**Acceptance Test- netEval**

**1. Database**

a. Check that the server is active:

• Check that we receive a response from the server in the relevant windows.

• Make sure the data is stored on the server efficiently and logically.

**2. Login**

a. Make sure the user is able to connect to the system:

• Enter the username and password with which we signed up and check that the login was successful in 5 seconds.

b. If this is not the first time then the user automatically connects to the software:

• After initial login we will exit the software again and check that the software has gone up without further login.

**3. Register**

a. Make sure you can sign up for the system-

• Enter details and press the sign button and see that the software has taken us to the home screen.

**4. GUI**

a. You can upload a file to the software and the software will load the relevant fields in the network data-

• Click the "Upload File" button and check that the data appears in the fields within 5 seconds.

• Willingness - We will create a network manually and all data in the fields appear to be calculated correctly as it came out manually.

third. Make sure the shared networks list works properly -

• We will add permissions to a user and check that they are listed.

• Check that when clicking on the username, a window appears showing the distribution of the user's network value.

**5. Permissions**

a. It will be possible to add a user to the list of permissions:

• give them permission or remove and check that the appropriate permission was updated. Log in to the "Add permissions" menu We will add permissions to a user, check that the permissions have been added to them, and then remove the permissions and check that they no longer have permissions.

**6. Network Evaluation**

a. Check that the system gives a reasonable value to the network and do it at the same time on several different networks:

• Taking one network we will review its valuation several times and the results appear to be coming close to each other.

• Take two different grids of the same size run the test on them and the value given to them seems to be similar.

• Take a big network and make sure the network valuation didn't take more than a minute.

• check that the code of the algorithm is clearly written.

b. Check that the value of the network is updated in the database:

• After calculating the value of the network, we will enter the distribution of the value of the network and it appears that the data has been updated on the graph.

• The net worth can be seen over several months. Going into the distribution of the net worth and the old data appears to be in the graph and accurate.

**7. Graph**

a. Check that the software shows the graph clearly and friendly.

• Go into the graph and it looks good.

b. Make sure the graph is displayed according to the network data.

• Take a small sample manually. We'll go through all the users, groups, contexts, etc. and check that everyone is on the graph correctly and accurately.

 • Take a big grid and check that the graph does not take more than a minute.

c. Each group is drawn in a different color on the graph:

• Manually we will move the vertices by their type and each type appears to appear in a different color on the graph.

• The name of the contact is written on the bow. Manually we will go over every relationship between two vertices and check that it is correct.

• If you click on a node you can see the name of the node.

• Check that network filtering works according to connections.

**8. Data Analysis**

a. The software displays the data distribution according to what the user has chosen:

• Take a small example of a network and for each distribution we will check that the data it contains is correct.

**9. My Account**

a. The user sees their user data:

• we will log into "My Account" and it appears that the username, network name appears there according to the data with which we have registered.

b. The user manages to update his data:

• We will update the password and check that the password has been updated.

**10. Disengagement**

a. The user manages to disconnect from the software :

• Disconnect from the software and check that we have reached the login screen.

• We will check that when we log in again after logging out we will see the login number.

• We will check that the identifier is deleted.

**11. Exercise quality**

• We will check that the code is readable, understandable, divided into departments, meaningful variables.

• The code works according to the SOLID principles. Simple code.