# **FULL STACK - CAPSTONE PROJECT**

## **SPORTING EVENTS SYSTEM**

You are required to write a software solution for the ecommerce business like Amazon or Flipkart which allows administrators to manage various products, cart, orders, offers and reviews.

The application has to be implemented using Spring Boot & Microservices as the backend and React as the frontend. This application stores details of products, cart, orders, offers and reviews.

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:

1. Login and authentication.
2. Saving details of products, cart, orders, offers and reviews.
3. Viewing and deleting the details of products, cart, orders, offers and reviews.
4. Updating various details periodically.
5. Search for the required details based on the id and key fields.
6. 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

1. The login id will be a valid email address.
2. Password should be a combination of at least an uppercase alphabet, lowercase alphabets, a digit, and a special character.
3. The email id and password cannot be blank.
4. The account should get disabled for 30 minutes on three invalid login attempts.
5. 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 renter 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 products, cart, orders, offers and reviews

As an Administrator, I want to create products, cart, orders, offers and reviews.

1. The products, cart, orders, offers and reviews should be created with a unique id.
2. The id should be 5 characters in the format “CNNNN” where the first character is an alphabet and last 4 characters are running numbers
3. The id should be autogenerated.
4. The id once generated cannot be modified.
5. 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 autogenerate the id

And I should be able to save products, cart, orders, offers and reviews

## US03: Viewing products, cart, orders, offers and reviews

As an Administrator, I want to view products, cart, orders, offers and reviews.

### 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 products, cart, orders, offers and reviews

## US04: Searching products, cart, orders, offers and reviews

As an Administrator, I want to search products, cart, orders, offers and reviews.

### 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 products, cart, orders, offers and reviews

## US05: Deleting products, cart, orders, offers and reviews

As an Administrator, I want to delete products, cart, orders, offers and reviews.

### 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 products, cart, orders, offers and reviews

## 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

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

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

## Requirements:

1. The Application should be implemented using Spring boot and Microservices architecture.
2. The microservices should be registered in a Service Registry.
3. You should provide access to all microservices using a dynamic API gateway
4. The API gateway should act to secure the application.
5. Perform client-side load balancing.
6. The REST APIs should be tested with Postman.
7. Perform ORM using the Spring Data JPA and Hibernate framework.
8. The backend to be used is MSSQL.
9. Appropriate validations and exception handling should be performed in the relevant layers of the application.
10. The relevant Design principles and practices should be applied.
11. Perform Unit Testing.
12. Write effective java code.
13. Extensively use features from Java 8 and advanced features till Java 21 wherever applicable.
14. Use Typescript features

## Best Practices

1. Follow proper naming conventions and ensure code quality
2. Create meaningful comments to explain the purpose of the class and behaviours.
3. Use StringBuilder for concatenating large strings
4. Enforce Compile-Time Type Checking using Generics
5. Validate Parameter Preconditions
6. Proper handling of NullPointerException and other relevant exceptions.
7. Perform validations across the different layers of the project
8. The code should have modularity.
9. The deleted records should be flagged as inactive or deleted, but not physically deleted in the database.
10. Validate Parameter Preconditions
11. The code should have modularity.

## Submission Requirements Backend:

* 1. Submit a single archive containing all the artefacts
  2. The artefacts are
     1. Project Source Code,
     2. Presentation,
     3. Postman Test Collection
     4. Database Scripts and
     5. A word document containing the screenshots of the
        1. Tests in postman collection
        2. Instances of microservices running on Eureka and

iii. Other relevant screenshots, if any.

* 1. Along with the code you are also required to submit a document that mentions
     1. Project Name
     2. A tabulation containing
        1. Application Name of your Microservices

ii. Port number assigned to that microservice

## Submission Requirements Frontend:

* 1. Submit a single archive containing React source code and resources.
  2. Along with the code you are also required to submit a document that mentions
     1. Project Name and
     2. The screenshots of React screens and forms
        + 1. With live data from the microservices.
          2. Forms with

Validation error messages and

Success Messages

## Project Presentation Requirements:

* 1. The Project should be presented individually with execution and presentation.
  2. Explain the details of API used, processes involved, workflows and frameworks

utilized.