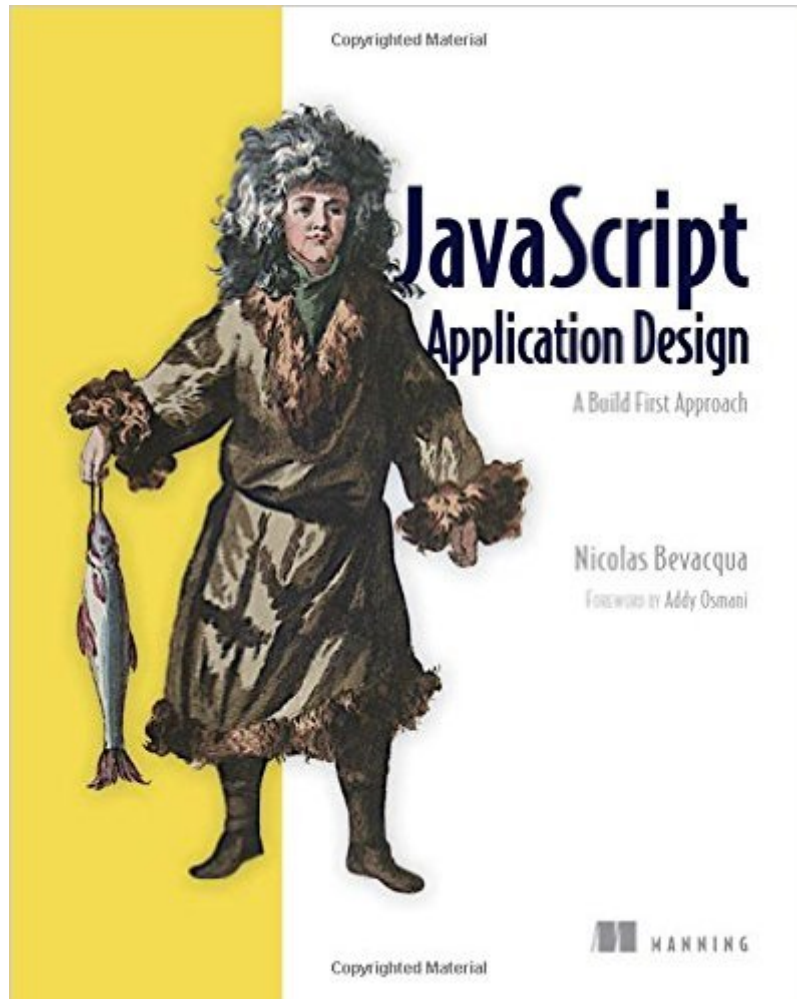


The book was found

JavaScript Application Design: A Build First Approach



Synopsis

Summary JavaScript Application Design: A Build First Approach introduces JavaScript developers to techniques that will improve the quality of their software as well as their web development workflow. You'll begin by learning how to establish build processes that are appropriate for JavaScript-driven development. Then, you'll walk through best practices for productive day-to-day development, like running tasks when your code changes, deploying applications with a single command, and monitoring the state of your application once it's in production. Purchase of the print book includes a free eBook in PDF, Kindle, and ePub formats from Manning Publications.

About the Book The fate of most applications is often sealed before a single line of code has been written. How is that possible? Simply, bad design assures bad results. Good design and effective processes are the foundation on which maintainable applications are built, scaled, and improved. For JavaScript developers, this means discovering the tooling, modern libraries, and architectural patterns that enable those improvements.

JavaScript Application Design: A Build First Approach introduces techniques to improve software quality and development workflow. You'll begin by learning how to establish processes designed to optimize the quality of your work. You'll execute tasks whenever your code changes, run tests on every commit, and deploy in an automated fashion. Then you'll focus on designing modular components and composing them together to build robust applications. This book assumes readers understand the basics of JavaScript.

What's Inside Automated development, testing, and deployment processes JavaScript fundamentals and modularity best practices Modular, maintainable, and well-tested applications Master asynchronous flows, embrace MVC, and design a REST API

About the Author Nicolas Bevacqua is a freelance developer with a focus on modular JavaScript, build processes, and sharp design. He maintains a blog at ponyfoo.com.

Table of Contents

PART 1 BUILD PROCESSES Introduction to Build First Composing build tasks and flows Mastering environments and the development workflow Release, deployment, and monitoring

PART 2 MANAGING COMPLEXITY Embracing modularity and dependency management Understanding asynchronous flow control methods in JavaScript Leveraging the Model-View-Controller Testing JavaScript components REST API design and layered service architectures

Book Information

Paperback: 344 pages

Publisher: Manning Publications; 1 edition (February 16, 2015)

Language: English

ISBN-10: 1617291951

ISBN-13: 978-1617291951

Product Dimensions: 7.1 x 0.7 x 9.3 inches

Shipping Weight: 1.4 pounds (View shipping rates and policies)

Average Customer Review: 4.3 out of 5 stars [See all reviews](#) (3 customer reviews)

Best Sellers Rank: #252,070 in Books (See Top 100 in Books) #37 in [Books > Computers & Technology > Internet & Social Media > Web Browsers](#) #62 in [Books > Computers & Technology > Programming > Software Design, Testing & Engineering > Tools](#) #73 in [Books > Computers & Technology > Web Development & Design > Web Services](#)

Customer Reviews

JavaScript Application Design: A Build First Approach The fate of most applications is often sealed before a single line of code has been written. How is that possible? Simply, bad design assures bad results. Good design and effective processes are the foundation on which maintainable applications are built, scaled, and improved. For JavaScript developers, this means discovering the tooling, modern libraries, and architectural patterns that enable those improvements.

JavaScript Application Design introduces techniques to improve software quality and development workflow. You'll begin by learning how to establish processes designed to optimize the quality of your work. You'll execute tasks whenever your code changes, run tests on every commit, and deploy in an automated fashion. Then you'll focus on designing modular components and composing them together to build robust applications. **What's Inside** Automated development, testing, and deployment processes JavaScript fundamentals and modularity best practices Modular, maintainable, and well-tested applications Master asynchronous flows, embrace MVC, and design a REST API

Although I program since 1990, I am neither an expert nor a fan of JavaScript, but I recognize that it's a great tool for web development. and not only web. I believe the title is a catches attention, and the subtitle was really intriguing to me: "A Build First Approach". The introductions to each chapter are short enough to not get you bored, but are very effective. The reading is always smoothly and interesting, and the author showed me JavaScript under a different light. I found myself reading it anytime I could. If you're looking for a book to learn JavaScript for your web pages, this book is not for you. But if you want to make your web pages working smoothly and safely with JavaScript you should get this book. You will learn how to implement javascript in a modular way according to MVC,

test your JavaScript in different ways. And not just syntactically, as I often found in many project. You will learn (or improve) how to use Node.js and Grunt, managing release, deployment and monitoring in a fancy and easy way. The book also gives a good guide to implement your APIs in a proficient way. Although those guidelines are not new, I found many project not following them, and it's a good refresh! After reading this book, I am implementing many of the things I found in it in the projects I am managing.

I didn't know how much I needed this book until I started reading it and exploring its code examples. Many of us who have worked with JavaScript started our connections to the language in very haphazard fashions. We likely learned it on the job, under deadline pressure to fix or update somebody else's code. Or we have taken introductory classes and then started picking up whatever else we can on the fly, including the bad habits of others around us who have seemed to know more about JavaScript than we do at the moment. JavaScript, unfortunately, is a big, messy programming language, and it offers numerous opportunities to crash and burn if you really don't know what you are doing. In his new book, JavaScript Application Design, Nicolas Bevacqua makes a compelling case for using "the Build First philosophy of designing for clean, well-structured, and testable applications before you write a single line of code." He writes: "You'll learn about process automation, which will mitigate the odds of human error.... Build First is the foundation that will empower you to design clean, well-structured, and testable applications, which are easy to maintain and refactor. Those are the two fundamental aspects of Build First: process automation and design." In his well-written text, he argues: "Front-end development requires as much dedication to architecture planning and design as back-end development does. Long gone are the days when we'd copy a few snippets of code off the internet, paste them in our page, and call it a day. Mashing together JavaScript code as an afterthought no longer holds up to modern standards. JavaScript is now front and center." He continues: "We have many frameworks and libraries to choose from, which can help you organize your code by allowing you to write small components rather than a monolithic application. Maintainability isn't something you can tack onto a code base whenever you'd like; it's something you have to build into the application, and the philosophy under which the application is designed, from the beginning. Writing an application that isn't designed to be maintainable translates into stacking feature after feature in an ever-so-slightly tilting Jenga tower." Bevacqua divides his nine-chapter book into just two parts: build processes and managing complexity. Here is how the chapters are organized: PART 1: BUILD PROCESSES 1 - Introduction to Build First 2 - Composing build tasks and flows 3 -

Mastering environments and the development workflow
4 - Release, deployment, and monitoring
PART 2: MANAGING COMPLEXITY
5 - Embracing modularity and dependency management
6 - Understanding asynchronous flow control methods in JavaScript
7 - Leveraging the Model-View-Controller
8 - Testing JavaScript components
9 - REST API design and layered service architectures

Bevaqua notes that "Linting is often referred to as the first test you should set up when writing JavaScript. Where linters fail, unit tests come in." He strongly pushes testing and automation right from the start, of course. Linting soon leads to Grunt, which Bevaqua uses as a task runner and key build tool (with selected modules) in this book. "Grunt is a tool that allows you to write, configure, and automate tasks--such as minifying a JavaScript file or compiling a LESS style sheet--for your application," he states. (It also works well on Windows machines, which I have found handy.) Grunt leads to running a bit of Node.js on the command line. And if you've never worked with Node.js, Bevacqua takes the reader smoothly through the process of installing it and using it in linting exercises. Indeed, he devotes an entire appendix (B) to installing and running Grunt and picking the right plugins for the right tasks and targets. One of the best parts of this book, to me, is how the author uses short code examples to introduce a concept, and then builds upon the examples with helpful descriptions and more short but expanded code samples. Nicolas Bevacqua offers his readers plenty of helpful how-to and why information. Using his book, I have begun applying the Build First approach to some new projects and learning to how test and automate more of my work. I feel like I have a good shot now at getting a lot better than I have been, at JavaScript.

One small but important glitch to note: At two points in my preview copy of the book, Bevacqua shows what he calls a simple way to create bare-minimum JSON manifest files. For example, `echo "{}" > package.json`. Creating a blank, starting-point manifest file did not work this way for me, however. Instead, I had to use `echo {"name": "project-name"} > package.json`. The empty package.json issue apparently is somehow related to certain versions of Node's npm. (My thanks to Manning for providing a review copy.)

This book covers the process of designing a well engineered JavaScript web app. It explains in detail crucial aspects such as automation, modularity or patterns with focus on practical web development and design. It assumes you understand the basics of JavaScript. The author shares his experience along a quite dense text. Plenty of technical details, figures and snippets of code are available in the book. The author follows an incremental approach. It starts with building stuff (tools, processes, workflow, environment, etc) and then it jumps to design (modularity, patterns, features, testing, architecture, etc) He gets a smooth transition and I suppose beginners will be happy to

catch all this stuff in one only book. My favourite part of the book was the second part, 'managing complexity'. I enjoyed the 'REST API design and layered service architectures' chapter where the book covers API design pitfalls, layered service or client side consumers among other topics. In summary, I found this book interesting. Although it doesn't contain novel stuff I think it is a great introductory book, it highlights the right way to get the things done with JavaScript and it introduces all technologies you could need with your first JavaScript application design.

[Download to continue reading...](#)

JavaScript Application Design: A Build First Approach Learning JavaScript: JavaScript Essentials for Modern Application Development High Performance JavaScript (Build Faster Web Application Interfaces) Effective JavaScript: 68 Specific Ways to Harness the Power of JavaScript (Effective Software Development Series) Build an HTML5 Game: A Developer's Guide with CSS and JavaScript ASP.NET Core Application Development: Building an application in four sprints (Developer Reference) Adobe ColdFusion 9 Web Application Construction Kit, Volume 2: Application Development Adobe ColdFusion 8 Web Application Construction Kit, Volume 3: Advanced Application Development Wiley GAAP: Interpretation and Application of Generally Accepted Accounting Principles 2011 (Wiley GAAP: Interpretation & Application of Generally Accepted Accounting Principles) Girls Life Application Study Bible NLT (Kid's Life Application Bible) The Handbook of Bible Application (Life Application Reference) A Smarter Way to Learn JavaScript: The new approach that uses technology to cut your effort in half My Very First Library: My Very First Book of Colors, My Very First Book of Shapes, My Very First Book of Numbers, My Very First Books of Words Head First JavaScript Programming Individual Income Taxation: An Application Approach Sunset Outdoor Design Guide: Backyard Cottages & Gazebos: Fresh Ideas for Outdoor Living (Sunset Outdoor Design & Build Guides) First Things First: Understand Why So Often Our First Things Aren't First Eargle's The Microphone Book: From Mono to Stereo to Surround - A Guide to Microphone Design and Application (Audio Engineering Society Presents) Measurement Systems: Application and Design Design and Implement Servlets, JSPs, and EJBs for IBM WebSphere Application Server (IBM Redbook)

[Dmca](#)