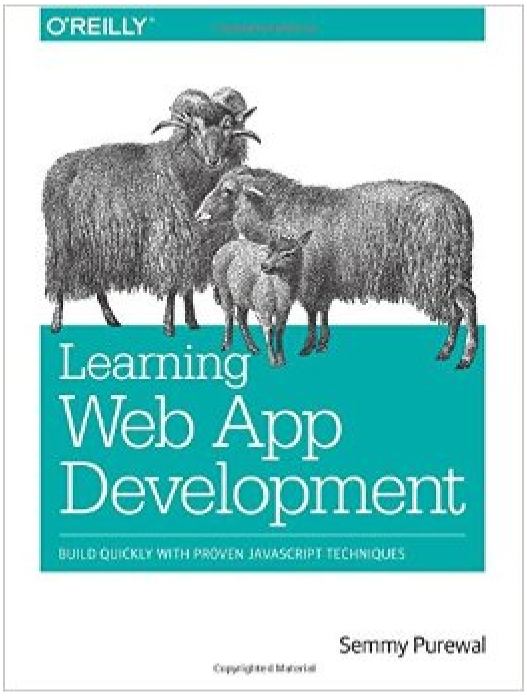
Learning Web App Development



Synopsis

Grasp the fundamentals of web application development by building a simple database-backed app from scratch, using HTML, JavaScript, and other open source tools. Through hands-on tutorials, this practical guide shows inexperienced web app developers how to create a user interface, write a server, build client-server communication, and use a cloud-based service to deploy the application.Each chapter includes practice problems, full examples, and mental models of the development workflow. Ideal for a college-level course, this book helps you get started with web app development by providing you with a solid grounding in the process.Set up a basic workflow with a text editor, version control system, and web browserStructure a user interface with HTML, and include styles with CSSUse JQuery and JavaScript to add interactivity to your applicationLink the client to the server with AJAX, JavaScript objects, and JSONLearn the basics of server-side programming with Node.jsStore data outside your application with Redis and MongoDBShare your application by uploading it to the cloud with CloudFoundryGet basic tips for writing maintainable code on both client and server

Book Information

Paperback: 306 pages
Publisher: O’Reilly Media; 1 edition (March 3, 2014)
Language: English
ISBN-10: 1449370195
ISBN-13: 978-1449370190
Product Dimensions: 7 x 0.7 x 9.2 inches
Shipping Weight: 1.4 pounds (View shipping rates and policies)
Average Customer Review: 4.3 out of 5 starsÂ Â See all reviewsÂ (37 customer reviews)
Best Sellers Rank: #241,584 in Books (See Top 100 in Books)   #125 inÂ Books > Computers & Technology > Mobile Phones, Tablets & E-Readers > Programming & App Development   #235 inÂ Books > Computers & Technology > Programming > Web Programming > JavaScript   #480 inÂ Books > Computers & Technology > Web Development & Design > Web Design

Customer Reviews

I’ll start off by saying that I’ve been developing code for 20+ years and am as comfortable rewriting machine code in memory to do arcane threaded kernel debugging as I am writing in high-level functional programming languages. However, Web app development is something I’ve never gained any facility with.The tutorials and books I’ve looked at (obviously not an exhaustive search) are typically so imprecise that it is maddening to an experienced developer w/o a driving need to learn the stuff. The Web app development domain is also filled with lots of jargon for decades-old concepts and technologies, further hindering people from other development domains. Finally, many of these works are more about promoting some particular stack or toolchain (written in good faith!) than actual expository works on the subject itself.Semmy has found an alternative path in this book. First, he has purposefully selected a solid and appropriate set of technologies. Second, he has defined a "micro-app" that, while very simple, possesses all of the key facets common to any Web application.With regard to the first item, I take it as axiomatic that new Web app development will involve HTML5, CSS, and Javascript. However, if you stick with just those life is very hard. Right off the bat you’re likely to end up making a real mess of client-side DOM manipulation. jQuery to the rescue! Is jQuery the only alternative? No, but it’s a fine place to start. The reader will also want a server-side stack. Semmy makes a good choice (IMO) in giving us an easy virtual machine-based path here. That keeps us out of the swamp of trying to set up something ourselves when we just want to continue with the example and gives us just a tiny taste of deploying things in real public or private clouds. It also sticks with a low-enough level setup that what is actually happening when the different parts of the application are interacting is still clear.I didn’t expect a book called "Learning Web App Development" to teach me design - application design is it’s rich own domain in which Web-specific issues are a small part. I also didn’t expect the book to be a highly complete exposition on the subject, something I don’t think is possible without either becoming so tied to a particular toolset that the book would be marginalized, or so enormous it becomes a "write only" tome.If you’re an experienced software developer but have never written anything for a browser other than some tiny Javascript apps, or last wrote stuff in the CGI days with HTML tables, this book will get you up to a point where you can both meaningfully communicate with Web developers and select/evaluate tools for whatever problem you are trying to solve. All in 10 to 40 hours depending on how many of the exercises you do - sweet.If you’re not an experienced developer at all, you’ll have been taken through a meaningful Web micro-app in a pedagogically defensible manner. Count yourself lucky.

[This is an excerpt of my review published in ACM ComputingReviews.com]Becoming proficient at web application development involves a very steep learning curve and is often a never-ending, career-long endeavor. And yet a newbie has to start somewhere. When getting started, itÃ¢Â Â™s easy to become overwhelmed by the vast array of concepts, technologies, and tools one has to master.Often, even proficient web application developers have blinders on when it comes to having a broad perspective on web application development. In general, most developers are adequately familiar with only one or two of the following aspects of web application development: client-side programming, middleware programming, server-side programming, text editors/integrated development environments (IDEs), source/version control systems, data stores, operating systems, network programming, and hardware configurations. Few developers have a broad enough perspective--or what I call a 360-degree Ã¢Â Âœpixel to metal (P2M)Ã¢Â Â• worldview--of web application development.ThatÃ¢Â Â™s why Semmy PurewalÃ¢Â Â™s new book is a highly welcome and valuable contribution. Along similar lines, at the New England Java Users Group (NEJUG, Boston), in August 2013, I presented a technology stack and sample application with source code for end-to-end web application development. After my presentation, I had wished to write a book similar to this one. Alas, I had neither the time nor the energy to see it through. However, the stack I presented was a polyglot stack, including JavaScript for client-side development, Java for server-side development, and a relational database. PurewalÃ¢Â Â™s book somewhat simplifies the learning curve, to the extent thatÃ¢Â Â™s possible, by choosing an entirely JavaScript-based stack.At less than 300 pages, PurewalÃ¢Â Â™s book may seem short, but it manages to introduce the reader to a surprisingly wide array of concepts, technologies, and tools. And in doing so, the author does a great job of spending just the right amount of time on each topic, gradually building on previously explained topics to construct increasingly sophisticated web application snippets in each successive chapter.HereÃ¢Â Â™s a listing of the concepts, technologies, and tools this book introduces: Sublime, Git, GitHub, Chrome, Linux, Hypertext Markup Language (HTML), cascading style sheets (CSS), JavaScript, jQuery, JSON, Ajax, VirtualBox, Vagrant, Node.js, Express, PuTTY, hypertext transfer protocol (HTTP), Redis, MongoDB, Mongoose, and Cloud Foundry. As a result, it is not only ideal for novices, but also helpful for experienced developers looking to plug gaps in their P2M perspective of web application development.This is a very thoughtfully put together book. There are very few typographical errors or inconsistencies, which is difficult to achieve in a book on programming. Exercises at the end of each chapter stimulate the reader to take on additional challenges, and some of the exercise results are leveraged in future chapters (for example, finding the number of occurrences of a string in an array of strings). There are useful pointers to further reading at the end of each chapter. Most of the bookÃ¢Â Â™s code is available on GitHub, organized by chapter.The author uses two major themes for most of the exercises and code examples. The first is a to-do list that the user can build, tag, and categorize by tags. The second is a poker hand evaluator that is used to demonstrate how best to leverage some of JavaScriptÃ¢Â Â™s built-in functions. Other smaller examples involve consuming JSON feeds from Yahoo, Flickr, and Twitter. I got some of the harder exercises working on my machine, and found the code examples to be accurate and helpful. The author follows a certain discipline for each project, including checking code into GitHub at regular intervals, which is good for developers to emulate.Along the way, the author manages to impart useful information on programming best practices, especially with respect to JavaScript, which is central to the programming environment that this book is sculpted around.A few words of caution. If youÃ¢Â Â™re using the Internet Explorer browser (for example, because your employer only allows IE), you wonÃ¢Â Â™t get the full experience. IEÃ¢Â Â™s JavaScript console is much inferior to that of Chrome. On IE 9, for example, I was not able to drill down into JavaScript objects in the way that Chrome permits. Also, many of the longstanding Node.js modules, for example, Express, have been broken up into their component modules, which have to be installed and linked separately. So, the code in the book isnÃ¢Â Â™t going to work as is, but it will point you in the right direction.Unfortunately, past chapter 6 (on Node.js) things start to get a bit light on details. On page 229, we connect to the Amazerrific data store in Mongo, but we never created the data store! Also, on GitHub, the code only goes up to chapter 6. So, at this crucial juncture, the reader is left to his or her own devices. We have a pretty decent to-dos app, but it has no ability to persist the to-dos. So, the next time you launch the app, all your to-dos are gone! (So, I took it upon myself to address the gaps--Google my GitHub to see a working demo with source code.)Whereas the author champions RESTful application programming interfaces (APIs), working through a project as I did will surely cause you to realize the shortcomings of an approach that attempts to use HTTP methods to imply which CRUD operation is taking place. A fundamental gap is that the four HTTP methods donÃ¢Â Â™t have uniform behavior. For example, an HTTP POST (meant to be used for updates) embeds parameters in the request body whereas an HTTP GET (meant to be used for gets or selects) places parameters on the uniform resource locator (URL). Apart from security and aesthetic considerations with placing parameters openly on the URL, it is unpleasant to not be able to work with uniform programming paradigms for these four HTTP methods either on the client side (jQuery doesnÃ¢Â Â™t have a $.put() method) or on the server side (Express).One area the author seems to omit completely is security. I donÃ¢Â Â™t expect a book to be able to cover all topics, but the author does a great job of alluding to important concepts even when he isnÃ¢Â Â™t planning to cover them in depth (for example, JavaScript Promise APIs). I believe readers ought to be given as complete a list of to-dos (pun intended!) required to make even a simple application production-ready. Furthermore, authentication and authorization are definitely must-haves.

Download to continue reading...

App Development: App Design and Development for Beginners Learning Web App Development

Troubleshooting PC Hardware: An Interactive Computer Diagnostic App (Help Desk in an eBook App 1) Innovation in Open and Distance Learning: Successful Development of Online and Web-based Learning (Open and Flexible Learning Series) Pro Web 2.0 Application Development with GWT (Expert's Voice in Web Development) Learning Mobile App Development: A Hands-on Guide to Building Apps with iOS and Android Python on Symbian: Mobile app development made easy iOS 9 App Development Essentials: Learn to Develop iOS 9 Apps Using Xcode 7 and Swift 2 Book 1: Diving In - iOS App Development for Non-Programmers Series: The Series on How to Create iPhone & iPad Apps Book 2: Flying With Objective-C - iOS App Development for Non-Programmers: The Series on How to Create iPhone & iPad Apps Android App Development For Dummies Microsoft SharePoint 2013 App Development (Developer Reference) Corba on the Web (Hands-on Web Development) Learning Web Design: A Beginner's Guide to (X)HTML, StyleSheets, and Web Graphics Motor Learning and Performance With Web Study Guide - 4th Edition: A Situation-Based Learning Approach Implementing Cisco IP Routing (ROUTE) Foundation Learning Guide: Foundation learning for the ROUTE 642-902 Exam (Foundation Learning Guides) Implementing Cisco IP Switched Networks (SWITCH) Foundation Learning Guide: Foundation learning for SWITCH 642-813 (Foundation Learning Guides) Deep Learning: Recurrent Neural Networks in Python: LSTM, GRU, and more RNN machine learning architectures in Python and Theano (Machine Learning in Python) Unsupervised Deep Learning in Python: Master Data Science and Machine Learning with Modern Neural Networks written in Python and Theano (Machine Learning in Python) Deep Learning in Python Prerequisites: Master Data Science and Machine Learning with Linear Regression and Logistic Regression in Python (Machine Learning in Python)