

Table of contents

- Introduction
- Architecture Overview
 - Distributed Systems & Microservices
 - Recative Manifesto & Reactive Principles
 - State-of-the-art Patterns
 - * DDD (Domain Driven Design)
 - * CQRS (Command Query Responsibility Separation)
 - * Event Sourcing
 - * Messaging
 - * Saga

GETTING STARTED

- Quick Start
- TodoList - RECQ Tutorial
 - Problem Description and Requirement Gathering
 - RECQ Payload Design
 - RECQ Components Design
 - Set up your Development Environment
 - RECQ Payload Event Implementation
 - * Domain Commands
 - * Domain Events
 - * Views
 - * Queries
 - RECQ Components Event Implementation with Spring Data
 - * TosoListAggregate
 - * TodoList Model with Spring Data
 - * TodoListProjector
 - * TodoListProjection
 - * TodoList Invoker
 - Expose the RECQ architecture with Spring Web
 - Test Your App
 - Functional Analysis - Event Storming Approach (Bonus Chapter)
- Extend TodoList - Handle Complexity Tutorial

RECQ Patterns

- Introduction
- RECQ System Pattern
 - Component
 - Message Gateway
 - System State Store
- RECQ Communication Pattern
 - Component to Component

- Component to System State Store
 - System State Store to Component
- RECQ Component Pattern
 - Aggregate
 - Projector
 - Projection
 - Service
 - Invoker
 - Saga
 - Observer

Evento Framework

- Introduction
- Payload
 - Command
 - * Domain Command
 - * Service Command
 - Event
 - * Domain Event
 - * ServiceEvent
 - View
 - Query
 - * Query Response
 - Multiple
 - Single
- @Component
 - @Aggregate
 - * Aggregate State
 - * @AggregateCommandHandler
 - * @EventSourcingHandler
 - @Projector
 - * Projector @EventHandler
 - @Projection
 - * @QueryHandler
 - @Service
 - @Invoker
 - @Saga
 - @Observer
- Bundle
 - Consumer State Store
 - Autoscaling Protocol

Evento Server

- Introduction

- Bundle Deploy Plugin
- Message BUS

EVENTO GUI

- Payload Catalog
- Component Catalog
- Bundle Catalog
- Cluster Status
- Flows
 - Performance Evaluation
- Application Graph

Evento CLI

- Update Version
- Publish