sitepoint®

FANCY FORM DESIGN

BY JINA BOLTON
TIM CONNELL
DEREK FEATHERSTONE

CREATE SENSATIONAL WEB FORMS THAT SPARKLE

Fancy Form Design

by Jina Bolton, Tim Connell, and Derek Featherstone

Copyright © 2009 SitePoint Pty. Ltd.

Technical Editor: Raena Jackson Armitage
Editor: Kelly Steele
Managing Editor: Chris Wyness
Chief Technical Officer: Kevin Yank
Indexer: Fred Brown
Cover Design: Alex Walker
Reviewer: Avi Miller

Printing History:

First Edition: October 2009

Notice of Rights

All rights reserved. No part of this book may be reproduced, stored in a retrieval system or transmitted in any form or by any means, without the prior written permission of the publisher, except in the case of brief quotations embodied in critical articles or reviews.

Notice of Liability

The author and publisher have made every effort to ensure the accuracy of the information herein. However, the information contained in this book is sold without warranty, either express or implied. Neither the authors and SitePoint Pty. Ltd., nor its dealers or distributors, will be held liable for any damages caused either directly or indirectly by the instructions contained in this book, or by the software or hardware products described herein.

Trademark Notice

Rather than indicating every occurrence of a trademarked name as such, this book uses the names only in an editorial fashion and to the benefit of the trademark owner with no intention of infringement of the trademark.

Published by SitePoint Pty. Ltd.

48 Cambridge Street
Collingwood, Victoria, Australia 3066
Web: www.sitepoint.com
Email: business@sitepoint.com

ISBN 978-0-9805768-4-9
Printed and bound in Canada

About the Authors

Jina Bolton is a designer at Crush + Lovely in San Francisco and speaks at conferences around the world. She co-authored *The Art and Science of CSS* (Melbourne: SitePoint, 2007) and has also written articles for web industry-related publications. Jina digs sushi and robots—so you'll find her at http://sushiandrobots.com.

Tim Connell lives and breathes the Web and can often be found atop a soapbox talking about users, accessibility, and sexy techniques. Tim is a technical consultant for Squiz (http://squiz.net), a leading enterprise content management company, and a developer with Fresh Interface (http://freshinterface.net). Outside of the Web, Tim likes to enjoy good company at the local watering hole. Tim lives in Sydney, Australia.

Engaging, surprising, and inspiring, Derek Featherstone is an internationally known authority on accessibility and web development. As founder of Further Ahead (http://furtherahead.com), he has been a user experience and accessibility consultant in demand since 1999, and regularly wows audiences with practical demonstrations of creative solutions to accessibility issues. He is the Group Lead of the influential Web Standards Project and blogs at http://boxofchocolates.ca.

About the Technical Editor

Raena Jackson Armitage made her way to SitePoint via a circuitous route involving web development, training, and speaking. A lifelong Mac fangirl, she's written for *The Mac Observer* and *About This Particular Macintosh*. Raena loves knitting, reading, and riding her bike around Melbourne in search of the perfect all-day breakfast. Raena's personal web site is at http://raena.net.

About the Chief Technical Officer

As Chief Technical Officer for SitePoint, Kevin Yank oversees all of its technical publications—books, articles, newsletters, and blogs. He has written over 50 articles for SitePoint, but is best known for his book, *Build Your Own Database Driven Website Using PHP & MySQL.* Kevin lives in Melbourne, Australia, and enjoys performing improvised comedy theatre and flying light aircraft.

Thank You For Downloading This Book

Thank you for downloading the sample chapters of *Fancy Form Design*, by Jina Bolton, Tim Connell, and Derek Featherstone, published by SitePoint.

Forms: boring, right? Not any more! Fancy Form Design is a complete, step-by-step guide to creating beautiful web forms that are aesthetically pleasing, highly functional, and compatible across all major browsers. All aspects of stunning form creation are covered in this book—from layout and composition through to best-practice CSS and HTML—topped off with a touch of JavaScript sparkle!

This excerpt includes:

- a summary of contents
- information about the authors, editors, and SitePoint
- the Table of Contents
- the Preface
- Chapter 1 from the book
- the Index

When you're ready to buy, visit https://sitepoint.com/bookstore/go/169/e23030.

What's In This Excerpt?

Preface

Table of Contents

Chapter 1: Planning
Your form design journey begins with planning and research.

Chapter 2: Designing
Dive into your form's design: grids, layouts, type, and more.

Index

What's In the Rest of the Book?

Chapter 3: Structure
Careful attention to correct markup ensures usable and accessible forms.

Chapter 4: Styling
Styling with CSS doesn't have to be as hard as it seems!

Chapter 5: Enhancing
Add functional enhancements and aesthetic sparkle with JavaScript.

About SitePoint

SitePoint specializes in publishing fun, practical, and easy-to-understand content for web professionals. You can visit http://sitepoint.com to access our books, newsletters, articles, and community forums.

To God, first and foremost. To Jason, Dad, and the rest of my family. To my amazing friends both online and in "real life." To my co-workers at Crush + Lovely who have become great friends. I love you all so much.

—Jina

Thanks to all the top people in my life: there are many of you, and you know who you are.

—Tim

For my wife Kathryn, and our wonderful children Kaitlyn, Kyla, and Kampbell. Without you, none of this would matter.

—Derek

Table of Contents

Preface xi

Who Should Read This Book? xi
What's Covered in This Book? xii
The SitePoint Forums xiii
This Book's Web Site xiii
The SitePoint Newsletters xiii
The SitePoint Podcast xiv
Your Feedback xiv
Conventions Used in This Book xv
Acknowledgements xviii

Chapter 1 Planning 1

The Elements of Forms 2
Research and Finding Inspiration 12
Interaction Design 16

Chapter 2 Designing 27

Grid and Typography 27
Color 31
Imagery 33
Summary 41

Chapter 3 Structure 43

The Structure of a Form 44
Instructions, Errors, and Advisory Text 55

The Final Markup 58
Conclusion 74

Chapter 4 Styling 75
Stuff to Consider 76
Preparing Your Canvas 77
Reset Styles 78
Fieldsets and Legends 81
Form Questions 83
Putting It All Together 96

Chapter 5 Enhancing 119
Fancier Form Widgets 120
Back to Our Forms 132
Conclusion 146

Index 147

Preface

Think about all the web sites you use each day. Perhaps you have a blog, or at least like to comment on blogs. You might use social networks like Facebook to keep in touch with friends, or a web-based email service like Yahoo or Google. Perhaps you have some favorite forums you like to visit, or a photo-sharing site where you can upload your happy snaps. Maybe you bought a product online recently, or reviewed one. What do all these activities have in common? Forms!

Every day, people use forms for all kinds of activities—they're some of the most interactive parts of any site. Effective, beautiful forms make for happy visitors who find it easy to interact with your site and come back for a second helping. Poorly designed forms annoy and frustrate users, and might even merit a quick trip to the browser's **Back** button.

Yet, despite their obvious importance and ubiquity, many web developers find the task of creating forms to be boring, difficult, frustrating, or time-consuming. The truth is that the secret to creating beautiful, user-friendly, engaging forms—and having a good time while you're at it—is no secret at all. Like everything else, it's easy once you know how.

This book is full of tips, techniques, and practical examples to help you build breathtaking, beautiful forms. We'll guide you through the whole process, from the initial research and planning stages, all the way through to designing, building, and enhancing your form.

By the time you reach the end of this book, you'll be able to jump into your next forms project with confidence. What's more, your forms will be more than just good—they'll be downright *fancy*.

Who Should Read This Book?

Perhaps you're an experienced web developer who's already built some forms, but found the results were less than ideal. Perhaps you're just starting out on your web development career. You might be an interface designer, a HTML and CSS coder, a JavaScript guru, or a bit of all three. If you're involved with any part of the form creation process, this book is for you.

What's Covered in This Book?

Chapter One: Planning

A great form must start with a great plan. We'll discuss some of the gadgets, widgets, and goodies that are available to you. Then, we'll examine how thorough research creates a strong foundation for your forms: use cases and scenarios, paper prototyping, visitor profiles, and more all come together to help you build a solid picture of what your forms ought to be like.

Chapter Two: Designing

Naturally, a beautiful visual design for your forms will wow your visitors. But a truly effective design is one that enhances usability as well as aesthetics. In this chapter, we examine how a grid system, color, typography, icons, and textures come together to create a pleasing and easy to use design for your forms.

Chapter Three: Structure

A strong skeleton for your form is essential, and that skeleton is made of semantic, structural HTML. We look into best-practice methods you can use today in your sites that emphasize usability, accessibility, and meaningful code.

Chapter Four: Styling

Veteran form developers can tell many a tale of browser bugs, strange inconsistencies, and irritating behavior. In this chapter we examine some proven CSS tricks and techniques for form layout that will mesh perfectly with our clean, semantic HTML, and match your form's design perfectly to the rest of your site.

Chapter Five: Enhancing

Careful use of JavaScript can make the difference between "Gee, that's a nice form," and "Wow, that's actually fun to use!" The techniques we describe in this chapter cover functional enhancements such as client-side validation and password strength testing, as well as aesthetic changes like customized pull-down menus and checkboxes.

The SitePoint Forums

The SitePoint Forums[1] are discussion forums where you can ask questions about anything related to web design, development, hosting, and online marketing. You may, of course, answer questions, too. That's how a discussion forum site works—some people ask, some people answer—and most people do a bit of both. Sharing your knowledge benefits others and strengthens the community. A lot of interesting and experienced web designers and developers hang out there. It's a good way to learn new stuff, have questions answered in a hurry, and just have fun.

The Design Your Site forum has subforums devoted to every facet of web design—HTML and CSS advice, graphics tips, accessibility and usability, and even site reviews and critiques.[2] It's free to sign up, and it takes just a few minutes.

This Book's Web Site

No book is perfect, and we expect that watchful readers will be able to spot at least one or two mistakes before the end of this one. The Errata page on the book's web site will always have the latest information about known typographical errors and updates. You'll find the book's web site at http://www.sitepoint.com/books/forms1/. If you find a problem, you'll also be able to report it here.

The SitePoint Newsletters

In addition to books like this one, SitePoint publishes free email newsletters, such as the *SitePoint Design View*, the *SitePoint Tribune*, and the *SitePoint Tech Times*, to name a few. In them, you'll read about the latest news, product releases, trends, tips, and techniques for all aspects of web development. Sign up to one or more SitePoint newsletters at http://www.sitepoint.com/newsletter/.

[1] http://www.sitepoint.com/forums/

[2] http://www.sitepoint.com/forums/forumdisplay.php?f=40

Turn those ugly forms into artistic masterpieces. Order the book today!

The SitePoint Podcast

Join the SitePoint Podcast team for news, interviews, opinion, and fresh thinking for web developers and designers. They discuss the latest web industry topics, present guest speakers, and interview some of the best minds in the industry. You can catch up on the latest and previous podcasts at http://www.sitepoint.com/podcast/ or subscribe via iTunes.

Your Feedback

If you're unable to find an answer through the forums, or if you wish to contact us for any other reason, the best place to write to is `books@sitepoint.com`. We have a well-staffed email support system set up to track your inquiries, and if our support team members are unable to answer your question, they'll send it straight to us. Suggestions for improvements, as well as notices of any mistakes you may find, are especially welcome.

Conventions Used in This Book

You'll notice that we've used certain typographic and layout styles throughout this book to signify different types of information. Look out for the following items:

Tips, Notes, and Warnings

Hey, You!

Tips will give you helpful little pointers.

Ahem, Excuse Me ...

Notes are useful asides that are related—but not critical—to the topic at hand. Think of them as extra tidbits of information.

Make Sure You Always ...

... pay attention to these important points.

Watch Out!

Warnings will highlight any gotchas that are likely to trip you up along the way.

Turn those ugly forms into artistic masterpieces. Order the book today!

Markup Samples

Any code—HTML, CSS, or JavaScript—will be displayed using a fixed-width font like so:

```
<h1>A perfect summer's day</h1>
<p>It was a lovely day for a walk in the park. The birds
were singing and the kids were all back at school.</p>
```

If the markup forms part of the book's code archive, the name of the file will appear at the top of the program listing, like this:

example.css

```
.footer {
  background-color: #CCC;
  border-top: 1px solid #333;
}
```

If only part of the file is displayed, this is indicated by the word *excerpt*:

example.css *(excerpt)*

```
  border-top: 1px solid #333;
```

If additional code is to be inserted into an existing example, the new code will be displayed in bold:

```
function animate() {
  new_variable = "Hello";
}
```

Also, where existing code is required for context, rather than repeat all the code, a vertical ellipsis will be displayed:

```
function animate() {
  ⋮
  return new_variable;
}
```

Some lines of code are intended to be entered on one line, but we've had to wrap them because of page constraints. A ➥ indicates a line break that exists for formatting purposes only, and should be ignored.

```
URL.open("http://www.sitepoint.com/blogs/2007/05/28/user-style-she
➥ets-come-of-age/");
```

Turn those ugly forms into artistic masterpieces. Order the book today!

Acknowledgements

Jina Bolton

Thank-you to the people of SitePoint for the opportunity to work on this book—particularly Raena who was patient, gave great feedback, and is also fun to follow on Twitter. Thanks to Derek for his expertise and for being a pleasure to work with. It's been an honor co-authoring a book with you. Thanks to Tim for helping shift this book out the door! Thanks to my co-workers at Crush + Lovely who have been amazing and understanding, and even gave me business hours to work on this book! And, of course, thanks to the rest of all my family and friends who have helped me along the way.

Tim Connell

Thank-you to my wonderful friends and family for all of your support and encouragement—especially to Mum, for being my mum; to Jerome, for making me smile and think at exactly the right moments; to Heather and Corine for being the most brilliant friends and business partners a guy could ask for, and to Squiz and the many inspirational people I work with. Thanks also to my amazing co-authors Derek and Jina; to Raena and the SitePoint crew; to Avi Miller for the feedback, and to everyone else who read, wrote, or pondered over the words in this book.

Derek Featherstone

First and foremost, thank-you to all of my family for accepting my geekiness and my love for the Web and trying to make it a better place. You don't always get what I'm doing, but you DO always let me be me, and for that I'm truly grateful. Thanks to my co-authors Jina and Tim for their hard work on this book, and the SitePoint crew for getting me involved. Finally, huge thanks go to my co-workers Jeff Smith and Joanna Briggs for all their support and assistance with this and everything we do. I'm proud that you're part of the team. Finally, thanks to Carolyn Wood who, in addition to being my friend, inspires me to be a better writer.

Chapter 1

Planning

Forms—maybe you love them, maybe you think they're … well … boring. Either way, forms are an essential part of web design and development. From small login forms to detailed administration panels, forms are one of the most important interactive elements of your web site or application. They're the carriers of data and—if you'll pardon the courier's cliché—should be handled with care.

The way you develop your forms can greatly impact various concerns:

- Usability—are your forms easy to understand and simple to use?
- Accessibility—are your forms available to people who are accessing your site in a non-traditional manner?
- Error prevention—do your forms help ensure that the information you're collecting is free of problems?

Creating good forms can be a complicated process for both designers and developers. There are a lot of factors that go into creating a form—planning, designing, structuring (with markup), styling (with CSS), and enhancing (with JavaScript)—it's a lot of work! But this work is worth it in the end: an error-free, accessible, and user-friendly form is a happy form, leading to happy users.

Throughout this book, we'll be guiding you through a start-to-finish workflow. We'll start with devising our forms, move on to working through markup and styling, and then laying on some JavaScript to enhance and support your forms.

The Elements of Forms

It's important to understand the various elements that make up a form, so that you're aware of what's available to you. Let's first look at some of the basic default elements. Then, we'll take a look at some examples of more advanced features that are possible through progressive enhancement with JavaScript.

Basic Elements

HTML provides for a number of form elements, each designed to accept various types of content. Whether it's booking a flight or ordering groceries online, chances are you can create a functional form with just these basic elements.

Text Fields

Text fields, like the fields in Figure 1.1, allow the user to type in whatever they like. It's possible to specify a maximum length, otherwise any text can be entered.

Figure 1.1. Input fields used for names

Radio Buttons

Sometimes, you need to limit the user's input to a set of predetermined values. **Radio buttons** provide an ideal solution if one choice only must be made by the user. These are often presented as small round buttons beside each option, as seen in Figure 1.2.

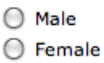

Figure 1.2. Radio buttons based on gender

Checkboxes

Checkboxes are another format for predetermined values, but allow for multiple values to be selected. You could use checkboxes to allow a person to choose several items. In Figure 1.3, we can see a series of checkboxes allowing a user to specify which vehicles they own. Most browsers present checkboxes as squares; selected checkboxes have a tick or cross inside.

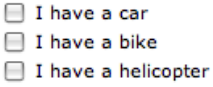

Figure 1.3. Multiple choices with checkboxes

Labels

Labels are essential elements that tell you what the field represents. Most browsers make the space occupied by the label a clickable area, which helps increase the usability of the form; so rather than having to focus on a single, tiny button, a user can aim for the general area of its name. In Figure 1.4, we're clicking the checkboxes we saw in Figure 1.3; clicking the name will select the box.

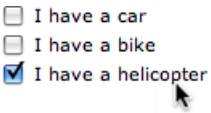

Figure 1.4. Labels are clickable too!

Textareas

Figure 1.5. A textarea as shown in Safari, with resizing handle

Textareas permit the user to type text, just like input fields, but they allow multiple lines of text to be entered. They can even scroll to accommodate content that exceeds the given width and height. In some browsers, such as Apple's Safari, these can even be resized by the user.

Turn those ugly forms into artistic masterpieces. Order the book today!

Select Menus

Select menus are yet another way to display predetermined values; this time, they're arranged as a menu. The benefit to using these over radio buttons or checkboxes is that they save on space, particularly if you have an extensive list of options. However, they're a little harder to use in terms of ergonomics (in that the user needs to hold down the button and drag simultaneously). If you only have a couple of choices, it might be best to just use radio buttons or checkboxes.

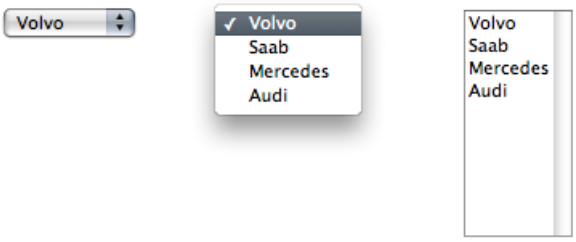

Figure 1.6. A drop-down select menu, closed (left) and open (center), and a select list (right)

Select menus come in two forms: **drop-downs** and **select lists**. Drop-downs, as shown in Figure 1.6, are employed when the user may select only one of the choices. Select lists, like the one to the right of Figure 1.6, are used when the reader can make multiple selections.

The options contained within a select menu may also be grouped, as shown in Figure 1.7.

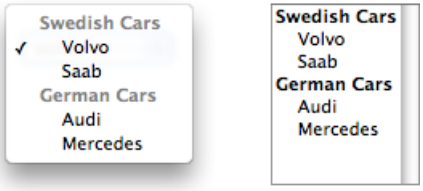

Figure 1.7. Option groups

File Upload

File upload fields are used for … well, uploading files. Perhaps you're uploading some photos to your favorite social network, adding a PDF attachment to an online job application, or sharing a snippet of code on a forum. Pressing the button to select a file will open your operating system's file chooser; from here, you pick the file you want to use.

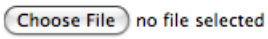

Figure 1.8. The file upload field on a Mac

Fieldsets and Legends

Fieldsets and **legends** are very helpful for organizing forms, especially if they're lengthy forms. They're a way to group related fields together, which can enhance accessibility and usability. The fieldset is the element that contains the group; below, in Figure 1.9, it's represented by the gray line around the fields. The legend is the text shown at the top of that fieldset: **Personalia**.

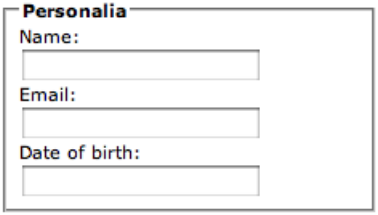

Figure 1.9. Fields contained within a fieldset

Buttons

Last, but certainly not least, we'll need a button to submit the form, like the one in Figure 1.10. Most browsers present these in a manner that suggests they're clickable—a raised or rounded effect, or sometimes both, as shown in Figure 1.10.

Figure 1.10. Click Me!

Enhanced Elements

While the basic form elements we just covered can handle most kinds of forms, progressive enhancements (typically implemented with JavaScript) can help make certain tasks easier, clearer, or more efficient. The best fancy form elements are **unobtrusive**, meaning that there's a way to use the form when JavaScript is unavailable in the browser. Let's look at some examples of enhanced form elements.

Turn those ugly forms into artistic masterpieces. Order the book today!

Split Buttons with Menus

Figure 1.11. Amazon's Wish List button, above, with menu, below

Split buttons are becoming increasingly popular. The button itself will perform a certain action, such as adding an item to a cart; however, a part of the button—often indicated by a segregated area on the right with a downward arrow—will trigger a drop-down menu with other options relating to that action. An example of this can be found with Amazon's[1] Add to Wish List buttons, shown in Figure 1.11. Clicking the main part of the button will add an item to your Wish List, while clicking the arrow at the right edge will reveal a box with further options.

[1] http://amazon.com/

Sliders

Sliders can be used to indicate a number or range. These normally take the form of one control that moves across a bar to indicate the desired amount or value, or with two controls indicating minimum and maximum values, such as time or quantity.

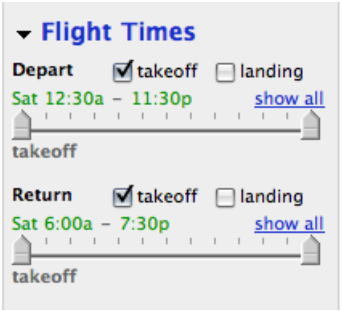

Figure 1.12. Time ranges for flights on Kayak

My favorite use for a range slider is at Kayak,[2] where a slider is used to help you choose acceptable flight times when booking a flight. As you can see in Figure 1.12, this widget uses two handles on the slider's range bar so that you can indicate a start and end time for your flight's departure and return.

Toggle Switches

A **toggle switch** can be used for two either/or choices: for example, on/off, true/false, or public/private. Brightkite,[3] a location-based social network, uses this feature to allow members to quickly switch profiles from public to private and vice versa; this is handy for situations in which you want to quickly and temporarily make your profile public (like when you're attending a conference). In Figure 1.13, you'll see Brightkite's toggle in action.

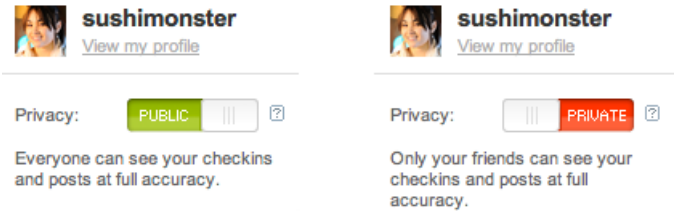

Figure 1.13. Now you see me, now you don't

[2] http://kayak.com/
[3] http://brightkite.com/

Turn those ugly forms into artistic masterpieces. Order the book today!

Auto-completion

Auto-completion is an ideal way to help the user complete fields quicker; it also helps avoid multiple spellings or variations of one specific entry. An example of this can be found in Facebook's profile editor,[4] when filling out your hometown. As you type, a menu appears below the text box displaying a list of possible matches for the city or town you've begun typing. This helps avoid misspellings on the city name or the wrong selection. For example, as seen in Figure 1.14, there are a number of towns named Nashville, so it's important to ensure that you select the right one.

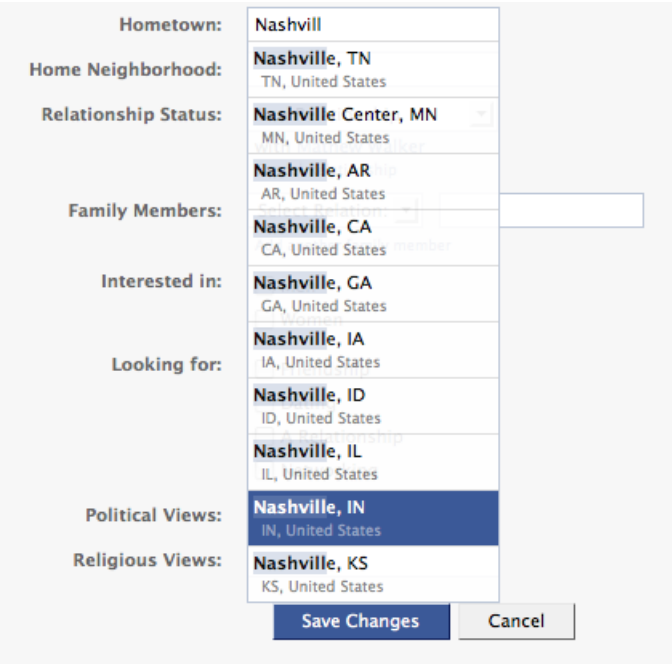

Figure 1.14. Which Nashville?

[4] http://facebook.com/

Date Pickers

Date pickers make choosing a date easier, especially when it might be far off into the future. The ability to see exactly which day of the week a certain date will fall on helps when booking trips or making appointments. Date pickers normally take the form of a calendar. In Figure 1.15, a screenshot from Dopplr,[5] we can see an example of a calendar-style date picker being used for an upcoming trip.

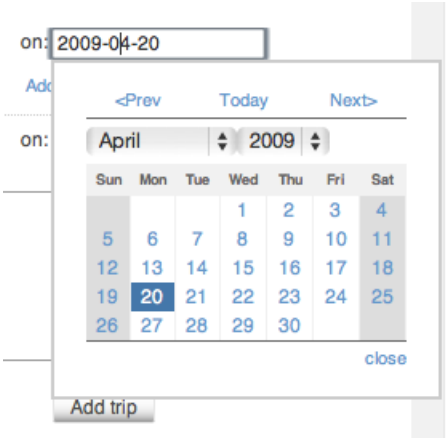

Figure 1.15. Selecting an upcoming date on Dopplr

Color Pickers

Color pickers are usually found in web applications that allow you to customize your site experience or your profile. Some users are unfamiliar with hexadecimal color code—the system used in web development to specify color—or they may lack the tools to help them find that information. Color pickers enable these users to select the color they want, though a good color picker still provides the ability to enter a hex code.

Figure 1.16 shows Virb's[6] color picker in action.

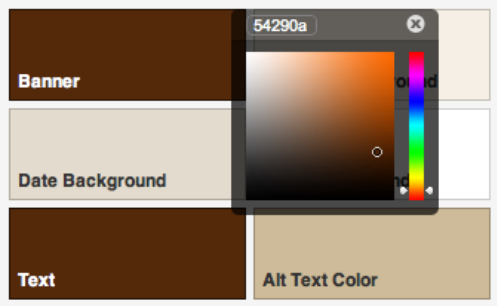

Figure 1.16. Virb's color picker, with hex code entry field at the top of the window

[5] http://dopplr.com/
[6] http://virb.com/

Turn those ugly forms into artistic masterpieces. Order the book today!

Advanced File Uploaders

An **advanced file uploader** can help make uploading multiple files faster and easier. This feature is often found on social networking sites that have photo albums. Flickr's file uploader—shown in Figure 1.17—permits multiple uploads, shows the upload progress of each file, and indicates the size of the files.

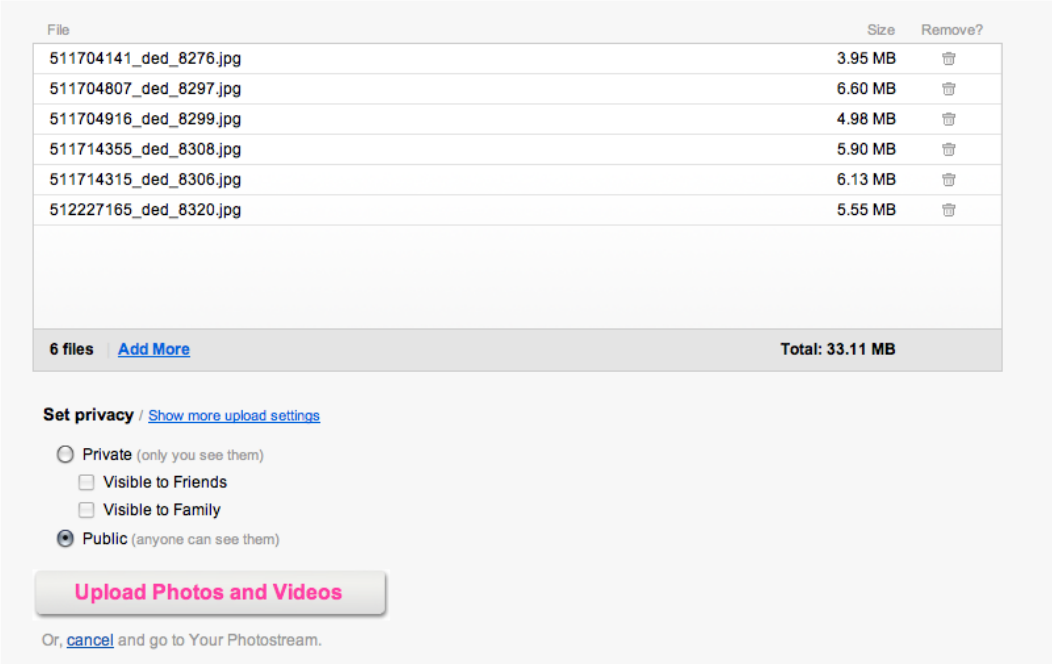

Figure 1.17. The Flickr uploader

Rich Text Editors

Rich text editors enhance the good old textarea by allowing content to be formatted and styled easily. This is commonly found on blogging and content management web sites. In WordPress, the text entry field—shown in Figure 1.18—allows a user to construct a blog post in a familiar, intuitive editor. To the right, there's an HTML tab, so that HTML-savvy users can switch to this mode to view or edit the markup.

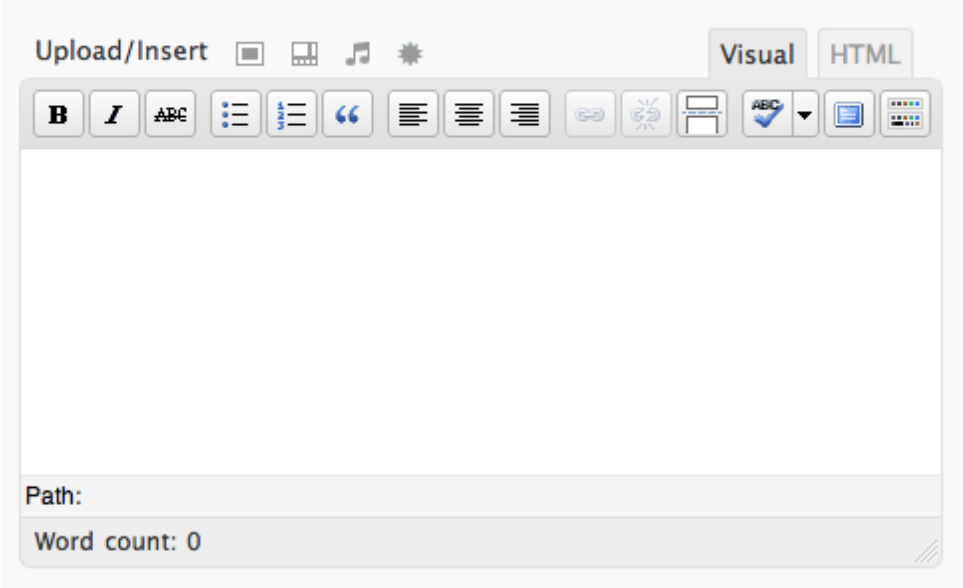

Figure 1.18. WordPress's rich text editor

Drag and Drop

Draggable items provide an intuitive way to reorder objects in a list or move items in and out of a target area. In Panic's T-shirt store,[7] seen in Figure 1.19, customers can pick up T-shirts and drop them into the cart area at the bottom of the screen.

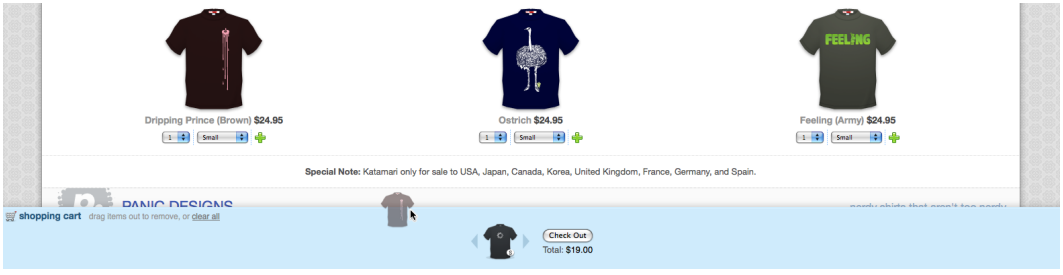

Figure 1.19. Panic's shopping cart

[7] http://panic.com/goods/

Turn those ugly forms into artistic masterpieces. Order the book today!

... And More!

As designers and developers explore what's possible with form enhancement, new methods and techniques appear. As you use the Web, keep an eye out for unusual or innovative uses of form elements. When you spot a form that impresses you, it can be useful to take notes or a screenshot; savvy designers and developers keep a collection of interesting and innovative stuff for later reference. To start your own file, use a scrapbooking-style application like Evernote,[8] a service like Flickr,[9] or even just a collection of screenshots in a folder on your hard drive.

Interaction Patterns

Jargon time! An **interaction pattern** is a way to describe a particular widget, function, or interactive element, describing the particular problems they solve and the rationale for using a given pattern.

Some keen collectors of interaction patterns make their collections available on the Web. At Welie.com,[10] you'll find dozens of different types of menus, widgets, and other interactive elements. The Yahoo Developer Network's Design Pattern Library[11] contains many patterns shown as videos, which makes it easy to understand how the interaction occurs. UI-Patterns[12] is a newer site with a small, but growing, collection. UI Pattern Factory[13] is based on a similar idea, with a growing collection of screenshots pulled from Flickr.

Research and Finding Inspiration

Before you dive into building your form, it's important to do your homework first. Forms are powerful, but without proper planning and design, they can make tasks overbearing or difficult. It's also good to look around for inspiration, for both the interaction and visual aspects of form design.

[8] http://evernote.com/
[9] http://flickr.com/
[10] http://welie.com/
[11] http://developer.yahoo.com/ypatterns/
[12] http://ui-patterns.com/
[13] http://uipatternfactory.com/

For this book, we'll be building a fictional social networking site, **Fit and Awesome**, for people that are into health and fitness. Members of the Fit and Awesome community will be able to store their statistics and training journals on the site, and share their fitness goals and activities with other members. Naturally, a site like this will require a number of different forms and widgets—there's plenty to sink our teeth into here!

Perform a Competitive Audit

Take a look at what web sites and applications similar to yours are doing. It's an ideal way to understand what's successful in other sites or to work out what fails to perform. If you're designing a social network, check out other social networks. How do different companies handle editing a profile? Or privacy settings? If you're working on an ecommerce web site, take a look at some of the successful competitors. Is their checkout process quick and easy, or is it tedious? Do you find yourself entering the same information over and over again?

As you're looking at competitors, take note of issues you come across as you try to use their product, as well as what works well. This information will be helpful in your brainstorming process. Look for possible areas of innovation while still considering established conventions; examine the reasons *why* a technique is effective or not.

In Figure 1.20, I've collected the registration forms for four different sites—Gimme20,[14] Fitness Magazine,[15] Fitness.com,[16] and SELF Magazine.[17] Each form has different features, questions, and interface elements, all of which are useful to consider.

[14] http://gimme20.com/
[15] http://fitnessmagazine.com/
[16] http://fitness.com/
[17] http://self.com/

Turn those ugly forms into artistic masterpieces. Order the book today!

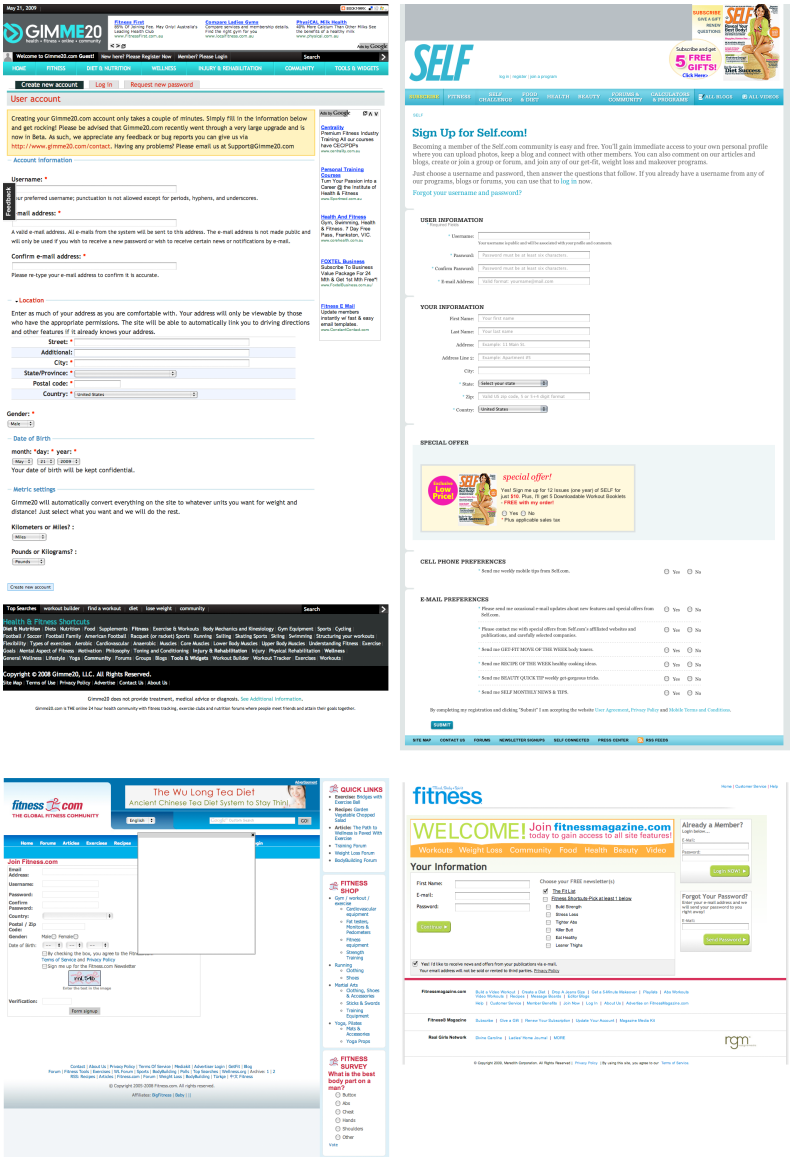

Figure 1.20. The registration screens for Gimme20, SELF, Fitness.com, and Fitness Magazine

Use Software as Inspiration

Desktop software can be a worthwhile place to look for inspiration. These days, more and more web applications are beginning to look and feel like software. This can be good or bad—a poorly chosen desktop widget could confuse a user, and there

are some interactions that just work better on the desktop than in a browser. When you're thinking about including a desktop-like widget in your site, try to be sure that you've chosen it because it's the right tool for the job, rather than because it's fashionable in desktop applications. Consider whether it's necessary. If you could do without it, chances are that you should leave it out.

Figure 1.21 shows the view size menu in Microsoft Word for the Mac; in many ways it resembles a regular select menu, which is hardly notable, but the range of choices and the order in which they're arranged could be a useful technique to emulate or improve upon.

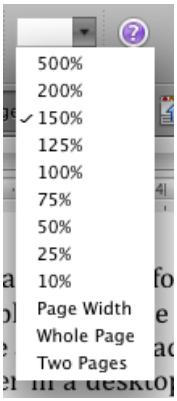

Figure 1.21. Microsoft Word's view size menu

Shown below, in Figure 1.22, Coda's new site panel uses a series of collapsible sections to help a user make more sense of the various requirements. The form itself occupies only a small amount of space, and makes use of a scroll bar at the right. These are both suitable ways to help make sense of a larger form.

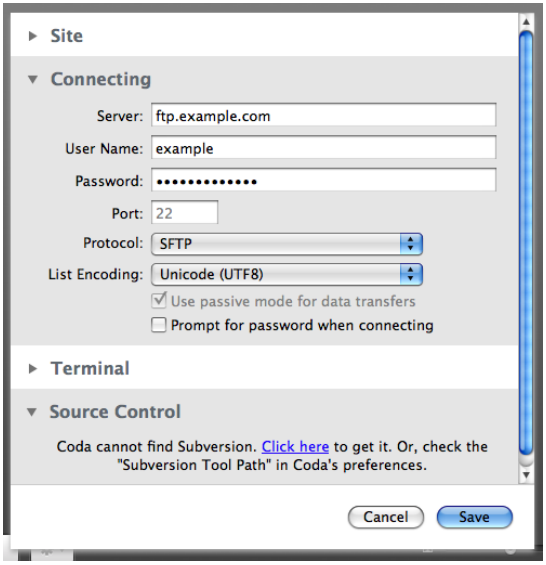

Figure 1.22. Coda's new site panel, with collapsible sections

Turn those ugly forms into artistic masterpieces. Order the book today!

Interaction Design

Interaction design is the process of defining your interface's behavior. For form design, I can't stress enough how important it is. Having a solid, user-centered plan for your forms' designs is the best way to ensure that they're a success. This is where the design helps support a relationship between the form and the user.

Defining the Goal

An important part of planning any project is to define what's required. Writing up documentation, defining task flows, and performing testing may seem a dull, unexciting venture. However, some attention to detail can greatly improve and inform your design process; with a solid idea of what your forms ought to achieve, it's easier to create the solution that best fits the needs of your users. Fancy effects and graphics can make a form look and feel beautiful, but if it fails to provide the solution needed, then the design falls flat.

Creating documentation to describe the expected behavior of a system is an important task, and the resulting material is quite handy when you're working with others or for a client. This type of document is formally known as a **functional specification**.

Identify the Users

Who's going to be using these forms? Are the users tech savvy? Will they benefit from the fancy, progressive enhancements you're thinking about employing or will that actually be a hindrance for them? Some designers go as far as to create **user personas**, fictitious characters that help a designer define the needs and abilities of the kinds of people who'll use the site.[18] For my part, I prefer talking to real people that fit into the target audience. The following table contains a mini profile of four real people whose interests and abilities we'll use as a benchmark when we plan and build our forms.

[18] For a discussion of user personas and how to create your own, visit http://www.hhs.gov/usability/analyze/personas.html

Persona	Profile
Derek Featherstone	Fitness Interests: Triathlons, cycle trainers Technical Level: Very comfortable with the Web
Jina Bolton	Fitness Interests: Gym, casual cycling, WiiFit Technical Level: Very comfortable with the Web
Kelly Steele	Fitness Interests: Casual cycling, tennis, gym Technical Level: Moderately comfortable with the Web
Mathew Walker	Fitness Interests: Basketball, golf Technical Level: New to the Web

Identify Use Cases and Scenarios

While considering the people that will be using your forms, you should think about the various **use cases** that go along with them: so as well as defining the people who'll use your site and their goals, this is about how they can reach those goals using the forms on your site.

Use cases help you answer a number of important questions. What do you (or your client) require? How will the form's data be used? For registering an account on a social network, will members be required to fill the form out in its entirety, or can they just fill out the basics and complete it at a later time? What information should remain public? Should certain kinds of information be kept private? Why would someone register in the first place, and what's important to them?

Turn those ugly forms into artistic masterpieces. Order the book today!

Understand Platforms and Devices

You might be using the latest version of Safari on your nice big MacBook Air, but another user filling out the form could be an everyday commuter browsing on their Blackberry.

When you're planning a form, it's important to consider all the platforms and devices that may be used for your forms, or you might cause a lot of heartache! For instance, I use Yelp[19] for looking up restaurants, and occasionally I submit reviews. I would love to write these reviews as soon as I've left the restaurant, but I never do. Why? Because the form is horribly annoying and tedious to use on my iPhone.

Below, we've specified a number of browsers which we'll use to test and refine our form design.

Desktop Safari 3, Firefox 2, Internet Explorer 6+ (limited functionality)

Mobile Generic mobile CSS, iPhone-specific

Define Task Flows

Now that you've given some careful consideration to the users, use cases, and platforms, you should now have enough information to plot out the steps needed to complete each form—**a task flow**. It's also time to think about alternate paths and error cases. Sketch it out visually, so that you have a clear idea of what your process looks like; Figure 1.23 shows us an example of a task flow diagram for a sign-up and login form.

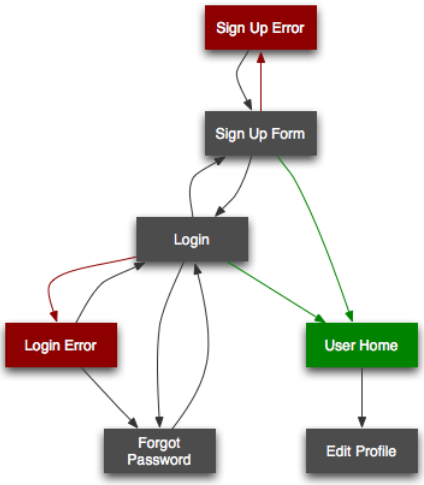

Figure 1.23. A task flow example for signing up or logging in

[19] http://yelp.com/

By this stage you should have a fairly solid idea of what's required in your form. It's now time to put pen to paper!

Paper Prototyping

Creating a **paper prototype** of your form is a quick and easy way to hash out your ideas and issues at the beginning of the form development process. Draw your forms on paper: keep it fast, lightweight, and simple, sketching out a basic idea of how each form would look. You can even use a quick and cheap option like a stack of sticky notes; if you use a note for each object in your form, it's easy to experiment with different arrangements. Use this hand-drawn form to assess how your form looks so far. It's amazing how much more clearer your decisions about form questions become when you see them in front of you.

Try the prototype out on your peers—they might see stuff you missed. If it's okay to show your forms to the public at this stage, perhaps you could head over to the closest café and try them out on some complete strangers after buying them a cup of coffee.

For a great, detailed introduction to paper prototyping, check out Shawn Medero's article in A List Apart.[20]

Wireframing

Now that you've completed your planning, it's time to start designing. Begin with rough diagrams or **wireframes** before obsessing over the shiny buttons. What we're focusing on right now is the layout. The diagrams you'll see over the next few pages, from Figure 1.24 to Figure 1.29, are wireframes for our example site.

Notice that they're plain, clean, and simple—there's no need for a lot of detail or intricate design work here. Right now, we're only hashing out the basic flow and general layout.

[20] http://alistapart.com/articles/paperprototyping

Turn those ugly forms into artistic masterpieces. Order the book today!

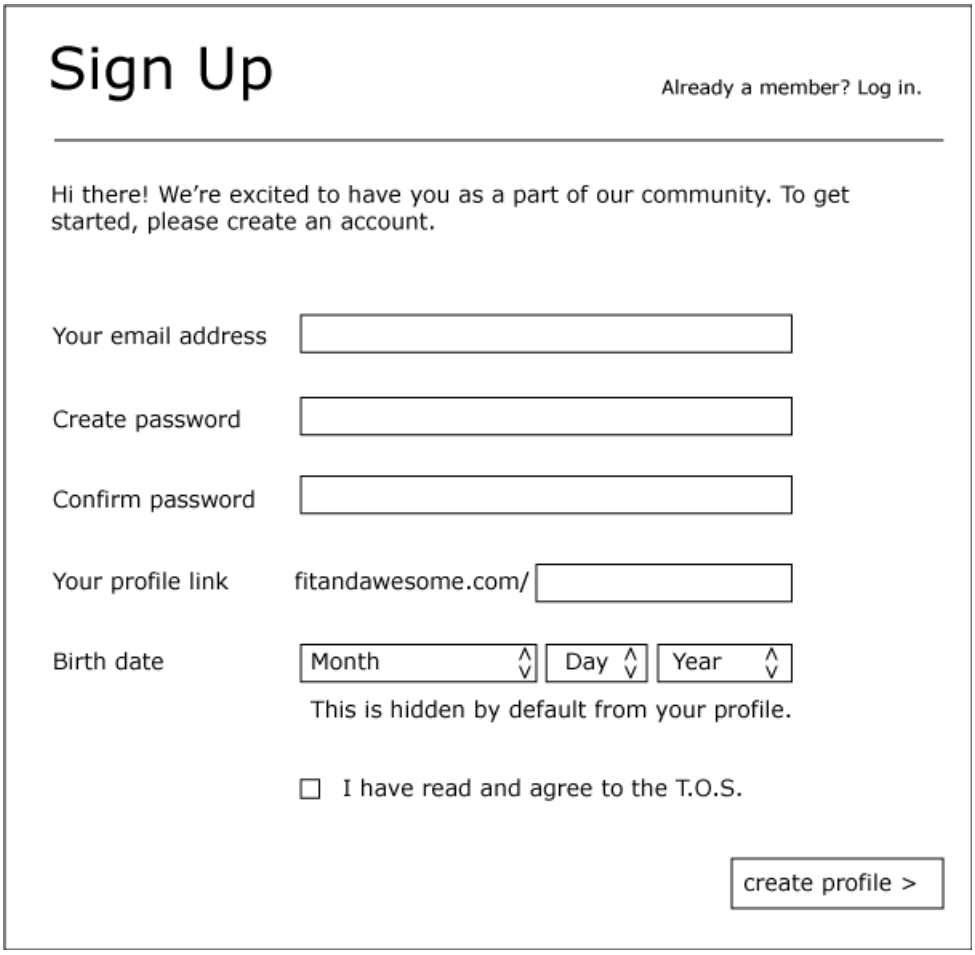

Figure 1.24. Sign Up wireframe

Figure 1.25. Advanced Search wireframe

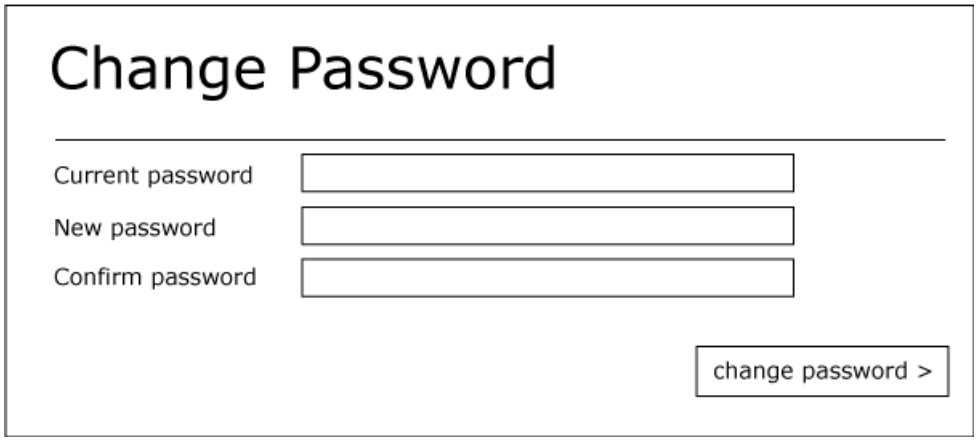

Figure 1.26. Change Password wireframe

Turn those ugly forms into artistic masterpieces. Order the book today!

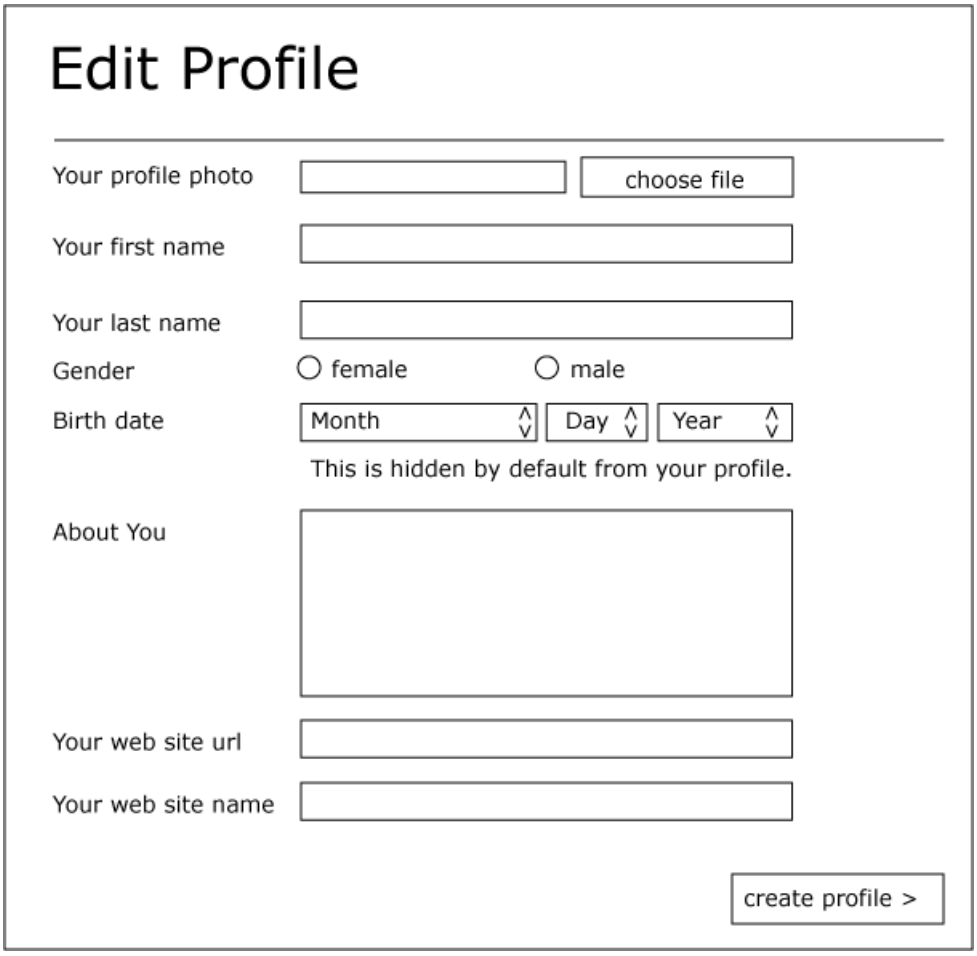

Figure 1.27. Edit Profile wireframe

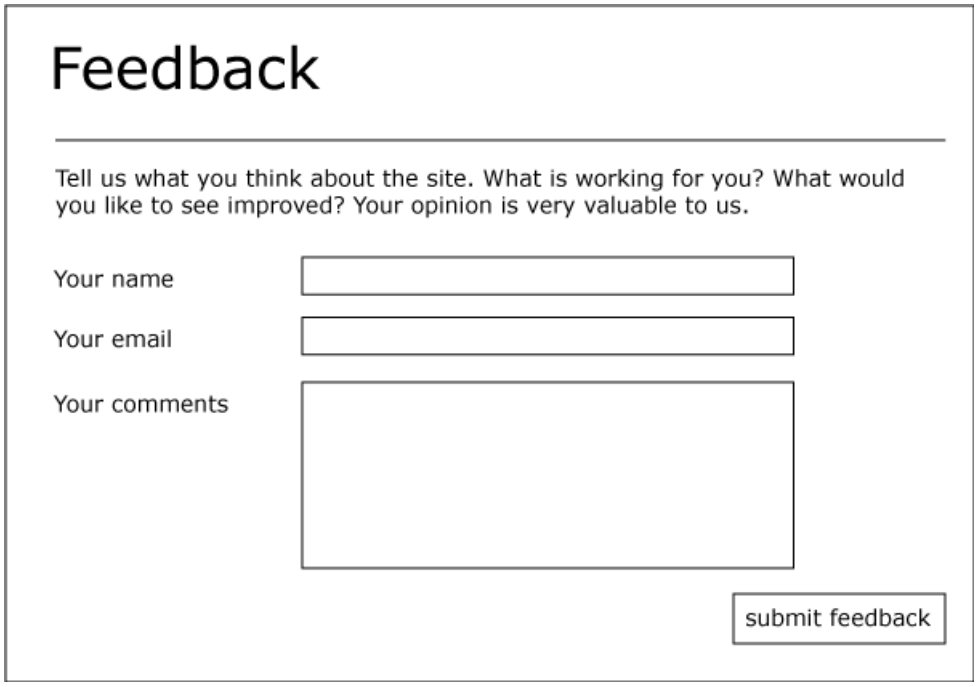

Figure 1.28. Feedback wireframe

Turn those ugly forms into artistic masterpieces. Order the book today!

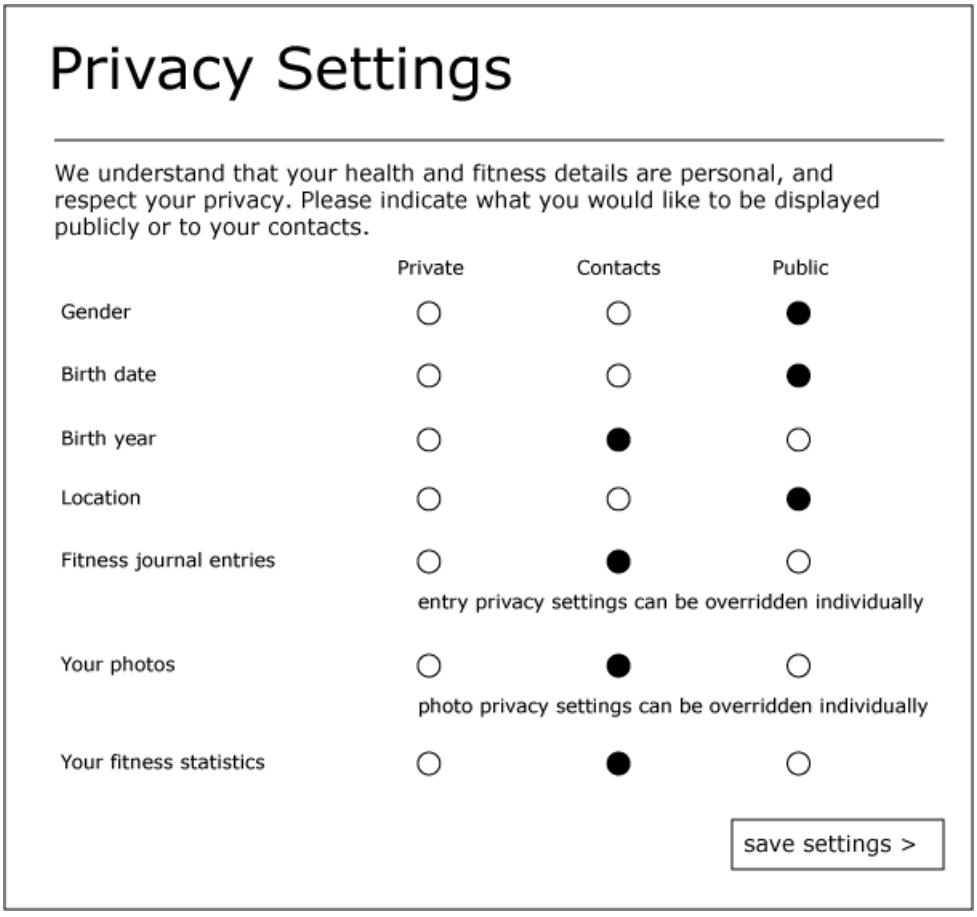

Figure 1.29. Privacy Settings wireframe

Summary

In this chapter we've explored the many elements of forms and why it's vital to develop your forms with an eye to usability, accessibility, and error-free data. We've looked at some of the basic elements like radio buttons, form fields, and select menus, and explored some of the enhanced features available like sliders and color pickers. We've covered the importance of research; talked about methods for drawing inspiration for your form creation; and looked at the process of interaction design, which involves specifications, prototypes, and wireframes.

Believe it or not, you're already halfway there to your goal of creating seriously fancy forms! You might be wondering how that's possible, as you've barely started this book. Well, proper planning is *that* important, and will save you a lot of headaches later. Trust me on this!

Now that we have a firm idea of the general contents and layout of the form, it's time to design the interface.

Turn those ugly forms into artistic masterpieces. Order the book today!

Chapter 2

Designing

In Chapter 1, we focused on the planning phase of fancy form design. With the interaction nailed down, we've actually begun the design process for the forms. Now, we're going to look at the visual design side of form design.

Visual design can either enhance or detract from usability, depending on its implementation. It's important to know how to use visual design in a way that enhances the form's usability, as well as making it aesthetically pleasing. In this chapter, we'll take a look at:

- grid and typography
- color
- imagery

Grid and Typography

When we created the wireframes for the form, we'd already begun the process of thinking about our form's layout. Now, we can tighten the grid and think about proportions and layout patterns, creating a structure for the visual design of our form. Then, we can tune our typography so that our form is legible and clear.

Grid Systems

The **grid** is one of the most fundamental elements of graphic design. It provides a solid foundation for placement of elements within the design of our forms. Using consistent grid systems throughout your web site or application is good for brand stewardship and usability, as well as organization.[1]

For Fit and Awesome, we'll use Figure 2.1 for our forms.

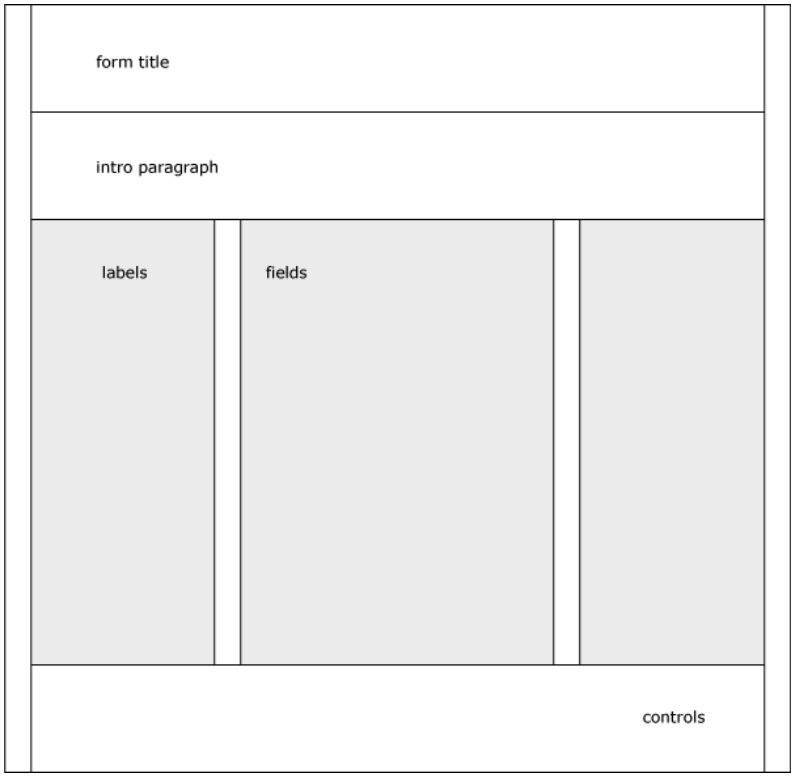

Figure 2.1. A grid system

[1] Some company brands have guidelines that go beyond fonts, colors, and logos; grid patterns are an ideal way to have a united and organized look and feel across materials, whether it's pages in a brochure, billboard ads, or in our case, forms used on a web site.

- The top section is the form's header, which will display the form's title.
- Directly beneath the header is room for an introductory paragraph, if required.
- Then, the form is broken into three columns:
  1. The first column is where we'll place our main labels for our form elements.
  2. The second column is where we'll place those form elements (`inputs`, `textareas`, `select` menus, and so on).
  3. The third column is blank for now, but can be used for messages or other contextual elements, like icons.
- Below the columns is an area for controls or buttons.

Of course, like any system there'll almost certainly be exceptions to the rule: it's okay to break out of the grid from time to time, as the design and interactivity of a form can change based on your goals and those of your users. We've kept our grid system quite simple to allow plenty of flexibility.

Type

A form that's easy to read is likely to be a form that's easy to use. This is especially important for forms that have multiple sections or steps, so that users avoid feeling overwhelmed or lost. Figure 2.2 shows our Sign Up form with elements placed within a grid system—but with poor typography, it's difficult to read. Since every bit of text looks more or less the same, it's harder to scan through the text.

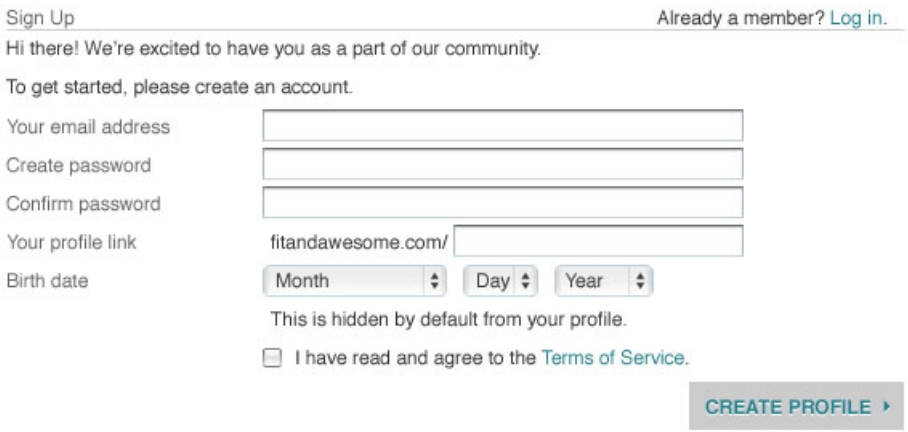

Figure 2.2. Poor typography makes this form difficult for the eye

Turn those ugly forms into artistic masterpieces. Order the book today!

Let's improve on it. In Figure 2.3, we've added various weights, sizes, and shades to our text to provide contrast and a hierarchy. There's also more space around each form question. With these simple adjustments, the form already feels a little more organized and is easier to read.

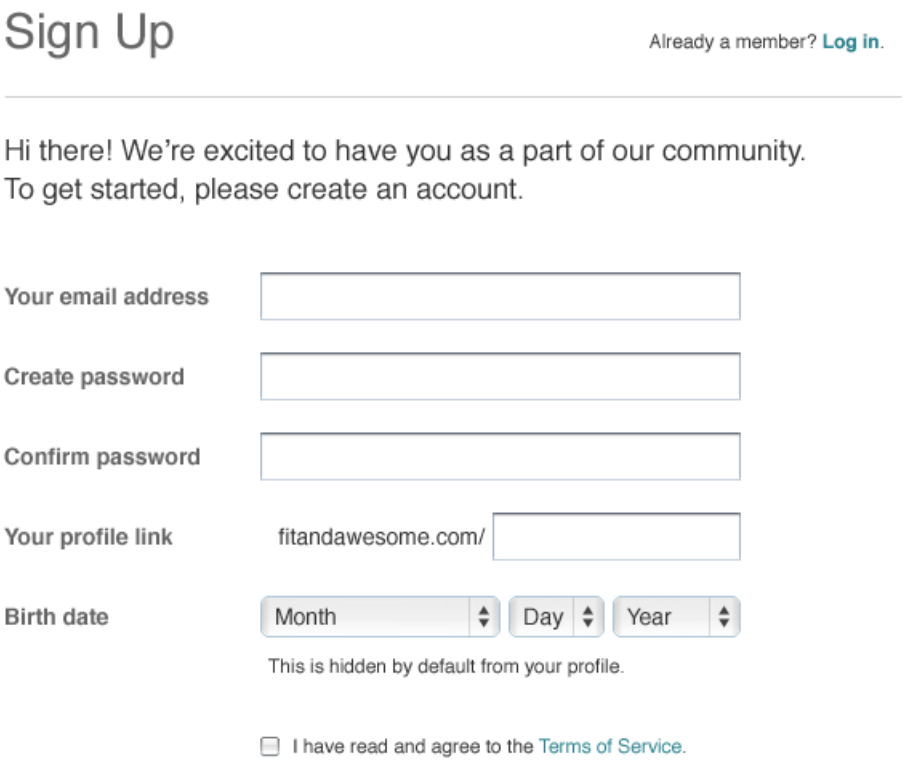

Figure 2.3. Much better!

Naturally, we can improve this further with some additional color.

Color

Color is, of course, a great way to make our forms look more interesting, but there's more to it than that. While grid systems help us organize the placement of elements, color systems can help emphasize an item's importance or meaning, making for a much more usable interaction. An example would be using color to indicate an error state, or to highlight a required field.

Highlighting Calls to Action

A **call to action** is a phrase used in interaction design that refers to the action you'd like your user to take. In the case of a form, the call to action often concerns the user entering some details or clicking a button in order to complete a particular task.

Primary calls to action associated with a form may include an Add to Cart button, a Sign In button, or a Post Entry button. It's a good idea to use noticeable, bold colors for these types of buttons. Secondary calls, such as a Cancel button, can use subtler, muted tones, to show that they're of less importance.

In Figure 2.4, we see an example of this used at West Elm,[2] an ecommerce site that sells furniture and home décor. The primary call to action here is to encourage users to add a product to the shopping bag, so the **Add to Shopping Bag** button uses a darker gray. It stands out more than **Add to My Project**, which uses a lighter shade.

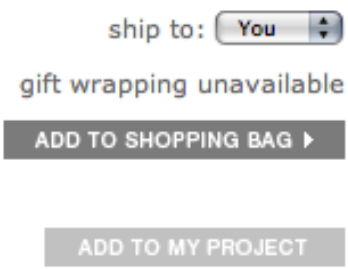

Figure 2.4. West Elm's calls to action

[2] http://westelm.com/

Turn those ugly forms into artistic masterpieces. Order the book today!

While shades of gray are capable of doing the trick, we'd like to use some green for the primary call to action in our forms for two reasons: the color will make the button more obvious, and it will complement the overall color scheme of the rest of our site.

Figure 2.5. Preview and Save buttons

In our Edit Profile form, saving the form is our primary call to action, so we've used green to ensure that it stands out more than the light gray-shaded preview button.

The Message of Color

When choosing colors, be aware of the message that each color may convey. Greens connote a positive tone—they send the message that it's okay to proceed, or that an action you performed was successful. Reds can signal a negative outcome: highlighting an error, or advising that your action may cause an undesired effect, such as canceling an activity or deleting a profile.

While color helps add hierarchy and meaning, it's important to remember that color is just one of many methods for indicating a message. Some users are unable to see color; they may be color-blind or visually-impaired, or they may use a device that only displays in grayscale, such as Amazon's Kindle ebook reader. Use a mix of icons, color, or text to indicate the fields that need attention. Some forms will explain the errors and omissions at the top of the form as text, as well as beside the fields.

In Figure 2.6, red is an appropriate color to indicate fields that were filled out incorrectly. Yet, there's also a message above the form which explains the problem, so that the meaning of the problem is clear regardless of whether the user comprehends the color red, or not.

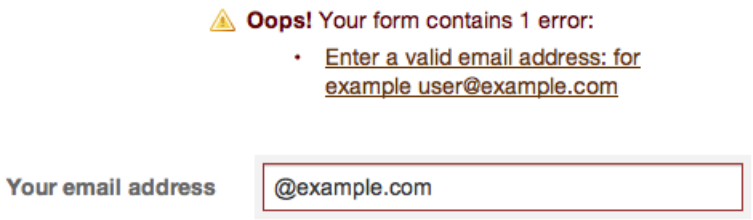

Figure 2.6. A message explains the problem, while a border highlights the incorrect field

Imagery

Imagery can add a nice touch to your forms, enhancing the design. More importantly, imagery can be used to provide additional context, such as highlighting an error or a required field in the form.

Iconography

Icons can be a highly effective visual indicator. Let's think again of our error message shown above, in Figure 2.6; this is an ideal place for an icon, and helps to further emphasize the error field. We'll place the icon to the right, which you can see below in Figure 2.7.

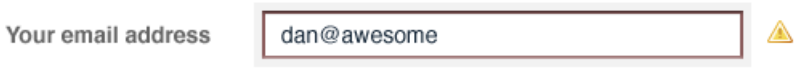

Figure 2.7. Adding an icon to the error message

Background Patterns and Textures

Subtle background patterns and textures help add volume and depth to a form. While it may seem a purely aesthetic consideration, textures that add dimension to a form can go a long way to improving its usability. Rather than being merely a bunch of flat rectangles, these form elements seem to lift from the page—they feel more authentic because their interactive nature is emphasized.

Turn those ugly forms into artistic masterpieces. Order the book today!

Here's our Sign Up form so far, in Figure 2.8. Our typography changes have helped lift the text, and the color on the Create Profile button is an improvement, but overall this is hardly a fancy form!

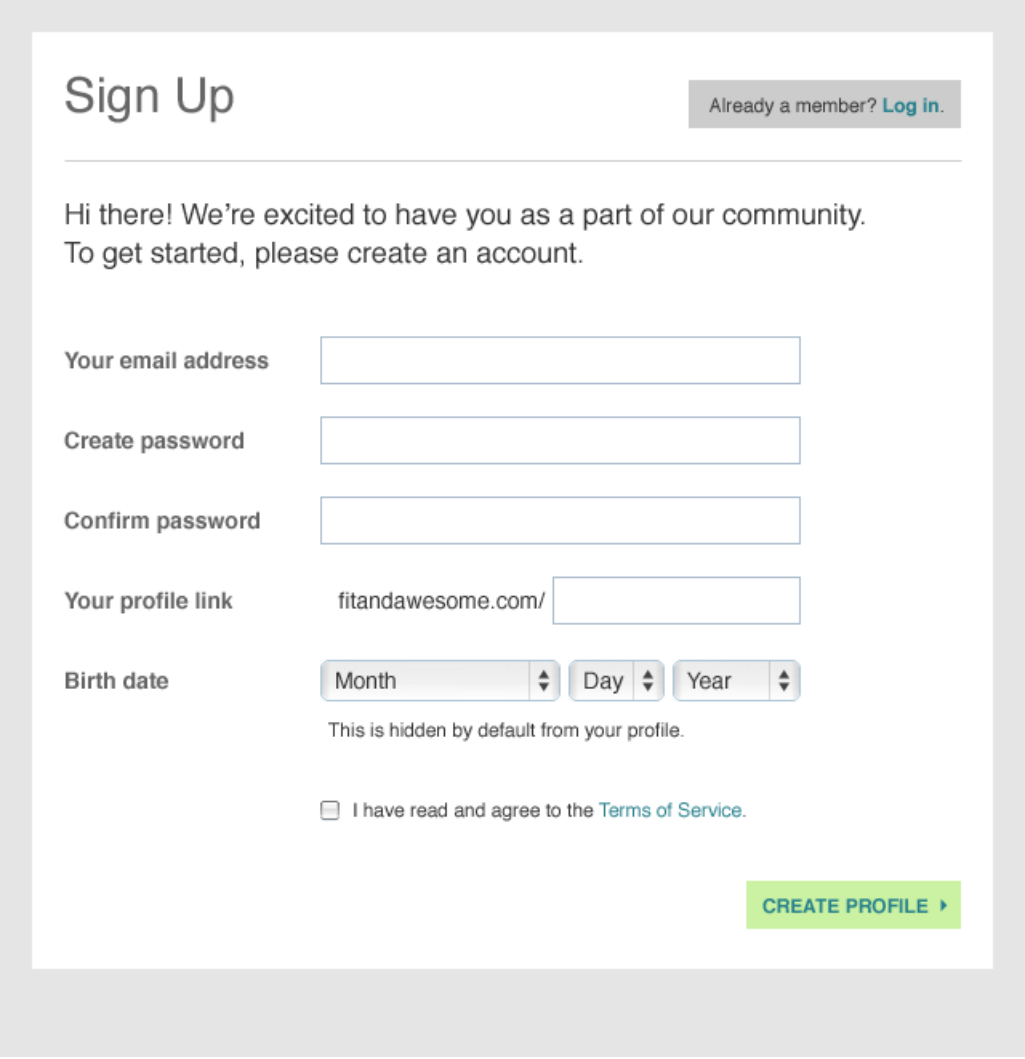

Figure 2.8. Our simple form is still a little drab

Now, let's add some depth and texture to the form. As seen in Figure 2.9, gradients and shadows provide definition: the fields are set off with a light gray background, making each field more distinctive. It's also a useful way to associate text with relevant sections, as seen under the birth date fields. Adding a raised effect to the Create Profile button helps to emphasize this important element, too. Overall, it's a much friendlier, more usable form!

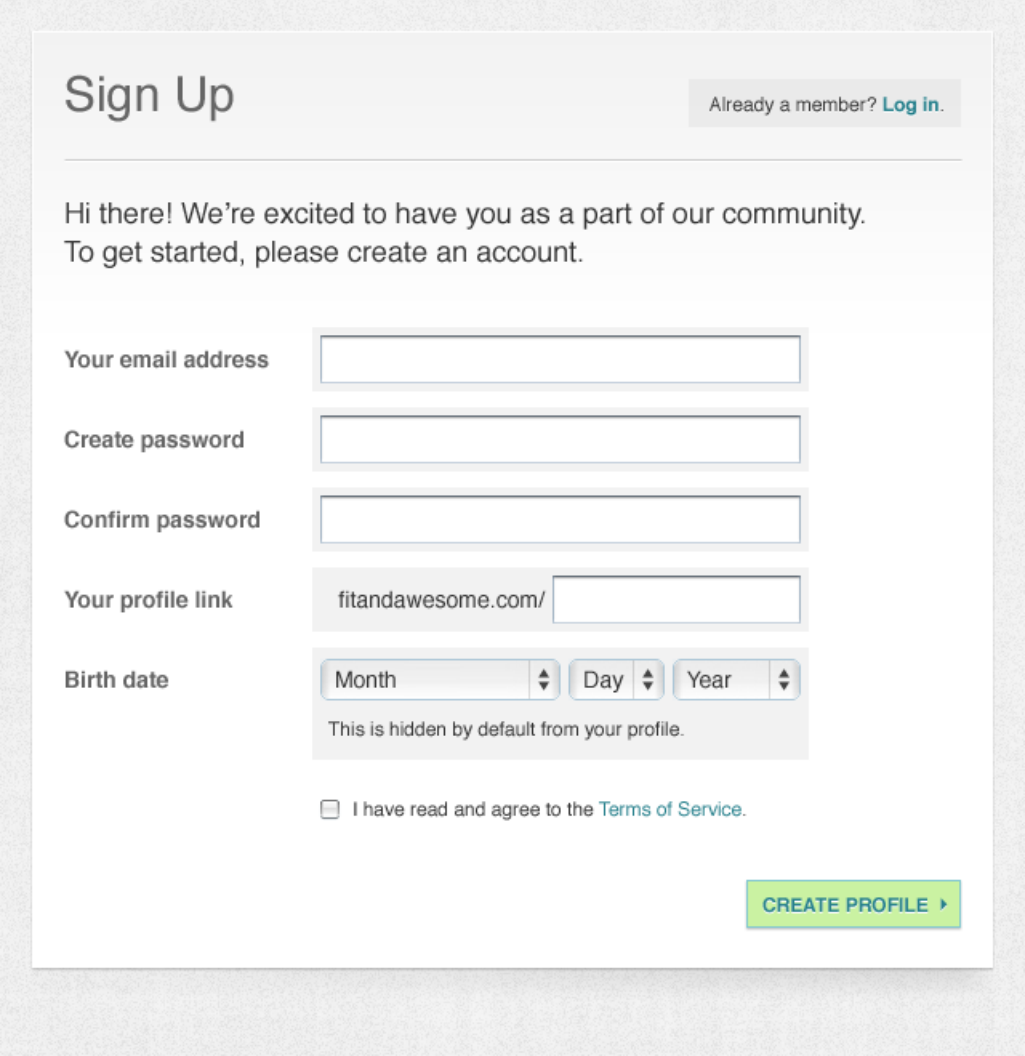

Figure 2.9. Gradients, shadows, and texture help this form pop!

Turn those ugly forms into artistic masterpieces. Order the book today!

Let's take a look at the rest of our forms with the new grid, typography, color, and image styles applied, seen in Figure 2.10 through to Figure 2.14.

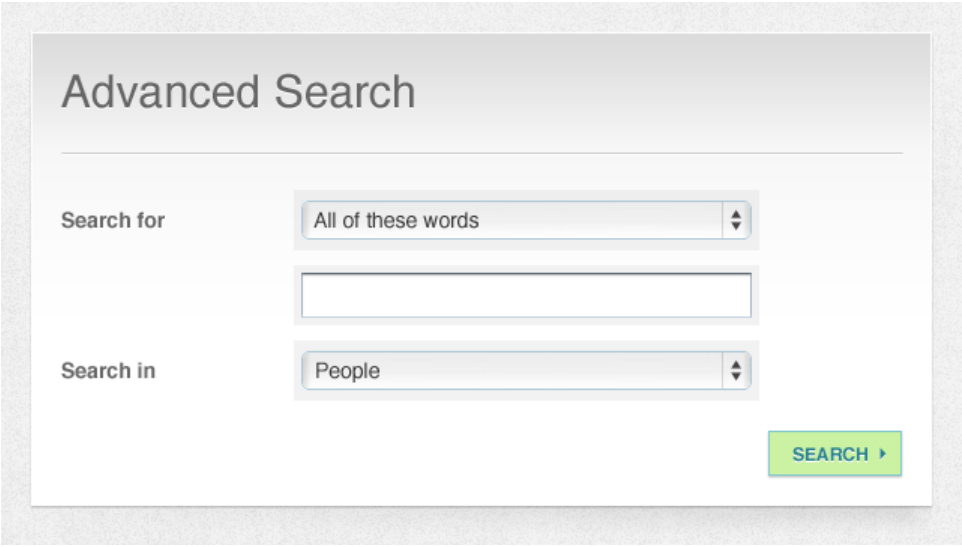

Figure 2.10. Advanced Search

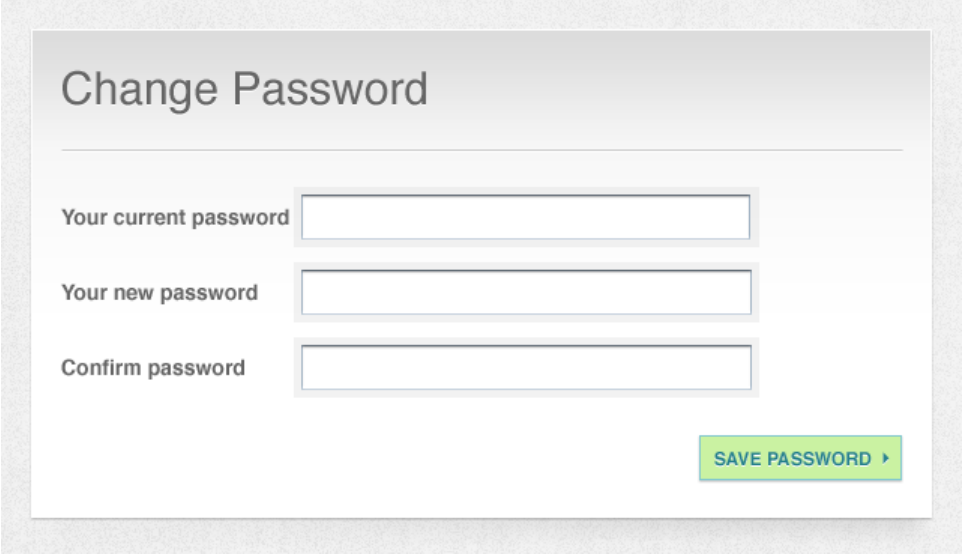

Figure 2.11. Change Password

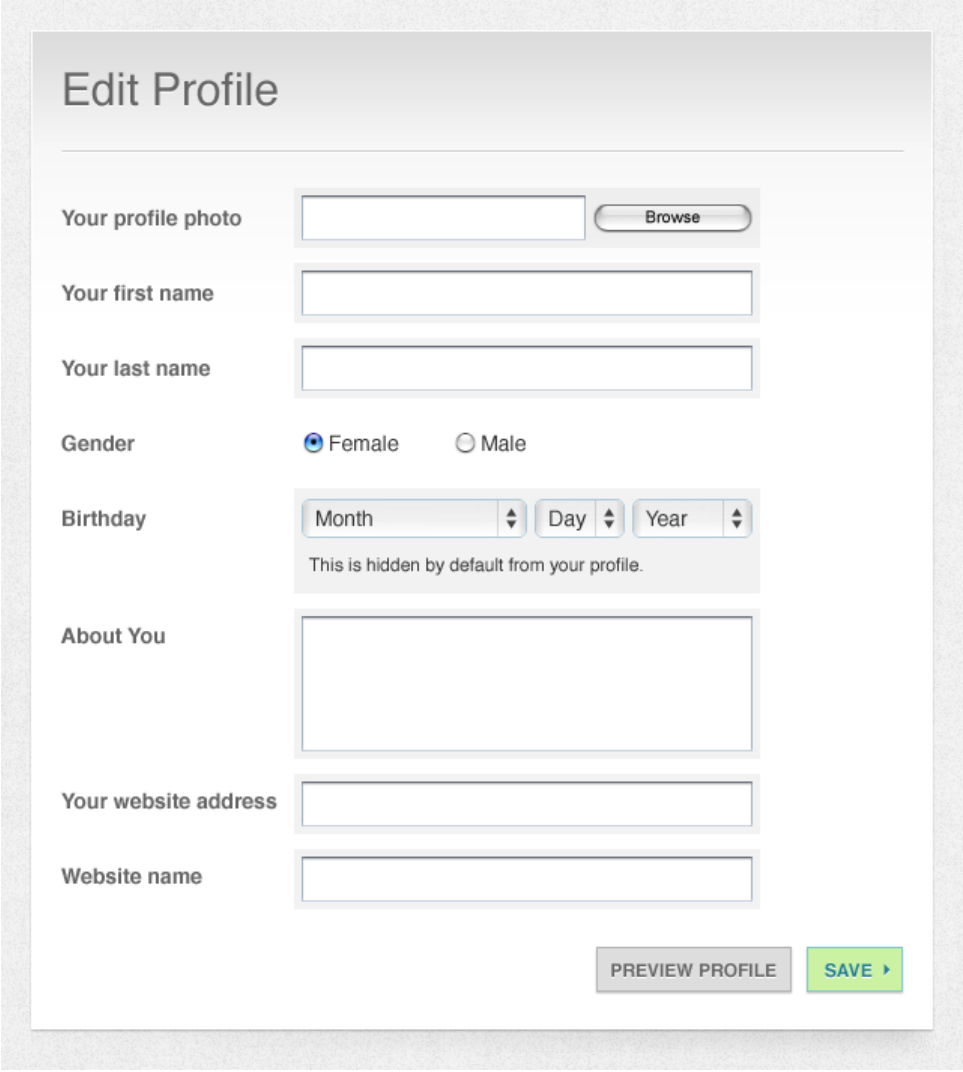

Figure 2.12. Edit Profile

Turn those ugly forms into artistic masterpieces. Order the book today!

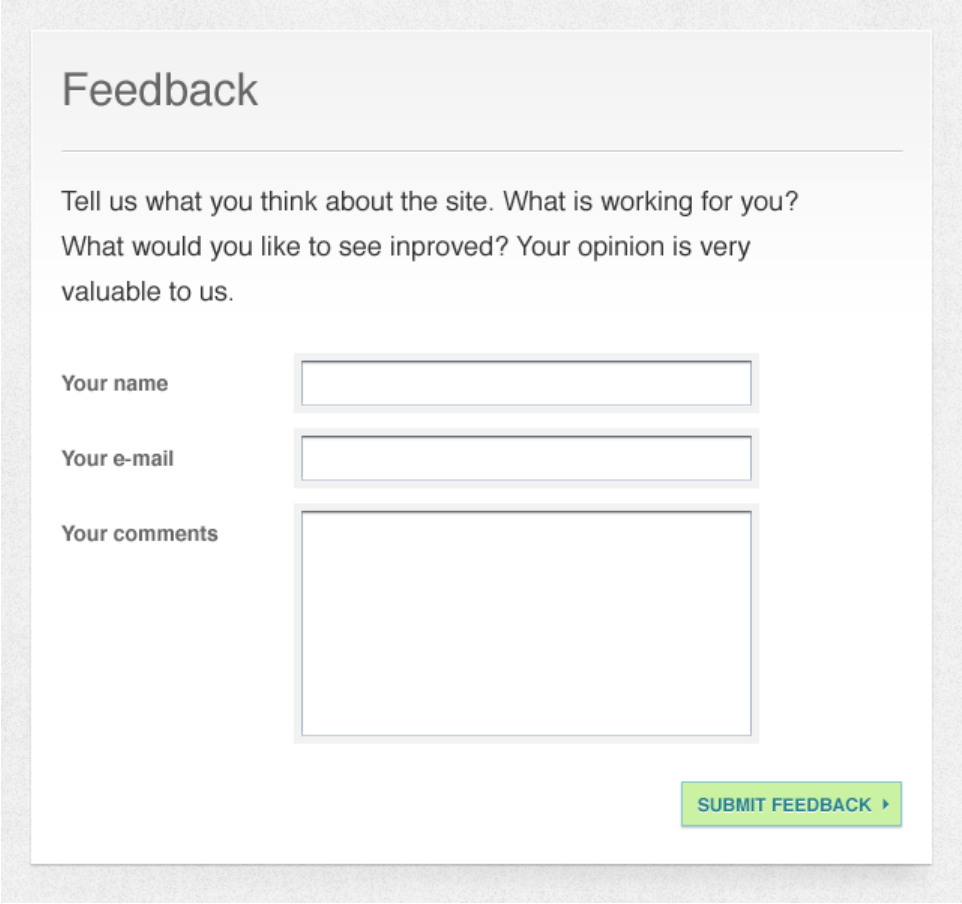

Figure 2.13. Feedback

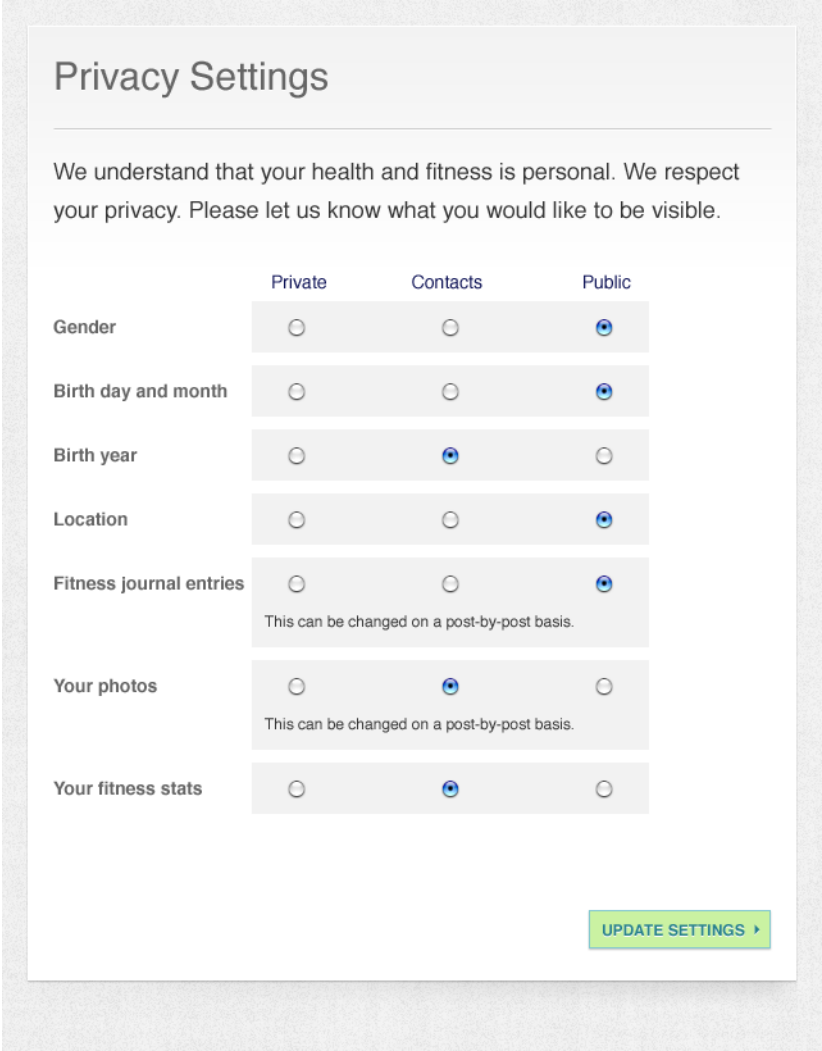

Figure 2.14. Privacy Settings

Turn those ugly forms into artistic masterpieces. Order the book today!

Figure 2.15 shows a detail from our Sign Up form, this time showing the error state and informative message.

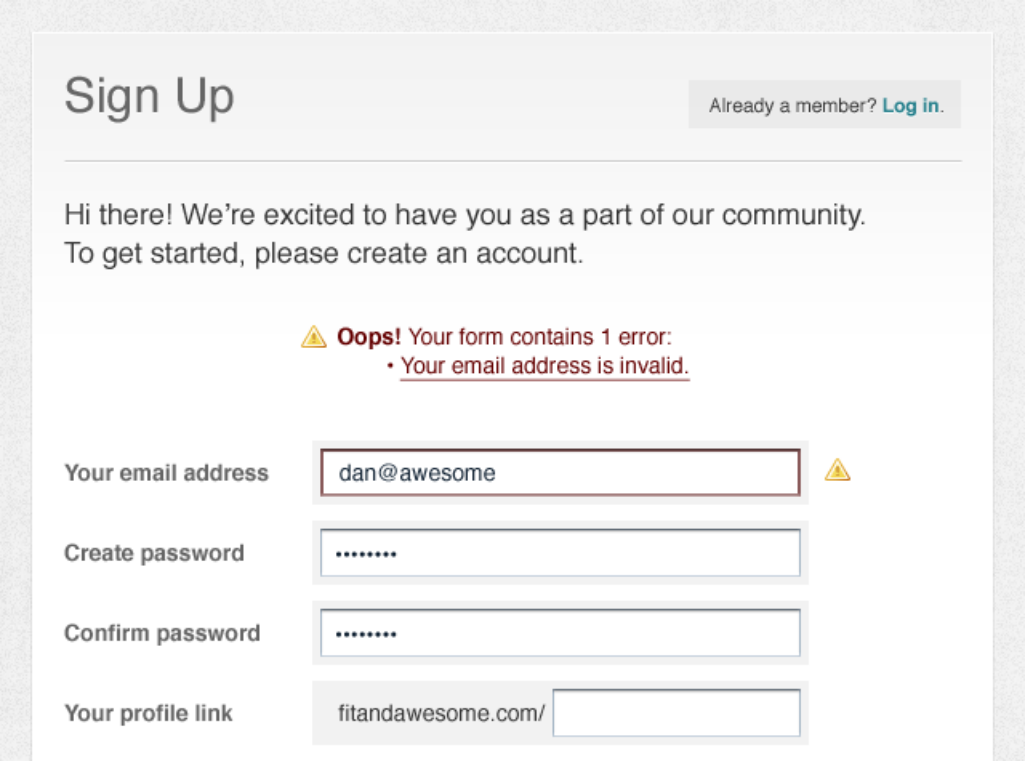

Figure 2.15. Oops!

Summary

In this chapter we discussed the value of strong visual design: as well as adding to the aesthetic appeal of forms, it should also enhance the usability experience for users. We looked at how well-structured grid systems and clear typography are necessary for layout and legibility. We examined how color can be employed to great effect, especially when used to highlight a primary call to action, but also that it's important to be mindful of what different colors can signify. Alongside this, we established that color should always be used in conjunction with other indicators, as some users—for various reasons—are unable to view color. Finally, we looked at how imagery such as iconography or background patterns and textures can be implemented to create further volume and depth.

With elegant and creative use of typography, grid, color, and imagery, your forms should have a much fancier experience—planned out well, and designed to be hot stuff, too! We're now ready to move on to the next step: structuring the form with markup.

Turn those ugly forms into artistic masterpieces. Order the book today!

What's Next?

We've just scratched the surface in this excerpt. In the rest of the book, we'll explore a whole workflow for creating great-looking forms. As you work through the rest of the book, you'll create a beautoful design, move through to building it with best-practice markup and stunning CSS styling, and finally top it all off with some beautiful, functional JavaScript enhancements.

100% Satisfaction Guarantee

We want you to feel as confident as we do that this book is an essential part of your development arsenal. That's why you have a full 30 days to read and use it with absolutely no risk. If in that time you feel it hasn't given you valuable tips and tricks, then simply send the book back and we'll give you a prompt refund of the full purchase price minus shipping and handling.

So for the cost of a new T-shirt, learn how to create stunning, usable, fancy forms!

To find out more and to order your copy, visit https://sitepoint.com/bookstore/go/169/e23030.

Index

A

accesskey attribute, 54
advanced search form example, 64
advisory text, 55
Ajax, submissions with, 131
aligning to grid, 101–104
Amazon's Wish List button example, 6
audience expectations, 76
audits, competitive audits, 13
auto-completion
  elements, 8
  widgets, 129

B

background patterns and textures, 33
Birth Date fieldset, 107
boxes, checkboxes element, 3
Brightkite's toggle example, 7
buttons
  button element, 5
  Preview and Save buttons example, 32
  radio buttons element, 2
  split button element, 6
  submit button, 94, 106

C

Çelik's undohtml.css, 78
calls to action, highlighting, 31
Camino browser, 95
canvas, styling, 77
change password form example, 65
checkbox element, 3, 121
client-side validation, 130
cloud-hosted frameworks, 133
Coda's new site panel example, 15
color design, 31
color picker element, 9
competitive audits, 13
condition question display, 122–126
conventions, web interfaces, 55
CSS frameworks, 79
CSS signature, 77

D

date picker element, 9
date selectors, 127
design, 27–41
  background patterns and textures, 33
  color, 31
  grid systems, 28
  iconography, 33
  interaction design, 16–24
  typography, 29
desktop software as inspiration, 14
disabled attribute, 54
div element, 54, 101
Dopplr date picker example, 9
drag and drop, 11
drop-downs menus, 4
Dylan Verheul's autocomplete plugin, 129

E

edit profile form example, 66
editors, rich text editors, 10

elements
(*see also* fields)
auto-completion of fields, 8
buttons, 5
checkboxes, 3, 121
color pickers, 9
date pickers, 9
div, 54, 101
drag and drop, 11
fieldsets, 5, 45–51, 81
file uploaders, 5, 10
form, 44
img, 57
labels, 3, 45–51, 83
legend, 5, 45–51, 81
radio buttons, 2, 121
rich text editors, 10
select menus, 4, 108
sliders, 7
span element, 81
split buttons, 6
structure of, 44–55
text fields, 2
textareas, 3
toggle switches, 7
Eric Meyer's reset CSS, 79
errors
about, 55
adding icons to error messages example, 33
indicating in forms, 56

F

Facebook's profile editor example, 8
feedback form example, 69
fields
(*see also* elements)
Birth Date fieldset, 107
prefixed field, 106
required, 55
text fields element, 2
fieldset element, 5, 45–51, 81
file upload element, 5, 10
Firefox preferences dialog example, 46
Firefox widgets example, 86
Flickr uploader example, 10
floats, 84
form element, 44
frameworks
cloud-hosted frameworks, 133
CSS, 79
jQuery, 120
functional specifications, 16

G

get method (form element), 45
global white space reset, 78
goals
design, 27
interaction design, 16
Google's Ajax Libraries API, 133
gradients, styling, 89
grid systems
about, 28
aligning to, 101–104

H

highlighting calls to action, 31
HTML
form elements, 2–12
and form structure, 44

I

iconography
  about, 33
  styling icons, 90–96
id attribute, 52
img element, 57
input validation, 130, 136–146
instructions on form, 55
interaction design, 16–24
  goals, 16
  identifying users, 16
  paper prototypes, 19
  wireframes, 19
interaction patterns, 12
Internet Explorer
  select menu, 108
  widgets examples, 87

J

JavaScript, 119
Jörn Zaefferer's autocomplete plugin, 129
jQuery, 120, 131, 133
jQuery Custom Input plugin, 122
jQuery Custom Selectboxes, 121
jQuery Geogoer VChecks, 122
jQuery iPhone-style Checkboxes plugin, 122
jQuery Password Strength Meter, 129
jQuery plugins, 126
jQuery Selectbox plugin, 121, 142
jQuery UI Selectmenu, 121
jQuery UI's Datepicker, 127
jQuery's validation plugin, 136

K

Kayak sliders example, 7

L

label element, 3, 45–51
  side-by-side, 84
  top-aligned, 83
legend element, 5, 45–51, 81
libraries
  Google's Ajax Libraries API, 133
  Yahoo CSS library, 79
  Yahoo Developer Network's Design Pattern Library, 12
Login link, 100

M

Mac-style widgets example, 85
method attribute (form element), 45
Meyer's reset CSS, 79
Microsoft Word's view size menu example, 15

P

Panic's shopping cart example, 11
paper prototypes, 19
PassRoids plugin, 129, 133
password strength indicators, 128, 133
patterns, background, 33
personas, user personas, 16
planning, 1–25
  elements, 2–12
  interaction design, 16–24
  research and inspiration, 12
platforms and interaction design, 18
post method (form element), 45

Turn those ugly forms into artistic masterpieces. Order the book today!

prefixed field, 106
privacy
  (*see also* security)
  privacy settings example, 24, 39, 53, 59, 70
  styling, 113
prototypes, paper prototypes, 19

R

radio button element, 2, 121
read-only attribute, 54
registration forms, examples of, 13
required fields, 55
resetting styles, 78
rich text editors, 10

S

scenarios, 17
security, password strength indicators
  (*see also* privacy)
select list, menus, 4
select menu element, 4, 108
server-side validation, 131
side-by-side labels, 84
sign up form example, 61
slider element, 7
sliding doors technique, 143
span element, 81
specifications, functional, 16
split button element, 6
sprites, 91
structure, 43–74
  elements, 44–55
  final markup, 58–74
  instruction, errors and advisory text, 55
styling, 75–117
  about, 75–77
  aligning to grid, 101–104
  Birth Date fieldset, 107
  checkboxes and radio buttons, 121
  fieldsets and legends, 81
  form questions, 83–96
  Login link, 100
  prefixed field, 106
  privacy, 113
  resetting styles, 78
  select menu, 120
  submit button, 94, 106
submit button, 94, 106
switches, toggle switch element, 7

T

tabindex attribute, 53
Tantek Çelik's undohtml.css, 78
task flows, 18
text
  advisory text, 55
  rich text editors, 10
text fields element, 2
textareas element, 3
textures, background, 33
toggle switch element, 7
top-aligned labels, 83
typography, 29

U

UI-Patterns, 12
undohtml.css, 78
uploading files, 5, 10
use cases, 17
user personas, 16

user style sheets, 77
users, identifying, 16

V

validating inputs, 130, 136–146
Verheul's autocomplete plugin, 129
Virb's color picker example, 9

W

web interfaces, conventions, 55
West Elm's calls to action example, 31
widgets, 120–132
  autocomplete, 129
  checkboxes and radio buttons, 121
  condition question display, 122–126
  date selectors, 127
  input validation, 130
  jQuery UI Selectmenu, 121
  password strength indicators, 128
  select menu, 120
  styling, 85
  submission with Ajax, 131
wireframes, 19
WordPress's rich text editor example, 10

Y

Yahoo CSS library, 79
Yahoo Developer Network's Design Pattern Library, 12

Z

Zaefferer's autocomplete plugin, 129

Turn those ugly forms into artistic masterpieces. Order the book today!