

## **CS 5330 Project**

### **Fall 2025**

The goal of the group project is to implement a database solution to the following application, using mySQL as the backend.

#### **Application: Program Evaluation**

Each degree program in a university is constantly under review. To facilitate reviews, data need to be corrected, stored and analyzed. In this project, we want to develop a database to evaluate a program for one specific department.

##### ***Degrees, faculty and courses***

The basis of evaluation is a degree. Each degree has a name (may not be unique), and a level (BA, BS, MS, Ph.D., Cert). The combination of name and level will be unique among degrees.

Each degree has a set of courses assigned to it. Each course has a unique course number (a 2-4 letter department code + a four-digit number), and a unique name. Some courses for the degree are designated as core courses (there is no limit to the number of core courses – a degree must have at least one core course).

Each semester (a year has 3 semesters, Spring, Summer, Fall) a set of courses are being offered. Each course offered has 1 or more sections, each section has a 3-digit number associated with it. You should record the number of students enrolled in each section.

Each section is taught by an instructor. For each instructor, we need to store his/her (unique) ID and name.

##### ***Learning objectives and evaluation***

To evaluate a degree, there is a set of learning objectives that need to be met. Each objective has a (unique) code associated with it. It also has a (unique) title (120 characters max) and a description (text, with arbitrary length). Different degrees may have different sets of objectives (although some may overlap)

Each core course of a degree will be associated with at least one objective of that degree. Each objective of that degree must have at least one course that is associated with it.

At the end of each semester, the instructor for each section will need to enter information that will help evaluate the program. For each objective associated with the course, the following information needs to be entered:

- What is used to evaluate that objective. Standard answers include: "Homework", "Project", "Quiz", "Oral Presentation", "Report", "Mid-term", "Final Exam". But you should also allow the possibility of other answers.
- For each objective, the instructor evaluates how each student performs based on the evaluation. For each objective, there are 4 levels (A, B, C, F). Notice that he/she is NOT required

to enter individual student's performance. All he/she needs is to enter how many students achieve an A, how many achieve a B etc.

- Also, the instructor may choose to enter a paragraph (of arbitrary length) that suggests any improvement needed for that course to improve the result of that objective for subsequent semesters.
- Notice that a course may be associated with multiple degrees. For the same section, the evaluation for the same objective across different degrees may differ.

### Things to do

You need to design a relational database to store all the information. You need to store it in a relational database, using mySQL or MariaDB.

You will also need to develop an application that allows one to enter information to the database and retrieve information from it. Your application needs to support the following operations:

- Data Entry
  - Enter basic information about
    - Degree
    - Courses
    - Instructors
    - Sections
    - Learning objectives
  - Associating courses with objectives
  - Entering courses/section for a given semester
- Entering evaluations. This should be done in the following way:
  - The system should ask for the degree, the semester and the instructor.
  - Then it should list the sections that he/she taught each semester
  - It should state what information is already entered, and how much information has not.
  - The instructor has the option to enter new data or change existing data or make no changes.
  - Notice that a course may be associated with multiple degrees. You should give the instructor the option to duplicate the evaluation of one degree to the other degree(s).
- Querying. You should support the following queries:
  - Given a degree.
    - List all the courses that is associated with the degree (denote which course(s) is/are core)
    - List all sections that is being offered (in chronological order, where user can supply the time range)
    - List all objectives
    - List courses that is associated with each objective (you can ask user to enter objectives to be shown)
  - Given a course:
    - List all sections of the course for a specific range of semesters.

- Given an instructor:
  - List all the sections that he/she has taught (given a range of semesters)
- Queries involving evaluations:
  - Given a semester, list all the sections, and for each section, determine whether the evaluation information has been entered (also differential whether the optional “improvement” paragraph has been entered), partially entered (some information has been entered), or not entered at all.
  - Given a semester, ask the user for a percentage, then output the sections where the numbers of students that did not get the ‘F’ grade reach that percentage. (You can also incorporate the information into the previous query)

### **Implementation details**

You are required to use either mySQL or MariaDB as the backend. (SQLite is not allowed). You only need to host the database locally on your own machine. Your program should read the username/password/database name from a file that is to be included in the submission (You can name them whatever you want).

You are free to choose how to program all the functionalities, but you should provide a standalone program to do the task. Under no circumstances should the user need/be allowed to directly enter the database by the standard database GUI.

Your program should provide a (very basic) GUI for users to enter data and queries. You can leverage a web browser to do it if you want to.

You are free to choose your language to implement your program, and whatever tool to help develop the GUI. However, all the tools you use must be freely available (free trial is NOT allowed)