conditions

● Design the solution to this problem using the OOP paradigm.

● The application should be implemented in C ++ with a layered architecture.

● The application must provide a console user interface.

● Please provide tests for non-trivial functions.

● Please provide specifications for your functions.

● All entries are stored in an array in the repository. Please add

Add at least 10 entries in your repository (in the code).

● Add a simple data validation (eg a validation class).

● Please handle the following situations:

○ If an entry already exists, a message will appear on the screen

output. In this case the entry is not saved. You need to

decide what makes an entity unique.

○ If the user tries to delete an entity that does not exist

output a message on the screen. No impact will be

considered on the entries of the repository.

problem

So many films, so little time! In order not to miss any good films, you absolutely need

a software application with which he / she can manage films and watch lists. The

Application can be used in two ways: administrator and user. If the

Application is started, it offers the possibility to select the mode.

Administrator mode: The application stores all films in a database (the repo).

The application allows the admin to update the database. That means: a new one

Add a film, delete a film and edit the information of a film. Every film

has a title, genre, release year, number of likes and a trailer.

The trailer is saved as a link to an online resource. The admin also has the

Possibility to watch all films in the database.

User mode: A user can create a watch list with the films they watch

would like to. The application allows the user to:

Viewing films in the database with a specific genre. If that

Genre is blank, all the films on the screen are displayed one by one

spent. If the user selects this option, the data of the

Lab: 5

Date: week 9

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

first film (title, genre, year of release, number of likes) and the

Trailer is run in the browser.

2. If the user likes the trailer, he / she can watch the movie from his list

Add.

3. If the user is not satisfied with the trailer, he can continue to the next one

go. In this case, the information corresponding to the next film is displayed

and the user is offered the option to add it to the watch list.

This can continue as long as the user wants.

4. Deleting a movie from the watch list (after the user has watched the movie

viewed). When deleting a film from the watch list, the user can use the

Rate the film too (with a like). In this case, the number of likes of the

Films in the repository increased.

5. Viewing the watch list.