**Use cases for module User Service:**

A new user registered:

1. Actors: Guest
2. Pre-conditions: none
3. Parameters: username, password
4. Actions:
   1. Guest: insert a username and a password and click “register”
   2. System: check for an available ID and if the name already exists and create a new user instance
5. Post conditions: the user will see the line “registered successfully” or “name already used”.

Acceptance Tests:

|  |  |
| --- | --- |
| Positive | Insert a name that doesn’t exist and a password |
| Negative | Insert a name that already exists and a password |
| Alternative | - |

A guest logged in:

1. Actors: Guest
2. Pre-conditions: none
3. Parameters: username, password
4. Actions:
   1. Guest: insert a username and a password and click “log in”
   2. System: checks for matching username and password-hash, in case of success checks the permission that user got and for each permission the system will give proper cookies that contain the information that is stored at the system, add relevant authentications
5. Post conditions: the user interface will be updated to contain all the relevant information that relates to the user, the Guest actor will be changed to User.

|  |  |
| --- | --- |
| Positive | Give a username and a password that are stored in the system |
| Negative | Give a false username/false password |
| Alternative (Negative) | A logged in user pressed login (can happen due to double clicking before the response arrives) |

A user logged out:

1. Actors: User
2. Pre-conditions: none
3. Parameters: none
4. Actions:
   1. User: press log out
   2. System: remove relevant authentications
5. Post conditions: the user interface is set back to guest interface, change Actor type from user to Guest.

|  |  |
| --- | --- |
| Positive | A logged in user pressed logout |
| Negative | A logged-out user pressed logout (can happen due to double clicking before the response arrives) |
| Alternative | - |