Final Test December, 21, 2016

Authors: João Costa Seco, Jácome Cunha
Notes: Students can use one (1) handwritten sheet of paper, no printed copied are allowed. The test has a duration of 1h30.

---

**[10 val.]** **Q-1** Consider a task manager application to allow users to be organized in teams, to manage a shared backlog, to assign a task to a member, and change the status of a task. The application is specified by the following user stories:

1. An anonymous user, can sign up in the system, and see his personal details, as a registered user.

2. An anonymous user, can log in the system, and see a list of assigned tasks.

3. A registered user, can access the application, and see the list of assigned tasks.

4. A registered user, can create a team, and see its empty backlog.

5. A registered user as owner of a team, can invite registered users to his team, as members.

6. A registered user, can add a task to the backlog of a team.

7. A registered user, can see the list of tasks in the backlog of a team.

8. A registered user, can assign a task in the backlog to some members of the team.

9. A registered user, can mark one of tasks assigned to him as completed.

10. A registered user, as owner of a team, can mark tasks in the team's backlog as completed.

A task is further described by a small text and a due date. All lists of tasks can be filtered by date and completion status. **Note** that the list of tasks of a user includes tasks from different teams.

This question is about the definition of a service based application, using the REST architectural style, Java Spring and JPA.

a) **[4 val.]** **Define** the necessary resources, operations, and parameters that are needed to (strictly) implement the user stories above. State the essential URLs, method (GET, POST, etc.), parameters, and example response (e.g. JSON).

b) **[6 val.]** **Define** the data model of the application using JPA annotations and a set of classes. Be sure to **represent** the relations between entities with JPA relation annotations. It is not necessary to list getters and setters.

**[5 val.] Q-2** This question is about the implementation of business logic in an application as the one described in **Q-1**.

a) **[3 val.] Define and implement** controller methods for user stories 5 and 7. **Declare** the method annotations, signatures, and code. **Define and implement** auxiliary services and repository methods to achieve the best possible solution in terms of correctness, efficiency, and modularity.

b) **[2 val.] Define** a repository method to support a controller method implementing user story 3. Provide the best possible solution in terms of correctness and efficiency.

---

**[5 val.] Q-3** This question is about the implementation of security policies at the level of a service based application.

a) **[1 val.] Compare** the following security models:

- the standard Role-based access control model in the development of service-based applications,
- the actual Spring role implementation, and
- the data dependent model based access control presented in the course lectures.

b) **[4 val.] Define** the controller methods' signatures and security annotations for the user stories 5, 7, 8, and 10 in **Q-1**. **Define and implement** auxiliary services to implement data dependent access policies.