Principles of Data Management CSCI 320
Movies Domain

v4.0

1 Project Description

This is a semester long project.

You will be required to create a database based application in a team. You will be given a choice of application domains. This document details the requirements for the movie domain.

This domain will require that you create a database to keep track of movies, their cast members, ratings, and watch count for multiple users. It is a scaled down version of what Netflix can do.

The project will be broken into 4 phases. Each phase will build on the prior phase. A basic description of each phase is outlined here. More details in the remainder of the document.

- Phase 1: Select a domain and identify the type of application you are going to develop. Due: January 28, 11:59 pm.
- Phase 2: Generate the conceptual model of the database for the approved domain. Due: February 18, 11:59pm.
- Phase 3: Generate and load data into the database. In addition, you will submit your database application. At this point, your application must allow users to perform basic operations to manipulate the data. Due: March 18, 11:59pm.
- Phase 4: Perform data analysis on your database and generate a poster with the results. Due: April 15, 11:59pm
- Peer Evals: Evaluate the performance of each member of your team, including yourself. Due April 15, 11:59pm (no late window)

2 Group Selection

You will work in a 3-4 members team for this project. You should find a team and self enroll in any available group at myCourses before the expiry date. For more information about the expiry date, please check myCourses.

You can look for a team by posting a message in the Discussion Topic at MyCourses, ”Looking for a Team”. Cross section groups will not be permitted. Students not enrolled in any group after the expiry date will be randomly allocated in a group by your instructor.

Teams are formed at the beginning of the semester, and they cannot be changed unless otherwise stated by your instructor.

3 Data Requirements

This section describes the minimum data to be stored in your database. Notice that you may need to store more information to comply with the application requirements from sections 4.3.1 and 4.4.1.

- Multiple users can be managed by your application system. You must stored at least the following information about each user:
  - Username (required, unique)
  - Password (required)
  - First Name (required)
  - Last Name (required)
  - Email (required, unique)
  - Creation date (required)
  - Last access date (required)
- The database will contain information about movies, cast members and directors. Each movie, cast member and studio must be uniquely identifiable.
  - People (cast members and directors) will have a name (required, unique).
  - Movies will have a title, genres, release date, length (in minutes), cast members and director and the studio that made the movie. All those fields are required. Movies should also have a rating (MPAA) and user rating.
- Users must be able to flag a movie as watched and give a star rating to it (1-5).
- Users will be able to create collections of movies. Each collection must have a name (required).

4 Phases

Below outlines the minimum submission requirements for each phase of the project. Every phase must be submitted through the appropriate assignment dropbox at myCourses. Submissions are due by 11:59 pm on the day they are due.

You can only submit one copy of each phase, only one member of the team should be in charge of the submission. Every dropbox will accept only one file. You can upload as often as you like, but only the last submission counts.

All phases will have a 48 hour late window. No projects will be graded before the end of the late window. During the late window, no questions will be answered.

All phases must be submitted in the appropriate dropbox in order for it to be graded.

4.1 Phase 1

During this phase your will begin to outline your project design. This description must be in the Introduction section of the project report using the LaTeX report template of this course (download the zip from myCourses). You will continue to fill in the rest of this report for future phases. This first phase of the report will include:

- Default group number from myCourses
- Team Name: You must choose a new name for your team
- The names of all team members
- Your selected domain, and a backup domain

- The introduction of your project including your approach, how you intend to provide an interface to your dataset (command line application, web app, desktop app), and the front end language you plan to use.

Your selected domain will be granted on a first come, first serve basis. Only 1/4 of the teams in any given section will be granted a domain. For that reason, your domain must be approved before proceeding with the remainder of the project. You can submit early (and email the instructor) to verify approval before the due date, though once it is approved, it becomes final. Your approved domain will be listed on myCourses under groups. For example, if your domain is Widgets, your team name in myCourses would be "SQL Injection (Widgets)".

Submit a PDF of the report outlined above, named **Phase1.pdf**. Submission of anything other than a PDF will earn a 0 for this phase.

Excepted submission structure is included in Part 9, Submission.

4.2 Phase 2

During this phase you will design your database system. This phase is a complete design for the rest of the semester. The goal is to create a conceptual data model using the EER model for your given domain from Phase 1. It must capture all the data requirements listed above (section 3), as well as all the application requirements from sections 4.3.1 and 4.4.1. The following items are due:

- The report template will be updated to include:
  - details the decisions that were made for the EER diagram
  - explains how the reduction to tables were done for each of the entries in the EER diagram
  - information in the report must match the EER diagram and the reduction to tables submitted
- An non-hand drawn EER Diagram that:
  - uses the proper EER notation (ie. rectangles for entity types, underlined primary keys, etc)
  - depicts the entity types from your domain (for all phases)
  - properly uses weak entities, specialization, etc. (ie. there must be a reason for the specialization; special relationship or attributes)
  - depicts the relationship types between the entity types. The relationship types must enforce the data and application requirements of your domain
  - has proper cardinalities. Cardinalities must make sense with the requirements provided and uses the correct notation
  - properly uses attributes types on entity and relationship types; including key attribute types, derived attribute types, etc.
- Reduction to tables that:
  - uses correct notation for reduction to tables
  - matches EER diagram (i.e. each entity type is an independent table, a multi-value attribute in the EER diagram is stored in its own table)
  - handles cardinalities properly (i.e. many-to-many relationships reduce to an additional table)

  - every table has a primary key
  - foreign keys are properly added where needed

Submit a Zip file containing 3 files, the report outlined above (Report.PDF), the ER Diagram (ERDiagram.PDF), and a Reduction to tables (Reduction.PDF), named **Phase2.zip** Submission of anything other than a Zip containing PDFs will earn a 0 for this phase.

EER diagram must be easily readable in the PDF.

You should update your peer evaluation for this phase at this time, though you are not required to submit it at this time.

Excepted submission structure is included in Part 9, Submission.

4.3 Phase 3

During this phase, you will load your data into a real database, based on the EER diagram and reduction to tables previously submitted (corrected based on any feedback). You will also submit your database application program, as well as a 6-10 minutes video demonstrating your application.

4.3.1 Application Requirements

- Users will be able to create new accounts and access via login. The system must record the date and time an account is created. It must also store the dates and times users access the application
- Users will be able to create collections of movies.
- Users will be to see the list of all their collections by name in ascending order. The list must show the following information per collection:
  - Collection's name
  - Number of movies in the collection
  - Total length of the movies (in hours:minutes) of movies in the collection
- Users will be able to search for movies by name, release date, cast members, studio, and genre. The resulting list of movies must show the movie's name, the cast members, the director, the length and the ratings (MPAA and user). The list must be sorted alphabetically (ascending) by movie's name and release date.
- Users can sort by: movie name, studio, genre, and released year. Results can be ascending and descending.
- Users can add and delete movies from their collection
- Users can modify the name of a collection. They can also delete an entire collection
- Users can rate a movie (star rating)
- Users can watch a movie individually or they can play an entire collection. You must record every time a movie is played by a user. You do not need to actually be able to play movies, simply mark them as played
- Users can follow a friend. Users can search for new friends by email
- The application must also allow an user to unfollow a friend

4.3.2 Submission Requirements

- The report template will be updated to include:
  - a current design phase section (if your ER or Reduction to tables changed, update them)
  - contains sample (2) SQL statements used to create your tables
  - contains sample queries (2 for each table) for populating the data
  - contains a description of how the data was loaded into the database (was the source manipulated, did you use a tool, etc) with 5 sample insert statements
- An EER Diagram (even if not updated).
- Reduction to tables (even if not updated).
- Large amount of data loaded into the postgres server that matches the reduction to tables and the EER diagram, see below for what is considered a large amount of data.
- Data in the database should be normalized to 3NF.
- The source code of your application program (in a src folder). At this point, your application must be able to perform all the operations stated in the **Application Requirements** section of this phase.
- A 6-10 minutes video demonstrating the application running and manipulating data with a voice over explaining the application. We will not be running your application, it is important that your demo covers the functionality of the application.

Data must be loaded into the proper account area and permissions to the database must be given to the instructor(s).

Data in the dataset may have been converted to numbers for storage. (ie. 1 for male, 2 for female). You must store them as their non-numeric values (ie. male and female).

You must have enough data in this phase to do phase 4 and have it loaded for this phase. How much is enough? 10s-100s of rows in each table, while a M:N table should have 200-500. To simplify, after a join with a complex query, you should have several thousand rows.

Submit a Zip file containing the report outlined above (Report.PDF), the ER Diagram (ERDiagram.PDF), and a Reduction to tables (Reduction.PDF), a Video (Video.mov or other type of movie file, no links) and your source files (inside a src folder) in a file name **Phase3.zip**. All 3 PDF files should be in the root of the zip. Failure to follow any submission guidelines (location of files, type of files, zipping, naming, etc) will earn a 0 for this phase.

You should update your peer evaluation for this phase at this time, though you are not required to submit it at this time.

Excepted submission structure is included in Part 9, Submission.

4.4 Phase 4

For this phase, you must complete your application program including the recommendation system and some extra functionalities. You must also perform some data analysis to discover useful information.

4.4.1 Application Requirements

- The application provides an user profile functionality that displays the following information:
  - The number of collections the user has
  - The number of followers
  - The number of following
  - Their top 10 movies (by highest rating, most plays, or combination)
- The application must provide a movie recommendation system with the following options:
  - The top 20 most popular movies in the last 90 days (rolling)
  - The top 20 most popular movies among my friends
  - The top 5 new releases of the month (calendar month)
  - For you: Recommend movies to watch to based on your play history (e.g. genre, cast member, rating) and the play history of similar users

4.4.2 Submission Requirements

- Final version of your application program. At this point, you application must include all the functionalities stated above. Submit your source code (in a src folder)

- The report template will be updated to include:
  - explaining the process/techniques used to analyze the data (what types of algorithms were used, did you use a tool for analytics, or did you create materialized views, etc)
  - explaining the indexes created to boost your application program's performance
  - containing an appendix listing all of the SQL statements used in this phase

- A poster showing:
  - your team name
  - the names of all team members
  - the observations from the data analytics
  - technologies used (Excel, Python, etc)
  - visual representation of the data (charts, graphs, and other visual representations are required)

  Designing an effective poster:
  - include the team name and the name of all team members at the top
  - keep any text brief
  - do not use all capital letters
  - user graphics (charts, graphs, etc) that can be understood in one minute or less
  - add a descriptive caption below each figure; table heads should appear above tables. Use the abbreviation (e.g. "Fig. 1") at the beginning of each caption. That will make easier to refer to those figures during your poster presentation.

    - have 3 graphs/charts explaining the data with brief descriptions. These should not be trivial. Two of these can be trivial charts or graphs, one must be complex (getting data from at least 4 tables)
    - use color, sparingly. Find a color palette that works and is not distracting.

  Poster must be easy to read and understand. The viewer should gain new knowledge or insight by just looking over your poster. You can find a poster sample at myCourses. You can also can take a look at the CS MS Capstone project's posters. To access to that information, you will need to login with your CS account.

- A **single** 7-10 minutes video that demonstrate the final version of your application and presents your poster. Your video must be structured as follows:
    - Start by demonstrating the final version of your program. Make sure to demonstrate all functionalities defined in the **Application Requirements** from Phase 4. During the demonstration, you must show, from your source code, at least two complex queries (e.g. multiple joins, nested queries, correlated queries) implemented during this phase. We will not be running your application. It is important that you demonstrate all the required functionality.
    - After that, you will present your poster.
        - Familiarize viewers with the fundamentals of your program quickly and easily
        - Use the information from your poster to present only the highlights
        - Explain the patterns or conclusions drawn from the data analysis

The type of data analysis you will perform is up to you. Here you have some ideas:

- use analytics tools (e.g. Weka, R)
- perform Exploratory Data Analysis (EDA) such as
    - export your data into Excel to determine interesting information and generate charts (e.g. bar chart, pie chart, histogram, scatter plot, correlation matrix)
    - time series analysis
    - descriptive statistics
      Notice that you are allowed to add some descriptive statistics in your report and/or poster, but your data analysis cannot be solely conform by this type of EDA

Note that you can also perform descriptive and/or predictive analysis of your data but it's not required for this course. **Important!** Your analytics must not be trivial (e.g. finding that people like pizza).

Submit a Zip file containing the report outlined above (Report.PDF), a Poster (Poster.PDF), the EER Diagram (ERDiagram.PDF), and a Reduction to tables (Reduction.PDF), a Video (Video.mov or other type of movie file, no links) and your source files (inside a src folder) in a file named **Phase4.zip**. All PDF files should be in the root of the zip. Failure to follow any submission guidelines (location of files, type of files, zipping, naming, etc) will earn a 0 for this phase. Excepted submission structure is included in Part 9, Submission.

5 Peer Evaluation

You must submit a peer evaluation for yourself and your team members. Failure to submit a peer evaluation will result in a 10% deduction on your project grade. A template is available on myCourses.

Your peer evaluation is submitted at the same time as Phase 4, though it is a separate submission from your phase 4 submission. The peer evaluation is confidential and will not be made available to your teammates.

It is recommended that you update the peer evaluation after each phase to keep the information as accurate as possible. Your peer evaluation must contain comments which support your ratings, simply stating, ”Person A did well on this phase” is not enough.

6 Datasets

There are plenty of free datasets available that you can use for your project. They may not contain all the information that your project required but you can combine several datasets together and/or you can also generate synthetic data. Find below some datasets you might find helpful:

- `https://www.kaggle.com/shivamb/netflix-shows`
- `https://www.kaggle.com/tmdb/tmdb-movie-metadata`
- `https://www.kaggle.com/fnunezsanchez/rotten-tomatoes-top-movies`

You can find more datasets in Kaggle or using Google dataset search engine.

7 Project Constraints

This section outlines details about any project constraints or limitations.

Constraints/Limitations:

- You must use a provided domain.
- The constraints `required` and `unique` defined in your selected domain must be enforced for the database system by the definition of integrity rules.
- There may be also some data attributes with an specific domain. That domain must be also enforced by creating integrity rules in the database system (e.g. the difficulty level of a recipe, the status of a borrow request, the rating value).
- The use of an ORM is prohibited. You must write your own SQL statements for phases 3 and 4. You are welcome to use a tool for importing data and creating the tables.
- Your data must be loaded onto the CS Postgres Server (reddwarf).
- Data analytics must be done on the data loaded, not the original source.
- Your analytics must not be trivial (e.g. finding that people like pizza).

8 Grading

Your project will be graded according to the following:

- 10%: Phase 1
- 25%: Phase 2
  Total points: 100
  - 15%: Report
  - 55%: EER Diagram
  - 30%: Reduction to tables
- 30%: Phase 3
  Total points: 100
  - 20%: Report
  - 10%: EER Diagram and reduction to tables
  - 30%: Database implementation and data population
  - 25%: Application Program
  - 15%: Demo video
- 35%: Phase 4
  Total points: 100
  - 30%: Report
  - 30%: Application Program
  - 20%: Demo video
  - 20%: Poster

General deductions:

- -10% - Incorrect zip (must be flat except source folder)
- -20% - Not following the report template
- -100% - Submission of anything other than a Zip file

9 Submission

Emailed solutions will not be accepted. **All phases will have a 48 hour late window.** During the late window, no questions will be answered.

Failure to follow the submission guidelines for each phase will result in a severe deduction for that phase.

Below is what you should submit for each phase. For more details about what these files contain, see the submission instructions for each phase.

- Phase1: Phase1.pdf
- Phase2: Phase2.zip
  - Report.pdf

  - ERDiagram.pdf
  - Reduction.pdf
- Phase3: Phase3.zip
  - Report.pdf
  - ERDiagram.pdf
  - Reduction.pdf
  - src
    - source file 1
    - ....
    - source file n
  - Video.mov
- Phase4: Phase4.zip
  - Report.pdf
  - ERDiagram.pdf
  - Reduction.pdf
  - Poster.pdf
  - src
    - source file 1
    - ....
    - source file n
  - Video.mov

FAQ

This section contains common problems and solutions when working on this project.

Why am I unable to connect to the database at peek times?

Make sure you close the DB connections. While they will eventually time out, it is important you close your connections the same way you'd close files.

How should we compute analytics? You did not provide details on the queries to write.

You are welcome to be creative in this phase. The important part is that the queries are not trivial (such as users increase over time). There should be something that you wouldn't say, ”of course they did”.

Do we need to deal with security?

No. Passwords can be stored in plaintext. In a real project, these should be encrypted, but is not required for this project.

Can we see an A-quality poster?

No. We used to provide an A-quality poster and every poster turned in looked exactly like it. Take a look at the Master's projects. You'll know which are good.

How can avoid checking in a file into github with my CS credentials for the DB connection?

We recommend you create a text file that has your credentials to connect to the database (something like dbInfo.ini), that is in your **.gitignore** file. This will prevent it from being checked in. Each one of your teammates will need to have a similar file with their own credentials in it.