**Capstone Project: Music Library - Full Stack Development**

**Project Overview:** The Music Library application will be developed with the following key components:

* **J2EE Stack:**
  + **Java EE**: Enterprise application framework.
  + **Spring Boot**: Microservices architecture.

**Detailed Project Breakdown**

**User Stories:**

* **User Functionalities:**
  + **Registration and Authentication:**
    - As a user, I should be able to log in, log out, and register with personal details like email ID and phone number.
  + **Song Library:**
    - As a user, I should be able to see the list of songs available in the library.
    - As a user, I should be able to view details of a song, including name, singer, music director, release date, and album name.
  + **Search Functionality:**
    - As a user, I should be able to search for any song using music director, album, or artist.
  + **Playlist Management:**
    - As a user, I should be able to create, view, update, and delete one or multiple playlists.
    - As a user, I should be able to add one or more songs to a playlist.
    - As a user, I should be able to search for any song within a playlist.
    - As a user, I should be able to perform CRUD operations on playlist songs.
    - I should be able to control playback within the playlist with play, stop, repeat, and shuffle options.
* **Admin Functionalities:**
  + **Registration and Authentication:**
    - As an admin, I should be able to log in and log out of the application.
  + **Song Management:**
    - As an admin, I should be able to perform CRUD operations on the song list.
    - As an admin, I should be able to restrict the visibility of songs from users.
  + **Notifications:**
    - As an admin, I should be able to send notifications whenever new songs are added to the library.

**Sprint Plan**

**Sprint I Objectives:**

* **Database Setup:**
  + Create the database schema with tables and relationships.
  + Use H2 database for Java Spring Boot services.
* **Backend Development:**
  + Implement CRUD operations for users and admins.
  + Create a microservice-based structure.
  + Develop templates using HTML and CSS.

**Sprint II Objectives:**

* **Feature Development:**
  + Develop search functionality for songs and playlists.
  + Implement CRUD operations for songs by the admin.
  + Implement visibility restrictions for songs.
  + Develop CRUD operations for playlists and playlist songs by users.
  + Implement Spring Security and JWT for authentication and authorization.

**Sprint III Objectives:**

* **Service Layer and Integration:**
  + Create Data Transfer Objects (DTOs).
  + Develop the service layer logic and repositories.
  + Implement REST API controllers.
  + Develop a notification microservice for new songs added.
  + Integrate the frontend with the backend.
  + Add extra features as needed.

**Instructions:**

* **Microservice Architecture:**
  + Create separate microservices for user and admin functionalities using Spring Boot.
  + Register all microservices with Eureka Server.
  + Develop an independent microservice for sending notifications.
  + Utilize H2 database for storage, with each microservice having its own database instance.
  + Implement Swagger for each microservice.
  + Establish connections between microservices.
  + Use Spring Cloud to share configurations.
  + Implement custom exceptions to handle invalid inputs.
  + Ensure data consistency and implement relationships where applicable.

**apstone Project: MyFin Bank - Full Stack Development**

**Project Overview:** The MyFin Bank application will be developed with the following key components:

* **J2EE Stack:**
  + **Java EE**: Enterprise application framework.
  + **Spring Boot**: Microservices architecture.

**Detailed Project Breakdown**

**User Stories:**

* **Customer Functionalities:**
  + **Registration and Authentication:**
    - As a Customer, I should be able to register, log in, and log out from the application.
  + **Bank Balance Operations:**
    - As a Customer, I should be able to:
      * Deposit funds.
      * Withdraw funds.
      * Transfer funds.
  + **Transaction Management:**
    - As a Customer, I should be able to generate a transaction ID for every transaction performed.
  + **Investment Transfers:**
    - As a Customer, I should be able to transfer money to other investment categories in the same bank, such as:
      * Loan
      * Recurring Deposits
      * Fixed Deposits
  + **Loan EMI Calculation:**
    - As a Customer, I should be able to calculate the loan EMI based on the loan amount, rate of interest, and number of months.
  + **Loan Application:**
    - As a Customer, I should be able to apply for loans.
  + **Customer Support:**
    - As a Customer, I should be able to chat with bank authorities for queries.
* **Admin Functionalities:**
  + **Registration and Authentication:**
    - As an Admin, I should be able to register, log in, and log out from the application.
  + **Customer Management:**
    - As an Admin, I should be able to perform CRUD operations on Customers (cannot delete but can deactivate/activate existing customers).
  + **Account Management:**
    - As an Admin, I should be able to perform CRUD operations on Customer Accounts.
  + **Balance Monitoring:**
    - As an Admin, I should receive notifications via email when any customer’s bank balance turns to zero.
  + **Loan Approval:**
    - As an Admin, I should be able to approve or deny loans based on customer balance.
  + **Customer Support:**
    - As an Admin, I should be able to chat with customers in case of queries.

**Sprint Plan**

**Sprint I Objectives:**

* **Database Setup:**
  + Create the database schema with all tables and their relationships.
  + Use H2 database for Java Spring Boot services.
* **Backend Development:**
  + Implement CRUD operations for users and admins using Spring Boot.
  + Develop templates using HTML and CSS.

**Sprint II Objectives:**

* **Feature Development:**
  + Develop loan functionality (apply/deny).
  + Implement bank balance operations (deposit/withdraw/transfer).
  + Implement investment transfers (Loan/Recurring Deposits/Fixed Deposits).
  + Implement loan EMI calculation.
  + Implement Spring Security and JWT for authentication and authorization.

**Sprint III Objectives:**

* **Service Layer and Integration:**
  + Create Data Transfer Objects (DTOs).
  + Develop the service layer logic and repositories.
  + Implement controllers to handle REST API requests.
  + Create an email notification microservice for balance monitoring.
  + Integrate the frontend with the backend.
  + Implement chat functionality.
  + Ensure the application is responsive and performs well.

**Instructions:**

* **Microservice Architecture:**
  + Create separate microservices for customer and admin functionalities using Spring Boot.
  + Register all microservices with Eureka Server.
  + Develop an independent microservice for dispatching email notifications.
  + Utilize H2 database for storage, with each microservice having its own database instance.
  + Implement Swagger for each microservice separately.
  + Establish connections between microservices.
  + Use Spring Cloud to share configurations between microservices.
  + Implement custom exceptions in each microservice to handle invalid inputs.
  + Ensure data consistency and implement relationships where applicable.