TECHNISCHE UNIVERSITÄT
IN DER KULTURHAUPTSTADT EUROPAS
CHEMNITZ

Fakultät für Informatik
Professur Datenverwaltungssysteme

Advanced Management of Data
Project Task
Winter Semester 2021/2022

1. Introduction

New films are produced non-stop and it is quite easy to watch them via online video streaming services nowadays. As binge watching has become popular, you might easily lose track of all these different movies and series episodes already seen. A streaming service might help to remember, which films were already watched by one person, and can also suggest new films based on this person’s taste, but if multiple persons share one account or if you use multiple streaming services or watch traditional linear TV or go to cinema or watch films using physical media like BD, DVD or VHS, things get more difficult.

Some kind of film manager can help to keep an overview about already watched films and should also be able to suggest new films based on the personal taste. Your task will be to implement a very simple prototype of such a tool according to the requirements on the following pages.

2. Task Description

There is only one practical project task for all students in this semester. This task may be processed in groups of up to two participants, but it is not allowed to share solutions with other groups. If there is more than one student in a group, the work should be divided evenly between both members and the indication of the authorship is required for all parts. The submission consists of

1. a PDF-file of the term paper in paper format A4 and
2. a separate ZIP-archive (with a maximum size of 10 MiB) that contains
   - the program sources including a script to initialize the database and
   - a small manual on how to use the sources to get a working program.

The detection of significant solution parts shared between submissions of different groups is considered an attempt to defraud and marked accordingly for all involved students (independent of who shared solutions or who used them).

2. Task Description
2.1 Term Paper

A term paper is to be written, that satisfies the following conditions:

1. There is an amount of about 12 pages of content (i.e. excluding cover, index, lists, appendix, bibliography, …).
2. A good form and a balanced ratio of pictures and text is kept.
3. On the cover page, there is recorded the name, study course and matriculation number of all participating students.
4. An overview of all utilised technologies (excluding PostgreSQL and PL/pgSQL) is given along with a short motivation, why they were chosen to solve a certain subtask.
5. The database model is visualised by an entity relationship diagram using the Unified Modelling Language as well as a corresponding relational schema.

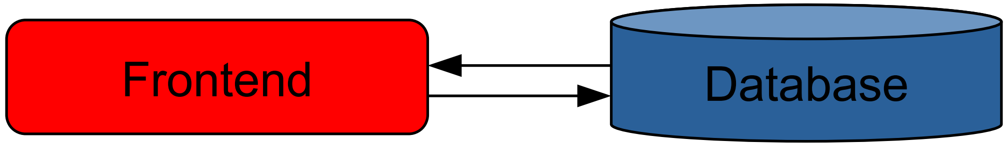
2. Task Description
2.1 Term Paper

6. The project is presented in such a way that, after reading, you know all parts of the program and their functionalities without having explicitly executed them.
7. It is discussed under which conditions a distributed database could be used with the project and how this distribution can be achieved in a good way. In case there are multiple possibilities, no more than 3 different scenarios are described.
8. All used sources, libraries and technologies are referenced.

2. Task Description
2.2 Programming

The program consists of

1. a PostgreSQL-database with the program logic implemented in PL/pgSQL,
2. a frontend for interaction with the data from the database and
3. some dummy data for demonstration purposes.



2. Task Description
2.2.1 Database

1. The database is provided by PostgreSQL.
2. The program logic is implemented directly in the database using PL/pgSQL as main programming language. The database offers the functionality to
   1. manage films,
   2. manage film related persons,
   3. manage film user ratings and
   4. get film watch suggestions for users.
3. Users are identified by a user providable name, which can be changed easily for demonstration purposes (so there is no need for complex user management).
4. All user and film data (and related information) is stored in the database.

2. Task Description
2.2.1.2.1 Management of Film Related Persons

The management of film related persons includes the options to:

1. Add a film related person.
   1. At least a unique name for identification has to be provided.
   2. More attributes (like date of birth, sex or curriculum vitae) may also be implemented.
2. Get an overview about all existing film related persons.
3. Change the attributes of a film related person.
4. Remove a film related person.
   1. Films, which have a participation of the removed person in any role, get removed, too.

2. Task Description
2.2.1.2.2 Management of Films

The management of films includes the options to:

1. Add a film.
    1. A film may be subordinated to one already existing film (no cycles or self references are allowed).
    2. At least title and release year have to be provided, which must be unique within current level of hierarchy.
    3. Films may have one or more genres.
    4. One or more film related person may be linked to a film with a role (like actor, director or producer).
    5. More attributes (like minimum age of audience or production country) may also be implemented.
2. Get an overview about all existing films.
    1. Subordinated films shouldn’t be shown at first and may be expanded in a second step.
3. Change the attributes of a film.
4. Remove a film.
    1. Subordinated films get removed, too, and related film user ratings are also removed.

2. Task Description
2.2.1.2.3 Management of Film User Ratings

The management of film ratings for a selected user includes the options to:

1. Add a film rating.
    1. A rating scheme of your choice may be implemented.
2. Get an overview about all existing film ratings.
3. Change a film rating.
4. Remove a film rating.

2. Task Description
2.2.1.2.4 Film Watch Suggestions for Users

1. A selected user with at least one film rating should get a list with other films to watch, which have not yet been rated by this user.
2. As a complex suggestion algorithm is out of scope of this work, just a very basic approach should be implemented, based on the already existing film ratings of the user, by searching for other films with shared attributes of preferred films, while avoiding attributes of disliked films.

2. Task Description
2.2.2 Frontend

1. There are no restrictions concerning the usage of programming languages, libraries or frameworks to create the frontend.
2. The frontend is just a stupid user interface, which accesses the application interface offered by your created database functions/procedures.
3. It helps to visualize and interact with the data, by taking usability aspects into account.

2. Task Description
2.2.3 Dummy Data

For demonstration purposes there are available at least:

- 10 film related persons, which are linked to films in different roles, whereat at least
  - 5 persons are linked to a minimum of 3 different films, while avoiding usage of same set of films for 2 different persons
  - 1 person is linked to 1 film with at least 3 different roles
- 10 films, each having at least 1 genre, whereat
  - 3 films are subordinated to 1 film (all to the same)
  - 5 films have at least 3 different genres, while avoiding usage of same set of genres for 2 different films
- 5 users, whereat
  - 1 has rated no films
  - 3 have rated about half of available films, while avoiding usage of same set of films for 2 different users
  - 1 has rated all available films

3. Examination

The examination consists of a 10-minute presentation, which should meet the same criteria as the content of the term paper, but the focus is on technologies and program functionality.

So the project and used technologies should be presented, while UML diagram, relational schema and distributed database discussion can be kept short (no more than one scenario here). The presentation should also include a live demonstration of the project or a demonstration video demonstrating all parts of the practical task. For group work, the presentation time should be divided evenly between both students.

Afterwards, some questions are given, which primary focus on the project and the term paper, but may also cover the lecture and exercise.

Finally, there will be a short consultation and you will be informed about your mark.

4. Dates

- Handout of task description:
  - starting 2022-01-06 16:00 CET (UTC+1)
- Submission of project:
  - until 2022-02-10 16:00 CET (UTC+1)
    - late submissions will be accepted, but your final mark will be reduced by 1/3 per 8 hours or parts thereof
      - your mark will be reduced by 1 per day, so you cannot pass, if you submit after 2022-02-13 16:00 CET
  - via OPAL:

    ➔https://bildungsportal.sachsen.de/opal/auth/RepositoryEntry/20312915968/CourseNode/1617158120350182003
- Oral exam and presentation:
  - starting 2022-02-15
    - until we are done (probably within two weeks, but depending on our time and the number of groups)
  - appointment allocation will be done via OPAL, too
  - examination will take place via video conference in same room like exercise