

### 3. Milestone project requirements PA165

#### Deadline:

Project done: **Friday 18. 12. 2015 23:59**

Evaluation submitted: **Tuesday 22. 12. 2015 23:59**

#### Project requirements:

- User interface
  - To start the web application you have to configure maven plugin (tomcat7 or jetty). It is very important that it's super easy to start the web application from command line. So for example using tomcat7 plugin, the following sequence of commands should start the web application: "mvn clean install && cd web && mvn tomcat7:run"
  - The web application must be available on the following HTTP context <http://localhost:8080/pa165>
  - Your application should use in-memory database. This means that after application restart (killing web container and starting it again with mvn tomcat7:run) the data may be reset.
  - Implement the user interface using Spring MVC or Angular JS. Spring MVC is preferred. Angular will be covered at the last minute so it might not be the best choice for you if you are not familiar with it already.
  - The user interface should allow to carry out all business functions of your system.
  - You should fill in all the necessary data automatically. So for example you can use Web Listener to load data during Web Application startup.
  - Make sure there are validations implemented on user interface.
  - Your user interface should use either Facade layer or REST layer to access the system. Do not directly access database and do not directly use Service layer.
  - The web interface layer may reside in separate maven module (if this is helpful).
  - Each member of the team must implement (mostly independently, without copy-pasting) part of user interface. Including controller and view.
  
- REST Layer
  - Your application should have a basic REST interface.

- At least one entity and operations on that entity must be exposed.
- This is mainly to demonstrate you can implement this, it's not necessary to have all application functions accessible through this interface.
- The REST must be accessible at <http://localhost:8080/pa165/rest>
- it is not required to have the interface secured.
- You should include a README file with instructions how to test the REST interface (e.g. CURL commands for command line)
- Security
  - There should be at least 2 roles in the system (e.g. Administrator, User). Each role should have some differences in user interface or in capabilities.
  - There should be login form (not HTTP Basic)
  - Registration is NOT required. You can prefill the users and their passwords in the database.
  - Password should not be saved in the database in open form.

### **Evaluation Checklist:**

1. The maximum number of points that you can assign for this milestone are 10 points;
2. Make sure to test the application mainly from the user perspective. Your evaluation report should list bugs in the system. This buglist will be used during final defense to make sure they are not present anymore;
3. -1 points if it is not possible to compile the project and run tests using "mvn clean install";
4. -1 points if there are test failures;
5. minimally -1 point for each of the unmet project requirements;
6. -1 point for each occurrence of a team member not contributing enough;
7. If a team member was not contributing at all, or very little, you should explicitly say this in your evaluation report. Tutor will follow up on this information;

### **Notes:**

Up to 5 additional points will be given by your seminar tutor based on the quality of the submitted project evaluation report, so the final maximum scores one group can get in this milestone are 15 points;