Hipchat : Case Study

↳ Created by Atlassian

↳ Utilized Microservices architecture

(1) Architecture

Completely decentralized with many sub-microservice.
Used event-driven architecture with WebSocket protocol

(2) Design Pattern

Used API Gateway pattern to provide a single entry-point for all client requests and to route each request to the appropriate microservice.

Used Circuit Breaker pattern to prevent a network or service failure from cascading to other service.

(3) Messaging Pattern

↳ Used publisher-subscriber pattern

↳ Used Spring Cloud Stream with RabbitMQ

Examples of independent microservices

- Authentication Service
- Chat Room Service
- Message Service
- Notification Service

Testing Strategies in Microservices

* Types of testing:-

i) Unit Testing
- ↳ JUnit
- ↳ Mockito

ii) Integration Testing
- ↳ @SpringBootTest

iii) Contract Testing
- ↳ Tests the communication path & interactions between microservices to ensure each service can correctly communicate with its peers.
- ↳ Pact
- ↳ Spring Cloud Contract

iv) End-to-End Testing
- ↳ Test a complete flow from user's perspective
- ↳ Covers all microservices involved in that flow.

v) Performance Testing
- ↳ Knowing how your microservices behave under load.
- ↳ Gatling
- ↳ Apache JMeter

vi) Chaos Engineering
- ↳ Pioneered by Netflix
- ↳ Intentionally causing failures in your system to test the ability to withstand & recover.
- ↳ Netflix Simian Army
- ↳ Chaos Monkey

vii) Functional Testing:- Example - Smoke Testing (Build verification testing)

Microservice Testing Tools

1) JUnit and Mockito
2) Rest-Assured
3) Spring Boot Test
4) Pact
5) WireMock
6) Gatling
7) Chaos Monkey