جامعة الأهرام الكندية
AHRAM CANADIAN UNIVERSITY

Faculty of Computer Science and Information Technology

Your Project Name

Phase 1: Software Engineering Project

Submitted To

Dr. Reham Adel

TA. Your TA Name

Submitted By

Student ID	Student Name	Section Number

Contact Email: Write your Team Leader Email

Instructions [To be removed]:

- IMPORTANT. Rename this document to
  - **SWE303-TAName-LeaderID-Phase1.pdf**
- Remove the following notes and any red notes
- This document is the template document for your Phase 1.

1. Introduction

Write your project description in more than 10 lines.

1.1 Purpose

- Summarize the purpose of the software

1.2 Scope

- Any software could have too many components / Major features .. but we should implement specific things...this is the scope
- In simple points, what is the software scope (focus on components / Major features, not tiny things)

1.3 Definitions, acronyms, and abbreviations

- In a table, list all needed ones. Consider the audience doesn't know
  - Think as following: Document has abbreviation ATM..IFF audience doesn't know it, let's clarify it.

2. Requirements

2.1 Functional Requirements

- Functional requirements describe what the system should do
  - Example, an ATM allows you to enter Card, enter user name password and withdraw a money
- List all the system requirements
- Each requirement should be clearly described, such that it can be understood without the presence of the one who wrote it.

2.2 Non Functional Requirements

- Non-functional requirements describe how the system works
  - Example, Withdraw operation will be done within 20 second. Network is using secured protocols. System allows up to 30,000 withdrawal per minute.
- Think about the operation / system quality
- There are too many non-functional requirements, pick the suitable ones for your system.
  - For each one, write the details
  - Be realistic
- Non-functional requirements must be VERIFIABLE, i.e., MEASURABLE.
- Some Types as just examples: Usability, Reliability, Performance, Security, Scalability, Portability, Maintainability

3. Used Technologies

- **Jira**: Is a website to make a board, put all tasks on it and assign each task to each member. We used it to manage our project and tasks upon the team members by organizing the tasks, creating sprints, and keeping track of our progress.

  Provide Screenshot about your project timeline on Jira including all sprints with their times.

- **GitHub repository link**: This repository is made to make it easy for our team to communicate with each other. Every member will upload his copy of code with his modification to the code, so every member will always know the updates. This repository will help avoiding and reducing the reasons of errors.

  Provide your repository public link

- **Frontend technology:** ____ (Select **Web application or Desktop application**).
  - Write programing languages, frameworks, tools, and IDEs you will use

- **Backend technology:**
  - Write programing languages, frameworks, tools, and IDEs you will use

4. System Models

Use tools to draw diagrams

4.1 Use Case Model

- Draw the use case model expressing the systems, actors, use cases, and relationships
- In case on diagram is so complex, divide it to several ones of reasonable size.

Insert Use Case diagram Image Here

- **Use Case Description Tables**
  - Using below table template, for each requirement write a use case table
  - If one requirement is so big, you could divide it to more than table

Use Case Number:		
Use Case Name:		
Actors:		
Overview:		
Related use cases:		
Event(Stimulus):	**User Action**	**System Action**
	1- User Enter Card and Password.	
		2- System Verify user data
	and so on	
Exceptions:		
Comments:		

4.2 Class diagrams

- You should provide your class diagram. In case on diagram is so complex, divide it to several ones of reasonable size.
- Put Relationships between classes and the types of the relationships.
- Put multiplicity.
- Put relationship name (example, faculty "offer" course).
- Put attributes in the classes.
- Put functions & Put parameters.
- Put data types of each attribute and the parameters.

Insert Class diagram Image Here

- **Use Classes Description Tables**

Class ID	Class Name	Description

4.3 Sequence diagrams

- List Sequence diagrams for all requirements. Provide each Sequence an ID.
- Not necessary diagram per requirement. May be better to merge more than a flow. Typically, 1 sequence diagram per your use case table.
- Make sure that each object in the sequence diagram has a corresponding class in the class description table above. If not, it will be REJECTED.
- Put actual function calls with proper parameters and return types corresponding to class diagrams.
    - Please always specify the parameters in the call, matching the class diagram.

Insert Sequence diagrams Images Here

4.4 Physical Entity-Relationship Diagram

- Provide the ERD Diagram
- Convert all entity objects of the class diagram and their relationships into ERD
- Put Relationships between entities.
- Put relationship name (example, faculty "offer" course).
- Put attributes of the entities (primary key, Foreign key,…).
- Put cardinality (1:1, 1:m, m:n,..).

Insert ERD diagram Image Here

Ownership Report

- Remove the following notes and any red notes
- For every item in this document, write the owners.
    - If someone is owner of something, s/he understands it 100.%

Student ID	Student Name	What the student done

Policy Regarding Plagiarism [To be removed]:

- Students have collective ownership and responsibility of their project.
- Any violation of academic honesty will have severe consequences and punishment for ALL team members.