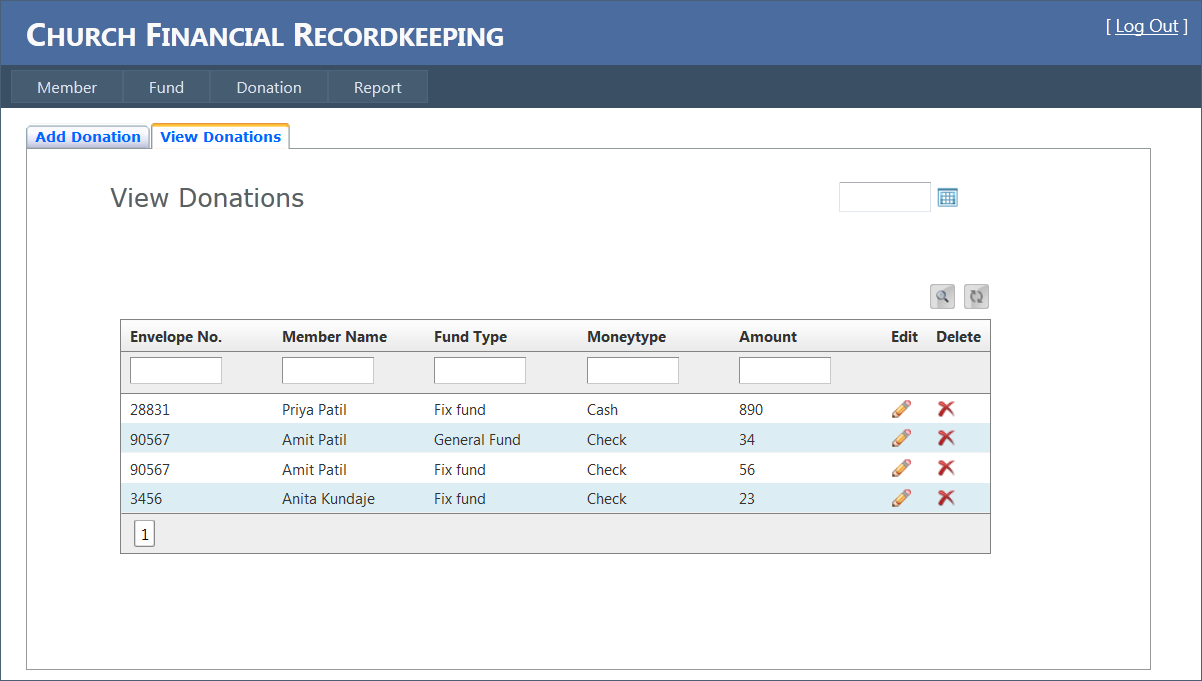


Figure 8: View Donation

The donation screen interface is the main page in the system. Member fund and donation information is put together.

Donation page contains:

* Drop down text field for envelope number to display and select the envelope number.
* Drop down text field to select appropriate fund.
* Drop down text field to select type of payment method (money type).
* Button, save, to save the fields information into database for that donation.
* Clear button to clear the text fields of the donation.
* Grid display used to fetch and display the donations information from the database.
  + Edit and delete button to edit and delete the donation record in grid.
  + Print button to print the list of donations.
  + Search button to search donation in database.
  + Refresh button to display recent donations on screen.

The figure below is the Report screen. The reports are generated for the donation received from time to time.

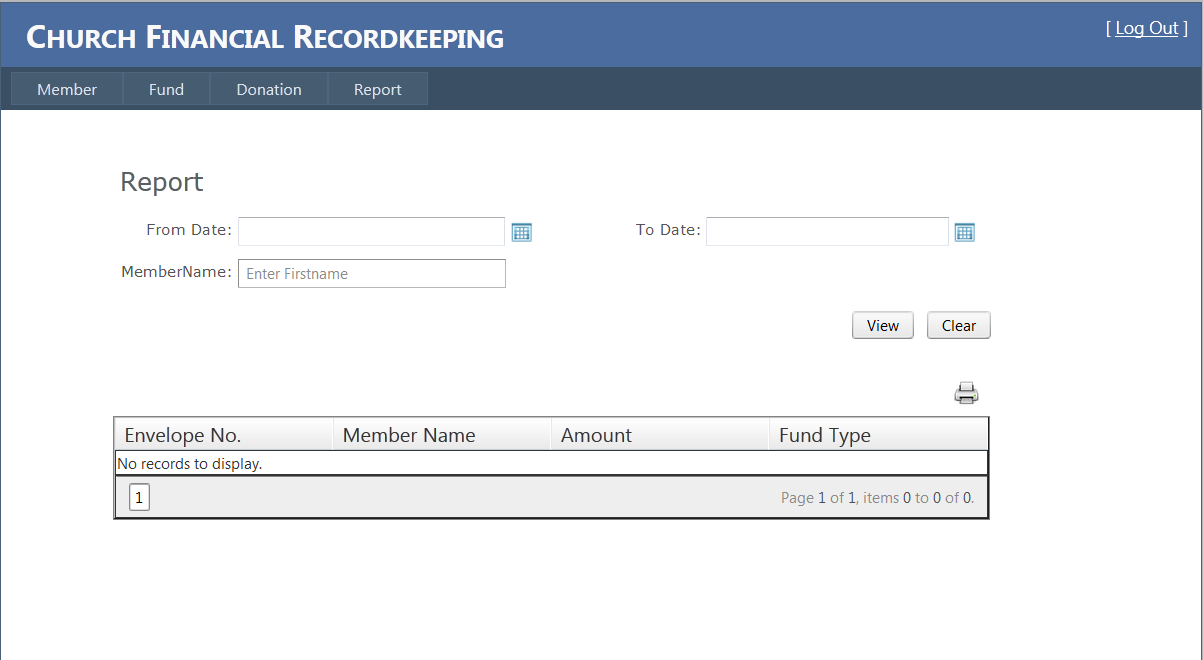


Figure 9: Report

The report is generated considering the values in the text field for a particular member and the date field to get the duration for which donations were received. Print button is provided to provide printing facilities.

In order to maintain consistency, it is decided to maintain uniformity in the functionality of most commonly used button in the interface. Buttons that have been identified are: Save, Clear, Cancel, Previous, and Submit.

An example report generated for the donations is shown below.

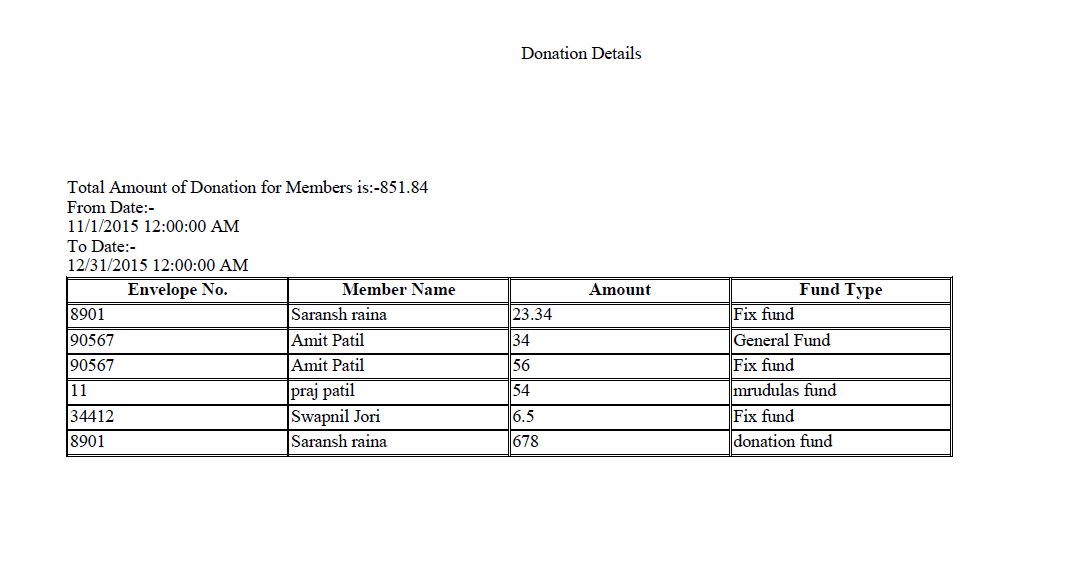


Figure 10: Sample Report

The functionality identified for the buttons are:

Save: It will take the input from the fields and store it into proper index database.

Clear: It will clear all the field that contain data.

Cancel: It will cancel the current transaction and display the main menu.

Previous: It will display the previous screen. Submit: It will display the next screen in sequence.

**4. Restrictions**

There are certain unexpected restrictions that may arise and affect developing. Proper planning and analysis is done and revised from time to time basis to solve and replace any restriction in future development.

**5. Limitations**

The application is designed with the features in consideration. Good efforts are put in for proper implementation of functionality. Tools and software's are used for developing the application. Limitations may come in developing, as software's and tools used may not contain 100% features. They may lack few functionalities and some features or options that would be implemented in some other way which may differ from its designs and planning.

**6. Constraints**

The application software developed can face some constraints like time, efforts, and skills. Proper design, analysis and planning is done to avoid any constraints in project development and efforts are put to complete the project in time, with proper functionality needed by the client.

# 7. Glossary

Client – The party for which services are rendered.

Database – The file containing the collection of data for the software.

SQL Server – Database management software.

Microsoft® Project – Scheduling software from Microsoft Corporation.

Microsoft® Visual Studio – It’s a framework which will be used in designing and coding from Microsoft Corporation.

C#.net- It is Programming language provided by Microsoft.

Microsoft® Windows – A family of operating systems from Microsoft Corporation.

Waterfall Process Model **-** Framework for managing product development.

Requirements Document – An abstract system definition to help the client understand what the software will do.

Specification Document – A detailed statement outlining the software’s functionality and its constraints.

Team leader – Leader of the software development group.

Timeline – Schedule detailing a chronological sequence of events.

**8. References**

# McGraw, R. J. (2015, june 15). *FOXMIS*. Retrieved october 27, 2015, from Google: http://community.mis.temple.edu/mis2101sec711summer2015/

* Larman, C., Applying UML and Patterns: Introduction to Object-Oriented Analysis and Design and Iterative Development, 3rdedition (October 2004).
* Sommerville, I., Software Engineering 9th edition book, Pearson (March 2004).