



# Testing Social Network

Ensure high quality of the platform

## Project Description

Your task is to verify that the delivered software meets the user requirements and is production-ready and to ensure stability by providing automated tests.

## Functional Requirements

### Planning

You should prepare a plan with estimates of the time and effort needed for each activity.

### Test Plan

Describe all activities needed to verify the quality of the product. Follow the templates and guides covered in the materials from the lectures.

### Test Cases

Develop high-level test cases that will verify a required quality level and guarantee the customer requirements coverage.

### Execution

#### Manual

Conduct exploratory testing for the entire system and cover main functionalities and happy paths. Create a coverage report that shows which cases are executed and with what result. Make sure to prioritize all scenarios.

#### Automation

Create automated suites that cover at least the happy paths. Make sure to have reports of each execution so you can provide info on how many tests pass/fail.

### Reporting

## Technical Requirements

- You should create functional tests covering the UI using Java + Selenium
- The tests covering the middle tier should be created using:
  - Postman
  - Java code
- Use GitHub as a source code management tool.
- Use a bug-tracking tool to log issues (JIRA/Xray/GitHub/etc.)

## Optional Requirements (bonus points)

- Non-functional testing

## Deliverables

- Test plan
- Test report
- Automated tests code/repository link
- Script for execution of automated tests
- A document with the required prerequisites to run the tests
- A document with a concise description of how to run the tests, filter them, etc.
- High-level test cases

## Public Project Defense

Each student should present (the presentation should not take more than 10 minutes, 30 minutes all):

- The functionality they tested, and the main scenarios covered.
- The automated tests created by them, and the specific functions needed for that.
- A sample run of a single test/test suite.
- A report with the testing activities performed (e.g. tests coverage and results, issues logged, issues fixed by priority/severity, created automated tests).

## Expectations

You **MUST** understand the project you have created.

Any defects or incomplete functionality **MUST** be properly documented and secured.

It is OK if your project has flaws or is missing one or two requirements. What is not OK is if you do not know what is working and what is not and if you present an incomplete project as functional.

Some things you need to be able to explain during your project defense:

- What are the most important things you have learned while working on this project?
- What are the worst “hacks” in the project, or where do you think it needs more work?
- What differently would you do if you were doing the project again?