

Software Engineering, Spring Term 2023

## Online Banking System

Milestone 1 - Grade 10% - Deadline 13 March 2023 11.59 pm

Milestone 2 - Grade 10% - Deadline 6 May 2023 11.59 pm

Milestone 3 - Grade 20% - Deadline 29 May 2023 11.59 pm

---

Please read the following instructions carefully:

- Any case of **plagiarism**, will result in a zero.
- Any case of **cheating**, will result in a zero.
- Working ahead is allowed but submitted a Milestone instead of another is not allowed and the missed Milestone will be given a zero.
- It is **YOUR responsibility** to ensure that you have:
  - Read and understood everything in the description
  - Submitted **ALL** the correct file(s) on your Team's GitHub repository.
  - Submitted before the deadline for each Milestone due.
  - Submitted the correct file(s).

Good luck! =D

# 1 Theme

The overarching theme of this project is to create an online banking system. Online banking systems are used on a daily basis by bank customers and have multiple modules Personal Financial Management and Bill Payment and Donations, which allow clients to carry out multiple functions without having to physically be at the bank.

## 2 Case Description

1. Consider a case where it is the end of the month and it is payday at work today and you expect your salary to be deposited into your bank account so you open the online banking system to check if you really did get paid.
2. You find that you did receive the money and you need to begin paying some of the bills that have been piling up that month. Your credit card balance is the first thing you want to pay off, in addition to transferring your housekeeper's and babysitter's monthly salaries to their bank accounts and donating your usual monthly amount to your favorite charity.
3. After paying for everything so far, you find that you do not have enough money to pay your gas, electrical, mobile phone and internet bills because you are severely underpaid. So you resort to paying these bills out of the savings you have.
4. You suddenly find out that the Egyptian Pound has dropped to half of its previous value in relation to all the other currencies in the world. You decide to put part of whatever is left of your (few) savings into a certificate which will last a year or two.
5. You then decide that you are in dire need of a car because Ubering everywhere on a daily basis has put a strain on your financial situation, so you apply for a loan.
6. After two weeks, your loan request is approved and you go and buy your car. You now have something extra to pay off every month - but hey, congratulations, you have a new car.
7. You find that with all your credit card use you should have gathered a good amount of points which are exchangeable for vouchers in certain stores, but for some reason you cannot see the points. You send a message requesting help and a couple of days later you receive a reply and you can now see your points. There are so many points so you go buy yourself a new pair of shoes.
8. You decide to take a look at your monthly spending and realize that someone somewhere has used your credit card to travel to Paris (a trip you can't even afford). You lodge a dispute with your bank stating that you never left the country and that your credit card number was stolen.

9. The bank acknowledges that your credit card has indeed been stolen and gives you the money back. You then cancel your old credit card and request the issuance of a new one.
10. This experience makes you with your bank would categorize your spending based on categories and notify you if your expenses approach a certain limit associated with that category.

## 3 Overview

### 3.1 Milestone 1

In every single paragraph presented earlier, there is a pain point that faces a user when trying to use an online banking system. **These are only SOME of the pain points that might face a user.** We would like to construct a software to help bank clients, bankers and other stakeholders with their own pain points.

Milestone 1 will focus on **Requirements Engineering**. Requirements engineering is critical to the success or failure of any software project. 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.

You are asked to define the requirements required to construct such a software covering **at least** the following modules with their corresponding main features:

- Users and Information Management System
  - Customer on-boarding process
  - Opening a new account
  - Client-bank communication and announcements
- Credit Cards and Loans
  - Credit card application
  - Credit card(s) payment and history
  - Credit card Theft Management
  - Loan application
  - Loan payment
- Personal Financial Management
  - Account(s) details
  - Account(s) history and activity
  - Bank Transfers
  - Cheque submission

- Bill Payment and Donations
  - Pay bills through bank and 3rd party applications
  - Donate to charities

You need to research how banks work and understand the different options offered to clients in banks. You can ask questions (in lectures, tutorials and on Piazza) to gather information alongside your own research. Course staff will act individually but they represent one client, so there may be conflicting requirements made by different individuals. It will be your task to come up with a **complete and consistent** set of requirements.

**BONUS:** Make the system accessible to blind people. Consider what is required by all stakeholders for this to happen.

## 3.2 Milestone 2

Milestone 2 will focus on the design part of the application, where you will apply some of the concepts you learned about Software Design as well use different Design Patterns in your class diagrams.

## 3.3 Milestone 3

Milestone 3 will act as the prototype of your system, which you will be presenting to your Product Manager. Your prototype will be made up of front-end design using HTML, CSS and JavaScript.

You are free to use additional technologies 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 RE and deriving requirements' specifications.
- Learn to research the whole scope of a given task using any relevant content. For example, text book(s), Doctors, TAs, and online resources (like [Product Backlog Example](#), etc.).

## 4.2 Milestone 2

- Learn how to translate a set of requirements into class diagrams and ERDs.
- Understand the different uses of Design Patterns in a system.

## 4.3 Milestone 3

- Learn how to use HTML, CSS and JavaScript to create a simple FE design.
- Collaborate together as a team on the same repository.
- Follow and implement the UI/UX rules taken in the lectures and tutorials.

# 5 Deliverables

## 5.1 Milestone 1

In Milestone 1, you are required to submit a sheet consisting 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 a criteria on how to verify/measure it.
4. You must specify which of the functional and non-functional requirements will be on the mobile app, on the online banking website or on both.
5. 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 loans be approved?
  - How can a new client open a bank account?
  - Who gets access to clients' data?
  - How can the client and the bank communicate?
  - How can any user report issues with the system?
6. 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**.

**N.B. The requirements sheet must follow the specification outlined in the excel sheet named “RequirementsTemplateM1”.**

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

## 5.2 Milestone 2

In Milestone 2, you are required to submit the following **FOUR** deliverables for the **online banking website**:

- A components diagram that shows how the different modules in your system will interact.
- The class diagram of your entire online banking system.
- The entity relationship diagram (ERD) of the database that will be used in your online banking system.
- The schema of the database that will be used in your online banking system.

## 5.3 Milestone 3

In Milestone 3, 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.

- You **must** make sure your entire project is on the GitHub repositories created for your teams.
- Too many similarities between FE designs will be considered a cheating case and will result in a **ZERO** for the entire Milestone.
- You **must not** use an existing bank’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).
- You **must** submit a video (via Google drive or YouTube link) with voice-over, explaining the project that is **not less than 5 minutes and not more than 10 minutes**. It should be a demo of your application showcasing the different features, the implemented journeys and commenting on the experience whenever necessary.

## 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 grade deduction:

- You must have a **minimum of 10 well-defined** non-functional requirements. These non-functional requirements must be suitable for the system at hand.
  - You must have a **minimum of 100 unique** user stories where each stakeholder must have a **minimum of 10** user stories and each module must have a **minimum of 20** user stories.
  - In order for each user story to be considered valid it has to be:
    - Clear
    - Consistent and coherent with other requirements
    - Implementable, testable, and deployable
- N.B.** The CRUD (create, read, update, and delete) of any business entity will be considered as **only one** user story.
- Your requirements should cover **all** the specifications communicated by the staff.

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

The grading of Milestone 2 will be based on the following criteria:

- You must submit a PDF of your components diagram.
- You must a PDF of your class diagram for your entire system.
- You must submit a PDF of the ERD of your entire system.
- You must submit a PDF of the database schema.

All diagrams must be clear, specific, meaningful and realistic.

### 6.3 Milestone 3

- You **must** cover all user journeys outlined with their sub-processes.
- Your system's UI should be self-contained. Meaning, 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 or drop outs that will face the examiner when he/she is navigating through the user journeys of your product.
- The navigation from page to page should be intuitive and reversible.
- Your system's UI **must** be coherent and consistent (even with your logo), 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 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. How close the UI matches the user-mental model
  7. How the UI considers the user efficiency to carry out commonly occurring tasks
  8. The feedback (or lack thereof) provided to users on certain actions.
- The "Home Page" as well as the overall theme of the system **must** be professional, based on your own personal research and not just a mere copy of other existing systems.