

# Software Requirements Specification

Version 1.0

May

Hand By Hand charities application

Angela Aladoud

Ruba Ramadan

Kinan Saad

Submitted in partial fulfillment

Of the requirements of  
CS 310 Software Engineering

# Table of Contents

Table of Contents .....	1
1.0. Introduction .....	2
1.1. Purpose .....	2
1.2. Scope of Project .....	2
1.3. Glossary .....	3
1.4. References .....	5
1.5. Overview of Document .....	5
2.0. Overall Description .....	6
2.1 System Environment .....	6
2.2 Functional Requirements Specification .....	7
2.2.1 Charity Use Case .....	8
Use case: Manage charity's Account .....	8
Use case: Create Account .....	9
Use case: Delete Account .....	10
Use case: Update Account .....	11
2.2.2 Member Use Case .....	12
Use case: Manage Member's Account .....	12
Use case: Create Account .....	12
Use case: Delete Account .....	13
Use case: Update Account .....	15
2.2.3 Donor Use Case .....	16
Use case: Create Account .....	16
Use case: Delete Account .....	17
Use case: Update Account .....	18
2.3 User Characteristics .....	19
2.4 Non-Functional Requirements .....	20
3.0. Requirements Specification .....	21
3.1 External Interface Requirements .....	21
3.2 Functional Requirements .....	22
3.2.1 Manage charity's Account .....	22
Create Charity's Account.....	22
Delete Charity's Account .....	23
Update Charity's Account.....	25
3.2.2 Manage Member's account .....	26
Create Member's account.....	26
Delete Member's account.....	29
Update Member's account.....	30
3.2.3 Manage Donor's Account.....	31
Create Donor's account.....	31
Delete Donor's account.....	33
Update Donor's account .....	35
3.3 Detailed Non-Functional Requirements .....	36
3.3.1 Logical Structure of the Data .....	36
3.3.2 Security .....	41

# **1.0. Introduction**

## ***1.1. Purpose:***

The purpose of this document is to present a detailed description of the Hand By Hand charities application. It will explain the purpose and features of the system, the interfaces of the system, what the system will do, the constraints under which it must operate and how the system will react to external stimuli. This document is intended for both the stakeholders and the developers of the system and will be proposed to the Regional Historical Society for its approval.

## ***1.2. Scope of Project***

This system is directed at charities that raise proposals to help the poor and needy it is also directed at people who wish to donate, as they are aware of the proposals sent by associations and leave them free to choose the assistance they wish to provide. It is also directed to all people who need financial assistance, food, or surgery ... as their information is registered with the associations in which they participated in order to provide them with the necessary assistance by these associations after receiving support from donors.

More specifically, this system is designed to allow a charity to communicate with donors and people, the software

will facilitate this communication. The system contains a database containing a list of Donors, proposal and Member.

### ***1.3. Glossary***

Member	People who need money, food, or surgery help and they send a help request to the charity
Donor	People who provide support to charities, whether it is with money, food, or anything else
charity	The organization that manage donations and communication with donors and members
Join to charity	This is the process in which the member join the association so that he can then interact with the active proposals
Active proposal	It is the document that the system manages and moves in, and the charity controls it, members and donors interact with it
Atlas cloud database	It is a database on Atlas Server Cloud that contains data of the association, members and donors
Software Requirements Specification	A document that completely describes all of the functions of a proposed system and the constraints under which it must operate. For example, this document.
User	Donor , Member or Charity
Stakeholder	Any person with an interest in the project who is not a developer.

Follow	This is the process in which the Donor Follow the Charity so that he can then interact with the active proposals and see all the proposals from all charity that follow its.
--------	--

## ***1.4. References***

IEEE. *IEEE Std 830-1998 IEEE Recommended Practice for Software Requirements Specifications*. IEEE Computer Society, 1998.

Terrastruct :

<https://app.terrastruct.com/diagrams/446321296/edit> .

SRSEExample-webapp.pdf

Software Engineering, Global Edition

## ***1.5. Overview of Document:***

The next chapter, the Overall Description section, of this document gives an overview of the functionality of the product. It describes the informal requirements and is used to establish a context for the technical requirements specification in the next chapter. The third chapter, Requirements Specification section, of this document is written primarily for the developers and describes in technical terms the details of the functionality of the product. Both sections of the document describe the same software product in its entirety, but are intended for different audiences and thus use different language.

## 2.0. Overall Description

### 2.1 System Environment

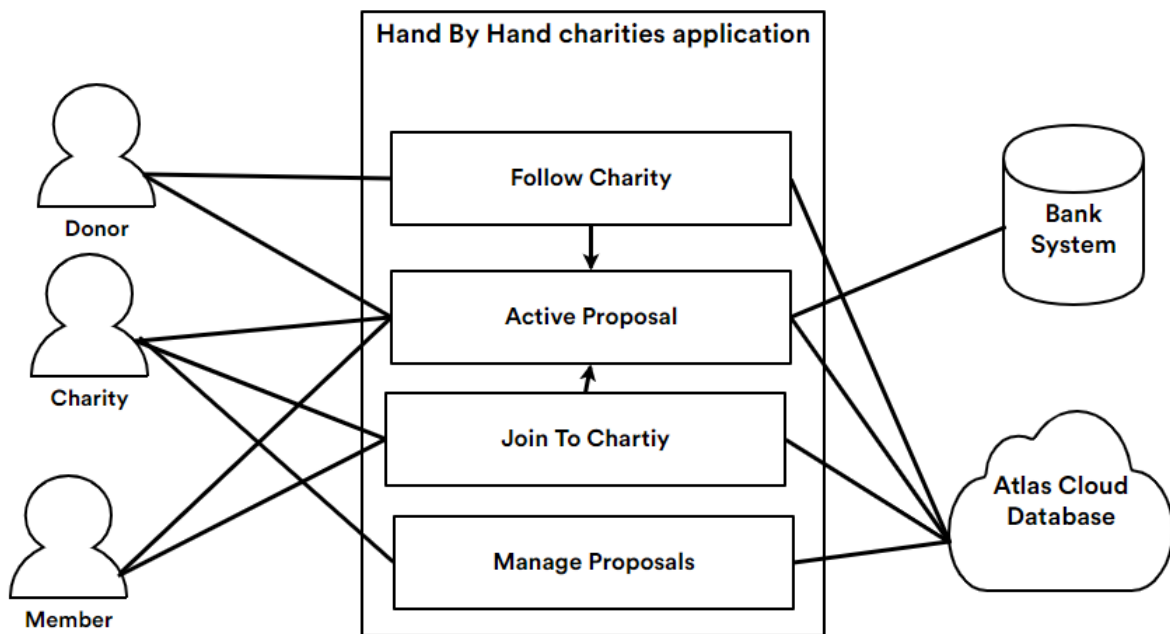


Figure 1 - System Environment

The Hand By Hand charities communication System has three active actors and one cooperating system. The Charity, Donors, Member, A cloud Atlas database and Bank System.

Charity manage the proposer and upload it to the internet after that can Members who join to charity and Donor who follow this charity to interact with active proposer. Any Donor or Member communication with the system is through email or username.

The Charity accesses the entire system directly. There is a link to the cloud Atlas database.

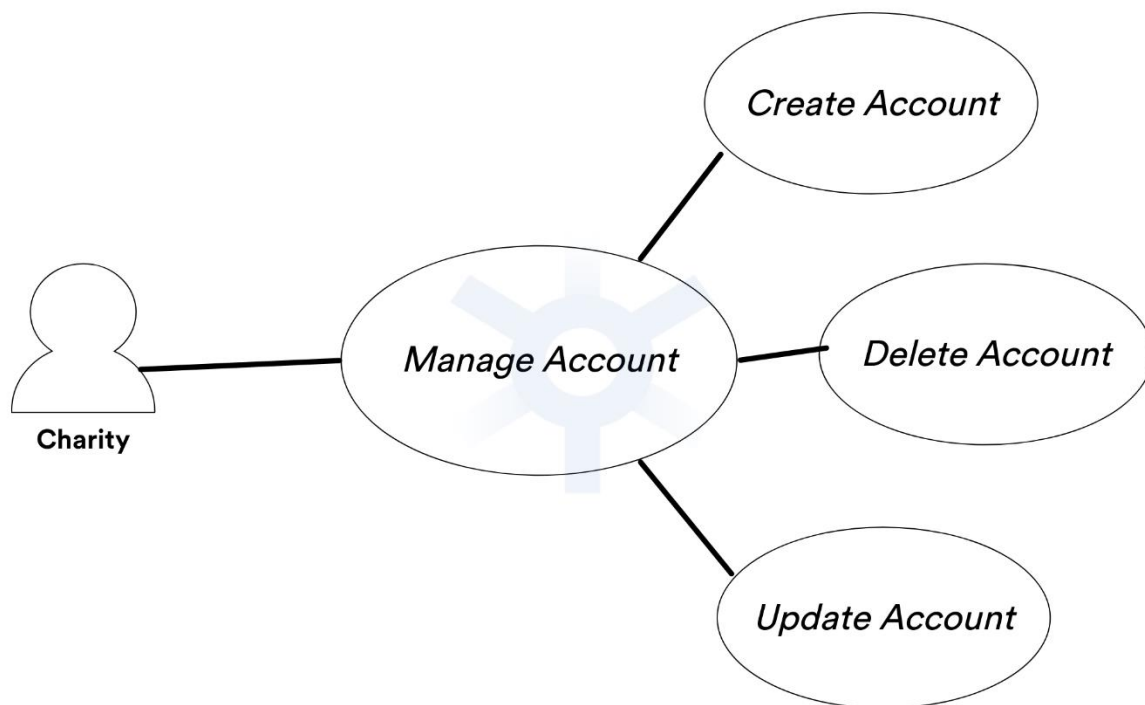
## ***2.2 Functional Requirements Specification***

This section outlines the use cases for each of Charity separately. The Donors, and the Member.



## 2.2.1 Charity Use Case

### Use case: Manage Account Diagram



**Figure 2 - Charity Manage Account**

## Use case: Create Account

### Diagram

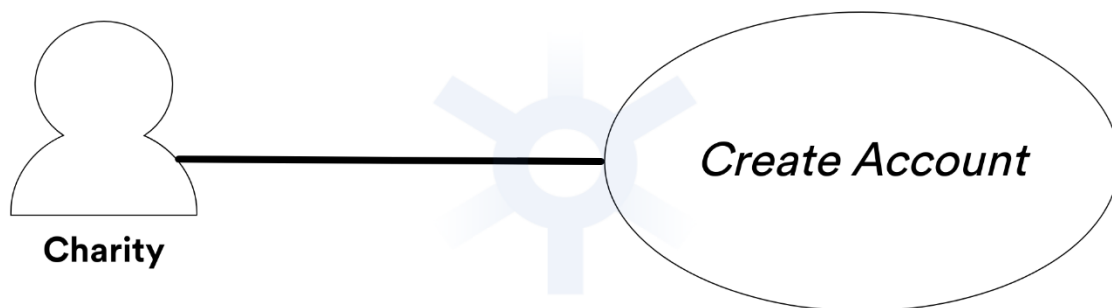


Figure 3 - Charity Create Account

### Brief Description

The charity add information to register in the application

### Initial Step-By-Step Description

1. The user select register as a charity
2. The user add personal information
3. The user click on add.
4. The system verifies the information.
5. The system add the authentication information to database

**Xref:** Section 3.2.1, Manage Account charity, create Account

Use case: Delete Account

## Diagram

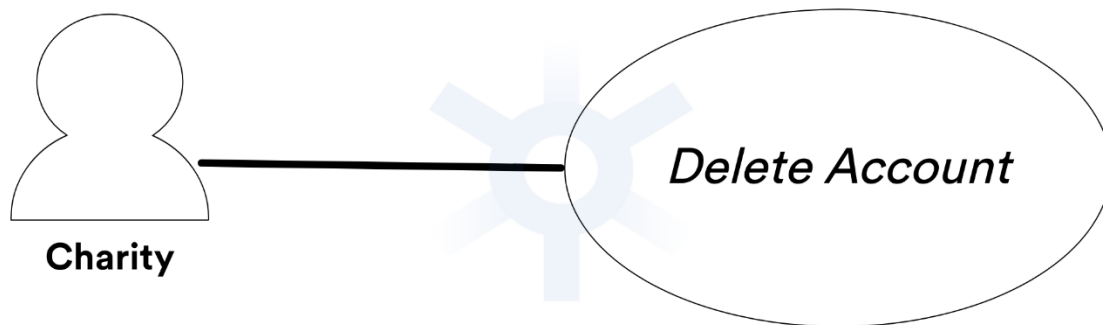


Figure 4 - Charity Delete Account

## Brief Description

The charity can delete their account on the database by system

## Initial Step-By-Step Description

Before this use case the charity has already accessed the main page of the application.

1. The charity click on tap profile
2. The charity click on delete bottom.
3. The charity input email address and password
4. The system check valid information

5. The system delete account and all information about this charity from database.

**Xref:** Section 3.2.1, Manage Account charity, delete Account

Use case: Update Account

## Diagram

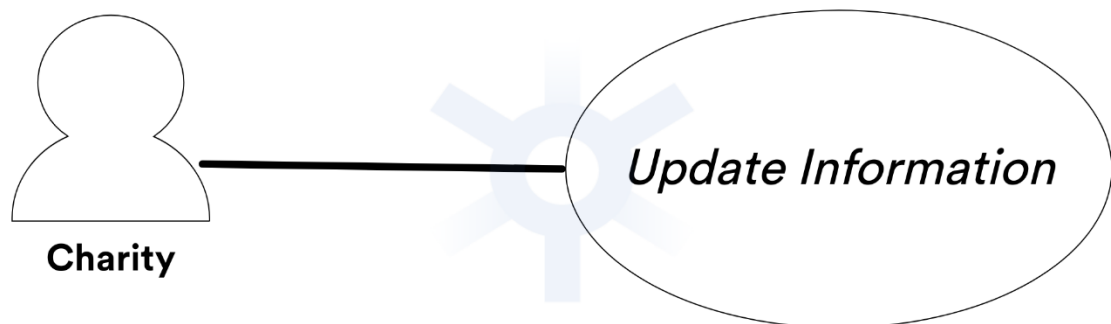


Figure 5 - Charity update Account

## Brief Description

The charity can add or update information about current charity.

## Initial Step-By-Step Description

Before this use case the charity has already accessed the main page of the application.

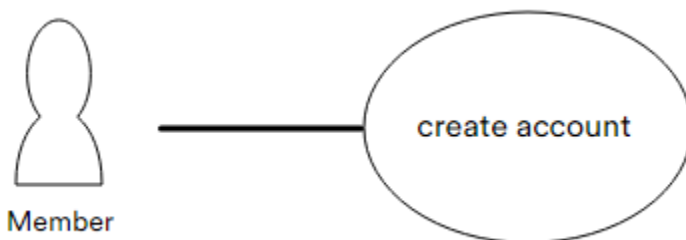
1. The charity click on tap profile.
2. Update the information that want to update it.
3. The charity click on the save bottom.
4. The system verifies the information.
5. The system update this information on database.

**Xref:** Section 3.2.1, Manage Account charity, update Account

## 2.2.2 Member Use Case

**Use case:** create account

**Diagram:**



**Brief Description:**

The member creates an account on the application after installing this application on his device in order to be able to interact with the services provided by this application

**Initial step-by-step description:**

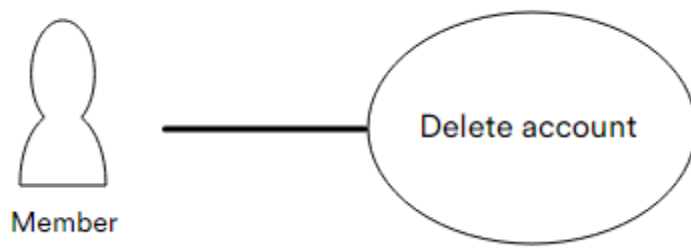
Before starting this use case, the member has previously installed the application on his mobile phone or accessed the website via the Internet.

1. The member chooses to register an account in the application as a member.
2. The system poses several questions for the member to answer.
3. Then the system saves the member's information.
4. The system displays the member's home page, where there are several tabs that enable the user to view his profile, view charities, or modify his information.
5. The member is allowed freedom of movement between tabs.

Xref: Section 3.2.2 Manage Account Member, Create member account.

## **Use case: Delete member account**

### **Diagram:**



### **Brief Description:**

A member can completely delete his account from the system

### **Initial step-by-step description:**

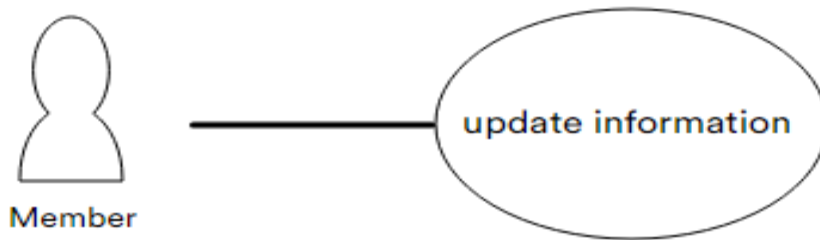
The condition for the member to use the deletion feature is that he already has an account

- 1- The member chooses to delete his account from the settings.
2. Then the system deletes all information related to this member
3. If the member wants to log in again, he must create a new account

Xref : Section 3.2.1 , Manage Account Member, Delete Account.

**Use case:** update member information

### **Diagram:**



### **Brief Description:**

A member can modify the information that he entered during the creation of his account

### **Initial step-by-step description:**

The member must be logged in from the account that he previously created

- 1- The member chooses to modify his information from the settings.
2. Then the system displays all its information and allows it to be modified
3. After modification and pressing the Save changes button, the system saves the new information

Instead of the old one

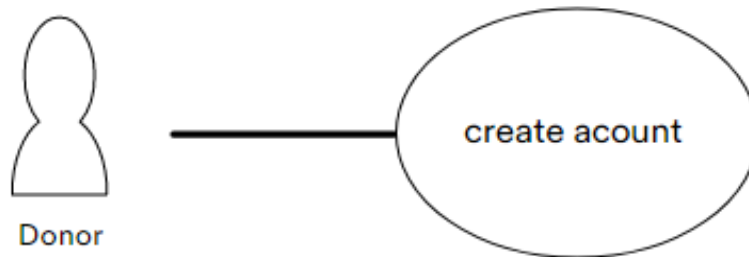
Xref: Section 3.2.1, Manage Account Member, Delete member account.



### 2.2.3 Donor Use Case

**Use case:** create account

**Diagram:**



**Brief Description:**

The donor creates an account on the application after installing this application or through the web page so that he can donate to charity through it.

**Initial step-by-step description:**

Before starting this use case, the donor had previously installed the app on their mobile phone or accessed their website online.

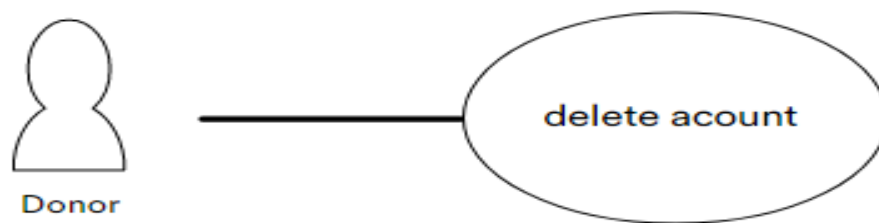
- 1- The donor chooses to register an account in the application as a donor.
2. The system poses several questions for the donor to answer.
3. Then the system saves the donor information.
4. The system displays the main page of the donor, where there are several tabs that enable the user to view his profile, view charities, or modify his information (settings).

5. The donor is allowed to move freely between tabs.

Xref : Section 3.2.3 ,Manage Donor's Account, Create Donor's account.

**Use case:** Delete donor account

**Diagram:**



**Brief Description:**

A donor can completely delete his account from the system.

**Initial step-by-step description:**

A requirement for the donor to use the delete feature is that they already have an account

1- The donor chooses to delete his account from the settings.

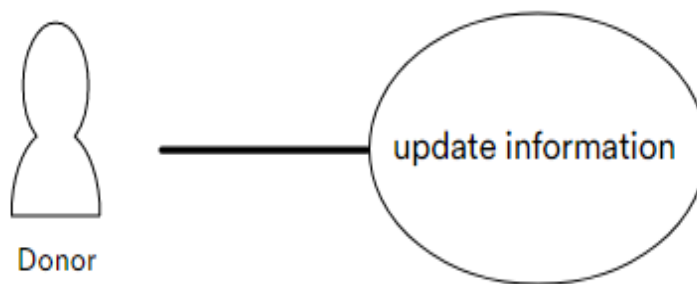
2. Then the system deletes all information related to this donor

3. If the donor wants to log in again, they must create a new account

Xref : Section 3.2.3 ,Manage Donor's Account, Delete Donor's account.

**Use case:** update donor information

**Diagram:**



**Brief Description:**

A donor can modify the information that he entered during the creation of his account

**Initial step-by-step description:**

The donor must be logged in from the account that previously created it

1- The donor chooses to edit his information from the settings.

2. Then the system displays all its information and allows modification

3. After modifying and pressing the Save Changes button, the system saves the new information

Instead of the old one

Xref : Section 3.2.3 ,Manage Donor's Account, Update Donor's Information.

## ***2.3 User Characteristics***

The charity is expected to be Internet literate and to be able to use email with attachments, and manage their bank account.

It is expected that both the member and the donor will be familiar with the Internet and able to interact with mobile applications or websites.

The company is expected to have a bank account and identify an employee who is able to work with the application or the website and who is very well versed in the Internet.

The donor is expected to have a bank account.

The detailed look of these pages is discussed in section 3.2 below.

## ***2.4 Non-Functional Requirements***

The system will be on Google Server Cloud. For connect to the system they need to access to the Internet.

To connect the system with Atlas Cloud Database must use mongoose tool.

The system will be able to communicate to the Bank System.

The System concerned with security statement limiting access to authorized donors or members.

The System the need to include user authentication facilities in the system.

The system will support both Arabic and English, and the user will be able to select the language he wants.

All users can search for associations by location by clicking on the place they want to search on the map, or specifying that through options within the application that contain the names of the regions.

The system will be available at all times 24/7 for all users, except for maintenance hours.

The system can be used at any time except during maintenance times and from anywhere.

The response speed will not exceed 200ms in the worst case.

## **3.0. Requirements Specification**

### ***3.1 External Interface Requirements***

The links to an external systems are the link to the Atlas Cloud Database to verify the members and donors information or to add a new member's documentation or donor's documentation. Charity is imposing to the member to add a specific information to add them to the Database, and is imposing to the donor to add an email and bank account to add them to the database. The charity system send an ID for both member and donor to check them out to allow them to active with proposer.

The second link is to communicate with the bank system. The charity send (a donor's bank account) to the bank system to check them out.

## ***3.2 Functional Requirements***

### **3.2.1 Manage charity's Account charity**

Create Account:

<b>Use Case Name</b>	Create Account
<b>XRef</b>	Section 2.2.1, update Account
<b>Trigger</b>	The charity add account to the database as a charity
<b>Precondition</b>	The person select register as charity
<b>Basic Path</b>	<ol style="list-style-type: none"><li>1. The system display some information they needed to add it.</li><li>2. The charity add this information.</li><li>3. The system check if the information is valid.</li><li>4. The system add this information to the Atlas database</li></ol>
<b>Alternative Paths</b>	<ol style="list-style-type: none"><li>1. In step 2, if the charity does not answer all the questions asked and fills in all the data required to be entered, the charity cannot send the information he entered into the system.</li></ol>

	<p>2. The system puts warning messages in the fields that the charity did not enter, stating that these fields are required to be entered</p> <p>3. The user enters the data that the system requested and continues the implementation progress, as in steps 3, 4 and 5</p> <p>4. Step 3 In the event that the data entered by the charity exists before, the system will show her an error message stating that he already has the account that he must log in</p> <p>It will not succeed to create the account again</p> <p>He will not be allowed to use the application until he logs in</p>
<b>Postcondition</b>	The charity has been added to the database.
<b>Exception Paths</b>	The charity may abandon the operation at any time.
<b>Other</b>	The charity information charity's name, password, email, phone, location, publication number, publication date, established date, bank account and logo.

### Delete Account:

<b>Use Case Name</b>	Delete Account
<b>XRef</b>	Section 2.2.1, Delete Account



<b>Trigger</b>	The charity delete their account to the database as a charity
<b>Precondition</b>	Previously, log in to the system (after creating an account as a member)
<b>Basic Path</b>	<ol style="list-style-type: none"> <li>1. After the charity chooses to delete his account by entering the settings and then choose the delete account button.</li> <li>2. The charity enter the email or charity name and password.</li> <li>3. The system check out the information if it is valid</li> <li>4. the system displays a message confirming the deletion</li> <li>5. If the charity answers yes, the system deletes the charity data completely and securely.</li> <li>6. After deleting the account, it is no longer possible for the charity to log in from it</li> </ol>
<b>Alternative Paths</b>	If their information is un valid the system display message to rewrite a right information.
<b>Postcondition</b>	The charity has been deleted from the database.
<b>Exception Paths</b>	The charity may abandon the operation at any time.
<b>Other</b>	The charity information (charity's name or email) and password.

## Update Account:

<b>Use Case Name</b>	update Account
<b>XRef</b>	Section 2.2.1, Update Account
<b>Trigger</b>	The charity Update their account to the database as a charity
<b>Precondition</b>	Previously, log in to the system (after creating an account as a member)
<b>Basic Path</b>	<ol style="list-style-type: none"><li>1. After the charity choose to update her information by entering the settings and then pressing the edit information button.</li><li>2. The system displays all the data related to the charity that he previously entered upon creating the account with the ability to overwrite it (i.e. modify it)</li><li>3. When you click on the save changes button, the system saves the new changes in the database</li></ol>
<b>Alternative Paths</b>	If their information is un valid the system display message to rewrite a right information.
<b>Postcondition</b>	The charity has been updated her information in to the database.
<b>Exception Paths</b>	The charity may abandon the operation at any time.
<b>Other</b>	The charity information charity's name, password, email, phone, location, publication

	number, publication date, established date, bank account and logo.
--	--

### 3.2.2 Manage Account Member

#### Create Member's Account:

<b>Use Case Name</b>	<b>Create member's account</b>
<b>XRef</b>	Section 2.2.2, Manage Account, Create Member's account
<b>Trigger</b>	The member creates an account in the application or on the website
<b>Precondition</b>	installing this application on his device or accessing the website through the browser online
<b>Basic Path</b>	<ol style="list-style-type: none"> <li>1. After the member chooses to create an account as a member, the system asks questions to the member and it asks for data that he must enter</li> <li>2. If the member answers all the questions as necessary, the member can send the information he entered into the system</li> <li>3. The system checks the entered data, in case it was not previously present, then the member accepts as a user of the system and his information is saved in the database</li> </ol>

	<ol style="list-style-type: none"> <li>4. after the creation of an account is successful, the system creates a profile for this member</li> <li>5. The member enters his private profile and is now allowed to use the application and see the available charities</li> </ol>
<b>Alternative Paths</b>	<ol style="list-style-type: none"> <li>1. In step 2, if the member does not answer all the questions asked and fills in all the data required to be entered, the member cannot send the information he entered into the system.</li> <li>2. The system puts warning messages in the fields that the member did not enter, stating that these fields are required to be entered</li> <li>3. The user enters the data that the system requested and continues the implementation progress, as in steps 3, 4 and 5</li> <li>4. Step 3 In the event that the data entered by the member exists before, the system will show him an error message stating that he already has the account that he must log in It will not succeed to create the account again He will not be allowed to use the application until he logs in</li> </ol>
<b>Postcondition</b>	Create an account for the member and enter it with this account to the home page of the application or the website.

<b>Exception Paths</b>	Create an account for the member and enter it with this account to the home page of the application or the website.
<b>Other</b>	<p>Information required by the system in order to create an account for a member:</p> <p>username, password, phone, first name, last name, age, national number, healthy status,</p> <p>if he has a work: {work type and salary}</p> <p>if his father a live {father's Name, and if his father have a work: {his work type, salary}, healthy status for his father}</p> <p>if his mother a live: mother's Information {Name ,if she has a work: {work type, salary}, healthy status for mother}</p> <p>if there are overage in home: overage's Information {Name, Old, healthy status}</p> <p>If he is married and he is a man, he must enter the number of wives and information on each wife</p> <p>{Name, if she has a work {type of work, salary}, her health status} As for if she is a married woman, she enters the same information only once for her husband's sake.</p> <p>if he didn't have children he enters this information for each child {Name, Age and if this child has a work {work type, salary}, healthy status for child}, and information about his home {ownership, ventilated, sunny}</p>

--	--

### Delete Member's Account:

<b>Use Case Name</b>	<b>Delete member account</b>
<b>XRef</b>	Section 2.2.2, Manage Account, Delete Member Account
<b>Trigger</b>	The member Delete an account from the application or the website And it will be completely deleted from database
<b>Precondition</b>	Previously, log in to the system (after creating an account as a member)
<b>Basic Path</b>	<ol style="list-style-type: none"> <li>1.After the member chooses to delete his account by entering the settings and then pressing the delete account button, the system displays a message confirming the deletion</li> <li>2. If the member answers yes, the system deletes the member's data completely and securely, meaning that it deletes everything related to it from database, even requests for joining and membership in charities and requests for assistance that it has raised.</li> <li>3. After deleting the account, it is no longer possible for the member to log in from it</li> </ol>

<b>Alternative Paths</b>	In step 2, if the member answered no, nothing happens, and the user returns to his previous state before choosing the Delete Account button
<b>Postcondition</b>	The member is deleted from the database and logged out from the application and website
<b>Exception Paths</b>	A member can only retract the deletion before confirming it.
<b>Other</b>	The required information (member's user name or email) and password.

### Update Member's information:

<b>Use Case Name</b>	Update Member's information:
<b>XRef</b>	Section 2.2.2, Manage Member's Account, Update Member's information
<b>Trigger</b>	The member modifies his information in the system They will be modified in the database
<b>Precondition</b>	Previously, log in to the system (after creating an account as a member)
<b>Basic Path</b>	<ol style="list-style-type: none"> <li>1. After the member chooses to amend his information by entering the settings and then pressing the edit information button,</li> <li>2. The system displays all the data related to the member that he previously entered upon creating the account with the ability to overwrite it (i.e. modify it)</li> </ol>

	<p>3 . The member modifies his data as he wants the existing one and presses the button to save modifications</p> <p>4. When he click on the Save changes button, the system saves the new changes in the database</p>
<b>Alternative Paths</b>	If their information is un valid the system display message to rewrite a right information.
<b>Postcondition</b>	New changes are saved in the system and dealings with the member are based on it
<b>Exception Paths</b>	A member can undo the saving of changes by pressing the back button before clicking on the save changes button
<b>Other</b>	The member can update all information he entered

### 3.2.3 Manage Donor's Account

Create Donor's Account:

<b>Use Case Name</b>	<b>Create Donors's account</b>
<b>XRef</b>	Section 2.2.3,Manage Donor's Account, Create Donor's account



<b>Trigger</b>	The donor creates an account in the application or on the website
<b>Precondition</b>	installing this application on his device or accessing the website through the browser online
<b>Basic Path</b>	<ol style="list-style-type: none"> <li>1- After the donor chooses to create an account as a donor, the system poses questions to him and asks him to enter some data</li> <li>2. If the donor answers all the required questions, he can send the information he entered into the system</li> <li>3. The system checks the entered data in the event that it is not there before, so the donor is accepted as a user of the system and his information is saved in the database.</li> <li>4. After successful account creation, the system creates a profile for this donor</li> <li>5. The donor enters his personal file and is now allowed to interact with the system.</li> </ol>
<b>Alternative Paths</b>	<ol style="list-style-type: none"> <li>1. In Step 2, if the donor does not answer all the questions asked and fills in all the data required to be entered, the donor will not be able to send the information he entered into the system.</li> <li>2. The system puts warning messages in the fields that the donor did not enter, stating that these fields are required to be entered</li> </ol>

	<p>3. The donor enters the data requested by the system and continues the progress of implementation, as in steps 3, 4 and 5</p> <p>4- Step 3 If there is the data entered by the donor before, the system will show him an error message stating that he already has the account that he should log in to</p> <p>The account will not be created again</p> <p>He will not be allowed to use the app until he logs in</p>
<b>Postcondition</b>	A donor account is created and this account is entered into the main page of the application or website.
<b>Exception Paths</b>	A donor may leave the account registration process at any time, so that he does not save his data.
<b>Other</b>	<p>Information required by the system in order to create an account for a donor: Donor name, phone, password, E-mail,</p> <p>And he chooses his capacity as a temporary or permanent donor</p>

Delete Donor's Account:

<b>Use Case Name</b>	<b>Delete Donor's account</b>
<b>XRef</b>	Section 2.2.3, Manage Donor's Account, Delete Donor's Account
<b>Trigger</b>	The donor Delete an account from the application or the website And it will be completely deleted from database
<b>Precondition</b>	Previously, log in to the system (after creating an account as a donor)
<b>Basic Path</b>	1- After the donor chooses to delete his account by entering the settings and then pressing the Delete Account button, the system displays a message confirming the deletion 2. If the donor answers yes, the system deletes the donor's data completely. 3. After deleting the account, the member can no longer log in from it
<b>Alternative Paths</b>	In step 2, if the donor answered no, nothing happens, and the user returns to his previous state before choosing the Delete Account button
<b>Postcondition</b>	The donor is deleted from the database and logged out from the application and website
<b>Exception Paths</b>	A donor can only retract the deletion before confirming it.
<b>Other</b>	The required information (donor's name or email) and password.

## Update Donor's information:

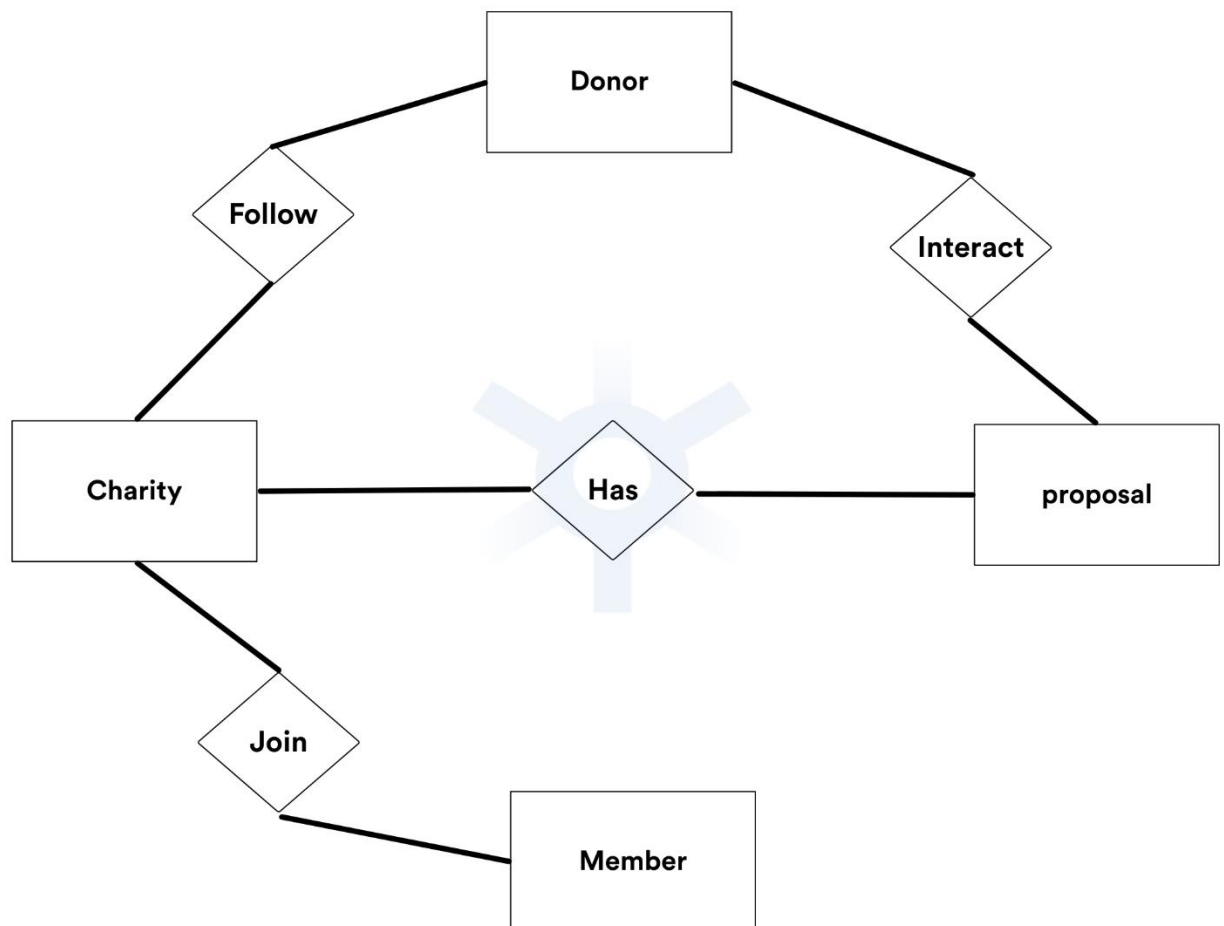
<b>Use Case Name</b>	Update Donor's information:
<b>XRef</b>	Section 2.2.3, Manage Donor's Account, Update Donor's information
<b>Trigger</b>	The donor modifies his information in the system They will be modified in the database
<b>Precondition</b>	Previously, log in to the system (after creating an account as a donor)
<b>Basic Path</b>	<ol style="list-style-type: none"> <li>1. After the donor chooses to amend his information by entering the settings and then pressing the edit information button,</li> <li>2. The system displays all the data related to the donor that he previously entered upon creating the account with the ability to overwrite it (i.e. modify it)</li> <li>3 . The donor modifies his data as he wants the existing one and presses the button to save modifications</li> <li>4. When donor click on the Save changes button, the system saves the new changes in the database</li> </ol>
<b>Alternative Paths</b>	If their information is un valid the system display message to rewrite a right information.
<b>Postcondition</b>	New changes are saved in the system and dealings with the donor are based on it
<b>Exception Paths</b>	A donor can undo the saving of changes by pressing the back button before clicking on the save changes button

<b>Other</b>	The donor can update all information he entered
--------------	---

### 3.3 Detailed Non-Functional Requirements

#### 3.3.1 Logical Structure of the Data

The logical structure of the data to be stored in the application database is given below.



The data descriptions of each of these data entities is as follows:

Charity Data Entity:

<b>Data Item</b>	<b>type</b>	<b>Description</b>	<b>comment</b>
Name	Text	Name of charity	
Password	Text	Password of the Charity	
Email	Text	Email of the charity	
Phone	Text	Phone of the charity	
Location	Text	Location of the charity	May contain a city, street and note
Publication Number	Text	Publication Number of charity	
Publication Date	Date	Publication Date of charity	

Established date	Date	Established date of charity	
Bank Account	Text	Bank Account of charity	
Logo	Image	Logo of charity	

#### Donor Data Entity:

<b>Data Item</b>	<b>type</b>	<b>Description</b>	<b>comment</b>
Name	Text	Name of Donor	
Password	Text	Password of the Donor	
Email	Text	Email of the Donor	
Phone	Text	Phone of the Donor	
Permanence	Boolean	Permanence of Donor	

#### Member Data Entity:

<b>Data Item</b>	<b>type</b>	<b>Description</b>	<b>comment</b>
User Name	Text	User Name of Member	
Password	Text	Password of the Member	

Email	Text	Email of the Member	
Phone	Text	Phone of the Member	
First Name	Text	First Name of Member	
Last Name	Text	Last Name of Member	
Age	Text	Age of Member	
Healthy Status	Text	Healthy Status of Member	
Work	Text	Work of Member	
National Number	Text	National Number of Member	
Married	List of map	Husband of Member	May be empty or more than one
Father	Map	Father of Member	
Mother	Map	Mother of Member	
Overage	List of map	Overage of Member	May be empty or more than one



Children	List of map	Children of Member	May be empty or more than one
House Property	Boolean	House Property of Member	
House ventilated	Boolean	House ventilated of Member	
House Sunny	Boolean	House Sunny of Member	

#### Proposal Data Entity:

<b>Data Item</b>	<b>type</b>	<b>Description</b>	<b>comment</b>
Proposal Type	Text	Type of proposal	May be more than one
proposal Amount	Integer	Amount of help	
Number of Beneficiaries	Integer	Number of people who Benefits from proposal	
Number of Donor	Integer	Number of donor who Donates to proposal	

### **3.3.2 Security:**

The server on which the application resides will have its own security to prevent unauthorized. The use an Email and password for Charity and Donor to login, and user name or phone and password for member to login.

The database will be on the Atlas Cloud Database that have its own security to store, delete and query to database.