

Software engineering project (phase 1)

Charity System

Dr.Lobna Mahdy

Team ID : 100

	Name	Section	Department	ID
1	أسماء سيد عبد الفتاح عبد الغفار	10	CS	20201701057
2	يوسف أحمد فؤاد عبدالعزيز	10	CS	20201700997
3	هند محمد مصطفى عبد الرحيم	10	IS	20201701180
4	محمد خالد سيد معوض	6	CS	20201700688
5	عبدالله مصطفى عبدالسلام أحمد	5	CS	20201700493
6	يوسف عماد الدين مسعد علي	10	CS	20201701023
7	ميّار هشام محمد جلال	5	Csys	20201700878
8	كريم السيد عبدالقوي السيد	6	CS	20201700600

(100) – Charity

Dr. Lobna Mahdy

	Name	Section	Department	ID
1	أسماء سيد عبد الفتاح عبد الغفار	10	CS	20201701057
2	يوسف أحمد فؤاد عبدالعزيز	10	CS	20201700997
3	هند محمد مصطفى عبد الرحيم	10	IS	20201701180
4	محمد خالد سيد معوض	6	CS	20201700688
5	عبدالله مصطفى عبدالسلام أحمد	5	CS	20201700493
6	يوسف عماد الدين مسعد علي	10	CS	20201701023
7	ميّار هشام محمد جلال	5	Csys	20201700878
8	كريم السيد عبدالقوي	6	CS	20201700600

1. Introduction:

A charity system is a software solution for a non-profit organization. It manages the charitable donations and also the donor base. Such a system includes features such as electronic fund transfer, donation management, fundraising, donor management, and event management. Time can be saved for more important areas of focus thanks to the automation of the manual processes involved in managing donors, several fund-raising campaigns, and various forms of financial management. The ability to create reports of any kind, especially those that can be customized to track the status of any aspect of the operation, is extremely important. This keeps the organization informed about its progress and financial management.

2. User Requirements:

- **employee**
 - As an **employee**, I want to be able to register and log into my account.
 - As an **employee**, I want to be able to add or delete events, and posts.
 - As an **employee**, I want to be able to make a fund request.
 - As an **employee**, I want to be able to view and analyse reports.
 - As an **employee**, I want to be able to confirm the charity case requests
 - As an **employee**, I want to be able to confirm my volunteer application
- **sponsor**
 - As a **sponsor**, I want to be able to register and log into my account.
 - As a **sponsor**, I want to be able to approve or reject fund request
 - As a **sponsor**, I want to be sure that my donation has arrived.
 - As a **sponsor**, I want to be able to track my donation records
- **Usual user**

- As a **Usual user**, I want to be able to register and log into my account.
- As a **Usual user**, I want to be able to view charity information, activities, events, brunches, videos, photos, and News
- As a **Usual user**, I want the communication information including social media to be available.
- As a **Usual user**, I want to be able to make a donation and use different ways of payment
- As a **Usual user**, I want to be sure that my donation has arrived
- As a **Usual user**, I want to be able to submit a volunteer application
- As a **Usual user**, I want to be able to apply new case request

3. Functional Requirements:

1. Register

- The user should fill out the registration form with his data in the specific format and the function verifies his identity by phone or e-mail before saving the data in the database
- The inputs: name, Username, e-mail, phone number, birth date, address, gender, password
- The source: employee or sponsor or user
Pre-condition: The user specifies if he is an employee or a sponsor or a user
- Post-condition: the user can log in
- The output: confirmation or rejection message

2- Log in

- The system must allow users to log into their accounts by entering their usernames and password.
- The system must allow users to reset their password by clicking on "I forgot my password" and receiving a link to their verified email address.
- The inputs: Username, password
- The source: employee or sponsor or user
- Pre-condition: the user should be registered
- Post-condition: the user can enter the system
- The output: confirmation or rejection message

3- Analyse the information

- The function generates some statistics and a summary of the organization
- The inputs: Data for generating a summary
- The source: employee
- Pre-condition: the employee logs in
- Post-condition: the employee can view the report
- The output: Analytical report

4- confirm volunteer application

- The employee reviews the applicant's information and accepts or reject the application
- **The inputs:** information on the Application for volunteering
- **The source:** Volunteer
- **Pre-condition:** the employee logs in and the Volunteer filled the application
- **Post-condition:** sending an e-mail to the applicant if (he\she) accepted so the volunteer can begin with the charity
- **The output:** the employee accepts or reject

5- Respond to fund request

- The sponsor should accept or reject the charity fund request
- **The inputs:** confirmation of State
- **The source:** The sponsor
- **Pre-condition:** the sponsor log in and the employee sent the fund request
- **Postcondition:** the employee receives the response
- **The output:** the sponsor's acceptance or rejection

6- Donate

- The donor should donate, and will be sent to a specific branch he will choose if he chooses to donate money, he must choose whether he will pay cash at the branch, send someone to him, or mobile wallet, then record the donation process in the database
- **The source:** the sponsor or the donor
- **The inputs:** The method of payment, donation, and the branch name(if needed)
- **Pre-condition:** the sponsor/donor log in
- **Postcondition:** record the donation and the employee can receive the money
- **The output:** Confirmation Message

7-View Communication info

- The communication information should be available and should be displayed to any user
- **The source:** Database
- **Pre-condition:** The User of the System Should be Logged in
- **Post-condition:** communication with the charity
- **The output:** Organization's contact information

8- Submit a volunteer application

- The applicant should fill out the application for volunteering then this information will be saved in the database.
- **The inputs:** name, username, e-mail, phone number, birth date, job, education, address, gender, available days for charitable work, branch, activity
- **The source:** Volunteer
- **Pre-condition:** his/her age must be over 15
- **Postcondition:** the employee can approve/disapprove the application

- **The output :** A message confirming the registration and it says, “Please wait for the response within a few days.”

9- Submit a new case request

- The function will save the new needy case in the database.
 - **The source:** recipient
 - **The inputs:** name, phone number, birth date, job, education, address, gender, nearest branches, the help type, case description
 - **Pre-condition:** all the information must be valid and meet certain conditions
 - **Postcondition:** A message confirming the ability to help
 - **The output:** A message confirming the request and telling the needy people that they will contact them
-

4. Non-Functional Requirements:

1-Database Security

- The system shall implement secure authentication and authorization mechanisms to ensure that only authorized users can access the Database.
- The system shall use HTTPS to encrypt all data transmitted between the user's browser and the server.
- The system shall use strong password encryption algorithms to store user passwords.

2- Performance

- The system shall be able to handle a large number of concurrent users.
- The system shall respond to user requests within 3 seconds.
- The system shall be scalable to accommodate future growth in the number of users and data volume.

3-Usability Requirement

- The system shall be easy to use and have a user-friendly interface.
- The system shall be accessible to users with disabilities.
- The system shall provide clear and concise error messages to users in case of errors or failures.

4- Reliability Requirement

- The system will perform without failure in 95 percent of use cases during a month.

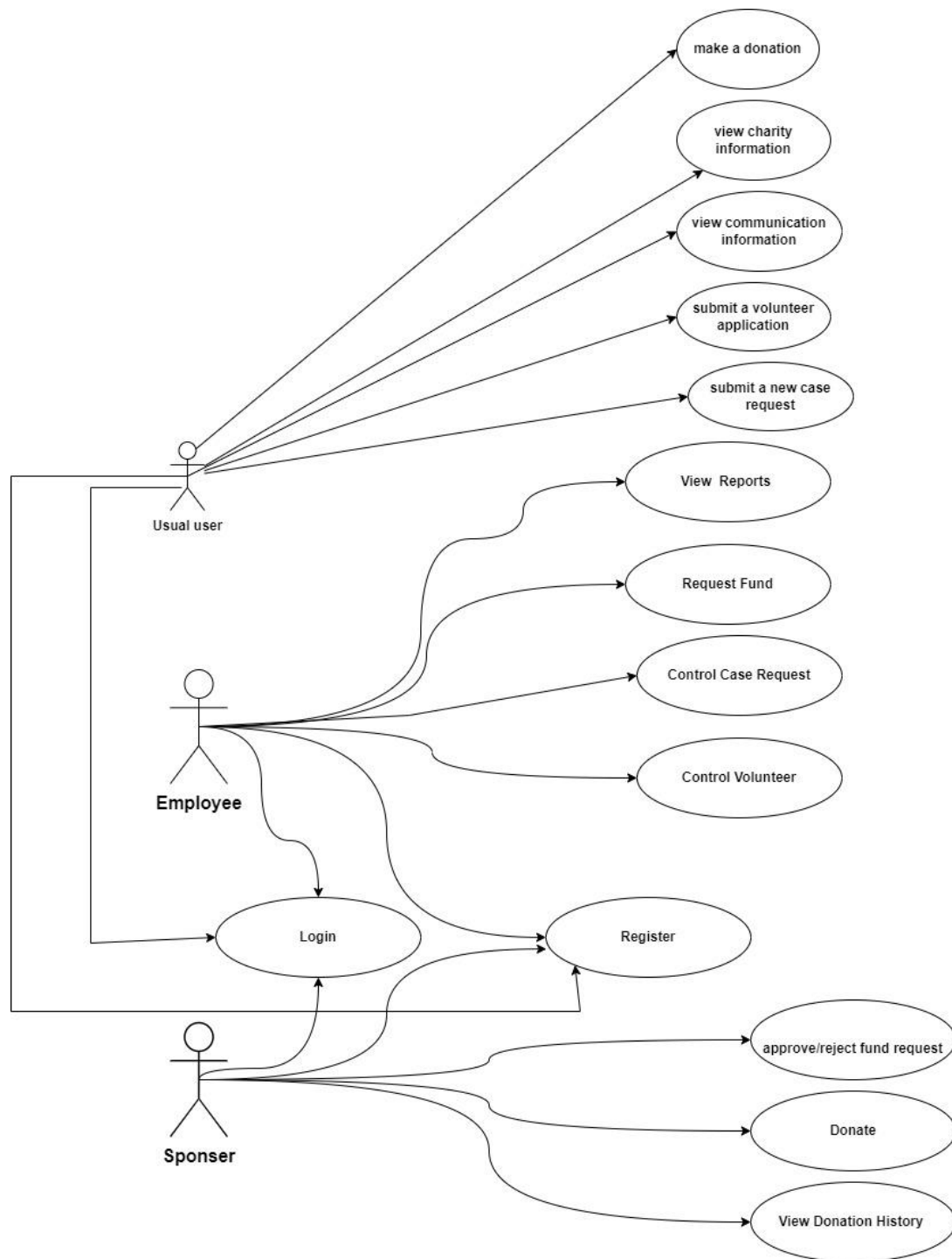
5- Portability Requirement

- The system shall ability to run on different platforms and operating systems without any change in its behavior and performance.

6- Maintainability Requirement

- The system shall be able to recover from failures within 10 minutes.
- The system shall be easy to maintain and update, with clear documentation and guidelines for developers and system administrators.
- The system shall follow best practices for software development, including version control, code review, and testing.

Use Case:



Sequence Diagram:

