CS 432 – Databases

Course Project (Task for Module – 2)

Due Date: 1st April 2025 - 11:59 PM

Task Description

For this module, you are required to create Entity-Relationship (ER) and Unified Modeling Language (UML) diagrams that accurately reflect your overall design of the mini-project proposal (course project). Using the ER and UML concepts, you can use the Module 1 task to reimage the overall project further. The goal is to visualize the structure and relationships of the database entities effectively and create a well-detailed report containing the UML diagrams, ER diagram, details on defined relationships/multiplicity, and the roles of each team member.

Prerequisites

- Online UML and ER diagram tools such as **Visio, draw.io, Lucidchart, or any similar tool as discussed in Lecture 17.**
- A clear understanding of the **database schema implemented in Module 1**.
- To learn how to create UML diagrams, refer to lecture 17 notes or book chapter 10 (Ramez Elmsari and Shamkant B. Navathe. Database Systems 6th Edition, Pearson, 2015.)
- To learn how to create an ER diagram, refer to lecture 15 notes or book chapter 7 (Ramez Elmsari and Shamkant B. Navathe. Database Systems 6th Edition, Pearson, 2015.) and chapter 6: Database Design Using the E-R Model, (Abraham Silberschatz, Henry Korth, and S. Sudarshan. Database System Concepts. 7th Edition)

---

1. Creating a UML Diagram

To systematically design the database, start with a UML class diagram before transitioning to the ER diagram.

- Identify classes (entities), attributes, and methods (if applicable).
- Clearly define relationships between entities (associations, generalization, aggregation, or composition).
- Represent multiplicity (1:1, 1:M, M:M) properly.
- Ensure that the diagram adheres to UML conventions.

*NOTE: Present as many UML diagrams as possible for your project, and refer to lecture notes for more information on identifying the types of UML diagrams.

Why UML First?

A UML class diagram provides a high-level conceptual design before transitioning to an ER diagram, ensuring a well-structured database schema. It helps to identify objects and their relationships before mapping them into database entities.

---

2. Creating an ER Diagram

Based on your UML diagrams, convert it into an ER diagram, ensuring adherence to database schema constraints from Module 1.

Design the ER Diagram:

- Represent entities as rectangles and relationships as diamonds.
- Include primary keys (underlined) and foreign keys.
- Clearly define one-to-one, one-to-many, or many-to-many relationships.
- Use appropriate notation for clarity and accuracy.
- Ensure that the ER diagram correctly maps to the SQL database schema from Module 1.

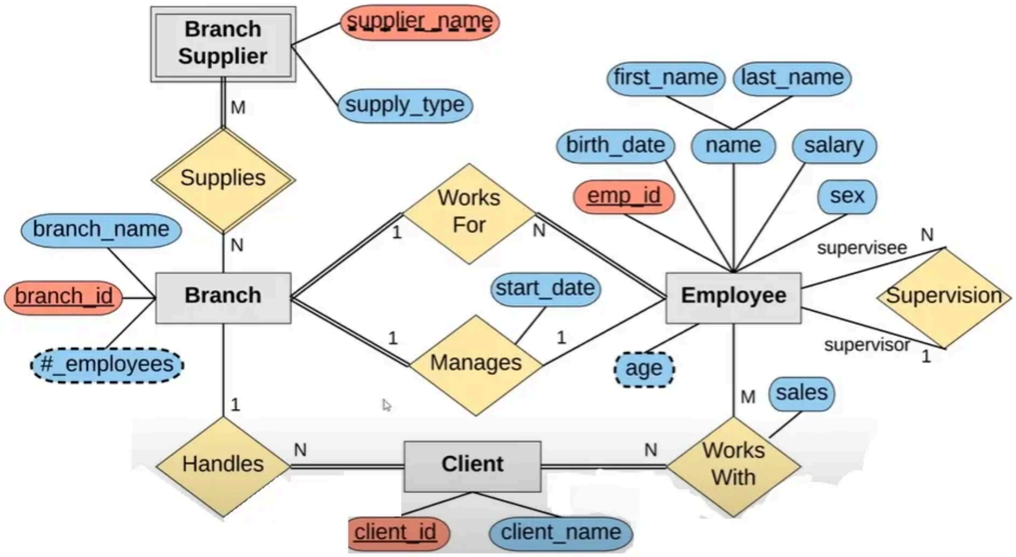
Example Workflow

- Entities: Student, Course, Enrollment
- Attributes:
  - Student (ID, Name, Age)
  - Course (CourseID, Title, Credits)
  - Enrollment (StudentID, CourseID, Grade)
- Relationships:
  - Student enrolls in Course (one-to-many)
  - Course has many Students (many-to-many)

How ER is Derived from UML

- Convert classes in UML to entities in ER.
- Convert associations into relationships.
- Define keys (primary & foreign) based on UML attributes.
- Adjust multiplicity (e.g., one-to-many in UML to a foreign key in ER).

Example ER Diagram (ER Diagram to Create Database Schema)



3. Submission Guidelines

Submit a single report in PDF format. Only one member from each team should make the submission. Your report should contain the following elements:

- UML Diagram (You should draw as many diagrams as possible for your project)
- ER Diagram
- Explanation of the UML diagrams, how entities and relationships are derived.
- Justification for the transition from UML to ER.
- Explanation and examples of why a relationship is one-to-one, one-to-many, or many-to-many in ER diagrams with examples.
- Any additional constraints or considerations done from your end for database implementation.
- Name and contribution of each team member.

4. Evaluation Criteria

Criterion	**Description**
UML Diagrams Completeness	Proper use of UML conventions and representation of classes/relationships as per diagram types.
ER Diagram Accuracy	Correct mapping from UML to ER, proper notation, and relationships.

Alignment with Module 1 Schema	The ER diagram must accurately reflect the SQL database schema implemented in Module 1.
Clarity of Explanation	Coherence and depth of the provided explanation.
Examples of Relationships	Proper justification with practical examples for relationship types (1:1, 1:M, M:M)
Team Contribution	Clear definition of roles and responsibilities of each member.

Note:

- If you have any doubts, refer to course resources or contact the TAs.
- Submissions made later then due date/time will result in a 10% mark deduction each hour.
- **Ensure that your diagrams strictly adhere to schema constraints from Module 1. The evaluation criteria will be strictly followed.**