This is a test assignment which covers WebSocket, DB, auth and form functionality in Spring Framework, Choose the latest stack you are using right now.

We need to create simple Backend for result management for students.

**Tech Stack:**

* Spring Boot Reactive (Java)
* Spring Security (JWT strategy)
* MongoDB

1) **BACKEND**

1.1) **Implement WebSocket’s API to manage students**

Students should have fields:

* Name
* Roll Number
* Father Name
* Grade
* Status
* CreatedOn
* UpdatedOn

Results will have below fields:

* Total Marks
* Obtained Marks
* Roll Number
* Grade
* Remarks
* PositionInClass
* CreatedOn
* UpdatedOn

1.2) Position of Student will be depending on the number of marks, student with highest marks will be on number one position

2) **Tasks**

2.1) This is a result compilation platform.

2.2) The Students Results will be posted in JSON format to websocket

{

“TotalMarks”:100,  
 “ObtainedMarks”:70,  
 “RollNumber”:1,

}

2.2-i) The Passing marks percentage will be 50 and remarks value will be **passed**/**failed** depending on the total marks and obtained marks field values.

2.2-ii) Roll Number must be validated from students’ collection with active status and Grade.

2.2-iii) Every time a new request will come, the data will be inserted in mongodb. The position attribute will be updated accordingly e.g., if there are two records already inserted in database and one new record came with more marks then the previous users then the new user will get the first position and the other users will get 2nd and 3rd positions respectively according to their marks.

2.2-v) send the appropriate response message on the websocket

2.3) Expose an API with URI /students with below mentioned request body

{

“Name”: “mike”,

“RollNumber”:1 **-- min value:1 and max value: 100**

“FathersName”:”john”,

“Grade”: 1, **-- min value:1 and max value: 10**

}

2.3-i) Student records will be inserted with Active status

2.4) Expose an API for soft deleting the student that will update the status of the student to deleted with below mentioned request body

{

“RollNumber”:1 -- min value:1 and max value: 100

“Grade”: 1, -- min value:1 and max value: 10

}

2.4) Expose a Rest API, with URI /students that will return results all the students

2.5) Expose Rest End point with URI /students/result/{roll-number}, this will return the result of student

2.7) only registered users can perform all above mentioned actions

3) **RESULT**

3.1) A new git repository should be created for the project and all changes should be committed progressively. We should be able to easily see the parts you coded and differentiate it from framework code. Please add the public repository URL to your reply.

3.2) in comment to that task provide summary of your work in English. Please, write it in formal way as if you're talking with client

3.3) Share the total time you have worked on the task.

**NOTES (Important pointers to follow)**:

- Create migrations and seeders to have at least 3 student records in the DB to start with.

- Write a **unit test** of your work. (Choose a testing framework of your choice) and try to achieve maximum code coverage.

- Add comments & format your code according to standards.

- Apply SOLID principles when writing your code.