GOING MOBILE
Developing Apps for Your Library Using Basic HTML Programming

SCOTT LA COUNTE
GOING MOBILE

DEVELOPING APPS FOR YOUR LIBRARY USING BASIC HTML PROGRAMMING

SCOTT LA COUNTE

AMERICAN LIBRARY ASSOCIATION
Chicago  2012

www.alastore.ala.org
Scott La Counte is a librarian at Anaheim (Calif.) Public Library. He is the author of the book *Quiet, Please: Dispatches from a Public Librarian* (Da Capo Press, 2008), which began as a series for McSweeney’s Internet Tendencies. He teaches writing online for the Gotham Writers’ Workshops. His iPhone app, LibFind, gives the addresses and phone numbers of public libraries across the United States.
CONTENTS

Introduction  v

1  Before You Begin  1
2  Developing a Mobile Web Application  7
3  Using CSS  13
4  JavaScript and Mobile Website Design  19
5  PhoneGap  25
6  Building Your First Native App  29
7  Beyond the Basics  41
8  Other Ways to Go Mobile  45

Index  51
There are more than one hundred thousand mobile apps for the iPhone, and Android is quickly catching up to that number. Universities have apps, sports teams have apps, authors have apps—everyone seems to be in the app market, and for obvious reasons. Unlike the previous generation, which didn’t mind waiting to get their information when they got home, today’s generation wants their information on the go. In many respects they could be called Generation Mobile, because for the first time, there is a group of people who can have everything they could ever want in the palm of their hand.

What’s alarming, however, is of the hundreds of thousands of mobile apps available for phones, the amount of apps that librarians have built for libraries can be counted on one hand! The District of Columbia Public Library has an app to help you find books in its catalog. By the time this book is published, a few other libraries will likely have joined in, but the number isn’t anywhere near where it should be.

What does this mean? That there’s an entire generation that’s being overlooked.

Businesses have successfully found a new generation of clients simply by promoting services to them in ways that speak to a new type of user: the mobile user. They have proved that successful apps work and should not be overlooked.

For as long as computers and the Internet have been around, libraries have been on the cutting edge of implementing them; perhaps for the first time, libraries are lagging, and it’s time to catch up.

Mobile app development has become a pretty lucrative business; it’s a fair statement to say that it is the twenty-first century’s equivalent of a gold rush—upstart companies are making millions. In 2010, a developer for the Android phone reported that he was making $10,000 a month on an app that helped people find where they had left their car. He’s not alone.

It’s not in libraries’ best interest to go into app development with the intention of making money, but still, there is a huge market for mobile development.

Mobile websites allow patrons to avail themselves of library services no matter where they are. The sites can be set up to allow patrons to reserve books, to surf the card catalog, and more. Mobile websites, however, are a much different interface from the sites that you’re used to seeing on the Internet.

There are a lot of reasons libraries have, by and large, stayed away from phone apps. One of the biggest reasons, however, is that the skills needed to successfully build them
have historically been greater than what most libraries can afford; an app programmer can cost several hundred dollars an hour. Building a phone app using the phone's native programming language can be hard work and can take hundreds of hours.

Fortunately, new open-source software has made it easy for people with more limited programming skills to take advantage of apps. Using the technologies available, it's possible for any library to create an app at a low cost and to offer patrons a way to access the wealth of knowledge and entertainment stored in every library facility.

The point of this book is to show you what an app is, how to build one, and how to successfully market your app to library patrons. In the forthcoming chapters, you will learn not just how to build an app with relatively no programming skills but also the best practices for marketing the book to your target audience, as well as both free and paid services that are available to make going mobile even simpler and seamless.

There are many challenges ahead for any library attempting to enter this new form of development—and the challenges are among the biggest roadblocks for many libraries interested in extending their library services into the mobile market. I address some of the problems and challenges in this book, but the main point is to offer cost-effective solutions to library staff who are interested in experimenting with building an app for the library.
MOBILE APP: A DEFINITION

It’s difficult to write about mobile devices because that phrase can mean so many things. When I say “mobile app,” does that refer to an app that runs on any phone? Or on a specific phone? Just to be clear, there are two types of cell phones:

**Feature phone:** A feature phone is what the average person carries in his or her pocket. Although many feature phones are becoming more sophisticated, for the most part, feature phones are the more popular basic phones that you can buy. They make calls, they offer texting, and most can take pictures and have very limited web browsing. More than 80 percent of Americans have a feature phone.

**Smartphone:** Smartphones are best described as computers in the palm of your hand. Several times more powerful than feature phones, smartphones not only can surf the Internet but also can surf at relatively fast, third-generation (3G) speeds. The iPhone, Android, BlackBerry, and Windows Mobile are all examples of this kind of phone.

Many of the ideas in this book are universal; they can be tried on pretty much any phone in existence. The majority of ideas, however, are maximized for use with smartphones.

What is the point of developing apps that fewer than 20 percent of the population can access? Smartphones are the future of cellular technology. Smartphone prices have dropped significantly in recent years. When the iPhone was released in 2007, it cost $599; less than five years later, it cost as little as $99. The same is true of BlackBerries and nearly all other smartphones. As costs continue to drop, the smartphone is becoming more and more affordable to the average consumer.

Mobile development has become one of the biggest growth sectors of web development. People still use computers to surf the Internet, but the amount of time they spend on a computer will decrease as other devices become more prevalent.

The advent of XHTML-MP (XHTML for the Mobile Phone) and WCSS (Wireless CSS) converted two of the most powerful tools in web design into forms that are suitable for those who need to build mobile websites. At the same time, WYSIWYG (what
you see is what you get) editors currently provide templates for mobile websites, which makes it possible for just about anyone to design simple websites.

Perhaps the biggest reason libraries need to embrace this growing technology is because the youth of the world use it to speak to one another. The average teen, if given the option, would probably prefer to text a librarian than to visit the reference desk.

**WHAT IS A MOBILE APP?**

Another question that should be answered before proceeding any further is, What do I mean by “mobile app”? Mobile apps generally refer to two different things:

**Mobile website:** A mobile website is mobile phone friendly and developed taking into consideration the limitations of cellular devices. If you invested the right amount of time in your library’s website, it probably looks pretty good on a ten- to twenty-inch screen (the size of the average desktop or laptop screen). But if you try to access the website from a mobile phone, it probably will look pretty lousy—if it even loads at all. Having a mobile-friendly website would simply mean that the library has a separate web address (e.g., m.library.com—the m standing for mobile) to point mobile phones to. This can be done easily by inserting code that redirects the device on the basis of its resolution. For example, if the site detects a device resolution of 1280 × 1024, then it’s obviously a desktop, laptop, or tablet computer, and so would be directed to the main page; if the resolution is 320 × 480 or less, then it’s some sort of mobile device, and so would be directed to a mobile page.

**Native app:** A native app is one for which the library has gone the extra step to create an app that is available for purchase in mobile app stores (e.g., iTunes, Android Market)

At the very least, every library should have a mobile-friendly website; it’s a relatively easy process that I’ll talk about in forthcoming chapters.

**DEVELOPING A MOBILE APP**

Libraries that want to take the extra initiative and do something more innovative should devote some time to developing a plan for the delivery of native apps to their users. To develop a native app, you have to understand how to develop a mobile website app—so by understanding one, you are actually learning to do two things. By the end of this book, you might realize that you don’t want to have a native app, but you will at least know how to develop one, and in doing so, you will be able to develop a mobile app.

The problem with developing a native app isn’t the skill level (which is actually about the same as for developing a mobile-friendly website); the problem is the number of phones that are on the market. If you want to develop a native app for all users, then
you have to develop one for every phone: iPhone, BlackBerry, Android, Palm, Windows Mobile—and those are just the major ones! In short, developing a native app requires a substantial time investment, and each library needs to consider that when deciding whether to invest in a native app.

The good news is that most phone makers supply app developers with plenty of free tools and resources to make development a little easier. The bad news is that actually getting the app into the marketplace requires a bit of money; each phone requires that developers pay a fee, and some apps are compatible only with Mac operating systems, such as those for the iPhone. These costs, though, help prevent people who are not serious about app development from flooding the markets with apps. Box 1 illustrates the system requirements and fees for each of the five major phones.

The information in box 1 can be a little overwhelming, and chances are that you will not be able to get a library app on every phone. What you need to consider, however, is that if you get a native app on just two phones (e.g., iPhone and Android), then it will be compatible with most smartphone users. In the summer of 2011, more than half of

---

**Box 1: System Requirements and Fees**

<table>
<thead>
<tr>
<th>Platform</th>
<th>Fee</th>
<th>System Requirements</th>
</tr>
</thead>
<tbody>
<tr>
<td>Android</td>
<td>$25 unlimited</td>
<td>The Android software development kit (SDK) is free. It requires</td>
</tr>
<tr>
<td></td>
<td></td>
<td>▪ Windows XP (32-bit) or Vista (32- or 64-bit)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>▪ Mac OS X version 10.5.8 or later (x86 only)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>▪ Linux (tested on Linux Ubuntu Hardy Heron)</td>
</tr>
<tr>
<td>BlackBerry</td>
<td>$200 for 10 app submissions</td>
<td>The BlackBerry SDK is free. It requires</td>
</tr>
<tr>
<td></td>
<td></td>
<td>▪ Windows 2000 SP1 or later, or Windows XP</td>
</tr>
<tr>
<td></td>
<td></td>
<td>▪ 32-bit Windows Vista (BlackBerry JDE v4.2.1 and higher)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>▪ Java SE JDK v6.0</td>
</tr>
<tr>
<td>iPhone</td>
<td>$99 per year</td>
<td>The iPhone SDK is free. It requires Mac OS X or later to run the program.</td>
</tr>
<tr>
<td>Palm</td>
<td>$99 per year</td>
<td>The Palm SDK is free. It runs on Mac, Linux, and Windows.</td>
</tr>
<tr>
<td>Windows Mobile</td>
<td>$99 per year</td>
<td>The Palm SDK is free. It runs on Mac, Linux, and Windows.</td>
</tr>
</tbody>
</table>

---

www.alastore.ala.org
smartphone users are expected to have iPhones; Android is catching up quickly, with more than 30 percent of the market. That means that if ten people walk into a store to buy a smartphone, eight or nine will walk out with either an iPhone or a phone with the Android operating system.

THE ESSENCE OF THE MOBILE WEB

Mobile websites should look and interact much differently than traditional websites. Mobile websites involve different design concerns than regular websites that you view on a computer. In some ways, designing mobile websites is like designing sites in the past, when the download speed of a page was of primary concern. Although today most mobile connections are quite fast, some mobile phone users pay by the minute for Internet connectivity; thus, it’s in their best interest to access sites that download in short order.

All of this plays into simplicity, a major component of all good mobile websites. Regular websites offer a host of options to visitors, as well as graphics, images, navigation menus, Flash elements, and JavaScript elements that add functionality and style to the page. But where mobile websites are concerned, the aesthetic elements have to work in hand in hand with practical concerns. On a small screen, there’s no room for overblown design elements, multiple navigation menus, or a great deal of text. Unlike regular websites, mobile sites also require that you accommodate the technology that users have to view them.

The language of the Web is HTML, or HyperText Markup Language. For older mobile browsers, the language is WML, or Wireless Markup Language. Today smartphones and mobile browsers generally make use of XHTML-MP. The differences between HTML and XHTML-MP aren’t too radical, and anyone with HTML experience should be able to figure out XHTML-MP with little difficulty. There are other requirements that mobile websites have to accommodate to make sure that their elements display correctly on all of the many mobile browsers on the market. Some of the features they require are the following:

- very simplified navigation
- layouts specified using Cascading Style Sheets (CSS), not tables
- compact, efficiently written content
- a color scheme that is consistent across all browsers
- reduction in bandwidth-heavy elements, such as pictures, videos, and audio
- navigation options such as “back” and “next” on every page

Most mobile websites are built around a utilitarian aesthetic. They aren’t the places to show off your design department’s abilities or to debut new and untested features. When people surf the mobile-friendly Internet, they’re generally looking to find the information they want as quickly as possible, and that information is usually not for pure entertainment. Efficiency is always the first order of business with mobile sites.
For example, for library sites, most users will primarily look to browse the catalog and reserve items from the stacks, to call the library, and to get directions to different branches. This means that the mobile site should be integrated with the library’s database to allow customers to log in and access their account.

Another concern in developing mobile websites is space. A smartphone, useful as it is, has a small-format screen. This means that you must maximize the space available and not create a mobile site that is useful only to those users who have relatively large mobile screens.

It’s also important to see the actual phones and know what they look like. Box 2 provides the range of dimensions for most smartphones; the average phone has a screen dimension of 320 × 480 (about the same as a playing card); ideally, that is the resolution that you should aim to develop your mobile website for.

### BOX 2 ■ Dimensions in Pixels

- **iPhone**: 320 × 480
- **Blackberry**: 160 × 160 to 480 × 360
- **Android SDK**: 320 × 240 (because different phones use the Android platform, here the dimension is the standard resolution in the developer kit)
- **Generic Windows Mobile**: 240 × 320 (standard resolution, but varies from phone to phone)
- **Nokia**: 95 × 95 and higher (varies from phone to phone, but 240 × 320 is standard)
- **Palm Pre**: 320 × 480

### SURVEYING USERS

Before you continue with developing an app, it’s a good idea to understand who your patrons are and what kinds of mobile services they want. For example, what percentage of patrons have smartphones? What kinds of smartphones do they have (many people don’t know which kind of phone they have, so it’s a good idea to ask to see their phone)? If they don’t have smartphones, do they use the Web on their feature phones? Most web analytics sites (e.g., Google Analytics, StatCounter) will indicate users’ resolution, which lets you figure out whether they are using a phone to view the site. But this doesn’t always help because these sites let you know only whether users are accessing the site via their phone, not what kind of phone (e.g., iPhone, BlackBerry, Droid).

A lot of this book focuses on the iPhone, because when people talk about wanting to learn how to make apps, they are usually referring to making them for the iPhone—the holy grail of app development. Depending on your patrons, however, you might want to consider starting on another device. There is no point developing an iPhone app if most of your smartphone-carrying patrons have BlackBerries.

If you decide to continue with developing an app, then you have to ask, What do your patrons want? Access to the library’s catalog? Photos from events? A calendar of events? GPS tracking to find the nearest library or book? To know when the next library
computer is available? Instant messaging with a librarian? It's important to consider that what librarians find important is not necessarily what patrons find important.

**TRAINING STAFF**

The library might face resistance early on from staff. Many people don't like change, so to avoid resistance, it's useful to get everyone on staff involved as early as possible. Another good idea is that when you survey your patrons, you can also survey your staff to find out who already understands the technology and wants to jump on board and who will need a little bit more help. Also, it helps to show, not tell. Basically, that means that it's better to show a scene than to simply talk about what is happening. So, if you simply tell staff what the library wants to do, many might not see the necessity of an app, but if you show them the app and what it can do for the library, they're more likely to get excited.

How do you show? The best way is to carry out a survey on an iPod Touch or Android tablet. SurveyMonkey and other popular online surveys are great tools for collecting data, but actually placing a device in the hand of a librarian unfamiliar with such devices will help him or her become more comfortable with using the interface. These devices also are small, so they're easy to store. And for surveying patrons, you can buy an iPod lock (e.g., the Targus Defcon Notebook/iPod Lock Combo costs $39.99) or lock an iPod to an unattended desk for patrons to fill out the survey.

The iPod and Android tablets also use the same kind of interface as the iPhone, and there are plenty of survey apps that you can purchase for them; best of all they are relatively cheap (less than $200). The Samsung Galaxy and Dell Streak tablet have received a lot of media attention, but there are many other tablets available—there are many Android tablets available on eBay for less than $150.

You can use Survey on the Spot (free for iPhone) or Askdroid and ODK Collect (free for Android) to create a survey and then have librarians use the app to ask patrons a series of questions. Or use Tally Counter (free for iPhone) or CodeArk Tally Counter (free for Android) to count the number of people using smartphones. More powerful (and detailed) tally counts are also available for a cost: Tallymander ($3.99 for iPhone) and Advanced Tally Counter Pro ($0.99 for Android) let users count several different stats at once (e.g., types of questions patrons ask at the reference desk).

The results of the surveys, in most cases, won't be as powerful or specific as those you would get from sites like SurveyMonkey, but that doesn't matter; what matters is your letting people who don't regularly use the device know what it is. Even though people might resist change, almost everyone loves to play with gadgets!

Many times libraries implement technological innovations that don't take off because the library staff members don't promote them to patrons. But in surveying staff and patrons on an iPhone or tablet, even though you might not convert anyone to a smartphone user, you will at least give people a better understanding of their importance.
Once you know who your users are, it’s time to start developing. This chapter and the following ones provide a crash course in mobile development. We’ll review the essence of HTML, CSS, and JavaScript programming, and we’ll learn the very basics of what it takes to develop a mobile website.

SETTING UP THE DOMAIN

Mobile websites are housed in a separate part of your server from your main site. In most cases, the site name will simply have “mobile” added on at the beginning of the domain. For example, the mobile version of the URL YourLibrary.com would be Mobile.YourLibrary.com. The “Mobile” directory holds all the files required to display your mobile site. This type of organization also makes things easier on your information technology staff, because the “Mobile” directory is a convenient way to deal with a site.

Once your domain is set up and ready to go, the hard work begins.

UNDERSTANDING THE PROCESS

Most of the process of designing a mobile website involves simplifying and reformatting the information and features that appear on your main website. This means that, to build a mobile website, you first need to prepare the text and images for the site so that they fit the site’s format.

MOBILE WEB VERSUS STANDARD WEB

Everything on your mobile website will be slightly different from how it appears on your regular website. This is inevitable. Although there are some features that your users will expect from your website, some of what you offer on your main site is impractical for mobile users. It’s important to remember that you’re trying to replicate the functionality and content of your regular website as much as possible on your mobile site. You’re not simply duplicating your main website and putting it in your mobile directory.
INDEX

A
absolute positioning (CSS elements), 16
accelerometer (PhoneGap feature), 26
Advanced Tally Counter Pro, 6
Appanda (website), 46
AppBreeder (website), 46
apps
installing, 33–37
moving, 37–39
testing users, 39–40
using widgets, 41–44
Askdroid, 6

B
Build an App (website), 47

C
camera (PhoneGap feature), 26
Cascading Style Sheets (CSS)
about, 13
creating documents, 17
ID selectors, 14–15
positioning elements, 15–17
tables versus, 15–16
usage considerations, 15
WCSS, 13–14
chat-based reference services, 42
CodeArk Tally Counter, 6
content for mobile apps, 9–10
CSS: The Missing Manual
(McFarland), 13

D
domains, setting up, 7

F
Facebook (website), 42
feature phones, defined, 1
floating technique, 16–17
forms, JavaScript and, 21
Foursquare (website), 45

G
Gale’s AccessMyLibrary app, 40
Game Salad (website), 47
golocation (PhoneGap feature), 26
Goodman, Danny, 19
Google App Inventor, 47–49
Google Talk, 42
 Gowalla, 45
HyperText Markup Language
(HTML)
about, 4
CSS and, 13–14
JavaScript and, 20
ID selectors, 14–15
images, mobile app, 11
interactivity for mobile apps, 10
iOS Dev Center site, 37
iPhone. See smartphones
iPhone SDK, 33
JavaScript
basics of, 20–21
compatibility considerations, 22–23
downloading, 19
forms and, 21
support for, 23
usage considerations, 21–24, 44
JavaScript Bible (Goodman), 19
LibraryH3lp (website), 42–43
McFarland, David, 13
Meebo (website), 42
Mobile App Loader (website), 47
mobile apps
alternatives for, 45–49
content for, 9–10
defined, 1–2
developing, 2–4, 7–11, 17–18
interactivity for, 10
navigation for, 8, 17–18, 22
surveying users, 5–6
testing core users, 39–40
widgets in, 41–44
WYSIWYG app editors, 46–47
mobile browsers
languages supported, 4, 22
navigation considerations, 8, 17–18, 22
size considerations, 15
mobile websites
defined, 2
essence of, 4–5
JavaScript and design, 19–24
redirecting to, 28
standard websites versus, 7–8
viewing on PhoneGap, 26–28
MockApp (website), 29
moving apps, 37–39
My App Builder (website), 46

N
native apps
creating, 29–33
defined, 2
installing on smartphones, 33–37
testing, 37–40
navigation for mobile apps, 8, 17–18, 22

O
ODK Collect, 6

P
PhoneGap software
code for sharing, 44
downloading, 33
<table>
<thead>
<tr>
<th>PhoneGap software (cont.)</th>
<th>staff training, 6</th>
</tr>
</thead>
<tbody>
<tr>
<td>features, 25–26</td>
<td>Survey on the Spot, 6</td>
</tr>
<tr>
<td>implementing, 28</td>
<td>surveying users, 5–6</td>
</tr>
<tr>
<td>redirecting to mobile sites, 28</td>
<td>SurveyMonkey, 6</td>
</tr>
<tr>
<td>viewing websites, 26–28</td>
<td>Swebapps (website), 46</td>
</tr>
<tr>
<td>positioning CSS elements, 15–17</td>
<td>振动 (PhoneGap feature), 26</td>
</tr>
<tr>
<td>provisions profiles, 37</td>
<td>Wireless CSS (WCSS), 13–14</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>R</th>
<th>T</th>
<th>W</th>
</tr>
</thead>
<tbody>
<tr>
<td>redirecting to mobile websites, 28</td>
<td>tables versus CSS, 15–16</td>
<td>web analytic sites, 5</td>
</tr>
<tr>
<td>reference services, chat-based, 42</td>
<td>Tally Counter, 6</td>
<td>Widgetbox, 42, 44</td>
</tr>
<tr>
<td>relative positioning (CSS elements), 16</td>
<td>Tallymander, 6</td>
<td>widgets in mobile apps, 41–44</td>
</tr>
<tr>
<td></td>
<td>testing native apps, 37–40</td>
<td>Widgipedia, 44</td>
</tr>
<tr>
<td></td>
<td>training staff, 6</td>
<td>Wireless CSS (WCSS), 13–14</td>
</tr>
<tr>
<td></td>
<td>Twitter (website), 41–42</td>
<td>Xcode, 34–36</td>
</tr>
<tr>
<td></td>
<td></td>
<td>XHTML-MP, 4</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>S</th>
<th>V</th>
<th>X</th>
</tr>
</thead>
<tbody>
<tr>
<td>simulators, 37–39</td>
<td>vibration (PhoneGap feature), 26</td>
<td>Xcode, 34–36</td>
</tr>
<tr>
<td>smartphones</td>
<td></td>
<td>XHTML-MP, 4</td>
</tr>
<tr>
<td>defined, 1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>dimensions in pixels, 5</td>
<td></td>
<td></td>
</tr>
<tr>
<td>installing native apps on, 33–37</td>
<td></td>
<td></td>
</tr>
<tr>
<td>system requirements and fees, 3</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

|                       | World Conference On Soft Computing (WCS3), 11 |
|                       | WYSIWYG app editors, 46–47 |

|                       | Wireless Markup Language (WML), 4 |

www.alastore.ala.org
You may also be interested in

**DOING SOCIAL MEDIA SO IT MATTERS**  
A LIBRARIAN’S GUIDE  
Laura Solomon

Facebook, Twitter, MySpace, LinkedIn: it’s difficult enough to keep abreast of social media Web sites, let alone understand how they fit into today’s library. This practical resource brings together current information on the topic in a concise format that’s easy to digest.

PRINT: 978-0-8389-1067-2  
EBOOK: 7400-0672  
PRINT/EBOOK BUNDLE: 7700-0672  
80 PGS / 8.5” x 11”

**OPEN ACCESS**  
WALT CRAWFORD  
ISBN: 978-0-8389-1106-8

**GRASSROOTS LIBRARY ADVOCACY**  
LAUREN COMITO, ALIAE GERACI AND CHRISTIAN ZABRISKE  
ISBN: 978-0-8389-1134-1

**PRIVATIZING LIBRARIES**  
JANE JERRARD, NANCY BOLT, AND KAREN STREGE  
ISBN: 978-0-8389-1154-4

**USING WORDPRESS AS A LIBRARY CONTENT MANAGEMENT SYSTEM**  
KYLE M. L. JONES AND POLLY-ALIDA FARRINGTON  

**BUILDING THE DIGITAL BRANCH**  
DAVID LEE KING  

**WEB SCALE DISCOVERY SERVICES**  
JASON VAUGHAN  

Order today at alastore.ala.org or 866-746-7252!  
ALA Store purchases fund advocacy, awareness, and accreditation programs for library professionals worldwide.
Patrons increasingly expect access to their libraries anywhere, anytime. This ALA Editions Special Report provides practical guidance for how librarians can put the library in the palms of their patrons’ hands. Using the HTML skills that many librarians already have, combined with flexible development tools, technology expert Scott La Counte shows how creating a customized mobile app doesn’t need to be expensive or require deep expertise. In straightforward, practical terms he

- Demonstrates how to establish a presence on the mobile web with mobile websites and phone apps
- Details open-source development tools such as PhoneGap that enable the creation of mobile apps that work on a variety of mobile operating systems
- Discusses methods for assessing a library’s user base and for getting buy-in from administrators

Following the pointers in this ALA Editions Special Report, libraries can easily go wherever their patrons do!