Distributed Systems Programming

A.Y. 2025/26

Exam Assignment for Exam Call on 19/01/2026

Deadline for submission: 15/01/2026 12:00 CET

Modify the REST APIs of Laboratory 1, by introducing the feature of acceptance of review invitations, according to the following specifications:

1. When a film owner invites a user to review a film, the invited user must explicitly accept the invitation before being enabled to make the review (i.e., to update it). While the invited user does not express acceptance, the review invitation remains in a pending state, which prevents the invited user from updating the review.
2. The issuer of a review invitation must be able to specify the maximum time allowed for the user to accept it. When this time expires, the review invitation is automatically cancelled by the system, and it becomes invisible for the invited user, but not for the user who created it.
3. The status of all the review invitations sent for a film, including their expiration time, must be visible, but only to the film owner who created those invitations. In particular, the film owner must be able to know if any review invitation is in the pending state, or if it has been accepted by the invited user, or if it is cancelled (i.e., expired). To avoid the transfer of unnecessary information, the film owner should also have the option to select the information to retrieve (e.g., only invitations in the pending state, if any).
4. Invited users must be able to retrieve their pending invitations, select what to retrieve (to avoid transferring unnecessary information), and accept all their pending invitations (in a single operation).

Modify the implementation of the service so that it implements the modified REST APIs. Of course, the database must also be extended accordingly. Note that it is not required to modify the REACT client. Test the modified REST APIs by using a client like Postman.

Submit the updated solution, including all the following items:

- All the json schemas
- The full Open API documentation of the REST APIs, including examples of JSON documents to be used when invoking the operations, and examples of invocations of the API operations, possibly as a Postman collection
- The REST APIs implementation code

- The database file, including some sample data
- README.md files that specify the contents of folders, the instructions on how to run the code from scratch and the main design choices made.

Important:

- Organize the project files as in the solution provided for Laboratory 1.
- The solution must work within the Labinf Ubuntu machines, with the software already installed in those machines.
- The solution will be tested by running it in the Labinf machines.
- The solution must be uploaded to a git repository for which you will soon get the credentials.