Department of Computer Science
New Mexico State University
CS 482/502 Database Management Systems I

Project – part 3

General requirements:

- You should form a group consisting of 2 or 3 people to work on this part of the project (i.e., the maximum number of students in a group is three). You team members do not need to be the same to the team members when you work on the other two parts of the project. Once you form a team for this part of the project, you cannot kick out your team member (in particular at the last minute). If you cannot work with your team member, you will lose points (denoted below).
- Each group just needs to submit one copy of your program. **Points will be deducted if multiple copies are submitted.**
- You need to finish the tasks stated in three parts. NOTE that the due dates of the three parts are different.

Part 3: Access databases through the creation of an interactive application

Objectives:

- Write SQL language to conduct database manipulation including insertion, deletion, search, and updating.
- Write an interactive application to manage a database by considering possible database operations.
- Practice teamwork skills.

Tasks:

For the database that you created in Project – Part 1, create an interactive application to manage the above database. In all the later descriptions, I use interface to represent the way that users interact with the application. The interface does NOT need to be Graphical User Interface (GUI). You have flexibilities to design this interactive application. The goal is to make it work correctly and friendly to the user.

For the interactive application, you have two options.

- Option 1: If you are familiar with any web development language (e.g., PHP, JavaScript, Python, or any other language), you can create a small web application as this interactive application.
- Option 2: If you do not know any web development language, use either Java or Python to develop this interactive application. In this case, your application will be command line based. You need to design your program as friendly as possible.

Functionalities:

- (10 points) **Login**: Your program should contain a login interface, which does not need to be GUI.
    - If your interaction application is Option 1 (i.e., a web application), you need to a main file named index.html. The index page should ask users to input the database host, database name, username, and password. There should be two buttons “login” and “reset”.
        - The “login” button should allow you to “login” to the database with the parameters you input. If any parameter is wrong, the application should show the error

message. The login information should be used during the whole session when a user works on the database. From this login page, a user should be able to go to the following sub-system.
        - The “reset” button clears all the parameters and allows users to re-input them.
    - If your interaction application is Option 2, you need to have a main file to start the program. After the program starts, it should prompt users to input database host, database name, username and password. Once users input these information, the database should show whether it has connected to the database correctly. It should list options to conduct the Main functions (see below) and to logout. One example of an interface can be a list as follows
        - 1. Display all the digital displays.
        - 2. Search digital displays given a scheduler system
        - 3. Insert a new digital display
        - 4. Delete a digital display
        - 5. Update a digital display
        - 6. Logout
- (75 points) **Main functions (see below for detailed description)**: you should design a main function interface (a webpage or a list shown above) and other interfaces for the following tasks. From the main interface, users can perform different operations described below. Note: There is no fixed interface requirement. You can design it in a way that you think friendly or as simple as you want. The grading will focus on the functionality, instead of the interface design.
- (5 points) **Logout**: from the main function panel page or main menu interface shown above, you should be able to “logout” and go back to the login page.

Main functions:

Please implement the digital display sub-system. In this sub-system, you should be able to

- (1) Display all the digital displays. For each display, if a user clicks the model no, the detailed model information should be displayed.
- (2) Search by inputting a scheduler system. Show the digital displays satisfying the search condition.
- (3) Insert a new digital display. When the model of this digital display does not exist, you should design a page to ask users to input the model information first. Display all digital displays after insertion.
- (4) When displaying all digital displays’ information, users can choose one digital display and delete it from the database. For the digital display that needs to be deleted, if none of other digital displays has the same model no, after the digital display is deleted, its corresponding model should also be deleted. Display all digital displays and all models after deletion.
- (5) When displaying all digital displays’ information, users can choose one digital display and update it. Show the information of all digital displays after updating.

(5 points) Create a README.txt to write down the work allocation of the different team members.

(5 points) Each team member needs to create a PeerEvaluation_<yourname>.txt to include a peer evaluation to your team members for this part of the project. Your peer evaluation should include a score (1 to 5, with 1 being poorest and 5 being best) and a justification for your score. The justification does not need to be long.

Submission: You have to submit your assignment electronically (through Canvas). Printed copies are not accepted. See course syllabus for policies on late submission and plagiarism.

- Please create a zip file to contain the source files and the readme file, name the zip file “p3-(your last names).zip”. For each team, only submit one copy of the zip file. **Points will be deducted if multiple copies are submitted**.
- Each team member should submit your own PeerEvaluation_<yourname>.txt so that this information is confidential to your team members.