Course 3 – Sprint 5: Spring Microservices

Introduction:

Microservices allow large systems to be built up from a number of collaborating components. It does at the process level what Spring has always done at the component level: loosely coupled processes instead of loosely coupled components.

For example, imagine an online shop with separate microservices for user-accounts, product-catalog order-processing and shopping carts.

What is a Monolith Application?

- Project with many features and functionalities
- Project with more than 30 members
- Adding new technologies and process which is complicated and are needed in the application
- Project in which debugging is complicated

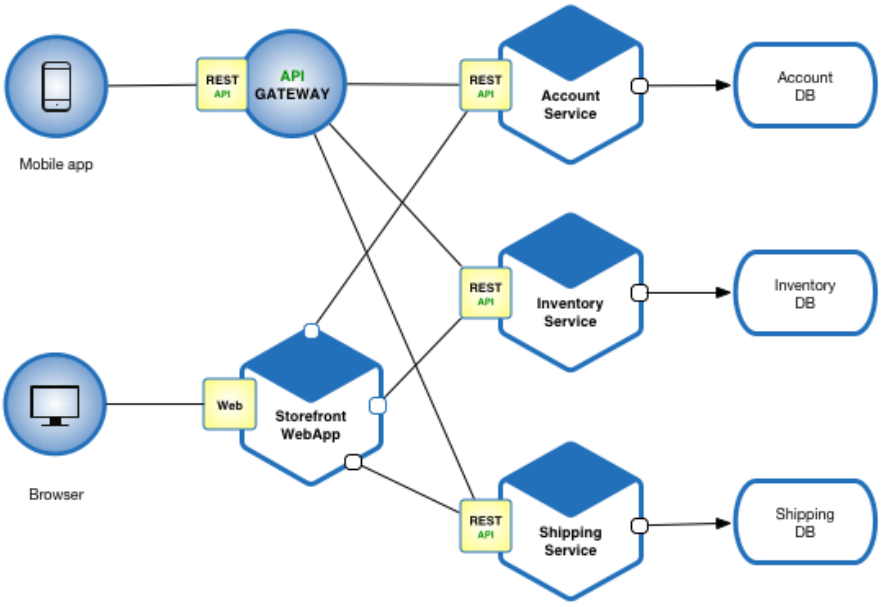
Monoliths are characterized by

- Large Application Size
- Long Release Cycles
- Large Teams

Microservices

Microservice is an architectural style that structures an application as a collection of services that are

- Highly maintainable and testable
- Loosely coupled
- Independently deployable
- Organized around business capabilities
- Owned by a small team



Courtesy: https://microservices.io/i/Microservice_Architecture.png

Microservice Architectures evolved as a solution to the scalability and innovation challenges with Monolith architectures.

Principles Behind Microservices:

- Scalability
- Load balancing
- Support for CI/CD
- Fault-tolerance
- Independent & Autonomous Services

Design patterns Used in Microservices:

- Aggregator
- API Gateway
- Chained responsibility
- Asynchronous messaging
- Branch
- Circuit breaker
- Decomposition
- Event Sourcing

- Command Query responsibility segregator

Note: https://www.edureka.co/blog/microservices-design-patterns for more details on each pattern

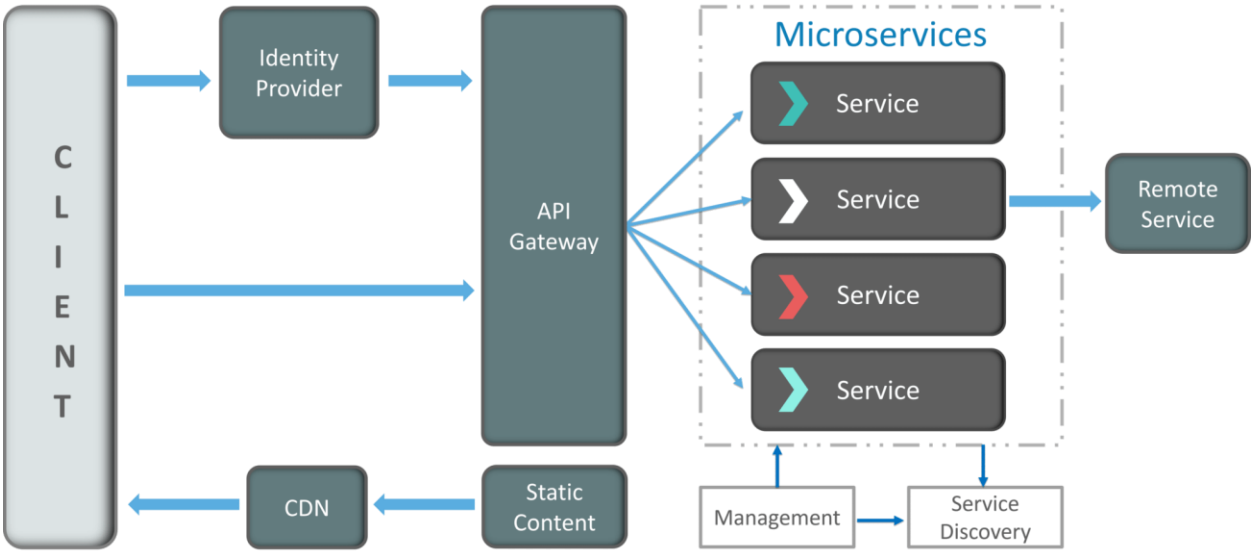
Guide to design Microservice:

- As a developer, we have to identify the domain and functionalities properly.
- Each Microservices must focus on only one Service of the application.
- Always ensure the communication happens properly and server less.
- We can break out microservices into how many smaller sub services too.

Components of Microservice:

A typical Microservice Architecture (MSA) should consist of the following components:

- Clients
- Identity Providers
- API Gateway
- Messaging Formats
- Databases
- Static Content
- Management
- Service Discovery



Note :

Please use the given link to create a spring boot quick start project

https://spring.io/quickstart