
Download Free Pdf Architectures Microservice And Services Restful Applications Web Your Secure Edition Third Security Spring

When somebody should go to the book stores, search introduction by shop, shelf by shelf, it is really problematic. This is why we give the book compilations in this website. It will utterly ease you to look guide **Pdf Architectures Microservice And Services Restful Applications Web Your Secure Edition Third Security Spring** as you such as.

By searching the title, publisher, or authors of guide you essentially want, you can discover them rapidly. In the house, workplace, or perhaps in your method can be all best place within net connections. If you plan to download and install the Pdf Architectures Microservice And Services Restful Applications Web Your Secure Edition Third Security Spring, it is no question easy then, in the past currently we extend the connect to purchase and create bargains to download and install Pdf Architectures Microservice And Services Restful Applications Web Your Secure Edition Third Security Spring consequently simple!

KEY=AND - SHAMAR PETERSEN

Microservices from Theory to Practice: Creating Applications in IBM Bluemix Using the Microservices Approach

IBM Redbooks Microservices is an architectural style in which large, complex software applications are composed of one or more smaller services. Each of these microservices focuses on completing one task that represents a small business capability. These microservices can be developed in any programming language. They communicate with each other using language-neutral protocols, such as Representational State Transfer (REST), or messaging applications, such as IBM® MQ Light. This IBM Redbooks® publication gives a broad understanding of this increasingly popular architectural style, and provides some real-life examples of how you can develop applications using the microservices approach with IBM Bluemix™. The source code for all of these sample scenarios can be found on GitHub (<https://github.com/>). The book also presents some case studies from IBM products. We explain the architectural decisions made, our experiences, and lessons learned when redesigning these products using the microservices approach. Information technology (IT) professionals interested in learning about microservices and how to develop or redesign an application in Bluemix using microservices can benefit from this book.

Microservice Architecture

Aligning Principles, Practices, and Culture

"O'Reilly Media, Inc." Have you heard about the tremendous success Amazon and Netflix have had by switching to a microservice architecture? Are you wondering how this can benefit your company? Or are you skeptical about how it might work? If you've answered yes to any of these questions, this practical book will benefit you. You'll learn how to take advantage of the microservice architectural style for building systems, and learn from the experiences of others to adopt and execute this approach most successfully.

Microservices Patterns

With examples in Java

Simon and Schuster "A comprehensive overview of the challenges teams face when moving to microservices, with industry-tested solutions to these problems." - Tim Moore, Lightbend 44 reusable patterns to develop and deploy reliable production-quality microservices-based applications, with worked examples in Java Key Features 44 design patterns for building and deploying microservices applications Drawing on decades of unique experience from author and microservice architecture pioneer Chris Richardson A pragmatic approach to the benefits and the drawbacks of microservices architecture Solve service decomposition, transaction management, and inter-service communication Purchase of the print book includes a free eBook in PDF, Kindle, and ePub formats from Manning Publications. About The Book Microservices Patterns teaches you 44 reusable patterns to reliably develop and deploy production-quality microservices-based applications. This invaluable set of design patterns builds on decades of distributed system experience, adding new patterns for composing services into systems that scale and perform under real-world conditions. More than just a patterns catalog, this practical guide with worked examples offers industry-tested advice to help you design, implement, test, and deploy your microservices-based application. What You Will Learn How (and why!) to use microservices architecture Service decomposition strategies Transaction management and querying patterns Effective testing strategies Deployment patterns This Book Is Written For Written for enterprise developers familiar with standard enterprise application architecture. Examples are in Java. About The Author Chris Richardson is a Java Champion, a JavaOne rock star, author of Manning's POJOs in Action, and creator of the original CloudFoundry.com. Table of Contents Escaping monolithic hell Decomposition strategies Interprocess communication in a microservice architecture Managing transactions with sagas Designing business logic in a microservice architecture Developing business logic with event sourcing Implementing queries in a microservice architecture External API patterns Testing microservices: part 1 Testing microservices: part 2 Developing production-ready services Deploying microservices Refactoring to microservices

Microservices Best Practices for Java

IBM Redbooks Microservices is an architectural style in which large, complex software applications are composed of one or more smaller services. Each of these microservices focuses on completing one task that represents a small business capability. These microservices can be developed in any programming language. This IBM® Redbooks® publication covers Microservices best practices for Java. It focuses on creating cloud native applications using the latest version of IBM WebSphere® Application Server Liberty, IBM Bluemix® and other Open Source Frameworks in the Microservices ecosystem to highlight Microservices best practices for Java.

Learn Microservices with Spring Boot

A Practical Approach to RESTful Services using RabbitMQ, Eureka, Ribbon, Zuul and Cucumber

Apress Build a microservices architecture with Spring Boot, by evolving an application from a small monolith to an event-driven architecture composed of several services. This book follows an incremental approach to teach microservice structure, test-driven development, Eureka, Ribbon, Zuul, and end-to-end tests with Cucumber. Author Moises Macero follows a very pragmatic approach to explain the benefits of using this type of software architecture, instead of keeping you distracted with theoretical concepts. He covers some of the state-of-the-art techniques in computer programming, from a practical point of view. You'll focus on what's important, starting with the minimum viable product but keeping the flexibility to evolve it. What You'll Learn Build microservices with Spring Boot Use event-driven architecture and messaging with RabbitMQ Create RESTful services with Spring Master service discovery with Eureka and load balancing with Ribbon Route requests with Zuul as your API gateway Write end-to-end tests for an event-driven architecture using Cucumber Carry out continuous integration and deployment Who This Book Is For Those with at least some prior experience with Java programming. Some prior exposure to Spring Boot recommended but not required.

Building Microservices

Designing Fine-Grained Systems

"O'Reilly Media, Inc." Annotation Over the past 10 years, distributed systems have become more fine-grained. From the large multi-million line long monolithic applications, we are now seeing the benefits of smaller self-contained services. Rather than heavy-weight, hard to change Service Oriented Architectures, we are now seeing systems consisting of collaborating microservices. Easier to change, deploy, and if required retire, organizations which are in the right position to take advantage of them are yielding significant benefits. This book takes an holistic view of the things you need to be cognizant of in order to pull this off. It covers just enough understanding of technology, architecture, operations and organization to show you how to move towards finer-grained systems.

Data Management at Scale

"O'Reilly Media, Inc." As data management and integration continue to evolve rapidly, storing all your data in one place, such as a data warehouse, is no longer scalable. In the very near future, data will need to be distributed and available for several technological solutions. With this practical book, you'll learn how to migrate your enterprise from a complex and tightly coupled data landscape to a more flexible architecture ready for the modern world of data consumption. Executives, data architects, analytics teams, and compliance and governance staff will learn how to build a modern scalable data landscape using the Scaled Architecture, which you can introduce incrementally without a large upfront investment. Author Piethein Strengholt provides blueprints, principles, observations, best practices, and patterns to get you up to speed. Examine data management trends, including technological developments, regulatory requirements, and privacy concerns Go deep into the Scaled Architecture and learn how the pieces fit together Explore data governance and data security, master data management, self-service data marketplaces, and the importance of metadata

Clean Architecture

A Craftsman's Guide to Software Structure and Design

Prentice Hall Practical Software Architecture Solutions from the Legendary Robert C. Martin ("Uncle Bob") By applying universal rules of software architecture, you can dramatically improve developer productivity throughout the life of any software system. Now, building upon the success of his best-selling books Clean Code and The Clean Coder, legendary software craftsman Robert C. Martin ("Uncle Bob") reveals those rules and helps you apply them. Martin's Clean Architecture doesn't merely present options. Drawing on over a half-century of experience in software environments of every imaginable type, Martin tells you what choices to make and why they are critical to your success. As you've come to expect from Uncle Bob, this book is packed with direct, no-nonsense solutions for the real challenges you'll face—the ones that will make or break your projects. Learn what software architects need to achieve—and core disciplines and practices for achieving it Master essential software design principles for addressing function, component separation, and data management See how programming paradigms impose discipline by restricting what developers can do Understand what's critically important and what's merely a "detail" Implement optimal, high-level structures for web, database, thick-client, console, and embedded applications Define appropriate boundaries and layers, and organize components and services See why designs and architectures go wrong, and how to prevent (or fix) these failures Clean Architecture is essential reading for every current or aspiring software architect, systems analyst, system designer, and software manager—and for every programmer who must execute someone else's designs. Register your product for convenient access to downloads, updates, and/or corrections as they become available.

Essentials of Microservices Architecture

Paradigms, Applications, and Techniques

Taylor & Francis Microservices architecture (MSA) is increasingly popular with software architects and engineers as it accelerates software solution design, development, and deployment in a risk-free manner. Placing a software system into a production environment is elegantly simplified and sped up with the use of MSA development platforms, runtime environments, acceleration engines, design patterns, integrated frameworks, and related tools. The MSA ecosystem is expanding with third-party products that automate as many tasks as possible. MSA is being positioned as the enterprise-grade and agile-application design method. This book covers in-depth the features and facilities that make up the MSA ecosystem. Beginning with an overview of Service-Oriented Architecture (SOA) that covers the Common Object Request Broker Architecture (CORBA), Distributed Component Object Model (DCOM), and Remote Method Invocation (RMI), the book explains the basic essentials of MSA and the continuous delivery of applications to customers. The book gives software developers insight into: Current and emerging communication models Key architectural elements of MSA-based applications Designing efficient APIs for microservices MSA middleware platforms such as REST, SOAP, Apache Thrift, and gRPC Microservice discovery and the API gateway Service orchestration and choreography for composing individual services to achieve a useful business process Database transactions in MSA-centric applications Design, composition, security, and deployment patterns MSA security Modernizing legacy applications The book concludes with a chapter on composing and building powerful microservices. With the exponential growth of IoT devices, microservices are being developed and deployed on resource-constrained but resource-intensive devices in order to provide people-centric applications. The book discusses the challenges of these applications. Finally, the book looks at the role of microservices in smart environments and upcoming trends including ubiquitous yet disappearing microservices.

Mastering Microservices with Java 9

Build domain-driven microservice-based applications with Spring, Spring Cloud, and Angular

Packt Publishing Ltd Master the art of implementing scalable microservices in your production environment with ease About This Book Use domain-driven design to build microservices Use Spring Cloud to use Service Discovery and Registration Use Kafka, Avro and Spring Streams for implementing event based microservices Who This Book Is For This book is for Java developers who are familiar with the microservices architecture and now wants to take a deeper dive into effectively implementing microservices at an enterprise level. A reasonable knowledge level and understanding of core microservice elements and applications is expected. What You Will Learn Use domain-driven design to design and implement microservices Secure microservices using Spring Security Learn to develop REST service development Deploy and test microservices Troubleshoot and debug the issues faced during development Learning best practices and common principals about microservices In Detail Microservices are the next big thing in designing scalable, easy-to-maintain applications. It not only makes app development easier, but also offers great flexibility to utilize various resources optimally. If you want to build an enterprise-ready implementation of the microservices architecture, then this is the book for you! Starting off by understanding the core concepts and framework, you will then focus on the high-level design of large software projects. You will gradually move on to setting up the development environment and configuring it before implementing continuous integration to deploy your microservice architecture. Using Spring security, you will secure microservices and test them effectively using REST Java clients and other tools like Rxjava 2.0. We'll show you the best patterns, practices and common principals of microservice design and you'll learn to troubleshoot and debug the issues faced during development. We'll show you how to design and implement reactive microservices. Finally, we'll show you how to migrate a monolithic application to microservices based application. By the end of the book, you will know how to build smaller, lighter, and faster services that can be implemented easily in a production environment. Style and approach This book starts from the basics, including environment setup and provides easy-to-follow steps to implement the sample project using microservices.

REST API Design Rulebook

O'Reilly Media, Inc. The basic rules of REST APIs - "many nouns, few verbs, stick with HTTP" - seem easy, but that simplicity and power require discipline to work smoothly. This brief guide provides next steps for implementing complex projects on simple and extensible foundations.

Spring Microservices

Packt Publishing Ltd Build scalable microservices with Spring, Docker, and Mesos About This Book Learn how to efficiently build and implement microservices in Spring, and how to use Docker and Mesos to push the boundaries of what you thought possible Examine a number of real-world use cases and hands-on code examples. Distribute your microservices in a completely new way Who This Book Is For If you are a Spring developers and want to build cloud-ready, internet-scale applications to meet modern business demands, then this book is for you Developers will understand how to build simple Restful services and organically grow them to truly enterprise grade microservices ecosystems. What You Will Learn Get to know the microservices development lifecycle process See how to implement microservices governance Familiarize yourself with the microservices architecture and its benefits Use Spring Boot to develop microservices Find out how to avoid common pitfalls when developing microservices Be introduced to end-to-end microservices written in Spring Framework and Spring Boot In Detail The Spring Framework is an application framework and inversion of the control container for the Java platform. The framework's core features can be used by any Java application, but there are extensions to build web applications on top of the Java EE platform. This book will help you implement the microservice architecture in Spring Framework, Spring Boot, and Spring Cloud. Written to the latest specifications of Spring, you'll be able to build modern, Internet-scale Java applications in no time. We would start off with the guidelines to implement responsive microservices at scale. We will then deep dive into Spring Boot, Spring Cloud, Docker, Mesos, and Marathon. Next you will understand how Spring Boot is used to deploy autonomous services, server-less by removing the need to have a heavy-weight application server. Later you will learn how to go further by deploying your microservices to Docker and manage it with Mesos. By the end of the book, you'll will gain more clarity on how to implement microservices using Spring Framework and use them in Internet-scale deployments through real-world examples. Style and approach The book follows a step by step approach on how to develop microservices using Spring Framework, Spring Boot, and a set of Spring Cloud components that will help you scale your applications.

Evolve the Monolith to Microservices with Java and Node

IBM Redbooks Microservices is an architectural style in which large, complex software applications are composed of one or more smaller services. Each of these microservices focuses on completing one task that represents a small business capability. These microservices can be developed in any programming language. This IBM® Redbooks® publication shows how to break out a traditional Java EE application into separate microservices and provides a set of code projects that illustrate the various steps along the way. These code projects use the IBM WebSphere® Application Server Liberty, IBM API Connect™, IBM Bluemix®, and other Open Source Frameworks in the microservices ecosystem. The sample projects highlight the evolution of monoliths to microservices with Java and Node.

Production-Ready Microservices

Building Standardized Systems Across an Engineering Organization

"O'Reilly Media, Inc." One of the biggest challenges for organizations that have adopted microservice architecture is the lack of architectural, operational, and organizational standardization. After splitting a monolithic application or building a microservice ecosystem from scratch, many engineers are left wondering what's next. In this practical book, author Susan Fowler presents a set of microservice standards in depth, drawing from her experience standardizing over a thousand microservices at Uber. You'll learn how to design microservices that are stable, reliable, scalable, fault tolerant, performant, monitored, documented, and prepared for any catastrophe. Explore production-readiness standards, including: Stability and Reliability: develop, deploy, introduce, and deprecate microservices; protect against dependency failures Scalability and Performance: learn essential components for achieving greater microservice efficiency Fault Tolerance and Catastrophe Preparedness: ensure availability by actively pushing microservices to fail in real time Monitoring: learn how to monitor, log, and display key metrics; establish alerting and on-call procedures Documentation and Understanding: mitigate tradeoffs that come with microservice adoption, including organizational sprawl and technical debt

Python Microservices Development

Build efficient and lightweight microservices using the Python tooling ecosystem, 2nd Edition

Packt Publishing Ltd Use Python microservices to craft applications that are built as small standard units using proven best practices and avoiding common errors Key Features Become well versed with the fundamentals of building, designing, testing, and deploying Python microservices Identify where a monolithic application can be split, how to secure it, and how to scale it once ready for deployment Use the latest framework based on asynchronous programming to write effective microservices with Python Book Description The small scope and self-contained nature of microservices make them faster, cleaner, and more scalable than code-heavy monolithic applications. However, building microservices architecture that is efficient as well as lightweight into your applications can be challenging due to the complexity of all the interacting pieces. Python Microservices Development, Second Edition will teach you how to overcome these issues and craft applications that are built as small standard units using proven best practices and avoiding common pitfalls. Through hands-on examples, this book will help you to build efficient microservices using Quart, SQLAlchemy, and other modern Python tools In this updated edition, you will learn how to secure connections between services and how to script Nginx using Lua to build web application firewall features such as rate limiting. Python Microservices Development, Second Edition describes how to use containers and AWS to deploy your services. By the end of the book, you'll have created a complete Python application based on microservices. What you will learn Explore what microservices are and how to design them Configure and package your code according to modern best practices Identify a component of a larger service that can be turned into a microservice Handle more incoming requests, more effectively Protect your application with a proxy or firewall Use Kubernetes and containers to deploy a microservice Make changes to an API provided by a microservice safely and keep things working Identify the factors to look for to get started with an unfamiliar cloud provider Who this book is for This book is for developers who want to learn how to build, test, scale, and manage Python microservices. Readers will require basic knowledge of the Python programming language, the command line, and HTTP-based application principles. No prior experience of writing microservices in Python is assumed.

EG-ICE 2021 Workshop on Intelligent Computing in Engineering

Universitätsverlag der TU Berlin The 28th EG-ICE International Workshop 2021 brings together international experts working at the interface between advanced computing and modern engineering challenges. Many engineering tasks require open-world resolutions to support multi-actor collaboration, coping with approximate models, providing effective engineer-computer interaction, search in multi-dimensional solution spaces, accommodating uncertainty, including specialist domain knowledge, performing sensor-data interpretation and dealing with incomplete knowledge. While results from computer science provide much initial support for resolution, adaptation is unavoidable and most importantly, feedback from addressing engineering challenges drives fundamental computer-science research. Competence and knowledge transfer goes both ways. Der 28. Internationale EG-ICE Workshop 2021 bringt internationale Experten zusammen, die an der Schnittstelle zwischen fortgeschrittener Datenverarbeitung und modernen technischen Herausforderungen arbeiten. Viele ingenieurwissenschaftliche Aufgaben erfordern Open-World-Resolutionen, um die Zusammenarbeit mehrerer Akteure zu unterstützen, mit approximativen Modellen umzugehen, eine effektive Interaktion zwischen Ingenieur und Computer zu ermöglichen, in mehrdimensionalen Lösungsräumen zu suchen, Unsicherheiten zu berücksichtigen, einschließlich fachspezifischen Domänenwissens, Sensordateninterpretation durchzuführen und mit unvollständigem Wissen umzugehen. Während die Ergebnisse aus der Informatik anfänglich viel Unterstützung für die Lösung bieten, ist eine Anpassung unvermeidlich, und am wichtigsten ist, dass das Feedback aus der Bewältigung technischer Herausforderungen die computer-wissenschaftliche Grundlagenforschung vorantreibt. Kompetenz und Wissenstransfer gehen in beide Richtungen.

Managing Software Crisis: A Smart Way to Enterprise Agility

Springer This book discusses smart, agile software development methods and their applications for enterprise crisis management, presenting a systematic approach that promotes agility and crisis management in software engineering. The key finding is that these crises are caused by both technology-based and human-related factors. Being mission-critical, human-related issues are often neglected. To manage the crises, the book suggests an efficient agile methodology including a set of models, methods, patterns, practices and tools. Together, these make a survival toolkit for large-scale software development in crises. Further, the book analyses lifecycles and methodologies focusing on their impact on the project timeline and budget, and incorporates a set of industry-based patterns, practices and case studies, combining academic concepts and practices of software engineering.

Monolith to Microservices

Evolutionary Patterns to Transform Your Monolith

"O'Reilly Media, Inc." How do you detangle a monolithic system and migrate it to a microservice architecture? How do you do it while maintaining business-as-usual? As a companion to Sam Newman's extremely popular Building Microservices, this new book details a proven method for transitioning an existing monolithic system to a microservice architecture. With many illustrative examples, insightful migration patterns, and a bevy of practical advice to transition your monolith enterprise into a microservice operation, this practical guide covers multiple scenarios and strategies for a successful migration, from initial planning all the way through application and database decomposition. You'll learn several tried and tested patterns and techniques that you can use as you migrate your existing architecture. Ideal for organizations looking to transition to microservices, rather than rebuild Helps companies determine whether to migrate, when to migrate, and where to begin Addresses communication, integration, and the migration of legacy systems Discusses multiple migration patterns and where they apply Provides database migration examples, along with synchronization strategies Explores application decomposition, including several architectural refactoring patterns Delves into details of database decomposition, including the impact of breaking referential and transactional integrity, new failure modes, and more

Microservices

Science and Engineering

Springer Nature This book describes in contributions by scientists and practitioners the development of scientific concepts, technologies, engineering techniques and tools for a service-based society. The focus is on microservices, i.e cohesive, independent processes deployed in isolation and equipped with dedicated memory persistence tools, which interact via messages. The book is structured in six parts. Part 1 "Opening" analyzes the new (and old) challenges including service design and specification, data integrity, and consistency management and provides the introductory information needed to successfully digest the remaining parts. Part 2 "Migration" discusses the issue of migration from monoliths to microservices and their loosely coupled architecture. Part 3 "Modeling" introduces a catalog and a taxonomy of the most common microservices anti-patterns and identifies common problems. It also explains the concept of RESTful conversations and presents insights from studying and developing two further modeling approaches. Next, Part 4 is dedicated to various aspects of "Development and Deployment". Part 5 then covers "Applications" of microservices, presenting case studies from Industry 4.0, Netflix, and customized SaaS examples. Eventually, Part 6 focuses on "Education" and reports on experiences made in special programs, both at academic level as a master program course and for practitioners in an industrial training. As only a joint effort between academia and industry can lead to the release of modern paradigm-based programming languages, and subsequently to the deployment of robust and scalable software systems, the book mainly targets researchers in academia and industry who develop tools and applications for microservices.

Java EE 8 Design Patterns and Best Practices

Build enterprise-ready scalable applications with architectural design patterns

Packt Publishing Ltd Get the deep insights you need to master efficient architectural design considerations and solve common design problems in your enterprise applications. Key Features The benefits and applicability of using different design patterns in JAVA EE Learn best practices to solve common design and architectural challenges Choose the right patterns to improve the efficiency of your programs Book Description Patterns are essential design tools for Java developers. Java EE Design Patterns and Best Practices helps developers attain better code quality and progress to higher levels of architectural creativity by examining the purpose of each available pattern and demonstrating its implementation with various code examples. This book will take you through a number of patterns and their Java EE-specific implementations. In the beginning, you will learn the foundation for, and importance of, design patterns in Java EE, and then will move on to implement various patterns on the presentation tier, business tier, and integration tier. Further, you will explore the patterns involved in Aspect-Oriented Programming (AOP) and take a closer look at reactive patterns. Moving on, you will be introduced to modern architectural patterns involved in composing microservices and cloud-native applications. You will get acquainted with security patterns and operational patterns involved in scaling

and monitoring, along with some patterns involved in deployment. By the end of the book, you will be able to efficiently address common problems faced when developing applications and will be comfortable working on scalable and maintainable projects of any size. What you will learn Implement presentation layers, such as the front controller pattern Understand the business tier and implement the business delegate pattern Master the implementation of AOP Get involved with asynchronous EJB methods and REST services Involve key patterns in the adoption of microservices architecture Manage performance and scalability for enterprise-level applications Who this book is for Java developers who are comfortable with programming in Java and now want to learn how to implement design patterns to create robust, reusable and easily maintainable apps.

Getting Started with IBM API Connect: Scenarios Guide

IBM Redbooks IBM® API Connect is an API management solution from IBM that offers capabilities to create, run, manage, and secure APIs and microservices. By using these capabilities, the full lifecycle of APIs for on-premises and cloud environments can be managed. This IBM Redpaper™ publication describes practical scenarios that show the API Connect capabilities for managing the full API life cycle, creating, running, securing, and managing the APIs. This Redpaper publication is targeted to users of an API Connect based API strategy, developers, IT architects, and technical evangelists. If you are not familiar with APIs or API Connect, we suggest that you read the Redpaper publication Getting Started with IBM API Connect: Concepts, Architecture and Strategy Guide, REDP-5349, before reading this publication.

Building Modern Web Applications With Jakarta EE, NoSQL Databases and Microservices

Create Web Applications Jakarta EE with Microservices, JNoSQL, Vaadin, Jmoordb, and MicroProfile easily

BPB Publications Build Modern Web Apps with JakartaEE, Jmoordb, and Vaadins Key Features ● Learn about the Java Enterprise Edition/Jakarta Enterprise Edition specifications. ● Learn how to create applications with frameworks such as Java Server Faces, Eclipse krazo and Vaadin. ● Get familiar with NoSQL databases and learn how to create Java applications that interact using Jakarta NoSQL and Jmoordb. ● Learn how to test and secure your application. ● Learn about Microprofile and how to create microservices with java. Description For many years, Java EE has been an important platform for mission-critical enterprise applications. To accelerate the development of enterprise applications for a cloud-native world, leading software vendors collaborated to transfer Java EE technologies to the Eclipse Foundation, where they will evolve under the Jakarta EE brand. This book will be your comprehensive guide to creating Jakarta EE applications and microservices with Microprofile. The book begins with an introduction to Jakarta EE and quickly goes on to teach you about the various databases and their advantages. After this, you will explore the JNoSQL and Jmoordb frameworks to understand how to build Jakarta EE applications with NoSQL databases. Moving forward, you'll explore Eclipse MicroProfile and see how it helps build microservices with Java. Also, you will learn about various development applications such as Java Server Faces, Eclipse Krazos, PrimeFaces, Vaadin, and understand how to integrate them with your backend. Towards the end, you will learn about security, testing, and understanding continuous integration. What will you learn ● Learn how to use the Jmoordb framework for Jakarta EE applications. ● Optimize Enterprise Java for microservices architecture using Eclipse MicroProfile. ● Create Web applications using Java Server Faces. ● Building a modern web application using Vaadin. ● Learn how to implement security using IdentityStore and JWT. ● Create CI/CD pipelines for Jakarta EE applications. Who this book is for This book is for developers with no previous experience in creating business applications with Java and for those who want to know about APIs and new frameworks for the development of cloud-oriented applications. Table of Contents 1. Jakarta EE Platform 2. NoSQL 3. Jakarta NOSQL 4. Understanding JMoordb 5. Exploring Microprofile 6. Java Server Faces 7. Vaadin 8. Integration Vaadin, JMoordb and NoSQL 9. Eclipse Krazos and Security of Microservices 10. Testing and Continuous Integration

RESTful Web APIs

Services for a Changing World

"O'Reilly Media, Inc." The popularity of REST in recent years has led to tremendous growth in almost-RESTful APIs that don't include many of the architecture's benefits. With this practical guide, you'll learn what it takes to design usable REST APIs that evolve over time. By focusing on solutions that cross a variety of domains, this book shows you how to create powerful and secure applications, using the tools designed for the world's most successful distributed computing system: the World Wide Web. You'll explore the concepts behind REST, learn different strategies for creating hypermedia-based APIs, and then put everything together with a step-by-step guide to designing a RESTful Web API. Examine API design strategies, including the collection pattern and pure hypermedia. Understand how hypermedia ties representations together into a coherent API. Discover how XMDP and ALPS profile formats can help you meet the Web API "semantic challenge". Learn close to two-dozen standardized hypermedia data formats. Apply best practices for using HTTP in API implementations. Create Web APIs with the JSON-LD standard and other the Linked Data approaches. Understand the CoAP protocol for using REST in embedded systems.

Accelerating Modernization with Agile Integration

IBM Redbooks The organization pursuing digital transformation must embrace new ways to use and deploy integration technologies, so they can move quickly in a manner appropriate to the goals of multicloud, decentralization, and microservices. The integration layer must transform to allow organizations to move boldly in building new customer experiences, rather than forcing models for architecture and development that pull away from maximizing the organization's productivity. Many organizations have started embracing agile application techniques, such as microservice architecture, and are now seeing the benefits of that shift. This approach complements and accelerates an enterprise's API strategy. Businesses should also seek to use this approach to modernize their existing integration and messaging infrastructure to achieve more effective ways to manage and operate their integration services in their private or public cloud. This IBM® Redbooks® publication explores the merits of what we refer to as agile integration; a container-based, decentralized, and microservice-aligned approach for integration solutions that meets the demands of agility, scalability, and resilience required by digital transformation. It also discusses how the IBM Cloud Pak for Integration marks a significant leap forward in integration technology by embracing both a cloud-native approach and container technology to achieve the goals of agile integration. The target audiences for this book are cloud integration architects, IT specialists, and application developers.

REST in Practice

Hypermedia and Systems Architecture

"O'Reilly Media, Inc." REST continues to gain momentum as the best method for building Web services, and this down-to-earth book delivers techniques and examples that show how to design and implement integration solutions using the REST architectural style.

Pro RESTful APIs

Design, Build and Integrate with REST, JSON, XML and JAX-RS

Apress Discover the RESTful technologies, including REST, JSON, XML, JAX-RS web services, SOAP and more, for building today's microservices, big data applications, and web service applications. This book is based on a course the Oracle-based author is teaching for UC Santa Cruz Silicon Valley which covers architecture, design best practices and coding labs. Pro RESTful APIs: Design gives you all the fundamentals from the top down: from the top (architecture) through the middle (design) to the bottom (coding). This book is a must have for any microservices or web services developer building applications and services. What You'll Learn Discover the key RESTful APIs, including REST, JSON, XML, JAX, SOAP and more Use these for web services and data exchange, especially in today's big data context Harness XML, JSON, REST, and JAX-RS in examples and case studies Apply best practices to your solutions' architecture Who This Book Is For Experienced web programmers and developers.

Pro RESTful APIs

Design, Build and Integrate with REST, JSON, XML and JAX-RS

Apress Discover the RESTful technologies, including REST, JSON, XML, JAX-RS web services, SOAP and more, for building today's microservices, big data applications, and web service applications. This book is based on a course the Oracle-based author is teaching for UC Santa Cruz Silicon Valley which covers architecture, design best practices and coding labs. Pro RESTful APIs: Design gives you all the fundamentals from the top down: from the top (architecture) through the middle (design) to the bottom (coding). This book is a must have for any microservices or web services developer building applications and services. What You'll Learn Discover the key RESTful APIs, including REST, JSON, XML, JAX, SOAP and more Use these for web services and data exchange, especially in today's big data context Harness XML, JSON, REST, and JAX-RS in examples and case studies Apply best practices to your solutions' architecture Who This Book Is For Experienced web programmers and developers.

Learn Microservices with Spring Boot 3

A Practical Approach to RESTful Services Using RabbitMQ, Eureka, Ribbon, Zuul, and Cucumber

Apress Build Java-based microservices architecture using the Spring Boot 3 framework by evolving an application from a small monolith to an event-driven architecture composed of several services. This revised book follows an incremental approach in teaching the structure of microservices, test-driven development, Eureka, Ribbon, Zuul, and end-to-end tests with Cucumber. This updated book now covers what's been added to the new Spring Boot 3 release, including support for the latest Java SE LTS; changes to the Stream Editor UI; Maven preemptive authentication; building Docker images using cloud-native build packs; building layered jars for optimized Docker images; E2E traceability for configuration properties; many dependency upgrades; support for Spring Data Neumann; and more. Author Moises Macero uses a pragmatic approach to explain the benefits of using this type of software architecture, instead of keeping you distracted with theoretical concepts. He covers some of the state-of-the-art techniques in computer programming, from a practical point of view. You'll focus on what's important, starting with the minimum viable product but keeping the flexibility to evolve it. What You Will Learn Build microservices with Spring Boot 3 Use event-driven architecture and messaging with RabbitMQ Master service discovery with Eureka and load balancing with Ribbon Route requests with Zuul as your API gateway Write end-to-end tests for an event-driven architecture using Cucumber Carry out continuous integration and deployment Who This Book Is For Those with at least some prior experience with Java programming. Some prior exposure to Spring Boot recommended but not required.

Python Microservices Development

Packt Publishing Ltd A practical approach to conquering the complexities of Microservices using the Python tooling ecosystem About This Book A very useful guide for Python developers who are shifting to the new microservices-based development A concise, up-to-date guide to building efficient and lightweight microservices in Python using Flask, Tox, and other tools Learn to use Docker containers, CoreOS, and Amazon Web Services to deploy your services Who This Book Is For This book is for developers who have basic knowledge of Python, the command line, and HTTP-based application principles, and those who want to learn how to build, test, scale, and manage Python 3 microservices. No prior experience of writing microservices in Python is assumed. What You Will Learn Explore what microservices are and how to design them Use Python 3, Flask, Tox, and other tools to build your services using best practices Learn how to use a TDD approach Discover how to document your microservices Configure and package your code in the best way Interact with other services Secure, monitor, and scale your services Deploy your services in Docker containers, CoreOS, and Amazon Web Services In Detail We often deploy our web applications into the cloud, and our code needs to interact with many third-party services. An efficient way to build applications to do this is through microservices architecture. But, in practice, it's hard to get this right due to the complexity of all the pieces interacting with each other. This book will teach you how to overcome these issues and craft applications that are built as small standard units, using all the proven best practices and avoiding the usual traps. It's a practical book: you'll build everything using Python 3 and its amazing tooling ecosystem. You will understand the principles of TDD and apply them. You will use Flask, Tox, and other tools to build your services using best practices. You will learn how to secure connections between services, and how to script Nginx using Lua to build web application firewall features such as rate limiting. You will also familiarize yourself with Docker's role in microservices, and use Docker containers.

CoreOS, and Amazon Web Services to deploy your services. This book will take you on a journey, ending with the creation of a complete Python application based on microservices. By the end of the book, you will be well versed with the fundamentals of building, designing, testing, and deploying your Python microservices. Style and approach This book is an linear, easy-to-follow guide on how to best design, write, test, and deploy your microservices. It includes real-world examples that will help Python developers create their own Python microservice using the most efficient methods.

Spring REST

Building Java Microservices and Cloud Applications

Apress Design and develop Java-based RESTful APIs using the latest versions of the Spring MVC and Spring Boot frameworks. This book walks you through the process of designing and building a REST application while delving into design principles and best practices for versioning, security, documentation, error handling, paging, and sorting. Spring REST provides a brief introduction to REST, HTTP, and web infrastructure. You will learn about several Spring projects such as Spring Boot, Spring MVC, Spring Data JPA, and Spring Security, and the role they play in simplifying REST application development. You will learn how to build clients that consume REST services. Finally, you will learn how to use the Spring MVC test framework to unit test and integration test your REST API. After reading this book, you will come away with all the skills to build sophisticated REST applications using Spring technologies. What You Will Learn Build Java-based microservices, native cloud, or any applications using Spring REST Employ Spring MVC and RESTful Spring Build a QuickPoll application example Document REST services, as well as versioning, paging, and sorting Test, handle errors and secure your application Who This Book Is For Intermediate Java programmers with at least some prior experience with Spring and web/cloud application development.

Spring Security

Secure your web applications, RESTful services, and microservice architectures

Packt Publishing Ltd Learn how to secure your Java applications from hackers using Spring Security 4.2 About This Book Architect solutions that leverage the full power of Spring Security while remaining loosely coupled. Implement various scenarios such as supporting existing user stores, user sign up, authentication, and supporting AJAX requests, Integrate with popular Microservice and Cloud services such as Zookeeper, Eureka, and Consul, along with advanced techniques, including OAuth, JSON Web Token's (JWT), Hashing, and encryption algorithms Who This Book Is For This book is intended for Java Web and/or RESTful webservice developers and assumes a basic understanding of creating Java 8, Java Web and/or RESTful webservice applications, XML, and the Spring Framework. You are not expected to have any previous experience with Spring Security. What You Will Learn Understand common security vulnerabilities and how to resolve them Learn to perform initial penetration testing to uncover common security vulnerabilities Implement authentication and authorization Learn to utilize existing corporate infrastructure such as LDAP, Active Directory, Kerberos, CAS, OpenID, and OAuth Integrate with popular frameworks such as Spring, Spring-Boot, Spring-Data, JSF, Vaadin, jQuery, and AngularJS. Gain deep understanding of the security challenges with RESTful webservices and microservice architectures Integrate Spring with other security infrastructure components like LDAP, Apache Directory server and SAML In Detail Knowing that experienced hackers are itching to test your skills makes security one of the most difficult and high-pressured concerns of creating an application. The complexity of properly securing an application is compounded when you must also integrate this factor with existing code, new technologies, and other frameworks. Use this book to easily secure your Java application with the tried and trusted Spring Security framework, a powerful and highly customizable authentication and access-control framework. The book starts by integrating a variety of authentication mechanisms. It then demonstrates how to properly restrict access to your application. It also covers tips on integrating with some of the more popular web frameworks. An example of how Spring Security defends against session fixation, moves into concurrency control, and how you can utilize session management for administrative functions is also included. It concludes with advanced security scenarios for RESTful webservices and microservices, detailing the issues surrounding stateless authentication, and demonstrates a concise, step-by-step approach to solving those issues. And, by the end of the book, readers can rest assured that integrating version 4.2 of Spring Security will be a seamless endeavor from start to finish. Style and approach This practical step-by-step tutorial has plenty of example code coupled with the necessary screenshots and clear narration so that grasping content is made easier and quicker.

Building Microservices with Go

Your one-stop guide to the common patterns and practices, showing you how to apply these using the Go programming language
About This Book* This short, concise, and practical guide is packed with real-world examples of building microservices with Go* It is easy to read and will benefit smaller teams who want to extend the functionality of their existing systems* Using this practical approach will save your money in terms of maintaining a monolithic architecture and demonstrate capabilities in ease of use
Who This Book Is For You should have a working knowledge of programming in Go, including writing and compiling basic applications. However, no knowledge of RESTful architecture, microservices, or web services is expected. If you are looking to apply techniques to your own projects, taking your first steps into microservice architecture, this book is for you.
What You Will Learn* Plan a microservice architecture and design a microservice* Write a microservice with a RESTful API and a database* Understand the common idioms and common patterns in microservices architecture* Leverage tools and automation that helps microservices become horizontally scalable* Get a grounding in containerization with Docker and Docker-Compose, which will greatly accelerate your development lifecycle* Manage and secure Microservices at scale with monitoring, logging, service discovery, and automation* Test microservices and integrate API tests in Go
Detail Microservice architecture is sweeping the world as the de facto pattern to build web-based applications. Golang is a language particularly well suited to building them. Its strong community, encouragement of idiomatic style, and statically-linked binary artifacts make integrating it with other technologies and managing microservices at scale consistent and intuitive. This book will teach you the common patterns and practices, showing you how to apply these using the Go programming language. It will teach you the fundamental concepts of architectural design and RESTful communication, and show you patterns that provide manageable code that is supportable in development and at scale in production. We will provide you with examples on how to put these concepts and patterns into practice with Go. Whether you are planning a new application or working in an existing monolith, this book will explain and illustrate with practical examples how teams of all sizes can start solving problems with microservices. It will help you understand Docker and Docker-Compose and how it can be used to isolate microservice dependencies and build environments. We finish off by showing you various techniques to monitor, test, and secure your microservices. By the end, you will know the benefits of system resilience of a microservice and the advantages of Go stack.
Style and approach The step-by-step tutorial focuses on building microservices. Each chapter expands upon the previous one, teaching you the main skills and techniques required to be a successful microservice practitioner.

Implementation Patterns

Pearson Education Software Expert Kent Beck Presents a Catalog of Patterns Infinitely Useful for Everyday Programming Great code doesn't just function: it clearly and consistently communicates your intentions, allowing other programmers to understand your code, rely on it, and modify it with confidence. But great code doesn't just happen. It is the outcome of hundreds of small but critical decisions programmers make every single day. Now, legendary software innovator Kent Beck—known worldwide for creating Extreme Programming and pioneering software patterns and test-driven development—focuses on these critical decisions, unearthing powerful “implementation patterns” for writing programs that are simpler, clearer, better organized, and more cost effective. Beck collects 77 patterns for handling everyday programming tasks and writing more readable code. This new collection of patterns addresses many aspects of development, including class, state, behavior, method, collections, frameworks, and more. He uses diagrams, stories, examples, and essays to engage the reader as he illuminates the patterns. You'll find proven solutions for handling everything from naming variables to checking exceptions.

Microservices for the Enterprise

Designing, Developing, and Deploying

Apress Understand the key challenges and solutions around building microservices in the enterprise application environment. This book provides a comprehensive understanding of microservices architectural principles and how to use microservices in real-world scenarios. Architectural challenges using microservices with service integration and API management are presented and you learn how to eliminate the use of centralized integration products such as the enterprise service bus (ESB) through the use of composite/integration microservices. Concepts in the book are supported with use cases, and emphasis is put on the reality that most of you are implementing in a “brownfield” environment in which you must implement microservices alongside legacy applications with minimal disruption to your business. Microservices for the Enterprise covers state-of-the-art techniques around microservices messaging, service development and description, service discovery, governance, and data management technologies and guides you through the microservices design process. Also included is the importance of organizing services as core versus atomic, composite versus integration, and API versus edge, and how such organization helps to eliminate the use of a central ESB and expose services through an API gateway. What You'll Learn Design and develop microservices architectures with

confidence Put into practice the most modern techniques around messaging technologies Apply the Service Mesh pattern to overcome inter-service communication challenges Apply battle-tested microservices security patterns to address real-world scenarios Handle API management, decentralized data management, and observability Who This Book Is For Developers and DevOps engineers responsible for implementing applications around a microservices architecture, and architects and analysts who are designing such systems

Hands-On RESTful API Design Patterns and Best Practices

Design, develop, and deploy highly adaptable, scalable, and secure RESTful web APIs

Packt Publishing Ltd REST architecture (style) is a pivot of distributed systems, simplify data integration amongst modern and legacy applications leverages through the RESTful paradigm. This book is fully loaded with many RESTful API patterns, samples, hands-on implementations and also discuss the capabilities of many REST API frameworks for Java, Scala, Python and Go

Cloud Native Java

Designing Resilient Systems with Spring Boot, Spring Cloud, and Cloud Foundry

"*O'Reilly Media, Inc.*" What separates the traditional enterprise from the likes of Amazon, Netflix, and Etsy? Those companies have refined the art of cloud native development to maintain their competitive edge and stay well ahead of the competition. This practical guide shows Java/JVM developers how to build better software, faster, using Spring Boot, Spring Cloud, and Cloud Foundry. Many organizations have already waded into cloud computing, test-driven development, microservices, and continuous integration and delivery. Authors Josh Long and Kenny Bastani fully immerse you in the tools and methodologies that will help you transform your legacy application into one that is genuinely cloud native. In four sections, this book takes you through: The Basics: learn the motivations behind cloud native thinking; configure and test a Spring Boot application; and move your legacy application to the cloud Web Services: build HTTP and RESTful services with Spring; route requests in your distributed system; and build edge services closer to the data Data Integration: manage your data with Spring Data, and integrate distributed services with Spring's support for event-driven, messaging-centric architectures Production: make your system observable; use service brokers to connect stateful services; and understand the big ideas behind continuous delivery

Practical Microservices Architectural Patterns

Event-Based Java Microservices with Spring Boot and Spring Cloud

Apress Take your distributed applications to the next level and see what the reference architectures associated with microservices can do for you. This book begins by showing you the distributed computing architecture landscape and provides an in-depth view of microservices architecture. Following this, you will work with CQRS, an essential pattern for microservices, and get a view of how distributed messaging works. Moving on, you will take a deep dive into Spring Boot and Spring Cloud. Coming back to CQRS, you will learn how event-driven microservices work with this pattern, using the Axon 2 framework. This takes you on to how transactions work with microservices followed by advanced architectures to address non-functional aspects such as high availability and scalability. In the concluding part of the book you develop your own enterprise-grade microservices application using the Axon framework and true BASE transactions, while making it as secure as possible. What You Will Learn Shift from monolith architecture to microservices Work with distributed and ACID transactions Build solid architectures without two-phase commit transactions Discover the high availability principles in microservices Who This Book Is For Java developers with basic knowledge of distributed and multi-threaded application architecture, and no knowledge of Spring Boot or Spring Cloud. Knowledge of CQRS and event-driven architecture is not mandatory as this book will cover these in depth.

Microservices

Flexible Software Architecture

Addison-Wesley Professional The Most Complete, Practical, and Actionable Guide to Microservices Going beyond mere theory and marketing hype, Eberhard Wolff presents all the knowledge you need to capture the full benefits of this emerging paradigm. He illuminates microservice concepts, architectures, and scenarios from a technology-neutral standpoint, and demonstrates how to implement them with today's leading technologies such as Docker, Java, Spring Boot, the Netflix stack, and Spring Cloud. The author fully explains the benefits and tradeoffs associated with microservices, and guides you through the entire project lifecycle: development, testing, deployment, operations, and more. You'll find best practices for architecting microservice-based systems, individual microservices, and nanoservices, each illuminated with pragmatic examples. The author supplements opinions based on his experience with concise essays from other experts, enriching your understanding and illuminating areas where experts disagree. Readers are challenged to experiment on their own the concepts explained in the book to gain hands-on experience. Discover what microservices are, and how they differ from other forms of modularization Modernize legacy applications and efficiently build new systems Drive more value from continuous delivery with microservices Learn how microservices differ from SOA Optimize the microservices project lifecycle Plan, visualize, manage, and evolve architecture Integrate and communicate among microservices Apply advanced architectural techniques, including CQRS and Event Sourcing Maximize resilience and stability Operate and monitor microservices in production Build a full implementation with Docker, Java, Spring Boot, the Netflix stack, and Spring Cloud Explore nanoservices with Amazon Lambda, OSGi, Java EE, Vert.x, Erlang, and Seneca Understand microservices' impact on teams, technical leaders, product owners, and stakeholders Managers will discover better ways to support microservices, and learn how adopting the method affects the entire organization. Developers will master the technical skills and concepts they need to be effective. Architects will gain a deep understanding of key issues in creating or migrating toward microservices, and exactly what it will take to transform their plans into reality.

Cloud Native Programming with Golang

Develop microservice-based high performance web apps for the cloud with Go

Packt Publishing Ltd Discover practical techniques to build cloud-native apps that are scalable, reliable, and always available. Key Features Build well-designed and secure microservices. Enrich your microservices with continuous integration and monitoring. Containerize your application with Docker Deploy your application to AWS. Learn how to utilize the powerful AWS services from within your application Book Description Awarded as one of the best books of all time by BookAuthority, Cloud Native Programming with Golang will take you on a journey into the world of microservices and cloud computing with the help of Go. Cloud computing and microservices are two very important concepts in modern software architecture. They represent key skills that ambitious software engineers need to acquire in order to design and build software applications capable of performing and scaling. Go is a modern cross-platform programming language that is very powerful yet simple; it is an excellent choice for microservices and cloud applications. Go is gaining more and more popularity, and becoming a very attractive skill. This book starts by covering the software architectural patterns of cloud applications, as well as practical concepts regarding how to scale, distribute, and deploy those applications. You will also learn how to build a JavaScript-based front-end for your application, using TypeScript and React. From there, we dive into commercial cloud offerings by covering AWS. Finally, we conclude our book by providing some overviews of other concepts and technologies that you can explore, to move from where the book leaves off. What you will learn Understand modern software applications architectures Build secure microservices that can effectively communicate with other services Get to know about event-driven architectures by diving into message queues such as Kafka, Rabbitmq, and AWS SQS. Understand key modern database technologies such as MongoDB, and Amazon's DynamoDB Leverage the power of containers Explore Amazon cloud services fundamentals Know how to utilize the power of the Go language to access key services in the Amazon cloud such as S3, SQS, DynamoDB and more. Build front-end applications using ReactJS with Go Implement CD for modern applications Who this book is for This book is for developers who want to begin building secure, resilient, robust, and scalable Go applications that are cloud native. Some knowledge of the Go programming language should be sufficient. To build the front-end application, you will also need some knowledge of JavaScript programming.

Implementing Domain-driven Design

Pearson Education Vaughn Vernon presents concrete and realistic domain-driven design (DDD) techniques through examples from familiar domains, such as a Scrum-based project management application that integrates with a collaboration suite and security provider. Each principle is backed up by realistic Java examples, and all content is tied together by a single case study of a company charged with delivering a set of advanced software systems with DDD.