Final Design Project

ENSF 608 Fall 2023

Department of Electrical and Software Engineering
Schulich School of Engineering

The objective of this project is to apply your understanding of course concepts, database design, and SQL queries on a real-world database application.

Due: Sunday, December 3rd, 11:59 PM (dropbox grace period to Wednesday, December 6th at 11:9 PM)

Submission: This is a team assignment. You may work in groups of 3-4. Your submission must be your team's own original work. Only one team member needs to submit the final files, but all names should be included on each document and within the dropbox description. Teams should verify the file upload process together- if one member uploads an incorrect file, it will impact the grade of the entire team.

There are multiple components to the project. Your submission should consist of five files:

- A single .pdf file with your EER diagram
- A single .pdf file with your relational model
- A single .sql file to build and populate your database
- A single .sql file with your query demonstration
- An .mp4 or link to your video demonstration

Please upload your submission to the project D2L dropbox folder.

Weighting: You will receive a project grade out of 100%. It is worth 30% of your overall grade.

Grading:

The EER diagram and relational data model should follow the formatting conventions outlined in the lecture notes. Your solution may be computer generated or hand-drawn but must be legible.

The EER diagram should include correct notation for entity types, relationship types, attributes, key attributes, relationship attributes, and cardinality. All relations should have a name, primary key, attribute(s) as necessary, and foreign key(s) as necessary. Use arrows to represent foreign keys (referential integrity).

Your SQL files will be run through MySQL Workbench. **All statements must compile and execute correctly to receive marks.**

Marks will be deducted for incorrect or missing information. Solutions must be neat and organized.

ENSF 608 Project Deliverables (30% of course grade)

Conceptual Database Design (33.33% of 608 project grade, 10% of course):

Based on the provided requirements narrative, design and draw an EER diagram for the described database application. Your solution should follow the model notation presented in class and should include cardinality ratios and participation constraints. Your diagram should also be accompanied by a half-page description explaining your design decisions and any assumptions that were made.

Your solution may be handwritten or typed, and you may draw your diagram by hand or by using software tools. Handwritten work may be scanned or clearly photographed. Marks will be deducted for incorrect or missing information based on the provided narrative. Solutions must be neat and organized.

Logical Database Design and Creation Code (33.33% of 608 project grade, 10% of course):

Map your conceptual schema into a relational data model, including all primary keys and referential integrity constraints (foreign keys). Then use your relational model to create a .sql script that could be used by someone else to initialize and populate your database. You are free to create your own example data that aligns with the narrative.

Your solution may be handwritten or typed, and you may draw your diagram by hand or by using software tools. Handwritten work may be scanned or clearly photographed. Marks will be deducted for incorrect or missing information based on your EER diagram design. Solutions must be neat and organized.

Query Code and Video Demonstration (33.33% of 608 project grade, 10% of course):

Implement your database in MySQL Workbench. Use Zoom or another tool of your choosing to record a short demonstration of your database (5 minutes or less!). All team members should be part of the demonstration. In the demonstration, show the following elements using your database. In your submission, you should include a .sql file that contains the queries listed below:

1) Show all tables and explain how they are related to one another (keys, triggers, etc.)
2) A basic retrieval query
3) A retrieval query with ordered results
4) A nested retrieval query
5) A retrieval query using joined tables
6) An update operation with any necessary triggers
7) A deletion operation with any necessary triggers

Marks will be deducted for incorrect or missing information and may also be deducted for videos of excessive length. Videos should have clear audio. Cameras should be on or you may use a professional-looking headshot. This is your opportunity to demonstrate how your database works and why it is a correct solution for the client.

Question Narrative

Large travelling performance companies, such as Cirque du Soleil, have a variety of performers, venues, and itineraries to consider. This information needs to be stored in a logical way that allows producers to access information quickly and effectively. For this project, you will model and implement a sample database that could be used to store and access the necessary information. Consider the elements of the functionality and how they interact with one another. Your design may be different from others- you have flexibility in how you choose to meet the requirements of the client. You may assume that only the aspects described below need to be considered.

Your client is Cirque du Soleil. Cirque du Soleil has several different travelling shows such as Alegria, Kooza, and Corteo, so they need to track information for all of them. Each show is identified by a combination of its name and its performance year (e.g. “Corteo” and “2006”). Each show has a sponsor and a producer.

Shows are hosted by venues, with a date and location assigned to each hosting. The stored venue information includes a unique numeric ID, the name of the venue, its capacity, address (street address, city, province/state), and whether it is accessible by transit (Y/N). Since the shows move around, a show can be hosted by multiple venues, and venues may host more than one show.

Performers are assigned to a specific show. The stored performer information includes a unique numerical ID, name (first and last name), citizenship, birthdate (use SQL date object), list of medications, and any special diet information (e.g. vegan, high protein, etc.). All performers can be classified as either a musician, aerialist, or entertainer. Each musician may play multiple instruments- all of their instruments should be list in the database. The database should track the sport and equipment of each aerialist, as well as the main act of each entertainer. Performers may have another performer listed as their understudy.

Cirque du Soleil can be a dangerous show given the aerial acrobatics. All performers also have one emergency contact listed, which includes their name (first and last), phone number, and relationship to the performer.