Practical Api Design Confessions Of A Java Framework Architect Jaroslav Tulach

Windows Presentation Foundation UnleashedPractical API DesignPractical API DesignTypes and Programming LanguagesThe Rails 3 WayAsp.net Core 1.1 for BeginnersThe Definitive Guide to DjangoSoftware CraftsmanshipWomen On SuccessConcurrent Programming in AdaAPI Design for C++REST API Design RulebookEthical Hacking and Countermeasures: Web Applications and Data ServersDesigning Ecommerce WebsitesUser Interface Design for ProgrammersRESTful API DesignCMake CookbookRich Client ProgrammingSkylight ConfessionsPractical Cloud SecurityUsable SecurityAPI Design for C++Programming for BetfairConfessions of a Prayer SlackerMore Programming PearlsConfessions of a Public SpeakerPro Java 6 3D Game DevelopmentPractical API DesignPractical API DesignDomain-driven DesignConfessions of an Advertising ManCreating Maintainable APIsAgile Project Management with ScrumThe Power of YESiOS 9 App Development EssentialsMaking Things HappenFive FrequenciesVolatilityMindful DesignDesign Patterns in Modern C++

Windows Presentation Foundation Unleashed

Does your ecommerce site attract visitors but fail to convert enough of them into paying customers? Are your sales numbers not where you want? Have you got an ecommerce site but not yet spent any time on improving the design? If it's time to upgrade your website's user experience then Designing Ecommerce Websites is the only book to tell you exactly how to do it. It provides you with 66 guidelines on how to best use every element on each page of an ecommerce website. This book contains the results of a decade's worth of UX design experience, and several years spent consulting with a wide range of different ecommerce startups. It is based on knowledge learned from user behaviour data and running many different usability tests. It tells you what works in reality, not in theory. The book itself was born from teaching the principles in workshops for over four years. These are principles that are useful to a range of job roles (not just designers) and no matter what your experience level. This book will take you step by step through the ecommerce funnel that applies to almost all ecommerce sites via scannable text and simple illustrations. It's a reference book that is designed to be easy to pick up and quickly learn from. Design LANDING pages that stop users from being confused and bouncing; Create LISTINGS that help your users find a product they actually want to buy; Design PRODUCT pages that don't leave the users with any fears about buying from you; Develop a CHECKOUT flow that results in more successful payments; And learn MORE advice about the other key pages on an ecommerce site. This is the second edition of the book and it features completely rewritten and updated advice for 2019, 15 totally new guidelines, and links to further reading for every guideline (so you can learn even more). The first edition was an Amazon Kindle bestseller in the ecommerce and technology business

categories.

Practical API Design

This book looks at the two most popular ways of using Java SE 6 to write 3D games on PCs: Java 3D (a high-level scene graph API) and JOGL (a Java layer over OpenGL). Written by Java gaming expert, Andrew Davison, this book uses the new Java (SE) 6 platform and its features including splash screens, scripting, and the desktop tray interface. This book is also unique in that it covers Java game development using the Java 3D API and Java for OpenGL--both critical components and libraries for Java-based 3D game application development

Practical API Design

Your organization's culture: It's either an asset or a liability. At this very moment, it's either lifting your business performance or "eating your strategy for breakfast." How do you ensure it's the former, not the latter? Based on the authors' 20+ years of research, the answer couldn't be clearer: Intentionally or unintentionally, you create culture through the signals you transmit on Five Frequencies: 1/ Your decisions and actions. 2/ What you reward and recognize. 3/ What you tolerate (or don't). 4/ How you show up informally. 5/ Your formal communications. To make a bad culture good, or a good culture great, you've got to deliberately transmit strong and steady signals. This guide shows you how. Fast-paced and loaded with real-life examples, Five Frequencies prescribes more than 20 practical Signal Boosts for turning culture into competitive advantage. Check out early endorsements and download the introduction and a discussion guide at www.fivefrequencies.com.

Types and Programming Languages

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.

The Rails 3 Way

The concept of concurrency presents a real difficulty to many programmers. In this book Dr Burns introduces the topic and shows how the Ada programming language, which had as a main design specification that it should contain the features necessary for concurrent programming, deals with it. The need to produce software that models inherent parallelism is important in many applications, including embedded real time systems, process control and general purpose multi-processor system software. Although for many other applications of Ada, knowledge of the Tasking Model is not necessary the full power of the language can only really be obtained if this model is understood.

Asp.net Core 1.1 for Beginners

Most programmers' fear of user interface (UI) programming comes from their fear of doing UI design. They think that UI design is like graphic design—the mysterious process by which creative, latte-drinking, all-black-wearing people produce cool-looking, artistic pieces. Most programmers see themselves as analytic, logical thinkers instead—strong at reasoning, weak on artistic judgment, and incapable of doing UI design. In this brilliantly readable book, author Joel Spolsky proposes simple, logical rules that can be applied without any artistic talent to improve any user interface, from traditional GUI applications to websites to consumer electronics. Spolsky's primary axiom, the importance of bringing the program model in line with the user model, is both rational and simple. In a fun and entertaining way, Spolky makes user interface design easy for programmers to grasp. After reading User Interface Design for Programmers, you'll know how to design interfaces with the user in mind. You'll learn the important principles that underlie all good UI design, and you'll learn how to perform usability testing that works.

The Definitive Guide to Django

There has been roughly 15 years of research into approaches for aligning research in Human Computer Interaction with computer Security, more colloquially known as ``usable security.'' Although usability and security were once thought to be inherently antagonistic, today there is wide consensus that systems that are not usable will inevitably suffer security failures when they are deployed into the real world. Only by simultaneously addressing both usability and security concerns will we be able to build systems that are truly secure. This book presents the historical context of the work to date on usable security and privacy, creates a taxonomy for organizing that work, outlines current research objectives, presents lessons learned, and makes suggestions for future research.

Software Craftsmanship

The EC-Council | Press Ethical Hacking and Countermeasures Series is comprised of five books covering a broad base of topics in offensive network security, ethical hacking, and network defense and countermeasures. The content of this series is designed to immerse the reader into an interactive environment where they will be shown how to scan, test, hack and secure information systems. With the full series of books, the reader will gain in-depth knowledge and practical experience with essential security systems, and become prepared to succeed on the Certified Ethical Hacker, or C|EH, certification from EC-Council. This certification covers a plethora of offensive security topics ranging from how perimeter defenses work, to scanning and attacking simulated networks. A wide variety of tools, viruses, and malware is presented in this and the other four books, providing a complete understanding of the tactics and tools used by hackers. By gaining a thorough understanding of how hackers operate, an Ethical Hacker will be able to set up strong countermeasures and defensive systems to protect an organization's critical infrastructure and information. Important Notice: Media content referenced within the product description or the product text may not be available in the ebook version.

Women On Success

Learn CMake through a series of task-based recipes that provide you with practical, simple, and ready-to-use CMake solutions for your code Key Features Learn to configure, build, test, and package software written in C, C++, and Fortran Progress from simple to advanced tasks with examples tested on Linux, macOS, and Windows Manage code complexity and library dependencies with reusable CMake building blocks Book Description CMake is cross-platform, open-source software for managing the build process in a portable fashion. This book features a collection of recipes and building blocks with tips and techniques for working with CMake, CTest, CPack, and CDash. CMake Cookbook includes real-world examples in the form of recipes that cover different ways to structure, configure, build, and test small- to large-scale code projects. You will learn to use CMake's command-line tools and master modern CMake practices for configuring, building, and testing binaries and libraries. With this book, you will be able to work with external libraries and structure your own projects in a modular

and reusable way. You will be well-equipped to generate native build scripts for Linux, MacOS, and Windows, simplify and refactor projects using CMake, and port projects to CMake. What you will learn Configure, build, test, and install code projects using CMake Detect operating systems, processors, libraries, files, and programs for conditional compilation Increase the portability of your code Refactor a large codebase into modules with the help of CMake Build multi-language projects Know where and how to tweak CMake configuration files written by somebody else Package projects for distribution Port projects to CMake Who this book is for If you are a software developer keen to manage build systems using CMake or would like to understand and modify CMake code written by others, this book is for you. A basic knowledge of C++, C, or Fortran is required to understand the topics covered in this book.

Concurrent Programming in Ada

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.

API Design for C++

Discover the power of "yes" and all the amazing things it can do for you. So often we are afraid of failure, of

disappointment, of being vulnerable, that we settle for "no". The practical tips and inspirational advice within these pages will help you embrace positivity and find a new sense of freedom in each area of your life, from your career, to your relationships, to your dreams and ambitions.

REST API Design Rulebook

By recognizing that software development is not a mechanical task, you can create better applications. Today's software development projects are often based on the traditional software engineering model, which was created to develop large-scale defense projects. Projects that use this antiquated industrial model tend to take longer, promise more, and deliver less. As the demand for software has exploded, the software engineering establishment has attempted to adapt to the changing times with short training programs that teach the syntax of coding languages. But writing code is no longer the hard part of development; the hard part is figuring out what to write. This kind of know-how demands a skilled craftsman, not someone who knows only how to pass a certification course. Software Craftsmanship presents an alternative--a craft model that focuses on the people involved in commercial software development. This book illustrates that it is imperative to turn from the technology-for-its-own-sake model to one that is grounded in delivering value to customers. The author, Pete McBreen, presents a method to nurture mastery in the programmer, develop creative collaboration in small developer teams, and enhance communications with the customer. The end result--skilled developers who can create, extend, and enhance robust applications. This book addresses the following topics, among others: Understanding customer requirements Identifying when a project may go off track Selecting software craftsmen for a particular project Designing goals for application development Managing software craftsmen Software Craftsmanship is written for programmers who want to become exceptional at their craft and for the project manager who wants to hire them. 0201733862B07242001

Ethical Hacking and Countermeasures: Web Applications and Data Servers

Want to learn how to build ASP.NET Core 1.1 MVC Web Applications? Prerequisites: * C# (Intermediate level) * HTML5/CSS3 (Basic knowledge) This book is primarily aimed at developers who want to learn how to build ASP.NET Core 1.1 MVC Applications. You should be an intermediate level C# developer with some experience in HTML5 and CSS3. The book presupposes that you have a solid C# foundation since the language won't be explained in any detail. You will learn ASP.NET Core 1.1 by building two MVC applications. The first application will be built using an empty template. The goal is to get you familiar with ASP.NET Core 1.1 by adding middleware and services one piece at a time, building a basic application. Then you will build a second MVC application using a template that already contains support for MVC, Entity Framework Core, and user authentication. This application is a video course website, where users can register to gain access to video courses. If you are already familiar with MVC 5, the content in this book can get you started with ASP.NET

Core 1.1 in a fast, no-fluff way. It's important to mention that this book is practical and tactical, where you will learn as you progress through the modules and build real web applications in the process. To spare you countless pages of fluff (filler material), only valuable information, pertinent to the task at hand, is discussed. The benefit is a shorter and more condensed book, which will save you time and give you a more enjoyable experience. The goal is to learn ASP.NET Core 1.1 by building two web applications, one from scratch and one from an existing template. This experience is something you can put in your CV when applying for a job or a consultant position, or when negotiating a higher salary. Technologies, frameworks and languages you will use: * ASP.NET Core 1.1 MVC (The framework that you will use) * Services (To provide you own functionality as a reusable service) * Middleware (To provide you own functionality to HTTP Request pipeline) * Entity Framework (To crate and communicate with a database) * View Component (To render data in the _Layout view with model data) * Razor syntax (To include server-side code in views) * Bootstrap (Used for styling and to create a responsive design) * LINQ (To query the database) * Dependency Injection (To inject objects into constructors) * Tag Helper (to clean up the HTML and enable re-use) * HTML Helper methods (to clean up your HTML and enable re-use) * Bower/NuGet (To install necessary front-end/back-end libraries) What you will implement: * Implement a web application from an empty template. * Implement a web application from an existing template. * Create a "real world" code-first database using Entity Framework Core. * Add and modify models, views and controllers to perform CRUD operations against the database. * Use client-side and server-side validation. * Secure the controllers, actions and view content with authorization and roles. * Styling HTML with CSS and Bootstrap. * Create a responsive website. * Register users with the site. * Manage users and their video courses. I hope you love the book! Sincerely, Jonas Fagerberg Now click the Buy button and start reading the book.

Designing Ecommerce Websites

A comprehensive introduction to type systems and programming languages. A type system is a syntactic method for automatically checking the absence of certain erroneous behaviors by classifying program phrases according to the kinds of values they compute. The study of type systems—and of programming languages from a type-theoretic perspective—has important applications in software engineering, language design, high-performance compilers, and security. This text provides a comprehensive introduction both to type systems in computer science and to the basic theory of programming languages. The approach is pragmatic and operational; each new concept is motivated by programming examples and the more theoretical sections are driven by the needs of implementations. Each chapter is accompanied by numerous exercises and solutions, as well as a running implementation, available via the Web. Dependencies between chapters are explicitly identified, allowing readers to choose a variety of paths through the material. The core topics include the untyped lambda-calculus, simple type systems, type reconstruction, universal and existential polymorphism, subtyping, bounded quantification, recursive types, kinds, and type operators. Extended case studies develop a variety of approaches to

modeling the features of object-oriented languages.

User Interface Design for Programmers

Writing at the height of her powers, Alice Hoffman conjures three generations of a family haunted by love. Cool, practical, and deliberate, John is dreamy Arlyn's polar opposite. Yet the two are drawn powerfully together even when it is clear they are bound to bring each other grief. Their difficult marriage leads them and their children to a house made of glass in the Connecticutcountryside, to the avenues ofManhattan, and to the blue waters of Long Island Sound. Glass breaks, love hurts, and families make their own rules. Ultimately, it falls to their grandson, Will, to solve the emotional puzzle of his family and of his own identity.

RESTful API Design

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.

CMake Cookbook

You might think more than enough design books exist in the programming world already. In fact, there are so many that it

makes sense to ask why you would read yet another. Is there really a need for yet another design book? In fact, there is a greater need than ever before, and Practical API Design: Confessions of a Java Framework Architect fills that need! Teaches you how to write an API that will stand the test of time Written by the designer of the NetBeans API at Sun Technologies Based on best practices, scalability, and API design patterns

Rich Client Programming

Build straightforward and maintainable APIs to create services that are usable and maintainable. Although this book focuses on distributed services, it also emphasizes how the core principles apply even to pure OOD and OOP constructs. The overall context of Creating Maintainable APIs is to classify the topics into four main areas: classes and interfaces, HTTP REST APIs, messaging APIs, and message payloads (XML, JSON and JSON API as well as Apache Avro). What You Will Learn Use object-oriented design constructs and their APIs Create and manage HTTP REST APIs Build and manage maintainable messaging APIs, including the use of Apache Kafka as a principal messaging hub Handle message payloads via JSON Who This Book Is For Any level software engineers and very experienced programmers.

Skylight Confessions

The Rails™ 3 Way is a comprehensive resource that digs into the new features in Rails 3 and perhaps more importantly, the rationale behind them. —Yehuda Katz, Rails Core The Bible for Ruby on Rails Application Development Ruby on Rails strips complexity from the development process, enabling professional developers to focus on what matters most: delivering business value via clean and maintainable code. The Rails™ 3 Way is the only comprehensive, authoritative guide to delivering production-quality code with Rails 3. Pioneering Rails expert Obie Fernandez and a team of leading experts illuminate the entire Rails 3 API, along with the idioms, design approaches, and libraries that make developing applications with Rails so powerful. Drawing on their unsurpassed experience and track record, they address the real challenges development teams face, showing how to use Rails 3 to maximize your productivity. Using numerous detailed code examples, the author systematically covers Rails 3 key capabilities and subsystems, making this book a reference that you will turn to again and again. He presents advanced Rails programming techniques that have been proven effective in day-to-day usage on dozens of production Rails systems and offers important insights into behavior-driven development and production considerations such as scalability. Dive deep into the Rails 3 codebase and discover why Rails is designed the way it is—and how to make it do what you want it to do. This book will help you Learn what's new in Rails 3 Increase your productivity as a web application developer Realize the overall joy in programming with Rails Leverage Rails' powerful capabilities for building REST-compliant APIs Drive implementation and protect long-term maintainability using RSpec Design and manipulate your domain layer using Active Record Understand and program complex program flows using

Action Controller Master sophisticated URL routing concepts Use Ajax techniques via Rails 3 support for unobtrusive JavaScript Learn to extend Rails with popular gems and plugins, and how to write your own Extend Rails with the best third-party plug-ins and write your own Integrate email services into your applications with Action Mailer Improve application responsiveness with background processing Create your own non-Active Record domain classes using Active Model Master Rails' utility classes and extensions in Active Support

Practical Cloud Security

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.

Usable Security

You might think more than enough design books exist in the programming world already. In fact, there are so many that it makes sense to ask why you would read yet another. Is there really a need for yet another design book? In fact, there is a greater need than ever before, and Practical API Design: Confessions of a Java Framework Architect fills that need! Teaches you how to write an API that will stand the test of time Written by the designer of the NetBeans API at Sun Technologies Based on best practices, scalability, and API design patterns

API Design for C++

Describes ways to incorporate domain modeling into software development.

Programming for Betfair

This is a book by Amazing Women to Inspire More Women around the World. The world is full of stories of successful women, and we need to capture them, or they will evaporate over time. These stories must be documented. And they must be shared to inspire the next generation of females. Stories and experiences move and motivate. They provide us with a boost to face our challenges head-on. The 37 contributing authors of this book come from 18 countries. We have 13 contributing authors from the Kingdom of Bahrain; the rest of the authors come from Canada, Egypt, Estonia, Finland, India, Indonesia, Kenya, Lebanon, Malaysia, New Zealand, Oman, Russia, Saudi Arabia, United Arab Emirates, United Kingdom, United States, and Venezuela. In essence, the book covers the six continents of the world. Through out the book, there is

one common theme: When there is a will, there is a way. Let the authors of this book guide you to the way of success!

Confessions of a Prayer Slacker

Gain a deep, intuitive and technical understanding of practical options theory The main challenges in successful options trading are conceptual, not mathematical. Volatility: Practical Options Theory provides financial professionals, academics, students and others with an intuitive as well as technical understanding of both the basic and advanced ideas in options theory to a level that facilitates practical options trading. The approach taken in this book will prove particularly valuable to options traders and other practitioners tasked with making pricing and risk management decisions in an environment where time constraints mean that simplicity and intuition are of greater value than mathematical formalism. The most important areas of options theory, namely implied volatility, delta hedging, time value and the so-called options greeks are explored based on intuitive economic arguments alone before turning to formal models such as the seminal Black-Scholes-Merton model. The reader will understand how the model free approach and mathematical models are related to each other, their underlying theoretical assumptions and their implications to level that facilitates practical implementation. There are several excellent mathematical descriptions of options theory, but few focus on a translational approach to convert the theory into practice. This book emphasizes the translational aspect, while first building an intuitive, technical understanding that allows market makers, portfolio managers, investment managers, risk managers, and other traders to work more effectively within—and beyond—the bounds of everyday practice. Gain a deeper understanding of the assumptions underlying options theory Translate theoretical ideas into practice Develop a more accurate intuition for better time-constrained decision making This book allows its readers to gain more than a superficial understanding of the mechanisms at work in options markets. Volatility gives its readers the edge by providing a true bedrock foundation upon which practical knowledge becomes stronger.

More Programming Pearls

Now available in paperback— You might think more than enough design books exist in the programming world already. In fact, there are so many that it makes sense to ask why you would read yet another. Is there really a need for yet another design book? In fact, there is a greater need than ever before, and Practical API Design: Confessions of a Java Framework Architect fills that need! Teaches you how to write an API that will stand the test of time Written by the designer of the NetBeans API at Sun Technologies Based on best practices, scalability, and API design patterns What you'll learn What an API is and what the theories are behind good API design When and why to build an API API design patterns applicable to all programming languages, especially modern, object–oriented languages How to optimize and test APIs Who this book is for This book is recommended to every API architect who prefers a bit more engineering design over a purely artistic one. Table

of Contents The Art of Building Modern Software The Motivation to Create an API Determining What Makes a Good API Ever-Changing Targets Do Not Expose More Than You Want Code Against Interfaces, Not Implementations Use Modular Architecture Separate APIs for Clients and Providers Keep Testability in Mind Cooperating with Other APIs Runtime Aspects of APIs Declarative Programming Extreme Advice Considered Harmful Paradoxes of API Design Evolving the API Universe Teamwork Using Games to Improve API Design Skills Extensible Visitor Pattern Case Study End-of-Life Procedures

Confessions of a Public Speaker

Django, the Python-based Web development framework, is one of the hottest topics in Web development today. Its creator (and co-author of this book) Adrian Holovaty has built a compelling array of Web applications using Django, including http://chicagocrime.org. Django creator Adrian Holovaty and lead developer Jacob Kaplan-Moss have created this book as the definitive guide to the technology. Beginning with fundamentals such as installation and configuration, the book tackles sophisticated features of Django, such as outputting non-HTML content such as RSS feeds and PDFs, caching, and user management. Also includes a detailed reference to Django's many configuration options and commands.

Pro Java 6 3D Game Development

Learn to create seamless designs backed by a responsible understanding of the human mind. This book examines how human behavior can be used to integrate your product design into lifestyle, rather than interrupt it, and make decisions for the good of those that are using your product. Mindful Design introduces the areas of brain science that matter to designers, and passionately explains how those areas affect each human's day-to-day experiences with products and interfaces. You will learn about the neurological aspects and limitations of human vision and perception; about our attachment to harmony and dissonance, such as visual harmony, musical harmony; and about our brain's propensity towards pattern recognition and how we perceive the world cognitively. In the second half of the book you will focus on the practical application of what you have learned, specific to interaction and interface design. Real-world examples are used throughout so that you can really see how design is impacting our everyday digital experience. Design is a responsibility, but not enough designers understand the human mind or the process of thought. This book explores the key factors involved and shows you how to make the right design choices. What You'll Learn Review how attention and distraction work and the cost of attentional switching Use Gestalt principles to communicate visual grouping Ensure your underlying models make sense to your audience Use time, progression, and transition to create a composition Carefully examine controlling behavior through reductionist and behaviorist motivation concepts Apply the theoretical knowledge to practical, mindful application design Who This Book Is For The primary audience for this book is professional designers who wish to learn more about the human mind and how to apply that to their work. The book is also useful for design-focussed product owners and

startup founders who wish to apply ethical thinking to a team, or when bootstrapping their products. The secondary audience is design students who are either studying a 'traditional' visual design course, or a UX/interaction design course who have a desire to learn how they might be able to apply mindful design to their early careers. Finally, a tertiary audience for this book would be tutors involved in teaching design, or peripheral, courses who may wish to incorporate its teachings into their lectures, workshops or seminars.

Practical API Design

Offers a collection of essays on philosophies and strategies for defining, leading, and managing projects. This book explains to technical and non-technical readers alike what it takes to get through a large software or web development project. It does not cite specific methods, but focuses on philosophy and strategy.

Practical API Design

The rules and practices for Scrum—a simple process for managing complex projects—are few, straightforward, and easy to learn. But Scrum's simplicity itself—its lack of prescription—can be disarming, and new practitioners often find themselves reverting to old project management habits and tools and yielding lesser results. In this illuminating series of case studies, Scrum co-creator and evangelist Ken Schwaber identifies the real-world lessons—the successes and failures—culled from his years of experience coaching companies in agile project management. Through them, you'll understand how to use Scrum to solve complex problems and drive better results—delivering more valuable software faster. Gain the foundation in Scrum theory—and practice—you need to: Rein in even the most complex, unwieldy projects Effectively manage unknown or changing product requirements Simplify the chain of command with self-managing development teams Receive clearer specifications—and feedback—from customers Greatly reduce project planning time and required tools Build—and release—products in 30-day cycles so clients get deliverables earlier Avoid missteps by regularly inspecting, reporting on, and fine-tuning projects Support multiple teams working on a large-scale project from many geographic locations Maximize return on investment!

Domain-driven Design

The open-source NetBeans Platform is an extraordinarily powerful framework for building "write once, run anywhere" rich client applications. Now, for the first time since the release of NetBeans IDE 5.0, there's a comprehensive guide to rich client development on the NetBeans Platform. Written for Java developers and architects who have discovered that basic Swing components are not enough for them, this book will help you get started with NetBeans module development, master

NetBeans' key APIs, and learn proven techniques for building reliable desktop software. Each chapter is filled with practical, step-by-step instructions for creating complete rich client applications on top of the NetBeans Platform and plugins for NetBeans IDE. Rich Client Programming 's wide-ranging content covers Why modular development makes sense for small, medium, and large applications Using NetBeans to accelerate development and improve efficiency Leveraging NetBeans productivity features, from the Component Palette to Code Completion Leveraging NetBeans' modular architecture in your own applications Implementing loosely coupled communication to improve code maintainability and robustness Managing user- and system-configuration data Building reloadable components with solid threading models Constructing sophisticated multiwindow applications and presenting rich data structures to users Adding user-configurable options Integrating Web services with NetBeans desktop applications Automating module updates and providing user help Foreword by Jonathan Schwartz Foreword by Jan Chalupa Preface About the Authors and Contributors Acknowledgments Chapter 1: Getting Started with the NetBeans Platform Chapter 2: The Benefits of Modular Programming Chapter 3: Modular Architecture Chapter 4: Loosely Coupled Communication Chapter 5: Lookup Chapter 6: Filesystems Chapter 7: Threading, Listener Patterns, and MIME Lookup Chapter 8: The Window System Chapter 9: Nodes, Explorer Views, Actions, and Presenters Chapter 10: DataObjects and DataLoaders Chapter 11: Graphical User Interfaces Chapter 12: Multiview Editors Chapter 13: Syntax Highlighting Chapter 14: Code Completion Chapter 15: Component Palettes Chapter 16: Hyperlinks Chapter 17: Annotations Chapter 18: Options Windows Chapter 19: Web Frameworks Chapter 20: Web Services Chapter 21: JavaHelp Documentation Chapter 22 Update Centers Chapter 23: Use Case 1: NetBeans Module Development Chapter 24: Use Case 2: Rich Unger on Application Development Chapter A: Advanced Module System Techniques Chapter B: Common Idioms and Code Patterns in NetBeans Chapter C: Performance Index

Confessions of an Advertising Man

iOS 9 App Development Essentials is latest edition of this popular book series and has now been fully updated for the iOS 9 SDK, Xcode 7 and the Swift 2 programming language. Beginning with the basics, this book provides an outline of the steps necessary to set up an iOS development environment. An introduction to the architecture of iOS 9 and programming in Swift is provided, followed by an in-depth look at the design of iOS applications and user interfaces. More advanced topics such as file handling, database management, in-app purchases, graphics drawing and animation are also covered, as are touch screen handling, gesture recognition, multitasking, iAds integration, location management, local notifications, camera access and video and audio playback support. Other features are also covered including Auto Layout, Twitter and Facebook integration, App Store hosted in-app purchase content, Sprite Kit-based game development, local map search and user interface animation using UIKit dynamics. Additional features of iOS development using Xcode 7 are also covered, including Swift playgrounds, universal user interface design using size classes, app extensions, Interface Builder Live Views, embedded frameworks, CloudKit data storage and TouchID authentication. The key new features of iOS 9 and Xcode 7 are

also covered in detail, including new error handling in Swift 2, designing Stack View based user interfaces, multiple storyboard support, iPad multitasking, map flyover support, 3D Touch and Picture-in-Picture media playback. The aim of this book, therefore, is to teach you the skills necessary to build your own apps for iOS 9. Assuming you are ready to download the iOS 9 SDK and Xcode 7, have an Intel-based Mac and ideas for some apps to develop, you are ready to get started.

Creating Maintainable APIs

Printed entirely in color, with helpful figures and syntax coloring to make code samples appear as they do in Visual Studio. Windows Presentation Foundation (WPF) is a key component of the .NET Framework 3.0, giving you the power to create richer and more compelling applications than you dreamed possible. Whether you want to develop traditional user interfaces or integrate 3D graphics, audio/video, animation, dynamic skinning, rich document support, speech recognition, or more, WPF enables you to do so in a seamless, resolution-independent manner. Windows Presentation Foundation Unleashed is the authoritative book that covers it all, in a practical and approachable fashion, authored by .NET guru and Microsoft developer Adam Nathan. · Covers everything you need to know about Extensible Application Markup Language (XAML) · Examines the WPF feature areas in incredible depth: controls, layout, resources, data binding, styling, graphics, animation, and more · Features a chapter on 3D graphics by Daniel Lehenbauer, lead developer responsible for WPF 3D · Delves into non-mainstream topics: speech, audio/video, documents, bitmap effects, and more · Shows how to create popular UI elements, such as features introduced in the 2007 Microsoft Office System: Galleries, ScreenTips, custom control layouts, and more · Demonstrates how to create sophisticated UI mechanisms, such as Visual Studio-like collapsible/dockable panes · Explains how to develop and deploy all types of applications, including navigation-based applications, applications hosted in a Web browser, and applications with great-looking non-rectangular windows · Explains how to create first-class custom controls for WPF · Demonstrates how to create hybrid WPF software that leverages Windows Forms, ActiveX, or other non-WPF technologies · Explains how to exploit new Windows Vista features in WPF applications

Agile Project Management with Scrum

You might think more than enough design books exist in the programming world already. In fact, there are so many that it makes sense to ask why you would read yet another. Is there really a need for yet another design book? In fact, there is a greater need than ever before, and Practical API Design: Confessions of a Java Framework Architect fills that need! Teaches you how to write an API that will stand the test of time Written by the designer of the NetBeans API at Sun Technologies Based on best practices, scalability, and API design patterns

The Power of YES

Apply modern C++17 to the implementations of classic design patterns. As well as covering traditional design patterns, this book fleshes out new patterns and approaches that will be useful to C++ developers. The author presents concepts as a fun investigation of how problems can be solved in different ways, along the way using varying degrees of technical sophistication and explaining different sorts of trade-offs. Design Patterns in Modern C++ also provides a technology demo for modern C++, showcasing how some of its latest features (e.g., coroutines) make difficult problems a lot easier to solve. The examples in this book are all suitable for putting into production, with only a few simplifications made in order to aid readability. What You Will Learn Apply design patterns to modern C++ programming Use creational patterns of builder, factories, prototype and singleton Implement structural patterns such as adapter, bridge, decorator, facade and more Work with the behavioral patterns such as chain of responsibility, command, iterator, mediator and more Apply functional design patterns such as Monad and more Who This Book Is For Those with at least some prior programming experience, especially in C++.

iOS 9 App Development Essentials

With their rapidly changing architecture and API-driven automation, cloud platforms come with unique security challenges and opportunities. This hands-on book guides you through security best practices for multivendor cloud environments, whether your company plans to move legacy on-premises projects to the cloud or build a new infrastructure from the ground up. Developers, IT architects, and security professionals will learn cloud-specific techniques for securing popular cloud platforms such as Amazon Web Services, Microsoft Azure, and IBM Cloud. Chris Dotson—an IBM senior technical staff member—shows you how to establish data asset management, identity and access management, vulnerability management, network security, and incident response in your cloud environment.

Making Things Happen

Confessions of an Advertising Man is the distillation of all the successful Ogilvy concepts, tactics and techniques that made this book an international bestseller. Regarded as the father of modern advertising, David Ogilvy created some of the most memorable advertising campaigns that set the standard for others to follow. Anyone aspiring to be a good manager in any kind of business should read this.

Five Frequencies

In this hilarious and highly practical book, author and professional speaker Scott Berkun reveals the techniques behind what great communicators do, and shows how anyone can learn to use them well. For managers and teachers -- and anyone else who talks and expects someone to listen -- Confessions of a Public Speaker provides an insider's perspective on how to effectively present ideas to anyone. It's a unique, entertaining, and instructional romp through the embarrassments and triumphs Scott has experienced over 15 years of speaking to crowds of all sizes. With lively lessons and surprising confessions, you'll get new insights into the art of persuasion -- as well as teaching, learning, and performance -- directly from a master of the trade. Highlights include: Berkun's hard-won and simple philosophy, culled from years of lectures, teaching courses, and hours of appearances on NPR, MSNBC, and CNBC Practical advice, including how to work a tough room, the science of not boring people, how to survive the attack of the butterflies, and what to do when things go wrong The inside scoop on who earns $30,000 for a one-hour lecture and why The worst -- and funniest -- disaster stories you've ever heard (plus countermoves you can use) Filled with humorous and illuminating stories of thrilling performances and real-life disasters, Confessions of a Public Speaker is inspirational, devastatingly honest, and a blast to read.

Volatility

A gifted writer aims to help readers uncover many misconceptions they have picked up, which have damaged their prayer life over the years, in an enlightening and inspiring guide to an entire new prayer journey. Original.

Mindful Design

Software -- Software Engineering.

Design Patterns in Modern C++

The Betfair exchange, coupled with its API, permits a suitably skilled trader to code complex trading applications, which would not look out of place in the financial markets. This book offers a sports trader the chance to build their own trading applications, regardless of their programming ability. Each chapter of Programming for Betfair contains snippets of code that combine to create a complete trading application. The application is geared towards horse racing but can easily be adapted to other sports on Betfair's exchange. Using Microsoft's Visual Studio (downloadable for free) the reader is shown how to code an application that will gather prices for any market on Betfair's exchange and then place bets into that market. The reader is shown how to automate their trading so that they can remove emotion from their trades and scale up their trading for increased profits. Further development of the application permits it to save data from Betfair onto the reader's hard drive for offline analysis and visualisation in a spreadsheet for the purpose of building trading algorithms. Also

covered is an enhancement of Betfair's charts so that charts can be automatically updated and compared. The final chapter of the book discusses ideas for taking the application and the reader's skills to the next level. Topics discussed include constructing your own trading indicators, volume analysis, trend following, arbitrage, low-latency trading and many more.

ROMANCE ACTION & ADVENTURE MYSTERY & THRILLER BIOGRAPHIES & HISTORY CHILDREN'S YOUNG ADULT FANTASY HISTORICAL FICTION HORROR LITERARY FICTION NON-FICTION SCIENCE FICTION