UTCN
Computer Science Department
Software Design 2021

ASSIGNMENT A2

==========

1. Objective

The objective of this assignment is to allow students to become familiar with MVC architectural pattern, services, repository and unit tests.

2. Application Description

Use JAVA Spring/C# Web API to design and implement an application for the tracking the laboratory activity for the Software Design laboratory. The application should have two types of users (student and teacher) which must provide an email and a password to use the application.

The teacher can perform the following operations:

- Login
- CRUD on students. When you create a student, a 128 characters token is created. Using that token student should be able to register. Teacher will send the token by email manually (not part of the scope of the application). For each student we should track: email address, full name, group (ex. 30434) and hobby – free field.
- Can add/edit/delete Laboratory classes. For each class we should track: laboratory number (#1-#14), date, title, curricula for what are the topics presented in that lab and a long description with the laboratory text.
- CRUD on attendance for each lab.
- CRUD on assignments. Some of the laboratory will have assignments: for each assignment we must track the name, deadline and a long description with the assignment text.
- Grade the submitted assignments individually.

The student can perform the following operations:

- Register using the token generated by the teacher, at this step they must provide the password.
- Login with the username and password.
- View a list of laboratory classes.
- View the assignments for a laboratory class.
- Create an assignment submission. Here, students should be able to insert a link to a git repository and a short comment (optional) for the teacher.

3. **Application Constraints**
   - The data will be stored in a relational database. Database model should respect 1st, 2nd and 3rd normal forms and proper relations between tables (1:1, 1:n, m:n)
   - Use the MVC architectural pattern to organize your application. For this assignment we will create only the backend part (Model, Controller, Services (Business layer) and Repositories).
   - API design should be RESTful.
   - Use an ORM (Hibernate / Entity framework) to access the database
   - Use dependency injection to inject Services in Controllers and Repositories in Services
   - Install and use Swagger to call your APIs / Or provide a Postman collection
   - Connection string and magic strings should be stored in a separate config file
4. **Requirements**
   - Create the analysis and design document (see the template).
   - Implement and test the application.

5. **Deliverables**
   - GIT/TFS link with:
     1. Analysis and design document.
     2. Source files.
     3. SQL script for creating and populating the database with initial values.
6. **Deadline – 2 weeks**
7. **Resources:**

Web Api:
https://app.pluralsight.com/library/courses/implementing-restful-aspdotnet-web-api/table-of-contents
Spring Boot:
https://javabrains.thinkific.com/courses/take/springboot-quickstart/multimedia/2784009-adding-a-rest-controller

Grading:

4 points	Correct & complete database & entity model (User, Laboratory, Assignment, Attendance) **Teacher:** CRUD of laboratories **Student:** View laboratory classes Postman collection Analysis & design doc **Server side validations:** Update Entity that does not exist Get Entity that does not exist Delete student that does not exist Create / Update with incorrect value
2 points	**Teacher:** Attendance & assignment creation
2 points	**Student:** Assignment Submission **Teacher:** Assignment grading
1 point	Correct usage of response codes (200,400,500) No magic strings Proper design of JSON Payloads when calling Rest APIs (use of ids and not strings)
1 point	Default

Tested flow for 10 (Example):

1. Login as Teacher
2. Create Student, save token
3. Create Laboratory
4. Create Assignment for lab created at 3
5. Register as student with user, password and token provided
6. View list of lab classes
7. View assignments for lab
8. Submit assignment for lab
9. Login again as Teacher
10. Grade submitted assignment

Tested flow for 8 (Example):

1. Login as Teacher
2. Create Student, save token
3. Create Laboratory
4. Create Assignment for lab created at 3
5. Register as student with user, password and token provided
6. View list of lab classes
7. View assignments for lab

Tested flow for 5 (Example):

1. Login as Teacher
2. Create Student, save token
3. Create Laboratory
4. Register as student with user, password and token provided
5. View list of lab classes