Api Design On The Scale Of Decades Learn How To Architect And Design Long Lasting Apis

Recognizing the showing off ways to get this book **api design on the scale of decades learn how to architect and design long lasting apis** is additionally useful. You have remained in right site to begin getting this info. acquire the api design on the scale of decades learn how to architect and design long lasting apis colleague that we find the money for here and check out the link.

You could purchase guide api design on the scale of decades learn how to architect and design long lasting apis or acquire it as soon as feasible. You could quickly download this api design on the scale of decades learn how to architect and design long lasting apis after getting deal. So, behind you require the ebook swiftly, you can straight get it. It's appropriately completely simple and so fats, isn't it? You have to favor to in this appearance

Scaling API Design **C++Now 2018: Titus Winters "Modern C++ API Design: From Rvalue-References to Type Design"** *Web API Design Maturity Model* Core Principles Of API Design REST Tutorial - How to Design a Good RESTful API How to Design Great APIs - Parse Developer Day 2013 *Modeling RESTful API Resources Amazon System Design Preparation (SIP)* API Design with Swaggerhub **9 best practices of REST API development** *Best Practices for API Design to Keep Your App Secure, Scalable and Efficient*

Best Practice For API Design in Elixir - Part 1 | Erlang Solutions WebinarOpenAPI 3.0 Tutorial| Swagger Tutorial For Beginners | Design REST API Using Swagger Editor 10 Best Practices For Developing An API System Design Interview Question: DESIGN A PARKING LOT - asked at Google, Facebook How the OpenAPI Specification Led to Better REST API Design | Joan James | API The Docs Virtual 2020 *OAuth 2.0: An Overview APIs | REST | REST APIs Demystified* Web APIs You [Probably] Didn't Know Existed **REST API \u0026 RESTful Web Services Explained | Web Services Tutorial** *What is an API? - Application Programming Interface* 5 Tips for System Design Interviews *A method for REST-API design Game Engine API Design GraphQL, gRPC and REST, Oh My! : A Method for Unified API Design* Better API Design with OpenAPI (Cloud Next '18) API Design for REST **Effective Design of RESTful APIs** *Building Modern APIs with GraphQL GraphQL Schema Design @ Scale (Marc-André Giroux)* **Api Design On The Scale**
The event detailed design techniques for platform longevity, as well as operational components of sustaining a working API business model. API Design on the Scale of Decades gathers some of the best information shared by speakers at the 2016 Platform Summit; so even if you weren't able to attend, you can now catch a part of the experience! For this book, we took the 15 most popular speaker sessions on API design technique, and used them as inspiration to craft in-depth chapters on a wide ...

API Design on the Scale of… by Nordic APIs [PDF/iPad/Kindle]
Expert insights dedicated to the idea of designing APIs with longevity in mind. We cover the most pressing current API design decisions for maintaining long-lasting API platforms, including REST, GraphQL, versioning strategies, common anti-patterns, OAuth 2.0 in IoT, microservices, and more...

Amazon.com: API Design on the Scale of Decades: Learn How ...
Amazon API Gateway is designed to make it easy for developers to create APIs at any scale without managing any servers. API Gateway will handle all of the heavy lifting needed including traffic management, security, monitoring, and version/environment management.

How to Architect APIs for Scale and Security | AWS ...
Rethinking API Design at Scale with the Stoplight Platform In 2016 Stoplight released the industry's first form-based OpenAPI designer, and since then we have helped developers, architects, and even product managers design tens of thousands of APIs.

API Design & Development at Scale | Scalable API Design ...
Preface APIsontheScaleofDecades Roy Fielding, the creator of the REST standard for API design,oncedescribedRESTas: "softwaredesignonthescaleofdecades:every

API Design on the Scale of Decades
API Design on the Scale of Decades Expert insights from the 2016 Nordic APIs Platform Summit, dedicated to the idea of designing APIs with longevity in mind. We cover the most pressing modern API design techniques for maintaining long-lasting API platforms, including REST, GraphQL, versioning strategies, REST hooks, OAuth 2.0 in IoT, microservices architecture, and more.

API Design on the Scale of Decades | Nordic APIs
Scale API Design Efforts Adopt an API Design-First approach across multiple teams with tooling that fits seamlessly into your existing workflows.

Enterprise API Design Management at Scale | Stoplight for ...

The processes teams follow to implement and maintain API design, will vary based on several factors, including: end consumers, number of APIs, as well as the strategic and business goals for their API program. For teams that are early on in their API design journey, setting up a scalable API design process won't be a major consideration.

Will Your API Design Process Scale?

In order for APIs to deliver on a myriad of benefits and objectives, organizations must design them with scale in mind. However, the need to build high-performing APIs that scale with the business ecosystem is pressuring many development teams to build APIs that may be restricting business growth.

Best Practices for Building a Secure and Scalable API ...

In a large-scale environment, many clients using different versions of a web API can result in a significant amount of duplicated data in a server-side cache. This issue can become acute if a client application communicates with a web server through a proxy that implements caching, and that only forwards a request to the web server if it does not currently hold a copy of the requested data in its cache.

API design guidance - Best practices for cloud ...

Roy Fielding, the creator of the REST standard for API design, once described REST as: " software design on the scale of decades: every detail is intended to promote software longevity and independent evolution. Many of the constraints are directly opposed to short-term efficiency .".

New eBook Released: API Design on the Scale of Decades ...

Efficiently scale large and multiple data pipelines to realize the impact of AI/ML projects. How It Works After sending us your data via API call, our platform through a combination of human work and review, smart tools, statistical confidence checks and machine learning checks returns scalable, accurate ground truth data.

Scale AI: The Data Platform for AI

Students will get the understanding on how to plan, design and develop RESTFul APIs. This will give them a framework to understand the approach of securing and scaling API. Learn how to use plan, design and deploy APIs on AWS Elastic Beanstalk, AWS Lambda, AWS API Gateway and AWS Dynamodb ...

REST API, Design, Test, Scale and Deploy | Udemy

Good API design takes account of the organization's specific strengths and limitations in terms of budget, personal skill sets and technical infrastructure. The larger lesson here is that API architects should always focus on real-world requirements rather than technical elegance.

API Design 101: API Design Basics - API Academy

API Strategy | Posted January 18, 2018. While more and more organizations are embracing design as a critical step in the software development lifecycle, and the lifecycle of their APIs, very few organizations have figured out how to effectively design APIs at scale. When you're working within a single team on a confined number of services, design is easier to manage and relying on PDF style guides or wiki pages may be enough to keep everyone aligned.

How to Scale Your API Design Process with OpenAPI

Scaling Your API Design With SwaggerHub During the webinar, the SwaggerHub Team introduced some of the advanced API design features available in SwaggerHub to help teams scale their API design...

Scaling Your API Design: How to Improve Usability and ...

Published on 2017-02-21.Last updated on 2020-07-22.. Introduction. This is a general design guide for networked APIs. It has been used inside Google since 2014 and is the guide that Google follows when designing Cloud APIs and other Google APIs.This design guide is shared here to inform outside developers and to make it easier for us all to work together.

API Design Guide | Cloud APIs | Google Cloud

Since 1924, the American Petroleum Institute has been a cornerstone in establishing and maintaining standards for the worldwide oil and natural gas industry. Our work helps the industry invent and manufacture superior products consistently, provide critical services, ensure fairness in the marketplace for businesses and consumers alike, and promotes the acceptance of products and practices ...

Principles of Web API Design is a comprehensive, start-to-finish guide to the processes required for effective API design. Unlike other books, it covers the entire lifecycle. Leading API and microservices consultant James Higginbotham shows how API developers can successfully integrate processes that occur before, during, and after API design, to scale API development far beyond single individuals or small teams. Higginbotham addresses REST in depth while also fully covering RPC and graph-based API design, as well as messaging, streaming, and event-based async APIs. Coverage includes: The art of API design, and an overview of the API design process Crafting job stories, conducting EventStorming sessions, and modeling API capabilities Designing APIs that can easily evolve Implementing APIs, and moving to microservices Improving API quality through effective testing, documentation, and protection mechanisms Establishing and maturing your API program: leveraging program and data management techniques that scale

API Design for C++ provides a comprehensive discussion of Application Programming Interface (API) development, from initial design through implementation, testing, documentation, release, versioning, maintenance, and deprecation. It is the only book that teaches the strategies of C++ API development, including interface design, versioning, scripting, and plug-in extensibility. Drawing from the author's experience on large scale, collaborative software projects, the text offers practical techniques of API design that produce robust code for the long term. It presents patterns and practices that provide real value to individual developers as well as organizations. API Design for C++ explores often overlooked issues, both technical and non-technical, contributing to successful design decisions that product high quality, robust, and long-lived APIs. It focuses on various API styles and patterns that will allow you to produce elegant and durable libraries. A discussion on testing strategies concentrates on automated API testing techniques rather than attempting to include end-user application testing techniques such as GUI testing, system testing, or manual testing. Each concept is illustrated with extensive C++ code examples, and fully functional examples and working source code for experimentation are available online. This book will be helpful to new programmers who understand the fundamentals of C++ and who want to advance their design skills, as well as to senior engineers and software architects seeking to gain new expertise to complement their existing talents. Three specific groups of readers are targeted: practicing software engineers and architects, technical managers, and students and educators. The only book that teaches the strategies of C++ API development, including design, versioning, documentation, testing, scripting, and extensibility. Extensive code examples illustrate each concept, with fully functional examples and working source code for experimentation available online. Covers various API styles and patterns with a focus on practical and efficient designs for large-scale long-term projects.

Using a web API to provide services to application developers is one of the more satisfying endeavors that software engineers undertake. But building a popular API with a thriving developer ecosystem is also one of the most challenging. With this practical guide, developers, architects, and tech leads will learn how to navigate complex decisions for designing, scaling, marketing, and evolving interoperable APIs. Authors Brenda Jin, Saurabh Sahni, and Amir Shevat explain API design theory and provide hands-on exercises for building your web API and managing its operation in production. You'll also learn how to build and maintain a following of app developers. This book includes expert advice, worksheets, checklists, and case studies from companies including Slack, Stripe, Facebook, Microsoft, Cloudinary, Oracle, and GitHub. Get an overview of request-response and event-driven API design paradigms Learn best practices for designing an API that meets the needs of your users Use a template to create an API design process Scale your web API to support a growing number of API calls and use cases Regularly adapt the API to reflect changes to your product or business Provide developer resources that include API documentation, samples, and tools

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.

Web APIs are everywhere, giving developers an efficient way to interact with applications, services, and data. Well-designed APIs are a joy to use; poorly-designed APIs are cumbersome, confusing, and frustrating. The Design of Web APIs is a practical, example packed guide to crafting extraordinary web APIs. Author Arnaud Lauret demonstrates fantastic design principles and techniques you can apply to both public and private web APIs. Purchase of the print book includes a free eBook in PDF, Kindle, and ePub formats from Manning Publications.

While GraphQL is a technology mainly driven by the needs of clients, there is a clear lack of resources on how to build reliable GraphQL servers. Over the last few years, I helped build and maintain some of the biggest GraphQL APIs out there at both Shopify and GitHub. During those years, I also worked with various companies with their adoption of GraphQL. From my experiences with GraphQL, I've observed the good, the bad, and the ugly. This led to many talks and blog posts on the subject, but still found that teams and individuals willing to use GraphQL in a pragmatic way lacked the resources to do so. This is what lead me to write this book: A collection of learnings and good practices when building GraphQL schemas at scale. Every language and every GraphQL implementation does things slightly differently. This book is completely language agnostic and instead focuses on concepts and patterns that are

achievable no matter how you're building a GraphQL server. Think of it as a complete journey of what goes into building a GraphQL API, from design, to architectures, to implementation, and even documentation.

Looking for Best Practices for RESTful APIs? This book is for you! Why? Because this book is packed with practical experience on what works best for RESTful API Design. You want to design APIs like a Pro? Use API description languages to both design APIs and develop APIs efficiently. The book introduces the two most common API description languages RAML, OpenAPI, and Swagger. Your company cares about its customers? Learn API product management with a customer-centric design and development approach for APIs. Learn how to manage APIs as a product and how to follow an API-first approach. Build APIs your customers love! You want to manage the complete API lifecycle? An API development methodology is proposed to guide you through the lifecycle: API inception, API design, API development, API publication, API evolution, and maintenance. You want to build APIs right? This book shows best practices for REST design, such as the correct use of resources, URIs, representations, content types, data formats, parameters, HTTP status codes, and HTTP methods. Your APIs connect to legacy systems? The book shows best practices for connecting APIs to existing backend systems. Your APIs connect to a mesh of microservices? The book shows the principles for designing APIs for scalable, autonomous microservices. You expect lots of traffic on your API? The book shows you how to achieve high performance, availability and maintainability. You want to build APIs that last for decades? We study API versioning, API evolution, backward- and forward-compatibility and show API design patterns for versioning. The API-University Series is a modular series of books on API-related topics. Each book focuses on a particular API topic, so you can select the topics within APIs, which are relevant for you.

Got RESTful APIs? Great. API consumers love them. But today, such RESTful APIs are not enough for the evolving expectations of API consumers. Their apps need to be responsive, event-based and react to changes in near real-time. This results in a new set of requirements for the APIs, which power the apps. APIs now need to provide concepts such as events, notifications, triggers, and subscriptions. These concepts are not natively supported by the REST architectural style. In thios book we show how to engineer RESTful APIs that support events with a webhook infrastructure. What are the alternatives to webhooks? We study several approaches for realizing events, such as Polling, Long Polling, Webhooks, HTTP Streaming, Server-Sent Events, WebSockets, WebSub and GraphQL Subscriptions. All of these approaches have their advantages and disadvantages. Can webhooks communicate in real-time? We study the non-functional requirements of a webhooks infrastructure, in areas such as security, reliability and developer experience. How do well-known API providers design webhooks? We examine the webhook infrastructure provided by GitHub, BitBucket, Stripe, Slack, and Intercom. With the best practices, case studies, and design templates provided in this book, we want to help you extend your API portfolio with a modern webhook infrastructure. So you can offer both APIs and events that developers love to use.

A lot of work is required to release an API, but the effort doesn't always pay off. Overplanning before an API matures is a wasted investment, while underplanning can lead to disaster. This practical guide provides maturity models for individual APIs and multi-API landscapes to help you invest the right human and company resources for the right maturity level at the right time. How do you balance the desire for agility and speed with the need for robust and scalable operations? Four experts from the API Academy show software architects, program directors, and product owners how to maximize the value of their APIs by managing them as products through a continuous life cycle. Learn which API decisions you need to govern and how and where to do so Design, deploy, and manage APIs using an API-as-a-product (AaaP) approach Examine ten pillars that form the foundation of API product work Learn how the continuous improvement model governs changes throughout an API's lifetime Explore the five stages of a complete API product life cycle Delve into team roles needed to design, build, and maintain your APIs Learn how to manage your API landscape—the set of APIs published by your organization

API Design Patterns lays out a set of design principles for building internal and public-facing APIs. Summary A collection of best practices and design standards for web and internal APIs. In API Design Patterns you will learn: Guiding principles for API patterns Fundamentals of resource layout and naming Handling data types for any programming language Standard methods that ensure predictability Field masks for targeted partial updates Authentication and validation methods for secure APIs Collective operations for moving, managing, and deleting data Advanced patterns for special interactions and data transformations API Design Patterns reveals best practices for building stable, user-friendly APIs. These design patterns can be applied to solve common API problems and flexibly altered to fit your specific needs. Hands-on examples and relevant use-cases illustrate patterns for API fundamentals, advanced functionalities, and even uncommon scenarios. Purchase of the print book includes a free eBook in PDF, Kindle, and ePub formats from Manning Publications. About the technology APIs are contracts that define how applications, services, and components communicate. API design patterns provide a shared set of best practices, specifications and standards that ensure APIs are reliable and simple for other developers to use. This book collects and explains the most important patterns from both the API design community and the experts at Google. About the book API Design Patterns lays out a set of design principles for building internal and public-facing APIs. Google API expert JJ Geewax presents patterns that ensure your APIs are consistent, scalable, and flexible. You'll improve the design of the most common APIs, plus discover techniques for tricky edge cases. Precise illustrations, relevant examples, and detailed scenarios make every pattern clear and easy to understand. What's inside Guiding principles for

API patterns Fundamentals of resource layout and naming Advanced patterns for special interactions and data transformations A detailed case-study on building an API and adding features About the reader For developers building web and internal APIs in any language. About the author JJ Geewax is a software engineer at Google, focusing on Google Cloud Platform, API design, and real-time payment systems. He is also the author of Manning's Google Cloud Platform in Action. Table of Contents PART 1 INTRODUCTION 1 Introduction to APIs 2 Introduction to API design patterns PART 2 DESIGN PRINCIPLES 3 Naming 4 Resource scope and hierarchy 5 Data types and defaults PART 3 FUNDAMENTALS 6 Resource identification 7 Standard methods 8 Partial updates and retrievals 9 Custom methods 10 Long-running operations 11 Rerunnable jobs PART 4 RESOURCE RELATIONSHIPS 12 Singleton sub-resources 13 Cross references 14 Association resources 15 Add and remove custom methods 16 Polymorphism PART 5 COLLECTIVE OPERATIONS 17 Copy and move 18 Batch operations 19 Criteria-based deletion 20 Anonymous writes 21 Pagination 22 Filtering 23 Importing and exporting PART 6 SAFETY AND SECURITY 24 Versioning and compatibility 25 Soft deletion 26 Request deduplication 27 Request validation 28 Resource revisions 29 Request retrial 30 Request authentication