**DBMS KEY Milestone 2**



**Spring 2025**

**CSE 403L: DATA BASED MANAGEMENT SYSTEM**

**Group members:**

**Muhammad Maaz(22PWCSE2204)**

**Muhammad Ibad Khan (22PWCSE2170)**

**Muhammad Talha Khan(22PWCSE2213)**

**Class Section: A**

“On my honor, as student of University of Engineering and Technology, I have neither given nor received unauthorized assistance on this academic work.”

**Submitted to:**

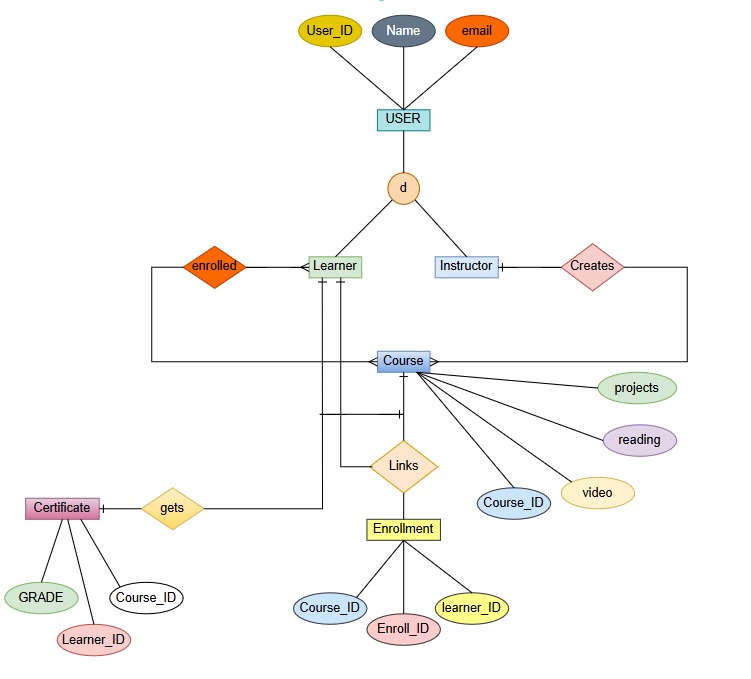
**Engr. Maam Sumayyea Salahudin**

**DATED: 7/8/2025**

**Department of Computer Systems Engineering**

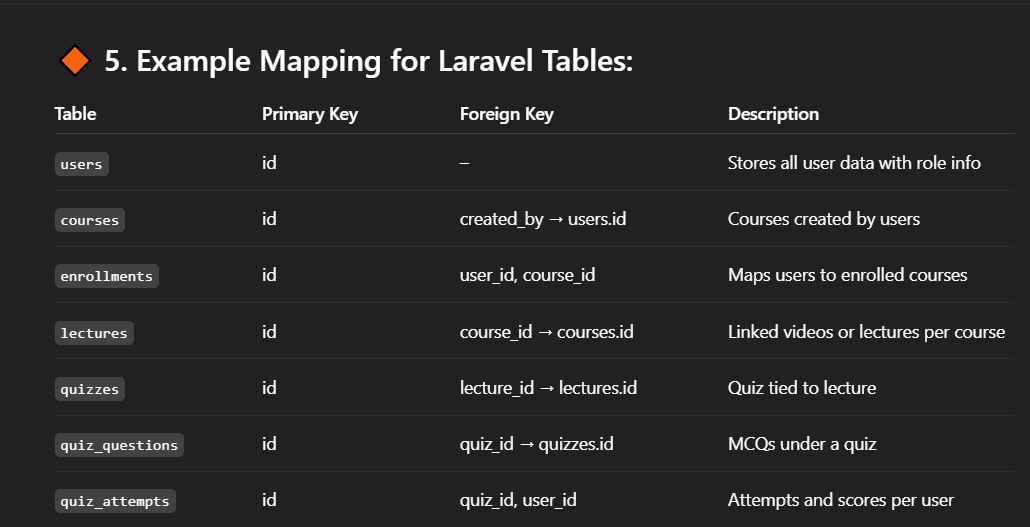
**University of Engineering and Technology, Peshawar**

1. **Converting Conceptual Schema (EER Diagram) to Relational Schema):**



**Relational Schema:**

**MAPPING FOR LARAVEL**

****

1. **USER:**

* **Attributes:**
  + User\_ID (Primary Key)
  + Name
  + Email

1. **Learner:**

* **Attributes:**
  + User\_ID (Primary Key)
* **Constraints:**
  + Foreign Key referencing USER(User\_ID)

1. **Instructor:**

* **Attributes:**
  + User\_ID (Primary Key)
* **Constraints:**
  + Foreign Key referencing USER(User\_ID)

1. **Course:**

* **Attributes:**
  + Course\_ID (Primary Key)
  + Projects
  + Reading
  + Video

1. **Enrollment:**

* **Attributes:**
  + Enroll\_ID (Primary Key)
  + Learner\_ID (Foreign Key referencing Learner(User\_ID))
  + Course\_ID (Foreign Key referencing Course(Course\_ID))

1. **) Certificate:**

* **Attributes:**
  + Course\_ID (Primary Key, Foreign Key referencing Course(Course\_ID))
  + Learner\_ID (Primary Key, Foreign Key referencing Learner(User\_ID))
  + Grade

**1st Normal Form:**

**USER**

(User\_ID PRIMARY KEY, Name, Email)

**Learner**

(User\_ID PRIMARY KEY REFERENCES USER(User\_ID))

**Instructor**

(User\_ID PRIMARY KEY REFERENCES USER(User\_ID))

**Course**

(Course\_ID PRIMARY KEY, Title, Description)

PRIMARY KEY (Course\_ID, Video\_ID))

**ENROLLMENT**

(Enroll\_ID PRIMARY KEY,

Learner\_ID REFERENCES Learner(User\_ID),

Course\_ID REFERENCES Course(Course\_ID),

Enrollment\_Date,

Status)

**CERTIFICATE**

(Cert\_ID PRIMARY KEY,

Enroll\_ID REFERENCES ENROLLMENT(Enroll\_ID),

Issue\_Date,

Expiry\_Date).

2nd Normal Form:

**USER**(User\_ID PRIMARY KEY, Name, Email, User\_Type)

**LEARNER(User\_ID PRIMARY KEY REFERENCES USER(User\_ID))**

**INSTRUCTOR**(User\_ID PRIMARY KEY REFERENCES USER(User\_ID))

**COURSE**(Course\_ID PRIMARY KEY, Title, Description, Instructor\_ID REFERENCES INSTRUCTOR(User\_ID))

**ENROLLMENT**(

Enroll\_ID PRIMARY KEY,

Learner\_ID REFERENCES LEARNER(User\_ID),

Course\_ID REFERENCES COURSE(Course\_ID),

Enrollment\_Date,

Status,

UNIQUE (Learner\_ID, Course\_ID) -- Prevent duplicate enrollments

)

**CERTIFICATE**(

Cert\_ID PRIMARY KEY,

Enroll\_ID REFERENCES ENROLLMENT(Enroll\_ID), -- Now depends on single enrollment

Issue\_Date,

Expiry\_Date)

**3rd Normal Form:**

USER(

User\_ID PRIMARY KEY, -- Unique identifier for all users

Name, -- User's full name

Email UNIQUE, -- User's email (must be unique)

User\_Type CHECK (User\_Type IN ('Learner', 'Instructor')) -- Role type

)

LEARNER(

User\_ID PRIMARY KEY -- Inherits from USER

REFERENCES USER(User\_ID) -- Foreign key to USER table

ON DELETE CASCADE, -- If user is deleted, learner record is too

-- Additional learner-specific attributes can be added here

)

INSTRUCTOR(

User\_ID PRIMARY KEY -- Inherits from USER

REFERENCES USER(User\_ID) -- Foreign key to USER table

ON DELETE CASCADE, -- If user is deleted, instructor record is too

-- Additional instructor-specific attributes can be added here

)

COURSE(

Course\_ID PRIMARY KEY, -- Unique course identifier

Title, -- Course title

Description, -- Course description

Instructor\_ID -- Who teaches this course

REFERENCES INSTRUCTOR(User\_ID) -- Foreign key to INSTRUCTOR

ON DELETE RESTRICT, -- Prevent deletion if courses exist

Creation\_Date,

Is\_Active BOOLEAN

)

ENROLLMENT(

Enroll\_ID PRIMARY KEY, -- Unique enrollment identifier

Learner\_ID -- Who is enrolling

REFERENCES LEARNER(User\_ID) -- Foreign key to LEARNER

ON DELETE CASCADE, -- Delete if learner is deleted

Course\_ID -- Which course they're enrolling in

REFERENCES COURSE(Course\_ID) -- Foreign key to COURSE

ON DELETE CASCADE, -- Delete if course is deleted

Enrollment\_Date, -- When enrollment occurred

Status CHECK (Status IN ('Active', 'Completed', 'Dropped')), -- Enrollment state

UNIQUE (Learner\_ID, Course\_ID) -- Prevent duplicate enrollments

)

CERTIFICATE(

Cert\_ID PRIMARY KEY, -- Unique certificate identifier

Enroll\_ID -- Which enrollment this certifies

REFERENCES ENROLLMENT(Enroll\_ID) -- Foreign key to ENROLLMENT

ON DELETE CASCADE, -- Delete if enrollment is deleted

Issue\_Date, -- When certificate was issued

Expiry\_Date, -- When certificate expires

Certificate\_URL -- Link to digital certificate

)

**References:**

* www.Deepseek.com
* [www.github.com](http://www.github.com)

THE END