

[← Project evaluation «Day 09. Sockets»](#)

## Git project

```
ssh://git@repos-ssh.21-school.ru:2289/students/Day_09._Sockets.ID_223243/java1acarmela/Day_09._Sockets-0.git
```

## Student



java1acarmela

## About

## 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
- 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

## 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 well as to the project.
- Meticulously check that nothing malicious has been used to mislead you and have you assessed 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 well as a case of cheating. In these cases, the assessment is complete.
- 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
- You must stop giving points from the first wrong exercise even if the following exercises are correct.

## Preliminaries

[Dashboard](#)[Calendar](#)[Progress](#)[Projects](#)[Activities](#)[More](#)

- The student is able to explain their work at any time during the assessment.
- The general rules and the rules of the day are respected throughout the assessment.

## MAIN PART

## Exercise 00 – Registration

- Does server application architecture match the one described in the task?
- Is it true that all components are bins of the context?
- Is repository implemented via JdbcTemplate?
- Is PasswordEncoder used for password hashing?
- Is encrypted password successfully saved to DB upon registration?

## Exercise 01 – Messaging

- Does application functionality enable to complete the life cycle specified in the task?
- Does each running client receive messages sent by other clients?
- Are all messages saved in database considering the required information?
- Is it possible to exit the application by entering an Exit message?

## Exercise 02 – Rooms

- When creating a room, can a user specify its name?
- Does a user receive last 30 messages when re-entering the application?
- Does a message sent inside a room reach other users in this room?
- Is JSON format used to exchange messages?

## Feedback

Fails 

## Comment

 Duration

 Passed checks

Git project

About

**Preliminaries**

MAIN PART

Feedback