Microservices Setup Guide

To run the microservices for your assignment, follow the steps outlined below. Ensure that you have an Integrated Development Environment (IDE) such as Eclipse or IntelliJ installed.

Prerequisites

GitHub Account:

- Make sure you have a GitHub account to host common configurations.

IDE Setup:

- Open each microservice project in your preferred IDE.

Dependency Loading:

- Load all project dependencies as specified in the respective pom.xml files.

Microservices Configuration

For centralized configuration management, follow these steps:

GitHub Repository for Configurations:

- Create a GitHub repository.
- Upload all common configurations to this repository.

Configuring Config Server:

- In the application.yml file of the Config-server microservice, provide the GitHub repository link containing common configurations.

Database Configuration (Account Service):

- Open the account-service/src/main/resources/application.properties file.
- Update the database configurations to match your setup:

Database Configuration (Customer Service):

- Open the customer-service/src/main/resources/application.properties file.
- Update the database configurations to match your setup:

Starting Microservices

Start the microservices in the following order:

1.**Eureka Server:**

- Start the Eureka Server, which serves as the service registry.

2.**Config Server:**

- Start the Config Server to manage centralized configurations.

3.**Account Service:**

- Start the Account Service, responsible for account-related operations.

4.**Customer Service:**

- Start the Customer Service, handling customer-related functionalities.

5**.API Gateway:**

- Start the API Gateway, acting as a central entry point for requests.

Running the Services

After starting all services, use the provided URLs to interact with the APIs using a tool like Postman. Refer to the API documentation for detailed information on how to make requests.

Order of Service Startup

Follow the sequence below to ensure proper service startup:

1. **Eureka Server**
2. **Config Server**
3. **Account Service**
4. **Customer Service**
5. **API Gateway**

Conclusion

Upon successfully starting all microservices, you can use the provided APIs to test and interact with the functionality of your system. Refer to the API documentation for details on how to use each service's endpoints.

Note: Ensure that you have the necessary configurations and dependencies correctly set up to avoid any issues during the startup process.