# Gold's Evaluation 3

Filip Krepinsky (410022)

## I. Introduction

Goal of this text is to evaluate milestone 3 of gold's project; System for Management of Sporting Events[1].

## II. Checklist

- **[OK]** Mvn clean install passed.
- **[OK]** Tests Passed.
- **[OK]** Every team member commited roughly enough functionality.
- **[OK]** User interface - small mistakes. Requirements met.
- **[NOK] [-1]** REST layer - lacking security (explained later). Requirements met.
- **[OK]** Security (Web). Requirements met.

**Total Points: 9**

## III. Code Review

### A. User interface

- It is not possible to see our created events at http://localhost:8080/pa165/my-account.
- Page crashes when you try to register same sportsman twice at http://localhost:8080/pa165/register
- NumberFormatException are shown in numeric fields if wrong values are inserted.
- It is possible to run nice scripts on our own account, if you register for an example with a name as
  <script>alert("You have been hacked!");</ script >
  Sadly I couldn't make this work for other cases, so it is quite useless for hackers.

### B. REST layer

- All user hashes are accessible through the rest api. I know the requirements say the rest interface doesn't have to be secured, but I think this tackles mainly authentication. In my opinion hashes shouldn't be accessible in any way.
- EventsRESTController and SportsmanRESTController have very inefficient GetAllEvents/getAllSportsmans with a name.
- Update methods in DAOs don't return updated objects. The consequence of that is that the REST controllers are not atomic when updating objects. They are also less efficient because find methods are called after update.

## IV. General Remarks

- I think it is not a good approach to store distinction between user and admin(manager) roles as boolean in db.

[1] https://github.com/m-mato/sem