**Technical Document: Version-1.0.0**

**Project**: **LMS**

**Required Software:**

1. SpringToolSuite4
2. apache-maven-3.6.3-bin
3. Spring boot -2.0.1
4. openjdk-8-jre
5. Jenkins
6. Docker
7. Kubernetes
8. Git/ tortoise git as client
9. Aws console
10. Mongodb
11. Slf4j Logging Lombok
12. Reactjs for UI
13. RabbitMQ/ activeMQ for messaging
14. Redis cache

**Microservices architecture:**

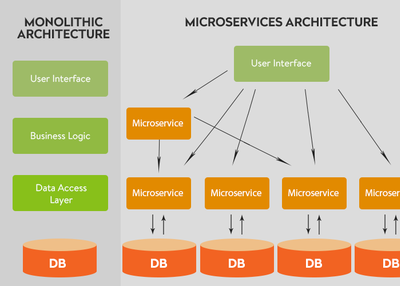
1. Circuit breaker design pattern
2. SAGA/ choreography DP
3. Zuul Api Gateway
4. Eureka Registry
5. Config Server
6. DB services
7. Communication using REST Template / Feign Client

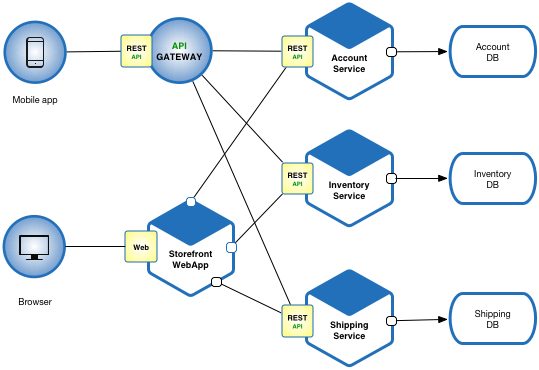
**List of Microservice:**

1. Booking service
2. Acounting Services
3. Notifiation Services

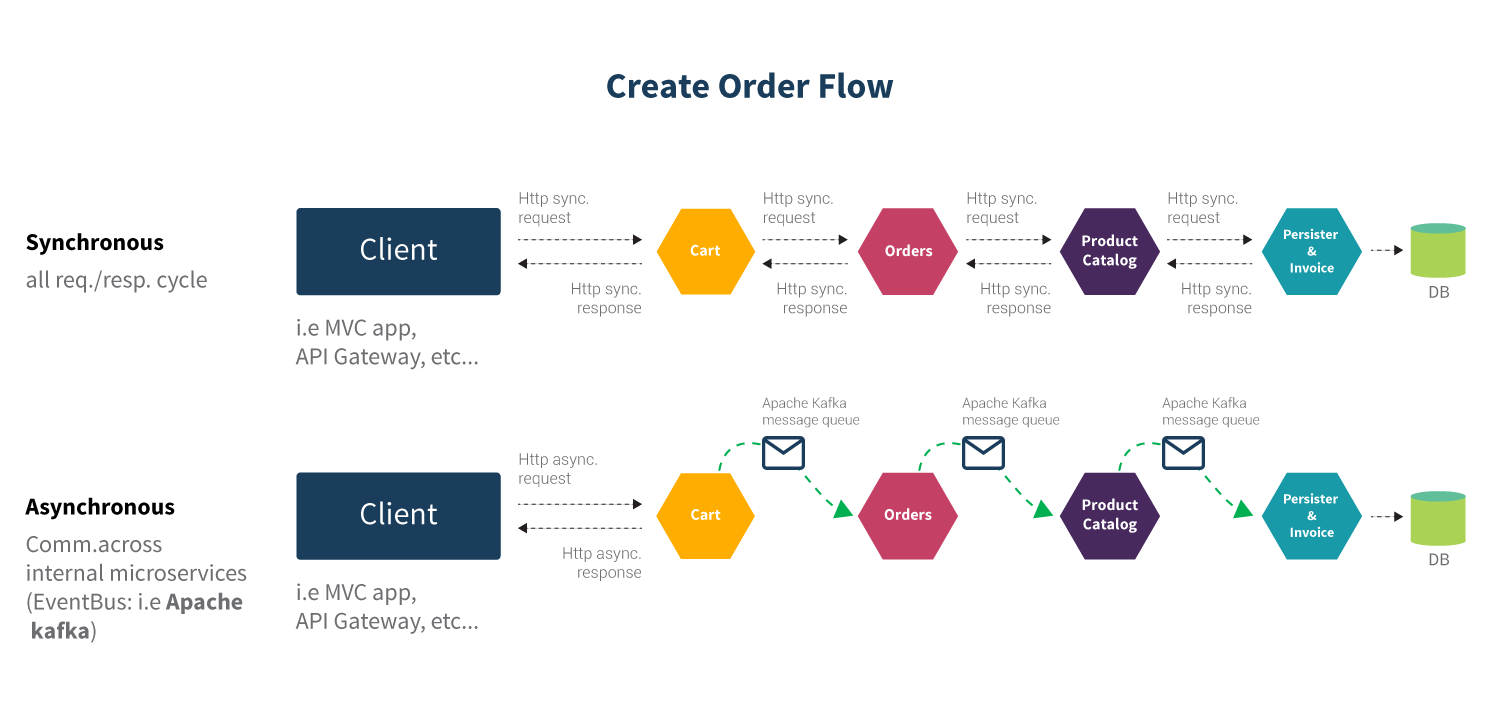
**Architecture of Microservices:**

**Monolithic architecture Vs Micro services:**

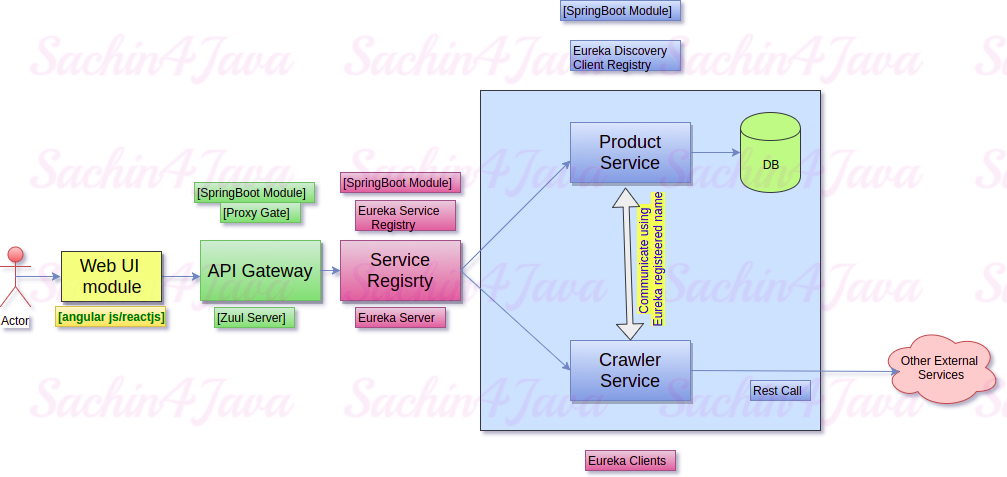


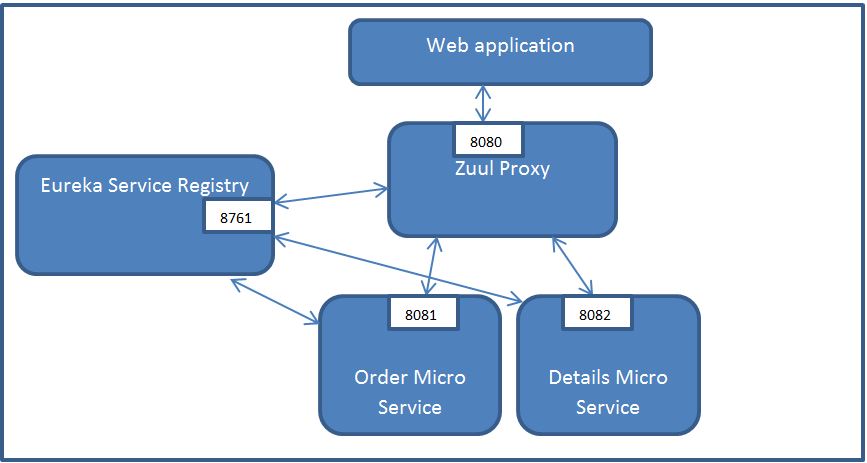


**Communication:**

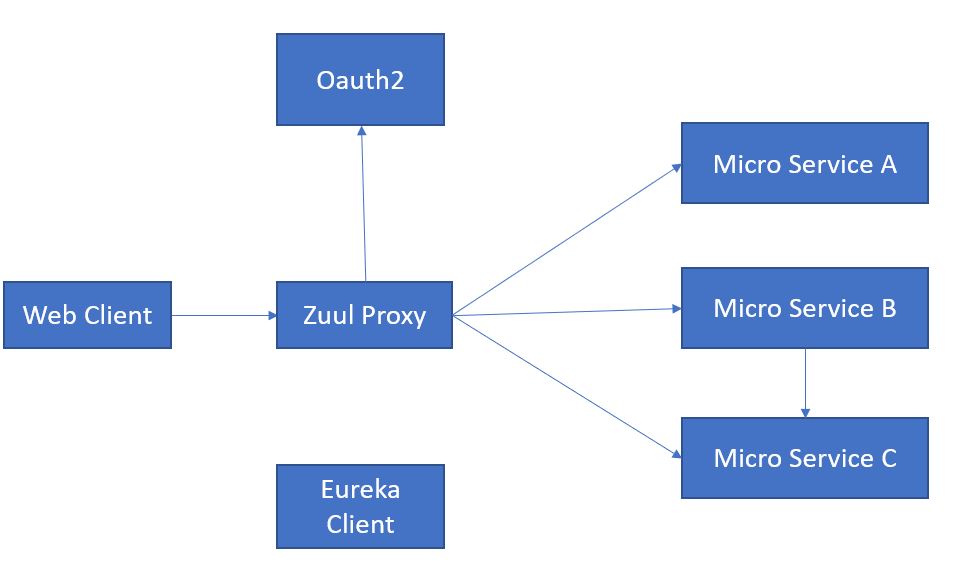


**Api gateway zuul and Service Registry:**





**Oauth2 for security:**



**Micro services Design:**

1. **Order Booking Service**
2. **POJO**
3. **Fields name of the Pojo**
4. **Controller Name**
5. **Service**
6. **DAO**
7. **DB**

**Endpoints:**

1. **POST**
2. **GET**
3. **PUT**
4. **DELETE**
5. **PATCH**
6. **Accounting Service**
7. **POJO**
8. **Controller Name**
9. **Service**
10. **DAO**
11. **DB**

**Endpoints:**

1. **POST**
2. **GET**
3. **PUT**
4. **DELETE**
5. **PATCH**
6. **Notification Service**
7. **POJO**
8. **Controller Name**
9. **Service**
10. **DAO**
11. **DB**

**Endpoints:**

1. **POST**
2. **GET**
3. **PUT**
4. **DELETE**
5. **PATCH**