CS3425 Course Project: Online Exam System

Table of Contents

Important information

This project will be implemented in 2 phases with a required project demonstration to TA.

Phase 1: Database Design and Admin Functions.
Due Week11 Monday 11:59pm.

Phase 2: User Function Implementations.
Due Final week Wed 11:59pm.

Demo to TA: Show selected functions (40p).
Due Final Week Tuesday 11:50pm. This is subject to change due to TA's schedule.

Project description:

In this project, you will design and implement an extremely simplified online exam system. The system will be used by many instructors and students in one semester. **If there is any ambiguous requirement to you, write down your assumptions in your project report.**

1. Instructors teach courses and create exams for the courses they teach.

   a) Each exam in the course is uniquely identified by an exam name, such "Midterm", "quiz1", "quiz2" etc. We will also record when the exam is created, when the exam is open for students and when it is closed, and the total points of the exam. Please note different courses may use same exam names.

   b) Each exam will contain many multiple questions. In each exam the question is uniquely identified by the questions' number, such as Q1, Q2, Q3. Different exams may use same question numbers. Each question has some description. Each question is worthy some points. Different questions may have different points.

   c) Each question has multiple choices.
      - Each choice is uniquely identified by a choice identifier, such as "A","B","C" or "1","2","3", and some text to describe the choice.
      - The number of choices can vary for different questions. For example, the first question may have 3 choices (A,B,C) and the second question may have 5 choices(A,B,C,D,E). There is no upper limit about how many choices that a question may have.
      - Instructor provides the correct choice for each question, which will be used for automatic grading.

2. Students register courses. They take the exams after open date and before the closed date. After a student submits their answers, the system grades the exam automatically. The system also keeps student's answer for each question as well as when the student starts the exam and when the student finishes the exam.

3. There are three roles in our course evaluation system: Student, Instructor, and Admin.

   a) There are many students. Each student has a login account with password. The system records their account, name and password. Password should be stored as hash value using **sha-256 algorithm**. Students can preview the course information and register for courses. Students will be able to an online exam for each course they take. We will record the date and time of starting and completion. Password should not be saved as cleartext.

   In Mysql, the function `sha2(cleartext_password, 256)` can be used to generate the hash value for the password. You may use `char(64)` to store the hashed value.

   b) There are many instructors. Each instructor has a login account with password. The system records his/her account, name and password. Each instructor will teach multiple courses. The instructor will be able to create exams for their courses.

   c) The admin persons perform the system administration duty. They have database accounts and can access to the database functions directly. They don't need application accounts like student and instructor.
      - Admin persons create student accounts and instructor accounts which are application accounts, not database accounts. Admin creates these accounts with a temp password. When user logins to the system first time, they are required to set their password.

- Admin persons create courses that will be offered. Each course has an id, title and credit.
- Admin persons assign instructors to courses.

Phase One Tasks (110 points)

In this phase, you will design the database and populate some data manually into the tables. Then you will implement some admin functions and auto grading trigger.

1. (20 points) Create a small data set

To help you understand the domain, please design a small but complete set of the testing data based on the requirement.

Type	Testing Data
Instructors At least 2	
Students At least 5	
Courses At least 2 Courses with instructor assigned	
Registration data: At least 3 students for each course	
Exams At least 2 exams for a course with at least 3 questions with some choices	Example: CS2311 Exam1 (20p) (5p) Q1: What is the missing number in the sequence: 3, 6, __, 12, 15? A: 9. (correct) B: 10 C: 11 (5p) Q2: What is $\log_2 1024$? A: 10 (correct) B: 512 (5p) Q3: What is $\log_2 1024$? A: 10 (correct) B: 512 …. CS2311 Exam2 ….
Student Answers At least 2 students completed the exam for an exam	Example: CS2311 Exam1 Alice: started on 10/11 9:20 finished at 10:45, got 15/20 points Q1: A (correct) , Q2: A (wrong) , Q3 E (…) …. Ben: …

Be sure to include the following in your report:

The competed table above. Please remove the Example data first.

2. (50 points) E-R Model and Relational Schema

a) (15p) Construct an E-R diagram representing the conceptual design of the database. Be sure to identify primary keys, relationship, cardinalities, etc.

b) (10p) Create relational schema from your E-R design, and refine it based on the principles of relational design.

c) (25p) Write create table SQL statement for the relations in b). Make sure you to choose appropriate data type for each column and define constraints such as PK, FK (cascade on delete and update or not?), unique, not null, etc.

Be sure to include the following in your report:

- E-R diagram plus any explanatory notes if you will.
- Relational schema, plus any explanatory notes if you will.
- Table creation SQL script.

3. (20 points) Admin functions admin.sql

The following functions should be implemented inside database as Stored Procedure with necessary arguments with your design.

a) Procedure create_instructor(): Create instructors account with temp password.
b) Procedure create_student(): Create students account with temp password.
c) Procedure create course(): Create courses.
d) Procedure assign_teacher(): Assign instructor to teach courses.

Be sure to include the following in admin.sql

- Create procedure SQL statements

4. (20 p) Insert the small set of data: insertdata.sql

Inserting the small set of data into tables is a way to validate your database design.

- Use these procedures to insert the data of instructor, student and course in the testing data set.
- Use Insert statement to insert the rest of the data into table. Don't insert the score data that should be calculated automatically.

Be sure to include the following in installdata.sql

- Delete statements to delete all existing data
- Call procedure statements
- Insert statement
- Select statements to show all data in all tables.

5. (20 points) Auto grading trigger: autograder.sql

When student submits, the questions should be graded automatically. In this project we will use trigger to implement the auto grading function.

Be sure to include the following in autograder.sql

The trigger statement

Be sure to include the following in report

Run the installdata.sql to replace all the data

Use select statement to show that the columns for scores have been updated automatically, including the scores each question and for the exam.