

# Group 43: Meeting 2

Opening 15.48

Present: Everyone except Nivard.

Chairman: Daan

Secretary: Merel

Opening:

Chairman explains to TA what we've been doing last two hours. We did some brainstorming on what the code should look like, what should go on the server side and what should go on the client side.

We also had two questions:

-Commit changes issue – Daan → Git ignore

-Is there a template for the meeting notes? → Just Google.

The TA tells us that we should focus on doing research first. Monday we need to have a working client-server communication and by then we should have figured out which libraries to use and what kind of API we want to make, for example do we want a strict or a non-strict API. We need to get the basic architecture ready.

Think about what kind of information you're going to be interchanging between the client and the server. This will influence some choices you make throughout the project. API's almost always send JSON (Almost default).

We might want to use JavaFX, but still have to do some research. None of us has any experience with making a Java application.

The client and server can use different libraries. You do need to have a client-server contract to make sure communication runs smoothly. Client libraries are more general, server libraries are more specific. Spring framework has everything (sadly, including a big learning curve). Smaller libraries are easier to work with in general.

TA: To decide on libraries it's best to do some tutorials and see what fits you and your project best.

**TO DO:** Research libraries to use. Do research about Java APIs and Java servers. Also do some tutorials to get the hang of it. Probably focus on JavaFX, since that is recommended by a lot of people. – Everyone

The TA put some issues on GitLab, including how to make a local git-ignore. We should look at those. The error Daan got when running his code was because his compiler files weren't in his git-ignore.

Issues are mainly used for inbuild documentation/discussion while you're coding. It helps you to see why you made certain decisions. You can for example use it when you encounter bugs in code. You give the issue a name (Including 'bug') and a description saying what the expected behaviour is and what the actual behaviour is. You can also tell your teammates why you think the code does something else than you expect. Then you use the comments on updates.

**TO DO:** Look at the issues the TA put on GitLab and make your local git-ignore.

We get extra points for working with Scrum. The TA made an issue about this. Since Scrum is not hard to use and can even make a project easier to oversee, we might want to consider using Scrum. GitLab has a Scrum feature which apparently works well with issues. Usually a Scrum board from developing team looks like this: Long term backlog, Sprint backlog, doing, testing, done. You divide tasks and put them on the right board: Does this have to be done immediately or is this a task for in the future? This way you can easily see if you are on track with your tasks for the next deadline (Sprint backlog).

**TO DO:** Get acquainted with Scrum in GitLab. – Everyone

**TO DO:** Make and upload next agenda by Sunday. – Next chairman (Sem)

**DEADLINE:** Monday: Have a working client side and server side with communication. – Everyone

End meeting at 16.19

### ***Brainstorming we did before the meeting:***

Client side:

1. User – Username, password, list of friends, score, list of activities/history, leader board.  
Methods for user: Add activity, add friend, remove friend, (remove activity), set username, set password, login → Login is hard to do, focus on easier stuff first.

Server:

1. Global leader board
2. Abstract class activity: Name, score, description
3. Child classes activities: Name, score, description
4. Level up/ranks?

Bad activities? Badge: At least you're honest.

Tutorial!!

Client side team:

Kristin  
Mika  
Sem

Server side team:

Daan  
Merel  
Ceren

Homework:

Everyone: Find out things about JavaFX + ~~Write user object~~

Everyone: Get acquainted with Java servers and set up server begin.

Upload agenda to GitLab on Sunday.