CS 3630
Database Design and Implementation
Spring 2020

Course Project

Goals

1. Be able to design a database for a small application.
2. Be able to programmatically interact with a relational database management system.

Task: Create a small application of your choosing, which accesses and operates on a database. The application should have a fully functional (not necessarily pretty) user interface, to interact with the database. The design of your database should follow all conventions and good practices taught in this course. I would like you to have a strong project, which you could potentially show to future employers at the end of this course.

1. **Database**
   Design and implement a database for your application.

- Relational Database Management Systems permitted
  - PostgreSQL
  - SQL Server (Express)
  - MySQL
  - Oracle (Express)
  - Other (consult with me for other RDBMS)
- The database must consist of at least 12 tables, which should include the following
  - Primary keys
  - Foreign keys
  - Unique keys
  - Constraints
  - Triggers / Functions
  - Indices
- Whenever feasible, do not allow invalid data to be stored in your database

2. **Application**
   Create an application with a user interface to interact with the database.

- There are no size requirements (such as lines of code) for the application.
- The application should be a standard desktop application
- Programming languages permitted
    - Java (Use JDBC)
    - C# (Use Npgsql)
    - C++
    - Ruby
    - Python
    - Other (consult with me for other languages)
- The application should operate on the database in the following ways
    - Insert data
    - Select data
    - Update data
    - Delete data (virtually if not physically)
- The application should be error-free. The user interface just needs to be functional. The user interface does not have to look good.

3. **ER_Diagram**
   Create an ER-Diagram for the database.
    - Foreign key relationships should point to the specific attributes.
    - All unique constraints should be labeled (single and multiple).

4. **Project Presentation**
   Each team will prepare a 10-minute presentation to present during the final exam period. The project presentation should consist of slides (PowerPoint or Google) and a demonstration of the application. The presentation slides should contain the following
    - Team name
    - Team members
    - RDBMS choice

- Programming language choice
- Description of application
- ER-Diagram of the database

5. **Team Report**
   Each team will submit a document stating how the team completed the work for the assignment (team dynamics). This should include a high-level description of what work each team member did (who did what). It should also include an estimated amount of time each team member spent working on their task. Explicitly state the time each team member worked on the project for each task they have worked on. I recommend using a table for this. Any issues of concern can be addressed in the team report (what were struggling points with the assignment). Include the team meeting times.

6. **Individual Report**
   Each student will submit an individual report in the separate corresponding assignment on canvas (the individual one). The individual report will include more specific details on what you have worked on. This differs from the team report, which will include only higher-level task information, where lower-level details can be discussed in this report. If the high-level details are sufficient, they can be restated. In addition to a more detailed description, any discrepancies in the team report can be included as well as any issues that you feel are important but did not feel comfortable including in the team report. The individual report should not be longer than a length of one page, however there is no strict page limit.

Submission

1. **Database creation script**: Submit a single SQL file as a file upload on Canvas, which creates your database. Include any other SQL scripts to prepopulate the database if necessary.
2. **Project submission**: Submit the subversion revision number on Canvas for your final project submission.
3. **ER-Diagram**: Submit an image (GIF, PNG, JPG) of your ER-Diagram as a file upload on Canvas. You may also include the .vpp file, if you used Visual Paradigm to generate the ER-Diagram.
4. **Presentation slides**: Submit the share link (Google or OneDrive) to the slides in the submission on Canvas.

5. **Team Reports**: Include a link to the shared document (Google or OneDrive) that has edit privileges for you team report. Name this doc "TeamReport_<team number>".
6. **Individual Reports**: Upload a word document for your individual report in the corresponding individual assignment on Canvas. Name this doc "IndividualReport_<your name>".

Grading

Criteria	**Possible Points**
Database design	45
Application functionality	20
Application user interface	5
ER-Diagram	15
Project presentation	5
Team report	5
Individual report	5
Total	100