**Faculty of Information Technology and Computer Sciences**

**Department of Information Technology**

**MIS 499: Graduation Projects**

**Semester: Spring 2020/2021**

**Graduation Project Phase-2:**

**System Design Assessment Criteria (30%)**

**Due Date Thursday April 22nd, 2021**

**Activities:**

* It is always recommended to start with the logical design, and then progressing to the (final) Physical design.
* Develop Module Design
* Develop Database Design
* Develop GUI Design
  + Input & Output Design (or Forms and Report Design)
  + Dialogue Design (Top-down structure of all system screens for user interaction with the software system)

**Required Document:**

Each project team must submit a **Design-Phase Document** contains the following components:

* **Cover Page** (one page): YU Logo & Name, Faculty and Department Name, Project Title, Team Leader & Members Names and Supervisor Name, Academic Year and Semester (We can provide a “TEMPLATE” for the students).

**Behavioral Modeling (Processing):**

* **Use-Case Diagrams:** Showing all activities along with the corresponding actors in the system. Each use-case is supported with fully-developed description……..………[15 points]

**Fully-Developed Description (in table format) of the use-case includes:**

* + - **Use-Case Name**
    - **Triggering Event**
    - **Use-Case Brief Description**
    - **Actor Name(s)**
    - **Pre-condition to the Use-Case**
    - **Post-condition of the Use-Case**
    - **Normal Steps of Processing**
    - **Exceptions or Possible Errors Handling Steps**
* **Data Flow Diagrams (DFDs)**: modeling the functionality aspect of the system.

These include:

* + **Context Diagram**: Showing the interaction of the system with the

corresponding sources and sinks. ………………………………………..……..……[5 points]

* + **Level-0 DFD**: Showing the major functions in the system...……….……[20 points]

One **Child-DFD** diagram is needed for each complex process on the Level-0 DFD.

If all processes on the Level-0 DFD are primitive then no need for Child DFD.

* **In case of team members are following Object-Oriented Approach, they can choose to produce one of the following diagrams instead of the Level-0 DFD. Either Sequence diagram or State-Machine diagram (depends on the nature of the undertaken system):**
  + **Sequence Diagram:** Showing all object and their interactions in chronological order. The actors, the objects, the swim lanes and the messages.………[15 points]
  + **State-Machine Diagram**: It depends on the nature of the system; if it does

need Showing stats and events, then you need to develop a State-machine

Diagram at different levels of details ...…..………………………..………….……[15 points]

**Data Modeling:**

* **Database Design:**
  + **Table/Entity**: Each table is illustrated as a separate structure, Table-Name, Primary-Key and Attributes’ Data types and Sizes. ..………………………[5 points]
  + **Database Entity Relational Diagram (ERD)**: Showing the relationships

between entities, both degree and cardinality. ……………….……………[10 points]

* + **Database Schemas Diagram:** Showing the arrangement of relation states.

Links of primary-keys and foreign-keys among table-structures………[5 points]

* **In case of team members are following Object-Oriented Approach, they can choose to produce the following diagrams instead of the ERD & Schemas diagrams:**
  + **Class Symbol**: Each class should be shown separately; including Class-Name,

Class-Data Section and Class-Methods Section.……………………………[5 points]

* + **Classes Diagram**: Showing the nature of relationships between classes (Inheritance, Aggregation and Association).…………………….……………[15 points]

**Graphical-Interface (GUI) Design (Screens Design):**

* + **Forms & Reports-Design**: Select only major Formsand couple of

Reportsand design them accordingly.……..………………………..……………[15 points]

* + **Error-Message:** Select couple of error messages only.…………………….[5 points]
  + **Dialogue-Design**: Showing the top-down structure of all major screens

of your software and how they connect to one another.……..…………[10 points]

**🡺 Make sure to apply all GUI design guidelines on all forms & reports. These include:**

* + - **Meaningful titles**
    - **Alignment of fields and labels**
    - **Grouping related data in boxes with meaningful headers**
    - **Meaningful Labeled data fields**
    - **Using appropriate “units” for currency, weight, distance, etc.**
    - **Margins and spaces**
    - **Highlighting (colors, font-size, etc.)**
    - **Smooth Traverse/navigation between fields and screens**
    - **Provide all necessary control buttons**
    - **Error messages must be clear, and direct user to corrections**
    - **Report info: Title, date, column-headers, spaces between columns & Page#.**
* **Data Dictionary:** In case your system involves special terminology**.** …….……………[5 points]

***🡺 Note: Add up team’s grade out of 90 and re-calculate it (divide by 3) out of 30***