Distributed Systems Programming

A.Y. 2022/23

Exam Assignment for Exam Call on 24/02/2023

Deadline for submission: 22/02/2023, 12:00 CET

Modify the solution of Laboratory 5, by extending the use of MQTT according to the following specifications:

- In the proposed solution of Laboratory 5, a request for film review update was accepted only if the *completed* property was set to true. For this exam assignment, that constraint must be relaxed, allowing a film reviewer to partially update a review without completing it (e.g., a reviewer may set a certain value for the rating with a first request, he may set a textual review description with a second request, he may change the rating to a different value after changing his mind with a third request, and so on).
- Whenever any modification to a review for a public film is performed by means of the REST APIs (e.g., modification of the rating or of the textual review description), the server must promptly inform all *interested* clients by means of MQTT messages. These messages must be JSON objects that carry the full information about the occurred change.
- The clients must be able to express their interest in receiving the MQTT messages related to the review updates of specific public films of their choice, or even of specific reviews.
- The REACT clients must be programmed so that:
    1. If the page that is currently visualized in the browser is displaying some review data that are updated and that refer to a public film for which the logged in user is a reviewer (e.g., when the review with ID 7 is updated the client where the logged-in user has ID 1 is displaying the page having URI http://localhost:3000/public/7/reviews), then the contents of that page must be automatically updated without any further interaction with the web server. For example, if only the textual description of a review has been updated, only that field must be updated on that page.
    2. If the page that is currently visualized in the browser is displaying some review data that are updated and that refer to a public film for which the logged in user is not a reviewer, then the contents of that page must not be updated.

Design the MQTT topics and messages according to the guidelines studied in the course. If you want, you can extend/modify the design proposed for Lab 5, making sure that, in the end, everything works. Then, implement the designed solution by modifying/extending both client and server implementations.

Submit the updated solution, which must have the same folder organization as the solution of Laboratory 5, including all the changed items and at least the following additional items:

- A README.md file in the main folder. The file must include an explanation of the design choices made about the MQTT topics, messages, and configurations (e.g., retention policy, QoS, etc.).
- The files containing the schema(s) of the messages used over MQTT (in the Server/json_schemas folder). Please refer to these files in the explanation.

Important:

- The solution must work within the Labinf Linux machines, with the software already installed in those machines.
- The solution must be uploaded to a git repository for which you will get the credentials.