designing the moment

web interface design concepts in action

robert hoekman, jr.

Designing the Moment:
Web Interface Design Concepts in Action

Robert Hoekman, Jr.

New Riders
1249 Eighth Street
Berkeley, CA 94710
510/524-2178
800/283-9444
510/524-2221 (fax)

Find us on the Web at: www.newriders.com
To report errors, please send a note to errata@peachpit.com

New Riders is an imprint of Peachpit, a division of Pearson Education

Editor: Wendy Sharp
Production Editor: Hilal Sala
Copyeditor: Jacqueline K. Aaron
Compositor: ICC Macmillan Inc.
Indexer: FireCrystal Communications
Cover design: Robert Hoekman, Jr.
Interior design: Joan Olson

ISBN-13: 978-0-321-53508-5
ISBN-10: 0-321-53508-1

9 8 7 6 5 4 3 2
Printed and bound in the United States of America

Contents

Introduction

A good user experience is all about good moments.

In one moment, a user's goal can be to figure out the name and purpose of a site after landing there from a Google search result. The goal of the next moment can be to figure out the controls for a video player to watch a screencast about an application. The next can be to figure out how to sign up, or to find out about pricing plans, or to contact the company.

These are not *life* goals, like those that many designers prefer to capture in the form of a "persona" (a description of an archetypal user within a particular product's audience), but rather *interaction goals*.

Achieving great design means asking ourselves, What is it we want the user to do in this moment and how does the interface encourage him do it?

What is the user's goal in this moment, and how does the design help her accomplish that goal? Is her goal to get oriented to a new site? Find specific information? Complete a form? Add something to the shopping cart?

Each moment has the potential to increase a user's confidence or destroy his trust in a product or company, and each one is an important piece of the whole experience.

Why? Because the task a person is attempting to complete at any given moment is the most important task to that person, at that moment.

It's our job to make sure nothing goes wrong. To make sure that moment is enjoyable and productive, and helps our user feel smart.

Our job is to solve for *all* of these moments. To design something that supports each of these goals without interfering with any of them. To create a cohesive whole out of the oft-disparate parts.

Our job is not to design screens, it's to design *moments.*

It doesn't matter how simple or complex an application might be. What matters is what happens when a user tries to accomplish his interaction goals for any given moment.

One of the keys to achieving great design is to look at our work in terms of what has to happen in each one of these moments to make it successful, and then solve for that.

Navigating a Moment

In these moments, users take all kinds of actions. In the course of a day, they can cruise from Amazon to Z Gallerie and do everything from register to quit.

The actions users take online are broad and varied. We do things like input, edit, search, format, create, upload, delete, share, organize, and participate. These are the actions designers want users to take. And much of the time, these are actions we *want* to take.

Of course, we also do things we don't usually realize we're doing. We analyze. We judge. We forget, make mistakes, lose our train of thought, change our minds, get lost, and become confused. If we're lucky, we learn things. We get oriented. We form ideas, memorize, habituate, trust, get inspired, and feel productive.

This book is about designing interfaces that support all of these behaviors—you know, the things that make us human—in a way that is conducive to good decision-making in the moment a choice is presented. It's about creating interactions that inspire people to input, edit, search, share, and do all those other wonderful things we want them to do. It's about designing applications that help people feel productive in spite of their innate tendency to forget, make mistakes, and change their minds.

More specifically, it's a collection of over 30 stories that illustrate how to put good design principles to work on real-world Web application interfaces to make them obvious and compelling. From the first impression to the last, these stories are about looking critically at designs and questioning every detail to ensure that human beings—the kind that make mistakes and do things we don't expect—can walk away from our software feeling productive, respected, and smart.

Designing for the Moment is

- A revealing and insightful "think out loud" approach to interface design
- A critical look at elements from every phase of a user's interaction with a Web application, one moment at a time
- A set of best-practice recommendations for the design of everything from page layouts to social-networking features

Most of all, it's a critical look into the subtle details of an interface that make or break a user's experience during the moments he navigates his way through them, and how to improve each and every one.

The Design of Interactions

This book is organized in order of the actions users typically take when they encounter a Web application, from the pivotal first few moments of getting oriented to the moment they finally close up their accounts and move on to something else. The titles of each part correspond directly to these actions. Each part is about something they actually *do* with a Web application in a given moment.

Parts 1 through 3 are about things users typically are able to do within the first 30 seconds or so of encountering a new application. The stories in these sections are about exploring an interface for the first time, finding their way around, reading bits of text to learn and get familiar, watching videos and animations, and searching for specific information.

Parts 4 through 6 are about things users do as they get more involved with an application. Once they decide to create an account, they input information, edit content, manage data, and participate in a variety of ways, and each of these moments can have a lasting impact on their confidence with an application.

And Part 7 focuses on what happens when users bail out, either temporarily or permanently.

In other words, I've intentionally structured this book so that it forms a complete picture of a user's experience with a Web application, in a way that addresses each and every phase of this human-to-product conversation, whether it lasts three seconds or two years. As a collection, the elements discussed in these stories comprise the complete picture of the online user experience.

The book is also organized in this way so that you can return to it later on and use it as a sort of "reference guide of inspiration". When grappling with the design of a new editing feature six months from now, you can open up the Participating section in this book, read a few pages, and hopefully get some ideas.

And it's all done via the timeless art of storytelling.

Postcards from the real world

Some of these stories are about specific implementations of design patterns. Some are about individual situations you may never face. Some are about new ideas. Some are about old ones. Regardless, they're all aimed at illuminating the thought processes, facts, findings, theories, and hunches that go into the design of great moments.

In *Designing the Obvious* (my previous book), I talked at length about seven core, guiding principles of Web application design that produce a common-sense approach that can be reproduced consistently and successfully in any Web project.

In *Designing the Moment*, I offer stories about how I've applied these ideas to real projects to create effective moments for users, and I offer a ton of new ideas along the way.

Some of these stories are personal—about things I've designed in the past. Some are brand-new, designed specifically for this book.

All of them are straight from the mind of someone who obsesses over the interaction design of, well, pretty much everything. Someone who lies awake at night thinking about how the local casual dining joint could tweak its drink counter to optimize the flow of customer traffic. Someone who spends the vast majority of his time thinking about the moments that make up a user experience and how to improve them.

Making decisions, out loud

At the end of *Designing the Obvious*, I talked about how important it is to step up and make design decisions, even if you're sure they will change later on. Regardless of how long each decision lasts, it needs to be made so that everything in a design is considered and deliberately directed. Nothing should be left to fate.

In this book, I attempt to do something I've never had to do before. I *articulate* the process of making those decisions.

I *think out loud* to try to shed some light on the *kaizen* ("continuous improvement") approach to design. To show how iteration is absolutely essential for achieving good results. To show how design principles, research, experience, hunches, and *feelings* are all applied to guide the design process.

In other words, I make decisions, out loud.

I also admit to mistakes, give credit to other people, and generally demonstrate that good design is the evolutionary result of a whole lot of bad design. Hopefully, through this, you'll see that good design is not merely the product of creative genius or moments of divine inspiration. It's the product of *forward momentum.*

You'll be able to go back over these pages for years and question my decisions. And you may reach a point where all the things I've said here have become ridiculously obvious to you, and you have surpassed anything I could offer you. My greatest hope is that this very thing happens for you.

I'm putting myself out there to try to help you learn to question yourself. To find fault in every design, and to continually look for ways to improve it.

Starting the conversation

With all that in mind, this book is *not* meant as a definitive guide for the design of Web interfaces. This book is offered as a conversation starter. It's meant to get you *thinking.*

These are not definitive answers, because there are no definitive answers. When I make a specific recommendation, it's based on my own experiences,

perspectives, and knowledge. Sometimes, it's based entirely on my *hunches* (something every designer should have and trust).

As such, I can practically guarantee that you will come across something in this book that you think you can do better—some way to improve a moment for your users in a way I hadn't thought about. If you do, I want you to talk about it. Don't be an armchair designer—go out and tell people about your improvements.

Send me an email about your ideas. Blog about them. Tell your friends. Better yet, tell your coworkers. And your bosses.

I don't believe for a second that I'm the most knowledgeable designer on the planet and that everything I do is gold. I learn from and get inspired by other people *all the time.* With *Designing the Moment*, I hope to teach and inspire *you*, but I strongly believe and hope that you'll end up teaching others as a result.

If you have a good idea, talk about it. Keep the conversation going.

Everything we do can be done better, even when we've already done our best. I've tried to do my best with these designs. If you find a way to take them even further, speak up!

And just so you don't need to take notes

Because typing out URLs found in books is a terribly annoying process of looking-typing-looking-and-typing-again, I've created an archive of all the Web sites, applications, blog posts, articles, research papers, and other stuff referenced throughout this book. You won't find URLs anywhere in these pages. If you want to check something out, visit **www.rhjr.net/dtm**. Click on the **Links from the book** link.

Without further ado, let's get to it.

13

Nailing Form Layout

Fortunately, the vast majority of forms we encounter on a daily basis online are decently short. They consist of only a few fields, maybe a drop-down list or two, and a button. And they exist to help us get to things we want, enabling us to register for a new site or application, make a purchase, or contact a company.

But have you ever wondered why something as simple as a short form seems different on every single site?

Some have two-column layouts with left-aligned labels. Some are right-aligned. Some are single-column with left-aligned labels above fields. Submit and Cancel buttons and the like appear on either the left or right side of the form, sometimes with the Submit button on the left of Cancel, sometimes with it on the right. And labels . . . well, don't get me started on labels.

For such simple things, there are a huge number of variants.

Sometimes, these designs are the result of pure circumstance. Maybe a designer decided that right-aligned labels simply looked better on a page, designing them to fit on the page instead of designing the page around them. Maybe a developer just threw the form together the same way he did the one on his own site. Better yet, maybe someone snagged the code from another site without bothering to modify it.

Regardless of the reason, there's a good chance the form wasn't tailored to create a good moment that contributes to the overall user experience. As a result, many forms are missed opportunities.

You may not realize it, but there are times when each variant of form layout can have a positive or negative impact on how the form (and your site) is used

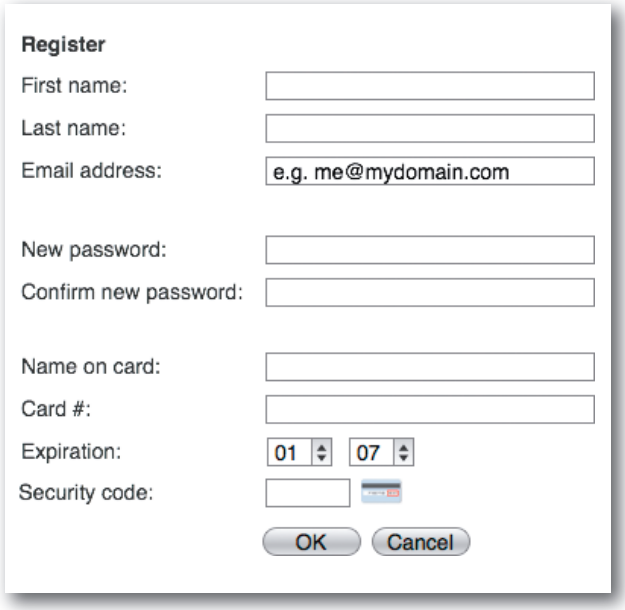
or perceived. And applying the right variant at the right time is one of the simplest things you can do to improve your user's experiences.

Designing Forms That Flow

A form needs to flow well, and the layout of a form can play a big part in how successfully a user interacts with it. Such was the case while designing a registration form for a community site.

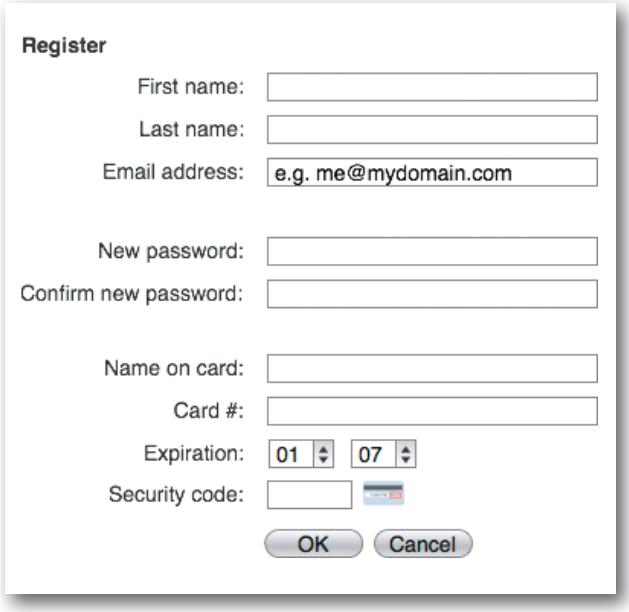
The form needed only to collect the user's name, email address, desired user name and password combination, and credit card information.

I considered a two-column layout with left-aligned labels.



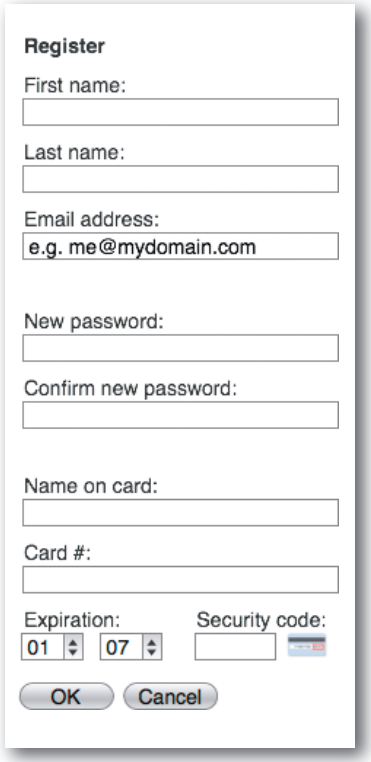
This made the form nice and short, and it's likely all of it could have appeared "above the fold" on the page (in the area in view by default, without scrolling), but this layout presents a higher risk of error than other layouts do. This is because it can be difficult to map a straight line from a field label, such as Email, to its corresponding field, which appears in the second column.

The two-column layout also slows users down, which can be good if the form needs to be read carefully, but is again conducive to mistakes. With a standard registration form, it wasn't necessary to slow users down, so I considered a right-aligned version.



When labels are right-aligned in a two-column layout, users make fewer mistakes than they do with left-aligned labels, because the labels and the fields are closer together. That said, right-aligned labels create an awkward, staggered edge along the left side of the page, and this lack of uniformity can be unappealing. Plus, the two-column layout still slows users down, no matter how the labels are positioned.

So I decided to plow forward with a single-column layout with top-positioned, left-aligned labels.



This layout takes users the least amount of time to associate the labels and the fields and so they tend to move through it more quickly than they do two-column layouts. The label and field can be viewed in a single eye movement rather than the two that are required to read a label in one column and glance at a field in another. Because of this improved grouping, the single-column layout also reduces mistakes.

I use a single-column layout with top-positioned, left-aligned labels in most of the forms I design, because I always consider how the layout of a form will affect the user's moment as she interacts with it, and I often end up choosing this layout as the right solution. It's rare that I need to intentionally slow users down as they complete a form, and in such cases I usually rely on good validation techniques instead of slowing the users down. We'll talk about this more in Chapter 15.

Perfecting OK/Cancel

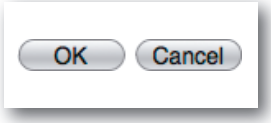
There's one more thing about even the simplest forms that can trip up users and lead to mistakes. It's that pesky OKyCancel button set that appears at the end of them.

OK and Cancel buttons are omnipresent on the Web. Of course, they aren't always labeled OK and Cancel and they don't always have the same purpose, but everyone has seen them a million times nonetheless. The combination of the two buttons has maintained its position as a global standard for a very good reason: almost any action you perform in a Web application can be canceled.

This is seen often in round-trip interactions, in which the user starts an interaction on page 1, completes it on page 2, and is then returned to page 1. It's also seen in inline interactions, such as changing the title of a Backpack page.

Typically, the set of buttons is displayed as two side-by-side buttons created with standard browser controls. Sadly, this design has persisted for years and has found its way into millions of applications.

In my form, I started with a typical OK/Cancel button set.



In this case, the object of the interaction is to register for an application. Simple enough. But there are numerous ways to solve any given design problem, and even something as simple as this requires a few decisions.

The first decision to make is about the position of the buttons. It really doesn't matter much in terms of usability whether the buttons are right-aligned or left-aligned to the field above them, but I generally stick to the left side for a few reasons.

First, left alignment keeps everything flush to the left edge, forming a nice, straight line from the top of the form to the bottom. Second—and this is simply an aesthetic choice—it seems to serve as a visual *anchor* to the form. As you

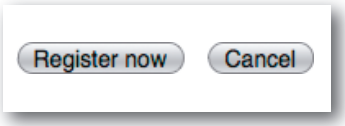
complete each field, you move steadily downward. Placing the action buttons at the tail end of it seems to . . . nail everything down. But that's just me.

The second decision to make is how to label the buttons. Sure, using the standard OK and Cancel requires less effort on a case-by-case basis, because you can simply label them without really thinking about it and move on, but are these terms really helpful?

OK works fine in a lot of cases, but more meaningful button labels can go a long way toward setting a user's expectations about the result of clicking the button.

What does clicking OK do in this example? It registers the person using it as a new customer. But elsewhere in the application, OK might mean saving a configuration setting or uploading a file.

Clicking OK doesn't set a clear expectation. Fortunately, labeling the button in a meaningful way was really no big deal. All I did was change the label from OK to Register Now. Rather painless.



Primary and secondary actions

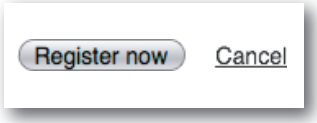
The final thing to think about here is what goes unchallenged by designers most frequently. It's the fact that *equal weight* is often given to both buttons, each of which triggers a remarkably different result. One registers the user, the other cancels the whole operation (in this case, returning the user to the home page), but the two buttons are identical in appearance.

I can understand why this apparent tradition started. Two options equates to two buttons. But what's important is not the number of options. What matters is which option is *most likely.*

The most likely option for users who have decided to register for the application and fill out this form is to click OK. This is the **primary action.** The less likely option is to cancel the registration process. This is the **secondary action.**

By giving both buttons equal visual importance, however, users have to actually read the labels—on *both* buttons—to decide which one to click.

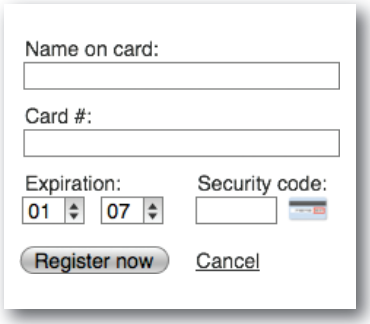
Applying Fitts' Law, which dictates that the time it takes to hit a target is a function of the distance to the target and the size of the target, the ideal solution is to take some of the focus away from the secondary action—the Cancel button. I did this by turning it into a text link.



This way, the Register Now button is more prominent and easier to click, while the Cancel link is less prominent and slightly less easy to click.

Is it any more difficult to see both buttons? No. But it's slightly more difficult now for the user to click the wrong one. The user's eye is drawn to the large button, and it's easy to recognize, subconsciously, that the large button is the probably the primary action. If that's the one the user wants, he doesn't even need to bother reading the label. If it's the secondary action he's after, he can easily infer that the less prominent option is the Cancel link. (Ironically, creating visual separation between the two makes both easier to read, so even though users can infer their meaning without consciously reading the labels, they're now more likely to read them.)

This, incidentally, is another reason why I prefer to keep the buttons left-aligned.



It simply looks better to have that large HTML button aligned to the left edge of the field above it than it would to align the Cancel link to the right edge.

So, why is it helpful to keep the button on the left and the text link on the right? Because any user who uses the Tab key to move through form fields will reach the Register Now button first, again putting focus on the most likely choice while simultaneously taking focus away from the less likely choice.

It matters

Now, don't get me wrong here. A few seconds' difference in the time it takes to complete the form in each layout isn't going to make or break a user's experience. When it comes down to it, what matters is not shaving off a precious few seconds from the process, it's whether or not the value in completing the form outweighs the difficulty or tediousness of doing so.

Convincing a user to sign up in the first place is the hardest part of getting him through the registration process. But once you've accomplished this, the goal becomes to treat the user as well as possible as a customer. And considering subtle differences such as these in form design—really paying attention to the details—can result in another smooth moment for a user. And each moment contributes to the overall user experience.

If you're going to spend all the time and energy to create a Web application, why leave the design of your Web forms to chance, or to circumstance? Treat these things intentionally, and you're guaranteed to earn happier customers.