

 <p>Helwan University Faculty of Computers & Information</p>	<p>CS 352 Group Work Project 2016-17</p>
<p>Module Title Software Engineering II</p>	
<p>Module Leader Dr. Amr Ghoneim</p>	<p>Semester Two (Spring)</p>
<p>Assessment Weight</p> <p><u>Tentatively</u> 25 % of the total course mark</p>	<p>Due Dates (<i>week starting on</i>):</p> <ul style="list-style-type: none"> - Saturday March 25th (week 7) - Saturday April 8th (week 9) - Saturday April 29th (week 12)

Instructions to Students:

1. This is a group work project. Each group consists of FIVE to SIX students (group members must be approved by the module leader through registration). Each group must develop **the idea assigned to them** using object oriented technologies.
2. Submission: Submission is done during the Weekly Labs and the Scheduled Final Discussion only.
3. Assessment: Assessment will be on the reports and code submitted, in addition to scheduled discussions with team members. All the team members must contribute to all the phases, and the role of each member must be clearly stated in each report.

4. Feedback: Feedback, for each group, will be given during the submission of each phase. Further details could be provided through discussion during office hours with the teaching assistants.
5. You can only submit your own work. Any student suspected of plagiarism will be subject to the procedures set out in by the Faculty (including failing the course entirely).

1. Project Objectives:

The objectives of this project can be summarized in the following:

1. Using UML diagrams in developing an object oriented application.
2. Applying design patterns.
3. Testing.

2. Project description:

Every team must follow the project description provided to them by the TA.

3. Development stages and Submission deadlines:

To develop your application you will go through the following stages:

1. **Requirements analysis**: Consider as more functionalities as possible in your application during this phase.

Deliverables (at least):

- a. Use case diagram.
- b. Use cases in detailed format.
- c. System sequence diagrams.
- d. Domain Model.

Deadline: The week starting on Saturday March 25th (week 7)

2. **Detailed design:** This stage specifies a design that will be transformed into programming code. You must apply design patterns in this phase and you should address the GUI and Database design.

Deliverables (at least):

- a. Detailed design class diagram.
- b. State clearly the design patterns you have applied.
- c. Interaction diagrams.
- d. GUI.
- e. Relational Database schema.

Deadline: The week starting on Saturday April 8th (week 9)

3. **Final report accompanied by Implementation, and presentation**

Deliverables:

- a. CD containing the code.
- b. Report including:
 - i. Refined version of class diagram, or any other diagrams
 - ii. Testing report

Deadline: The week starting on Saturday April 29th (week 12)

For each stage one report (softcopy) per group should be submitted to the assigned TA by the specified deadline for each Report. A hardcopy version from each report should be submitted during the discussions.

Late reports will be accepted after the specified deadline with penalty applied for each late day. Extensions will be granted only on the basis of material evidence like medical certificates. Remember that late report leads to late feedback.

4. Assessment criteria:

The Project as a whole will be assessed based on the following criteria:

1. The complexity determined by amount of functionalities covered by your project.
2. The quality of your documentation.
3. The correctness of your analysis and design diagrams.
4. Design Quality: using patterns, levels of abstractions, etc.
5. Implementation correctness.

Every member in a group will be assessed upon their role during each stage. All members must share in developing all the stages. Each group must identify clearly in the report the role of each member during each development stage (students in a group may not be given the same mark)

5. Marking Schema:

The project will be marked out of 100. The following is a **tentative** marking breakdown

Requirements analysis:	22
- Use Case diagram	4

- Detailed Use Cases	8
- SSD	5
- Domain Model	5
Design:	30
- Detailed Design Class Diagram	15
- Interaction diagrams	15
Coding and Testing:	48
- Database	10
- GUI	8
- Application Logic (OOP)	20
- Testing	10
Academic Integrity:	
Helwan University's policies on academic integrity will be enforced on students who violate University standards of academic integrity.	
Examples of behaviour that is not allowed are:	
<ul style="list-style-type: none"> • Copying all or part of someone else's work and submitting it as your own; • Giving another student in the class a copy of your work; and • Copying parts from the internet, text books, etc. 	
If you have any questions concerning what is allowed, please don't hesitate to discuss with me.	