CHAPTER 3

Site Planning

When you complete this chapter, you will be able to:

◎ Understand the Web site development process

◎ Create a site specification

◎ Identify the content goal

◎ Analyze your audience

◎ Build a Web site development team

◎ Create conventions for filenames and URLs

◎ Set a directory structure

◎ Create a site storyboard

◎ Publish your Web site

◎ Test your Web site

A good Web site design requires a detailed initial planning phase. Before starting to code your site, pick up a pencil and paper and sketch out your site design. Creating the stylistic conventions and conceptual structure of your site beforehand saves time during development. Whether you are creating a single personal Web site or working on a professional Web development team, you save time and improve quality by thoroughly planning the site design and development process before you start creating it. This chapter walks you through planning and building a framework for your site, resulting in more efficient development when you build your Web site.

Understanding the Web Site Development Process

What are your objectives for building a Web site? You may want to gain visibility, provide a service, sell a product, create a community, attract new customers, or disseminate information. Although the content may vary, a good project outcome requires a sound development process to ensure that you have valid and achievable goals for your site.

A good project plan encompasses all stages of the project and is accessible to everyone involved. The complexity and depth of each stage of the planning process can vary based on the complexity of the site. In commercial Web site development, a project manager controls and disseminates the project plan using tools such as Microsoft Project to create charts and track all phases of production.

You can choose from many types of models of the Web site development process and use one as a framework for planning your site development. Figure 3-1 shows a typical high-level project plan. The stages in this illustration would match scheduled calendar dates that are milestones in the project plan. The life-cycle of the project is the complete project plan from inception to completion. Stages can overlap at different points in the process. As the project evolves, design changes must decrease to ensure a successful implementation, meaning that the beginning of the project is where most design experimentation should occur. As the project progresses, the design must become stable for successful completion. Major design changes cannot be introduced late in the project without significant rework and risk of extending the project schedule.
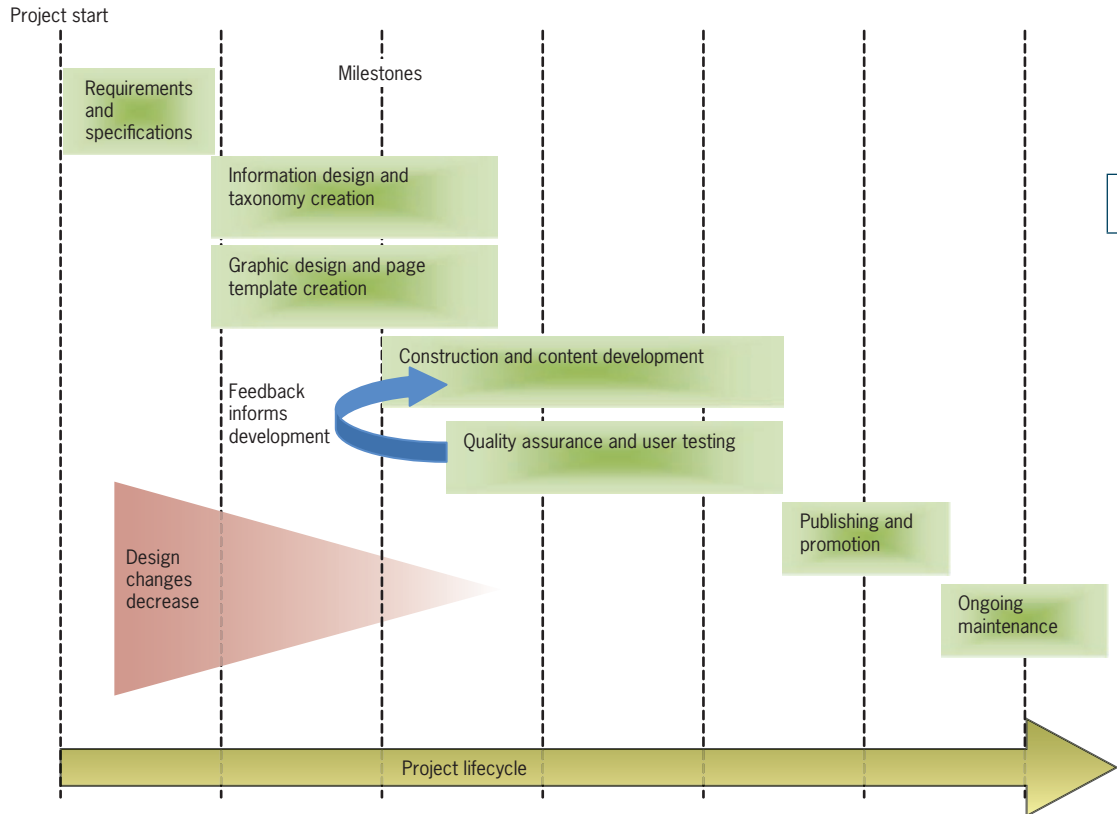
Project start

Milestones

Requirements and specifications

Information design and taxonomy creation

Graphic design and page template creation

Construction and content development

Feedback informs development

Quality assurance and user testing

Publishing and promotion

Ongoing maintenance

Design changes decrease

Project lifecycle

Figure 3-1 Web development project lifecycle

Requirements and Specification

In this stage of a Web development project, the client presents the requirements for the Web site. **Requirements** are the list of customer needs, such as search capability, tabbed menu navigation, specific color and branding requirements, or anything else that will create the desired outcome for the site. The Web project team must analyze these requirements for viability and then break them down into tasks. This is a good time to assess the talents of the team members to make sure all requirements can be met. For example, if a particular programming language is required, then a team member must have those skills to meet the requirement. The Web project team works with the client during this stage to analyze and define the audience. After analyzing and defining requirements and determining the user profile, the team prepares a project

specification that contains the design requirements, page layout sketches, audience definition, and technical requirements. As a Web designer, the specification will guide you to the next stages of the project.

Information Design and Taxonomy Creation

In this stage, user analysis guides the designers as they prepare and test different organizations of the site content. The goal of this stage is to structure the site content in a way that is the most meaningful and easiest to navigate by the intended user. During this stage, the taxonomy of the site information is developed. A **taxonomy** is a classification and naming of content in a hierarchical structure. The taxonomy of the site directly translates to the navigation through the top-level content topics down to individual pieces of information that the user is looking for. The taxonomy is often reflected in topic section names and in the navigation and menu system of the site.

Graphic Design and Page Template Creation

At the same time designers are testing the organization of the site information, they prepare design sketches and page mockups to represent page layouts that will be used in the site. All page layouts start with a mockup that is usually just a sketch of the desired design. Designers can submit the page layout mockups to the Web site stakeholders for discussion and critique. Generally, you create a mockup for each page layout in the Web site. A sketch of a proposed design, as shown in Figure 3-2, indicates the general layout of a Web site home page.
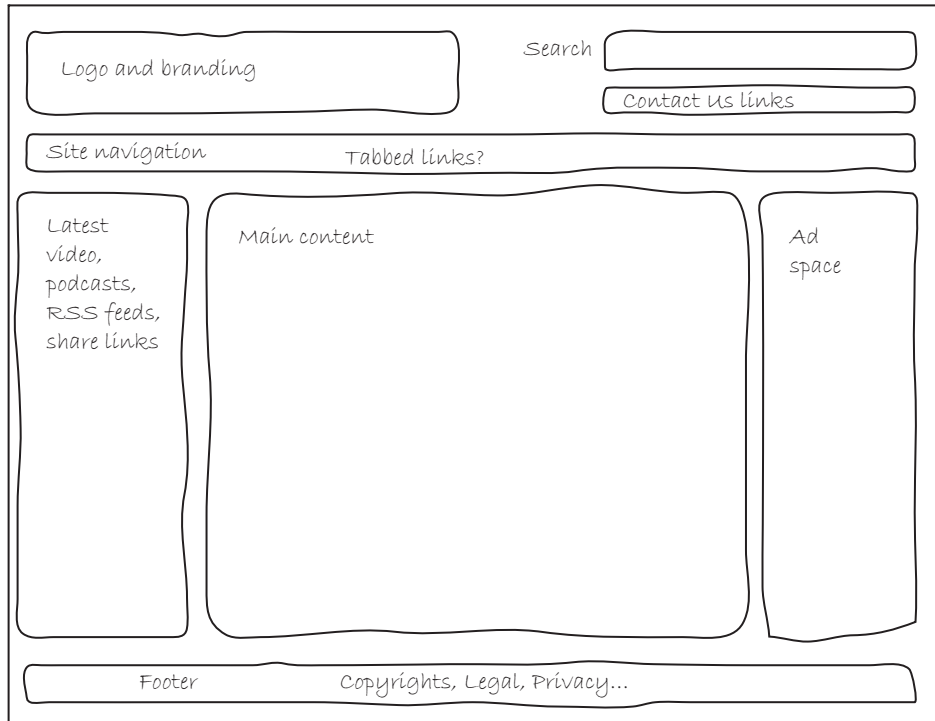
Figure 3-2 Web page mockup

These mockups can be easily edited and changed based on feedback and input from the design team. As the design becomes more stable, the mockup can evolve to a more refined state, often called a **wireframe**. Wireframes show a more complete version of the page designs, often including navigation elements, search functions, advertising space, and other similar elements. The wireframes offer stakeholders a more complete view of what the final design will look like. Designers also use wireframes to gain insight and reactions from content developers and software engineers and to test design changes. Remember that it is always easier to make changes to a page mockup or wireframe than it is to a Web site once you have started coding it.

Figure 3-3 shows a wireframe that builds on the page design articulated in the mockup sketch in Figure 3-2.
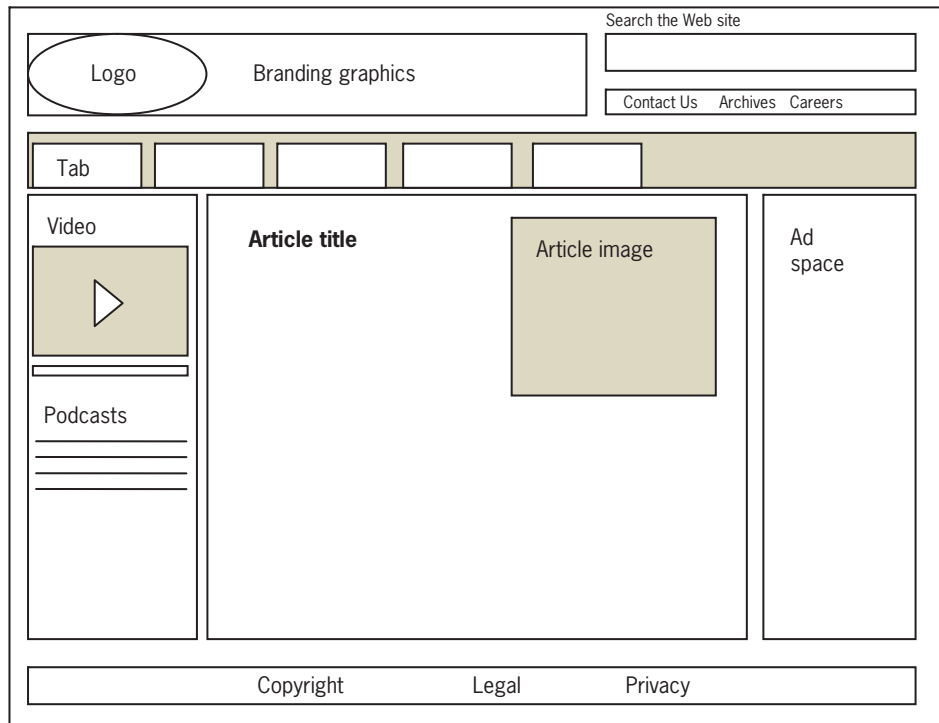
Figure 3-3 Sample wireframe for page layout

You can find freeware wireframe software tools at the following Web sites:

- Pencil (*http:// pencil.evolus.vn/ en-US/Home.aspx*)

- Gliffy (*www.gliffy.com/ wireframe-software*)

Construction and Content Development

When the design stage is mostly complete and page designs are stable, the construction stage can begin. This stage encompasses all of the technical development of the site, including page coding and validation, application development, and content preparation. Some testing occurs during this stage of loading content in page templates and evaluating the performance of applications or multimedia.

Quality Assurance and User Testing

As the site construction nears completion, the quality assurance and user testing phase validates the design of the site. The development team performs various tests for cross-browser compatibility, accessibility to all users, and connectivity at different bandwidths. They test links and all user interfaces, data forms, and multimedia technologies. Usability testing ensures that users can access content, navigate through the site, and understand the taxonomy.

Publishing and Promotion

During this stage, the site is published to the Web or the organization intranet, and the client begins to publicize and promote the site. This includes making the Web site address available in all collateral media, such as print and broadcast; advertising on other Web sites; registering with search engines; distributing press releases; and starting publicity campaigns.

Ongoing Maintenance

This stage begins at the moment the site goes live, as Web content must be updated and kept fresh to remain vital. New sections of content may be added that restart certain phases of the project lifecycle, such as new audience definitions, designs for new content areas, or the design and development of new interface elements or interactive features.

Creating a Site Specification

Start your planning by creating a **site specification**; this is the design document for your site. If you completed the Individual or Team Case Project at the end of Chapter 1, you created a basic draft of a project proposal. You can use some of that information in your site specification. After you read this chapter, you will be able to answer additional questions about your site. You can return to the site specification as you build your site to help maintain your focus. If you are providing a Web site design to a client, the site specification is the first document the client sees that establishes the basic site design. Answer the following questions in your site specification:

You can read a sample site specification in Chapter 12.

- Who is the client for the Web site? This is not the user, but the person who has employed you to build a Web site. Are you creating a personal site (in which case, you are the client), or are you part of a design team working independently or for a corporation or nonprofit organization? What do you or the client's company or organization hope to gain from creating and maintaining a Web site?

- Can you write a mission statement that succinctly states the site's focus and goals?

- What are the requirements for the Web site? Look to your client for their list of wants. Explicitly state the required functions that you want the site to contain.

- Are the requirements feasible? Do you or your team have the necessary technical and editorial skills to build the site that the client requires?

- How will you judge the success of the site? What are the factors you can use to assess the effectiveness of the site?

- Who is the target audience? What are some common characteristics? How can you find out more about your target audience?

- What are the limiting technical factors affecting your site?

- What is the budget? What is the schedule or target milestone dates? Are the dates realistic and achievable?

- Is this a new site or an upgrade to an existing site? If this is a site upgrade project, what can be learned from the first version of the site?

Identifying the Content Goal

Consider carefully what type of site you are building. What you and the design team want the Web site to accomplish and what your users want from your site may differ. For example, designers and other stakeholders are often more concerned with the look than the feel. Don't let this deter you from advocating for the user. Your users probably care more about how quickly they can find information. Adopt your user's perspective, and let your user analysis guide your content decision. Think about the type of content the Web site will provide and what will give it the greatest value to your user. Look at your search and navigation capabilities carefully to ensure ease and direct access. Look to the Web for examples of how best to present, organize, and focus your content. The following types of Web sites demonstrate ways to focus your content.

- *Billboard*—These sites establish a Web presence for a business or commercial venture. In many cases, they are informational and offer limited content, acting as an online business card or brochure rather than offering Web-based interaction. Many smaller businesses build this type of site first and then expand as necessary, adding functions when needed.

- *Publishing*—Every major newspaper and periodical publishes both to print media and to the Web. These Web sites are some of the most ambitious in breadth and depth of content, often containing multiple levels of information with many page designs. Many publishing sites use content management systems (CMS) to dynamically create Web pages, drawing content from the same databases that produce their paper-based versions. This allows their authors to write the article once, but have it published to multiple destinations, such as the daily newspaper and the Web site.

- *Portal*—Portals act as gateways to the Web and offer an array of services including searching, e-mail, shopping, news, and organized links to Web resources. Many major search engines have been converted into portals to attract more users. These sites are often heavy with advertising content, which is their main source of revenue.

- *Special interest, public interest, and nonprofit organization*—These sites include news and current information for volunteers, devotees, novices, a specific audience, or the general public. Public-service Web sites contain links, information, downloadable files, addresses, and telephone numbers that can help you solve a problem or find more resources. Nonprofit organizations can state their manifestos, seek volunteers, and foster grassroots virtual communities.

- *Blog*—Short for "Weblog," a blog is a personal Web page that reflects the personality and interests of the author. No matter what your interest, a community of *bloggers* (blog authors) on the Web is devoted to it. Many blogs are personal diaries or commentaries on life. Most blogs are published with tools that can archive content, provide comment threads that allow visitors to comment on blog posts, and offer built-in page designs. WordPress (*www.wordpress.com*) is a popular example of this type of software. There are literally millions of blogs on the Web. You can find listings at *www.blogcatalog.com* and *www.blogarama.com*.

- *Social networking*—Sites such as MySpace, Facebook, Bebo, and LinkedIn are the most visible of these types of virtual communities, which allow users to post profiles and connect with friends and family. Users can share comments and exchange messages with their friends, play games, take surveys, and post photos, videos, and links to interesting sites. The popularity of social networking sites has risen sharply in the past few years. They are often the reason many people use the Internet.

- *Wikis*—A wiki is a type of online database that accepts contributions from multiple authors. Wikis are collaborative Web sites that allow contributors to use wiki software to easily edit information and create linked Web pages. Wikis allow any user to comment on or change another user's entries. Wikipedia is the most visible example of a wiki, but they are appropriate anywhere collaboration and sharing of content is needed, and are used in both academic and commercial environments.

- *RSS (Real Simple Syndication)*—This is a service provided by Web sites. More technically, RSS is a format for Web feeds that automatically update users who have subscribed for this service. RSS feeds usually contain headlines or summaries of content, which are read using a software tool called an RSS Reader. RSS Readers are built into the major browsers. With RSS feeds, any author updates to a Web site are automatically transmitted to subscribers.

- *Virtual gallery*—The Web is a great place to show off samples of all types of art and design. Photographers and artists can display samples of their work; musicians and bands can post audio files of their songs; writers can offer sections of text or complete manuscripts. However, keep in mind that any copyrighted material you display on a Web site can be downloaded to a user's machine without your permission. As a solution to this problem, software companies such as Digimarc (*www.digimarc.com*) offer digital watermarking technology that lets artists embed digital copyright information in their electronic files as a deterrent to piracy of proprietary content. This information cannot be seen or altered by the user.

- *E-commerce, catalog, and online shopping*—The Web as shopping medium continues to expand as more users improve their Internet access and learn to trust the security of online commerce. Web commerce competes successfully with traditional retailing, offering many advantages over mail-order shopping, such as letting the customer know immediately whether an item is in stock. Other types of commerce on the Web include stock trading, airline ticketing, online banking, auctions, and more. Many software vendors offer turnkey systems that can be integrated with existing databases to speed the development of a commerce site. A good electronic commerce (e-commerce) site provides users with quick access to the item they want, shopping carts and wish lists to store their choices, detailed product descriptions, and easy, secure ordering.

- *Product support*—The Web is a boon to consumers who need help with a product. Manufacturers can disseminate information, upgrades, troubleshooting advice, documentation, and online tutorials through their Web sites. Companies that provide good product support information on the Web find that the volume of telephone-based customer support calls decreases. Software companies especially benefit from the Web; users can download patches and upgrades and use trial versions of software before they buy.

- *Intranet and extranet*—Private Web sites are often hosted on a company intranet or extranet. An **intranet** is a smaller, limited version of the Internet on a company's private local area network (LAN), accessible only to those who are authorized to use their network. Many companies have telecommuting employees who need access to company policies, documentation, parts lists, pricing information, and other materials. These employees can be reached via an **extranet**, which is a part of the private intranet extended outside the organization via the Internet. Web sites on an intranet or extranet are typically developed by companies for use by employees, customers, and suppliers. Many organizations mandate a particular browser for employee use, making the Web designer's job a little easier, because they only have to code and test for one browser.

Analyzing Your Audience

If possible, analyze your audience and produce an **audience definition**, a profile of your average user. If you are building a new site, work from your market research, look at sites with content similar to yours, and try to characterize your average user. If you have an existing user base, contact your typical users and try to answer the following questions:

- What do users want when they come to your site? Can they easily find desired content?

- How can you initially attract visitors and entice them to return?

- What type of computer and connection speed does your typical visitor have?

Though your users may fit no common profile, you can gather information about them in a few ways. One way is to include

an online feedback form in your site. Figure 3-4 shows a sample online survey from the State of Maine Web site (*www.maine.gov/portal/survey.html*).

The survey asks users about their experiences visiting the Web site. It uses both scaled (or rated) and open-ended questions to elicit a variety of responses from the user concerning the visual and information design of the site.

Figure 3-4 Sample user feedback form

If you cannot survey your users, or if you feel you are not getting good survey results, try to adopt a typical user's perspective as you define your audience. Here are some questions to consider:

- Who are the typical members of your audience? Are they male or female? Do they have accessibility issues? What is their level of education? What is their reading and vocabulary level? What is their level of technical aptitude?

- Why do people come to your site? Do they want information? Do they want to download files? Are they looking for links to other Web sites?

- Do you have a captive audience, such as a base of loyal customers that want up-to-date information? Are you designing for an intranet, where users are employees of an organization?

- If users are unfamiliar with the site, will they know what you offer?

- How often will users return to your site? Why would they come back?

- What computing platform do your users have? What is their typical connection speed? What type of browser do they use? If you are on an intranet, does it use standards for browsers, connection, and screen resolution?

- Whose skills do you need to build the site? Who will create the graphics, code the pages, and write the text? Do you have the talent and economic resources that you need? Will the results meet the expectations of your users?

Refine your content and presentation even after your site is built and running. Continue soliciting user feedback to keep your site focused and the content fresh.

You can set up a free Web-based survey with SurveyMonkey (*www.survey-monkey.com*).

Using Web Analytics

Web analytics are statistics that are gathered by Web servers and then analyzed. Web servers track and record various usage and traffic statistics in files called server logs. Reporting tools can aggregate and analyze the data in these files to help you learn about your users and their activity on your Web site. The data gathered by the Web server contains relevant details about your visitors, including how they came to your site, whether by bookmark, search engine, or by clicking a link on another Web site. If visitors used a search engine, you can see what term they entered to find your site. You can see what pages are most visited, and even see an approximate view of where your users are geographically located, based on their identifying Internet address.

Figure 3-5 shows a sample of statistics from W3 Counter (*www.w3counter.com*), a free Web analytics tool that you can add to any Web site. This figure shows two important Web log statistics: page views and unique visitors. A **page view** is the number of times a page is viewed by a user, which could be multiple times by the same user. The unique visitor statistic tracks the number of unique visitors to a site, no matter how many times they access the same page. So the page views tell you which pages are the most popular with your users, while the unique visits tell you how many different users are visiting your site.
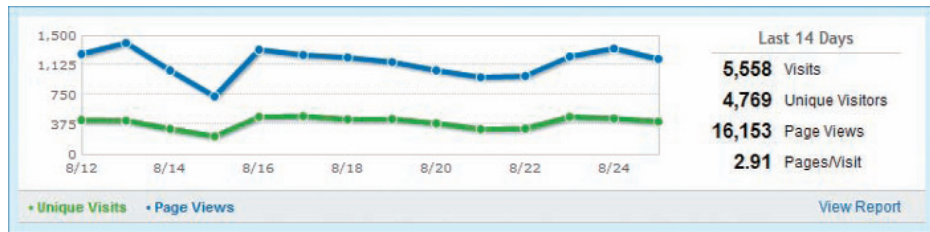

Figure 3-5 Sample Web analytics results

In addition to viewing how many times users accessed your pages and visited your site, you can also see where your users originated from geographically, as shown in Figure 3-6.

Figure 3-6 Sample geographical results

Web analytics are an important method of determining details about your users and how they interact with and traverse the content areas of your site. Use Web analytics as a guide to enhancing your content and making important pages accessible, but remember that they are not flawless. Many factors can affect the results you will receive. Statistics are just one method of determining user preferences.

Identifying Technology Issues and Accessibility Constraints

Make your best effort to identify any technological limitations or advantages that members of your audience share. As you read in Chapter 2, you have to make assumptions about the user's browser, connection speed, operating system, and screen resolution. Users in developed countries have greater access to the latest hardware technologies and consistent high-speed Internet connections. Many other users throughout the world may be using older equipment. Think about where your users are located and what their technology level might be. Test in different environments and with different technologies to make sure these users can view your site.

You also need to consider the physical capabilities of users that are visiting your site. How do they interact with your content? Are they older users with sight or dexterity issues? Will they need to magnify your pages to read your text? Many types of physical limitations can affect the way people interact with a computer.

When planning a Web site, you can identify accessibility constraints in the following ways:

- Review the WCAG 2.0 and Section 508 accessibility guidelines as needed.

- If you are building a new Web site, plan for accessibility.

- If you have an existing Web site, assess the current accessibility of your content.

- Review case study examples of real-life accessibility implementations.

- Discuss accessibility solutions, authoring tools, and evaluation tools with other Web professionals.

Study your Web statistics and user feedback. If you think your user is the average person browsing the Web, you may have to adopt settings that represent the lowest common denominator to satisfy the widest audience. If you find that your users are savvy about technology and use the latest computer hardware and software, higher resolution and connection speeds may apply. If you are designing an intranet site, you may have the luxury of knowing your users' exact operating systems and browser versions. Whatever the particulars, make sure to design at an appropriate level, or you risk losing visitors.

Identifying Software Tools

Determining the software requirements for your Web site is important during the planning process. Try to choose software that matches the complexity and needs of your site so that you do not end up with a tool that is either underequipped or over-specialized. Simple Web sites, including many student sites, can be built with one of the many shareware and freeware tools (see Table 3-1) that are available on the Web. As your site and skills grow, you might choose to move up to more robust tools such as Adobe Creative Suite or individual Adobe tools such as Dreamweaver (*www.adobe.com*). Microsoft offers Expression Web as its Web site design tool. These tools offer complete coding, design, and site management capabilities. You may also need graphics tools (discussed in Chapter 8), database software, and online credit and shopping programs, based on the skills and talents of the members of your Web site team, as described in the next section.

One popular type of software is **shareware**, programs that you can download and use for a trial period. After the trial period, users can register the software for a relatively small fee compared to commercially produced software. Another type of software already mentioned in this chapter is **freeware**, which is available free of charge or with an optional donation fee if you want to contribute to support the software developers' efforts. Table 3-1 lists freeware HTML development tools.

Development Tool	Platform	URL
Amaya	Windows, Macintosh, Linux	www.w3.org/Amaya
BBEdit	Macintosh	www.barebones.com
Bluefish	Windows, Macintosh, Linux	bluefish.openoffice.nl
Firebug Firefox add-on	Windows, Macintosh, Linux	addons.mozilla.org/en-US/firefox/addon/1843
Kompozer	Windows, Macintosh, Linux	www.kompozer.net
SeaMonkey	Windows, Macintosh, Linux	www.seamonkey-project.org
Trellian	Windows	www.trellian.com/webpage

Table 3-1   List of Freeware Web Site Development Tools

Building a Web Site Development Team

Although one person can maintain small Web sites, larger sites require groups of people filling a variety of roles. Of course, the line between these roles can be blurred, and many aspects of site design require collaboration to solve a problem. The following are examples of the types of talent necessary to build a larger, well-conceived site.

- *Project management*—The project management team is responsible for planning, scheduling, and integrating the many tasks that it takes to create a Web site. They create the milestones for deliverables and balance the staffing resources to keep the project on schedule and within budget. The project manager coordinates communication among team members and keeps the focus on the deliverables promised to the client.

- *HTML developers*—These are the people responsible for creating the HTML code, conforming to standards, validating code, troubleshooting the site, and testing the site across different operating systems and Web browsers.

- *Designers*—Designers are the graphic artists responsible for the look of the site. They use graphic design software such as Adobe Photoshop or Adobe Fireworks. Designers are responsible for the wireframes, page template design, navigation icons, color scheme, and logos. If your site uses photographic content, the designers are called upon to prepare the photos for online display. They might also create animations and interactive content using Adobe Flash.

- *Writers and information designers*—Writers prepare content for online display, including taxonomies, hypertext linking conventions, and navigation paths. In addition, many writers are responsible for creating a site style guide and defining typographic conventions, as well as consistency, grammar, spelling, and tone. They also work closely with the designers to develop page templates and interactive content.

- *Application developers*—Developers write the software programs and scripts you need to build interaction into your site. They may write a variety of applications in different programming languages for user interaction or write back-end applications that interact with a database.

- *Database administrators*—The people who are responsible for maintaining the databases play an important role in commercial Web sites. Databases store all the information for customer transactions and e-commerce. Database administrators, application developers, and HTML developers work together when designing front-end forms used to collect data from the user. Database administrators are also responsible for data security backup and data recovery.

- *Server administrators*—Get to know and appreciate the technical people who run your Web server. They take care of the sticky technical issues such as firewalls, ports, internal security, file administration, and backup procedures. Consult with them to determine your Web site's default filename and directory structure. They also can manage the server logs that contribute to your Web analytics reporting to determine how many visitors your site is attracting, where the visitors are coming from, and what pages they like best.

Creating Conventions for Filenames and URLs

Before you sit down at the keyboard, plan the filename conventions for your site. Find out from your system administrator what type of operating system your Web server uses. Typically you develop your Web site locally on a PC or Macintosh and upload the files to the Web server as the last step in the publishing process. If the Web server runs a different operating system from your local development system, any filename or directory structure inconsistencies encountered in transferring your files to the server may break local URL links.

Naming Files

A filename's maximum length, valid characters, punctuation, and sensitivity to uppercase and lowercase letters vary among operating systems, as described in Table 3-2. Note that the **ISO 9660 Standard** is the base file-naming convention designed to work across all operating systems.

Operating System and File System	Filename Conventions
ISO 9660 Standard	Maximum of eight letters followed by a period and a three-letter extension; allowed characters are letters, numbers, and the underscore (_)
Newer PCs: Windows 7, Windows Vista, Windows XP (NTFS), Windows 2000 (NTFS), Microsoft Windows/NT (NTFS)	Maximum of 255 letters, all characters allowed except \ / * " < >	: ?
Older PCs: Windows 98 (FAT32), Windows 95 (VFAT), DOS, and Windows 3.× (FAT file system)	The same as ISO 9660 but with the following additional characters allowed: $ % ' ` - @ ^ ! & [ ] ( ) # This format is also compatible with newer PC operating systems
Newer Macintosh: O/S 8.1 to OS X	Maximum of 255 characters, all characters allowed except the colon (:)
Older Macintosh: Operating systems released before O/S 8.1	Maximum of 31 letters, all characters allowed except the colon (:) This format is also compatible with newer Macintosh operating systems
UNIX	Maximum of 255 letters, all characters allowed except the forward slash (/) and spaces

Table 3-2   File Naming Conventions

Case Sensitivity

If you have an image file named Picture.gif, for example, and you reference that file as , the image is displayed properly on a Macintosh or Windows machine. On a UNIX server, however, the image does not load properly because UNIX is case sensitive; *Picture.gif* and *picture.gif* are recognized as two different files. It is best to use lowercase letters for all filenames, including filenames in your HTML code.

Character Exceptions

As shown in Table 3-2, it is best when naming your files to leave out special characters such as <, >, /, \, &, *, and blank spaces to ensure cross-platform compatibility. Some special characters that may be valid on one operating system will not work on another.

File Extensions

You must use the correct file extensions to identify your file to the browser. HTML text files created in HTML-editing programs commonly end in .htm or .html unless they are generated dynamically by an application. In this case, they may have extensions such as .asp, .php, or others. You also must use the correct file-name extensions to identify image file formats. For example, Joint Photographic Experts Group (JPEG) files must end in .jpg or .jpeg; Graphics Interchange Format (GIF) files must end in .gif; and Portable Network Graphic (PNG) files must end in .png.

Choosing the Correct File-Naming Conventions

It is best to set conventions for your filenames right from the beginning of the Web development process. Create a list of conventions and refer to it frequently. Here are some guidelines to remember:

- Don't use spaces in your filenames; use underscores instead. Instead of *about web design.html*, use *about_web_design.html*.

- Avoid all special characters. Stick to letters, numbers, dashes, and underscores.

- Use all lowercase letters for your filenames.

Default Main Page Name

Every Web site has a default main page that appears when the browser requests the main URL of the site, such as www.google.com rather than a specific file. In this instance, the Web server must decide which file to provide, which is usually the home page of the site. The default is generally *index.html*, but others may apply based on site particulars. Always check with your system administrator to verify the correct main page filename.

Using Complete or Partial URLs

Although you may know that URLs are the addresses you type into your browser to access a site, you may not realize that there are two types of URLs: complete and partial.

Complete URLs

A **Uniform Resource Locator (URL)** is the unique address of a file's location on the World Wide Web. A **complete URL** includes the protocol the browser uses, the server or domain name, the path, and the filename. Figure 3-7 shows an example of a complete URL.
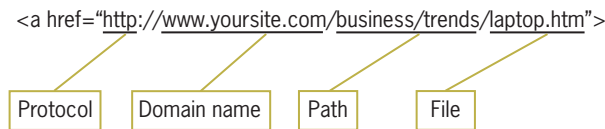

Figure 3-7 Parts of a complete URL

In this example, *http* is the protocol, and *www.yoursite.com* is the domain name. The path shows that the destination file, *laptop.htm,* resides in the *business/trends* folder. Use complete URLs in your HTML code when linking to another Web site.

Partial URLs

Use a partial URL when you are linking to a file that resides on your own computer or server. **Partial URLs** omit the protocol and domain or server name, and specify the path to the file on the same server. Files that reside in the same directory (or folder) need no path information other than the filename. The following code shows an example of a partial URL.

```
<a href="laptop.htm">link text</a>
```

Setting a Directory Structure

You will probably build your Web site on a computer that is different from the computer that hosts your site. Keep this in mind when you are designing the directory and file structure. All of the files for your Web site will need to be transferred from your computer to the Web server that will be hosting your site. Because your files will be transferred to another computer, any URLs you specify to link to other pages in your site must include paths that

are transferable. This is why you should never specify an absolute path in your partial URLs. An absolute path points to the computer's root directory, indicated by a leading (forward) slash in the file path:

```
/graphics/logo.gif
```

If you include the root directory in your partial URLs, you are basing your file structure on your development machine's file system. If the files are moved to another machine, the path to your files will not apply, and your site will include links that do not work because the browser cannot find the files.

To avoid this problem, use relative paths. Relative paths tell the browser where a file is located relative to the document the browser currently is viewing. Because relative paths are not based on the root directory, they are transferable to other computers.

Using a Single Folder Structure

One easy way to ensure that all your path names are correct is to keep all of your HTML and image files in the same directory. Because all files are kept together, the only information you need to put in the src or href attribute is the filename itself. In Figure 3-8, User2 has a simplified directory structure. To reference the file logo.gif, User2 adds the following code in one of the HTML files:
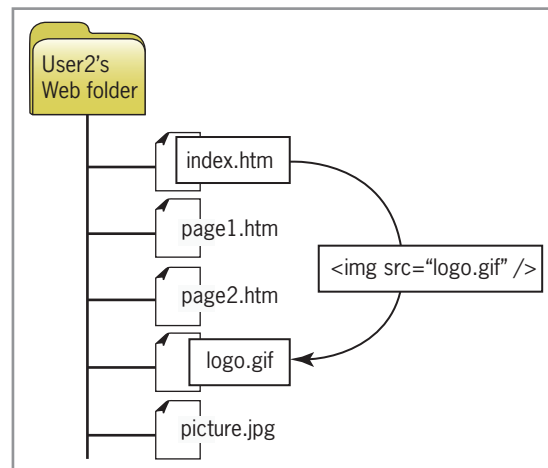
```
<img src="logo.gif" />
```


Figure 3-8 Simplified single folder file structure

Using a Hierarchical Folder Structure

The simple directory structure shown in the preceding example is fine for a small Web site, but as your site grows you may want to segregate different types of content into separate folders for ease of maintenance. Take a look at the relative file structure for User2's Web site as depicted in Figure 3-9. Notice that User2's Web folder contains three HTML files and one subfolder named images, which contains the graphics and pictures for the Web site.
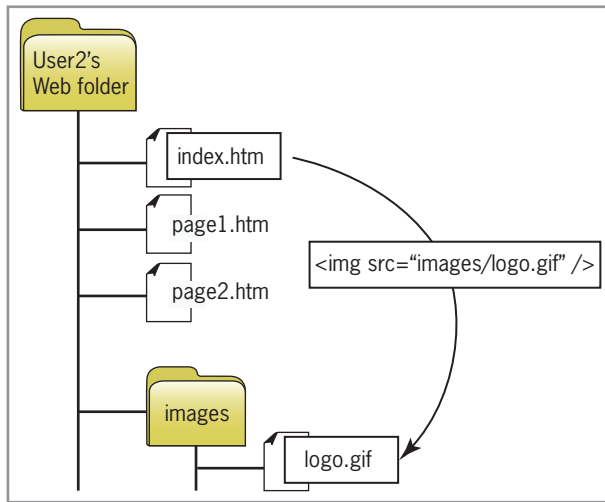

Figure 3-9 Basic hierarchical folder structure

To include the image file logo.gif in index.htm, User2 adds the following code to index.htm:

```
<img src="images/logo.gif" />
```

The path in the src value tells the browser to look down one level in the directory structure for the images folder and find the file logo.gif. The path to the file is relative to the file the browser is viewing. This type of relative file structure can be moved to different machines; the relationship between the files does not change, because everything is relative within the Web folder.

User2's Web site may need a more segregated directory structure, as shown in Figure 3-10. In this example, common files such as the index (the home page) and site map reside in the top-level folder. Multiple subfolders contain chapter and image content. Two linking examples are illustrated in this figure:

- *Example 1*—To build a link from page1.htm (in the chapter1 folder) to index.htm, use ../ in the path statement to indicate that the file resides one level higher in the directory structure, as shown in the following code:

```
<a href="../index.htm">Home</a>
```

- *Example 2*—To include the image file logo.gif in page1.htm, use ../ to indicate that the file resides in the images folder, which is one level higher in the directory structure, as shown in the following code:
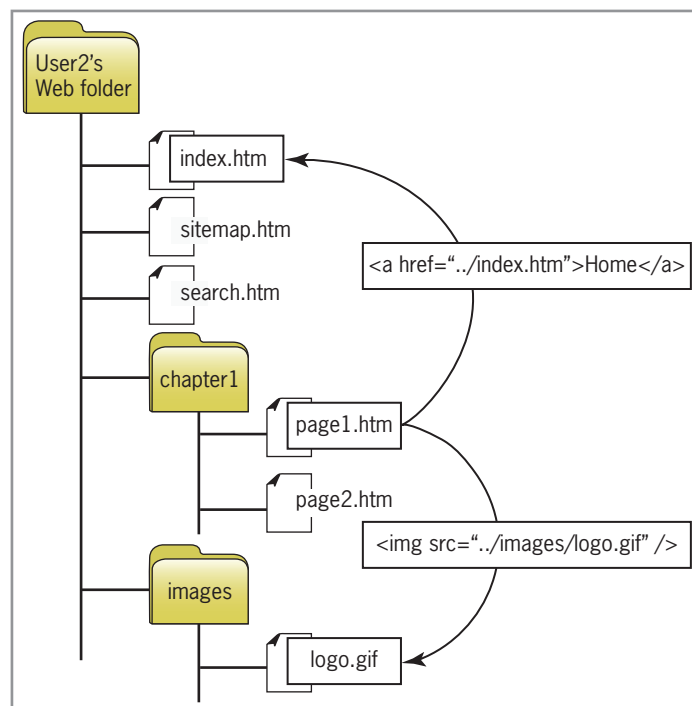
```
<img src="../images/logo.gif" />
```


Figure 3-10 More segregated hierarchical folder structure

Creating a Site Storyboard

Plan your site by creating a storyboard flowchart that shows the structure, logic, and taxonomy behind the content presentation and navigation choices you offer. You can sketch your site with paper and pencil or create it using flowchart diagramming software. Sometimes it is helpful to use sticky notes or cards to plan the structure visually. This method lets you easily move pages from one section or level to another. Whichever method you choose, this preliminary planning step is one of the most important in

planning your site. You can move pages and whole sections of content freely, plan navigation paths, and visualize the entire site. This is the stage at which to experiment and refine your designs. Once you have started coding the site, it is much more difficult and time consuming to go back and make major changes. Remember to adhere to the file-naming conventions for each of your pages.

Organizing the Information Structure

Think about your users' information needs and how they can best access the content of your site. How should your information design map look? Review the sample structures provided in this section, and judge how well they fit your information. Your design may incorporate several structures, or you may have to adapt the structures to your content. Each sample structure is a template; you may have more or fewer pages, sections, topics, or links. You may choose to use bidirectional links where only single-direction links are indicated. Use these examples as starting points and design from there.

Linear Structure

The linear information structure, illustrated in Figure 3-11, guides the user along a straightforward path. This structure lends itself to booklike presentations; once into the content, users can navigate backward or forward. Each page can contain a link back to the main page if desired. Pages may also contain links to a related subtopic. If the users jump to the subtopic page, they only can return to the page that contains the subtopic link. This structured navigation returns them to the same point in the content path.
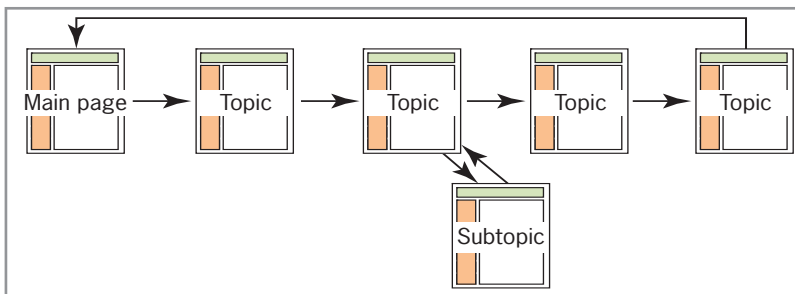

Figure 3-11 Linear information structure

Tutorial Structure

The tutorial structure illustrated in Figure 3-12 is perfect for computer-based training content such as lessons, tutorials, or

task-oriented procedures. The tutorial structure builds on the simple linear structure in Figure 3-11. The user navigates the concept, lesson, and review pages in order. Because the lessons use hypertext, users can leave the lesson structure and return at any time. They also can choose the order of lessons and start anywhere they want. Notice that the table of contents, index, and site map pages are linked to—and from—all pages in the course. Within each lesson users can navigate as necessary to familiarize themselves with the content before they review. This structure can be adapted to fit content needs; for example, the group of pages in the illustration could be one section of a larger training course.
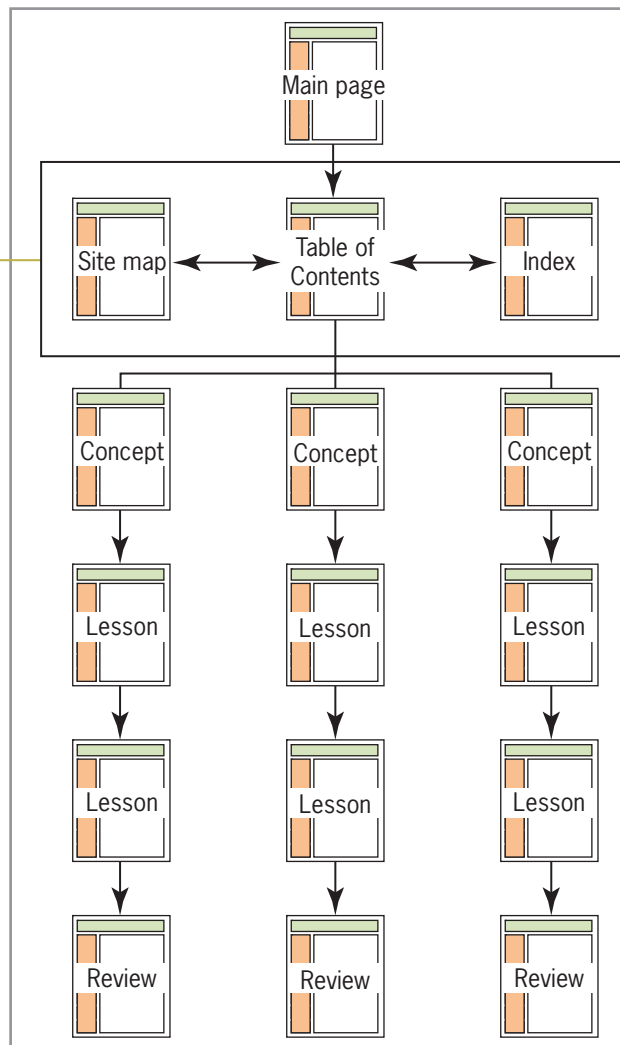

Figure 3-12 Tutorial structure

Web Structure

Many smaller Web sites follow the Web-type content structure illustrated in Figure 3-13, which is nonlinear, allowing the user to jump freely to any page from any other page. If you choose to use this type of content structure, make sure that each page includes clear location information and a standardized navigation bar that not only tells users where they are, but where they can go.
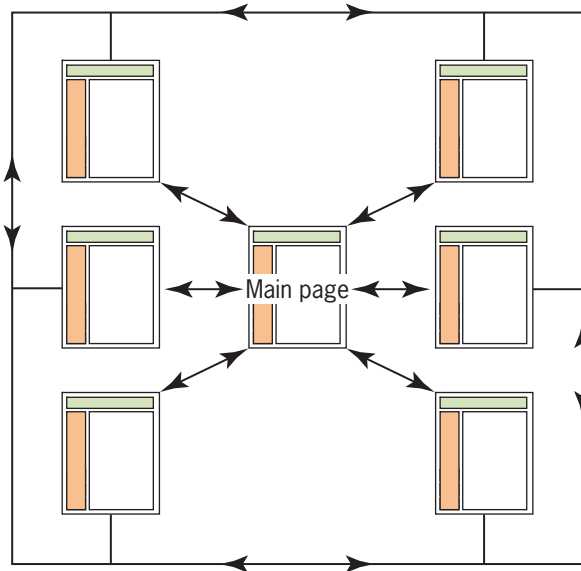

Figure 3-13 Web structure

Hierarchical Structure

The hierarchical structure illustrated in Figure 3-14 is probably the most common information design. It lends itself to larger content collections because the section pages break up and organize the content at different levels throughout the site. Navigation is primarily linear within the content sections. Users can scan the content on the section page and then choose the content page of their choice. When they finish reading the content, they can return to the section page. The site map allows users to navigate freely throughout the site. A navigation bar on each page lets the user jump to any section page, the main page, and the site map.
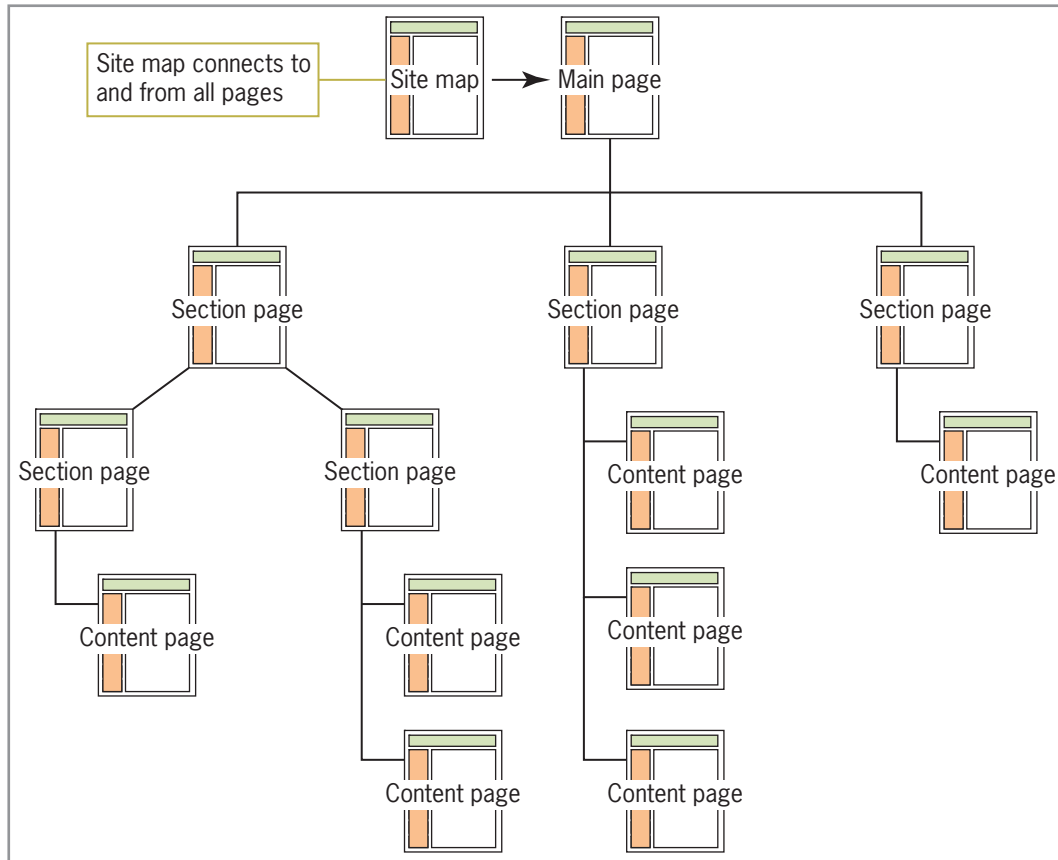
Figure 3-14 Hierarchical structure

Cluster Structure

The cluster structure illustrated in Figure 3-15 is similar to the hierarchical structure, except that every topic area is an island of information, with all pages in each cluster linked to each other. This structure encourages exploration within a topic area, allowing the user to navigate freely through the content. All pages contain a navigation bar with links to the section pages, main page, and site map.
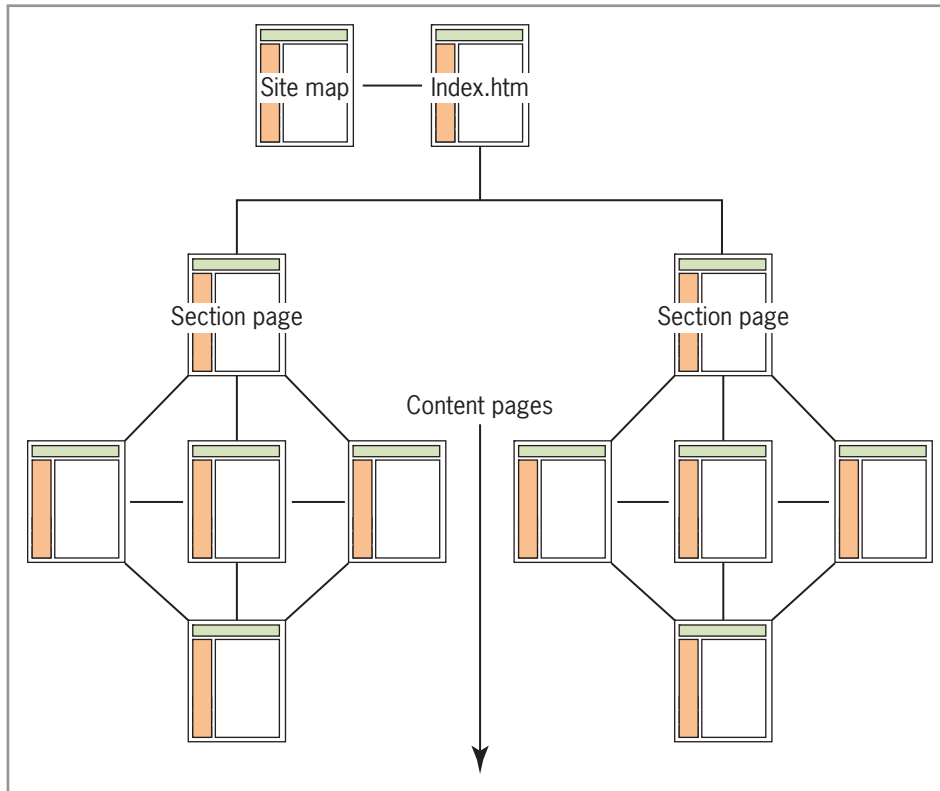
Figure 3-15 Cluster structure

Catalog Structure

The catalog structure illustrated in Figure 3-16 is ideally suited to electronic shopping. The user can browse or search for items and view specific information about each product on the item pages. Users can add items to their shopping cart as they shop. When they are finished, they can review the items in their shopping cart and then proceed to checkout, where they can enter credit card information and finalize the order.

Figure 3-16 Catalog structure

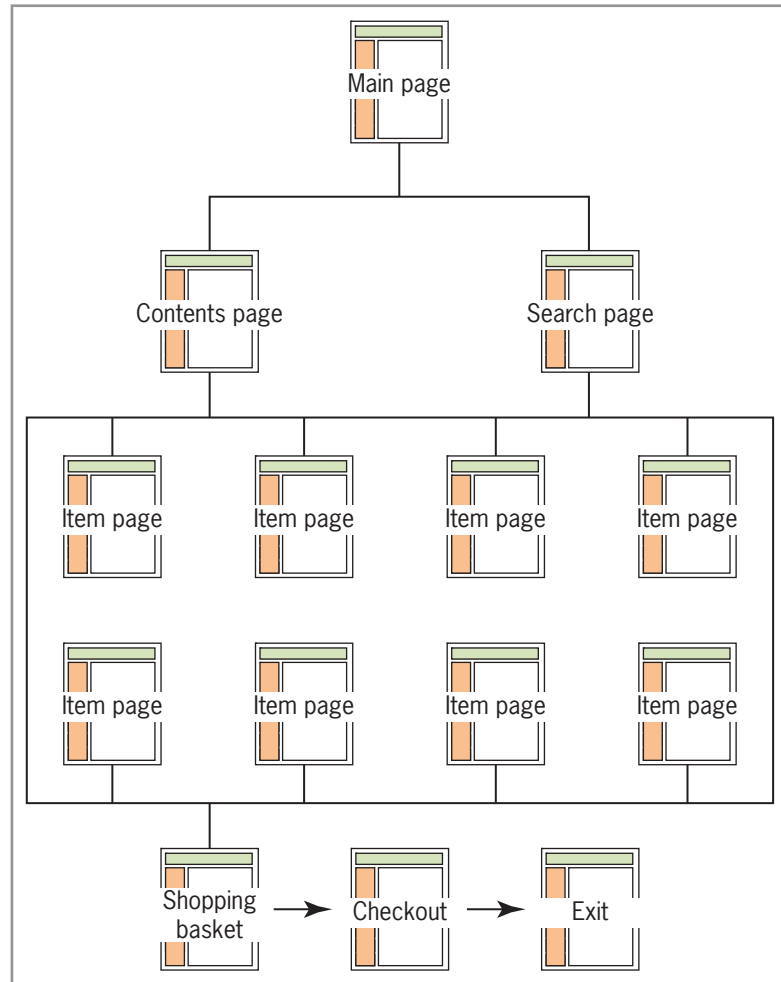This type of Web site requires back-end data transaction processing to handle the shopping cart tally, process credit card information, and generate an order for the warehouse. Businesses that want to set up an e-commerce site can purchase ready-made commerce software packages or develop their own from scratch.

Publishing Your Web Site

To make your Web site live, you transfer your Web site files to a **Web server**, a computer connected to the Internet and running server software. The software lets the computer use the Hypertext Transfer Protocol (HTTP) to serve HTML files to Web browser clients. Unless your company or organization has a Web server and hosts its own content, you must use the services of a Web hosting provider. After you choose a server to host your files, you need to select file transfer software and upload the Web site files from your development machine to the Web server.

Choosing a Web Hosting Service Provider

One of the most important choices you will make is your Web hosting service. This is the company that hosts your Web pages on a Web server, making them available to anyone who knows your URL. Most Web hosting companies offer hosting services for both personal and business use. The Web host provides you with Internet access, e-mail accounts, and space for a personal or business Web site. If you are building a Web site for business use, your Web host can register a personalized domain name for your Web site.

Small Web sites (around 15–20 pages of content) do not need much more than 1 or 2 MB of server space to hold all of the HTML pages and graphics. Your Web hosting package should provide at least 10 MB of space so your Web page has room to grow. Many personal Web sites can be hosted on the free server space that comes with many cable and digital subscriber line (DSL) modem connection packages. Check with your service provider to see if this feature is available.

Larger or more complex sites need more server space, especially if you have downloadable files, archives, lots of graphic content, or databases. If you are building a business Web site, seek out larger hosting services that are more appropriate for hosting a complex commercial site.

Shopping for a Web hosting service can be a confusing experience, as no two are exactly alike. Do some research and learn about offerings from different vendors. The following sections discuss the features you should seek in a hosting service.

Some Web hosting services offer proprietary design tools and templates to assist you in building a Web site. It's best to avoid these types of tools as they tend to tie you to one vendor and make it difficult to switch hosting services and post your Web site elsewhere.

DSL and Cable Access

Most Internet users now have access to high-speed, broadband connection services through a DSL or cable modem. To take advantage of DSL or cable access to the Web, you need a network card for your computer and a DSL or cable modem. **Internet service providers (ISPs)** usually supply a modem with their service. Check to make sure that the monthly fee does not include the equipment costs for the modem. Because DSL and cable are "always-on" connections, they introduce an increased security risk that can make your network vulnerable to hackers. If your provider does not offer network security, you must purchase a network security device, known as a gateway router, to protect your computer with a security firewall. The router allows multiple computers in your home or business to share the high-speed Internet connection, while the firewall software blocks intruders from accessing your network.

Accessible Technical Support

Technical support is not a feature—it is an absolute necessity. Make sure that your Web hosting service has competent, accessible customer service. When you are checking into Web hosting services, call and talk with the companies' customer service representatives. Tell them how experienced you are with computers, and let them know what you hope to accomplish (such as the type of Web site you want to build). Note how long you are on hold when waiting to speak with customer service. Make sure that you are comfortable with the level of service you receive on these initial inquiries.

E-Mail Addresses

All Web hosting accounts come with variable number of e-mail addresses that you can assign to yourself and anyone else you want to have an e-mail address that uses your domain name. If you are part of a group, multiple mailbox accounts let each person receive his or her own e-mail. You can also set up a "catch-all" e-mail address that will accept any e-mail sent to your Web site regardless of whether it is addressed to you.

SQL Database Support

If you are planning on any type of electronic commerce or customized data presentation, you need database support. Databases that understand **Structured Query Language (SQL)**, a programming language that lets you select information from a database, are the most common and powerful type of database.

Secure Sockets Layer (SSL) Support

The **Secure Sockets Layer (SSL)** is an Internet communications protocol that allows encrypted transmission of data between the user and the server. SSL is necessary if you are planning to set up an electronic commerce site or transmit other sensitive data. Encrypting the data ensures the information cannot be read if the transmission is intercepted.

Registering a Domain Name

A domain name is an alias that points to your actual location on the Web server, as shown in Figure 3-17. User2 has purchased the domain name *www.mysite.com*. The actual path to User2's content is hidden, and the visitor to the site sees only the domain name.
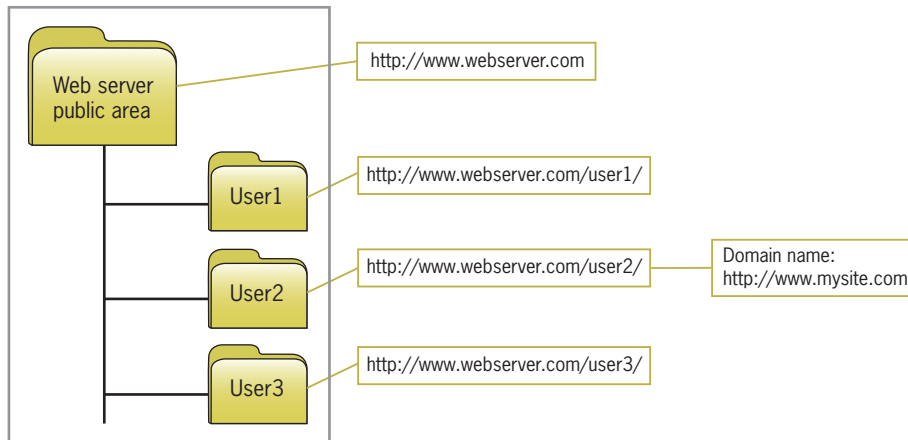
Figure 3-17 Domain name hides the actual path

Domain names are managed by the Internet Corporation for Assigned Names and Numbers (ICANN). ICANN has agreements with a number of vendors to provide domain name registration services. You can purchase a domain name through a vendor, and they will register it with ICANN. Current popular domain name registration services include Go Daddy (*www.godaddy.com*), Register.com (*www.register.com*), and Dotster (*www.dotster.com*).

Web Hosting Service Comparison Checklist

Use the following checklist when you compare Web hosting services:

- Is the Web host local or national?

- What are the details of the different hosting packages? How much server space comes with each? What are the limits (if any) on uploads and downloads?

- Are there bandwidth limits for the number of visitors your site receives per month?

- Does the Web host offer technical support? When is support staff available?

- How many e-mail addresses do you get with an account?

- Does the Web host provide software, such as a **File Transfer Protocol (FTP)** client to transfer files over the Internet?

- Does the Web host support the latest connection technologies?

- Does the Web host offer enhanced services, such as SQL database support, SSL, a scripting language environment, and support for streaming audio and video?

Always keep a backup of your Web site files in case you have any problems during FTP transmissions, or if you accidentally delete or overwrite existing files. Of course, if you accidentally delete or overwrite files on your local computer, you can always use your Web site files as a backup.

Uploading Files with the File Transfer Protocol

To publish your pages on the Web, you must send your HTML code, image, and other files to the Web server. To do this, you need FTP software, often called an FTP client. Some HTML-authoring software, such as Microsoft Expression Web and Adobe Dreamweaver, include built-in software packages that let you upload files to your Web server. You also can choose from many shareware and freeware FTP programs to upload your files. Table 3-3 lists some popular FTP clients available as freeware and shareware. Visit your favorite shareware site, such as Shareware.com, and search for FTP clients.

FTP Client	Web Site
Cute FTP—Shareware	*www.cuteftp.com*
Filezilla—Freeware	*http://filezilla-project.org*
FireFTP (Firefox plug-in)—Freeware	*http://fireftp.mozdev.org*
Fugu (Macintosh)	*http://rsug.itd.umich.edu/software/fugu*

Table 3-3   Freeware and Shareware FTP Clients

When you have decided which FTP software to use, verify your correct FTP address from your Web host. You also need your account name and password, which in most cases automatically points your FTP program to the proper directory on the server.

To upload your files, start your FTP program and connect to your Web server using the FTP information provided by your service provider. Your password allows you write access to your directory on the Web server, which means you can copy files to and from the directory. Once the FTP client has connected to the Web server, you have the option of choosing the files you want to transfer. The FTP client usually displays directories on both the local and remote computers. Figure 3-18 shows the FireFTP plug-in for Firefox, which lets you transfer files using your browser. You can see that the files on the local computer and Web server match, as files on the Web server are duplicates of the files maintained on the local computer.
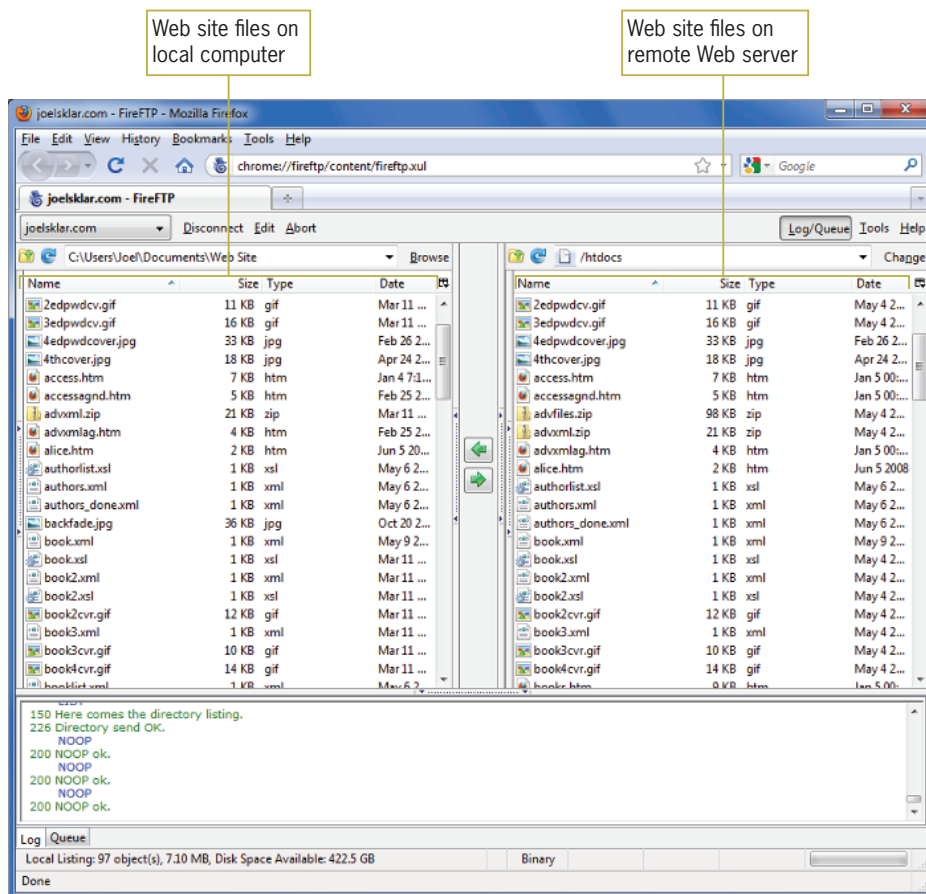
Web site files on local computer

Web site files on remote Web server

Figure 3-18 Typical FTP software

140

Select the files that you want to upload in your local directory listing and transfer them to the Web server. You also can transfer files from the Web server to your computer. The first time you go live with your Web site, you must transfer all the files. Later you upload only the files that you have updated. After the files reach the Web server, they are available for access immediately on the Web.

After you find a Web hosting service and publish your Web site to the World Wide Web, it is time to test your Web site in the real-life Internet environment.

Testing Your Web Site

Even though you performed tests throughout the development of your Web site, you need to continue testing after you post your files live on the Web. If possible, load your files to the Web server and test them before making your URL available for users to access the Web site. If you have enough server space, you may want to establish a testing area on the Web site. You can do this by creating a subdirectory in your public HTML directory. Do not publicize the URL so that your testing area can remain private.

Testing Considerations

Always test in as many different environments as possible. Remember to test for the following Web design variables:

- *Multiple browsers*—Test your site using as many browsers as you can to make sure your work is portable and is displayed consistently.

- *Multiple operating systems*—If you can, test your site from different operating systems. If you have a PC as a development machine, use a Macintosh for testing, and vice versa. You even can run different versions of UNIX on a PC, if necessary. Because computer chip development moves at a lightning pace, machines become outdated quickly. You can find discounted and used machines that often are Internet-capable as long as they have a good Internet connection. Because you won't use these machines to *develop* Web sites (only to *view* them), you do not need the latest or most powerful hardware.

- *Connection speeds*—Do not rely on the same connection speed when testing your Web site, especially if you work in a corporate environment where the connection to the Internet usually is faster than the average user's. Go to a friend's house, library, or Internet café and access your Web site from there. Test for download times at different connection speeds.

- *Display types*—Test at different screen resolutions and monitor types to make sure your colors are displayed consistently.

- *Link testing*—Use a link validation tool to ensure that all of your links connect to a live page. Link validation tools are built into many HTML editors and are available as stand-alone tools. Many Web sites also offer validation, including the W3C's link validator at *validator.w3.org/checklink*. Any pages that link outside of your Web site need to be tested on a regular basis to make sure that the destination site has not moved, shut down, or posted content different from what you expect.

Usability Testing

Usability testing can be as simple as asking a few colleagues to look at your Web site, or as complex as conducting extensive formalized testing. Some companies invest in special user testing labs with videotaping and one-way mirrors to record user behavior, or software that can track users' mouse movements and eye coordination as they look at their Web site. Even if you do not need this level of sophisticated testing, you should perform some type of user assessment of your work. The goal of user testing is to determine whether your Web site is easy to navigate with easy access to content. Following are some points to consider when planning for user testing of your site.

Vary Your Subjects

Draw your test subjects from a variety of backgrounds, if possible. Gather test subjects who are representative of your target audience. Find users with varying computing skills and familiarity with the information. Avoid using friends as test users, as they may only compliment your work. You might choose to let users look at the Web site on their own time, but you can learn a lot by watching users interact with your Web site. Make sure to let them navigate and use the Web site without any outside help from you. Just stand back and watch.

Formalize Your Testing

Formalize your testing by creating replicable methods of testing your Web site. Prepare a series of questions that users have to answer after viewing the Web site. Give them a specific task to complete or have them find a particular piece of information. Let them rate the ease of completing such tasks. Compare the results from different users to find any problem areas in navigation. Administer the same testing methods to a variety of users, and watch for trends and consistencies. This lets you compare results or focus on a particular feature of the Web site.

Develop a Feedback Form

Develop a feedback form that users can fill out after they have tested the Web site. Include a set of criteria, and let them rate the Web site on a progressive scale, or ask them a series of open-ended questions. You also may want to provide the feedback form online, letting users offer feedback directly from the Web site. Here are some sample questions you might ask:

- Did you find the information you needed?

- Was it easy or difficult to access the information you needed?

- Did you find the Web site visually attractive?

- Did you find the content easy to read?

- Did you find the Web site easy to navigate?

- Did you think the information was presented correctly?

- Did the information have enough depth?

- What area of the Web site did you like the best? Why?

- What area of the Web site did you like the least? Why?

- Would you recommend the Web site to others?

Chapter Summary

A successful Web site is the result of careful planning. The steps you take before you actually start coding the site save you time, energy, and expenses in the long run. Remember these guidelines for successful planning:

- Become familiar with the stages in the Web site development cycle.

- Start with pencil and paper; your ideas are less restricted and you can easily revise and recast without recoding. Follow a development process to ensure fewer revisions and design reworking.

- Write a site specification document. You will find it invaluable as a reference while building your site.

- Identify the content goal by adopting your users' perspective and learning what they expect from your site.

- Analyze your audience and create an audience profile. Focus your site on the users' needs, and continue to meet those needs by adapting the site based on user feedback.

- An effective site is most commonly the result of a team effort. Leverage different skill sets and experience to build a Web site development team.

- Plan for successful implementation of your site by creating portable filename conventions. Build a relative file structure that can be transferred to your Web server without a hitch.

- Select a basic information structure for your site, and then manually diagram it, customizing it to the needs of your site.

- Shop carefully and compare features when you are looking for an ISP or Web host. Consider the future disk space and technology needs of your content.

- Download and learn to use an FTP client for use in the often-repeated task of transferring files to your Web site.

- After your Web site is live, test it against the basic Web variables of browser, operating system, display resolution, and connection speed.

- Test your Web site with a variety of users. Listen carefully to their feedback to identify trouble spots in your information design.

- Plan for the maintenance, upkeep, and redesign of your Web site. Keep your content up to date. Let users know when you have made updates to the Web site.

Key Terms

audience definition—A profile of your average user.

complete URL—An address of documents and other resources on the Web that includes the protocol the browser uses to access the file, service, domain name, the relative path, and the filename.

extranet—A private part of a company's intranet that uses the Internet to securely share part of an organization's information.

File Transfer Protocol (FTP)—A standard communications protocol for transferring files over the Internet.

freeware—Programs available free of charge or with an optional donation fee.

Internet service provider (ISP)—A company that provides Internet access and Web site hosting services to individuals and organizations.

intranet—A private collection of networks contained within an organization. Intranet users gain access to the Internet through a firewall that prevents unauthorized users from getting in to the intranet.

ISO 9660 Standard—A file system standard published by the International Organization for Standardization (ISO) that supports computer operating systems such as Windows, classic Mac OS, and UNIX-like systems, to exchange data.

page view—The number of times a Web page is viewed by a user.

partial URL—A Uniform Resource Locator (URL) that omits the protocol and server name and only specifies the path to the file relative to one another on the same server.

requirements—The list of customer needs for a Web site, such as search capability, tabbed menu navigation, specific color and branding requirements, or anything else that will create the desired outcome for the site.

Secure Sockets Layer (SSL)—Communications software that allows transmission of encrypted secure messages over the Internet.

shareware—Software that is distributed free so users can try it before they buy it. Users then can register the software for a relatively small fee compared to software produced commercially.

Shareware usually is developed by individuals or very small software companies, so registering the software is important.

site specification—The design document for your Web site.

Structured Query Language (SQL)—A programming language that lets you select information from a database.

taxonomy—A classification and naming of content in a hierarchical structure.

Uniform Resource Locator (URL)—The global address of documents and other resources on the Web.

Web analytics—The analysis of statistics that are gathered by Web servers.

Web server—A computer connected to the Internet that runs server software. The software lets the computer use the Hypertext Transfer Protocol (HTTP) to serve HTML files to Web browser clients.

wireframe—Web page mockups that represent page layouts for a Web site.

Review Questions

1. List three technology constraints that can affect the way a user views your Web site's content.

2. Consult your Web server administrator when you need to determine the _____ and _____ for your site.

3. Name two inconsistencies that can cause broken links when you upload your files to a Web server.

4. List three characteristics of filenames that vary by operating system.

5. The international standard for filenames often is called _____.

6. Which computer operating system is case sensitive?

7. Rename the following files so that they are compatible across all operating systems:

    My file.htm _____

    case:1.htm _____

    #3rdpage.htm _____

8. What is the default main page filename for a Web site?

9. What are the two types of URLs?

10. What are the four parts of a complete URL?

11. What type of URL links to another server?

12. What type of URL links within a server?

13. What affects the format of the URL for your Web site?

14. What is the benefit of purchasing a domain name?

15. Why should you never specify an absolute path in partial URLs?

16. What is the benefit of building a site with relative paths?

17. Files that reside in the same directory need only the _____ to refer to each other.

18. List two benefits of diagramming your site before you start coding.

19. How does a Web site become live?

20. List the four variables to consider when testing your Web site.

Hands-On Projects

1. Browse the Web and find a site you like. Write a brief statement of the Web site's goals.

2. Browse the Web and find Web sites that fit the following content types:

   a. Billboard

   b. Publishing

   c. Special interest

   d. Product support

   Write a short summary of how the content is presented in each Web site, and describe how each site focuses on its users' needs.

3. Browse the Web and find a site that does not contain a user survey form. Write a 10- to 15-question user survey that you would use on the site. Tailor the questions to the site's content and goals.

4. Browse the Web to find examples of the following site structures, and describe how the content fits the structure. Think about how the chosen structure adds to or detracts from the effectiveness and ease of navigation of the site. Determine whether the site provides sufficient navigation information. Print examples from the site, and indicate where the site structure and navigation information is available to the user.

   a. Linear

   b. Hierarchical

5. Browse the Web to find a site that uses more than one structure type, and describe why you think the site's content benefits from multiple structures. Consider the same questions as in Project 3-4.

6. Are there other structure types that are not described in this chapter? Find a site that illustrates a structure content not covered in this chapter. Create a flowchart for the site, and determine how it benefits from the different structure type.

7. Write a test plan for your Web site.

    a. Create a section for each design variable.

    b. Spell out the exact steps of the test and the different variables to be tested. State explicitly which browsers and versions should be used, and on which operating system. Detail the different screen resolutions and connection speeds. List the exact pages that should be tested.

    c. Walk through the test procedure to test its validity.

8. Write a sample user feedback questionnaire.

9. Write a maintenance plan for your Web site.

    a. Include a schedule of content updates for the different sections of the Web site.

    b. Include a schedule of design reviews.

    c. Plan for link maintenance.

Individual Case Project

Write a site specification for the site you defined in Chapters 1 and 2. Include as much information as possible from the project proposal you completed at the end of Chapter 1. Make sure to include a mission statement. Determine how you will measure the site's success in meeting its goals. Include a description of the intended audience. Describe how you will assess user satisfaction with the site. Include technological issues that may influence the site's development or function.

Prepare a detailed flowchart for your site using the preliminary flowchart you created at the end of Chapter 2. Create a filename for each page, using a consistent naming standard. Indicate all links between pages. Write a short summary that describes the flowchart. Describe why you chose the particular structure, how it suits your content, and how it benefits the user.

Use the page layouts you sketched in Chapter 2 and create wireframes. Download one of the free wireframe tools listed in this chapter, or use a drawing program to create the wireframes.

Team Case Project

Collaborate to write a site specification for the site you defined in Chapters 1 and 2. Include as much information as possible from the project proposal you completed at the end of Chapter 1. Make sure to include a mission statement. Determine how you will measure the site's success in meeting its goals. Include a description of the intended audience. Describe how you will assess user satisfaction with the site. Include technological issues that may influence the site's development or function.

Work individually to determine the information structure you think is optimal for the type of content your site will contain. Then meet and work as a team to determine the information structure using the best pieces of each team member's information structure plan. Prepare a detailed flowchart for your site using the preliminary flowchart you created at the end of Chapter 2. Create a filename for each page, using a consistent naming standard. Indicate all links between pages. Write a short summary that describes the flowchart. Describe why you chose the particular structure, how it suits your content, and how it benefits the user.

Use the page layouts you sketched in Chapter 2 and create wireframes. Download one of the free wireframe tools listed in this chapter, or use a drawing program to create the wireframes.