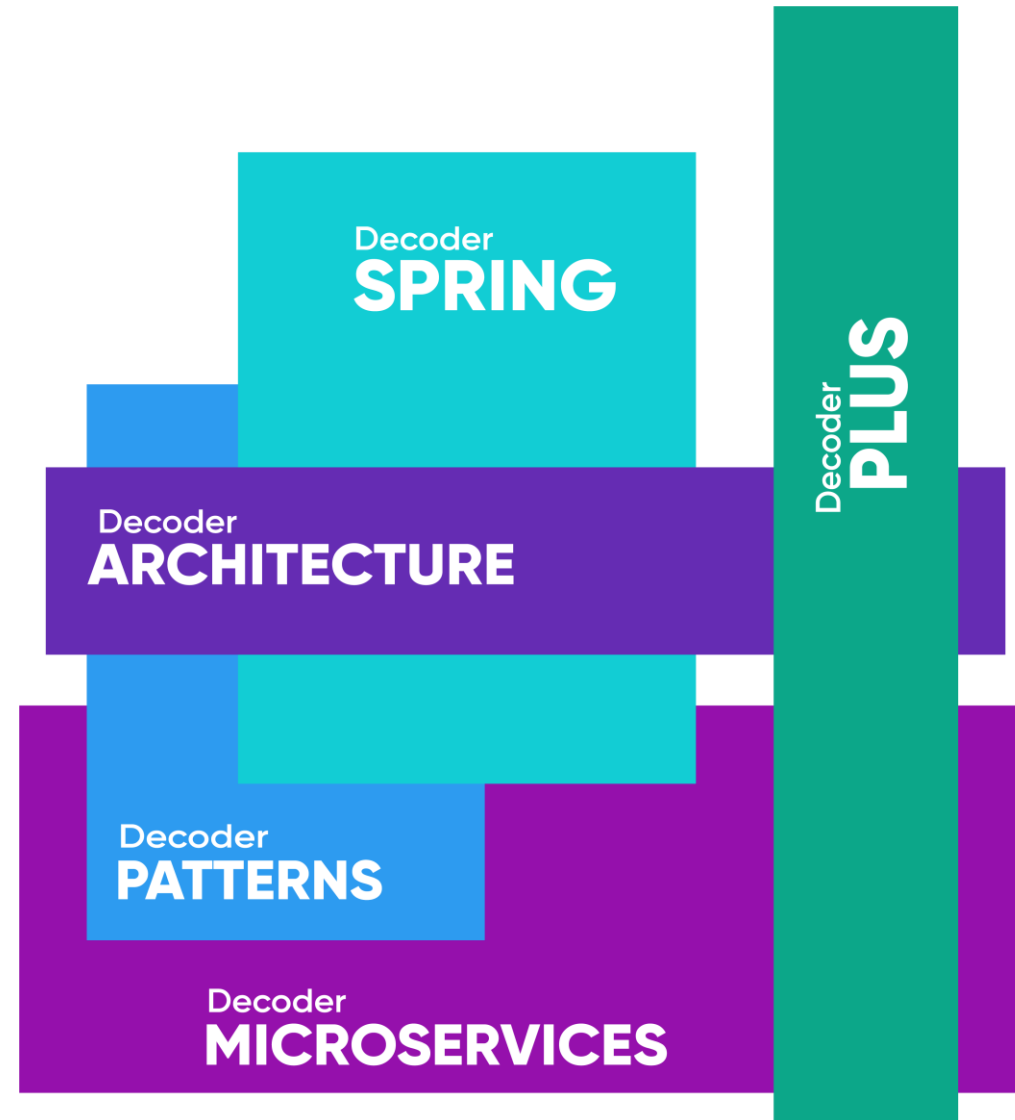


Overview Projeto Decoder

Formação Especialista em Microservices com Spring

Projeto Decoder: 5 PILARES



+ Decoder Microservices

Fundamentos Arquiteturas Distribuídas

Princípios Microservices

Fim dos Sistemas Monolíticos

Cultura Distribuída

Comunicações entre Microservices

Comunicação Síncrona

Comunicação Assíncrona

Mensageria

Messages via Commands

Messages via Events

Teorema CAP

Dados Distribuídos

Consistência Eventual

UUIDs Universais

Sincronia Dados

Comunicação OneToOne

Comunicação OneToMany

Request/Response Assíncrono

Consumer/Producer

Request/Response Síncrono

Comunicação Assíncrona One-Way

Exchanges / Queues / Bindings

Publisher/Subscribe

Brokers

Resiliência Síncrona

Retry

JWT

Autenticação Distribuída

Autorização com Roles

+ Decoder Patterns

Maturidade de Richardson para RESTful

Padrões de Datas

ISO 8601-UTC

MVC - Hexagonal

Database per Service Pattern

Shared Database Pattern

API Composition Pattern

API Gateway Pattern

Service Registry Discovery Pattern

Global Configuration Pattern

Cross-Cutting

Circuit Breaker Pattern

Event Driven Pattern

Event Notification Pattern

Event Carried State Transfer Pattern

Broker Pattern

Mediator Pattern

SAGA Pattern

Comunicação por Coreografia

Comunicação por Orquestração

Authentication e Authorization com Basic Authentication

Authentication e Authorization com JWT

Access Token Pattern

Observability

ECS Logging

Log Aggregation Pattern

+ Decoder Spring

Spring Boot

Spring Data

Spring Web

Spring Data JPA

Spring AMQP

Spring Validation

Spring HATEOAS

Spring Security

Spring Utils

Spring Initializr

Spring Cloud Netflix Eureka

Spring Cloud Gateway

Spring Cloud Config Server e Client

Spring Cloud Resilience4j

Spring Actuator

Spring Logback

Spring Log4j2

Serializações com Jackson ModelMapper

Lombok

LoadBalance

RestTemplate

Bibliotecas para Filters Specification e Pageable

Core Container

Spring Framework

Estereótipos do Spring

Inversão de Controle

Injeção de Dependência

Métodos Produtores

Beans

Pontos de injeção via @Autowired

JsonView

Pontos de injeção via Construtores

Custom Validator

@Constraint

@Query

@EntityGraph

FetchMode/FetchType

Relacionamentos JPA

+ Decoder Arquitetura

Projetar Arquiteturas de Microservices

Modelar Sistemas Distribuídos

Definição Arquitetural

Aplicabilidade dos Patterns

Modelos Arquiteturais

Concepção Arquiteturas Distribuídas

Granularidade dos Microservices

Arquitetar Componentes

Divisão dos Microservices

Gerir Microservices

Fluxos de Processos para Microservices

Layouts

Ferramenta Modelagem

Desenvolver Arquitetura de Ponta a Ponta

Fluxo do Spring Web MVC

Fluxo Spring Security

Fluxo Geração de Tokens com JWT e Spring Security

Fluxo Validação de Tokens com JWT e Spring Security

Cross-Cutting Authentication e Authorization com JWT em Microservices

Fluxo Híbrido de Comunicação Síncrona e Assíncrona

+ Decoder PLUS

Observabilidade com Elastic Stack (ELK)

Métricas de Logs com Kibana

Logs Estruturados com ECS Logging e Log4j2

Elasticsearch (Search engine)

Beats e FileBeat

Logs Centralizados em Tempo Real

Deploy Arquitetura de Microservices na Plataforma Heroku

Configuração Variáveis de Ambiente

Configuração de Múltiplos Ambientes: Dev e Prod

Utilização de Add-ons Heroku

Add-on Postgres Heroku

Add-on CloudAMQP Heroku

Deploy via Commands Line

Configuração de planos e instancias Cloud

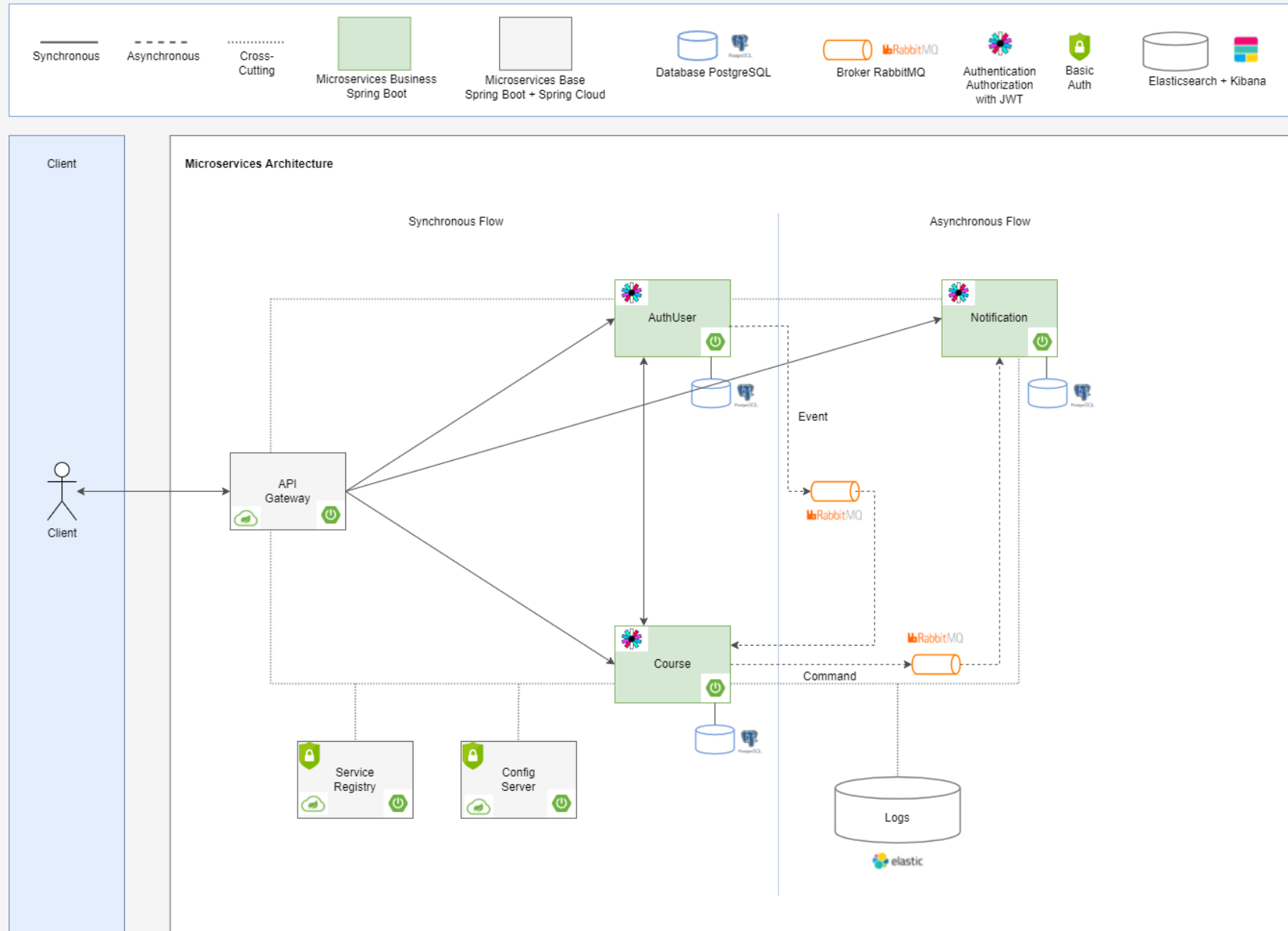
Encryptografia Config Server

Arquitetura Hexagonal: Da Teoria à Prática

Portas e Adaptadores

Desacoplamento e isolamento do Core Application

EAD Application



Obrigada!

Michelli Brito