Group 43: Meeting 3

Opening: 16.45	Chairman: Sem
Present: Everyone except Nivard.	Secretary: Ceren

Opening:

Chairman explains to TA what we've done since last week. We made our first working client-server communication. Everyone in the group took its copy and ran it on their own computer. We used Gradle and Spring. Chairman informs TA about the uploading file which includes the overview of APIs. Then, the chairman gives details about the brainstorming we did on what should be our next steps, how should we develop our code and which properties should we build firstly.

We also had four questions:

- There is a 'Add license issue' in issues. What does it mean?
  Answer: Every code you upload to gitlab should have a license. Go through licenses and choose whatever you like. TA suggests us this website: choosealicense.com (the most common one is MIT). Then gives instructions about how can we add the license to gitlab.

- Question about gitignore
  Answer: Gitignore only ignores staff after you add it. TA gives feedback about gitignore.

- Question about commit graph

- CI question. Do we get point for CI?
  Answer: When you setup CI you see the code coverage for each build. Showing code coverage is mandatory but CI is an extra point.

TA also had given us many suggestions:

- Scrum board / Done and closed are the same. Issues should be more and explaining. We get an extra point for scrum board

- Using merge / Chairman asks what is the advantage of using merge?
  Answer: Bigger teams and online communities generally use merge request. In merge request, every time you open a branch you push it to gitlab and other members can give feedback to you. With this way, you can try to find some improvement points. We get an extra point when we use merge requests.

- CI (Continuous Integration) / TA says that you don't have to use it. It is extra but it has advantages. You continuously see the state of your project. You can determine what the environment does. You can see what is the situation of your checkstyle, code coverage, warning levels. Is your test still pass? You get the notification. Then TA explains how can we use it.

- JavaFX / TA says that you need more specific libraries

- Directory / TA suggests us to clean up the directory. Because it doesn't look clear. We can throw most of them to doc folder. It should be clear.

DEMO TIME: We showed our first demo (working server-client communication) and have done it.

TA suggests us to create small groups working on different areas of the project. Because 6 people can be difficult to work on the same project at the same time. Therefore, TA suggests us to have client team, server team, data team, and GUI. We already have teams.

Next Steps:
- JavaFX (GUI)
- Database
- Authentication (server sees connections, basic username, and password)
- Once logged in some unique codes should be given
- Whole user structure should be built

TA'S SUGGESTIONS FOR THIS WEEK:
- The database, authentication, GUI, experience with GET, POST, PUT etc.
- Searching libraries in Spring which are useful for us
- Question about database and TA gives suggestions

NEXT DEADLINE(18 March): full workflow "Eating a vegetarian meal" (click button to register meal, send a request to the server, store it and persists it.

End meeting at 17:42

TODO LIST
- Add license issue
- Fix some points in Scrum board
- Clean up directory
- Using merge requests
- Start to work on next deadline
- Add agenda for next week
- Search and start to use CI (as an extra)