German University in Cairo
Media Engineering and Technology
Assoc. Prof. Mervat Abuelkheir
Eng. Nada Ibrahim Eng. Hager Khaled
Eng. Nada Labib Eng. Kamilia Hosny Eng. Marina Nader

---

Software Engineering, Spring 2024

PROJECT DESCRIPTION

# Non-Monetary Donation Application for NGO

Repository Creation Deadline: 3 March, 2024 11:59pm

Milestone 1: Grade 10% - Deadline 13 April, 2024 11:59pm

Milestone 2: Grade 30% - Deadline 6 May, 2024 11:59pm

---

Please read the following instructions carefully:

- Any case of **plagiarism or cheating** in Milestone 1 or Milestone 2, will result in a zero.
- It is **YOUR responsibility** to ensure that you have:
  - Read and understood **everything** in the project description (this document).
  - Created a GitHub repository.
  - Submitted before the deadline.
  - Submitted the correct file for Milestone 1 on your GitHub repository.
  - Made commits for Milestone 2 on your GitHub repository.

.

# 1 Theme

You need to build a non-monetary donation application for an NGO (non-government organization) that will help both the donors and anyone who is requesting a donation to reach each other. This application will facilitate communication between both of them and will be able to provide donations **that do not include money** such as clothes, medical supplies, school supplies, furniture, toys, etc.. to those that need them.

The project includes 2 milestones:

- Milestone 1: requirements engineering (web application and mobile application)
- Milestone 2: front-end design and development (web application only)

# 2 Case Description

Refer to the following case description to understand how the web application/ mobile application should work from a donor's point of view:

1. You decide you want to donate some clothes to a family in need so you open the web application. You browse all the different cases that are available with their needs and urgency (including ages, sizes, etc..).
2. You find a case that needs the clothes you have available to donate so you send a request to donate the clothes.
3. You find that you have some electrical appliances you want to donate to an organization (may also include Mosques and Churches) that will be able to give the appliances to those who need them. You look through the available organizations that can accept the donation and will locate those that need them on your behalf.
4. You find an organization that is accepting electrical appliances as donations and you send a request to donate the appliances.
5. You are then informed that a delivery person(s) will drop by to pick up the items you are donating.
6. You track the item(s) until they are delivered to the organization/family/ individual in need.

# 3 Overview

In every single paragraph presented earlier, there is a pain point that faces a donor when trying to donate. **These are only SOME of the pain points that might face a donor**. You need to construct a web and mobile application to help donors, donation receivers, and other stakeholders with their own pain points, all while covering **at least** the following modules with their corresponding sub-modules:

1. **Health sector donations:**
    - Blood donations (through finding the nearest blood drive or based on demand in hospitals)
    - Medication and medical supplies
    - Pro-bono doctor visits/ appointments
2. **School supplies donations**:
    - School books
    - School supplies
    - Pro-bono teaching
3. **Refugee and people living below the poverty line needs**:
    - Apartments/ living quarters (free or low rent)
    - Clothes, shoes, bags, etc..
    - Kitchen supplies
    - Food and groceries (includes fresh food)
    - Electrical appliances
4. **Orphanage donations**:
    - Clothes, shoes, bags, etc..
    - Toys
    - School supplies
    - Luxury foods (cakes, sweets, etc..)
5. **Pickup/ delivery of donations** (similar to Uber):
    - Pickup of donated items from donor
    - Delivery to individual/ hospital/ orphanage/ organization

## 3.1 Product Managers

For **BOTH** Milestones 1 and 2, you can ask the **Product Manager** assigned to your team (according to the **Teams** file on the CMS) questions to gather information alongside your own research. Stick to your Product Manager to avoid receiving conflicting responses throughout your project.

### 3.2 Milestone 1

Milestone 1 will focus on **Requirements Engineering**, where you are required to write a set of complete and consistent **functional and non-functional requirements** for a **web application and mobile application** based on the modules and sub-modules listed previously. The requirements should be documented, actionable, measurable, testable, traceable, related to identified software needs, and defined to a level of detail sufficient for system design.

### 3.3 Milestone 2

Milestone 3 will act as the prototype of your system, which you will be presenting to your Product Manager. Your prototype will be for **your web application only** and will be made up of front-end design using HTML, CSS and JavaScript (there will be **no database nor back-end programming**).

You are free to use **additional technologies** (such as but not limited to React and Angular) as long as your entire system is presented properly. Failure to design parts of your system due to choosing complex front-end design technologies will result in losing grades. It is also important to understand that UI/UX evaluation is purely subjective and based on the opinion of your Product Manager.

## 4 Objectives

### 4.1 Milestone 1

- Learn how to analyze a problem scenario.
- Learn the process of requirements engineering and deriving requirements' specifications.
- Learn to research the whole scope of a given task using any relevant content. For example, textbook(s), Doctors, TAs, and online resources (like Product Backlog Example, etc.).

### 4.2 Milestone 2

- Learn how to use HTML, CSS and JavaScript to create a simple FE design.
- Collaborate as a team on the same repository on GitHub.
- Follow and implement the UI/UX rules taken in the lectures and tutorials.
- Follow the guidance for the FE design presented by the Product Manager assigned.

# 5 Deliverables

## 5.1 GitHub Repository

Each team is required to create a GitHub repository (repo) to use for this project to submit **both Milestone 1 and Milestone 2**.

- Each team leader should create a GitHub repository named after their team's name, with each word in the team name separated by a hyphen/ dash (-). For example, if your team name is "This is a random team", your GitHub repository should be called "this-is-a-random-team".
- Each team leader **must** grant their **assigned Product Manager according to the Teams file on the CMS access as Admin** to this repo.
- Each team leader **must** grant his/her other team members access to the repo.

**ALL SUBMISSIONS FOR MS1 AND MS2 WILL BE ON THIS REPO.**
Since the team leaders have complete control over the creation of the repos, no excuses for not having a repository will be accepted.

## 5.2 Milestone 1

In Milestone 1, you are required to submit a filled Excel sheet of well-written **user stories** that covers the following:

1. All stakeholders of the system.
2. All functional requirements as user stories.
3. All Non-Functional requirements of the system. For each one you must highlight if it maps directly to other **functional requirements**. In case it does, then point these functional requirements out. Otherwise, provide criteria on how to verify/measure it.
4. Open-ended questions. You are required to find answers to all the open-ended questions that will arise when thinking about the intricacies of the project. **Your answers must be reflected in either functional or non-functional requirements**. Some of those questions are as follows:
    - How will the web app facilitate the connection between donors and donation receivers?
    - How can donors and donation receivers communicate?
    - How are these requests collected without violating the privacy of the people receiving the donations?
    - How can the system be integrated with charity organizations that are already present and how can the system facilitate their work and needs?

- How will the requests be gathered?
- How will the cases be validated?
- How will the application assign the drivers to pick up the donated items?
- How will drivers be able to collect the donated items to transport them?
- How will the drivers picking up the donations know which items need to be picked up first?
- Which requirements will be on the mobile application, which will be on the web application and which will be on both?

5. Other questions might arise and it will be your task to find answers to them. The main objective is to have a functioning system that solves the problems that currently face the stakeholders and **will not create additional issues**.

6. Too many similarities between requirement submissions from different teams will be considered a cheating case and will result in a ZERO for the entire Milestone.

7. **The requirements sheet must follow the specification outlined in the Excel sheet named "RequirementsTemplate"** and must be submitted on your team's GitHub repository. **For Milestone 1, it does not matter who uploads the Excel sheet onto the GitHub repository; we will not be checking commits in Milestone 1**

**ANY SUBMISSIONS OF MS1 BY EMAIL WILL NOT BE ACCEPT. ALL MS1 SUBMISSIONS MUST BE ON THE REPO CREATED.**

Also note that in the template, there are examples of functional and non-functional requirements to show you the format of how to write requirements. These examples are not related to your project.

## 5.3 Milestone 2

In Milestone 2, you are required to design and implement ONLY the front-end of the Online Banking Website using HTML, CSS and JavaScript. You may use dummy data to represent any data that would otherwise be retrieved from a database.

1. You must make sure your entire project is on the GitHub repositories created for your teams, and each team member **MUST** have commits on the repository.

   **TEAM MEMBERS WITH NO COMMITS AT ALL WILL RECEIVE A ZERO FOR ALL OF MILESTONE 2**.

2. Too many similarities between FE designs will be considered a cheating case and will result in a ZERO for the entirety of Milestone 2.

3. You must not use an existing charity's/ oragnization's logo, you are designing your own system so you should create your own logo. You can use any tool/website for the creation of your logo (there are a lot of free online tools).

# 6 Grading Criteria

## 6.1 Milestone 1

The grading criteria of Milestone 1 will depend on the following points and any unfulfilled requirement will result in a grade deduction:

- You must have a **minimum of 7 well-defined** non-functional requirements. These non-functional requirements must be suitable for the system at hand.
- You must have **exactly 150 unique user stories**. If you write more than 150 requirements, we will only look at the **first** 150 requirements in your submission. Any requirements after requirement number 150 **will be ignored**. Therefore, be sure not to have any duplicated requirements.
- Each stakeholder must have a **minimum of 10** user stories.
- Each module must have a **minimum of 20** user stories.
- Your requirements should cover **all** the specifications communicated by your assigned Product Manager.
- For each user story to be considered valid and counted towards your requirements, it has to follow the following rules:
  - Clear
  - Consistent and coherent with other requirements
  - Implementable, testable, and deployable
  - The CRUD (create, read, update, and delete) of any business entity will be considered as **only one** user story.
  - User stories with the same action and intent but different stakeholders should be written as "As a stakeholder1/stakeholder2/stakeholder3, I should be able to [action] so that I can [intent]". This will be considered **one** requirement of the 150 requirements **but will count towards each of the stakeholders mentioned**, i.e. 1 of the 10 requirements for stakeholder1, 1 of the 10 requirements for stakeholder2, and 1 of the 10 requirements for stakeholder3.

The grading of this assignment will be broken down alongside the previously mentioned specifications as follows:

- Accuracy (clarity, precision and completeness of the specification)
- Presentation (organization and consistency of the document)
- Composition (format, spelling and usage of English)

## 6.2 Milestone 2

- You must cover all user journeys that will be posted after Milestone 1 submission with their sub-processes.
- Your system should not be just a mere copy of other existing systems.
- Your system's UI should be self-contained. Meaning, that all functionalities needed to be fulfilled should be doable through the system's UI.
- The system's UI must be intuitive and easy to use, as it should follow the basic UX/UI guidelines outlined in lectures and tutorials. This means there should be no hurdles while the Product Manager is navigating through the user journeys of your software.
- Your system's UI must be coherent and consistent (even with your logo), and it must follow a color palette. For color palette examples you can check https://www.canva.com/colors/color-palette-generator among other online resources.
- The grading of your system will **purely subjective based on the opinion and viewpoint of your assigned Product Manager** and will consider the degree of:
  1. Learnability.
  2. Visibility.
  3. Efficiency.
  4. Design for errors.
  5. Overall satisfaction of the different aspects of your design.
  6. Navigation from page to page and navigation reversibility.
  7. How close the UI matches the user-mental model.
  8. How the UI considers the user efficiency to carry out commonly occurring tasks.
  9. Feedback (or lack thereof) provided to users on certain actions.
  10. Responsiveness of the UI.
  11. Consistency of the IU elements.
  12. How professional the overall theme of the system looks.