Users and Connections Tool

Algorath Test

Jorge Moreno Peralta

General requirements

1. *The user can see all the users that are registered in the system.*
2. *The user can add new users to the system providing their names and their Id numbers.*
3. *The user can connect two users who belong to the system.*
4. *The user can see all the connections that involve a given User. The user chooses a user from the system, and the system shows all the connections belonging to the given one.*
5. *The user can remove a user from the system.*
6. *The system shows some stats about the connection of a given user. It is possible to see the average number of connections per user and the percentage of users that have fewer connections than the chosen one.*
7. *The system uses a database to store all the data from the users and the connections of the system.*
8. *When a user is deleted, the system automatically removes all his connections from the system.*

Technologies.

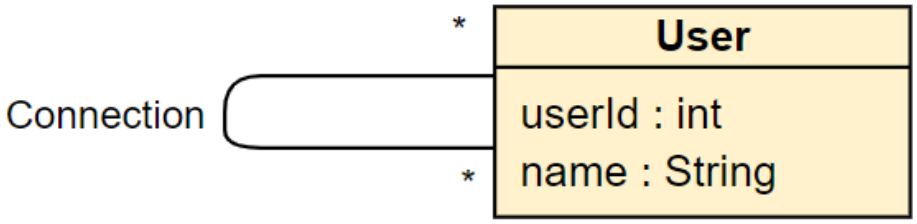
The technologies applied to elaborate this project are:

- Java v15 and Eclipse.
- MySQL and MySQL_connector to work with the database.
- JavaFX and WindowBuilder to elaborate the graphical part of the Software.

Application Structure.

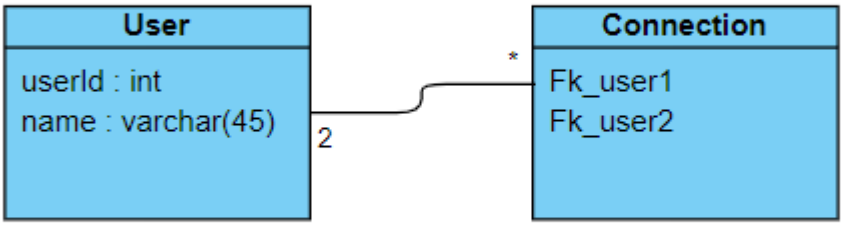
The application consists of a simple system that keeps track of users and their connections.

We can see a diagram with the structure of the system which is quite simple.



We can see how the class *User* is the main class that encapsulates all the behavior of the system. We have some additional classes used to connect to the database and manage exceptions.

However, we can see how we store two different tables in the database to manage the users and the connections.

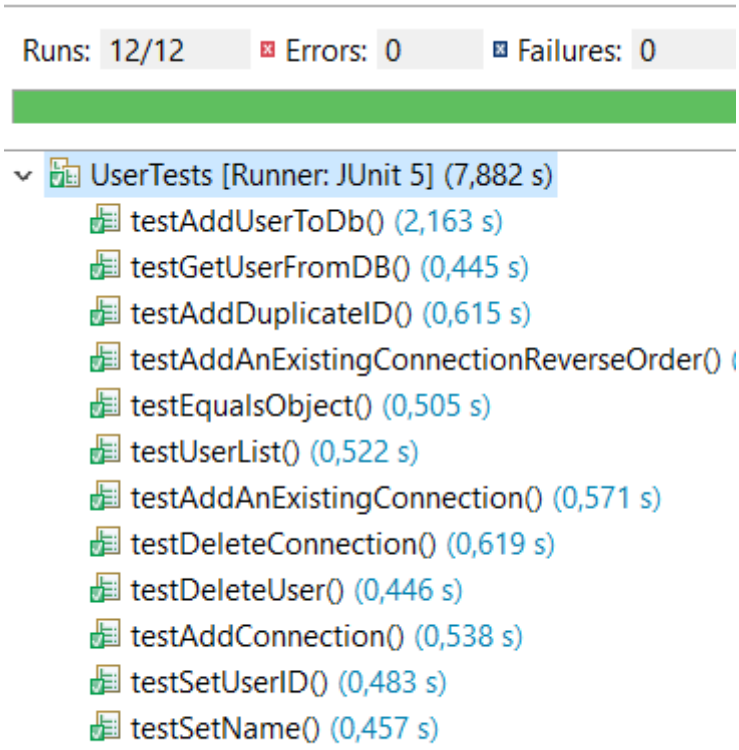


Tests

The system provides a group of tests created using Junit5 to check the correct behavior of every part of the system. We have tests to check the following aspects:

- How the user is properly added to the database.
- How we can get a user from the database when giving his user Id.
- How the system can connect two users that were no connected and the database reflects this change.

- How the system throws an exception when trying to add a user with a user id that belongs to another user who is currently on the database.
- How the system throws an exception when trying to add a connection between two users that are already connected.
- How the system throws an exception when inserting an existing connection in reverse order. (Connect A to B when B is connected to A)
- Check that two users are equal to the system when they have the same number Id.
- Check how the system is able to list all the users from the database.
- How the system can delete an existing user and it is no longer on the database.
- How the system can delete an existing connection and it is no longer on the database.
- How the system is able to obtain and modify the information of a user on the database.



Finally, the project also provides five different interfaces to settle the communication with the users. These views are made with the tool WindowBuilder, and the behavior is managed by java code.

Deployment instructions

To deploy the application is necessary to have the following tools:

Java v15 and a version of Eclipse that can run it.

MySQL technologies and the current database, whose script is added to the project.

The MySQL_connector is essential to access the database. It is added in the referenced libraries, the version used is the 8.0.21.

To run the tests, it is also necessary to have installed Junit 5. Probably it will generate some errors with previous versions.

When all these tools are installed, the final steps are to decompress the solution, import it to eclipse, and run the project.