



BIRZEIT UNIVERSITY

Advanced Computer Systems Engineering Laboratory – ENCS5150

TODO 4: Simple Hospital Management Server

Objective:

Develop a Spring Boot application that exposes RESTful APIs for a hospital management system, including patients and doctors.

Models:

1. Doctor:
 - id (Long)
 - name (String)
 - maxPatients (int)
 - currentPatients (int)
 - experience (double)
2. Patient:
 - id (Long)
 - name (String)
 - illnessExperienceRequirement (double)
 - isCured (Boolean) -initially false-
 - doctors (ArrayList<Long>)

API Endpoints:

1. Utilize path variables in the API endpoints to access individual objects:

- **GET /doctors/{doctorId}** should retrieve only the information of that doctor.
- **GET /patients/{patientId}** should retrieve only the information of that patient.

2. Provide general endpoints for doctors and patients to retrieve all items of each type:

- **GET /doctors** should retrieve the information of all doctors.
- **GET /patients** should retrieve the information of all patients.

3. Post and Delete endpoints:

- **POST /doctors** (Request Body: Doctor object) should add a new doctor.
- **POST /patients** (Request Body: Patient object) should add a new patient.
- **DELETE /doctors/{doctorId}** should delete the specified doctor.

- **DELETE /patients/{patientId}** should delete the specified patient.

4. Assignment Endpoint (assigns a doctor to a patient):

- **POST /assign/{doctorId}** (Request Body: **Long** patientId)
 1. If the doctor's experience is equal to or above the illness experience requirement and the doctor is not fully booked:
 - Increment Current number of patients variable in the doctor object.
 - Add doctor ID to the patient's doctors list.
 - Return true.
 2. Else:
 - Return false

5. Patient Releasing Endpoint:

- **PUT /release/{patientId}**

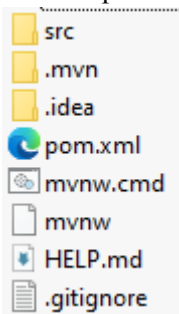
Change isCured variable to true, and delete all doctors in the patient's doctors list.

6. Welcome Message on the root path “/”:

- **GET /** should return a welcome with your name: "Welcome, I'm YOUR_NAME"

Important Notes:

- Use ArrayList for storage, no database is required.
- Test your application with Postman to verify that all endpoints are working properly.
- POST/DELETE/PUT should return true or false Boolean, based on the successfulness of the request.
- Provide appropriate error messages for any incorrect GET requests such as requesting a non-existing patient.
- Make sure to use port 8080.
- The returned data from GET requests as well as the request body in POST requests should be in **JSON** format, except the Assignment Endpoint.
- **Use the same variables and path names mentioned above, maintaining the case sensitivity.**
- ToDo is individual work and cheating will result in a **0 mark**.
- Export your project using IntelliJ (File → Export → Export to Zip File).
- Name it “ID_FirstName_LastName.zip”, and send it as a reply to this message.
- The compressed file should contain something similar to this:



- Deadline: **30/05/2024 Midnight**