CMPUT 291 - Mini Project 1

DESIGN DOCUMENT

Submission Date: March 9, 2022

Group Members:

1. Hridyansh Dugar , ccid:dugar (1667524)
2. Swastik Sharma, ccid: swastik (1670432)
3. Rana Thind , ccid:ranasunj (1670185)

---

1. **A general overview of our system with a small user guide:**

The goal of this project is to use SQL in Python, and to demonstrate some of the functionalities that result from combining SQL with interfaces like Python. In this project, a software system is built to keep the enterprise data in a database and to provide services to users. It will store data in a SQLite database. A user, who can be a customer agent or a traffic editor, can run the program using the terminal and typing in “python3 main.py <data-base-name.db>” and interact with the database to perform various queries.

User Guide:

- The user starts the application by making sure that the db file is in the same working directory as the other source code files.
- Then the user opens a terminal window and enters the command - python3 main.py <data-base-name.db>

- After which the login screen comes up and the user can enter the id and password. If the user is not registered then he can register using the register option in the next menu
- After successful login the user can perform various tasks and the program will not end until the user logs out.

2. **Detailed Design of the program:**

Roles were assigned to each type of user and it was linked to the functions they have permissions for. For instance, all users can login, quit the program, and logout of the program. In addition, a customer can register using a name and a passcode, for a person in the database. While editors cannot perform these operations, they are able to add a movie and update a recommended one.

Login Screen:

In addition to being able to log in on the login screen using a valid user id and password, a user is able to perform subsequent operations based on their type (customer or editor).

System functionalities allowed to a customer after a successful login:

1. Start a session:
   Using this option the user can start a session, it is automatically started when the user starts to search for movies

2. Search for movies:

   This function displays the movies and info which have the given keywords in the title, role, or movie person name.

3. End watching a movie.

   A user can stop watching the movie that is being played using this function.

4. End the session.

   The user can end the session using this function

System functionalities allowed to Editors :

1. ***Add a movie:*** In this the editors can do several functions. Editors can add a movie to the database by providing a unique movie id, a title, a year, a runtime, and a list of cast members and their roles. To add a cast member, the editor will enter the id of the cast member, and your system will look up the member and will display the name and the birth year. The editor can confirm and provide the cast member role or reject the cast member. If the cast member does not exist, the editor should be able to add the member by providing a unique id, a name, and a birth year.

2. ***Update a recommendation:*** The editor is able to select a monthly, an annual, or an all-time report and see a listing of movie pairs m1, m2 such that some of the customers who have watched m1 have also watched m2 within the chosen period. Editor gets a table printed which shows the watched movie id, recommended movie id, an indicator showing if the pair is in the recommendations and its score in the stated table. The editor is also able to select a pair and (1) add it to the recommended list (if not there already) or update its score, or (2) delete a pair from the recommended list.

3. **Testing Strategies:**

   - Started building the program by having all the set-up data in place and writing one query at a time. This allowed us to build a bug-free program in small steps.

   - A small test database was also created to test the functionality of the queries.

   - Error checks were performed by adding try and accept blocks.

   - Rigorous error testing was done after every completed portion of the devised assignment goal.

4. **Group Break down Strategy:**

Break-down of the work items:

   - Rana worked on the login screen (authentication system), SQL injection, testing and error checking.

   - Swastik worked on string matching, testing and creating system design for editors (This includes adding a new movie or updating the recommended ones)

- Hridyansh worked on designing the system functionalities (starting a session, searching a movie, ending watching a movie, and ending a movie session)testing and error checking.

- At the end of each individual component we sat down together to integrate the components into the collective program.

Time spent of each member :

- Hridyansh spent approx.16-17 hours
- Swastik spent approx. 14-16 hours
- Rana spent approx. 12-13 hours

Method of coordination to keep the project on track

- The group met at the beginning of the project to discuss an overall strategy and divide the work among themselves.

- The group devised to use version control software such as Github

- The group also met on the submission day to go through the code together, divide it into separate files for improving code quality. Changes were also made in the code to ensure it works well with the python version on the lab machine

- The code was pushed on GitHub so the members could see the current state of the project at all times and review each others' work as well.

- All the other communication was done in a group chat on a Discord server, including modifying work distribution as necessary and tracking progress.

- The design document was prepared by collaborating on a google doc

Assumption:

The user starts a session and watches one movie in that session and doesn't try to start a new session when there is an active session.