Distributed Systems Programming

A.Y. 2023/24

Exam Sample

This is a sample of a possible exam assignment, to be completed within 12 days from assignment. IT IS NOT THE REAL ASSIGNMENT.

Extend the solution of Laboratory 1, according to the following specifications:

1. Introduce a precedence relation on reviews: a review can precede other reviews and other reviews can precede it (many-to-many relation). If two reviews A and B are bound by this relation, i.e., review A precedes review B, then review A must be done before review B, i.e., review A must be completed before review B can be completed. One consequence is that the reviewDate of review A must be strictly before the reviewDate of review B, when both reviews have been completed. Another consequence is that review B can be marked as completed only if review A is also marked as completed. These constraints must be checked by the system when review creation and update operations are done (for example, if a client tries to modify a review in a way that invalidates these constraints, the update must be refused by the system; the same must happen if a client tries to create a review that does not respect these constraints, because, for instance, it should precede another review which has already been completed). The system must also check create and update operations and refuse them if they would introduce circularity in the precedence relation (a review cannot directly or indirectly precede itself).
2. If a review is removed, the precedence relation must be updated accordingly. For instance, let us assume review A precedes review B and review B precedes review C. If review B is removed, after its removal review A must precede review C.
3. In the REST API, provide a way for a reviewer to retrieve all the due reviews that are not preceded by other uncompleted reviews.
4. In the REACT client, provide ways for the user to create reviews that must precede or follow some other reviews, and to get the reviews that are not preceded by other uncompleted reviews.

Submit the updated solution, including all the following items:

- The json schemas

- The full Open API documentation of the REST APIs
- The REST APIs implementation code
- The database file, including some sample data
- The code of the REACT client
- README.md files that specify the contents of folders and instructions on how to run the code from scratch

Important:

- Organize the project files as in the solution provided for Laboratory 1
- The solution must work within the Labinf 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