Chapter 1

Getting Acquainted with Facebook Application Development

In This Chapter
▶ Knowing where to build your application
▶ Understanding Facebook applications
▶ Getting to know the tools
▶ Finding help through documentation

So you’re ready to build a Facebook application! Whether you want to build a simple HTML Facebook tab to welcome visitors to your Facebook Page or a full-blown, fully integrated Web site that enables visitors to log in with their Facebook profile and see their friends, there’s no doubt that Facebook is a powerful platform that improves just about any brand it touches. Whether you’re a marketer looking to just find out what’s possible with Facebook and maybe get your feet a little wet, or an already-experienced developer looking to expand the possibilities of your development experience, in this chapter I show you how Facebook apps can help you achieve your goals.

What’s a Facebook Application?

If you keep reading, you’ll quickly become acclimated to what a Facebook application is and why you might want to consider using such a thing. However, there may be a good chance you’ve never even touched Facebook Platform before and you want to find out why so many other developers and businesses are interested in Facebook.

Facebook applications basically come in two forms:

✓ Facebook hosted. This is typically what most people mean when they say “Facebook application.” Facebook allows developers, companies, and brands to build their own Web sites right on Facebook.com, which includes Facebook’s own headers and footers. When you build a
Part I: Understanding the Basics of a Facebook Application

Facebook-hosted application you are in essence making an extension of Facebook itself. Users of your Facebook-hosted application never have to leave Facebook to use your Web site or application.

- **Hosted on your own Web site.** You can integrate Facebook features right into your own Web site. For instance, you can authenticate users through their Facebook accounts right on your own site. You can get a list of the user’s friends and list them on your site. You can pull almost any type of data about users and their friends and apply it to your own Web site’s content.

In this book I show you how to implement the two methods, but I also show you some simple ways you can integrate in both environments with very little effort. In most cases, it involves a little HTML knowledge, and the capability to copy and paste. Even marketers should pay attention to this stuff, as you’ll have that one step ahead of the competition.

**Understanding the Facebook Application Development Process**

Anyone can create a Facebook application, and the great thing about it is there is no approval process to get up and going! To develop a good application, you just need to follow the rules Facebook has set up (I cover those in Chapter 15), know how to set up your application (I cover that a little later in this Chapter and in Chapter 2), and then start getting people to use it. For the most part, there are no crazy application processes like Apple App store or Android Marketplace. You create it and it’s live.

Social applications aren’t exactly like typical programs you would typically write. When you work with Social APIs, you have to consider that your code will naturally promote itself. The code you’re writing enables people to share information quickly and effectively. In essence, writing social apps makes you both a programmer and a marketer, because your code is now selling your product. So what things do you need to consider?

- **Expect you’ll need to scale.** Although this is no longer the gold rush of 2007 when simple Facebook applications would go from one to two million users in just two days, there is still a good chance your Web site or application could grow very quickly. All it takes is for someone with hundreds of thousands of fans to talk about your application or Web site and soon you are getting slammed with new users. Use a hosting facility that can handle your scaling quickly. Hosting services like Amazon EC2 and Slicehost or Rackspace or Fibernet all have great services for this. I cover this more later.
Never build on just one network. You’re building in a very volatile environment. Social APIs ebb and flow. They change with the wind. You’ll find entire feature sets get removed during your development process. Policies change. So relying entirely on a single network could end up completely killing your product. It’s a huge risk! I recommend you diversify. Not only that, but come up with ideas that have your own foundation. Build something that works well on its own without social networks, and then use social networks to complement the product you’re building. If you keep that attitude you’ll never have a problem if a social network kills a feature or changes its policy. Stay self-reliant.

After you’ve thought of what you want to build, or how you want to integrate Facebook Platform into your Web site, you can start to build it. Getting started is really easy — here are a few things you should know:

- **You can host anywhere you want.** At a minimum, you need only a place you can render HTML files. Any hosting provider can do this. If you’re doing something that simple, pick something cheap.

- **There is no approval process to launch an application.** Although Facebook does have an application directory, it is nowhere near as prominent as it used to be. Any application can still launch without being included in the Facebook application directory.

- **You can write in any language you like.** There are libraries for just about every language out there. Pick your favorite, then go. PHP is always good because that’s what Facebook develops internally, but you can really use anything! I share a few of the various libraries later in this chapter.

- **To get going immediately, just go to** [http://facebook.com/developers](http://facebook.com/developers). That’s it! You can go through the steps in Chapter 2 to learn details of what you need to enter, and what you need to do, but this will at least get you started. The docs are all at [http://developers.facebook.com](http://developers.facebook.com) if you prefer to get the latest and greatest information right away.

---

Thinking socially

As you’re planning your application or Web site, think “social.” I’m not just talking about the capability to allow your users to chat with each other, either. A great social application builds relationships between users. You should look over your Web site and application and consider how you can use social technologies and APIs to build relationships between your users and customers.

Are you organizing your data categorically, or by the content your users’ friends are liking and sharing? Are you giving them easy ways to share your data with their close friends and family? Do your layout and application facilitate easier sharing between individuals? All these will help make a successful use of Facebook API within your application or Web site. Always ask yourself this, and you’ll come up with a great application.
A basic Facebook application hosted on Facebook will be hosted on your own servers somewhere, and will get rendered through an iFrame in your browser. On your Web site, you either include Social Plugins, simple pieces of HTML you can copy and paste into your Web site, or you can make simple requests to Facebook to get and render information about your users that are also on Facebook. I cover all that in detail later.

Picking a Platform That Works Right for You

You have two ways of developing on Facebook. Each of these ways makes for a different platform of development. The term Facebook Platform in general just means accessing Facebook’s APIs to access these two methods, but as you’re developing, you’ll first want to decide within Facebook Platform which path you want to take.

What you build on Facebook all depends on your strategy, your goals, and the time frame available to you. Over the course of building your application, there’s a good chance you’ll choose a number of different strategies. I share just a few strategies that I think you should consider as you’re planning your application. In this section, I dive a little deeper into what you can do with these different Facebook Platforms.

Fishing where the fish are

A generally accepted principle in marketing is what marketers call “fishing where the fish are.” The idea is that as a marketer (or developer trying to market your application), in essence, you’re fishing for those potential customers and users you want to bring into your application or Web site. It really makes no sense to go fishing and just wait for the fish to come to you. The most effective means for you to find new customers and users is to go out where they are and “fish where the fish are.”

In the early days of the Web, this was difficult to do. Marketers and Web site builders were stuck building Web sites and having to come up with their own means to advertise and bring users and customers to their brands. They would try to “fish where the fish are” by placing their Web site name on their products and alongside their TV commercials, but all that was doing was letting people know where they were. There was no way to get into the conversations of people and to enable those conversations to happen.
Along came social media and social networks like Myspace, Ning, and Facebook. Companies could create Myspace profiles and Facebook Pages and have a presence right on the networks and places on the Web where people were talking with each other. Instead of people coming to your Web site, you were able to bring your Web site to the people, reducing the steps it would normally take for an individual to find your brand.

Soon, social networks were providing as many ways as possible to enable your brand to get in front of its users and to get into the conversations of those people passionate about your brand. Facebook was one of the first, and in 2007, it opened its massive database of close friends and families for brands, companies, and Web sites to build things right on top of Facebook.com, where the users were already participating.

This global movement toward Facebook.com for brands was compared to a “gold rush” of sorts, with applications seeing millions of users in just days, and it is something you can still embrace today. Here are some simple ways you can “fish where the fish are” on Facebook:

- **Facebook Pages:** Consider a Facebook Page (that’s with a Capital “P,” not to be confused with a user’s personal profile) your brand’s profile on Facebook. Because profiles must be occupied by real people, Facebook has provided a place, Pages, for brands to build community and discussion around their brand.

  With a Facebook Page, people interested and passionate about your brand are able to “like” the Page by clicking the Like button on the left side (or sometimes top) of the page (see Figure 1-1). When they like your Page, your Page appears in their list of interests on their profile and the act of liking appears on that user’s wall in his or her profile. Other people have the opportunity to see that and like your Page, and it naturally promotes itself.

  In addition, as a Page administrator, you can post updates to your Facebook Page. Each update you post appears in the news feed of users who have liked your Page. They have the opportunity to comment on and like these posts, and their friends can see them and might also want to comment and like those posts. You have just enabled something for them to talk about, and they are now talking about your brand!

- **Facebook applications:** Perhaps you have seen your friends throwing sheep, or maybe they just planted a new vegetable in their virtual garden. Each of these actions is happening as a result of some “application” providing a means for their users to share their activities on Facebook.

  You can build your own applications on Facebook. In fact, that’s one of the first things I cover in Chapter 2. As a business, developer, or brand,
you can build out your own version of your Web site right on Facebook.com, and users never have to leave the site.

In addition, you’re given “integration points” (covered in Chapter 4), where you can interrupt the process of users who are using your application to enable custom publishers (that’s the box where you enter your status updates), custom tabs, automated posts to the news feed, and more. It’s good you understand what you can do with an application on Facebook — it enables you to truly dive down and actually swim with the fish.

✓ **“Share to Facebook” links:** At a minimum, you should have some means on your own Web site to allow your users to share with their friends on Facebook.com (see how Mashable does this in Figure 1-2). You can use a site called ShareThis.com to provide simple share links, or you can use simple social plugins that Facebook provides to enable a “Like button,” where users can like things on your site and share those with their friends on Facebook.
Facebook tabs (or custom links): Facebook tabs (now custom links, as Facebook has removed what were previously tabbed links at the top of the page in exchange for a list of links on the left, below your Page's profile picture) give you a customized theme for your own Facebook Page by allowing you to create your own customized experiences within the Page. This can be a promotion, encouraging users to like the Page in exchange for something free, or maybe just a welcome message, greeting users to the Page. I cover this more in Chapter 4.

Facebook advertising: This is a powerful one. Go, right now, to http://facebook.com/ads and set up a simple ad (see Figure 1-3). You don’t have to publish it, but at least familiarize yourself with the process.

You’ll notice immediately that Facebook gives you very granular controls over who sees that ad. You can type in specific interests, genders, age ranges, and locales, and Facebook tells you the exact number of people who will see that ad. You can then set a budget regarding how much you want to spend on that ad per day.

Or, you can target the friends of all the fans of your Facebook Page and make it an even more personalized and familiar experience. Facebook ads are a very powerful tool in your arsenal to “fish where the fish are” if used right.
From fishers to farmers

I talked about Facebook enabling brands to go out and be where their users and customers are using applications, advertising, and Pages. Fishing is critical to the success of your company if you want to stay ahead.

Throughout history, civilizations of fishers and hunters always turned to farming eventually as the more effective means of bringing food to the table. In fact, even today, the most thriving nations in the world have all learned to farm. Farming is, in essence, the process of confining the prey to your own turf such that the prey can be easily harvested at the least cost.

Thus it is with brands and companies. You too will be more effective when you can bring the “hunt” to your own turf and go from “fishing” to the farm. As a brand, you should know how to farm, and Facebook makes this possible as well (and no, I’m not talking about FarmVille!).

Facebook provides several means to allow you to bring customers to your own turf through simple HTML and coding on your part. You can take any of
these methods and apply them to your own Web site, in essence cultivating your own “farm” in the process. Here are some of the tools Facebook provides to cultivate your farm:

**Facebook Graph API:** Facebook has provided a simple application programming interface (API) that, with a little JavaScript knowledge (at a minimum), you can access on your Web site and bring users’ friends, likes, comments, and news streams right to your own Web site. Facebook Graph API is for those who want to have the full flexibility of customizing the Facebook experience right on their own turf. I cover this in more detail in Chapter 7.

**Social Plugins:** If JavaScript is a little too complex for your expertise or needs, you should definitely consider Social Plugins. Social Plugins are basically just pieces of HTML that you can place on your own Web site and immediately include functionality such as Like buttons, widgets that include streams of the latest activity (likes, shares to Facebook, for instance) from Facebook users who visit your site, as well as widgets that display the most popular pages on your site based on the number of Facebook users who have liked and shared those pages. Facebook provides a simple tool, at http://developers.facebook.com/plugins, that you can preview your plugins and copy and paste code right into your Web site. With Social Plugins, all you need to know is how to copy and paste, and you’ll be farming with the pros!

**Open Graph Protocol:** Did you know that your own Web site could be a Facebook Page of its own? That’s right — by adding some simple meta tags, called Open Graph Protocol, and a Social Plugin or two, you can set your site so that when someone likes your Web site (using the Social Plugins I mention previously), Facebook recognizes that Web site as a Page on its network. You can now post updates to those who liked your Web site and track analytics surrounding Facebook users who visit your site, and your site will appear in users’ Facebook search results and in their interests for their profile. Open Graph Protocol is how you build the foundations for your farm using Facebook. I cover Social Plugins and Open Graph Protocol in Chapter 6.

**Facebook Credits:** At the time of this writing, these are only in very limited beta. However, in the future, these will (sometimes literally!) be the currency of your farm. Facebook is enabling a credits-type system, where, with simply your Facebook credentials, you can purchase anything on the Web with a prefilled credits account you will have set up. Facebook Credits are powerful for you as a farmer, because they enable a very simple and convenient way for customers to purchase your products with as few steps as possible, in a manner that is already familiar to them. I talk about Credits more in Chapter 15 — you’ll want to read that one.
Part I: Understanding the Basics of a Facebook Application

Releasing your application to the building block Web

After you create your farm, you can take it one step further and allow other people to access your farm. You do this by enabling APIs and pieces of data you release to other developers to use on their own Web sites (or farms). I call this “the building block Web.”

In the early days of the Web (back in those days, they versioned it Web 1.0), Web site owners built their own sites. These sites were built entirely for readers to find, come back frequently, and read or purchase items from. You couldn’t extract the data from those sites, nor could you know when new data existed. All users had was a browser, something like Netscape Navigator or Internet Explorer, which they used to view the Web. These browsers were users’ only peek into those Web sites. In those days, the browser was “the platform” and was what Web developers would use to organize and share their data with users.

Shortly afterward, developers began to build simple access points into their data. Really Simple Syndication (RSS) came about, enabling programs to know when new data existed for a Web site and enabling readers to easily recognize and parse that data. XML-RPC, SOAP, and REST all came about, giving developers even more access to data from those Web sites. Very soon, Web 2.0 emerged, making the Web itself “the platform.”

I see a new platform emerging with technologies such as Facebook. This one focuses on the very platforms themselves. Now every Web site is expected to provide its own API and its own access points. Each Web site is providing its own little component that developers can take from and build their own much larger products as a whole. Although Facebook may provide a user’s friends and family connections for an application, Google may provide search for that application. At the same time, Flickr may provide the photos, and YouTube the videos. Each Web site has its own piece of the puzzle to contribute.

You should be thinking of how you can contribute as you consider your application. What core strength do you offer? Your application should be simply another building block in the new Web, and you should make that as accessible as possible to others so that they can incorporate it into their own farms on the Web.
Understanding How Your Application Will Access Facebook

Now that you understand the general philosophies and paradigms of Facebook development and hopefully have a good idea of where you want to start and what your strategy is, it’s time to start understanding the depths of the Facebook application environment. Depending on whether you’re fishing or farming, your application will access Facebook’s APIs entirely differently, and your users will interact with your application in a different way. I start with how an application on the Facebook.com environment is set up.

The Facebook.com environment

You certainly want to read Chapters 2 and 3 to get more details on this, but I give you a summary here. When you host an application on Facebook, you are not really “hosting” it, per se, but instead you are using Facebook as a proxy between you and Facebook users.

A typical Facebook application works like this:

2. Facebook makes a call to your servers (through an iFrame HTML tag).
3. Your servers look at what was called and format data accordingly. During this time, your servers may also make calls back to Facebook’s API to retrieve additional information (such as friends, profile information, and so on) before returning that data to the user.
4. Your server then returns the formatted data to Facebook.com in an iFrame (note that sometimes this can just be a redirect message sending the user to authenticate or authorize your application).
5. Facebook.com parses (reads) that data and formats it further by adding the Facebook header. Read more about this in Chapter 2 and even more thoroughly in Chapter 4.
6. Facebook returns the entire formatted page to the user.

See Figure 1-4 to further understand the flow of a typical application in the Facebook.com environment.
Accessing Facebook from your own Web site

The architecture of an application built on your own Web site will be a little different from that of a Facebook.com-wrapped application. The difference is that instead of Facebook.com proxying and controlling your application, your own Web site will have 100 percent control over the rendering of your application. The user might possibly never see the Facebook.com header and footer, and you can basically render the data however you like. Here’s how the flow of an application built on your own Web site works:

1. The user visits your Web site.
2. Your Web site renders simple JavaScript and HTML-like tags (called XFBML) as part of your normal HTML back to the user.
3. The user’s browser runs the JavaScript, which makes calls back to Facebook (note that this can also happen back on your server before rendering the HTML back to the user). Facebook returns data such as authentication information, user profile information, information about the user’s friends, and more.
4. The browser renders that data back to the user in a dynamic, personalized manner that includes his or her Facebook profile information and friends.

Figure 1-5 shows the flow of a Facebook application rendered through your own Web site.
Selecting a Development Language

Now it’s time to figure out what you’re going to build this thing with! If you’re already a developer, you probably have a preferred language you like to write in. If you’re a marketer or a business owner, you probably don’t really care to do much development, but you’re reading this book to hopefully get your hands at least a little bit dirty. That’s okay. Facebook has solutions for both types.

Facebook either provides, or has developers who have provided libraries for just about every programming language. I cover a few of those in the following sections.

The Facebook JavaScript SDK

The Facebook JavaScript SDK is Facebook’s default and easiest-to-use library, and it’s the focus of most of the new examples in Facebook’s documentation at the time of this writing. Including the software development kit (SDK) is simply a matter of calling the library and then initializing it using some simple JavaScript code. I show you how to do this in Chapter 3.

You can find out more about the SDK under the developer documentation. At the time of this writing, you could find it here: http://developers.facebook.com/docs/reference/javascript. All information, API method calls, and references can be found on the developer documentation site.

The Facebook JavaScript SDK is open source! Although most of its calls are just making requests to Facebook’s Graph API, you can get access to all the underlying code that makes those calls on its GitHub repository. If you have some coding skills, this might be a great opportunity for you to contribute and give back to the other developers who may be experiencing similar needs as you. This also gives you, the developer, the added advantage of being able to sift through the code and debug on your own if you encounter any difficulty.
The Facebook PHP and Python SDKs

Facebook’s official server-side SDKs are PHP and, even more recently, Python. Facebook has made these libraries as simple as requiring the library code, instantiating a new Facebook object (usually passing in your App ID and API secret), and then simply passing in the path to the Graph API method call that you want to make to the Facebook object you just created.

The Facebook PHP and Python SDKs are also completely open source. You can check out the source code to the PHP SDK on GitHub at http://github.com/facebook/php-sdk. You can check out the source code to the Python SDK on GitHub at http://github.com/facebook/python-sdk. On each GitHub page, you can find valuable readme documents, code updates, Wikis, and more. You can also report issues with your given library and converse with those who wrote it. If you ever experience an issue with the PHP or Python libraries for Facebook, you should report your problems and ask your questions here before going to the Facebook Developer Forum (which I talk about in Chapter 2).

The iPhone and Android SDKs

Facebook isn’t just for the Web! At the time of this writing, Facebook has provided two official mobile SDKs for development on mobile phones. They have provided an iPhone SDK, and more recently, an Android SDK for mobile application developers to integrate into their iPhone and Android applications.

Each mobile SDK is available on GitHub just like the other Facebook SDKs. I cover this in detail in Chapter 13, and walk you through the process. You can also get all the details for this at http://github.com/facebook/facebook-iphone-sdk.

For Android, read http://github.com/facebook/facebook-android-sdk to get all the details. Android involves loading the libraries into Eclipse and adding those into your project. Also see Chapter 13 for more details on Android integration. I walk you through setting up an entire application in Android there.

Other “nonofficial” SDKs

If Python, PHP, or JavaScript doesn’t suit your fancy, someone has probably created an API for your language of choice. Here are a few of the more popular ones, but you can always use Google Search to find more:
✓ **Perl:** WWW::Facebook::API is probably the most thorough Perl SDK at the time of this writing. You can find it by search on CPAN or by going to http://search.cpan.org/~unobe/WWW-Facebook-API-0.4.18/lib/WWW/Facebook/API.pm. The cpan WWW::Facebook::API command should also install it for you via the command-line cpan shell.

✓ **Ruby:** Many Ruby libraries are available for the Facebook API. rfacebook (http://rfacebook.rubyforge.org), the OpenGraph Ruby library (http://github.com/intridea/opengraph), and the MiniFB Ruby library (http://github.com/appoxy/mini_fb) all came up in my search on Ruby Facebook libraries when I wrote this.


✓ **.Net:** Although you find many forks and even some smaller initiatives in the .Net world, the two main libraries for .Net are Facebook.NET (www.nikhilk.net/FacebookNET.aspx) and the Facebook Developer Toolkit (www.codeplex.com/FacebookToolkit). Look at each and decide which one works best for your needs. Also, be sure to look at the latest commit dates for each to see how active the community is behind it.

---

**Referring to the Facebook Documentation**

When in doubt, look it up! Facebook has amazing references online to all its API materials. Whether simple API lookups or more in-depth library calls, Facebook provides documentation to take you every step of the way through the development process.

**Perusing Developers.facebook.com for information**

Developers.facebook.com (see Figure 1-6) should be your start for any information you want to find about Facebook APIs, tools, and libraries. Especially considering that Facebook changes this documentation frequently, this is a good place to start, because what may be the documentation today may be in a different location by the time you read this. Search around and get to know this place and what is available to you — and check back frequently.
At the time of this writing, you could find all developer documentation for Facebook APIs, tools, and libraries at http://developers.facebook.com/docs. This may or may not be the correct URL by the time you read this, so be sure to search around if that’s not the case. (I’ll try to post new locations on my blog as well as my own Facebook page and this book’s Facebook Page if things change. You can find the book’s Facebook Page at http://facebook.com/dummiesbook.)

The Facebook Documentation page on Developers.facebook.com (see Figure 1-7) serves as a single location for you to find all you need to get started and continue going in Facebook development. This is where you can find a reference of every API call, helpful tutorials, how-tos, and wizards to take you through the process of creating, understanding, and continuing to develop your application.
The documentation page consists of a few major sections that cover pretty much the entire Facebook API. Starting with a few getting-started sections, you can then move on to gain a general sense of each component of the Facebook API, such as the following:

**Graph API:** In Facebook's simple, URL-based API into the Facebook social graph, you find out all about how to use it and how it can be powerful to you in learning about the people who visit your Web site or application. See Part IV for a more thorough explanation and walk-through of this.

**Authentication:** Facebook uses OAuth 2.0 to authenticate users. Find out what that means and see how to use it in your application to identify users and incorporate their Facebook profiles into your application. I cover this in Chapter 9.

**Social Plugins:** In Facebook's most simple way of integrating its API into a Web site or application, a simple copy and paste puts simple Like buttons and stream widgets into your application or Web site. Chapter 6 also covers these.
Open Graph Protocol: This is an open standard Facebook set up to identify pages and things on the Web. Simply place a few meta tags and you’re set. This overview covers what that is, why it’s important, and how you can implement it on your Web site. I go into more detail about this in Chapter 6.

After you acclimate yourself, you can then delve a little deeper and use the references to find out more about each API call or tag that you need to place in your HTML or code. You can discover the following:

- **Core APIs:** Get every single Graph API call and Social Plugin available, and get a description of how each can work.
- **Developer SDKs:** From PHP to Python, from the iPhone to Android, discover what you can do and how to use your preferred language and programming environment to write Facebook applications.
- **FQL (Facebook Query Language):** Facebook is gradually making this less necessary with its Graph API. However, at times, you may need to retrieve specific data that Graph API just cannot provide. FQL enables a customized solution to retrieve granular data from Facebook in an SQL-like format. This section of the documentation shows every table available on Facebook and demonstrates how to access data from those tables.
- **XFBML (Facebook Markup Language for Web sites):** With Facebook’s tag language, you can access every single tag, see examples, and know what it can do on your Web site.
- **Old APIs and client libraries:** Facebook hasn’t always supported the same API formats. For deprecated APIs, you can still look up information about those APIs and find out how to use them through this documentation. Sometimes you can find a few things by using these APIs that haven’t yet been migrated to the new formats. It might help reading through these as well.

Documentation is important! Get to know what you can and can’t do with it. If you can’t understand JavaScript or programming concepts, study the XFBML and overview sections. Find out as much as you can about what you can do with the API, and your app will be much richer and much more powerful to you in the long run.

OpenGraphProtocol.org

Facebook created the Open Graph Protocol so that it could easily identify content that users were sharing and liking on the Web (see Chapter 6). However, it wanted to open the standard so that anyone who wanted to duplicate it or use it for his own Web site could do so without worry of copyright or patent infringement, or repercussions from Facebook. So Facebook released it under a special, commonly known license called the Open Web Foundation License and gave it its own Web site independent of Facebook for people to read and study from.
OpenGraphProtocol.org is your source for everything Open Graph Protocol. (See Figure 1-8.) It covers the standard and what it is, along with examples of how to implement it. It’s a basic, one-page site, so it’s good for you to read it and understand it if you can. If it isn’t enough, you can always go back to the Facebook developer documentation and find out more about the Open Graph Protocol there as well.

**GitHub**

Facebook hosts most of its official development SDKs (PHP, Python, JavaScript, iPhone, and Android) on GitHub as open source (see Figure 1-9). GitHub is an open source repository of code that anyone can sign up for and post code for others to use (according to the code’s licensing terms, of course). The great thing about GitHub is that it enables any developer to take a piece of source code and “fork” it into the developer’s own repository on the site. The developer can then safely develop his or her own instance of the code without worrying about the code interfering with the main branch of code it was forked from. Then, if the maintainer of that code likes what the person who forked the code has done, he or she can then reincorporate that code into his or her main branch.

![The Open Graph Protocol](http://opengraphprotocol.org/)
Another major feature of GitHub is that it provides a great place to store documentation surrounding the code that is being stored. Documentation for all the Facebook SDKs is found there, including setup instructions and code examples. I encourage you to check it out and study what is available so that you can discover how to implement your preferred SDK. Maybe if you’re feeling particularly savvy, you can even contribute a little back to the code base and help the overall ecosystem on Facebook thrive.

GitHub’s a little hard to explain in such a short amount of space. If you’re looking for a greater tutorial on how to use GitHub and what steps you need to take to get started, I recommend the great walk-through at http://learn.github.com/p/intro.html. Also, see the github.com links I share in the SDKs section above to get an idea of what it looks like and where some of the Facebook code lies.

Figure 1-9: The Facebook JavaScript client libraries on GitHub.