

## FULL STACK - CAPSTONE PROJECT

### SPORTING EVENTS SYSTEM

You are required to write a software solution for the Football World cup that allows administrators to manage various teams, players, tournaments, matches and tickets

The application has to be implemented using Spring Boot & Microservices as the backend and Angular as the frontend. This application stores details of teams, players, tournaments, matches and tickets.

The application has to be implemented as two versions:

- Version 1: An application with monolithic architecture using Spring Boot.
- Version 2: An application with microservices architecture using Spring Boot and Spring cloud.

This software needs to address the following requirements:

#### **Business Requirements:**

The project is for administrators who use the system to perform the following operations.

The Administrator module should allow administrators perform:

- A. Login and authentication.
- B. Saving details of teams, players, tournaments, matches and tickets.
- C. Viewing and deleting the details of teams, players, tournaments, matches and tickets.
- D. Updating various details periodically.
- E. Search for the required details based on the id and key fields.
- F. New administrators to register.

## **User Stories:**

### ***US01: Login***

As an Administrator,

I want to login to the application; so that I can get access to the welcome screen and application menu.

The following conditions and validations are to be performed during login

- The login id will be a valid email address.
- Password should be a combination of at least an uppercase alphabet, lowercase alphabets, a digit, and a special character.
- The email id and password cannot be blank.
- The account should get disabled for 30 minutes on three invalid login attempts.
- Proper messages should be displayed in case of errors and exceptions.

### ***Acceptance criteria***

Given that I am an administrator,

When I enter correct User Id with Password in the Login screen

Then I should be able access the application menu

And navigate the menu to view the required details

Given that I am an administrator,

When I enter incorrect User Id with Password in the Login screen

Then I should be able prompted for a maximum of three times to reenter the credentials.

Given that I am an administrator,

When I enter incorrect User Id with Password in the Login screen consecutively for three times

Then my account should get disabled for 30 minutes.

### ***US02: Creating teams, players, tournaments, matches and tickets***

As an Administrator, I want to create teams, players, tournaments, matches and tickets.

- The teams, players, tournaments, matches and tickets should be created with a unique id.
- The id should be 5 characters in the format "CNNNN" where the first character is an alphabet and last 4 characters are running numbers
- The id should be autogenerated.
- The id once generated cannot be modified.
- The required fields should be accepted and validated.

#### *Acceptance criteria*

Given that I am an administrator,

When I select the option to "Create by clicking the appropriate buttons or links"

And enter the required details

Then I should be able to autogenerated the id

And I should be able to save teams, players, tournaments, matches and tickets

### ***US03: Viewing teams, players, tournaments, matches and tickets***

As an Administrator, I want to view teams, players, tournaments, matches and tickets.

#### *Acceptance criteria*

Given that I am an administrator,

When I select the option to "View by clicking the appropriate buttons or links"

I should be able to view teams, players, tournaments, matches and tickets

### ***US04: Searching teams, players, tournaments, matches and tickets***

As an Administrator, I want to search teams, players, tournaments, matches and tickets.

### *Acceptance criteria*

Given that I am an administrator,

When I select the option to "Search by clicking the appropriate buttons or links"

And enter the id or key fields

I should be able to view teams, players, tournaments, matches and tickets

### ***US05: Deleting teams, players, tournaments, matches and tickets***

As an Administrator, I want to delete teams, players, tournaments, matches and tickets.

### *Acceptance criteria*

Given that I am an administrator,

When I select the option to "Delete by clicking the appropriate buttons or links"

I should be able to delete teams, players, tournaments, matches and tickets

### ***US06: New Admin Registration***

As a New Administrator, I should be able to register myself to keep track and work on the system.

The following conditions and validations are to be performed when the new admin is registered

- An error message is displayed if the email id is already in use by some other user
- An error message is displayed if all the details are not filled
- An error message is displayed if the name contains numbers or special characters
- Password should be a combination of at least an uppercase alphabet, lowercase alphabets, a digit, and a special character.
- An error message is displayed if the password entered by the customer is not in the expected format
- An error message is displayed if the phone number is already in use by some other user
- The user should be able to login with his/her emailId and password

### *Acceptance criteria*

Given that I am an administrator,

When I select the option to register a new admin

And enter all the required details

Then I should be able to save the register the administrator.

### **Database Design:**

- Create the database ER Diagram and implement the database.
- Create required constraints, procedures and functions as relevant and applicable.
- The database should be well normalized.
- Recommended to have a minimum of 3 fields to a maximum of 10 fields in each table.

### **Functional Requirements Backend:**

- A. The Application should be implemented using Spring boot and Microservices architecture.
- B. The microservices should be registered in a Service Registry.
- C. You should provide access to all microservices using a dynamic API gateway
- D. The API gateway should act to secure the application.
- E. Perform client-side load balancing.
- F. The REST APIs should be tested with Postman.
- G. Perform ORM using the Spring Data JPA and Hibernate framework.
- H. The backend to be used is MySQL.
- I. Appropriate validations and exception handling should be performed in the relevant layers of the application.
- J. The relevant Design principles and practices should be applied.

- K. Perform Unit Testing.
- L. Write effective java code.
- M. Extensively use Java 8 and advanced features wherever applicable.

### **Functional Requirements Frontend:**

- A. The Application should be implemented using Angular and Twitter Bootstrap.
- B. The application should have a separate routing module.
- C. The URL fragments in the application should be guarded using Route Guards.
- D. The service layer should connect to the existing microservices backend.

### **Best Practices Backend**

- A. Follow proper naming conventions and ensure code quality
- B. Create meaningful comments to explain the purpose of the class and behaviours.
- C. Use StringBuilder for concatenating large strings
- D. Enforce Compile-Time Type Checking using Generics
- E. Validate Parameter Preconditions
- F. Proper handling of NullPointerException and other relevant exceptions.
- G. Perform validations across the different layers of the project
- H. The code should have modularity.
- I. The deleted records should be flagged as inactive or deleted, but not physically deleted in the database.

### **Best Practices Frontend**

- A. Follow proper naming conventions and ensure code quality
- B. Validate Parameter Preconditions
- C. The code should have modularity.

- D. The routes should be placed in its own module.

### **Submission Requirements Backend:**

- A. Submit a single archive containing all the artefacts
- B. The artefacts are
  - a. Project Source Code,
  - b. Presentation,
  - c. Postman Test Collection
  - d. Database Scripts and
  - e. A word document containing the screenshots of the
    - i. Tests in postman collection
    - ii. Instances of microservices running on Eureka and
    - iii. Other relevant screenshots, if any.
- C. Along with the code you are also required to submit a document that mentions
  - a. Project Name
  - b. A tabulation containing
    - i. Application Name of your Microservices
    - ii. Port number assigned to that microservice

### **Submission Requirements Frontend:**

- A. Submit a single archive containing Angular source code and resources.
- B. Along with the code you are also required to submit a document that mentions
  - a. Project Name and
  - b. The screenshots of the
    - i. The angular screens and forms
      - 1. With live data from the microservices.
      - 2. Forms with

- a. Validation error messages and
- b. Success Messages

**Project Presentation Requirements:**

- A. The Project should be presented individually with execution and presentation.
- B. Explain the details of API used, processes involved, workflows and frameworks utilized.