Testing

General part

- Explain why it is important to have a isolated test data (e.g. separate database) to run your tests on.
- Explain the concept of mocks and the benefits of using mocks in unit testing and provide practical examples from this or previous semesters

Practical part

Unpack and open the project found in *Testing.zip*[1]. This is an alternated version of the seed used in this semester.

Tasks:

Frontend: Factory and Controller testing

Look at view 5:

- Test the factory that fetches users using the $httpBackend
- Test the controller that displays users (consider writing a mock for the factory)

Backend: Rest api testing

Implement the tests necessary to:

- Test that the user “user” cannot access the new endpoint
  - Example of how to grab the token is provided in the test package
- Test that we get the right data from the endpoint when logged in

If you have time:

- Make a new rest endpoint in rest.User on the path(”/user”) returning only the username of all the users stored in the façade.
- Test the endpoint using Rest-Assured
- Make a new view in the application (view6) that utilizes the newly created endpoint.
- Create a controller/factory to fetch data from the end point created above

[1] Can also be downloaded at this git link: https://github.com/sofusalbertsen/test-exam-test