I. Problem Description

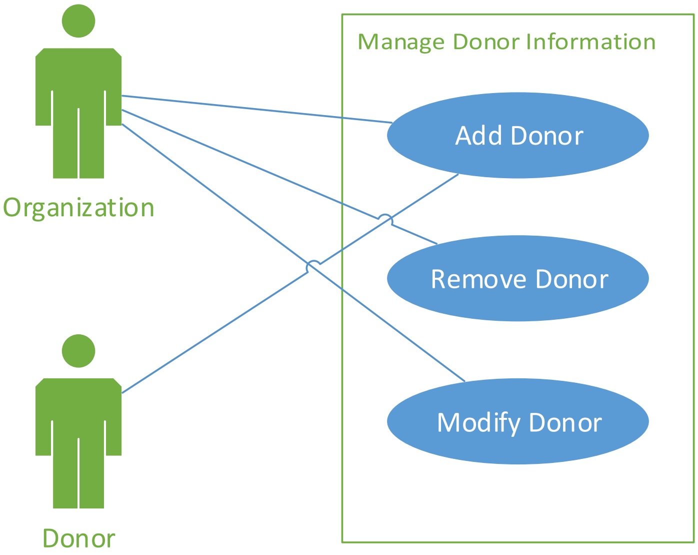
After discussing the problem and hearing out the requirements we came to conclusion that the problem will be solved in a manner of implementing a MySQL database and using PHP to make do all the coding required for the software. The Donor Management Software will be hosted on the Apache servers at Christian Brothers Service's(CBS) office in Romeoville as Software as a Service(SaaS). Clients will access the software from their own machines using a portal and they will receive dynamic web pages created using PHP on the CBS servers. All the coding will be done using PHP and HTML, meaning that all the execution will be done at run-time. This front-end of the system will work in conjunction with the back-end of the system, a MySQL database.

The CBS server will also feature a MySQL database for the donor, event, and campaign records to be stored. The database will have table for donor info, event info, soft credits, pledges, and many other records which Christian Brothers will require to store. A bridge between the SaaS and the MySQL database will be written in PHP. When software will receive a request it will automatically generate a query and send it to the database server where it will be executed. Results of the query will be returned back to the software and displayed for the user.

II. Functional Requirements

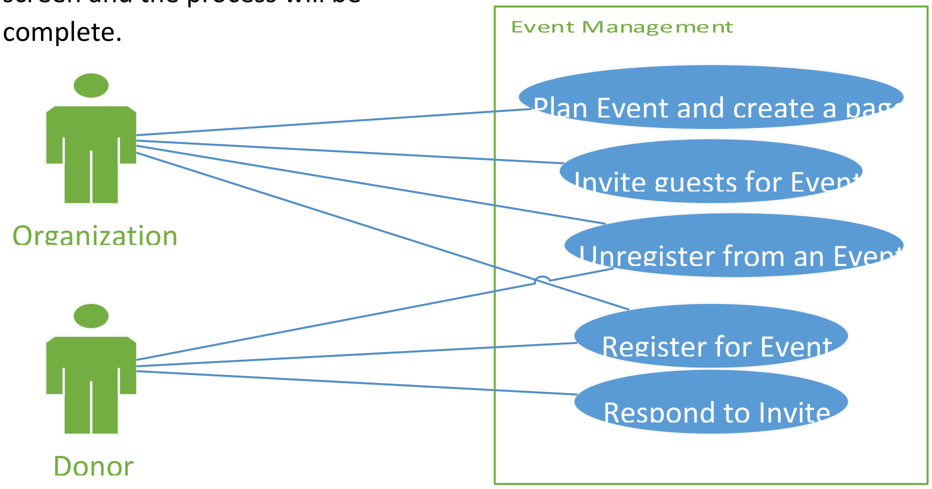
Manage donor information

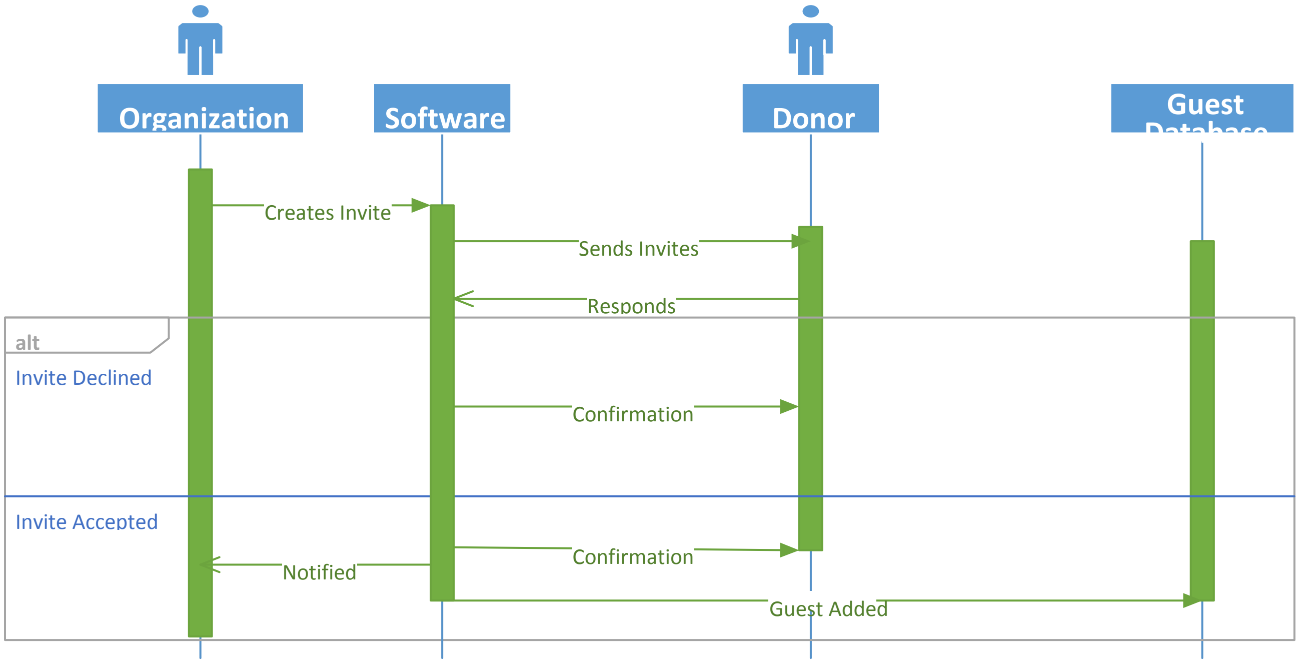
The software will be able to add, delete, and modify information for each of the donors who are part of the system. As changes are going to be made, the information will be written to the database which will be created to store donor info. The donor's themselves should be able to become a donor by filling an online application which will send the data to the database of the system and thus make a person a donor.



Event Planner

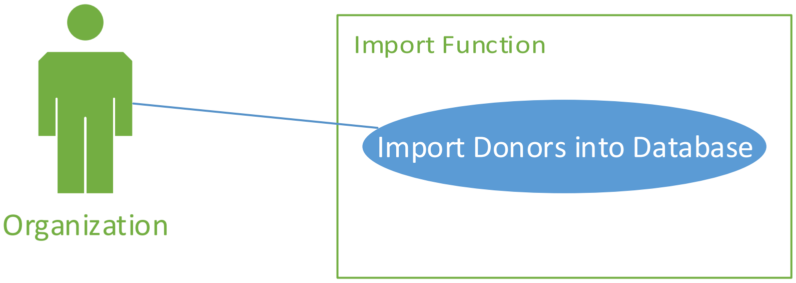
The organization will be able to create an event for which it will be able to send out Invitations. As the page will be created it will also generate a database table which will store guests who accepted the invitations. The software will be able to send invitations either to all the members of the donor database, or only to select ones chosen using targeting tools. Results of targeting tools, a list of email addresses, will be used to send out invitations. When a donor will get the invite he will use a link to connect to the service provided by software where he would be able to either accept or decline an invite. Other donors who did not receive an invited will also be able to register for event by using a link posted on the event website which will redirect the donor to the registration page. If the donor will not be part of the database of donors, he will be added to it based on the info he provided at the time of registry. After the donor accepts the invite and registers for event he will be added to the guest database table for that event. Also a confirmation will be displayed and a email with confirmation will be send to the donor with a link to unregister from the eventincase the donor will not be able to make it. F he decides to unregister his name will be removed from the guest table. If the donor decides to decline the invite he will receive the confirmation on screen and the process will be complete.





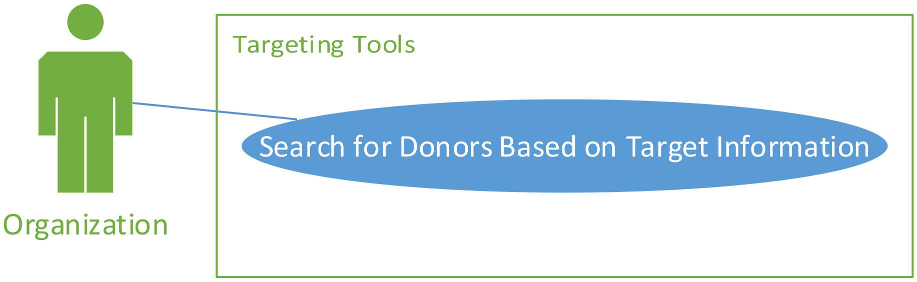
Import function

The software will be equipped with an easier way to add the customers to the database by using an import function. Import function will take an existing list of donors and it will map the fields to the corresponding fields in the database so that the organization could easily add the donors it already has without a hustle. When the fields will be mapped the software will generate the new records and add them to the database.



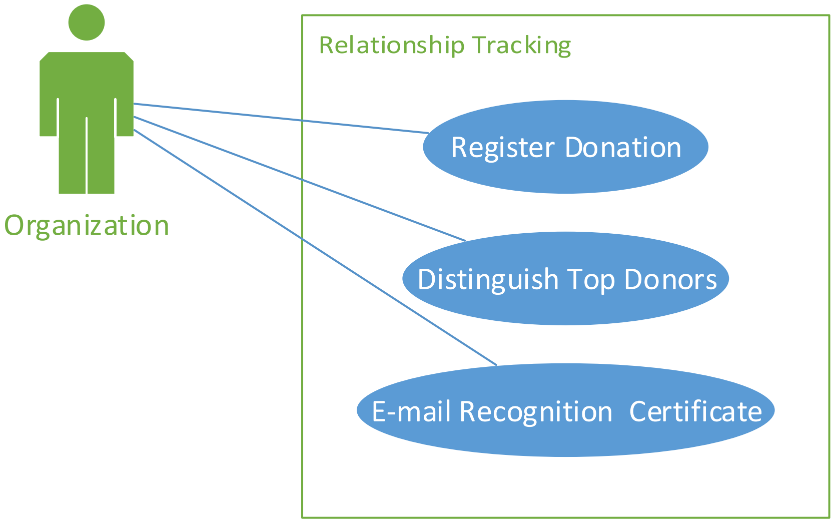
Targeting tools

The organization should be able to filter the donors using some kind of specific criteria so that only the donors who might be interested in the event be contacted based on the donors preferences and interest. Info should be gathered using a short survey that the user could answer during registration process. These tools will allow the user to search for donors based on specified criteria. When the search criteria will be generated the software will generate an appropriate select query and return a list of results for the user. The results should be a list of email addresses of the donors who met the criteria specified in the select query.



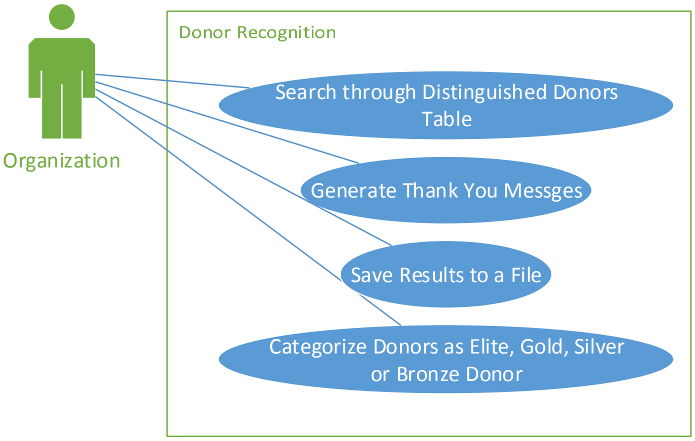
Relationship Tracking

The software should be able to keep track of all the donations given to the organization by the donor. Statistics should be kept in a separate table of the database and used to distinguish the top donors from the rest, so that the elite could be recognized by the organization during events or via email certificates. Information like total donation, donation this month, average donation, donation this year, last donation date, and last donation amount should be all kept in separate columns of the database table.



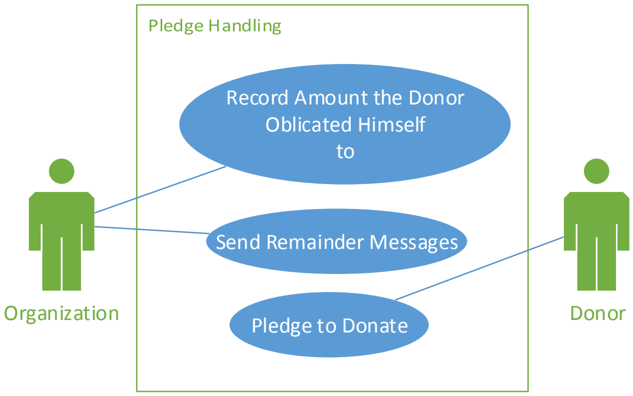
Donor recognition

The software should be able to search through the table to distinguish donors based on the criteria specified by the user. These donors could be then recognized by the organization as elite, gold, silver, and bronze donors based on the amount of money they donated to the organization during certain time period. After returning the list of names from the database the software will come with ability to generate a thank you message for the donor as well as ability to save the results in a file.


Organization
Donor Recognition
Search through Distinguished Donors Table
Generate Thank You Messges
Save Results to a File
Categorize Donors as Elite, Gold, Silver or Bronze Donor

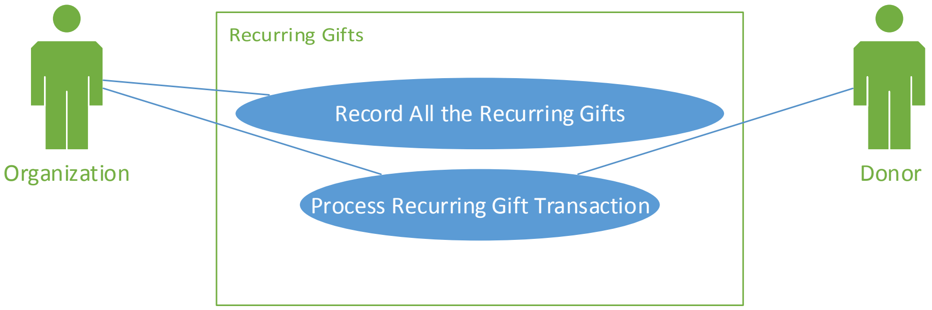
Handles Pledges

The software will keep track of all the donations to which the donors obligated himself or herself to. A separate database table should be used to keep track of the donor and the amount of obligation, amount paid, and the amount left to pay. This table should be usable in conjunction with targeting tools to generate a remainder email for the donors who pledged to pay money but did not do so.



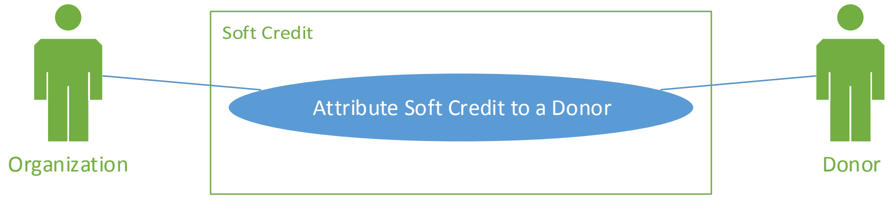
Handles recurring gifts

The software will be designed to handle transactions that will be scheduled on certain date for a specific period of time. A form will be available to users so that they could specify the amount of money they want to donate, the day of the month they want the donation to occur, and number of donations the donor wants to generate. Furthermore a copy of the document should be stored in the database as well as all the information specified above and the date on which the document was filed.


Recurring Gifts
Record All the Recurring Gifts
Process Recurring Gift Transaction
Organization
Donor

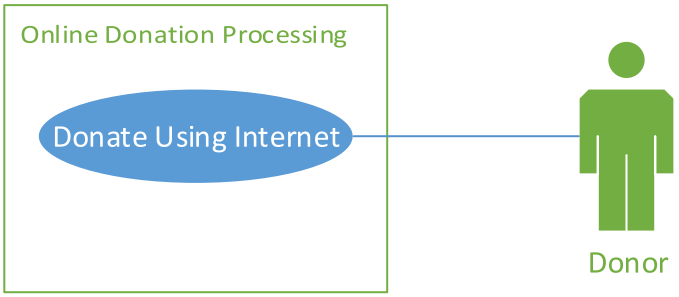
Tracks soft credits

The software will keep track of each of the donations that were given because of some other donor's influence. A separate database table should be kept to keep track of the amount of money which was generated because of particular donor, but not given by him or her. There should also be a field on the donation form on which the donor who is donating money could specify a donor who influenced him or her to donated money to organization. This field a dropdown list of all the names in the database. If a match will be found the donor's soft credit table will be updated with the appropriate amount, the total for soft credit, and the last modification date.



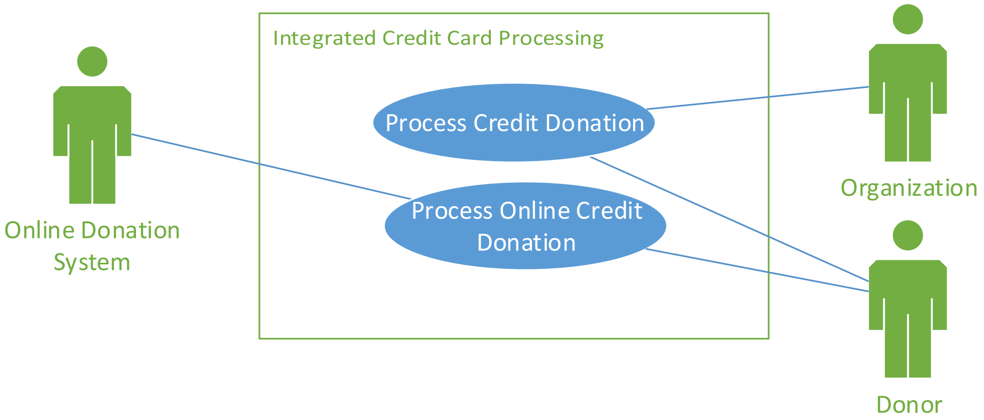
Online donation processing

The donors will be able to donate online by filling an online donation form which will work in conjunction with credit card processing to process the transaction. The processingwill take the information entered by the donor and add the donation to the database if the donor already exists. If the donation is made by a new donor, the information will be processed and the donor will be added to the database for future reference. After the donation will be complete the donor will get a confirmation message for the transaction. Also, when the process will end, a copy of receipt should be mailed to the email provided by the donor using tax receipt function of the software.



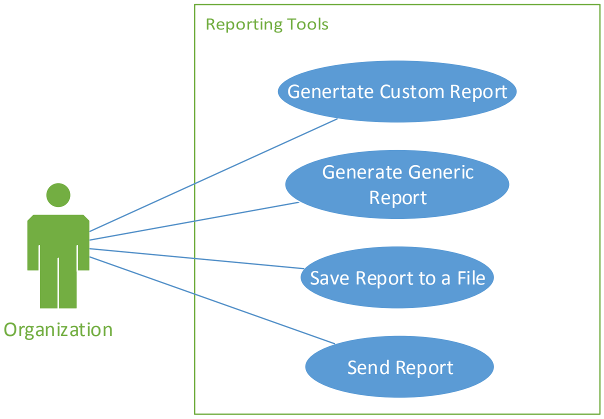
Integrated credit card processing

The software should be able to accept the credit card information and process the transactions made via credit. When a credit card will be used the software will use the integrated credit service to contact bank and acquire verification. If everything will be correct the user will get a confirmation message and transaction will be completed, otherwise it will be terminated. A database table should keep track of users account and other important information so that the user will not be required to enter the information each time that they want to donate money. This table should store values which will be hashed equivalents of info entered by the users.


Integrated Credit Card Processing
Process Credit Donation
Process Online Credit Donation
Online Donation System
Organization
Donor

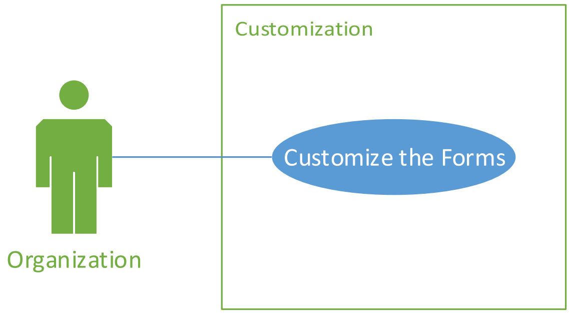
Reporting tools

The organization should be able to generate generic or custom reports based on criteria entered by the user. A list of criteria should be given to the user on the initial report creation page. When the criteria will be specified a query will be created in MySql and results of the query will be returned to the user in form of a report. User can then save it as a file, send it to someone, or print it out for records.



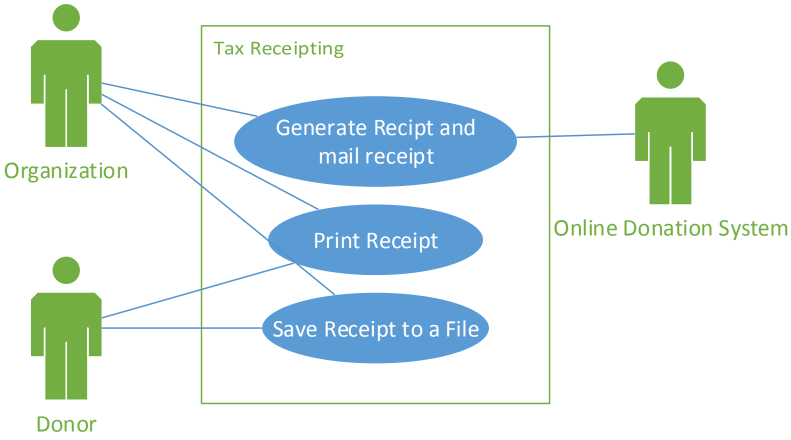
Customizable

All the forms in the software should be customizable; meaning that the fields could be labeled by the organization as desired. These labels should be matched to correspond to fields in the database created in MySql so that all the information could be kept as record.



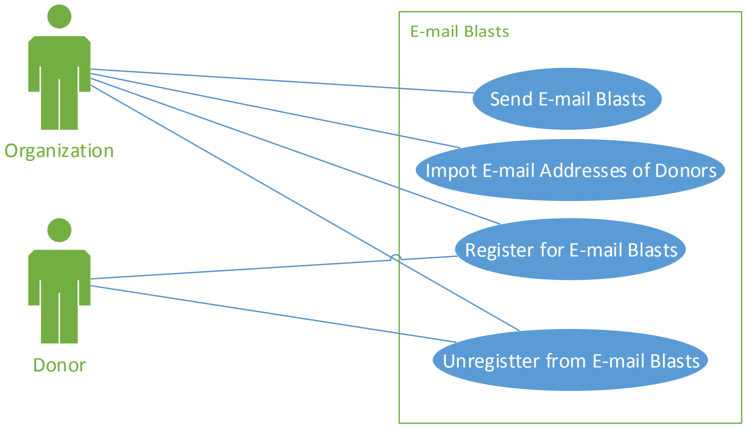
Tax receipting

The software should generate a tax exemption receipt for the donor when the donation process is complete or when the organization or donor will need to do so. It will take some of the data entered by the donor and map it over to the tax receipt form. When the process will be complete, it will generate receiptmail the copy of the receipt to the email provided by the donor, or the email stored on record in the donor database. The organization as well as the donor should be able to print he receipt if they desire to do so or save it to a file on a storage device.



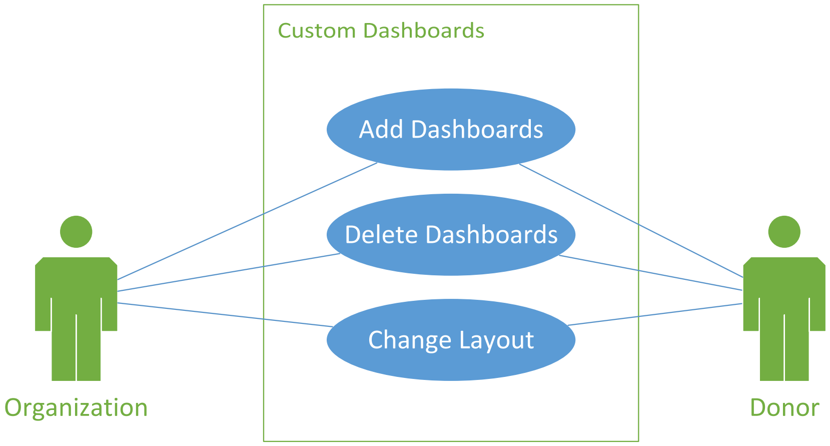
Email Blasts

The software will be able to generate a message sent to all recipients with one click of a mouse. When the organization will generate an email blast, the software will connect to the database, automatically select all email addresses available in the database which are marked for email blasts, and input the those email addresses into the “To:” field of email creation form. The donors as well as the organization will be able to register or unregister from email blasts at any given time. When a donor or an organization will register a donor for email blasts, a field in the database table will be set to true value. When they will try to unregister from the service, the field will be se to false and emails will not be selected for email blasting.


Organization
Donor
E-mail Blasts
Send E-mail Blasts
Impot E-mail Addresses of Donors
Register for E-mail Blasts
Unregistter from E-mail Blasts

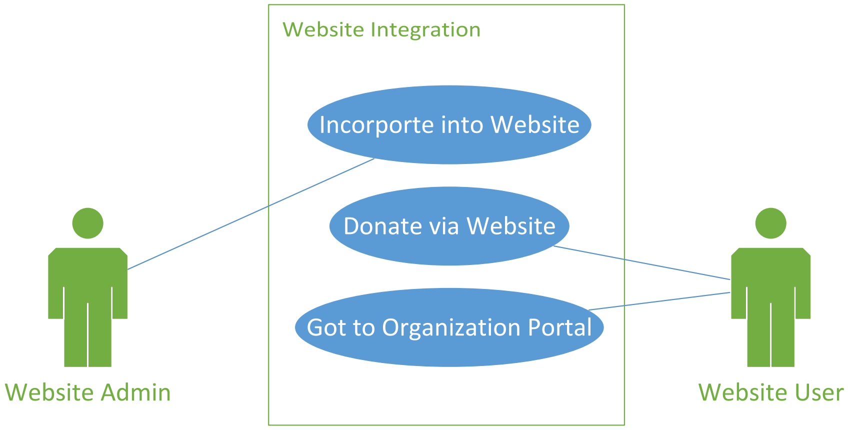
Custom Dashboards

The software will be designed so that the donor organization will be able to rearrange the dashboards to meet expectations of the user. The dashboards could be switched on and off as desired depending if the organization has a use for particular dashboard or not. Different fields, modules, and quick lunch menus could be added to make access to resources as easy as possible.



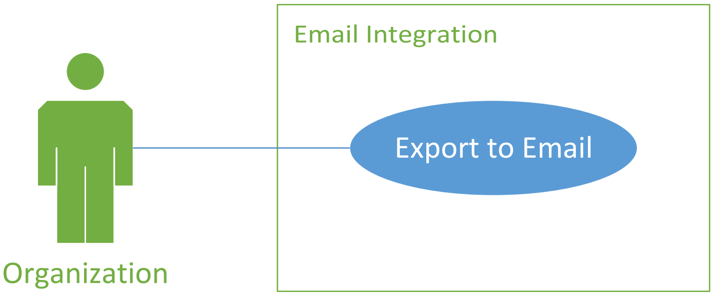
Website Integration

The application should be easy to integrate in to a website. It should come with source code that could be copied and pasted into the website source code to generate an applet for application. This applet will come with a short description and functions to process online donations for the organization. This will allow it to be incorporated into other websites by simply incorporating the code in website's html code. When donor will use the code applet, it will then use the integrated credit card processing to process the donation. The applet will contact the database and try the donor in the database. If found, the database will be updated with the donation amount and date, of the last donation but f the donor will not be found in the database, the donor will be added to the database by executing add query in the background. After the donation will be complete, a confirmation message will be displayed and the receipt module of the software will generate a receipt and send it to the email address provided by donor.



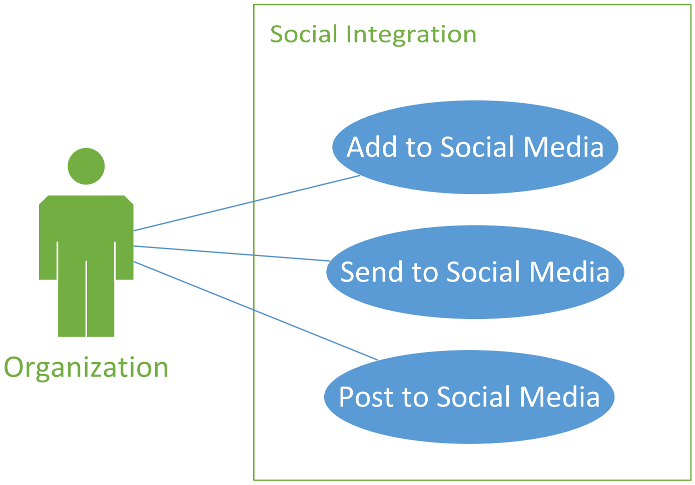
Email Integration

A user should be able to send data to email with a simple click of a button. A raport or any other data could be easily imported to an email by using the fact that the email is integrated in the software. It will take the data you want to email and it will automatically send it to email as a text or an attachment if user is working with a file. This button will automatically open an email session and attach the file as an attachment.



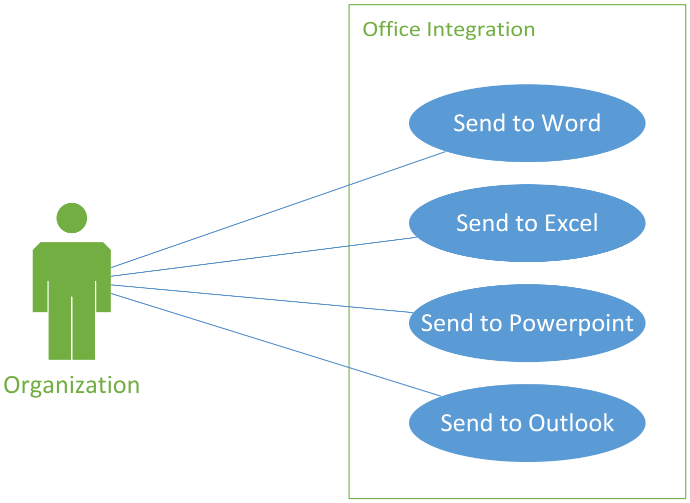
Social integration

Ability to integrate the software in social media such us facebook or twitter should be implemented so that the donor could use the social media to donate the money. It will also create a greater outreach and allow to contact and keep in touch with donors via social media. This will allow the or organization to post to social media with a a click of a button.



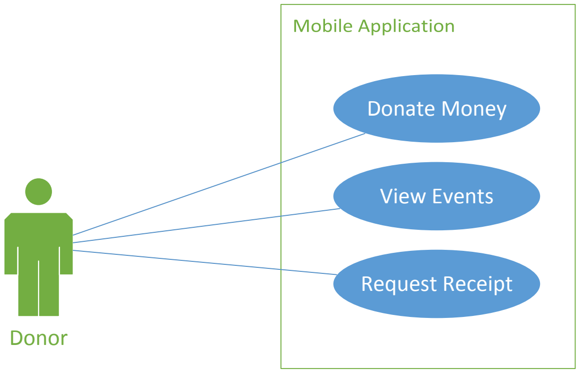
Office integration

The software should be able to support the file types supported by Microsoft Office. User should be able to send a created file, a report for example, to one of the Microsoft Office programs with one click of a button. It will take the data on the report and send it to MS Office file.



Mobile Application

The donor software should have a mobile application which could work in conjunction with software to allow the donor to donate money via phone application. This software will connect to the software via portal to enable access to services available for donors.



III. Non-Functional Requirements

Accessibility

The service should be available 24 hours a day so that donors could donate money at any time of the day. The servers should have a backup and backup power source, so in case of power outage or system crash the service could be restored as quickly as possible.

PHP

The SaaS(Software as a service) will utilize PHP-programming to create dynamic web-pages on the server. A bridge to the database will also be created using PHP-programming as well as all of the coding for the software.

MySQL

All of the database creation and implementation for the back-end of the system will be done using MySQL. The database table queries will also be structured using MySQL commands, so it could be run by the back-end when the front-end generates the query.

Data Retention

Data retention is to be in accordance with State and federal laws which means that the records should be kept for 3 years for donor information and lifetime for DMS users.

Server Type

The server that will be used is supposed to be an Apache Server.

Security

Since we will be handling people's personal information and money, security will play a big role. First off, a login function for donors can be implemented. A password and username are the first form of defense. A password with at least 8 characters (1 digit, and 1 symbol) must be used. This login information must also be confirmed via email. This information will then be encrypted and stored in our database. The password must then be changed every month or so. Another way we can prevent breaches is to include a "Last login: " function for the user profile. This will allow user to view recent activity on their account.

It is our duty to ensure the information flowing through the application is safe and secure. The database will be very important. We will have to store all the donors' personal information in a database, so it is crucial that the information is secure. One way we could secure this information is to encrypt it. This does not fully protect the information, but it will make it nearly impossible to hack. Another way to secure this information is to have two separate databases to store the information. For example, we can have one database for the user profiles and passwords, and another for the monetary exchanges. This will make it harder for someone to hack and gather the necessary information to do harm.

Backup

All the data in the database should be backed up at least twice a month so that all the records will be protected in case of lose during system failure. This will ensure that the data is safe and that the system will be able to be restored quickly in case of system crash.