Extracted from:

Developing Facebook Platform

Applications with Rails

This PDF file contains pages extracted from Developing Facebook Platform, published by the Pragmatic Bookshelf. For more information or to purchase a paperback or PDF copy, please visit http://www.pragprog.com.

Note: This extract contains some colored text (particularly in code listing). This is available only in online versions of the books. The printed versions are black and white. Pagination might vary between the online and printer versions; the content is otherwise identical.

Copyright © 2008The Pragmatic Programmers, LLC.

All rights reserved.

No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form, or by any means, electronic, mechanical, photocopying, recording, or otherwise, without the prior consent of the publisher.

Beta Book

Agile publishing for agile developers

The book you're reading is still under development. As part of our Beta book program, we're releasing this copy well before we normally would. That way you'll be able to get this content a couple of months before it's available in finished form, and we'll get feedback to make the book even better. The idea is that everyone wins!

Be warned. The book has not had a full technical edit, so it will contain errors. It has not been copyedited, so it will be full of typos and other weirdness. And there's been no effort spent doing layout, so you'll find bad page breaks, over-long lines with little black rectangles, incorrect hyphenations, and all the other ugly things that you wouldn't expect to see in a finished book. We can't be held liable if you use this book to try to create a spiffy application and you somehow end up with a strangely shaped farm implement instead. Despite all this, we think you'll enjoy it!

Throughout this process you'll be able to download updated PDFs from your account on http://pragprog.com. When the book is finally ready, you'll get the final version (and subsequent updates) from the same address. In the meantime, we'd appreciate you sending us your feedback on this book at http://books.pragprog.com/titles/mmfacer/errata, or by using the links at the bottom of each page.

Thank you for being part of the Pragmatic community!

► **Andy Hunt**

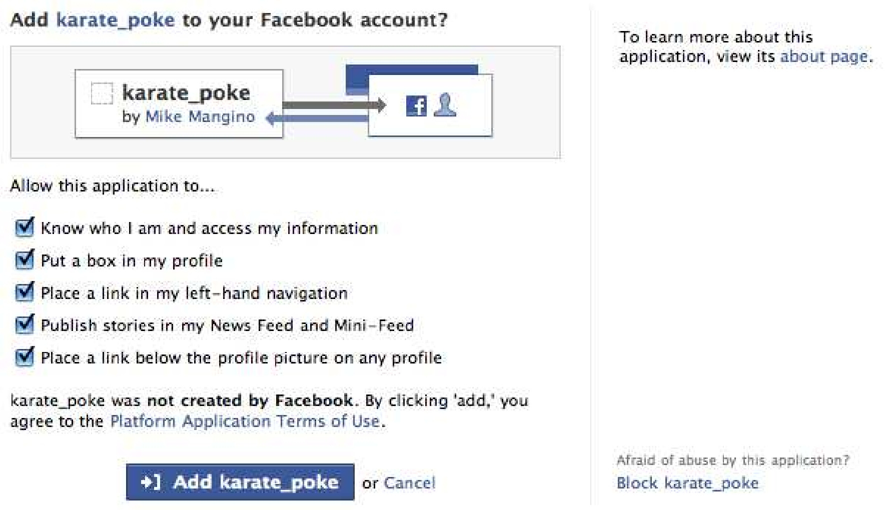
Chapter 2

Getting Started with the Facebook Platform

We've talked a little about Facebook Platform applications and why you would want to build one. Before we dive into coding, let's look at the parts of a Facebook Platform application. To do that, we'll walk through Karate Poke, the application we'll be building throughout this book. Karate Poke is a simple game where you battle your friends and other users. As you progress through the game, you'll earn karate belts and learn new moves.

While Karate Poke may not be identical to the Facebook applications you'll want to build, it serves as a good introduction. It's a small application, so we'll be able to build the whole thing as we go. It also uses almost the entire Facebook API. We'll see how to build invitations and notifications. We'll use most of the UI elements that Facebook provides. We'll even be able to use some advanced performance tuning techniques to make it handle the demands of millions of users.

In this chapter, we'll start out by taking a tour of Karate Poke. After we've seen the basics of Karate Poke, we'll create an application using the Facebook Developer tool. Next, you'll set up your computer to run a pre-built test application. We'll finish up by creating a few test users. By the time we've finished this chapter, we'll be ready to write some code. We've got a lot to do, so let's get started.



Add karate_poke to your Facebook account?

karate_poke
by Mike Mangino

Allow this application to...

☑ Know who I am and access my information

☑ Put a box in my profile

☑ Place a link in my left-hand navigation

☑ Publish stories in my News Feed and Mini-Feed

☑ Place a link below the profile picture on any profile

karate_poke was **not created by Facebook**. By clicking 'add,' you agree to the Platform Application Terms of Use.

To learn more about this application, view its about page.

Afraid of abuse by this application?
Block karate_poke

Figure 2.1: Facebook asks you to install an application before you use it.

2.1 Adding the Karate Poke Application

Open your browser and go to the main url[1] for Karate Poke. You're seeing the application installation page shown in Figure 2.1. You must install an application before it can get access to any of your information. This is the first of several privacy features that are part of the Facebook Platform.

Installing a Facebook application isn't like installing other software. There is nothing for you to download and nothing for you to run. Instead, you are just required to decide what information you are willing to share with the application. During application installation, you can choose to limit what the application can access. By default, an application can:

- Access public information about you
- Display information in your profile
- Place a link in your left-hand navigation
- Publish stories in your News Feed and Mini-Feed

[1] http://apps.facebook.com/karate_poke

Click Here to purchase this book now.

- Place a link below the profile picture on any profile

These permissions aren't set in stone. You can change these permissions and even uninstall an application at any time. As an applicatio developer, you'll need to be aware of this. It's important to make sure it degrades nicely when a user limits its capabilities.

2.2 The Parts of a Facebook Application

Now click "Add Karate Poke" to install the application. Once you've done that, you will be taken to the *canvas page* of the application. That seems like a good place to start our tour.

The Facebook Canvas

The canvas page is the main place where your users interact with your application. You'll notice that it looks like any other Facebook page, including the navigation areas along the top and the left sides. Facebook actually inserts your application right into the middle of the page, just like a Rails layout.

We'll cover the Facebook canvas in detail in Chapter 5, *Getting Into the Facebook Canvas*, on page 66. For the purpose of this demo, you should attack one of your friends. You can select any move from the pull down menu. Once you've picked a move, start typing one of your friend's names into the text field. Isn't that typeahead cool? That's just one line of code!

Select any of your friends and then click Attack. Karate Poke will let them know that you've attacked them. You should be taken to your battle history page, where you see the result of your attack. Your battle record will be added to your profile.

The Profile Page

Click on the profile link in the top navigation. At the bottom of the page you should see a box from Karate Poke. Every Facebook application you install gets access to a little box in your profile. The profile area is a place to show off information about its owner. For instance, my Karate Poke profile box shows my battle history. My Growing Gift profile box shows gifts that my friends have sent me.

Facebook doesn't limit the amount of information an application can write to a user's profile, but it does limit the content. Facebook allows

Click Here to purchase this book now.

advertising in the canvas, but not the profile. Along with just displaying information, an application can also add actions to its users' profiles. While you're on your profile page, scroll up and look at the links under your picture. You should see an "View your battles" link. Profile actions are a way that Facebook encourages interaction between users.

We'll look at the profile area in more detail in Section 7.3, *The Facebook Profile In Depth*, on page 113. Let's move on to the last major part of a Facebook application.

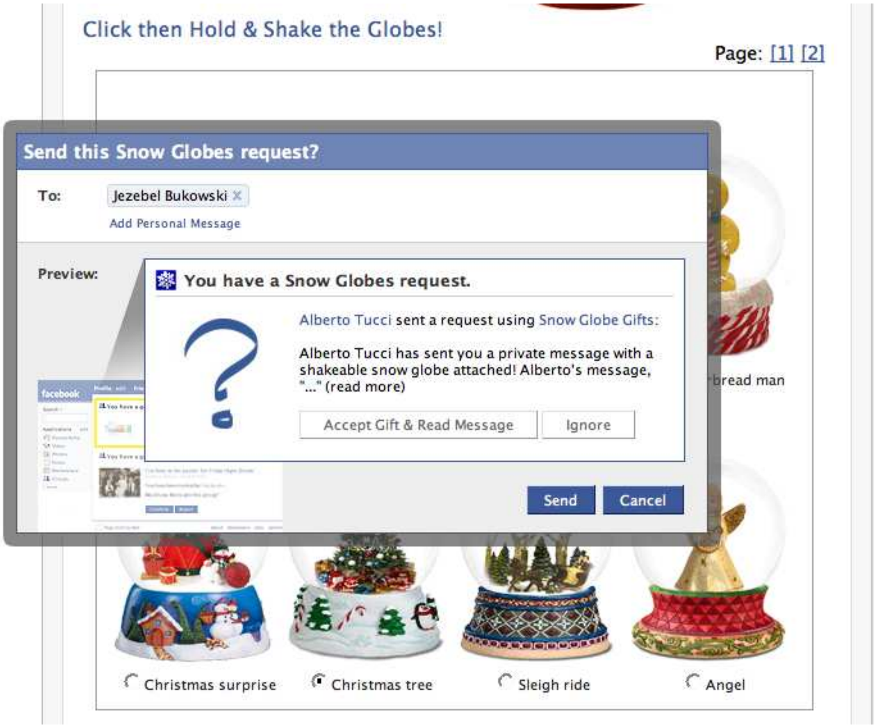
Messages

Along with giving you access to the canvas and profiles, the Facebook Platform also gives you several different types of messages. Your application can send *requests*, *notifications* and *newsfeed* items. This may seem like a lot of different message types, but they all fill very different needs. If you're a Facebook veteran, you've probably used all of these message types without even noticing.

A request is exactly what its name implies, a request to perform an action. When somebody asks to be your friend, that's a request. A Request is just a message with buttons for performing actions. Your application can send requests on behalf of a user, but only to their friends. Users must approve requests before they are sent. You can see the approval process in Figure 2.2, on the next page.

Notifications are similar to Requests but lack the call to action. Your application can send notifications on behalf of the user without requiring their interaction. Unlike a request which can only be sent to a user's friends, notifications can be sent to any user of your application. Notifications also do not require an explicit approval from the sending user. Facebook limits the number of notifications that an application can send for a user in a given day. Users can choose to block notifications from certain applications, and can even mark a notification as spam. Be careful sending notifications. If enough users mark your notifications as spam you'll lose the ability to send notifications for 30 days.

Newsfeed items are the last form of communication available to the application developer. Newsfeed items are what populate your mini feed. They tell your friends a little bit about what you've been doing. Your application can send quite a large number of newsfeed items each day but they aren't guaranteed to be visible. Facebook tries to decide what is interesting for each user and only shows a subset of the available items.



Click then Hold & Shake the Globes!

Page: [1] [2]

Send this Snow Globes request?

To: Jezebel Bukowski

Preview:

You have a Snow Globes request.

Alberto Tucci sent a request using Snow Globe Gifts:

Alberto Tucci has sent you a private message with a shakeable snow globe attached! Alberto's message, "..." (read more)

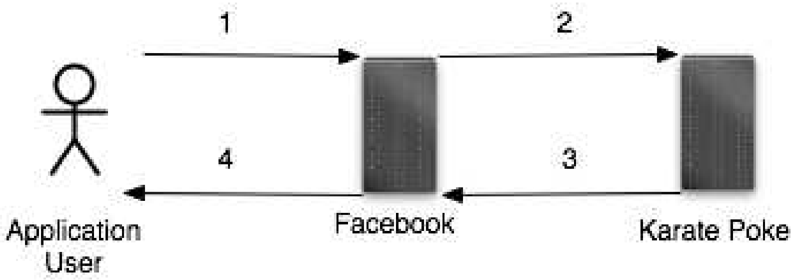
Figure 2.2: Facebook makes you approve a request before it is sent.

If you've used Facebook for a while, you may wonder why I haven't mentioned Facebook messages. Facebook doesn't provide an API for sending a message on behalf of a user. There are ways to help users send messages, but they don't have the nice integration that these other options provide.

2.3 Getting Inside the App

Let's dig in a little deeper to understand how your application actually becomes part of Facebook. I'm going to start with the canvas since it will probably be the largest portion of your application. Building the canvas portion your Facebook application is very similar to any other website. Your web server will receive requests and send back pages. Of course, it's a little different since your page ends up inside a Facebook page.

CLICK HERE to purchase this book now.



1) A user requests a page from Facebook. 2) Facebook sends the request to your server. 3) You send a response to Facebook. 4) Facebook processes your response and puts your page into theirs and sends it back to the user.

Figure 2.3: The path of a Facebook HTTP Request

When a user requests a page from Facebook, they send the request on to you. Your application responds with your content. Facebook then puts your page into the middle of theirs and sends it back to the browser. You can see a picture of this in Figure 2.3

This all happens seamlessly from your users' perspective. In fact, it's pretty seamless from your standpoint as well. For the most part, you can forget about Facebook standing between you and the user. Sure, it's not exactly like regular development, but it's pretty close.

This may seem like a complex architecture, but Facebook has a good reason for the complexity. By acting as a middleman, Facebook can process the page you send back to the user. This allows them to insert your page into their layout, and also to provide you with some really powerful tools. We'll look at this in detail in Chapter 5, *Getting Into the Facebook Canvas*, on page 66.

Things are different for the Profile. I have ten applications installed on my profile page. If each application took even a second to respond, it would take ten seconds to load my profile! To prevent long profile load times, Facebook caches profile information. You can update it at any time by just sending them new content to display. This can feel a little painful at first, but you'll be happy to avoid the millions of page views

CLICK HERE to purchase this book now.

The Pragmatic Bookshelf

The Pragmatic Bookshelf features books written by developers for developers. The titles continue the well-known Pragmatic Programmer style, and continue to garner awards and rave reviews. As development gets more and more difficult, the Pragmatic Programmers will be there with more titles and products to help you stay on top of your game.

Visit Us Online

Developing Facebook Platform Applications With Rails' Home Page
http://pragprog.com/titles/mmfacer
Source code from this book, errata, and other resources. Come give us feedback, too!

Register for Updates
http://pragprog.com/updates
Be notified when updates and new books become available.

Join the Community
http://pragprog.com/community
Read our weblogs, join our online discussions, participate in our mailing list, interact with our wiki, and benefit from the experience of other Pragmatic Programmers.

New and Noteworthy
http://pragprog.com/news
Check out the latest pragmatic developments in the news.

Buy the Book

If you liked this PDF, perhaps you'd like to have a paper copy of the book. It's available for purchase at our store: pragprog.com/titles/mmfacer.

Contact Us

Phone Orders:	1-800-699-PROG (+1 919 847 3884)
Online Orders:	www.pragprog.com/catalog
Customer Service:	`orders@pragprog.com`
Non-English Versions:	`translations@pragprog.com`
Pragmatic Teaching:	`academic@pragprog.com`
Author Proposals:	`proposals@pragprog.com`