

DELIVER FIRST CLASS WEB SITES

101 ESSENTIAL CHECKLISTS

BY SHIRLEY KAISER



ESSENTIAL CHECKLISTS FOR USABILITY, ACCESSIBILITY, AND MORE

Deliver First Class Web Sites: 101 Essential Checklists (Chapters 6, 12, and 14)

Thank you for downloading these three chapters of Shirley Kaiser's *Deliver First Class Web Sites: 101 Essential Checklists*.

This excerpt encapsulates the Summary of Contents, Information about the Author and SitePoint, Table of Contents, Preface, and three chapters of the book.

We hope you find this information useful in evaluating the book.

[For more information, visit sitepoint.com](http://www.sitepoint.com)

Summary of Contents of this Excerpt

Preface	ix
6. Color	79
12. Search Engine Optimization	219
14. Testing.....	257
Index.....	323

Summary of Additional Book Contents

1. Let's Get Started ... But How?	1
2. What to Find Out: Initial Questions to Answer	73
3. Preparing Web Site Content	155
4. Managing all the Content	155
5. Web Site Usability: Focus on the User	155
7. Information Architecture	155
8. Navigation	155
9. Best Coding Practice: W3C Standards and Recommendations	155
10. Creating Accessible Web Sites	155
11. Web Site Optimization.....	155
13. Design	155
15. Preparing for Launch	155
16. Post-launch Follow-up	155
A. Ecommerce Checklists.....	155

Deliver First Class Web Sites: 101 Essential Checklists

by Shirley Kaiser

Deliver First Class Web Sites: 101 Essential Checklists

by Shirley Kaiser

Copyright © 2006 SitePoint Pty. Ltd.

Technical Reviewer: Ian Lloyd

Managing Editor: Simon Mackie

Technical Editor: Priya Singh

Technical Director: Kevin Yank

Editor: Georgina Laidlaw

Index Editor: Nigel d'Auvergne

Cover Design: Jess Bentley

Cover Layout: Alex Walker

Printing History:

First Edition: July 2006

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 to be 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.

424 Smith Street Collingwood
VIC Australia 3066.

Web: www.sitepoint.com

About the Author

Shirley Kaiser created SKDesigns¹ in 1996 to focus on creating visually appealing web sites that are based on web standards and universal accessibility, and are optimized for speed and search engines. You'll find her designs featured as best practice examples in major publications and textbooks, such as *JavaScript + CSS + DOM Magic*² and *The Complete Guide to Using and Understanding the Internet*.³ Shirley also helps to promote web standards as a Web Standards Project (WaSP) Steering Committee member.

In addition to authoring this book, Shirley is a contributing author to *Creating Web Pages with HTML and Dynamic HTML*. She's also written dozens of articles and tutorials for WebReference.com, Digital Web Magazine, WebsiteTips.com, and her weblog, Brainstorms & Raves.⁴

Aside from her web work, Shirley is a professional musician with Bachelor of Arts (B.A.) and Master of Arts (M.A.) degrees in Piano Performance. She's also a devoted mom who cherishes time with friends, spoils her dogs, and enjoys photography, reading, movies, growing flowers, and cooking and eating tasty food all over the world.

Shirley lives in central California's Sacramento area (Carmichael) with her family, not far from where she grew up.

About the Expert Reviewer

Ian Lloyd is a web designer and developer who believes passionately in the importance of web standards. He is a member of the Web Standards Project⁵ and has spoken at several high profile web conferences, including South By Southwest in Austin, Texas and @Media in London.

Unlike many of his peers, Ian does not run his own company but remains a 'salary man', working for Nationwide Building Society in the UK (where he constantly harps on about web standards, accessibility and usability to anyone with a sympathetic ear!).

Ian lives in Swindon, UK, a town that is known only for two things: the famous "magic roundabout"—a mega roundabout that comprises five individual but joined roundabouts, and for being that place off television show *The Office* (thus making it second in dullness only to Slough). That said, Ian does his best to get out of Swindon in his treasured air-cooled VW Camper van⁶ when he's not glued to the laptop for one reason or another.

¹ <http://skdesigns.com>

² Makiko Itoh, *JavaScript + CSS + DOM Magic* (Indianapolis, New Riders Press, May, 2002).

³ Linda Bird, *The Complete Guide to Using and Understanding the Internet* (Englewood Cliffs, NJ: Prentice-Hall, 2004).

⁴ <http://brainstormsandraves.com>

⁵ <http://webstandards.org>

⁶ <http://vwkombi.com>

Ian is married to Manda who really doesn't get all this techy stuff—or, for that matter, the Volkswagen fascination—but puts up with it regardless.

About the Technical Director

As Technical Director 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.

About SitePoint

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

*To my son Chris and my
daughter Heidi, who are truly
and forever my sunshine.*

Table of Contents

Preface	ix
1. Let's Get Started ... but How?	1
Planning to Stay on Schedule and Within Budget	2
Effective Pre-planning	2
Doing your Homework and Making Educated Decisions	6
Developing a Web Site Style Guide	9
Managing Like Magic: Effective Organization	12
Working Together as a Team	14
Summary	15
2. What to Find Out: Initial Questions to Answer	17
Establishing your Target Audience	17
Who will Visit and Why?	18
How will they Visit?	19
Identifying your Initial Goals	19
Identifying your Budget	21
Domain Name Costs	21
Web Hosting	22
Branding and Marketing	25
Bringing in the Experts	28
How Much is a Picture Worth?	29
Ecommerce	30
Computer Software and Hardware	30
Summary	32
3. Preparing Web Site Content	33
Gathering and Itemizing Content	33
Providing Accessible Content	36
Preparing Web Copy	37
Preparing Images	39
Summary	43
4. Managing all the Content	45
Assessing your Web Site Content	45
Determining your Content Management Needs	47
Web Authoring Tool Features	49
Content Management System Features	50
Summary	59

5. Web Site Usability: Focusing on the User	61
Creating a User-friendly Index Page	62
Ensuring your Web Site Focuses on Users	64
Providing Fast-loading Web Pages	69
Accommodating User Abilities, Disabilities, Browsers, and Altern- ative Devices	71
Ensuring Predictability	72
Providing User-friendly Corrections for User Errors	75
Summary	78
6. Color	79
Preparing a Color-friendly Work Environment	79
Choosing a Color Scheme	81
Using Color to Enhance Functionality	85
Using Color to Enhance Readability	92
Summary	92
7. Information Architecture	95
Laying the Foundations	96
Improving Findability	97
Organizing for Success	98
Organization Schemes	98
Organizing Content Structure	100
Preparing Web Page Information Architecture	104
Information Architecture for Ecommerce Pages	106
Organizing Directories	109
Using Friendly URIs, URLs, and Filenames	112
Hiding Filename Extensions as Much as Possible	114
Summary	116
8. Navigation	117
Golden Rules for Effective Web Site Navigation	117
Creating User-centered Global Navigation	132
Creating Local or Section Navigation	138
Providing Internal Page Navigation	145
Adding Supplemental Navigation	147
Creating Sitemaps	147
Creating Site Indexes	150
Creating Search	151
Summary	155
9. Best Coding Practice: W3C Standards and Recommendations	157
Magic Markup	158

Separating Content from Presentation	163
Ensuring the Integrity of your Markup	167
Excellent XHTML	169
Spectacular CSS	173
Using Appropriate CSS Naming Conventions	175
Beware of Browser Bugs and Problems	176
Using CSS for Print	177
Summary	181
10. Creating Accessible Web Sites	183
The Bare Bones of Creating an Accessible Web Site	184
Creating Valid, Structured, Semantic Markup	187
Providing Accessible Navigation	192
Color for the Masses	195
A Thousand Words to Each Picture	196
Accessible Image Maps	197
Multi-purpose Multimedia	198
In Good Form	199
Sturdy Tables	202
Summary	206
11. Web Site Optimization	207
Creating Clean, Lean Markup	207
Minimizing URLs	212
Optimizing CSS	213
Optimizing JavaScript	219
Supporting Speedy Server Responses	222
Optimizing Images, Multimedia, and Alternative Formats	223
Summary	225
12. Search Engine Optimization	227
Successful SEO in a Nutshell	228
Working with Keywords and Keyphrases	228
Using Keywords and Keyphrases in your Markup	230
Being Search Engine-Friendly	232
Avoiding Being Banned by Search Engines	236
Getting Listed	238
Creating an Ongoing Links Campaign	241
Planning for Ongoing Maintenance	242
Summary	245
13. Design	247
Basic Design Principles	248

Effective Typography	254
Brainstorming	258
Finalizing the Design	261
Summary	263
14. Testing	265
Getting Started	266
Good Testing Practice	267
General Testing	269
Solid Security	274
Accessibility Testing	276
Conducting a Preliminary Review	276
Conducting a Comprehensive Review	277
Testing with a Graphical Browser	279
Testing with a Voice or Text-only Browser	281
Usability Testing	282
Summary	287
15. Preparing for Launch	289
Conducting Final Checks	289
Checking Templates and Markup	290
Checking Layouts	291
Finalizing Content	291
Checking Functionality	292
Checking Server-side Requirements	293
Conducting a Soft Launch	294
Launching the Web Site	295
Completing your Web Site Handover	296
Checking and Finalizing Project Documentation	296
Creating a Handover Package	297
Archiving Project Documentation and Files	300
Summary	302
16. Post-launch Follow-up	303
Conducting a Post-launch Review	303
Completing Initial Post-launch Tasks	305
Orientating New Staff	307
Getting Attention	308
Managing Maintenance	310
Undertaking Daily Duties	315
Conducting Monthly Minding	316
Performing Periodic Processes	317
Conducting Annual Activities	318

Collecting Data	319
Collecting and Using Quantitative Data	319
Collecting and Using Qualitative Data	321
Summary	323
A. Ecommerce Checklists	325
Assessing Ecommerce Content Usage and Management Needs	325
Features to Seek in a Shopping Cart Program	326
Index	331

Preface

There are literally thousands of books, and millions of articles and tutorials, covering just about every aspect of creating web sites. So, why write another book? Why write this particular book? And why should you buy it?

I've been in the business of designing, developing, and maintaining web sites for an entire decade now. While I've seen plenty of well planned, well executed, and wonderfully maintained web sites and web projects, I've also seen or fixed the problems that result when people ignore, skip, or aren't even aware of important issues, best-practice approaches, or critical steps in the web site development process.

In addition, I've known incredibly talented programmers who can practically program in their sleep, but, for instance, don't know much about the appropriate use of color to help the visually impaired use web sites. I've met plenty of amazingly talented designers who don't know much about best practice approaches to CSS, or how to optimize a web site for search engines. There is indeed a lot to know about creating and maintaining truly effective, efficient, successful web sites that are easy to maintain.

So, as I planned this book, I thought about how helpful it would be to have a single source of solid information on best practices for everything from planning, designing, developing, and testing, to launching a web site and maintaining it in the long term. By compiling all this information into checklists, we've turned a vast wealth of detail into convenient, easy-to-scan, simple-to-use chunks organized by topic. Using the checklists in this book will help you to make sure that you cover all the bases for every web site project you undertake. And, while the checklists draw on my ten years of experience creating and maintaining web sites and managing web projects, I also reference plenty of experts. The references included here are also fabulous resources for learning more about particular topics.

How should you use these checklists? I consider them guidelines, not definite requirements set in stone. At the same time, each individual checklist item is included in this book for a good reason. If you understand a particular checklist item's purpose, you'll be able to make an educated decision about whether or not to include it for a specific web project. To help with that, you'll find reasonably detailed explanations for nearly all of the individual checklists, along with helpful references and real-world examples.

Who Should Read this Book?

This book is ideal for anyone who's responsible for delivering high-quality web sites. If you're a web developer, designer, or project manager, by using these checklists you can be assured that you've delivered a site that follows accepted best practices and is of the highest standards. If you're responsible for commissioning or specifying a web site, you can use these checklists as a basis for your specification, and to ensure that the web site is right the first time. No detailed technical knowledge is assumed or required to use the checklists contained in this book.

What's in this Book?

This book contains 16 chapters that are designed to parallel your progress through a web project's development, launch, and subsequent ongoing maintenance. As such, you can apply any of these chapters as they relate to the work you have at hand, and jump around between chapters and checklists as you need to. On the other hand, for an overall view of the process—perhaps in developing your first site, or first commercial site—the procedural, chapter-by-chapter approach might be the best path to take.

Chapter 1: Let's Get Started ... but How?

So, you're going to develop a web site? Great! ... But where do you begin? This chapter explains the key starting points for any web project, beginning with a checklist to help you start project planning. We also consider a checklist to help you complete valuable research so that you can make educated decisions, and outline the key considerations as you begin to develop a web site style guide. Finally, we provide a checklist of tips to help you work with others in the project team.

Chapter 2: What to Find Out: Initial Questions to Answer

In this chapter, we look at the site- and client-specific questions you'll need to answer as you begin the project. The first checklist will help you define the site's target audiences; the second will help you determine the site's objectives among those users. Next, we look in depth at the budgetary considerations of the project, with checklists that will help you ascertain the costs of hosting, marketing, expert advice, ecommerce, hardware and software, and more.

Chapter 3: Preparing Web Site Content

Sometimes, content can be the most overwhelming aspect of a web project, but this chapter aims to make the job easy! We kick off with a basic checklist for gathering and itemizing content, which should put you in good stead to complete a full content inventory a little later on. We then address the questions of providing accessible web content, before reviewing checklists that will help you prepare web copy and images for use within the site.

Chapter 4: Managing all the Content

This chapter begins with a checklist that explains how to undertake a full content inventory as part of a content assessment. Once you've done so, you'll be able to assess your content management needs, and compare content editing and management tools on a range of important points, using the detailed checklists provided.

Chapter 5: Web Site Usability: Focusing on the User

In focusing on the user, our first step in Chapter 5 is to create a user-friendly index page. From there, we expand our view with checklists designed to ensure that the site focuses on users. These checklists will help you ensure that the site's pages load quickly; that the site caters to users with a range of abilities, disabilities, and access devices; that the site's creators follow established conventions and best practices; and that the site presents user-friendly error messages to visitors.

Chapter 6: Color

It may sound extreme, but the colors you choose can make or break a web site! Before you can really begin to work with color, you need to set up your work area and computer so that colors are displayed accurately—the topic of the first checklist in this chapter. Next, we discuss checklists that will help you choose an effective, attractive, and usable color scheme for the site, and see how we can use color to enhance the site's functionality and readability.

Chapter 7: Information Architecture

This chapter introduces the basic concepts behind information architecture before providing checklists that will guide you in improving the findability of content on your site. Next, we take a close look at the common organization schemes and structures, before exploring in detail the considerations involved in planning the architecture of key page types and directories. The final checklists in this chapter provide advice on creating user- and search engine-friendly URLs for all the files on your site.

Chapter 8: Navigation

Navigation is a critical element of your site's design, so this chapter looks at the topic at its most basic—the Golden Rules of navigation—before delving into the various levels of navigation. Checklists in this chapter explain the details of creating usable and functional global, sectional, in-page, and supplemental navigation systems. We also discuss the intricacies of providing useful site indexes and search tools in separate, highly focused checklists.

Chapter 9: Best Coding Practice: W3C Standards and Recommendations

The separation of content from presentation is an accepted best practice, and key among the foundations on which this chapter is based. The detailed checklists here will help you to create HTML, XHTML, and CSS markup that is standards-compliant, bug-free, and that functions successfully—or degrades gracefully—on as many browsers and alternative devices as possible.

Chapter 10: Creating Accessible Web Sites

Having explored the bare bones of creating an accessible site—including markup and navigation accessibility considerations—first up in this chapter, we move on to look at the ways in which color, images, multimedia files, forms, and tables can be used to enhance—or destroy—a user's experience of your web site. The tips and advice provided in these checklists will help you to hone your site so that most users (if not all!) are able to access the information it contains easily and without hassle.

Chapter 11: Web Site Optimization

Optimizing your web site starts at code level, which is why this chapter's first checklists explore the ways in which you can clean up your markup: by minimizing URLs and optimizing CSS and JavaScript. Later checklists explain the optimization of image, multimedia, and alternative file formats.

Chapter 12: Search Engine Optimization

After a quick primer on SEO success, this chapter jumps straight into checklists that will help you work with keywords and phrases, make your site search engine-friendly, and avoid offending, or being blacklisted by, search engines. Substantial checklists are provided to help you streamline your efforts to list the site in search engines, and create ongoing links and SEO maintenance campaigns that will see your search rankings rise, rather than decline over time.

Chapter 13: Design

Whole books can—and have—been written on web design, but this chapter aims to encapsulate the key considerations in a series of checklists that begin

with basic principles. We then move on to explore the specifics of working with typography, discuss brainstorming techniques, and address that most challenging of tasks: finalizing the site's design!

Chapter 14: Testing

The Getting Started checklist in this chapter will help you establish your site's testing requirements and protocols, and identify the hardware and software that will be needed during the testing phase. After a quick review of the constituents of good testing practice, the chapter steps through checklists that explain general testing procedures, as well as guidelines for accessibility and usability testing. By the chapter's end, you'll have the know-how and confidence to complete the tests and ensure that your site is ready to launch!

Chapter 15: Preparing for Launch

These checklists will prove invaluable to you time and time again. Pre-launch can be a stressful time, and the checklists in this chapter have been designed to take the pressure off and let you focus on the bigger issues in those final days before the site is launched to the world. Final checks, including templates, markup, layouts, content, functionality, and server-side requirements are covered, before we discuss the processes of soft launch, launch, and site handover.

Chapter 16: Post-launch Follow-up

A post-launch review is a great way to ensure that the site launch has been completed, the client is happy, and the project team has learned all they can from the process. Checklists are provided to help you conduct the review and complete those niggling post-launch tasks before you tackle the bigger jobs—orienting new staff to the site, and promoting the site—for which detailed checklists are also provided. Finally, this chapter breaks down the responsibilities involved in managing and completing ongoing maintenance tasks into a series of checklists that will help save your sanity in the months to come!

Appendix A: Ecommerce Checklists

This appendix, provided specially for those working with ecommerce projects, contains two compact but valuable additional checklists. The first will help you to assess your needs for ecommerce content usage and management, while the second highlights the kinds of features you should seek in an ecommerce package. Together, these checklists should help you to assess the various tools on the market and choose those that will meet the specific needs of the project you have at hand.

This Book's Web Site

Located at <http://www.sitepoint.com/books/checklists1/>, the web site supporting this book will give you access to the following facilities.

Updates and Errata

The Corrections and Typos page on the book's web site, at <http://www.sitepoint.com/books/checklists1/errata.php>, will always have the latest information about known typographical and code errors, and necessary updates for changes to technologies.

The SitePoint Forums

While I've made every attempt to anticipate any questions you may have, and to answer them in this book, there is no way that any book could cover everything there is to know. If you have a question about anything in this book, the best place to go for a quick answer is <http://www.sitepoint.com/forums/>—SitePoint's vibrant and knowledgeable community.

The SitePoint Newsletters

In addition to books like this one, SitePoint offers free email newsletters. The *SitePoint Tech Times* covers the latest news, product releases, trends, tips, and techniques for all technical aspects of web development. The long-running *SitePoint Tribune* is a biweekly digest of the business and moneymaking aspects of the Web. Whether you're a freelance developer looking for tips to score that dream contract, or a marketing major striving to keep abreast of changes to the major search engines, this is the newsletter for you. The *SitePoint Design View* is a monthly compilation of the best in web design. From new CSS layout methods to subtle Photoshop techniques, SitePoint's chief designer shares his years of experience in its pages. Browse the archives or sign up to any of SitePoint's free newsletters at <http://www.sitepoint.com/newsletter/>.

Your Feedback

If you can't find your answer through the forums, or you wish to contact me for any other reason, the best place to write is books@sitepoint.com. We have a well-manned email support system set up to track your inquiries, and if our

well-manned email support system set up to track your inquiries, and if our support staff is unable to answer your question, they send it straight to me. Suggestions for improvement as well as notices of any mistakes you may find are especially welcome.

Acknowledgements

I wish to thank everyone at SitePoint for making this book possible, especially Priya Singh, the technical editor, and Georgina Laidlaw. Special thanks go to Simon Mackie, my tireless, incredibly helpful, and exceptionally kind editor, who was always full of encouragement and provided invaluable insight throughout this entire process. I agreed to write this book because Simon would be my editor.

In addition, big thanks go to Ian Lloyd, my expert reviewer, who packed his helpful advice with humor that made me laugh throughout the comments he made on every single chapter.

Thanks also to Dan Thies and Jill Whalen for so kindly taking the time to answer my SEO questions for the checklists on SEO.

I also appreciate all the support and understanding I received from my family and friends throughout this book's development, especially my daughter, Heidi. I couldn't have written this book without Heidi taking care of so much for me while I lost myself to writing at my computer.

Lastly, I couldn't have written this book without my two cocker spaniels nudging me to take a break and go play when I needed it the most, and keeping me company during all those long hours of writing when everyone else was sound asleep.

6

Color

I think most of us would agree that the world would be a pretty boring place without color—as would your web site! Color is a primary building block of visual design that enhances your site’s aesthetic qualities, and allows it to have a lasting impression on visitors. Color can also boost the usability of your site, drawing visitors’ attention to certain areas of a web page, improving readability, enhancing raw functionality, and highlighting the visual organization of content, pages, and the site as a whole.

The effective use of color can make a valuable difference in your site’s success. This chapter provides checklists that will help you to provide a positive first impression for visitors, enhance your site’s functionality, and use the most appropriate colors for accurate display on a wide range of computer monitors and other devices.

Preparing a Color-friendly Work Environment

To avoid unwelcome, ugly surprises as you apply colors to your site design, it’s important to keep the colors you use as accurate as possible—especially if you’re preparing images for use on your web site. To create accurate colors for your web

site, you'll need to adjust your hardware and software. This checklist will help you to view and work with colors accurately.

Checklist 6.1

Position your monitors in areas with subdued light, out of strong or direct light.

Use color-neutral, daylight-balanced lighting in your work area.

OTT-LITE Technology¹ is one well-known provider of suitable lighting.

Set your displays to 24-bit or 32-bit color (True Color).

Windows users can access the appropriate options by clicking Start > Control Panel > Display. Then, in the Display Properties window, click the Settings tab. Macintosh users need to click System Settings > Display.

Set your display contrast to its maximum unless the image display is too bright or harsh on this setting.

Set your display gamma to 2.2 (Windows users) or 1.8 (Macintosh users).²

Calibrate your displays to the sRGB ICC profile.

Especially important, in this regard, are computer displays that you use to work with and produce images for your web site. You've probably already noticed that a web site's colors can look quite different on various computer displays—maybe you've even had clients tell you the colors you've used on their web sites are “all wrong,” even though you've employed their corporate colors. There are millions of computer displays in the world, and they can employ a wide range of settings, including color space—the range of color that the display is capable of reproducing. No two devices are exactly alike. The goal of the ICC (International Color Consortium)³ is to standardize vendor-neutral, cross-platform, device-independent color profiles.

To manage the colors in web site images, use the sRGB color profile. Although, increasingly, the sRGB color profile is being used as the default

¹ <http://ottlite.com>

² For more on display gamma, see Norman Koren's article, *Making Fine Prints in your Digital Darkroom: Monitor Calibration and Gamma*

[<http://www.normankoren.com/makingfineprints1A.html#Monitorsetup>].

³ <http://www.color.org/>

display, millions of existing displays still use other profiles. Nonetheless, using the sRGB color profile will help you ensure that the colors in your images display as close to the intended colors as possible.



Using Visual Calibration Tools

Visual calibration is offered within Adobe products, such as Adobe Gamma for Windows, and Monitor Calibrator for Macintosh. To access this functionality in Windows, click Edit on the menu bar, then click Color Settings. Macintosh users must click Photoshop on the menu bar, then click Color Settings. Then, in the Working Spaces box, select sRGB IE61966-2.1. For higher levels of accuracy, try calibration software such as X-rite,⁴ Color Savvy,⁵ Alwan Color Expertise,⁶ or Gretag-Macbeth.⁷

Choosing a Color Scheme

Color matters to users, whether they know it or not. In fact, a usability study found that “the best predictor for the overall judgment by typical users of a web site was its beauty.”⁸ Harmonious colors engage web site visitors and provide an inner sense of order, resulting in a peaceful visual experience that’s less likely to intimidate or frustrate them. The use of triadic color schemes—three colors the same distance apart on the color wheel—achieves this inner sense of order the best.⁹

The key to choosing a harmonious color scheme for your web site lies in your branding: choose a color scheme that works well with your existing logo and company colors. The following checklist will help you to choose the right colors for your web site.

⁴ <http://www.xrite.com/>

⁵ <http://www.colorsavvy.com/>

⁶ <http://www.alwancolor.com/>

⁷ <http://www.gretagmacbethstore.com/>

⁸ Laurie Brady and Christine Phillips, *Aesthetics and Usability: A Look at Color and Balance*, Usability News (May 1, 2003) [<http://psychology.wichita.edu/sur/usabilitynews/51/aesthetics.htm>].

⁹ Leatrice Eiseman, *Pantone Guide to Communicating with Color* (Cincinnati: Grafex Press, 2000), 11; Vader, Beverly, *Choosing a Color Palette*, Encyclopedia of Educational Technology (no date) [<http://coe.sdsu.edu/eet/Articles/colorwheel/start.htm>].

✓ **Base your web site's color scheme on your branding and logo colors.**

Color is an important aspect in building recognition of your brand. Stay consistent by basing your web site's design on your branding, including the colors used in your logo.



Getting to Know the Color Triads

Triadic colors can be bright and strong, such as the primary triad of red, yellow, and blue, and the secondary triad colors of orange, purple, and green. For more subtle hues, lighten, darken, or add gray to triadic colors to create an endless variety of interesting, eye-pleasing color combinations.

To help you choose attractive colors, use a color wheel, such as the Primary color wheel (mixed from red, yellow, and blue), the Secondary color wheel (mixed from orange, green, and violet), the Tertiary color wheel (one primary and one secondary color mixed together), or the Visual color wheel (mixed from red, green, and blue). The Visual color wheel's RGB primaries are used for computer monitors, cameras, scanners, and other digital technologies, and the secondary (subtractive) triad is CMY (cyan, magenta, yellow), a printing standard.

Add color wheels, such as those from The Color Wheel Company,¹⁰ to your bookmarks—they're very handy reference tools. Make use of other color palette and color scheme tools as you choose web site color schemes; try VisiBone's Color References,¹¹ Dominique Trapp's Color Harmony Guide,¹² and Color Wheel Pro.¹³ For helpful software and more online color tools, visit WebsiteTips.com's Color section.¹⁴

To read more about using color on your web site, try the excellent book by Mimi Cooper and Arlene Matthews, *Color Smart: How to Use Color to Enhance Your Business and Personal Life* (New York: Pocket Books, 2000), and Leatrice Eiseman's *Pantone Guide to Communicating with Color* (Cincinnati: Grafex Press, 2000).

¹⁰ <http://colorwheelco.com>

¹¹ <http://visibone.com/color/>

¹² <http://www.ideabook.com/color.htm>

¹³ <http://color-wheel-pro.com>

¹⁴ <http://websitetips.com/color/>

- ✓ **Use a neutral, subtle color scheme for your web site to help direct the focus to your products or services.**

Neutral doesn't have to mean dull and boring, of course! But, if you're going to get users to focus on your products or services, your web site colors must remain fairly neutral. For example, if you visit an art gallery, you'll notice that the walls and interior colors are neutral colors. This allows visitors to focus on the art—the surrounding colors don't distract visitors from the works themselves.

Branding and logo colors may intentionally be strong, as this is a great way to grab attention. If you're working with a strong palette like this, you might choose to use paler shades of the branding and logo colors to create a more neutral color scheme for the site, reserving the strong colors for use in small accents here and there. One example of this approach in action is the color scheme at Amazon.com. The Amazon logo uses black text and a light-orange color on a white background. The color scheme doesn't use the orange prominently, but shades of it are applied throughout the web site as accents: a darker shade of this orange is used for heading text and arrows, and to create a light-orange background bar at the bottom of each page. Amazon.com's global navigation tabs are a neutral, subtle color.

- ✓ **Use color to convey meaning.**

Some brands define themselves, at least in part, by color—look at Coca-Cola (red and white) or McDonald's (red and yellow), for example. When you plan your web site's color scheme, don't leave your choices to chance. Understand what colors mean, and use them effectively, keeping in mind the cultural differences mentioned next.

- ✓ **Consider the different meanings associated with colors in particular cultures.**

Each region of the world has its own predominant colors, distinctive palettes, and color associations. For example, in the United States, red is often associated with power, love, passion, and rage. By comparison, in China, red is the color of weddings, and red, pink, and burgundy are associated with good luck, good health, and prosperity.¹⁵ As you choose your web site colors, be sure to use colors that are appropriate for the cultures and countries in which your target markets or audiences live.

¹⁵ Mimi Cooper with Arlene Matthews, *Color Smart: How to Use Color to Enhance Your Business and Personal Life* (New York: Pocket Books, 2000), 117.



Discovering the True Meanings of Color

Three helpful books that detail the cross-cultural meanings of colors are Leatrice Eiseman's *Pantone Guide to Communicating with Color* (Cincinnati: Graftix Press, 2000), her new book, *The Color Answer Book: 100+ Frequently Asked Color Questions for Home, Health and Happiness* (Herndon, Virginia: Capital Books, Inc., 2003), and Mimi Cooper and Arlene Matthews' *Color Smart: How to Use Color to Enhance Your Business and Personal Life* (New York: Pocket Books, 2000).

- ✓ **Use no more than six colors in your web site color scheme, if possible.**

Studies show that using more than six colors can slow down the time users take to search through information. While other studies suggest that as many as nine colors may be acceptable, provided those colors are separated within the color space,¹⁶ I think it's a good idea to remember that old cliché—"better safe than sorry"—and to create color schemes that use a maximum of six colors.

- ✓ **Avoid placing blocks of saturated, complementary colors next to one another.**

When you place next to each other saturated complementary colors, such as red and blue-green, yellow and dark blue, or purple and yellow-green, you'll see a flickering effect. This effect is actually generated as your eyes' focus shifts quickly back and forth, as it's unable to settle on either complementary color. Not surprisingly, this effect causes eye strain!¹⁷

- ✓ **Test for color differences, inconsistencies, and problems using a variety of displays, resolutions, and platforms.**

While your web site colors will look different on nearly every display, it's important to check for unsightly problems. By testing on a variety of displays and resolutions, you can see if your web site colors shift in undesirable ways, and then adjust your images, your CSS, or both, to eradicate any problems. For example, you may have a logo image that blends seamlessly

¹⁶ H. Smallman and R. Boynton, *Segregation of Basic Colors in an Information Display*, *Journal of the Optical Society of America A*, 10, 1985–1994 (1990); U. Derefeldt and H. Marmolin, *Search Time: Color Coding and Color Code Size*, (Paper presented at the First Annual Conference on Human Decision Making and Manual Control, 1981.)

¹⁷ See Victor R. Stanwick's article, *Avoiding Problems with Adjacent Colors* [<http://www.fast-consulting.com/color/coloravoidingproblems.htm>].

against your site header's background color on one monitor; however, you may find the two elements appear in different shades on another display.

New flat-panel LCD (liquid crystal display) monitors tend to be brighter and crisper than CRT (cathode ray tube) displays and older LCD displays; images generally appear lighter on Macintosh computers than on Windows PCs, due to the different gamma settings used. Lower-cost laptops are inclined to sacrifice accuracy in color reproduction.¹⁸ Color displays on PDAs and cell phones are even less predictable, and if your web site is targeting alternative device users, keep in mind that plenty of older PDAs with black-and-white displays are still in use.

Using Color to Enhance Functionality

As well as creating a serene browsing experience for your visitors, color can be used to enhance the functionality of your web site. The following checklist provides pointers on how best to use color to improve the functional aspects of your site.

Checklist 6.3

Use color to direct user focus.

Use color to focus user attention on certain sections on a page. For instance, you might use red to highlight a sale item, or apply a yellow NEW! icon to attract attention to new articles, products, or services. Yellow is the brightest and most reflective of all the primary colors, while red causes the most active physiological responses. While yellow and red command attention, it's best use them judiciously—too much of these bright colors can be exhausting and distracting.¹⁹ Brighter colors stand out from darker colors, warm colors stand out more from a cool background,²⁰ and saturated colors (intense, pure colors to which no gray has been added) stand out more than their less-saturated counterparts.

The NEW! icon in Figure 6.1 shows how we can draw attention to new items in stock through the use of color and bold text—even in this grayscale example. The shaded background and bold text attracts our focus.

¹⁸ For more on this, see Joe Gillespie's article, *Paper vs. Pixels—Part 2* [<http://www.wpdafd.com/editorial/wpd0904news.htm#feature2>].

¹⁹ Cooper and Matthews, 25, 39, 50; Eiseman, 19.

²⁰ Color wheels visually illustrate color temperature, i.e., warm vs cool colors. Warm colors include red-violet, red, red-orange, orange, yellow-orange, and yellow. Cooler colors are yellow-green, green, blue-green, blue, and blue-violet.

Figure 6.1. Using color to direct user focus

- **NEW!** Version 8 of our graphics software now available!
- **NEW!** Version 2.1 Update of our print software now available
- Same old stuff here, though
- More of our same old stuff here, too

This NEW! icon is achievable with a mixture of text and CSS, which eliminates the need to create a graphic, and helps to minimize page load times and bandwidth usage.

Use colors consistently to avoid user confusion.

Use colors consistently throughout your site. For example, if you choose dark red for your main headings and dark blue for article titles, don't decide to use dark blue for main headings in a particular section, and some other color for the titles of articles displayed there. Consistency reduces confusion. Perhaps your color scheme uses a different color for each section—blue for electronics products, and green for software, for example. That's fine. If, in the blue section, you use navy blue for the main headings, and in the green section, you use green for the main headings, don't diverge from this standard. Stay consistent within the color scheme you've created.

See Figure 6.3 through Figure 6.5 below for examples of color schemes in action. And don't forget that your site will be easier to manage and maintain if you use CSS to designate the colors displayed on your pages.

Use color to distinguish hyperlinks from regular text.

Designate a color for styling hyperlinks that makes them stand out from surrounding text. Make sure you underline hyperlinks as well, especially within your text content, so that you're not solely relying on users to distinguish the color in order to identify the links. I sometimes style hyperlinks bold, to make them even more obvious.

Avoid using purple to style unvisited hyperlinks.

✓ **Avoid using blue to style visited hyperlinks.**

Browser defaults use blue for unvisited links and purple for visited links, so if you switch these colors, it's very likely that you'll confuse your site visitors.

✓ **Where possible, use CSS, instead of graphics, to create web site colors.**

You can use CSS for a multitude of web site color purposes. Using CSS instead of graphics reduces page load times and bandwidth use, and results in a web site that's infinitely easier to maintain than one created without CSS. For example, if you use CSS to create borders around photographs, instead of adding those borders to the actual photographs themselves, you'll be able to change border colors and styles for photographs site-wide, simply by changing the CSS rules for the photograph borders. As we saw above, Figure 6.1 depicts a NEW! icon created using CSS instead of a graphic, saving bandwidth and page load time. Figure 6.2 uses CSS to add color and style—and a drop cap—to the headings on a page.

Figure 6.2. Using CSS to apply colors and style to headings

M a i n H e a d i n g

Article Title Here

THIS IS THE FIRST LINE AND THE FIRST paragraph. Lorem ipsum dolor sit amet, consetetur sadipscing elitr. Sed diam nonumy eirmod tempor invidunt ut labore. Et dolore magna aliquyam erat, sed diam voluptua.

At vero eos et accusam et justo duo dolores et ea rebum. Stet dita kasd gubergren, no sea takimata sanctus est Lorem ipsum dolor sit amet.

You can also use CSS to create design accents, to color particular web site functions, and to designate media-appropriate colors for specific media.²¹ For additional checklists and tips on CSS and web site design, see Chapter 9 and Chapter 13.



Perform a “wireframe” test to ensure that you’re catering for various user preferences and disabilities.

Users can customize their browser settings to turn off colors, alter font sizes, and select fonts; they can also choose to use their own style sheets. Rather than being alarmed by this fact, embrace this user-friendly flexibility by ensuring that your web site design makes sense in black-and-white—especially its navigation and other important functions. Test your design without color to make sure it’s still user-friendly and works well.

Many graphics programs offer a grayscale feature, which makes checking your design layouts in grayscale easy to do. If you need to, simply take a screenshot of the web page in question, then open the screenshot image in your graphics program. Alternatively, in Photoshop CS, select Image > Mode > Grayscale from the menu bar to access this functionality. If you’re using your design files, create a duplicate copy first!

On Mac OS X, you can switch your entire display to grayscale: click on System Preferences > Universal Access > Set Display to Grayscale. This feature makes it easy to check design layouts and test web site functions in grayscale.

Windows users can test web pages in grayscale using a free online testing tool called the Colorblind Web Page Filter.²² Also, check to see if your particular display has grayscale settings.



Designing in Grayscale

Try creating your initial web site design and page layouts in grayscale. Graphic designers learn in school to create initial designs and layouts in grayscale because it’s important that the designs have impact even without color. Taking this approach for your site design and page layouts will help ensure that your web site functions, and that visual

²¹ For more on the topic, see Elika Etemad’s and Jorunn D. Newth’s article, *Pocket-Sized Design: Taking Your web site to the Small Screen* [<http://www.alistapart.com/articles/pocket/>], and the W3C’s *Media Types in Cascading Style Sheets* [<http://www.w3.org/TR/CSS21/media.html>], part of Level 2 Revision 1 in the CSS 2.1 Specification.

²² <http://colorfilter.wickline.org/>

cues won't depend solely on color in order to make sense.²³ Designs that work well in grayscale are solid—though color will enhance them, they won't depend on color in order to succeed.

If you've created wireframes to plan your site's information architecture, it's a good idea to use those wireframes to create your web site design and page layouts. For more on information architecture, wireframes, and design, see Chapter 7 and Chapter 13.

 **Use color to emphasize the organization of information and provide visual cues.**

When you use color to organize information, you have the opportunity to give your web site visitors subtle cues that make your web site easier to navigate and your pages quickly comprehensible. For example, the major sections of some web sites are organized by color. Amazon.com uses green for its Books section, blue for Electronics, and dark red and light-orange for the Today's Deals area. Amazon.com changes some of these section colors on its local sites to suit the cultural color preferences of those sites' specific users.

²³ For more on creating grayscale designs, see Heather Quinn's article, *Where / What Vision Systems and Visual Design* [<http://www.evolt.org/article/rating/22/60261/>] and the DevX article, *Design with Color* [<http://www.devx.com/projectcool/Article/19987>].

Figure 6.3. Using color to create visual cues



Graphics Software

Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua.

Figure 6.3 shows how colors have been used as visual navigation cues on the MyByz site. You can use color as a visual cue for headings—try using different colors for headings and subheadings, as well as differentiating them using the font sizes and page layout, as shown in Figure 6.4.

Figure 6.4. Headings: creating visual cues using color and size

Heading, Midnight Blue

Article Title, Medium Teal

Text is black. Background is white to provide contrast for better legibility. Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua.

In Figure 6.5, the main heading uses a different font from that used in the article title heading, providing an additional visual cue.

Figure 6.5. Headings: creating visual cues with color, size, and font

Heading, Midnight Blue

Article Title, Medium Teal, and
Different Font

Text is black. Background is white to provide contrast for better legibility. Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua.

These text effects are easily achieved using CSS.

Don't rely solely on color to communicate important visual cues.

Your use of color in the organization of information should enhance the user experience; the user experience should not depend upon it. When your web site doesn't solely rely on color to communicate important visual cues, your web site will be more device-independent, and therefore more usable. This approach also ensures that visually impaired site visitors can use your web site easily.

As an example, consider Amazon.com's navigation. It doesn't depend solely on color in order to be usable or understandable; nor does the navigation system for the fictional MyByz.com site shown in Figure 6.3. The navigation makes sense in grayscale, which means that it will be accessible by those with color blindness and other visual disabilities.

Using Color to Enhance Readability

Checklist 6.4

For best legibility, use colors that provide a strong contrast between the page background and the text.

Use solid, light-colored backgrounds behind dark-colored text for page content areas.

Black text against a solid white background provides the best contrast and is typically the easiest text to read.

Avoid displaying patterns, textures, and colorful backgrounds behind text.

Summary

This chapter identified the important considerations affecting the best use of color on your web site. It explained how to set up a color-friendly work environment, provided guidelines to help you choose a color scheme for your web site, and showed how best to use that color scheme to enhance the functionality of your web site. I also included references to color wheels and other helpful tools that will prove very valuable as you work with color on your web site.

Well, that's the fun stuff taken care of! Let's move on to what are generally seen to be the more "serious" details (though I think they sound more serious than they really are!), starting with information architecture.

12

Search Engine Optimization

Optimizing your web site for search engines should be an integral part of your web site project, from the very beginning to the very end. Search engine optimization (SEO) should be considered, and if possible, implemented, throughout the planning, design, development, and maintenance stages of your web site.

The checklists in this chapter can be used as a guide to optimize your web site for search engines. However, it will definitely pay you to subscribe to some helpful SEO newsletters, visit web sites that cover the latest on SEO, and consider purchasing books by highly regarded SEO/SEM experts, such as *The Search Engine Marketing Kit*, by Dan Thies.¹

You will *see* the results of your work first-hand if you consider SEO right from the start. For example, at the time of this book's publication, my own site is the number one listing on Google for its keywords, and has consistently remained within the top three (unsponsored) listings on the first page of Google and other search engines for several years. Yes, SEO is important, but more than that, it's rewarding! Let's get started.

¹ Dan Thies, *The Search Engine Marketing Kit* (Melbourne: SitePoint Pty. Ltd., 2005) [<http://www.sitepoint.com/books/sem1/>] [<http://www.sitepoint.com/books/sem1/>].

Successful SEO in a Nutshell

Checklist 12.1

- ✓ **Plan your web site's SEO before you create the site.**

It's important to create your web site with SEO in mind. Changing a web site, or potentially even redesigning your web site, to optimize it for search engines can end up being an expensive proposition. Plan to create a search engine-friendly web site from the start.

- ✓ **Ensure that every page includes text, links and popularity components² that will help boost your search engine rankings.**

- ✓ **Employ other means to market your web site.**

In addition to SEO, consider paid submission programs, PPC (pay-per-click) advertising, reciprocal links campaigns, and advertising your site on others. Depending on your budget, you might consider advertising through traditional media, such as newspapers, magazines, radio, and television.

Working with Keywords and Keyphrases

Checklist 12.2

- ✓ **Research keywords and keyphrases (multi-word phrases) before you create your web site.**

It's important to create your web site with keywords and keyphrases in mind *and* to put them in place as you develop the pages and content.

- ✓ **Target keyphrases rather than single keywords.**

Single keywords are much more commonly entered by web users, which makes them more difficult to target effectively than multi-word keyphrases. Unless the single keywords are highly unique, your best results will be achieved using keyphrases.

² Shari Thurow, [*Before You Build*] in *Search Engine Visibility* (Indianapolis: New Riders Publishing, 2002), 20.

✓ **Use longer words and plurals.**

If you target plural versions of your keywords or phrases, you'll get hits from people searching for the singular and plural versions of those words.



Use that Thesaurus!

As part of your brainstorming for appropriate keywords and keyphrases, try using a thesaurus to find similar words. There are helpful thesauri online (you might start at [Thesaurus.com](http://thesaurus.com)³), but see also the [Search Term Suggestion Tool](#)⁴, and [Wordtracker](#).⁵

✓ **Focus on a few specific keyphrases for each web page.**

Pages that rank well in search engines tend to focus on specific keyphrases that usually appear in the HTML `title` element, heading elements, breadcrumb navigation links, product names and descriptions, and cross-links.⁶

✓ **Create keyphrase-rich text content, especially for your homepage.**

Identify the top two or three keyword phrases that potential visitors would use to find your web site, then write 200–250 words of homepage text that utilizes those keyword phrases—not the other way around. Follow this approach as you create content for other web pages, too. As noted above, keep in mind that titles and headings are considered more important than other content by some search engines, so consider this as you optimize your content. We'll discuss the topic of optimizing your markup for search engines in just a moment.

✓ **Include keyword and keyphrase research in your ongoing web site SEO maintenance plan.**

³ <http://thesaurus.com>

⁴ <http://inventory.overture.com/d/searchinventory/suggestion>

⁵ <http://wordtracker.com>

⁶ Shari Thurow, *Top Five SEO Design Mistakes*, ClickZ (September 27, 2004) [<http://www.clickz.com/experts/search/results/article.php/3412591>].

Using Keywords and Keyphrases in your Markup

Search engines use a page's structural markup as a guide to rank the relative importance of its content. It's important to include keywords and keyphrases within your web pages—especially your homepage—and to place them within certain markup elements. Providing content that is relevant to your target audience will, naturally, help your search engine rankings, especially if you intentionally make strategic use of appropriate keywords and keyphrases. Use the checklist below to help.

Checklist 12.3

- ✓ **Include keywords and keyphrases in your site's information architecture.**

Keyword and keyphrase research results can play an important role in the words and phrases you use within your web site's information architecture. For instance, you might use keywords and keyphrases in your global and local navigation, your category labels, page `title` element text, heading element text, and internal links.

Using these terms in conjunction with the checklists from Chapter 5 and Chapter 7 not only helps your web site visitors find information more readily—it helps your site achieve good search engine ranking, too!

- ✓ **Use focused keyphrases within each web page's `title` element.**

Currently, creating keyphrase-rich text for your web page `title` element is critically important, because nearly all search engines give the `title` element's text a lot of weight. Create each page's `title` element text to reflect the specific content of that page, using keyphrases that people might type into search engines to find your web site.

- ✓ **Use your keyphrases within each web page's `description meta` element.**

Note that many search engines, including Google, Yahoo!, Inktomi, and others, index and use the text within the `<meta name="description">` tag; for instance, some use it as description text for display in search results. However, they don't use the text for search engine rankings at this point. Also, they largely ignore other `meta` elements, including the `keyword meta` element. While there's no guarantee that it will help improve your rankings,

it's still worthwhile to include the `description` meta element in your web pages as shown below.⁷

```
<head>
:
  <meta name="description"
        content="Add your descriptive sentence or two here." />
:
</head>
```

- ✓ Use your focused keyphrases within the page's heading elements, beginning with the `h1` element.



Tip

Keep h1 Close to body

The closer that you can locate your keyphrase-loaded `h1` element content to the opening `<body>` tag, the better. Some search engines will give a close proximity between the two a higher rating—it appears to the search bot that the content is important enough to earn itself a high-level heading, and that content appears early in the document. In addition, content that's placed closer to the top of the page is also considered more important by many search engines, so make sure your top 200–250 words are packed with keyphrase-rich content.

- ✓ Use your focused keyphrases within link URLs and corresponding **title** attributes.

Here's an example:

```
<a href="/design/checklists/"
    title="Deliver First Class Web Sites">Deliver
  First Class Web Sites: 101 Essential Checklists</a>
```



note

Keywords and Domain Names

Although some Search Engine Marketers believe that placing keywords in domain names can provide a significant boost to sites, others feel it doesn't really do much. According to SEO expert Jill Whalen, keywords in domain names might be afforded a small amount of weight by the search engines, but only when other site owners link to the web site using its domain name, rather than the web site title. The result is similar to having keyword- or keyphrase-rich links, which

⁷ Jill Whalen, *The Meta Description Tag*, High Rankings (updated October, 2004) [http://www.highrankings.com/metadescription.htm].

is indeed helpful. Another SEO expert, Shari Thurow, also feels that other factors have much more impact than the inclusion of keywords in a domain name, such as keyphrase-rich content, links, and the site's popularity.⁸

- ✓ Use relevant keyphrases within your `img` elements' `alt` attributes.

Here's an example:

```

```

Being Search Engine-Friendly

Great keywords, keyphrases, terrific content, and having other web sites linking to yours does nothing for your search ranking if the search engines can't access your web pages to index them. Use this checklist to help ensure that your web site is based on search crawler-friendly design and markup.

Checklist 12.4

- ✓ Avoid the use of splash pages.

A splash page is a main entry page that displays either a large graphic image or a Flash animation, usually with a link to Enter a web site or Skip Intro (skip the animated introduction page). Splash pages usually redirect to a new web page after the animation has completed. As you might expect, splash pages typically lack keyword- or keyphrase-rich content, as they contain little or no visible body text other than Enter or Skip Intro links. Given little or no text content, the search crawlers have nothing to index.

Typically, splash pages use redirects to automatically advance the user to the web site's actual homepage. Currently, search engines tend not to index web sites that use redirects, and they'll ban web sites that create artificial redirects in an attempt to achieve higher rankings.⁹

⁸ Jill Whalen, *Search Engine Marketing Q&A*, High Rankings Advisor, Issue 091 (March 24, 2004) [<http://www.highrankings.com/issue091.htm>]; Shari Thurow, *Search Engine Visibility* (Indianapolis: New Riders Publishers, 2002), 20.

⁹ Shari Thurow, *Top Five SEO Design Mistakes*, ClickZ (September 27, 2004) [<http://www.clickz.com/experts/search/results/article.php/3412591>].

So, by using a splash page that contains little or no text content, and uses redirects, you'll likely have ruined your chances of having your web site indexed—let alone ranked—by search engines.

✓ **Balance HTML text and graphics.**

Professional, high-quality graphics can help support your credibility, as we discussed in Chapter 3. It's also critically important to provide plenty of keyword- and keyphrase-rich text content for search engine crawlers. There's no need to avoid using graphics; in fact, avoiding graphics could be detrimental to your web site's search engine success. Instead, plan your web site design to use graphics that have a purpose, insert meaningful text within each `img` element's `alt` attribute, and provide plenty of keyphrase-rich content that will help your search engine rankings, and benefit site visitors who read them.

✓ **Create search-engine friendly HTML text navigation and cross-links.**

Search engine crawlers follow HTML text links. If you provide good, search engine-friendly internal HTML text links within your site, crawlers will likely follow those links.

✓ **If any of your web pages' URLs change, use server-side redirects to send search bots using the old link to the new URL.**

Instead of allowing search engine crawlers (or web site visitors!) to find dead links, create server-side redirects, perhaps using `.htaccess` and `mod_rewrite` techniques, that will automatically take search engines to the new URLs. On the server side, use a 301 code for permanently moved files and a 302 code for temporarily moved files (assuming that your host allows you access to such settings; many do not).

If you're not able to use server-side redirects, create a web page for each changed URL to inform search engine crawlers and visitors that the web page has moved temporarily or permanently, and provide a link to the new page. If there is no new URL, then be sure to provide a link to your homepage, sitemap, and local search to help your visitors and search engine crawlers (this is especially important in ensuring that the crawlers continue indexing your web site via these links). Don't use JavaScript redirects on these pages!

note

Rules for Using Static Redirect Pages

If you're forced to use static pages to inform people and search engines that a page has moved, be sure to have an inventory—a spreadsheet, perhaps—that you can refer to six to 12 months later. This should be enough time for the search engine crawlers to index the new locations, at which point you can check your search engine listings and consider removing these old holding files if the search engines are listing the new pages. You might consider leaving these static pages longer, though, especially for use by visitors who've bookmarked your moved pages.¹⁰

- ✓ **Use external CSS and JavaScript whenever possible, to maximize relevance.**

Search engines will read your markup, beginning at the top of your web page, but they'll also review embedded CSS and JavaScript within head elements. Generally, search engines tend to give more weight to the text that appears at the top of web pages than to text published further down the page.¹¹

- ✓ **Use structural markup, and separate content from presentation as much as possible.**

Search engine crawlers use structural markup to understand what your web page content is about, and they give more weight to text within heading elements (h1, h2, h3, etc.).



Content Stuffers will be Penalized!

Don't try to fool search crawlers by stuffing content into heading elements or elsewhere, as they know about such trickery, and you could be banned from the search engine listings for trying to artificially improve your rankings. Honesty is always the best policy. Creating keyphrase-rich content that visitors will love to read is the best way to go. Other web sites will also link to it, which only helps boost your search engine rankings further.

¹⁰ Ian Lloyd, *The Perfect 404*, A List Apart (January 16, 2004) [<http://www.alistapart.com/articles/perfect404/>].

¹¹ Danny Sullivan, *Search Engine Placement Tips*, Search Engine Watch (October 14, 2002) [<http://searchenginewatch.com/webmasters/article.php/2168021>]; Danny Sullivan, *Hiding JavaScript*, Search Engine Watch (June 10, 2003) [http://searchenginewatch.com/_subscribers/more/article.php/2153181] (subscribers-only area).

- ✓ **Provide a web site sitemap, and link to it from your homepage as well as all your other web pages.**

Providing a sitemap that links to your main pages will help search engines find your content, as well as being helpful to web site visitors. If your sitemap contains more than about one hundred links, though, break it logically into pages of no more than one hundred links each.

- ✓ **If you use JavaScript links, include fallback href attribute links within the <a> tags.**

For example, at my web site, I use DHTML for a show/hide toggle menu, which includes a crawler-friendly JavaScript link with an a element with an href attribute that looks like this:

```
<a href="/inc/arch.shtml" onclick="return toggleMenu('arch');
return false;" onkeypress="return toggleMenu('arch');
return false;">Toggle Show/Hide</a>
```

Here, href="/inc/arch.shtml" provides an alternative to the JavaScript event handlers. Since it's an HTML text link, search engine crawlers can read it and continue on to index the /inc/arch.shtml page. I had accessibility in mind when I decided on this particular approach, but it helps search engine crawlers, too.



JavaScript and Popups

Popup windows are another popular way to use JavaScript links. Ian Lloyd of Accessify.com fame wrote an excellent article for SitePoint, titled, *The Perfect Pop-Up*.¹² Check it out for a helpful approach that's accessibility-friendly and search-crawler friendly, too.

- ✓ **Avoid dynamic URLs that contain ?, &, \$, =, +, and % characters, cgi-bin, session IDs, or cookies.**

These URLs are usually the result of query strings on dynamic pages. Some search engine crawlers are wary of such URLs because they could potentially be infinite links that would overload the crawler. As a result, they may not index pages that use these URLs, especially when three or more of these characters are included in the address. As Eytan Seidman, MSN Search Program Manager, states, "The algorithm starts to wonder whether it is

¹² <http://www.sitepoint.com/article/perfect-pop-up/>

going to get stuck in a loop endlessly crawling every single permutation of the query parameters. Thus, URLs with many (definitely more than 5) query parameters have a very low chance of ever being crawled.”¹³ Here’s an example of the kind of URL Mr Seidman’s talking about:

```
http://mybyz.com/products/page.aspx?view=14&tab=6&pcid=24B8-1000849@@@&section=586&origin=mybyzsearch&cookie=false
```

Use text alternatives to Flash content.

If your site uses Flash to present content that you really want or need search engine crawlers to index, provide text alternatives that they can index. If you use Flash for your entire web site without providing text alternatives, don’t expect search engine crawlers to index your site.

Provide metadata and text alternatives for audio and other rich media files.

Typical metadata information includes the title, author, copyright, and a description of the content. Specialty search tools, such as Singingfish,¹⁴ use this metadata to help users find rich media, including MP3 files. Be sure to include helpful text information about your rich media content within your web pages, too.

Avoiding Being Banned by Search Engines

Search engines will ban web sites that try to improve their rankings artificially, so it’s important not to do anything accidentally that even hints at trying to fool them. Use the checklist below to help you avoid any misunderstandings between your web site and the search engines.

¹³ Eytan Seidman, *Crawling the Internet...*, msnsearch’s WebLog (November 18, 2004) [<http://blogs.msdn.com/msnsearch/archive/2004/11/18/266087.aspx>].

¹⁴ <http://singingfish.com/>

Use genuine SEO methods.

Don't even think about trying to play tricks on the search engines to improve your rankings! You run a good chance of having your web sites banned from their search results.

Avoid cloaking.

Cloaking is an artificial technique that malicious users employ to deliver to search engines content that's not delivered to web site visitors.¹⁵ Some web sites try to trick search engines by delivering specific content that only the search engines will see. There are many debates around the Web about what the term "cloaking" does or doesn't cover; for instance, some question whether or not using CSS to hide content from web site visitors, or using JavaScript sniffing to serve different content to specific search engines than is served to web site visitors, is cloaking.

Avoid using doorway pages or domain names used for doorways.

Doorway pages, or domain names used for doorways, are web pages or domain names that are stuffed with keyphrases and submitted to search engines.

Avoid stuffing keywords into comments or `title` or `alt` attributes.

Content in comment tags offers no benefit at all, and the image `alt` and `title` attributes should only be used as a text alternatives for an image.

Avoid using JavaScript or `meta refresh` redirects to try to trick search engines.

Often, search engines won't index pages that use JavaScript or `meta` redirects because, historically, so many web sites used them to trick search engines. So even if you want to use these techniques without trickery in mind, avoid doing so! Otherwise, your web site may be banned from search engines by mistake. Some argue that these techniques fall under the definition of cloaking, but regardless of whether or not you agree with that opinion, it's better to avoid using JavaScript or `meta` redirects.

¹⁵ "What is cloaking?" in *Google Information for Webmasters*, Google.com (no date) [<http://www.google.com/webmasters/faq.html#cloaking>].

Getting Listed

Getting your web site listed on search engines, other web sites, and directories is an important goal, but even more important is identifying the right time to do so, to which engines you should submit your site, and which pages you should submit. Doing some careful research and preparation will help make the submission process smoother and more efficient, and will help you achieve better results in the long run. Use this checklist as your guide.

Checklist 12.6

Submit your web site to search directories only after initial SEO is complete and content has been finalized.

Submit your web site to search directories only after your web site is live and has been tested for broken links.

Research each directory or search engine and its categories individually, read each directory or engine's FAQ, and follow instructions precisely before submitting your web site.

The mistakes that are all too easily made during the submission process are very difficult to change later, and can potentially cost you dearly. These errors can greatly affect the likelihood that search engine crawlers will find your site.

Research the best description to use for each search directory before you submit.

Search each directory for your most important keywords. Study the categories of results, and descriptions, and base the style of your description on those of web sites that have already been accepted in that particular directory. In addition, if a directory doesn't have a category for your business type, consider suggesting a category that's similar to those already in the directory.

Ensure your unique selling description is also included in your directory description.

Even though your description should resemble the description style of other web sites that have been accepted by that particular directory, make

sure your description is unique. For example, include specific services or products that you offer, and mention your target audience.

- ✓ **Write several descriptions of varying lengths to copy/paste into submission forms.**

Each directory has different requirements in the number of words allowed in a web site description. Write several descriptions within a plain text file that you can copy/paste into the submission forms as appropriate. Include descriptions with seven, ten, 15, 20, 25, 30 and 50 words.

- ✓ **Ensure your most important keywords are in every description, but avoid keyword stuffing.**

Make sure your web site's most important keywords are in every description, but don't stuff too many keywords into it! Directory editors don't like descriptions that basically read like a list of keywords.

- ✓ **Consider paid or expedited submissions as a way to have your web site listed more quickly.**

The sooner you can have your web site listed in search directories, the sooner your web site can receive referrals from those search engines. Using paid or expedited submissions is one way to help speed up the process of having your web site listed, and can sometimes make the wait as short as a day or two. It's important to note that paid inclusion doesn't typically guarantee high placement. Be sure to do your homework, and be sure to read quality SEO/SEM web sites and helpful books, such as Dan Thies's *The Search Engine Marketing Kit*, cited earlier.¹⁶

- ✓ **Submit your homepage to search directories before submitting other web pages.**

- ✓ **Submit other pages, particularly those on specialized topics, once your homepage is listed.**

You'll have a better chance of having your other web pages, particularly those on specialized topics, accepted for different categories once your web site's homepage is listed at that directory.

¹⁶ See also Danny Sullivan, *Buying Your Way In: Search Engine Advertising Chart*, Search Engine Watch (November 22, 2004) [<http://searchenginewatch.com/webmasters/article.php/2167941>].

- ✓ **Submit your web site manually to search directories, rather than using automated submission software.**

Search engine crawlers, including Google, advise against using automated submission software.¹⁷



Automated Software Violates some Terms of Service

Note that some automated software violates the terms of service of many search engines and directories, so before you consider using any automated software for your search submissions, to check link popularity, or to check rankings, read the terms of service for each search engine and directory to ensure that you don't violate the rules.

- ✓ **Submit your web site to search directories such as Yahoo!, DMOZ, JoeAnt, and Gimpsy.**

Listings with major directories are critically important to the success of crawlers from search engines such as Google finding your web site.

- ✓ **Avoid over-submitting.**

Check to see if your web site is already listed on the search engine, and don't submit it again if it is listed. It's important to follow up and keep good records of your submissions, acceptance, listings, and rankings.

- ✓ **Use a `robots.txt` file and meta robots tags to denote content that you don't want indexed.**

You might have web pages, images, sound files, or other content that you don't want to have indexed. If so, you can specify that those files should be ignored in a `robots.txt` file that is saved to your server's root directory. Here's an example of a `robots.txt` file that instructs Google not to index images, and tells all search crawlers not to index the `procedures` directory, on the MyByz.com web site:

```
# robots.txt for http://mybyz.com/  
# exclude images from Google  
# see http://www.google.com/remove.html#images  
User-agent: Googlebot-Image
```

¹⁷ *Google Information for Webmasters* Google (2005) [<http://www.google.com/webmasters/guidelines.html>].

```
Disallow: /images
# exclude specific areas from all search engine crawlers
# (in this case, a directory containing procedural files
User-agent: *
Disallow: /procedures
```

You can also include the `meta robots` element on specific pages. For example, if you had a page that you didn't want to have indexed, but you did want the spider to follow the links on that page to other pages, you'd add the following instruction to your page's markup:

```
<meta name="robots" content="noindex,follow" />
```

note

Ignoring meta robots and robots.txt Files

Note that not all search crawlers will honor `meta robots` attributes or `robots.txt` files, but many will (including the most-used engine of all, Google), so these techniques are definitely worth using, if needed.

Creating an Ongoing Links Campaign

All major search engines place a lot of importance on a web site's overall link popularity. Highly-ranked web sites that link to yours can help boost your own site's ranking. Use the checklist below to help you plan and manage your web site's incoming and outgoing links.

Checklist 12.7

- Provide ongoing, link-worthy content.**
 If you provide quality content at your web site, other web sites will link to it, often without you even asking.
- Provide plenty of outbound links to web sites that you recommend.**
- Ask other web site owners to consider linking to your web site or exchanging links.**



Don't Abuse Forums

Avoid posting on forums solely to create links to your web site. The forum owners will see right through your tactics, and you'll likely lose

all credibility. Such tactics would probably also cause other site owners to avoid linking to your web site legitimately. If you're serious about building links to your web site, create great content that others will want to link to, and don't hesitate to email other web site owners to ask if they'd be willing to link to your site.

- ✓ **If any URLs change at your web site, be sure to inform those who link to it.**

Keeping these other sites and site owners informed will help them, as well as your web site, to find and link to your current, live web pages, rather than maintaining old, dead links. Remember that it's important to use server-side redirects to the new URLs, as noted above.



SEO Takes Work

When it comes to optimizing your web site for search engines, there is no one single, magic solution. It takes time and patience to achieve high rankings, but you can work to boost your search engine rankings by following tips from SEO industry experts, using the checklists above, and remaining diligent about optimizing and promoting your web site. Don't use artificial techniques.

Planning for Ongoing Maintenance

Checklist 12.8

- ✓ **Follow-up every four to five weeks with search engines and directories to ensure that your web site is listed.**

If you don't find your web site listed at a particular directory within four to five weeks, go ahead and resubmit.



Track your Submissions

It's critically important to keep track of the dates of your submissions to each directory, along with the details of the categories to which you've submitted, and the descriptions and title content you used.

- ✓ **Contact a directory representative if you don't find your web site listed at a directory after three submissions.**

- ✓ **Check your server logs regularly to see how visitors find your web site and adjust your web site accordingly.**

You might need to do some tweaking to improve your rankings. You might also write new content with new keywords and keyphrases to support, or shift, your search rankings.

- ✓ **Add announcements about, and links to, new content from your homepage, to help web site visitors and search crawlers find it.**

- ✓ **Add new keywords and keyphrases to your SEO strategies when you add new content or make other changes to your site.**

If you've already optimized your web site, search crawlers will find your new content, especially if you link to that content from your homepage, as noted above.

- ✓ **Consider using analytics software to review keyphrases, search engine rankings, and listings status.**

Web analytics software, such as Web Position Gold, can help track top entry pages, top referring sites, top referring URLs, top search engines, top search keywords and keyphrases. You don't necessarily have to pay for expensive statistics software, though: free services, such as Extreme Stats,¹⁸ provide much of this information, although in using many of these services you are obliged to display on your site a graphical links to the stats, which makes them available to all your site visitors.

- ✓ **Use the results of keyphrase analysis to add to your web site content that visitors seek.**

- ✓ **Avoid making changes to your site's SEO tactics for at least three months after you optimize the site and submit it to search directories.**

It takes time for your web site submissions to begin to appear at search engines and directories—it can often take two to three months or more, especially if you have a new site.¹⁹ Try to be patient! Consistently high rankings take time.

¹⁸ <http://www.extreme-dm.com/tracking/>

¹⁹ Jill Whalen, *Robots Meta Tag – Changing Domain Names*, High Rankings Advisor Issue 137 (April 20, 2005) [<http://www.highrankings.com/issue137.htm>].

- ✓ **Avoid constantly tweaking your web site in an attempt to improve your search engine rankings.**

As I mentioned above, it takes time for your tweaked pages to show up at the search engines, so you'll need to wait to see how your pages are doing. After you've given your rankings plenty of time to filter through, if you decide you need to tweak any pages, wait to see what impact those tweaks have before making more changes. Time and patience are an important part of SEO success.

- ✓ **Check your link popularity at search engines regularly.**

Many web sites will link to yours without letting you know, and in addition to checking your referral logs, using search engines to find web sites that link to yours can be an easy way to find such sites. This information can help you ascertain and follow the efficacy of your reciprocal links campaign.



Checking Incoming Links

You can enter special queries at each search engine to check for web sites that link to yours. For example, at Google's search box, type:

link:www.domainname.com

At AltaVista's search box, type:

link:domainname.com

Keep in mind that only those web sites listed at a particular search engine will appear in that search engine's query results.

- ✓ **Depending on the industry, consider creating a calendar of keywords and keyphrases for future reference.**

In some industries, the popularity of given keywords can fluctuate throughout the year. Site owners operating in such industries might find it worthwhile to keep a calendar that lists which keywords and keyphrases are popular at different times of year.

Summary

This chapter covered the steps involved in optimizing your web site for search engines. The basic principles of successful search engine optimization (SEO) were explained in a nutshell, and were followed by checklists that showed how to re-search, prepare, and use keywords and keyphrases within your content and markup.

We also discussed checklists that showed how to create crawler-friendly markup, and how to avoid tricking crawlers. We highlighted in general terms the techniques that could see your web site banned from search engines and directories.

Finally, we stepped through checklists that addressed the important questions of creating and managing an ongoing links campaign, and maintaining your web site's SEO over time.

14

Testing

Testing plays a critical role in the development of your web site and its long-term maintenance. While smaller web sites—especially those with more limited budgets—may not need to follow the formal testing procedures that are required for large-scale, commercial web sites, every site needs to be thoroughly tested to ensure that it’s error-free, user-friendly, accessible, and standards compliant.

Testing should be completed during each phase of a site’s development. Two of the most costly web project mistakes are to delay testing until just before launch, or not to test at all. Testing *during* production makes it easier to locate and resolve errors, and minimizes the chance of existing bugs being replicated throughout later stages of development. Early and continued testing can eliminate the need for the costly redesigns and other major fixes that can result from overlooked errors.

This chapter’s checklists will help you test your site both during development, and after.

Getting Started

Checklist 14.1

Document your baseline web site testing requirements.

Make use of the preliminary data you collected in Chapter 2 to help determine and document your baseline site testing requirements. For instance, the information you collected on the browsers and connection speeds that your visitors will use, and those visitors' skills and age groups, can be used as a basis for your testing plan.

Source and install all necessary tools.

You can conduct tests very cheaply using free and low-cost testing tools. Most browsers are free or offer conditionally free versions, and you can download and install multiple browsers—including several versions of Internet Explorer for Windows—on one computer. Alternative device simulators and emulators, such as those for cell phones and PDAs, can also be downloaded and used for free, and most shareware and commercial software creators allow a 30-day trial of their products for testing purposes. Some web design-related discussion lists, as well as the SitePoint Forums, welcome testing requests and posts that solicit bug reports and feedback.

Provide acceptable testing protocols.

Ultimately, the design and layout for each page of the site must be both aesthetically pleasing and functional across multiple platforms and browsers.

One of the difficulties associated with testing design elements on different platforms is that the test results can be subjective. For example, a designer may create a CSS-driven site that works perfectly on newer browsers, but doesn't look that hot in older browsers such as Netscape 4. The tester may interpret this as a failure point rather than acceptable "design degradation."

To avoid the time and effort involved in re-submitting pages for testing, spell out acceptable cross-browser, cross-platform differences in advance—including protocols for supporting old browsers.

**Tip**

Use Annotated Screenshots to Determine Display Protocols

At the end of the final prototyping stage, ask the designer to provide a series of screenshots that have been annotated to identify the differences in the ways the site displays on different browsers and platforms. These screenshots can be used in later testing to identify the displays that are deemed acceptable on each platform. Testers can then compare the protocols and screenshots with their findings.



Set up a staging server.

A staging or test server that's identical to your web site's live server environment should be used for testing and development purposes prior to your site's launch.

**Tip**

Automated Programs for Larger Sites

Larger web sites typically require more robust testing procedures that test large sets of records, values, and content. You might consider using an automated program, such as Badboy,¹ eValid,² or TestComplete³ for such purposes.

Good Testing Practice

Checklist 14.2



Systematically test individual pages.

Thoroughly test your site one page at a time, making sure each page is solid before proceeding. Carry out a comprehensive test of the final version of the site in the same way.



Track bugs and confirm fixes.

Document the details of discovered bugs and their resolutions.

¹ <http://www.badboy.com.au/>

² <http://www.soft.com/eValid/>

³ <http://www.automatedqa.com/products/testcomplete/>

 **Regression test, especially when fixing bugs.**

Regression testing, which is also known as verification testing, is the process of retesting pages or sections of a site to make sure that a recent bug fix hasn't broken some other aspect of the site, or reinstated bugs that were fixed previously. Conducting and documenting regression tests will allow you to identify any parts of the site that break, and document the causes of those errors and how you resolved them. If a bug resurfaces at some point, it will be spotted immediately and resolved quickly.⁴

 **Validate the markup for each individual web page.**

Validate the markup for all your web pages to ensure that each uses structural, compliant markup. This will allow you to confirm that you've used the correct markup for the specified DOCTYPE, fix any typos you may have made, and resolve any syntax errors. Validating your markup as you build a page, rather than waiting until you've created an entire page (or worse, an entire web site), makes it easier to isolate, find, and correct markup problems.



Using Markup Validators

Some HTML editors integrate validation tools such as W3C's HTML and CSS validators, the WDG HTML Validator, and the CSE HTML Validator (an HTML/XHTML syntax checker and validator), which allow you to validate your markup conveniently while you work.

The CSE and WDG HTML Validators also include helpful batch processing features. So, for example, although I validate each page with the CSE HTML Validator and the W3C's validators as I work, I use CSE HTML Validator's batch processing to recheck multiple pages simultaneously. I also use batch processing as a final check when I've finished developing all the pages of the site.

Ensure that the validation tool you use is based on W3C Recommendations.

 **Validate all CSS.**

Validate all the CSS you use on the site, including each external style sheet, plus all embedded and inline CSS.

⁴ *Regression Testing*, Webopedia Computer Dictionary (no date) [http://www.webopedia.com/TERM/R/regression_testing.html].

- ✓ **Conduct load testing to stress-test programming technologies and server hardware capacities.**

You need to make sure that the programming technologies you've used, and the capacities of your server hardware, can cater to higher levels of traffic than you expect the site to receive. Taking this approach will help you ensure that your site remains online and fully functional at all times. Load testing procedures should include a trial run—tests that usually involve the use of automated scripts that emulate multiple simultaneous user sessions. The trial run will reveal how the programming technologies and server hardware you've used will cope with traffic spikes.

note

Matching Hosting with Projected Server Loads

Your ISP should match your hosting package with your projected server loads if you're using a web hosting service, so you should let them know what your loads will be. Your web host might recommend a different web hosting package that offers greater capacity if your load testing reveals that your existing setup has the potential for problems as traffic levels increase.

If you're managing your own server(s), check whether your server software provider offers free load testing software—many do. You might also consider using dedicated load-testing software products, such as those offered by WebPerformance⁵ and NetMechanic.⁶ In addition, numerous open-source performance analysis software tools⁷ are available to help you assess, monitor, and manage your web site's performance.⁸

General Testing

Checklist 14.3

- ✓ **Test your web site on multiple browsers and platforms.**

The fact that your web site looks good and works well in one browser doesn't mean it will look as good or function as well on other browsers and/or platforms. Even if you're developing an intranet web site for an or-

⁵ <http://www.webperformanceinc.com/>

⁶ <http://netmechanic.com/monitor.htm>

⁷ <http://www.opensourcetesting.org/performance.php>

⁸ See also *Stress Tools to Test your Web Server*, Microsoft.com [<http://support.microsoft.com/kb/231282/en-us>].

ganization whose employees use the same browser and operating system, the hardware and software used by those employees can and will vary. In the long run, it's better to develop your web site on the basis of the W3C recommendations, then to follow up with extensive cross-browser, cross-platform testing. Stay informed of the release of new or updated browsers and operating systems so that you can keep your cross-browser, cross-platform tests current.



Which Browsers and Platforms to Test?

Check your server logs—as well as other on- and offline resources—that cover browser statistics, operating systems, and emerging Web use trends, so that you know which browsers, versions, and operating systems your web site visitors use now, and are likely to use in future. As part of your approach to site maintenance in the long term, plan to check your server logs weekly (monthly checks might be fine if you run a smaller web site with low traffic volumes) to follow changes and trends in your browser statistics. Take this data into account as you hone your cross-browser, cross-platform testing procedures, to ensure that you're testing all of the browsers and operating systems that your visitors use. Here is a list of popular operating systems and browsers on which you might test your site:

Operating systems	Macintosh OS X, Macintosh OS 9, Windows XP SP1 and SP2, Windows 2000, Windows 98, Linux
Browsers for Macintosh OS X	Safari 1.2, Mozilla 1.6, Firefox 1.0, Opera 8, Opera 7, Internet Explorer 5.2
Browsers for Macintosh OS 9	iCab, Internet Explorer 5
Browsers for Windows XP	Opera 8, Opera 7, Mozilla 1.7, Firefox 1.0, Netscape 7.1, Internet Explorer 6.0, Lynx browser
Browsers for Windows 2000	Opera 8, Opera 7, Mozilla 1.7.3, Firefox 1.0, Netscape 7.1, Netscape 7.0, Netscape 6.2, Netscape 4.78, Internet Explorer 6, Internet Explorer 5.5, Internet Explorer 5.0, Lynx browser

Browsers for Windows 98 Internet Explorer 4.0, Lynx browser

Browsers for Linux Konqueror 3.0.5, Mozilla 1.6, Opera 8, Opera 7, Emacs/W3, Netscape 7, Netscape 4.8

✓ **Test page optimization with every update.**

Optimizing your pages right from the start can help ensure that your site design and images support fast page load times, and that the site's development is smooth and efficient. Whenever you add anything to your web site—new content, new images, or new web pages—check that content's optimization to help keep the site streamlined. If you're not yet familiar with web site optimization, see Chapter 11.



Document Weight and Load Time Tools

Some HTML editors include features that tell you a document's weight, or a page's load time at various connection speeds, and most image editing software products indicate file size load times at various connection speeds.

There are also free online tools that calculate document weight, composition, and page load times, and even offer recommendations for optimizing web documents. Siteoptimization.com's helpful Web Page Analyzer⁹ is one such tool.

✓ **View pages on a variety of displays.**

You may find that your beautiful design, which is wonderfully contrasted on an LCD monitor, is impossible to read on a CRT monitor, and completely unusable on alternative devices. Different displays will not always interpret your site's design and colors consistently, so it's worth testing your initial design, as well as the site's ongoing development, in a variety of displays.

⁹ <http://www.webpageanalyzer.com/>

**View pages on different screen resolutions and with various color settings.**

Some HTML editors provide a feature that allows you to check your pages at various screen resolutions and with a range of color settings. Similar tools can also be downloaded as plug-ins for some browsers. For example, the highly recommended Web Developer Toolbar extension for Firefox includes a customizable window resizing tool, among many other useful features.¹⁰

**Consider Nonstandard Resolutions**

Screen display sizes may or may not reflect the actual dimensions of the viewable web page. Many users don't expand browser windows to fill the entire screen (especially on larger monitors), and various toolbars and side panels can take up portions of the screen width and height. For example, on an 800x600 display, a browser that has a side panel and a couple of custom toolbars across its top would display only 500x400 pixels of a web page—less if the browser window isn't maximized. Be sure to conduct testing that accounts for this wide range of variability—not just the standard maximum screen resolutions. Table 14.1, which shows screen resolutions of various devices, and Table 14.2, which shows the color depth settings of different displays, should help your testing.

Table 14.1. Typical screen resolutions and displays (in pixels)

Computer displays	640x480, 800x600, 1024x760, 1280x1024, 1440 x 900 (iMac G5 17" display), 1680 x 1050 (iMac 20" G5 display)
WebTV/MSN TV	544 x 383
PDA, Pocket PC with Windows Mobile, smartphones	240x160: Palm; 320x240: Palm, Pocket PC with Windows Mobile pre-2003, v2; 640x480: Windows Mobile pre-2003, v2; 320x320: Palm OS smartphone ^a

^a Lisa Gade, *palmOne Treo 650 Palm OS Smartphone*, *Mobile Tech Review* (December 10, 2004) [http://www.mobiletechreview.com/Treo_650.htm].

¹⁰ For more on Firefox's tools for web developers, see Chris Pederick's article, *Web Developer Extensions for FireFox and Mozilla* [<http://www.chrispederick.com/work/firefox/webdeveloper/>].

Table 14.2. Typical display color depth settings

Computer displays	Newer displays: 24-bit, 32-bit, 65,000 colors, 16,777,216 colors; Old displays: 16-bit, 256 colors, 65,000 colors ^a
PDA, Pocket PC with Windows Mobile, smartphones	Pocket PC: 16-bit, 65,000 color screens; Old PDAs: 8-bit, grayscale ^b

^a *Browser Statistics: What Is the Trend in Browser Usage, Operating Systems and Screen Resolution?* W3Schools (January 2005) [http://www.w3schools.com/browsers/browsers_stats.asp].

^b *What is a Pocket PC (PPC)? What models are out there?* Mobile Tech Review (no date) [http://www.mobiletechreview.com/ppc.htm].

In addition to the displays mentioned in Table 14.1 and Table 14.2, there are standalone tools—such as BrowserCam,¹¹ an online subscription service—which provide screenshots depicting the way your content will appear in a wide range of screen resolutions and browsers.

Check for adequate color contrast.

A good way to check color contrast is to change your browser display to grayscale, or print the web page in black and white. A tool that can be very helpful for checking your design's color contrasts is Vischeck,¹² which simulates colorblind vision, allowing you to see your pages as a colorblind user might.

Test the functionality of external and embedded scripts and functions.

Some scripts and functions might work fine in one browser, though they may not work correctly, if at all, in other browsers or devices. Minor tweaking can resolve many cross-browser, cross-platform functionality issues. If older browsers or alternative and adaptive technologies don't support some functions, you may wish to provide alternative methods to allow those users to access those capabilities some other way.

Test all links, including navigation.

Ensure that all links work properly, and point to the correct location. Link checking software can check for valid, broken, and redirected links. Some HTML editors, such as Macromedia Dreamweaver¹³ and HomeSite,¹⁴ offer

¹¹ <http://browsercam.com>

¹² <http://www.vischeck.com/>

¹³ <http://www.macromedia.com/software/dreamweaver/>

¹⁴ <http://www.macromedia.com/software/homesite/>

link checking capabilities. Free tools are also available—try Xenu’s Link-Sleuth¹⁵ and W3C Link Checker.¹⁶ See WebsiteTips.com’s section, “Link Checkers, Maintenance”¹⁷ for more. Note, though, that link checkers aren’t intuitive enough to check whether links direct to the correct location—be sure to check for this manually.

Check error pages.

Intentionally enter incorrect URLs into the browser’s address bar to check that the appropriate error pages (404 errors, etc.) are in place.

Make sure that the error pages include helpful information and links that help users find what they seek. Keep in mind that, sometimes, visitors will incorrectly guess a domain name or slightly misspell a URL. Help those visitors by accepting common misspellings, typos, incorrect case sensitivity, different terminology for the same items, and predictable domain name errors: provide suggestions to redirect users to the proper page.¹⁸

Test all downloads.

Check to ensure that all your download links point to the correct files, and that the download files exist.

Test the search feature.

Ensure that the search feature functions correctly, and that search results are accurate and helpful.

Solid Security



Clear your Cache!

Be sure to clear the browser cache, including cookies, before each test.

¹⁵ <http://home.snafu.de/tilman/xenulink.html>

¹⁶ <http://validator.w3.org/checklink>

¹⁷ <http://websitetips.com/tools/checklinks/>

¹⁸ Matthew Linderman with Jason Fried, *Defensive Design for the Web: How to Improve Error Messages, Help, Forms, and Other Crisis Points* (Indianapolis: New Riders, 2004), 99, 224.

✓ **Check that digital certificates and SSL URLs work correctly.**

Your SSL URLs should begin with `https://`. Also, when a visitor accesses a web page protected by SSL, most browsers will show a “lock” icon at the bottom of the browser, like that being shown by IE 6 for Windows in Figure 14.1.

Figure 14.1. The “lock” icon displaying at the bottom of the browser window



SSL is commonly used to encrypt online transactions and other sensitive data, as well as intranets and extranets. Verisign¹⁹ and Thawte²⁰ are two popular companies that issue SSL certificates.

✓ **Check that all pages requiring SSL access are accessible only via SSL.**

✓ **Test the security of restricted areas.**

Users might share or try to guess the URLs of content and downloads in protected areas of your site. Restricted content URLs might appear within publicly available referrer logs. People might even share usernames and passwords without authorization or permission. As such, it’s critical that you anticipate the kinds of security breaches that might take place, and test methods for their prevention.



Tip

Hack your Site: the Ultimate Security Test

Download popular hacking tools to use in testing, to see if your protected areas can easily be comprised. Those managing larger-scale web sites might even consider outsourcing or hiring a hacking expert for testing purposes.

¹⁹ <http://verisign.com/>

²⁰ <http://thawte.com/>

Ensuring security and data integrity—especially in terms of your confidential data, including customer credit card information—is critical for promoting and maintaining trust among your site’s visitors. Don’t assume that your web site’s security is always okay. Test regularly to make sure it remains secure.

Test forms and form controls.

Check to ensure that forms are submitted correctly, and that they’re only submitted when the correct information is entered and required fields have been completed. Review form error messages to ensure that they are helpful and informative within the context of the form itself.

Test online shopping facilities.

If your web site includes a shopping cart or similar functionality, thoroughly test back-end operations to ensure that all transactions are secure, and everything runs smoothly.

Accessibility Testing

If you plan to make your site accessible, and include accessibility testing in every phase of your project, chances are good that by the time your site is ready to launch, it will meet the W3C’s WCAG Guidelines, as well as any other accessibility guidelines that it must address.

Conducting a Preliminary Review

The W3C recommends a two-phase process for testing a site’s accessibility. First, developers should conduct a preliminary review of the site that includes testing a few pages to get an idea of the site’s accessibility. Then, they should undertake a thorough, comprehensive review that includes an evaluation of every web page (or representative pages on larger web sites) with a variety of tools and users who have disabilities.²¹ The checklist below is a guide for conducting a preliminary accessibility review.

²¹ W3C, *Evaluating Web Sites for Accessibility*, W3C Web Accessibility Initiative (November 14, 2002) [<http://www.w3.org/WAI/eval/>].

Select a random sampling of pages to test.

You should test every page of a small web site. For larger web sites, select a random sampling of pages that includes your homepage, at least one or two pages from each section, and especially important or popular pages, such as the contact page and the sitemap.

Use a graphical browser to test sample pages.

Use a voice *or* text-only browser to test sample pages.

A text-only browser, such as the free Lynx text-only browser, can provide valuable insight into your web site's accessibility.

Use two evaluation tools.

Manually examine representative pages using the checkpoints from WCAG 1.0, your government accessibility guidelines, or other requirements.

Summarize the results.

If your results summary indicates a number of accessibility issues, you might wish to make site-wide changes before you conduct a comprehensive review. Conduct another preliminary test after you implement these changes.

Conducting a Comprehensive Review

A comprehensive review is much more thorough and detailed than a preliminary review. Here's a checklist that will help you ensure you don't miss a thing!

Identify, determine, and document site-wide conformance requirements and levels.

You may include WCAG 1.0 Level 1, local government requirements, and company guidelines in your documentation, for example.

Test the site's accessibility conformance.

Test every single page of smaller web sites. For larger web sites, test a representative sample that includes the homepage and all especially important or popular pages, pages from each section, pages for each different layout, and pages generated from databases.

Use *at least* two different accessibility evaluation tools.

Manually examine representative pages using the checkpoints from WCAG 1.0, your government accessibility guidelines, or other recommendations.

Use a graphical browser to test sample pages.

Use a text-only browser and a voice browser to test pages.

Ensure that interface elements are operable with multiple input devices.

For example, test all interface elements with a mouse, with keyboard only, and with adaptive technologies. If you use JavaScript to create navigation rollover effects, ensure that links work without JavaScript, provide HTML text links, or implement both.

Proofread all content.

Read over each page of the site to check that all the text is clear, simple, and appropriate for your web site's target audience, as well as those segments of the market that have cognitive or learning disabilities.

Make sure that your text is clear, simple, and written for the Web, then boost its accessibility by allowing for those with disabilities.



How Complex is your Writing?

Clear Language and Design (CLAD) offer a great Reading Effectiveness Tool²² that's designed to assess the literacy skills that users will require in order to understand your text.

For helpful insight on cognitive and learning disabilities, see Roger Hudson, Russ Weakley and Peter Firminger's article, *An Accessibility Frontier: Cognitive Disabilities and Learning Difficulties*.²³

Conduct usability testing with a diversity of participants.

If possible, conduct usability testing with people who have a variety of disabilities, technical expertise, and varying levels of familiarity with your web site. Use a variety of adaptive technologies, if possible.

There's no better way to assess your site's accessibility than to have disabled users test the site. It should be noted, for example, that a sighted person will not use a screen reader, such as JAWS, in the same way as a blind person who uses the tool daily.

Testing with a Graphical Browser

You can do a fair amount of accessibility testing with your browser. Opera is especially conducive to accessibility testing, since the browser makes it quite simple to turn off images, animations, style sheets, JavaScript, frames, and so on. Ian Lloyd has written a great tutorial about using Opera for testing a site's accessibility.²⁴

There are also helpful accessibility testing extensions, sidebar panels, and toolbars for other graphical browsers, including Mozilla, Firefox, and Internet Explorer. As well as visiting your favorite browser's web site, see Derek Featherstone's *Testing Tools for Developing Accessible Web Sites*²⁵—a helpful list of annotated browser links.

²² <http://www.eastendliteracy.on.ca/ClearLanguageAndDesign/readingeffectivenessstool/>

²³ <http://www.usability.com.au/resources/cognitive.cfm>

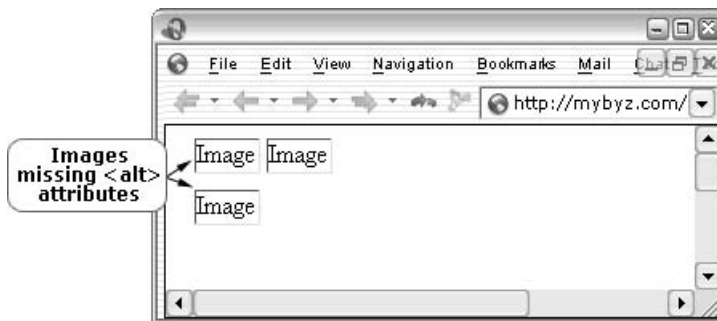
²⁴ <http://www.accessify.com/features/tutorials/opera-accessibility/>

²⁵ <http://www.wats.ca/show.php?contentid=46>

View pages without images.

As long as you've included the `alt` attribute for each image (for use by browsers that don't support images, or visitors who are browsing with images turned off), the browser will hide decorative images that have blank `alt` attributes. However, if you don't include `alt` attributes, browsers will not hide the images, and screen readers will read the word "Image," as indicated in Figure 14.2.

Figure 14.2. Viewing images with blank `alt` attributes in Opera 7 for Windows



View pages without style sheets.

Check that your pages allow users to vary the font sizes displayed.

View tabular data with tables disabled or in a browser that doesn't offer table support.

This process will indicate how usable your tabular data might be for visitors using screen readers, text-only browsers such as Lynx, and older PDAs.

View the web site with new technologies turned off.

These "new technologies" include JavaScript, Java, sound capabilities, and plugins. Your page should remain functional and accessible when these technologies are unavailable.

Testing with a Voice or Text-only Browser

If you don't have handy access to a voice browser, you can use Lynx, a free text-only browser, or Delorie's online Lynx Viewer, for preliminary testing. For comprehensive testing, the W3C recommends testing with a text-only browser and a voice browser.²⁶

Checklist 14.8

Check that web site navigation is available and works properly.

If your web site uses images—including image maps—for navigation purposes, text hyperlinks should appear in their places (via `alt` attribute text, and possibly also in the `title` attribute text, if needed). If you use Flash or Java applets for navigation, your site should include alternative text navigation links to help ensure accessibility.



Helping Everyone get Around

Keep in mind that this approach also helps search engine bots to crawl and index your web site. For more on optimizing your web site for search engine crawlers, see Chapter 11.

Check all hyperlinks.

Check that decorative images don't appear in text-only browsers and aren't read by voice browsers.

As we discussed above, all decorative images, such as bullets and design-related images, should have blank `alt` attributes so that text-only browsers and voice browsers ignore them.

Check that the information is presented in a way that's comprehensible when read serially.

Lynx allows you to simulate serial access, so that you can make sure the information you've presented is understandable to users of voice and text-only browsers.²⁷

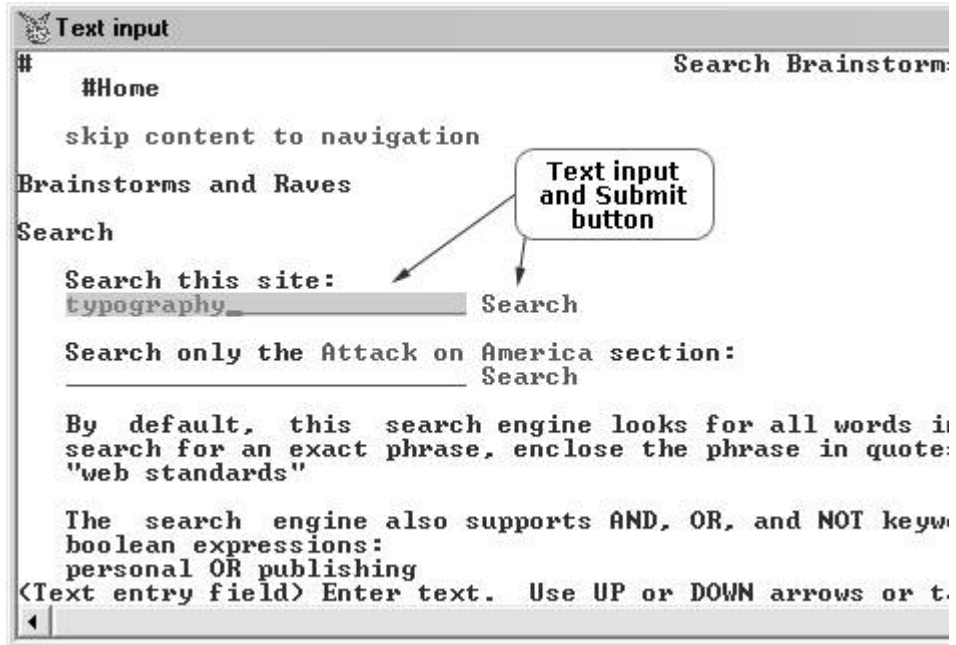
²⁶ W3C, *Evaluating Web Sites for Accessibility*, Web Accessibility Initiative, W3C (November 14, 2002) [<http://www.w3.org/WAI/eval/>].

²⁷ Joe Clark, *Building Accessible Websites* (Indianapolis: New Riders Publishing, 2003), 236.

- ✓ Test form controls.

Figure 14.3 depicts the testing of an accessibility-friendly form with Lynx 2.85.

Figure 14.3. Testing a form in Lynx



Usability Testing

If you find yourself short on money, time, or both, you might be tempted to consider skipping the usability testing phase. Rather than skipping this process altogether, it's far better to test on an abbreviated, informal scale. It's better to conduct a number of small, fast tests during development, rather than one big test at the end—especially if that final usability test is to be scheduled just before the site's launch.

The main goal of usability testing is to make sure your final-version, live web site is user-friendly and works well for your visitors. Three main dimensions of usab-

ility testing are effectiveness (testing to see if the user can accomplish desired tasks), efficiency (testing to see how much effort is involved in accomplishing the desired tasks), and satisfaction (testing to see if the user has a satisfactory experience and will return).²⁸ Use the following checklist as a guide to determining and conducting your usability tests.

Checklist 14.9

✓ **List test objectives and concerns.**

Be specific, making note of tasks and areas of the web site that might be of concern—for instance, forms that might be difficult to complete or use, or the methods by which users are expected to find certain products. Be sure to specify these age groups and computer skill levels that you wish to test.

To help determine your objectives and concerns, check with your clients, management, designers, developers, and others involved in the project.

✓ **Choose test approaches.**

Numerous approaches have been developed to testing various aspects of sites' usability. In planning your usability tests, it's important that you research and consider the available approaches in light of the resources and time you have available.

For example, you may choose to use personas or card sorting to help test your web site's information architecture—one single, but critical aspect of your site's usability. By creating personas, you can document the steps typical visitors would take to accomplish particular objectives, then test those steps—looking specifically at the information architecture, navigation, and forms, for example, that that persona might use to achieve that goal.

Card sorting can be a helpful way to understand how your web site visitors will likely understand and explore the information on your site.



Conducting a Card Sorting Test

To conduct a card sorting test, give users index cards on which are written the titles of pieces of content, features, or categories. Include brief descriptions on each card, but don't hint at any information structure or classification. If you plan to add more content to your

²⁸ Dey Alexander, *Quick Wins with Usability Testing* (presentation, Web Workshop Series at Monash University, Melbourne, 2002), slide 3.

web site in future phases, it can help to add those specific elements to the cards if you can.

Be sure to brief your test users, explaining that there are no right or wrong answers. Ask them to read each card carefully and group them into piles that make sense to them. If the users aren't sure about a card, they can set it aside and come back to it whenever they wish. When they finish, ask them to label each pile.

Compare the results with your web site's information architecture as one way to determine whether or not further tweaks or changes are required.

Recruit users for testing based on your target market.

Test users should represent your target audience profiles. For example, if your typical users aren't very technology-savvy, but you test with technology-savvy users, your test results won't uncover potential technology-related problems, such as confusion with drop-down menus or multiple browser windows.

A test group of five users is considered large enough to provide information that will help improve your site's usability.

Prepare an NDA (non-disclosure agreement) ready for users to sign prior to starting your tests.

Determine testing location(s).

Jakob Nielsen and other usability experts talk about conducting testing in usability labs. Indeed, labs can be helpful when you're testing a highly sensitive and secure web site, you have significant hardware components or specific hardware that must be used for testing, you anticipate long testing sessions (more than a couple of hours or so), and you have a project for which you'd like a team of observers to watch the tests.²⁹

Alternatively, there can be advantages to "field" testing—testing the site in an office or home setting, using testers' own computers at these locations. This approach eliminates the costs of a usability lab while allowing you to observe how your web site will work for typical visitors. In addition, testers

²⁹ Julia Houck-Whitaker, *Remote Testing versus Lab Testing*, Bolt | Peters, Inc., (February 3, 2005) [<http://boltpeters.com/articles/versus.html>].

may be more comfortable and relaxed using their own computers. However, Jakob Nielsen reports that field studies tend to be expensive, and it can be challenging to get permission to conduct them at company locations.³⁰

Consider also the possibility of conducting remote testing. For example, you could be at your office and use the telephone to verbally communicate instructions, obtain user feedback, and engage in discussion. Unless you have online video cameras set up strategically for testers' computers, though, you'll miss the important observation part of your testing. You'll need to encourage them to "think out loud," and remember to phrase your responses and questions to remain as neutral as possible. Remote testing can be quite effective for large web sites, those with ecommerce capabilities, web applications, and intranets.³¹



Videotape your Tests

Videotaping tests can be helpful, as the videos can facilitate later review, allowing you to see details that you may not have noticed during testing, and to show the tests to others on the project team. Time-stamping the videos will make it easier to reference key findings later.



Top Testing References

For more on testing approaches, see Janice Fraser's *The Culture of Usability: How to Spend Less and Get More from Your Usability-Testing Program*,³² and Lane Becker's *90% of All Usability Testing is Useless*.³³



Create a relaxed atmosphere for testers.

Rather than making people think they're in a lab being assessed, decorate your test area as if it were a typical office setting or perhaps a living room. The SitePoint article, *Why You Don't Need a Usability Lab*,³⁴ has more on this.

³⁰ Elizabeth Neal, *Why You Don't Need a Usability Lab*, SitePoint (October 8, 2004)

[<http://www.sitepoint.com/article/dont-need-usability-lab>].

³¹ Houck-Whitaker, Velda Bartek, and Deane Cheatham, *Experience Remote Usability Testing, Part 1* IBM developerWorks (January 1, 2003)

[<http://www-106.ibm.com/developerworks/web/library/wa-rmusters1/>].

³² <http://www.ddj.com/dept/architect/184411669>

³³ <http://www.adaptivepath.com/publications/essays/archives/000328.php>

³⁴ <http://www.sitepoint.com/article/dont-need-usability-lab>

- ✓ **Establish realistic tasks for users to perform and identify what, specifically, you will measure.**

For example, to test a web site that offered numerous articles categorized by topic, you could set tasks that require users to find specific articles or topics. From these tests you could collect data on the time the test subjects needed for each task, whether or not the navigation was helpful, whether the local search was used, and if the results were accurate and helpful.

- ✓ **Provide simple, clear instructions only.**

For accurate test results, do *not* provide any hints.

- ✓ **Allow users to perform their own tasks in addition to your pre-defined tasks.**

Users will often come up with tasks you haven't thought of, and by pursuing these tasks, can uncover important subtleties and problems with a site.³⁵

- ✓ **Create a post-test questionnaire.**

Consider what, in particular, you'd like feedback on, and create a questionnaire that addresses those issues. Also, include areas for comments and suggestions. You may choose to make a questionnaire with a 1–5 ranking system (1 being poor and 5 being excellent), and that asks users to rate the following:

- their overall satisfaction with the web site
- their overall satisfaction with the design and layout
- the ease with which specific tasks were completed
- the ease with which specific information was found
- whether navigation labels, page titles, headings, and other terms were easy to understand and accurate
- whether icons and other images were helpful and easy to understand
- whether downloadable items downloaded quickly and satisfactorily

³⁵ Carolyn Snyder, *Paper Prototyping: The Fast and Easy Way to Design and Refine User Interfaces* (Amsterdam: Morgan Kaufmann Publishers, 2003), 330.

- whether password-protected areas were easy to log into and use
- whether errors occurred, and, if they did, whether appropriate and helpful error messages were encountered
- whether product information was sufficient, or there was too little or too much information
- the ease with which they completed ecommerce transactions
- whether or not the user would return to the web site
- whether or not the user would recommend the web site to others
- suggested improvements

Using a ranking system with areas for comments gives you valuable data. You can analyze the ranked responses, and use the comments and explanations for further helpful insight.

Analyze test results.

You may decide that it is necessary to meet with your entire web site project team following each testing session to discuss what worked, what didn't, and what the possible solutions may be.

Concentrate first on high-level functionality issues, such as global navigation, links, and page layout; then focus on specific areas and recommendations.³⁶

You may also wish to create a report on the results and recommendations to present to the designers and developers and, if applicable, to clients.

Summary

This chapter emphasized the importance of checking your work and testing the site throughout every phase of its development and long-term maintenance. There were checklists to help you test your designs, content, and markup; procedures for cross-browser, cross-platform testing; and discussions on testing labels, functionality, security, accessibility, and usability.

³⁶ Kelly Goto and Emily Cotler, *Web ReDesign 2.0: Workflow that Works* (Indianapolis: New Riders / Berkeley: Peachpit Press, 2005), 230.

What's Next?

If you've enjoyed these chapters from *Deliver First Class Web Sites: 101 Essential Checklists*, why not order yourself a copy?

Make sure that you deliver web sites of the highest standard by following the guidelines in this book. You'll cover all the bases of each of your website projects with this book. *Deliver First Class Web Sites: 101 Essential Checklists* is a comprehensive coverage of current best practices in web design, containing over 100 checklists with 500+ individual checkpoints covering design, usability, project management, SEO and more.

In the remaining chapters, you'll find checklists for

- ❑ Project planning
- ❑ Usability
- ❑ Design
- ❑ Information Architecture
- ❑ Navigation
- ❑ Accessibility
- ❑ Project handover
- ❑ And a whole lot more...

As an added bonus, you'll also have access to downloadable PDF versions of all the checklists included in the book, so you'll be able to print and use the checklists for all your web site projects!

[Order Now and Get it Delivered to your Doorstep!](#)



Index

Symbols

&, XHTML, 172

A

accessibility testing, 276–282
 browsers, 281–282
 forms, 282
 JavaScript, 280
 navigation, 281
 plugins, 280
 tables, 280
accessible web sites, 183–206
alignment, design, 248
alt attribute, XHTML, 171
ampersand, XHTML, 172
ANCHOR pseudo-class rules, CSS, 174
animations, 196–197
archiving, 300–302

B

backing-up
 automated processes, 43
 checking, 293–294
 maintenance, 310–311
 post-launch follow-up, 306
banning by search engines, SEO, 236–237
best practices/standards
 coding, 157–181
 CSS, 173–181
 learning from, 9
 markup, 158–169
 XHTML, 169–173
bookmarking, web site usability, 68–69
borders, CSS, 173–174
brainstorming, design, 258–261

brand protection
 domain names, 21–22
branding, consistency, 26
broken content, internal page navigation, 146
browsers
 accessibility testing, 279–282
 problems, CSS, 176–177
 testing, 269–271
 web site usability, 71–73, 131–132
budgets/budgeting, 21–32
bug tracking/fixing
 post-launch follow-up, 304
 testing, 267–268

C

cache clearing, testing, 274
calendar, keyphrases, SEO, 244
Cascading Style Sheets (CSS)
 ANCHOR pseudo-class rules, 174
 best practices/standards, 173–181
 browser problems, 176–177
 optimizing, 213–219
 SEO, 234
case studies, learning from, 6
CGI (*see* Common Gateway Interface)
changing over time, 4–5
checking
 (*see also* errors; launch preparation)
 automated software, 292–293
 fresh eyes, 291
 functionality, 292–293
 legal issues, 292
 licensing, 318
 links, 244, 292–293
 optimizing, 290
 orphaned pages, 318
 PDF files, 293
 search features, 317

- SEO, 290, 316
 - server-side requirements, 293–294
 - validation errors, 290
 - cloaking, SEO, 237
 - closing elements, XHTML, 170
 - CMS (*see* content management system)
 - coding, best practices/standards, 157–181
 - color, 79–93
 - accessible web sites, 195–196
 - complementary, 84
 - contrast, 273
 - CSS, 87–88, 218
 - design, 260
 - monitor settings, 272–273
 - schemes, 81–85
 - combo boxes, forms, 199
 - comments
 - CSS, 173
 - markup, 211
 - XHTML, 172–173
 - Common Gateway Interface (CGI)
 - web hosting, 24
 - complementary color, 84
 - connection speeds
 - navigation, 129–130
 - web site usability, 70
 - consistency, 10
 - alignment, 251–252
 - branding, 26
 - color, 84–85
 - fonts, 255–256
 - marketing, 26
 - content management system (CMS), 50–58
 - content, web site, 33–44
 - accessibility testing, 278
 - checking, 291
 - CMS, 50–58
 - design, 260
 - inventory, 34–35
 - managing, 45–59
 - navigation, 117–132
 - proofreading, 278–279
 - separating from presentation, 163–166, 185, 234
 - sitemaps, 148–150
 - software, 30–32
 - costs
 - researching, 8
 - CRM (*see* customer relations management)
 - cropping images, 40
 - CSS (*see* Cascading Style Sheets)
 - customer relations management (CRM), data collection, 322
- ## D
- data collection, 319–323
 - delegating, 3–4
 - design, 247–263
 - basic principles, 248–254
 - basis, 260
 - brainstorming, 258–261
 - color, 260
 - finalizing, 261–262
 - flexibility, 261
 - fonts, 254–258
 - prototyping, interactive, 262
 - simple, keeping it, 261
 - whitespace, 248
 - diagramming, organizing content
 - structure, 100–104
 - digital certificates, testing, 275
 - directories, organizing, 109–111
 - displays, testing, 271–273
 - DOCTYPE, markup, 158–159, 208
 - documentation, handover, 296–297
 - domain names
 - brand protection, 21–22
 - branding, 20
 - keywords, 231–232
 - redirecting, 22

TLDs, 21–22
trademarking, 22
downloads, testing, 274
duplications elimination, search features, 154

E

ecommerce, 325–329
 information architecture, 106–108
 PayPal, 30
 shopping cart, 75, 106–108, 320, 326–329
email, web hosting, 25
embedded scripts/functions
 markup, 164
 testing, 273
errors
 (*see also* checking)
 CSS, 290
 error pages, testing, 274
 user, 75–77
external scripts/functions, testing, 273

F

faking markup, 167–169
feedback, post-launch follow-up, 305
filename extensions, URIs, 114–116
filenames, SEO, 112–113
final checks, launch preparation, 289–294
find features (*see* search features)
findability, information architecture, 97–98
Flash
 navigation, 130, 192–193
 SEO, 236
fonts
 consistency, 255–256
 sizes, 71
forms
 accessible web sites, 199–202

 testing, 276, 282
forums abuse, SEO, 241–242
fragment identifiers, XHTML, 171
frames
 accessible web sites, 186–187
 web site usability, 71–72
fresh eyes, checking with, 291

G

GIF format, images, 41–42
goals, initial, 19–21
GoogleAdwords, advertising, 27
graphics (*see* images)
Gutenberg Project, navigation, 125–126

H

hacking
 testing, 275–276
 URL hackers, 110
handover, 296–302
hierarchies, web site, 101–102
hosting, web, 22–25
HTML
 compression, 222
 cross-links, SEO, 233
 editors, markup, 162
 HTML-stripping software, 38
 text navigation, SEO, 233
 tidying-up, 38
HTTP requests, minimizing, 208–209
hyperlinks (*see* links)

I

images
 accessible web sites, 196–197
 backing-up, 43
 captions, 224
 cropping, 40
 CSS, 178–179
 formats, 41–42
 graphics programs, markup, 162

- licensing, 313
 - navigation, 129–130
 - optimizing, 223–225
 - resizing, 43
 - thumbnails, 40, 224
- index page, web site usability, 62–64
- indexes, site, 150
- information architecture, 95–116
 - findability, 97–98
- initial goals, 19–21
- initial questions, 17–32
- integrity, markup, 167–169
- internal page navigation, 145–147
- inventory, content, website, 34–35

J

- JavaScript
 - accessibility testing, 280
 - external, 220, 234
 - navigation, 130
 - optimizing, 219–221
 - SEO, 234
- JPG format, images, 42

K

- keyphrases, SEO, 228–229
- keywords
 - domain names, 231–232
 - SEO, 228–229
 - stuffing, SEO, 237, 239

L

- launch preparation, 289–302
- layouts
 - checking, 291
 - web copy, 38
- learning from
 - best practices/standards, 9
 - case studies, 6
 - others,

- legal issues
 - accessible web sites, 317
 - legal information, 292
 - legal policies, 11
 - trademarking, 22
- licensing, maintenance, 313
- line breaks, XHTML, 172

links

- absolute, 164
- automatic creation, 313
- checking, 292–293
- checking incoming links, SEO, 244
- directory default names, 212–213
- reciprocal links campaigns, 228, 309–310
- relative paths, 212
- SEO, 241–242, 244
- testing, 273–274

- listed, getting (*see* search engine optimization)

load times

- (*see also* speed)
- fast-loading web pages, 69–71
- testing, 271

- local/section navigation, 138–145

logos, 10

- branding, 25–26
- design, 260

- long term filenames, 113

M

maintenance

- post-launch follow-up, 310–319
- resources, 306
- SEO, 242–244
- web site, 295
- web site content, 13–14

maps/mapping

- idea maps, 100–101
- image maps, 197
- mind maps, 100–101

- sitemapping tools, 104
- sitemaps, 147–150
- marketing
 - (*see also* search engine optimization)
 - advertising, 27–28, 228
 - budgets/budgeting, 27–28
 - post-launch follow-up, 306
 - promoting, 308–310
 - reciprocal links campaigns, 228, 309–310
 - training, 28–29
- markup
 - best practices/standards, 158–169
 - checking, 290
 - HTTP requests, minimizing, 208–209
 - keyphrases, 230–232
 - optimizing, 207–221
 - semantic, 187–192
 - validation, 159, 187–192, 268, 312
 - W3C standards/recommendations, 184–185
- menu, navigation, 130–131
- meta elements, SEO, 161
- monitors, testing, 271–273
- multimedia
 - accessible web sites, 198
 - optimizing, 224–225

N

- navigation, 117–155
 - accessibility testing, 281
 - accessible, 192–195
 - accessing, 119–132
 - Gutenberg Project, 125–126
 - local/section, 138–145
 - skip links, 193–194
 - SSI, 128
 - supplemental, 147–155
 - web site usability, 64–65

- nesting
 - tables, 203
 - XHTML, 169–170

O

- Opera, accessibility testing, 279
- optimizing, 207–225
 - (*see also* search engine optimization)
 - checking, 290
 - page optimization, testing, 271
 - PDF files, 224–225
 - server responses, 222
- organizing directories, information architecture, 109–111
- organizing for success, information architecture, 98–104
- orphaned pages, checking, 318
- oversubmitting, SEO, 240

P

- page optimization, testing, 271
- paper prototyping
 - brainstorming, 259
 - design, 261
- password-protecting, soft launch, 295
- PayPal, ecommerce, 30
- pay-per-click (PPC) advertising, 27, 228
- PDF files
 - accessible, 198
 - checking, 293
 - optimizing, 224–225
- periodic processes, 317–318
- photos (*see* images)
- pictures (*see* images)
- planning, 2–15
 - post-launch activities, 304
 - SEO, 228
- platforms, testing, 269–271
- plugins
 - accessibility testing, 280
 - web site content, 37

plurals, SEO, 229
popups, SEO, 235
post-launch follow-up, 303–323
PPC advertising (*see* pay-per-click advertising)
predictability, ensuring, 72–75
presentation, separating from content, 163–166, 185, 234
privacy policy, web site content, 37
project management, tools, 14–15
promoting (*see* marketing: search engine optimization)
promoting a business, website purpose, 20
proofreading, web site content, 278–279
prototyping, interactive, design, 262
prototyping, paper
 brainstorming, 259
 design, 261
 information architecture, 105

Q

quality assurance
 documentation, 297
 post-launch follow-up, 306
questionnaires, usability testing, 286–287

R

radio buttons, forms, 199
readability
 color, 92
 content, web site, 278–279
reciprocal links campaigns, marketing, 228, 309–310
redirecting
 domain names, 22
 URLs, 113
redundancy, markup, 211
regression testing, 268

relative paths
 links, 212
 markup, 212
repetition, design, 248
reviewing, testing, 276–279
Rule of Seven, findability, 97

S

schedules, 2–15
search engine optimization (SEO), 28, 227–245
 (*see also* marketing)
 automated software, 240
 balancing HTML text and graphics, 233
 banning by search engines, 236–237
 checking, 290, 316
 checking incoming links, 244
 cloaking, 237
 CSS, 234
 doorway pages, 237
 endless loop, 235–236
 filenames, 112–113
 Flash, 236
 forums abuse, 241–242
 getting listed, 238–241
 HTML cross-links, 233
 in a nutshell, 228
 JavaScript, 234
 keyphrases, 228–232
 keyword stuffing, 237, 239
 maintenance, 242–244
 search-engine friendly, 232–236
 URLs, 112–113
search engine submissions, 26–27
search features
 checking, 317
 global navigation, 151–154
 testing, 274
secure socket layers (SSL)
 testing, 275

-
- web hosting, 25
 - security, testing, 274–276
 - SEO (*see* search engine optimization)
 - server hardware, testing, 269
 - server needs, web hosting, 24–25
 - server response, optimizing, 222
 - server-side includes (SSI)
 - navigation, 128
 - web hosting, 25
 - server-side redirect techniques
 - SEO, 233
 - URLs, 112
 - shopping (*see* ecommerce)
 - shopping cart
 - abandonment, 320
 - features, 326–329
 - placement, 75, 106
 - simple, keeping it
 - design, 261
 - pre-planning, 2–3
 - web copy, 37–39
 - site indexes, 150
 - sitemapping tools, organizing content structure, 104
 - sitemaps
 - creating, 147–150
 - maintenance, 313
 - SEO, 235
 - sitemapping tools, 104
 - web site content, 148–150
 - sketching, design, 259
 - skip links, navigation, 193–194
 - soft launch, launch preparation, 294–295
 - software
 - budgets/budgeting, 30–31
 - checking, 292–293
 - HTML-stripping, 38
 - images, 31
 - SEO, 240
 - testing, 266
 - web design, 30–32
 - speed
 - (*see also* time considerations)
 - bandwidth, 23
 - connection speeds, navigation, 129–130
 - connection speeds, web site usability, 70
 - fast-loading web pages, 69–71
 - load times, testing, 271
 - server responses, 222
 - web hosting, 23
 - web site usability, 23
 - splash pages, SEO, 232–233
 - SSI (*see* server-side includes)
 - SSL (*see* secure socket layers)
 - staff, orientating, 307–308
 - standards, W3C standards/recommendations, 157–181
 - static redirect pages, SEO,
 - style guides
 - documentation, 296–297
 - planning, 9–12
- ## T
- table of contents, local/section navigation, 141–142
 - tables
 - accessibility testing, 280
 - accessible web sites, 202–205
 - target audience, 17–19
 - testing, 265–287
 - (*see also* accessibility testing)
 - CSS, 177
 - reviewing, 276–279
 - usability, web site, 276–287
 - thesaurus, SEO,
 - three-click rule, web site usability, 102
 - time considerations
 - (*see also* speed)
 - changing over time, 4–5
-

- launching, 295–296
- load times, 69–71, 271
- schedules, 13
- SEO, 243
- updating web site content, 13–14
- tools
 - communication, 14–15
 - hacking, 275–276
 - project management, 14–15
 - site-monitoring, 315
 - style guides,
 - testing, 266
 - web authoring, 49–50
- tooltips, markup,
- trademarking
 - domain names, 22
 - legal issues, 22
- training
 - ecommerce, 308
 - marketing, 28–29
 - post-launch follow-up, 304
 - web hosting, 24

U

- updating
 - maintenance, 311–312
 - site indexes, 150
 - web site content, 13–14
- URIs (Universal or Uniform Resource Identifiers)
 - filename extensions, 114–116
 - information architecture, 112–116
- URLs (Universal or Uniform Resource Locators)
 - information architecture, 112–116
 - minimizing, 212–213
 - publishing,
 - redirecting, 113
 - “hackers”, organizing directories, 110
- usability, web site, 61–78
 - browsers, 71–73, 131–132

- color, 85–92
- errors, user, 75–77
- index page, 62–64
- links, 67–68, 73–74
- testing, 276–287
 - user abilities, 71–72
 - user focus, 64–78
- user feedback, post-launch follow-up, 305
- user-based navigation, 117–132
- user-centered global navigation, 132–135
- users, test, 282–287

V

- validation
 - CSS, 268
 - CSS errors, 290
 - markup, 159, 268, 312
- visiting web sites
 - other, 6–7
 - reasons, 18
- visual cues
 - color, 89–92
 - navigation, 129
- visual identity, design, 250
- voice browsers, accessibility testing, 281–282

W

- W3C (*see* World Wide Web Consortium)
- Web Content Accessibility Guidelines (WCAG)
 - accessible web sites, 184–185
- web hosting, 22–25
- whitespace
 - CSS, 173–174, 213
 - design, 248
 - markup, 210
 - XHTML, 172

World Wide Web Consortium (W3C),
standards/recommendations, 157–
181
accessible web sites, 184–185

X

XHTML, best practices/standards, 169–
173