Build and integrate maps, geo-location based services, and more into your apps

Pro iOS Geo

Building Apps with Location Based Services

Giacomo Andreucci

Apress®

Pro iOS Geo

Building Apps with Location Based Services

Giacomo Andreucci

Apress®

Pro iOS Geo: Building Apps with Location Based Services

Portions of this book are inspired by *Applicazioni iOS e Android con Google Maps* (Edizioni FAG Milano) by Giacomo Andreucci published September 2011.

ISBN 978-1-4302-4704-3

ISBN 978-1-4302-4705-0 (eBook)

Trademarked names, logos, and images may appear in this book. Rather than use a trademark symbol with every occurrence of a trademarked name, logo, or image we use the names, logos, and images only in an editorial fashion and to the benefit of the trademark owner, with no intention of infringement of the trademark.

The use in this publication of trade names, trademarks, service marks, and similar terms, even if they are not identified as such, is not to be taken as an expression of opinion as to whether or not they are subject to proprietary rights.

While the advice and information in this book are believed to be true and accurate at the date of publication, neither the authors nor the editors nor the publisher can accept any legal responsibility for any errors or omissions that may be made. The publisher makes no warranty, express or implied, with respect to the material contained herein.

President and Publisher: Paul Manning
Lead Editor: Steve Anglin
Developmental Editor: James Markham
Technical Reviewer: Nishant Prant
Editorial Board: Steve Anglin, Mark Beckner, Ewan Buckingham, Gary Cornell, Louise Corrigan, Morgan Ertel, Jonathan Gennick, Jonathan Hassell, Robert Hutchinson, Michelle Lowman, James Markham, Matthew Moodie, Jeff Olson, Jeffrey Pepper, Douglas Pundick, Ben Renow-Clarke, Dominic Shakeshaft, Gwenan Spearing, Matt Wade, Tom Welsh
Coordinating Editor: Katie Sullivan
Copy Editor: Jill Steinberg
Compositor: Spi Global
Indexer: Spi Global
Artist: Spi Global
Cover Designer: Anna Ishchenko

Distributed to the book trade worldwide by Springer Science+Business Media New York, 233 Spring Street, 6th Floor, New York, NY 10013. Phone 1-800-SPRINGER, fax (201) 348-4505, e-mail orders-ny@springer-sbm.com, or visit www.springeronline.com. Apress Media, LLC is a California LLC and the sole member (owner) is Springer Science + Business Media Finance Inc (SSBM Finance Inc). SSBM Finance Inc is a Delaware corporation.

For information on translations, please e-mail rights@apress.com, or visit www.apress.com.

Apress and friends of ED books may be purchased in bulk for academic, corporate, or promotional use. eBook versions and licenses are also available for most titles. For more information, reference our Special Bulk Sales–eBook Licensing web page at www.apress.com/bulk-sales.

Any source code or other supplementary materials referenced by the author in this text is available to readers at www.apress.com. For detailed information about how to locate your book's source code, go to www.apress.com/source-code/.

To the awesome rock bands that I have listened to all my life, from U2 to Truffle Shuffle!

Contents at a Glance

Contents

About the Author

Giacomo Andreucci has a Ph.D. in History and Geography at the University of Bologna, Italy. His main research interests embrace the use of new media in historical and geographical/cartographical research. He is also tech writer for the Italian publisher FAG for which he wrote books on Google Earth/Maps API, and SketchUp. More information about his books is available here: http://www.fag.it/autore_andreucci_giacomo_181.aspx.

About the Technical Reviewer

Nishant Pant is a software architect, mentor, and web and mobile developer with more than 15 years of experience in all aspects of software development. He is also the co-founder of Loqly, a platform for connecting local businesses with their customers.

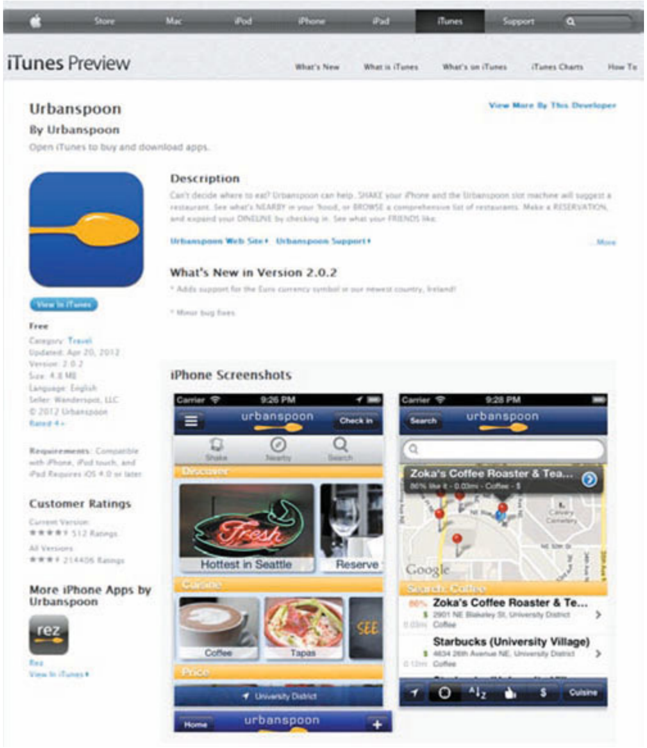
Acknowledgments

Eric Racher proofread the text. I thank him for his patience and the quality of his work.

Introduction

Since its first release in 2007, iOS has offered geographic services: the (Google) Maps app was bundled with it by default. When the iOS SDK was released in 2008, developers were given the opportunity to implement geo components in their apps using the Map Kit Framework based on Google Maps services. Annotations (such as points of interest) could be overlaid on the maps provided by Google, and it became possible to use geocoding services and iOS geolocation to display data on the maps. In 2008 Google released its powerful Google Maps JavaScript API V3, which was optimized for use on mobile devices. This new API has since become the most widely used platform to implement maps services on the web, both for desktop computers and for mobile devices. In the meantime, competing platforms were created to deliver geo data through websites and web apps: Open StreetMap was (and still is) one of the most popular, even if it is less so than Google geo services. (Other competing platforms include Microsoft Bing Maps and Yahoo Maps.)

Thanks to these geo platforms, many iOS developers have featured geo functionalities in their apps, not only for iPod Touch and iPhone devices but for the iPad as well. The iPad is a very powerful device because it permits users to comfortably display and manipulate cartographic and geographic data, due to its dimensions and its geolocation capabilities. In the Apple App Store thousands of apps make use of online cartography and geo services, the majority of which rely on the Google Maps platform. Among those map-based apps, both free and commercial, some can display homes and apartments for sale within a certain distance from your current location, while other apps find parking lots, movie theaters, and restaurants (Figure 1).



Urbanspoon

By Urbanspoon

Open iTunes to buy and download apps.

Description

Can't decide where to eat? Urbanspoon can help. SHAKE your iPhone and the Urbanspoon slot machine will suggest a restaurant. See what's NEARBY in your 'hood, or BROWSE a comprehensive list of restaurants. Make a RESERVATION, and expand your DINELINE by checking in. See what your FRIENDS like.

What's New in Version 2.0.2

* Adds support for the Euro currency symbol in our newest country, Ireland!

* Minor bug fixes

Free

Category: Travel
Updated: Apr 20, 2012
Version: 2.0.2
Size: 4.8 MB
Language: English
Seller: Wanderspot, LLC
© 2012 Urbanspoon
Rated 4+

Requirements: Compatible with iPhone, iPod touch, and iPad Requires iOS 4.0 or later

Customer Ratings

Current Version:
★★★★½ 512 Ratings
All Versions:
★★★½ 214406 Ratings

More iPhone Apps by Urbanspoon

Rez

iPhone Screenshots

Carrier 9:26 PM

urbanspoon

Carrier 9:28 PM

urbanspoon

Zoka's Coffee Roaster & Tea...
86% like it - 0.03mi - Coffee - $

Search: Coffee

86% Zoka's Coffee Roaster & Te...
$ 2901 NE Blakeley St, University District
0.03mi Coffee

Starbucks (University Village)
$ 4634 26th Avenue NE, University District
0.12mi Coffee

***Figure 1.** Urbanspoon is an example of a successful geo app (http://itunes.apple.com/us/app/urbanspoon/id284708449?mt=8)*

There are also guidebook-style apps that can give you information in real time about nearby monuments. These apps combine location data, maps, and StreetView imagery to create fascinating augmented reality experiences. Augmented reality apps are flourishing given the growing hardware capacities of mobile devices and the spread of broadband connections through 3G and 4G services. These apps can be used in many different fields, ranging from commercial guides to tourism or cultural initiatives supported by the public administration, institutions, museums, and so on. Additionally, social networks have been implementing geo services in their apps in recent years.

In this blossoming landscape, an important change took place in June 2012, when at the WWDC in San Francisco Apple announced its own new mapping platform, included in iOS 6 (Figure 2). This platform is going to take the place of Google Maps in native iOS apps. It was a commercial move to limit the power and earnings of Google in the iOS world. The iOS 6 default Maps app is indeed no longer based on Google services but on the new Apple Maps platform.

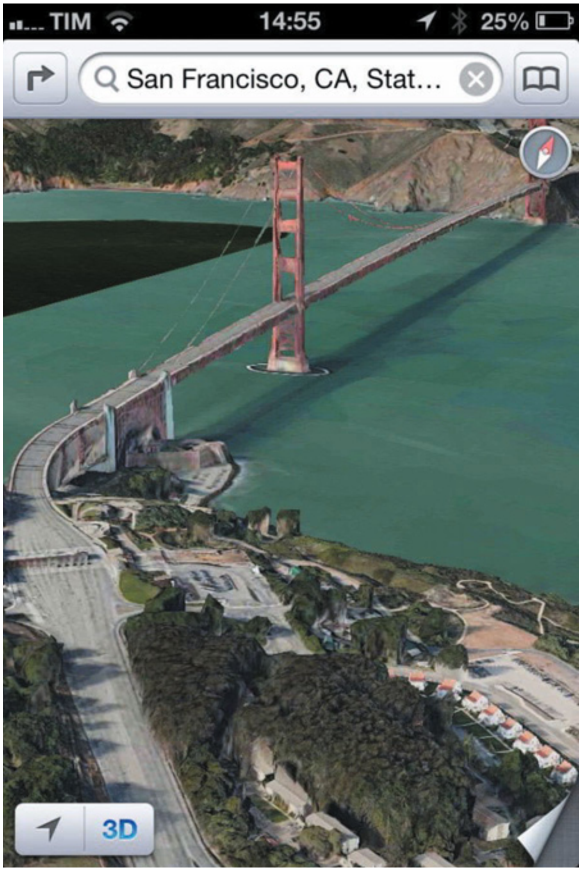


***Figure 2.** On June 11, 2012, at the WWDC in San Francisco, Apple announced its own mapping platform intended to replace Google Maps services*

Similarly, the Apple Map Kit framework is no longer based on Google Maps but on the Apple maps platform and this affects all new native apps that are going to be developed for iOS. Apple's own mapping platform has become the reference point for any developer who wants to develop native iOS apps, while Google Maps remains the main mapping platform used in developing web apps for iOS since its powerful JavaScript API is still the best solution for the wealth and quality of services it offers.

The Purpose of This Book

The purpose of this book is to introduce the reader to the primary resources that are currently available for developing map-based services for iOS apps. As I have mentioned, the geo dimension can play different roles inside an app. The geo component might just be an accessory to a larger app (such as a social networking app that, among its functions, can show the location of friends in the vicinity) or it might constitute the main part of an app (such as an app whose only purpose is to show the nearest underground station).

In this book I will not focus on a specific genre of apps. Instead, I present a series of ideas and suggestions on how best to integrate and exploit the geo resources currently available for developers. However, in Parts 2 and 3 of the book (see the next section), I present a case study, which is the realization of a geo app, named "U2's Dublin." The purpose of this app is to show landmarks related to U2's rock band history implementing some geo functions we will be talking about extensively in the next chapters.

How This Book is Organized

As I will explain in more detail in the first chapter, it is possible to integrate geo services in an app in different ways, depending on the kind of app we are developing: a web app, a hybrid app, or a native app. This book deals with all three ways of creating an app.

After establishing the groundwork for building iOS map-based apps in Chapter 1 (Part 1), Chapters 2–6 (Part 2) focus on web apps. A greater amount of space is given to Part 2 because I believe that web apps are the easiest and most effective way to implement powerful geo services in an iOS app, which makes them easily accessible to other mobile platforms as well. The drawback to web apps is that they cannot be directly uploaded and distributed through the App Store. In the first section of the book I mainly focus on the Google Maps JavaScript API V3. Since web apps are the subject of this first part, it is necessary for the reader to have a basic knowledge of HTML5 and JavaScript.

Part 3 (Chapter 7) covers hybrid apps, which are web apps accessed through a browser window embedded in a native app. Hybrid apps are like web apps but they can be uploaded and distributed through the App Store. To develop hybrid apps, it is not necessary to know Objective-C; it is sufficient to know how to develop a web app and have basic knowledge of the Xcode environment.

In Part 4 (Chapters 8–10) I cover the new Apple Map Kit framework, which makes it possible to integrate mapping services in native iOS apps. The reader of this last part needs basic knowledge of the Xcode environment and Objective-C language.

This book is organized so that readers may choose to read the section or sections in which they are interested: knowledge of earlier chapters is not necessary for a thorough understanding of later sections. Nonetheless, I strongly recommend that everyone read the first chapter since it is a general introduction to developing geo and mapping functionalities for mobile devices. I also recommend that everyone read Appendix A of the book since it discusses Terms of Use and legal aspects related to the services mentioned in the book.

What You Won't Find in This Book (and Where to Find It)

As I mentioned earlier, readers must know certain languages depending on which sections of the book they are interested in. In this book I take that knowledge for granted and I do not provide introductions to the above-mentioned languages. The reader who is not familiar with the languages can easily learn them by consulting online resources or buying specific books. Regarding the languages necessary for the development of web apps, which are mainly HTML5 and JavaScript, I recommend the free courses at W3C Schools, `http://www.w3schools.com/`. For the Google Maps JavaScript API, I suggest bookmarking the official documentation site, which also hosts many tutorials, `https://developers.google.com/maps/documentation/javascript/`.

I also suggest reading the Apress title, *Learn HTML5 and JavaScript for iOS* by Scott Preston, Apress 2012.

For people interested in developing native apps it is a good idea to be familiar with the documentation hosted in the iOS Dev Center at the official Apple developers site (`https://developer.apple.com/devcenter/ios/index.action`). A good book to begin developing with iOS is *Objective-C for Absolute Beginners. iPhone, iPad and Mac Programming Made Easy* by Gary Bennett, Mitchell Fisher, Brad Lees, 2nd Edition, Apress 2011.

To grasp the concepts in the second and third sections of the book (hybrid and native geo apps), it is also necessary to have a working Xcode /iOS 6 development environment and to be enrolled in the iOS Developer Program (`https://developer.apple.com/programs/ios/`).

The Companion Site

In my blog I deal with topics related to online maps and mobile map design. The site also makes available all the code samples I have used in this book.
`http://www.progettaremappeonline.it`

Contacting the Author

If you have any questions or comments – or you spot a mistake you think I should know about – you can contact me at `giacomo.andreucci@unibo.it`.

Happy reading!

Part 1

Introduction

Chapter 1 lays the foundation for the rest of the book by comparing computer and mobile map-based apps, and highlighting the primary characteristics of mobile map, such as its key services and functionalities. You will also find out about the available options for building iOS map-based apps.

Chapter 1

Getting started

Throughout most of history, maps were used mainly in static rather than dynamic form. For example, on ships they were placed flat on a surface to allow for easy readability. Measurements and other notations were drawn recorded on the map itself. With the growth of cartography in the first half of the 20th century, maps became easily portable and handy, showing up in road atlases and guidebooks. Paper maps thus became fully mobile. The advent of smartphones, tablets and other mobile devices equipped with broadband Internet connectivity, GPS, and other features brought about the introduction of mobile cartography (now made digital).

Thanks to their sensors, geolocation capabilities, elaboration power, and display quality, iOS devices can implement map-based apps that give users a unique geo experience. In this chapter I lay the groundwork for building iOS map-based apps. First, I focus on the main differences between computer and mobile map-based apps, highlighting the primary characteristics of mobile maps. I then present a list of key services and functionalities mobile maps can offer to make you more aware of the possible uses of this technology. To build a map-based mobile app it is important to respect some basic design and usability guidelines specific to this kind of app, so later in this chapter I provide a simple set of usability guidelines. Next, I introduce you to some options for building iOS map-based apps. These are web apps based on the Google Maps JS API, hybrid apps, and eventually native apps using the Apple Maps API. I complete the chapter with a brief section on testing, thereby providing you with a complete guide to getting started.

Comparing Desktop Computer Maps and Mobile Maps

Online desktop computer maps, like the maps you can use on `maps.google.com`, are usually accessed in relatively comfortable circumstances, for example, by a user seated at a computer. Users of these maps can view complex legends and show or hide the data levels available on the map. These are also the best conditions to enjoy thematic maps – that is, maps that display the spatial patterns of physical or social phenomenons, such as population density, life expectancy, etc. Here's an example of a thematic map showing housing data within determined areas (Figure 1-1).

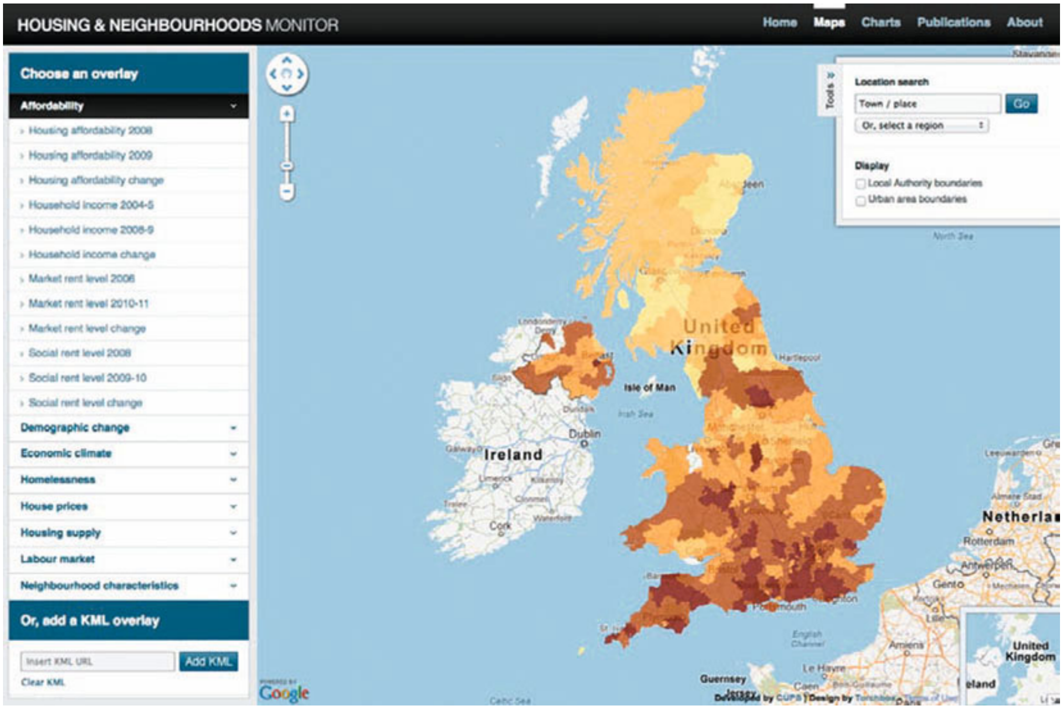


Figure 1-1. Thematic map displaying the affordability of housing in local authority areas in UK in 2008 (`http://www.hnm.org.uk/maps.html`)

Thematic maps like this are best enjoyed on large desktop computer screens, while on small mobile devices they would be very cumbersome to use (Figure 1-2).

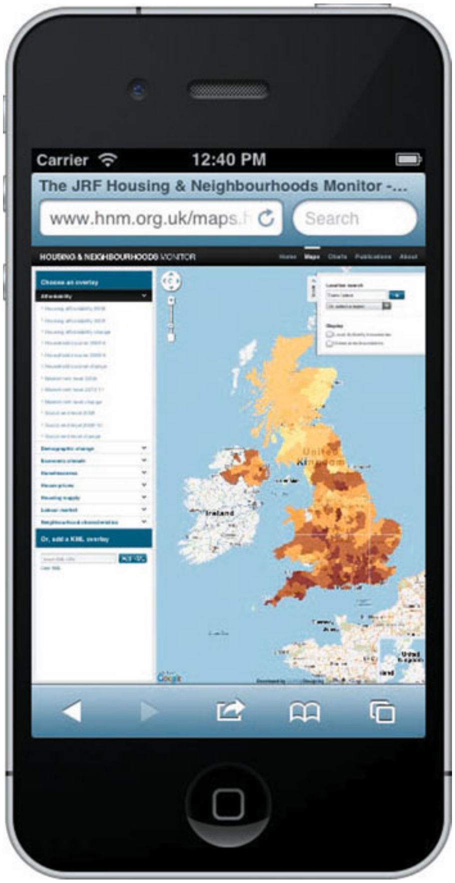

Carrier
12:40 PM
The JRF Housing & Neighbourhoods Monitor -...
www.hnm.org.uk/maps.h
Search
HOUSING & NEIGHBOURHOODS MONITOR
United Kingdom
Ireland

Figure 1-2. The same map displayed on an iPhone

Compared to maps viewed on a large screen, maps on mobile devices have certain limitations due to the technical characteristics of these devices, such as smaller displays, less computing power, and, in some circumstances, limited bandwidth. For example, you can see how the thematic map shown in Figure 1-1 appears clunky on an iPhone device, as shown in Figure 1-2.

These limitations are also related to where and how the mobile device is being used. In a mobility context, the available time to interact with the device may be shorter, the interaction with the map is less comfortable, etc. The user needs to access maps quickly and easily through an interface that does not require many finger movements or extra effort. Users of mobile devices may have less time to react, so geographical services have to be adapted to their specific conditions and limitations.

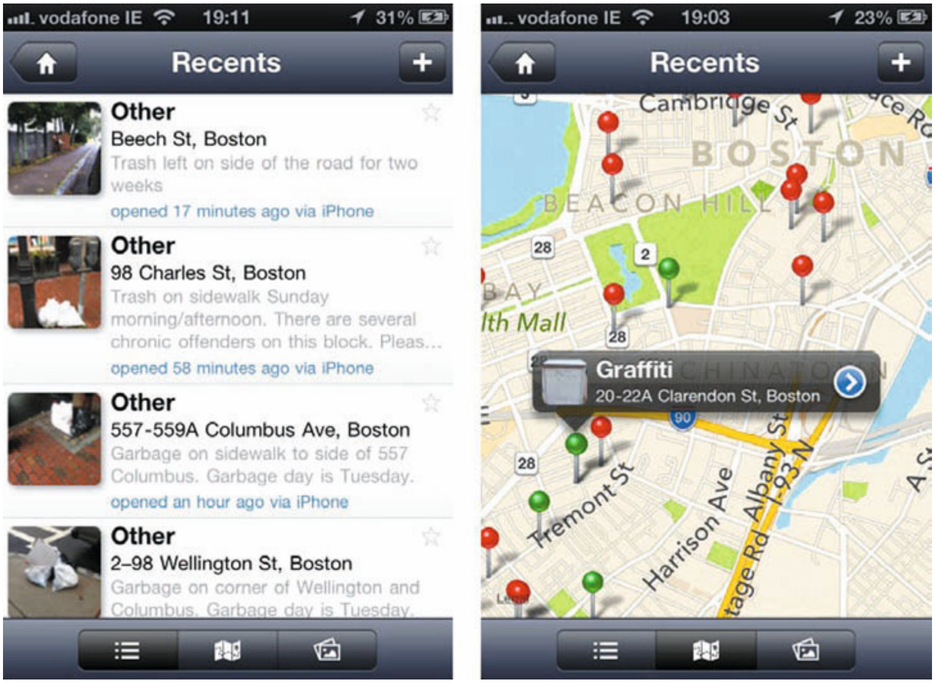
But mobile devices have also created new possibilities for map-based applications. Mobile apps possess a more "individual" character, meaning that they may be more related to the person using them. Mobile apps are indeed context-sensitive. The sensors that smartphones and tablets possess allow mobile apps to know the users' current and previous position data (coordinates, altitude, orientation, speed of movement) in addition to knowing what they have previously visited on the web, what they have searched for, and so on. Technically speaking, we can say that mobile applications are context aware. Users of these devices can take this "awareness" and become producers of new geodata rather than mere passive consumers. For example, GPS and other sensors allow users to georeference many types of data (that is, establishing their location in terms of coordinates), such as photographs, voice recordings, or routes followed while walking. Such a level of awareness is generally not available for geo services on a desktop computer.

Let's now take a closer look at what mobile maps have to offer.

Reviewing Mobile Geo Apps Services

Before you begin to develop your own mobile geo app, it's important to review the main types of geo services that existing apps can offer users. Here is a simple list of map-based services:

- **Navigation-mobility support services.** Included in this category are "You are here" services (real-time calculation of user position and visualization on a map), "You will go" and way-finding services (calculation of routes), navigation services.
- **Information-supply services.** This category includes all those services that allow users to visualize information layers on maps, such as Google Local (ex-Places) categories (restaurants, monuments, museums); in this category "event-calendar" services are also included to show all the current events (exhibitions, concerts) in a user-defined area. Augmented reality services also belong to this group – for example, a guidebook app that can show information layers on the monuments around the user's current position.
- **Information sharing.** This category includes those map-based services that permit users to produce geo information in order to share it. They can be simple services that enable users to share their position data with other users (like Google Latitude, Foursquare, Facebook, and many other social networks) or more complex applications that permit users to record many sets of geodata and share them online. An example: apps created by many municipalities which permit citizens to report urban problems, such as trash, bad road conditions, and abandoned buildings by geolocating them and referencing them with text, pictures, and videos. For example, Citizens Connect is a municipality app that empowers Boston residents to be the City's "eyes and ears" throughout the neighborhoods (Figure 1-3). Through this app, people can alert the City of Boston to neighborhood issues such as potholes and graffiti (`http://itunes.apple.com/us/app/boston-citizens-connect/id330894558?mt=8`).



Other
Beech St, Boston
Trash left on side of the road for two weeks
opened 17 minutes ago via iPhone

Other
98 Charles St, Boston
Trash on sidewalk Sunday morning/afternoon. There are several chronic offenders on this block. Pleas...
opened 58 minutes ago via iPhone

Other
557-559A Columbus Ave, Boston
Garbage on sidewalk to side of 557 Columbus. Garbage day is Tuesday.
opened an hour ago via iPhone

Other
2–98 Wellington St, Boston
Garbage on corner of Wellington and Columbus. Garbage day is Tuesday.

Figure 1-3. An example of an app whose strength lies in its information sharing functionalities

One or more types of the aforementioned services can obviously be integrated into a single mobile app.

Let’s now look at the main usability guidelines to follow while implementing the previously mentioned services.

Designing Guidelines for Optimal Usability of Map-Based Apps

Building a map-based mobile app requires some knowledge of basic design and usability guidelines specific to this kind of app. These guidelines are useful both for creating map-based apps from scratch and for converting already existing geo services into a mobile app. Before I get into the guidelines it is useful to consider the ways in which users commonly interact with maps on mobile devices – so you can avoid having to invent new ways that may confuse users.

Compared to computer users, mobile device users interact with maps in more ways: using physical buttons, touch screens, pens, and voice, or moving the device itself. When using a map-based app, users might employ a single interaction type (monomodal) or a combination of types (multimodal).

For instance, users can pan the map with the touch screen and start a search with vocal commands. The most common types of interaction users can perform while using a map-based app are the following:

- Panning: Users may define the visible portion of the map by dragging the map itself in different directions.
- Zoom: Users may zoom in and out, changing the scale of the map; when users zoom, new map data (with more or less detail) is usually loaded (Figure 1-4).

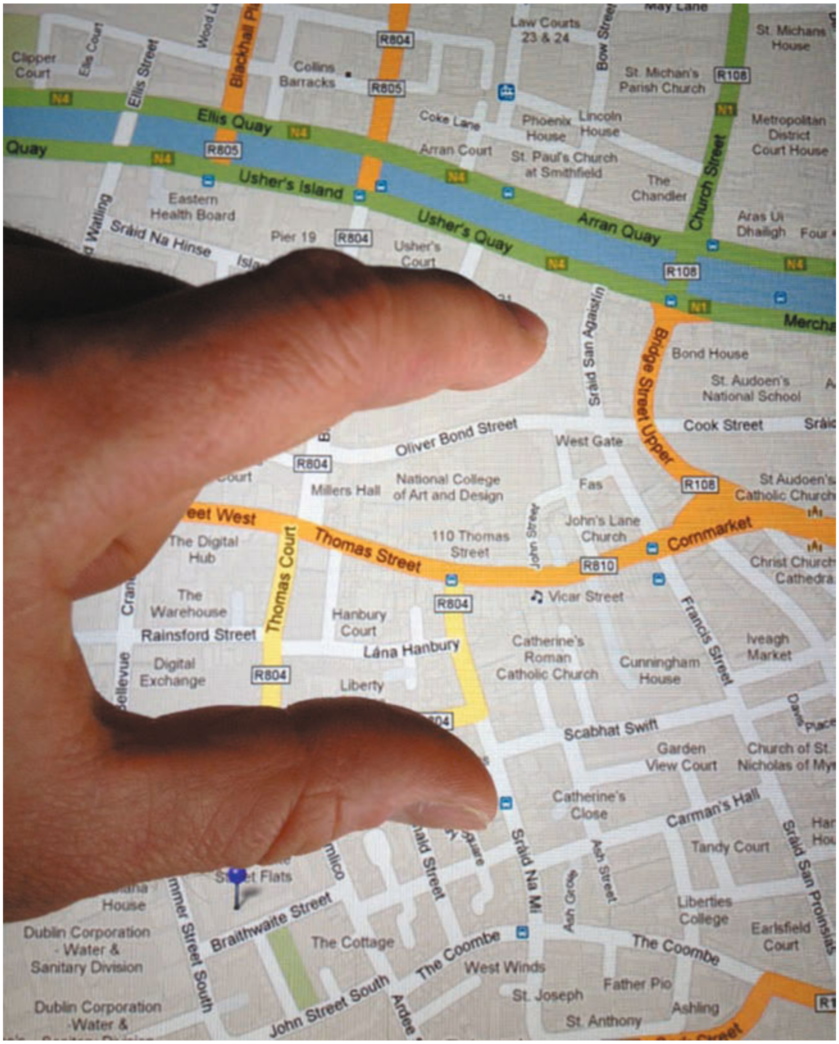


Figure 1-4. Example of user interaction: pinch-zoom performed on an iPad device

- Switching: Users may switch views or presentation styles: For example, they can switch from a view with map data to a view with aerial/satellite imagery or to a view with Street View data.

- Show/hide data layers: Users may show or hide data layers on the map – for instance, a traffic layer or a Panoramio photo layer.
- Focus: Users may click on a specific object on the map – for example, a placemark – and retrieve information connected to it. This information is generally displayed in a balloon or an information window.
- Search: Users can start a search by inputting search terms in a form; the results will be displayed on the map itself.
- Map settings setup: Users can define map settings like map language, map fonts, colors, 3D settings, and GPS activity.

The types of interaction I have just shown mostly relate to access to existing data, but other types of interaction allow users to produce their own geo content (if the app is designed for this purpose). For example, users might drag a marker onto a position on the map to create a new placemark. They might then enter text to fill some fields of the placemark, such as title or description. Typically, users must interact with dialog boxes to produce new geo information.

The developer has to take into consideration these basic types of interaction and be aware of the limitations users of a mobile app are subject to (scarcity of time, uncomfortable circumstances, to name a couple.). The developer is urged to find ways to make the tasks users have to perform as easy as possible. To meet users' needs, the developer can work in three strategic areas:

- Self-adaptability of the app: It is possible to build applications that automatically adapt to users' needs by exploiting the device's sensors.
- Usability of the app's GUI: The way in which buttons and menus are designed is essential to the effective use of the app.
- Usability of map contents: The maps and the geo data should be displayed in such a way that users can quickly and easily decipher them.

Self-Adaptability of the App

It is important to exploit the mobile device's capabilities in the best way possible. If, for instance, the device is equipped with a digital compass, then the map might automatically orient itself, without requiring user intervention. To give another example, if you create an app that shows the user's location, the app might automatically frame the user's current position when that person moves out of the visible area of the map. Creating apps with a good level of adaptability makes it possible for users to focus on essential tasks, without wasting time on settings. The developer can also make an adaptable app by recording the user's habits. For example, if a user generally prefers satellite imagery to default maps when using a certain app, the software itself might remember this setting so that the next time the user opens the app, it will display satellite imagery instead of maps.

Usability of the App's GUI

When designing an app's GUI, a simple but essential guideline to keep in mind is to be in tune with the look and feel of the operating system environment (iOS) and its already existing apps. It is necessary to integrate the GUI into a sort of ecosystem. It would be a frustrating user experience if a GUI displayed different buttons and menus from those interfaces to which the user is accustomed on that particular OS. It is more effective to show users something familiar, instead of trying to seduce them with new interfaces, buttons, and eye-candy menus that require a lot of time to learn how to use. Another important suggestion for a good GUI is to avoid long instructions.

This does not mean that developers shouldn't guide users when necessary, but these situations should be reduced as much as possible with simple, short, and clear instructions.

As far as GUIs are concerned, if we develop map-based apps for iPad devices, it is important to be aware of the capacities of iOS tablets regarding the manipulation of maps and geo data.

Thanks to their resolution (iPad 3 features a 2048-by-1536 pixel resolution), touch screen, Internet connection, GPS chip, and other sensors, iPads are powerful geo devices. They can be used to access maps and also to input geo data in various field more comfortably than is possible using the iPhone. Even if iPads are not usually associated with walking to view maps and access geodata, there are a growing number of people who use them while walking or traveling for their geo capabilities. In fact, by 2010 a Forbes's article reported an increasing availability of map-based apps for iPad devices (http://www.forbes.com/2010/07/14/navigation-gps-apple-technology-ipad.html). The article reported how iPad maps were being used in sports activities like hiking and sailing and how many cartography producers who had not previously released iPhone apps made their first entrance in the App Store with iPad apps (Figure 1-5).

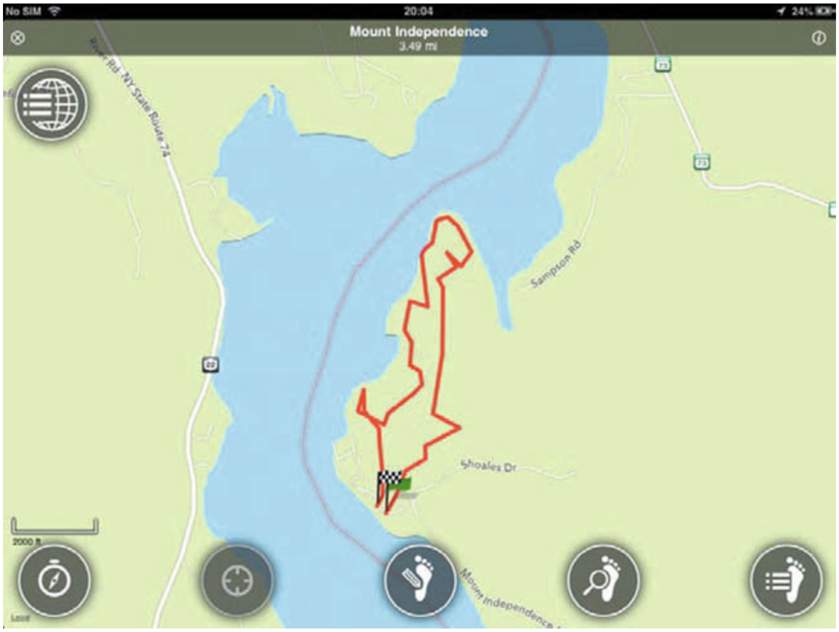


***Figure 1-5.** OutDoors America: Example of a geo app running on an iPad device (http://itunes.apple.com/app/outdoors-america/id451136498?mt=8)*

Usability of Map Contents

Cartography involves the art of reduction and the synthesis of information and these disciplines are well suited to the world of mobile devices, which is constrained by some relevant limitations. Some of these limitations are technological while others are due to the mobility context in which the user *acts.* As I mentioned before, in designing mobile apps it is crucial to avoid anything extraneous, anything that can create confusion or waste the user's time. Here I present a general list of things to avoid when displaying maps and related data layers:

- Avoid useless explanations and image captions. The user might need to be introduced to map data. For instance, a legend might be necessary to explain the symbols used. However it is important to reduce explanations to a bare minimum. For instance it is usually not necessary to explain that the areas in green on a map represent parks or other natural areas.
- Avoid showing a map view overcrowded with data and therefore difficult to decipher in a few seconds.
- Avoid the absence of key areas on the map. If the map is designed to show specific geo data in particular areas, it is important to guide users toward the areas instead of presenting a map where users must find interest data.
- Avoid filling balloons with too much information like large images or long text blocks, which strain users' eyes. Just as text on web pages should be brief, so should those in maps' balloon content; the balloon should open in the shortest time possible and should not occupy all the available visible space, thus completely covering the map.
- Avoid designing complex and overly elaborate icons, which are difficult to read and decipher; the icon's symbol should show a clear relationship with the content to which it refers.
- For icons, lines, and polygons, avoid using colors that are difficult to distinguish when these objects are overlayed on the map. The right choice of colors depends on the map type that we choose as the default map type for the app (for example, satellite images and standard map type).
- Avoid using acronyms and abbreviations that users are unlikely to know. If you must use these shortened forms, find a place to explain them.
- Avoid showing aerial/satellite/street View imagery (available with Google Maps API) or similar cumbersome layers as default map type when the app is opened, unless it is strictly necessary.

In short, it is better to start with a standard map type and permit the user to switch, if he prefers, to other map types, satellite images, etc.

After the review of usability guidelines it is now time to take a close look at the main developing options currently available to realize map-based apps for iOS.

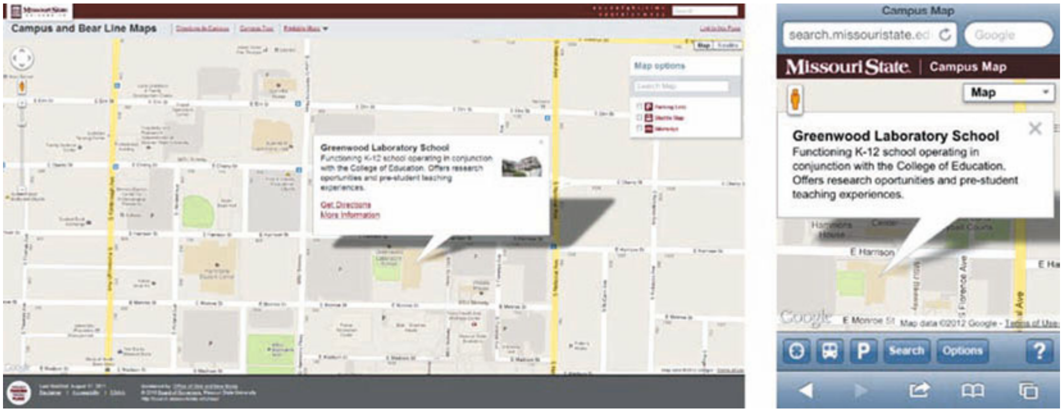
The Primary Options to Develop Map-Based Apps for iOS

As I mentioned in the introduction, this book is organized in three parts, according to the three main available options to develop iOS map-based apps: web apps (Google Maps-based), hybrid apps (Google Maps-based) and native apps (Apple Maps-based).

Developing Web Apps with Google Maps

Google Maps-based web apps are apps developed using web technologies, mainly HTML5, CSS3, and JavaScript. Web apps are accessed directly through the iPhone or iPad's browser. Geolocation functionalities are managed through HTML5 and JavaScript. Maps and other geo data and services are implemented through Google Maps JavaScript API V3 and connected libraries. The Google Maps JavaScript API is by far the best available option for developing a map-based web app. None of the other platforms, such as OpenStreetMap or Microsoft Bing Maps, currently offer such a powerful set of geo data and services through a JavaScript API. The new Apple Maps platform is not even available to be implemented on a web app (it is only for native apps) and in any case it does not yet offer all the range of data and services that Google does. Paradoxically, a web app using Google Maps API V3 can offer certain types of services and data not available on native apps based on Apple Maps.

Developing web apps offers some other relevant advantages when compared to native apps. First of all, with mobile web apps you can reuse already existing map-based services available in the form of web apps for desktop computer users. All it takes is a few lines of code (Figure 1-6).



Greenwood Laboratory School
Functioning K-12 school operating in conjunction with the College of Education. Offers research opportunities and pre-student teaching experiences.

Get Directions
More information

Figure 1-6. The desktop computer map wep app of the University of Missouri (left); the web app optimized for mobile devices (right) (`http://search.missouristate.edu/map` and `http://search.missouristate.edu/map/mobile`)

Compared to native apps, updating a web app is very easy and transparent. It is just a matter of updating the app's code on the web server and *voilà!* The user does not have to update anything but just reloads the web page. Moreover, every time Google updates its API and the geo data connected to that, changes are immediately reflected on the web app. Another advantage is that developers can use the HTML5 geolocation API to easily implement location-aware functionalities in their apps. By using HTML5 Javascript/jQuery it is also possible to develop GUIs very similar in their look and feel to native apps' GUIs. Many developers can develop map-based web apps, as