SCALE FOR PROJECT

JAVA / DAY 09

Introduction

The methodology of School 21 makes sense only if peer-to-peer assessments are done seriously. This document will help you to do it properly.

- Please, stay courteous, polite, respectful and constructive in all communications during this assessment. The bond of trust between community 21 and you depends on it.
- Highlight possible malfunctions of the work done by the person and take the time to discuss and debate it.
- Keep in mind that sometimes there can be differences in interpretation of the tasks and the scope of features. Please, stay open-minded to the vision of the other.

Guidelines

- Evaluate only the files that are on the GIT repository of the student or group.
- Doublecheck that the GIT repository is the one corresponding to the student or the group as long as to the project.
- Meticulously check that nothing malicious has been used to mislead you and have you assess something except the content of the official repository.
- If you have not finished the project yet, it is compulsory to read the entire instruction before starting the review.
- Use the special flags in the scale to report an empty or non-functional solution as long as a case of cheating. In these cases, the assessment is completed and the final grade is 0 (or in a case of cheating is -42). However, except for a case of cheating, you are encouraged to continue reviewing the project to identify the problems that caused the situation in order to avoid them for the next assessment.
- You must stop giving points from the first wrong exercise even if the following exercises are correct.

Attachments

- The exercises

Preliminaries

Respect the rules:

- The repository contains the work of the student (or group).
- The student is able to explain their work at any time during the assessment.
- The general rules are respected throughout the assessment.

Yes | No

Exercise 00 - Registration

1. Архитектура серверного приложения соответствует приведенной в задании?
2. Все компоненты являются бинами контекста?
3. Репозиторий реализован через JdbcTemplate?
4. Для хеширования паролей используется PasswordEncoder?
5. После прохождения регистрации зашифрованный пароль успешно сохраняется в БД?

Yes | No

Exercise 01 - Messaging

1. Функционал приложения позволяет пройти жизненный цикл, указанный в задании?
2. Каждый запущенный клиент получает отправленные сообщения из других клиентов?
3. Все сообщения сохраняются в базе данных с учетом необходимой информации?
4. Предусмотрен выход из приложения путем ввода сообщения Exit?

Yes | No

Exercise 02 - Rooms

1. При создании комнаты пользователь имеет возможность задать название этой комнаты?
2. Пользователь получает последние 30 сообщений в случае повторного входа в приложение?
3. Сообщение, отправленное в рамках какой-либо комнаты доходит до остальных пользователей этой комнаты?
4. Обмен сообщениями происходит через формат JSON?

Yes | No