COS 221 Practical Assignment 5

UNIVERSITEIT VAN PRETORIA
UNIVERSITY OF PRETORIA
YUNIBESITHI YA PRETORIA

- Date Issued: **2nd May 2024**
- Date Due: **30th May 2024** before **09:00 AM**
- Submission Procedure: **Upload to ClickUP**
- Submission Format: **zip or tar + gzip/bzip2 archive**
- This assignment consists of **9 tasks** for a total of **255 marks**.
- There are **10 marks** assigned to group cohesion for this practical assignment.

1 Overview

This last practical assignment in COS221 serves to expose you to projects and activities that will be required of you in the second semester of your second year as well as on the third-year level, especially COS301. This project is a group-based project which will require you to use all of your obtained knowledge in COS221 plus the effort of your group to complete this assignment. This assignment has minimal specifications to allow you to implement all you have learned, but also discover and showcase to the lecturers and yourself what you are capable of achieving as a Computer Science student of the University of Pretoria.

For this assignment, you are welcome to do and implement everything that you think is suitable for this project. This means that you are welcome to do more than what is required, but not less. Where specifications might seem vague to yourself or your group, make the required decisions and assumptions, and add them to your uploaded PDF. Simply put, you are required to use your creativity for this assignment, and working in groups will make it easier to find ideas and implement a good project. This process is your first introduction to what will be required of you in your final year and industry as a University of Pretoria graduate.

2 Project Scenario

Streaming entertainment has revolutionised how people consume media globally. Our fictional streaming service, **Hoop** , aims to take this experience further by offering a curated collection of movies and TV series to viewers worldwide. As part of an initiative to enhance user experience, **Hoop** has partnered with your second-year Computer Science group to develop and implement new features for its platform.

Your task is to create a user-friendly streaming application that not only showcases existing content but also enhances it with additional data where necessary. The platform should allow users to easily discover and filter movies and TV series based on genres, ratings, release dates, and more. Users should be able to view detailed information about each title, including cast, crew, plot summaries, and user reviews.

Furthermore, **Hoop** wants to incorporate social features, allowing users to create profiles, rate and review titles, and share their favorite content with friends.

For this assignment, you can make use of the data which is provided below. You're encouraged to augment this data with realistic mock data or integrate external APIs to enrich the user experience.

- `https://developer.imdb.com/non-commercial-datasets/`
- `https://www.kaggle.com/datasets/dgoenrique/netflix-movies-and-tv-shows`
- `https://www.omdbapi.com/`
- `https://archive.ics.uci.edu/dataset/132/movie`
- `hhttps://huggingface.co/datasets/hugginglearners/netflix-shows`
- `https://github.com/amirtds/kaggle-netflix-tv-shows-and-movies`

- `https://grouplens.org/datasets/movielens/latest/`
- `https://www.episodate.com/api`
- `https://www.tvmaze.com/api`
- `https://trakt.docs.apiary.io/`

3 Outcomes

After successful completion of this assignment you should be able to:

- Analyse and understand data from multiple sources
- Able to curate data
- Design a database schema to be implemented in a RDBMS for the curated data
- Design and build a web-based application and:
  - Be able to execute a connection to an RDBMS from a programming language
  - Query and manipulate a relational database from a programming language
  - Build a Graphical User Interface (GUI)
  - Utilise the GUI to query and manipulate a relational database.

4 Constraints

1. You must complete this assignment in groups of 5 - 7 students (no less and no more). Make sure you register your teams before the provided registration deadline (this will be announced on Clickup).
2. You may ask the Teaching Assistants for help but they will not be able to give you the solutions.
3. The PDF, database dump, source code, GUI, and git history will be marked.
4. The GUI interfaces:
   (a) Which run and perform what they are supposed to do get full marks
   (b) Which run but do not perform as required, will receive partial marks
   (c) Which do not run will be allocated partial marks based on the functionality they would have exhibited.
5. You need to use a RDBMS and tools/languages you require to build a web-based application to complete the practical assignment.
6. You may utilise any text editor or IDE, upon an OS of your choice.

5 Milestones

To be able to successfully deliver and demonstrate the project to the Hoop team on 22nd May 2024, you need to adhere to the following milestones:

- Register your teams from 2nd May till 5th May at midnight.
- Discuss the functional requirements and the (E)ER-diagram, as a group, with your assigned tutor on 8th May.
- Show your database with valid data and some queries to your tutor on 15th May.
- Demonstrate your initial web-based application that integrates with the database on 22nd May.
- Bookings for the team demonstration will open on 28th May.
- Demonstrate your final project on 30th May.

6 Submission Instructions

You are required to upload a single archive that includes the following files:

- An archive containing your web-based application. If you used a management tool you are required to mention it.
- A pdf containing the answers to the tasks.
- A file or files containing the SQL statements from your database dump to:
  - Create your database,
  - Create the tables in your database
  - Populate the tables with the data you populated the tables with.
- Your archive containing your `.git` folder
- A **readme.txt** file informing the marker what they should do to build and execute your application.

Upload your archive to ClickUP. No late submissions will be accepted, so make sure you upload in good time.

7 Online resources

- Git: `https://git-scm.com`
- GitHub: `https://github.com`
- PHP: `https://www.php.net`
- MariaDB: `https://mariadb.com`
- Composer: `https://getcomposer.org`
- Getting Started with MariaDB at: `https://mariadb.com/get-started-with-mariadb/`
- Platform for developers to learn, share knowledge and build career: `https://stackoverflow.com/`

8 Assignment Instructions

Task 1: Research ..... (20 marks)

Conduct research on the entertainment industry, focusing on movies and TV series. Include references and limit your research to one page. Cover aspects such as industry trends, audience preferences and popular genres

Task 2: (E)ER-Diagram ..... (30 marks)

After you have conducted your research in Task 1, you are now required to construct an (E)ER-diagram of your project and provide the final model in your uploaded PDF. You are required to mention all assumptions you have made during your modeling and any other information you deem necessary. If you made multiple iterations to get to your final (E)ER-diagram, be sure to include all iterations, as well as notes on how each iteration improves on the previous.

Task 3: (E)ER-diagram to Relational Mapping ..... (30 marks)

Apply the steps for converting your (E)ER-diagram into a relational model. Provide a relational mapping of your (E)ER-diagram in your uploaded PDF. Be sure to indicate the conversion for each step, as well as the assumptions you have made and the choice you have made if the conversion could provide multiple solutions.

Task 4: Relational Schema ..... (30 marks)

Design a relational database schema based on your mappings obtained in Task 3 in the form of both a visual diagram and SQL statements targeted towards MariaDB. Ensure that you include, where applicable, all of the following:

- Primary, Secondary and Foreign Keys
- Constraints and Checks
- Data Types and Length Constraints

Task 5: Web-based Application ..... (50 marks)

In COS216 you learned how to create a web application using PHP, HTML5, CSS, and JavaScript. For this task, you are required to implement a web application to manage and perform different queries you find necessary for this project. Your web application must at a minimum be able to:

- Login and manage users
- Add, edit, and delete Movies and TV series
- Manage actors, directors, genres and production studios
- Sort and filter content based on various criteria
- Update the database and/or extend, delete, etc.
- Recommend Movies and TV series based on a criteria

Task 6: Data ..... (20 marks)

To ensure your web application has sufficient data for testing and demonstration purposes, you are required to populate your database with a substantial amount of Movies, TV series, Actors, Directors, Production Studios and Review information. You are welcome to augment the given data and include all your relevant data in your database by hand, a script that creates artificial data, a script that parses an API, or loading the data with XML files from some website/feed/API. Explain your data population method and reasoning in the uploaded PDF.

Task 7: Analyse and Optimise ..... (10 marks)

You are required to make use of your chosen RDBMS and tools to analyse at least one of your query execution plans and report on the performance to execute provided queries in your uploaded PDF. Thereafter you are required to explain how you would optimise your query in your PDF, implement your proposed optimisation and report on the performance gains/losses achieved. You are also required to explain why you believe the gain/loss in performance was observed.

Hint: Ensure that you have sufficient data in your database, or else you might not observe the gain/loss.

Task 8: Development ..... (25 marks)

Your project will be assessed, not only on the functionality it provides but also on your overall development practices such as:

- Usage of git
- Data validation techniques
- Utilisation of a package manager
- Ease of understanding, spelling and grammar, and structure in your code, git commit messages, uploaded PDF, README etc.
- Quality of the overall delivered solution

Task 9: Demo ..... (40 marks)

Your team will be required to demo your solution to the lecturers of COS221. Booking slots will be opened closer to the time. Note, that all team members are required to be present for the demo to receive marks. In addition, each team member has to explain in the uploaded PDF all their contributions to the project. The demo will be evaluated based on functionality, user experience, presentation and individual contributions.

IMPORTANT NOTE(S):

- Please refer to the rubric for the detailed allocation of marks.
- Plan your study time and start well in advance with this practical.

9 Rubric for marking

Research	**20**
General overview and explanation	4
Different types or genres of Movies and TV series explained	4
Information on how content is rated or categorised	4
Additional features/information provided (Recommendations, User Reviews)	4
References	4
(E)ER-diagram	**30**
Entities and Attributes	10
Complex and Derived Attributes	10
Relationships and Cardinality	10
Mapping	**30**
Regular Entity Types	3
Weak Entity Types	3
1:1 Relationships	3
1:N Relationships	3
M:N Relationships	3
Multivalued Attributes	3
N-ary Relationships	3
Specialisation and Generalisation	3
Unions	3
Correctness	3
Relational Schema	**30**
Visual Representation	10
Primary, Secondary and Foreign Keys	5
Constraints and Checks	5
Data Types and Length Constraints	5
Correctness	5
Web-Based Application	**50**
Functional Requirements	10
Media Management Features	10
User Account and Profile Management	10
Sorting and Filtering Options	10
Queries	10
Sample Data	**20**
Explanation of Data Sources	5
Data Entry Methods (Script or Manual)	5
Quality and Relevance of Data	10
Analysis and Optimisation	**10**
Explanation	2
Optimisation	3
Interpretation	5
Development	**25**
Version Control (e.g. Git)	5
README Documentation	5
Overall Quality and Impression	15
Demo	**40**
Functionality and User Experience	20
Presentation Quality	20
Group Cohesion	**10**
Total	**265**