

Глава 2

От концепции к реализации

В этой главе...

- Определение целей и требований сайта
- Разработка плана
- Создание карты сайта и макета страницы
- Разработка стратегии размещения текста и графики
- Тестирование, запуск и поддержка

Разработка профессиональных Web-сайтов является весьма серьезным занятием, сродни созданию кинофильмов. При огромном количестве стратегических элементов сайта, этапов разработки и людей, спорящих о реализации тех или иных элементов (от сжатия графики до исправления ошибок и лицензирования аудиоклипов), вы наверняка будете страдать от суматохи.

Множество модных компаний по Web-дизайну рассказывают своим клиентам о применяемой ими пятиэтапной технологии разработки сайтов, дабы поведать клиенту о хронологии процесса разработки. Эти пять этапов таковы: определение целей и требований, проектирование, разработка, тестирование, запуск и поддержка сайта. Перечень этих этапов создания Web-сайтов не является чем-то особенным или секретным. Для их определения достаточно здравого смысла, но их названия красиво звучат для клиентов.

В этой главе будет рассказано о процессе, который реализуют большинство Web-агентств и корпоративных разработчиков. Кстати, этапы создания практически аналогичны действиям при создании контента для мобильных устройств и DVD. Итак, эту главу стоит прочитать.

Этап 1. Определение целей и требований

Какое искушение — изобразить схематически структуру нового сайта, но первый этап должен быть “разведывательным”. Необходимо определить бизнес-цели и понять требования заказчиков, для которых вы взялись создавать Web-сайт.

Определение целей сайта

И для маленьких компаний семейного бизнеса, и для компаний размера Disney первый шаг одинаков: надо определить в порядке приоритета три (или около того) цели, преследуемые компанией, решившей заказать у вас разработку или модернизацию Web-сайта. Эти цели напрямую влияют на структуру сайта, его внешний вид, схему и содержимое каждой страницы.

Например, модернизируя сайт школы плавания, заказчик выдвинул три цели. Первая — реализовать различные варианты перехода с главной страницы. Вторая — реализовать быст-

рый доступ к основной информации для постоянных клиентов. И третья — привлечь внимание потенциальных клиентов к телефонному номеру, по которому нужно позвонить для получения интересующей информации у агента по продажам. Главная страница сайта *La Petite Baleen* представлена на рис. 2.1, где элементы содержания и дизайн Web-страницы полностью отвечают поставленным целям.

Определение бизнес-требований



Следующий полезный шаг: сядьте рядом с заказчиком и составьте список того, что должно быть реализовано с помощью сайта. Предложения должны быть составлены по такому образцу: “Web-сайт должен обеспечивать...”.

Вот ряд общих указаний.

- ✓ **Постройте разговор на высоком уровне.** Расскажите клиенту, по какой схеме он должен отвечать на ваши вопросы. По моему опыту, клиенты часто переходят от того, *что* им нужно, к тому, *как* это реализовать.
- ✓ **Составьте список.** Запишите список предложений, начинающихся со слов “должен обеспечить”:
 - должен обеспечить регистрацию;
 - должен обеспечить вход (выход);
 - должен обеспечить рекламу конкретного товара каждый месяц;
 - должен обеспечить возможность нашей команды обновлять содержимое главной страницы каждый день.
- ✓ **Расположите требования в порядке приоритета.** Отсортируйте все предложения “должен обеспечить” по возрастанию приоритета: 1 — обязательно обеспечить, 2 — желательно обеспечить, 3 — можно реализовать в будущем.

Создание плана проекта

После того как вы тщательно продумали все на предыдущих этапах, можно приступить к составлению расширенного плана проекта — в соответствии с пятью этапами. Расширенный план проекта составляется для того, чтобы четко определить: сколько времени, денег и человеческих ресурсов вам понадобится для выполнения работы на каждом этапе.



Для составления плана проще начать с конца — с даты сдачи сайта в эксплуатацию, наметив основные этапы. Такой подход позволяет вам наглядно увидеть *готовность* проекта, как мне нравится это называть.

Многие руководители Web-проектов используют для упрощения своей работы программу Microsoft Project, позволяющую назначить этапы разработки, распределить задания между всеми членами команды, установить *зависимости* между заданиями — или, иными словами, определить события, зависящие от более ранних событий. Обратите внимание на рис. 2.2, где показан поэтапный план работ в виде *диаграммы Ганта*. Вначале дизайнер создает несколько вариантов дизайна сайта и предлагает заказчику сделать свой выбор. В результате заказчик останавливается на одном из вариантов дизайна, который и берется в дальнейшем за основу. Если клиент выбирает дизайн более двух дней или вообще отвергает все варианты, предложенные ему разработчиком, то план не выполняется, и дата сдачи проекта в эксплуатацию оказывается под угрозой смещения.

В заголовке отображаются различные варианты перехода по сайту



Телефонный номер

Основная информация отображена на главной странице

© La Petite Baleen, www.swimlpb.com

Рис. 2.1. Первый шаг в создании Web-сайта — определение бизнес-целей сайта, которые непосредственно влияют на его структуру, содержимое и графический дизайн

☐ Look and feel development	\$0.00	13 days	Wed 8/3/05	Fri 8/19/05
Collaborate on preliminary wireframing	\$0.00	6 days	Wed 8/3/05	Wed 8/10/05
Develop design directions	\$0.00	5 days	Thu 8/11/05	Wed 8/17/05
Review with client	\$0.00	1 day	Mon 8/15/05	Mon 8/15/05
Refine design from client feedback	\$0.00	3 days	Tue 8/16/05	Thu 8/18/05
Client sign off on design direction	\$0.00	1 day	Fri 8/19/05	Fri 8/19/05
Look and feel complete	\$0.00	0 days	Fri 8/19/05	Fri 8/19/05

Рис. 2.2. На этом графике для каждой задачи предусмотрена отдельная строка. Обратите внимание на то, что некоторые задания нельзя выполнить, пока не реализованы предыдущие

Этап 2. Проектирование

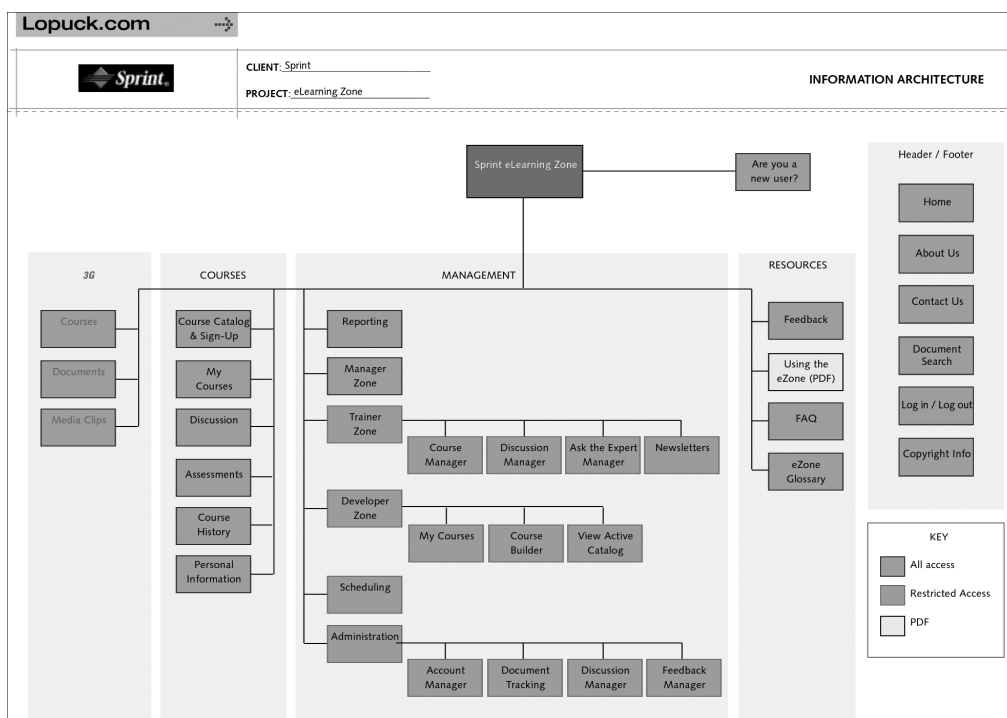
Покупая или получая в подарок игрушки с надписью “требуется сборка”, вы понимаете, что вам предстоит решить головоломку. Многие, и я в том числе, около получаса пытаются выполнить задание, не читая инструкцию. Пока вы не начнете рвать на голове волосы и не сдадитесь, вы ни за что не заглянете в руководство. Этап проектирования Web-сайта предполагает написание такого руководства для всех членов команды разработчиков до того, как кто-то приступит к выполнению работы.

Создание схемы Web-сайта



Первоочередной задачей на этапе проектирования Web-сайта, как крупного, так и небольшого, является создание схемы сайта, или, говоря на языке Web-индустрии, *карты сайта*. Без создания карты сайта, которая, например, показана на рис. 2.3, ваш проект обречен на провал из-за того, что в команде разработчиков не будет согласия и общей направленности действий. Во-вторых, без плана вы не сможете учесть все содержимое страницы и других элементов сайта.

Например, представьте, что вы уже сделали полработы по созданию сайта и вдруг вспоминаете, что забыли внедрить область ресурсов. Теперь вам придется переделывать интерфейс, а при наличии плана вы бы сделали все правильно с первого раза. Ознакомьтесь с главой 3, чтобы узнать более подробно о создании карты сайта.



@Sprint, Inc.

Рис. 2.3. На карте сайта показаны все основные элементы сайта и их связь между собой. В дальнейшем эти взаимосвязи станут основой при разработке схемы навигации

Составление плана для каждой страницы

Карта сайта позволяет схематически увидеть весь сайт в целом, но она не дает возможности детально рассмотреть каждую страницу. Для этого вам понадобится набор *макетов страниц*.



На этапе проектирования архитектор информации собирает данные у всех членов команды и разрабатывает схему, которая называется *макетом*, для каждой Web-страницы сайта. Макет страницы (рис. 2.4) содержит следующие элементы:

- ✓ схема *глобальной навигации* (элементы навигации, которые содержатся на всех страницах сайта);
- ✓ текстовые блоки и элементы мультимедиа, их относительное расположение на страницах сайта;
- ✓ интерактивный дизайн (как пользователи будут работать с элементами на странице).

Более подробно о создании макетов страниц рассказывается в главе 3.

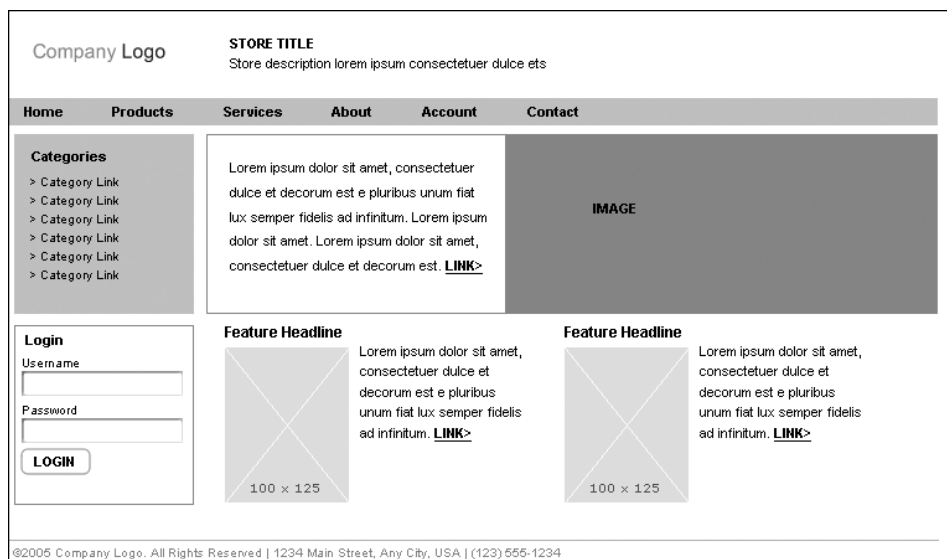


Рис. 2.4. Макет Web-страницы — это общая схема страницы и расположенных на ней элементов

Тестирование пользователями

Тестирование макетов страниц пользователями представляет собой эффективный процесс взаимодействия, который начинается на этапе проектирования и должен продолжаться и на этапе разработки. Тестирование элементов интерфейса на ранних стадиях проектирования поможет понять правильность проектирования, а также избежать поспешной разработки элементов интерфейса, которые бы пришлось переделывать из-за несоответствия требованиям пользователей.

В главе 6 более подробно описывается методология эффективного тестирования пользователями основных элементов интерфейса Web-сайта. Для этого нужно создать набор маке-

тов Web-страниц с элементами интерфейса, на которых пользователь может щелкнуть мышью и узнать, как это может выглядеть и работать в будущем. Приглашая заказчиков для тестирования, нужно выбрать представителей разных групп пользователей.

Сборка содержимого сайта

Параллельно с созданием макетов страниц надо разрабатывать и план содержимого сайта. В конце концов, вы не сможете расположить область новостей на странице, не зная, откуда эти новости будут приходить на сайт, насколько большое пространство они будут занимать и как часто их надо обновлять.

Если на сайте планируется использовать *систему управления содержимым* (система управления базами данных, которая будет хранить всю текстовую информацию вашего сайта и автоматически заполнять Web-страницы), то контент-менеджер создает список всех текстовых элементов на каждой странице и разделяет их по типам (например, заголовки, названия, описания), а типы делит на подтипы. Контент-менеджер также занимается разработкой правил форматирования каждого типа текста, определяет максимальное количество символов в строке для разных типов текста.

И наконец, на этапе проектирования команда контент-дизайнеров предлагает стиль текста разрабатываемого Web-сайта, который максимально будет удовлетворять бизнес-целям, поставленным на этапе определения задач и требований. Например, должен ли текст сайта быть завлекающим, или побуждающим, или просто констатирующим факты? Команда писателей должна четко определиться с этими вопросами на этапе проектирования.

Дайте клиенту “посмотреть и почувствовать”

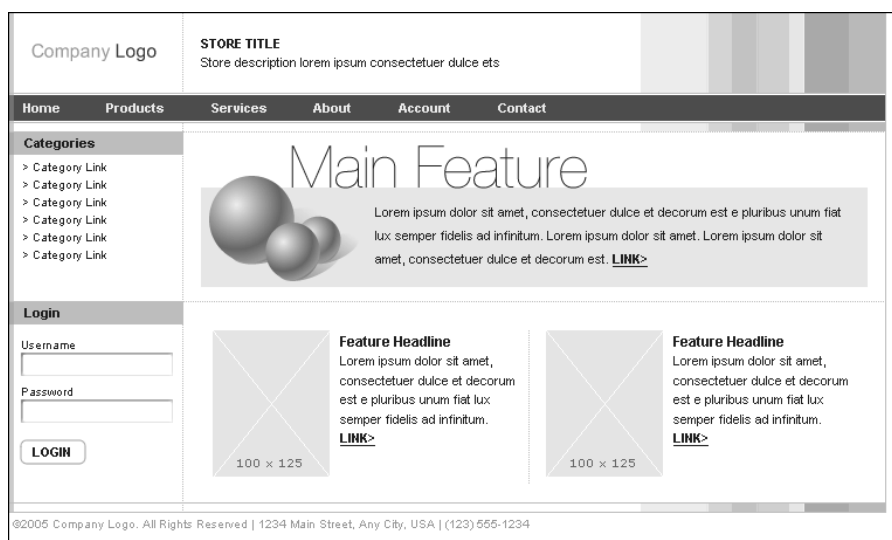
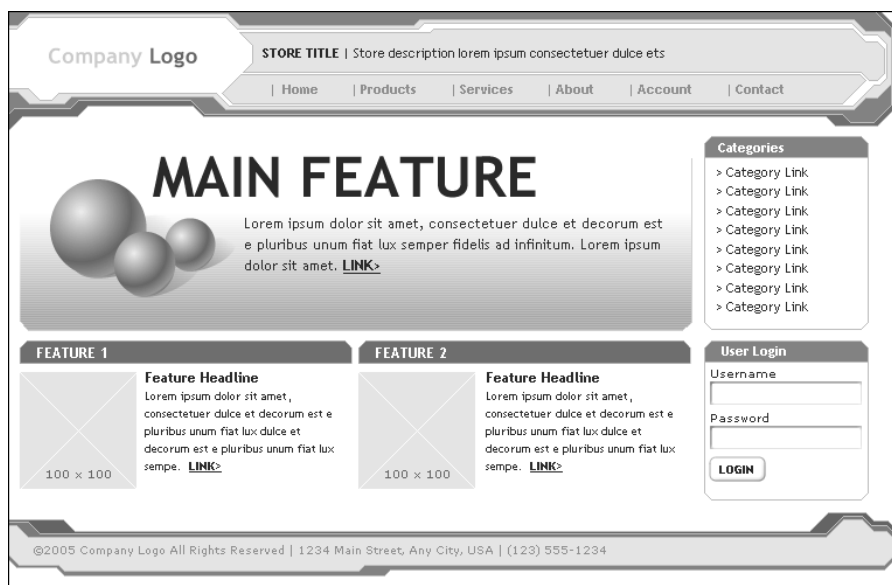
После того как вы создали карту сайта и утвердили ее у клиента, разработали макеты страниц с более-менее четким расположением элементов, к работе могут приступить графические дизайнеры, которые создадут несколько вариантов оформления сайта.



Исходя из моего опыта, лучше разработать несколько вариантов дизайна главной страницы сайта и *внутренних страниц* (любая страница, не являющаяся главной). Это заставляет дизайнеров определиться с дизайном набора страниц, используя навигационную схему всего Web-сайта. Целесообразно создать как минимум три разных варианта дизайна. Эти варианты надо представить клиенту, чтобы он выбрал наиболее оптимальный с его точки зрения. На рис. 2.5 представлено два варианта, выполненных разными дизайнерами, на базе одного и того же макета страницы (см. рис. 2.4). Обратите внимание на то, что каждый дизайн вызывает у вас разные эмоции и ощущения.

Воспользуйтесь мнением фокус-группы

Зачастую заказчик — единственное лицо, которое определяет дизайн сайта. Рассказывая о своих пожеланиях, заказчики иногда просят сделать Web-сайт с взаимно противоречивыми элементами дизайна. Однако есть хорошая идея: при разработке крупномасштабного и бюджетного сайта перед тем, как показать примеры дизайна заказчику, можно воспользоваться мнением фокус-группы. После того как у вас будет ответная информация от фокус-группы, вы можете показать варианты дизайна непосредственному заказчику, направляя таким образом его внимание и выбор на правильный путь.



©WebAssist, Inc, www.webassist.com

Рис. 2.5. Можно получить два совершенно разных варианта дизайна на основе одного и того же макета Web-страницы

Этап 3. Разработка

Пришло время заняться сборкой деталей на экране для формирования Web-сайта. На этапе разработки все элементы (видео, звук, Flash-анимация) готовятся к размещению на Web-странице. Осуществляется сжатие кадров для уменьшения времени загрузки, сохранение

в Web-совместимом формате, переименование и загрузка на сервер. Выполняется программирование и написание HTML-кода, затем все это собирается воедино. Теперь продукт готов к тестированию.

Разработка и окончательная сборка Web-графики

После того как заказчик выбрал окончательный вариант дизайна, закипает работа по созданию графики. Наиболее эффективный способ разработки большого количества графических элементов заключается в использовании дизайнерских шаблонов. Например, большинство Web-сайтов содержит часто повторяющиеся графические элементы: кнопки, заголовки, навигационные панели. Использование шаблонов в дизайне позволяет выдерживать некоторый стиль при создании Web-сайта, а также разделить обязанности по разработке графики между несколькими членами команды, позднее соединив их труды воедино.

Для создания дизайнерских шаблонов и изображений вашего Web-сайта используйте программу Adobe Photoshop или Macromedia Fireworks. Обратите внимание на слои, созданные в файле Adobe Photoshop (рис. 2.6). Применив этот файл с шаблонами кнопок, вы сможете экспортировать любую из кнопок в разрабатываемый Web-сайт столько раз, сколько потребуется.

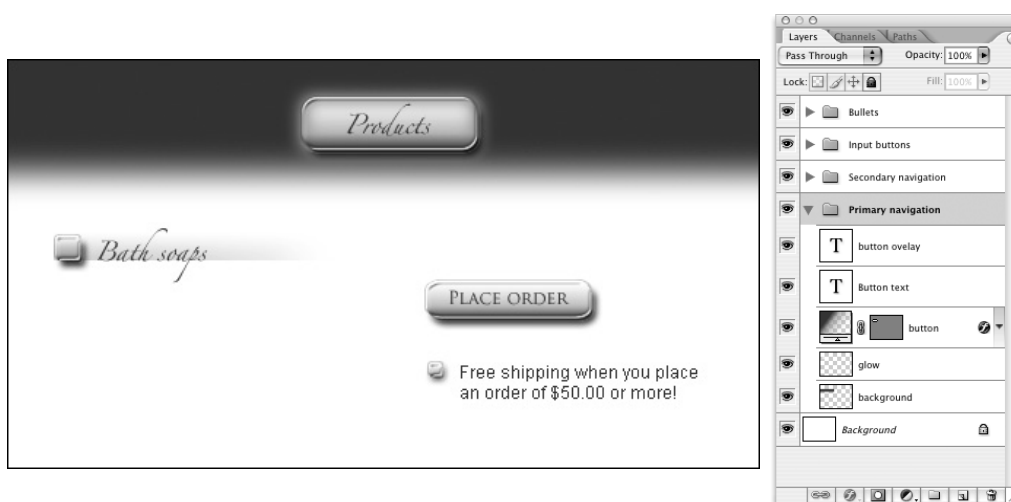


Рис. 2.6. На различных слоях в файле Photoshop можно создать элементы будущего Web-сайта

Текстовое содержимое сайта

А тем временем команда контент-дизайнеров создает текст для Web-сайта. Откуда они знают, что нужно написать? Если вы вернетесь к этапу проектирования, то увидите, что контент-менеджер формирует список всех типов текста, требуемых на сайте. Контент-менеджеры вместе с копирайтерами также определяют стиль текста. Вооружившись информацией о типах текстов, их количестве и стиле, копирайтеры могут приступить к выполнению своей задачи.

Разработка мультимедийного контента

Сегодня большинство сайтов оснащено мощными спецэффектами — Flash-анимацией, звуком и видеоэффектами. Некоторые сайты, например сайт, страничка которого представлена на рис. 2.7, и вовсе разработаны полностью во Flash, а на HTML написана лишь основа страницы, на которой помещается Web-сайт.



С разрешения библиотеки Конгресса, www.loc.gov/exhibits/churchill/interactive

Рис. 2.7. Этот сайт полностью написан на Flash на основе лишь одной HTML-страницы

Разработка Flash-приложений — сложная задача, которая стала уже настоящим профессиональным направлением Web-дизайна. Нельзя сказать, что Flash-дизайнерам нужна была какая-то особенная своя карта сайта или макет страниц, нет. Для эффективной разработки Flash-анимации Flash-дизайнеру точно так же нужна команда других разработчиков и проектировщиков, без которых ему не справиться.

Что касается элементов звука и видео на Web-сайте, то для их использования необходим соответствующий материал. Это значит, что вы должны лицензировать используемый материал или создать собственный — а это огромное количество другой интересной работы. Вам понадобятся помещения, кастинги, творческие споры, костюмерная и множество всего, чем нужно будет грамотно руководить для получения желаемого результата. После этого необходимо будет отредактировать полученный материал, сжать его и отформатировать Web-совместимый файл. Можете себе представить, насколько трудоемким и материально затратным оказывается разработка мультимедийного контента, хотя, конечно, это зависит от проекта.

Объединение всего с помощью HTML

Существует слоган: “канал, который соединил мир”. Так и HTML, как клей, слепил Всемирную Паутину. Язык гипертекстовой разметки HTML — обычный язык программирования, поясняющий вашему Web-браузеру последовательность расположения и форматирования элементов на Web-странице: текста, таблиц, графики, каскадных таблиц стилей CSS (Cascading Style Sheets) и медиаданных.

Каждая страница Web-сайта (рис. 2.8) разрабатывается с помощью HTML, даже если HTML служит лишь основой для сайта, созданного на другом языке программирования. HTML-код можно написать в любом, даже самом примитивном, текстовом редакторе. Вы можете воспользоваться и специализированным редактором, например BBEdit. После того как вы закончите с кодом, загрузите файл в Web-браузере для просмотра результатов.

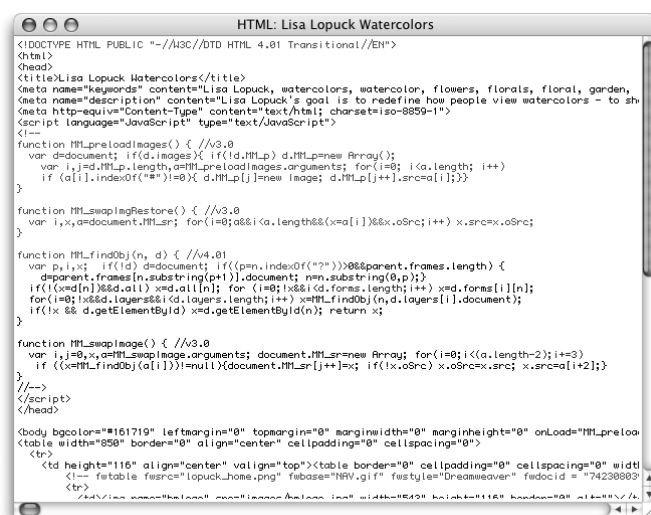


Рис. 2.8. Как и у монеты, у Web-страницы есть две стороны: красивое представление в окне Web-браузера и скучный HTML-код, позволяющий этой странице так отображаться

Функциональные возможности HTML ограничены. Например, нельзя создать электронный магазин, используя только HTML. Для такого случая вам придется задействовать более тяжелую артиллерию — другие языки программирования, которые описаны в главе 15.

Базы данных, программирование и другие сложности

Для разработки Web-сайта, на котором кроме красивого текста и графики будет что-то еще, вам нужно будет подключать программистов. Специального программирования требуют сайты, на которых реализованы следующие задачи: просмотр персональных сообщений, по-

купка продуктов при помощи кредитной карточки, регистрация каких-либо событий и многое другое. Вы не сможете заложить в Web-сайт функциональность такого рода, используя только HTML.



Программные средства разработки, например ColdFusion, и языки программирования, например ASP (Active Server Pages), можно интегрировать в HTML-код Web-страницы, создав своего рода гибрид языков программирования. Теги HTML отвечают за расположение элементов на странице, а языки программирования осуществляют остальные манипуляции, например связь с базой данных для отображения товара. На рис. 2.9 показан пример соединения с электронной базой данных при помощи ASP-кода. Из базы можно получить таким образом сведения о каком-то человеке, например его имя, вес и годовой доход. HTML-код ловко форматирует полученную информацию и отображает ее на Web-сайте.



Рис. 2.9. Функции этой страницы нельзя реализовать, используя только средства HTML

Для создания страницы, представленной на рис. 2.9, можно, закатив рукава, написать ASP-код непосредственно на HTML-странице, а можно воспользоваться программным инструментом Dreamweaver и его расширениями. Расширения Dreamweaver (рис. 2.10) доступны на сайте webassist.com. Их применение даст мощный толчок вашей карьере разработчика.

И вот теперь, когда ваша голова идет кругом от этих технических подробностей, пришло время заняться тестированием и запуском разработанного Web-сайта.

Этап 4. Тестирование и запуск



Ни в коем случае не выставляйте сайт в Интернете до того, как не протестируете его и не исправите ошибки. Если вы это сделаете, пользователи заметят ваши ошибки правописания, неработающие ссылки и тому подобное и покинут Web-сайт, сохранив негативное впечатление о вашей деятельности. Перед тем как выставить сайт в Интернете, досконально протестируйте его и *отладьте* (устраните ошибки).

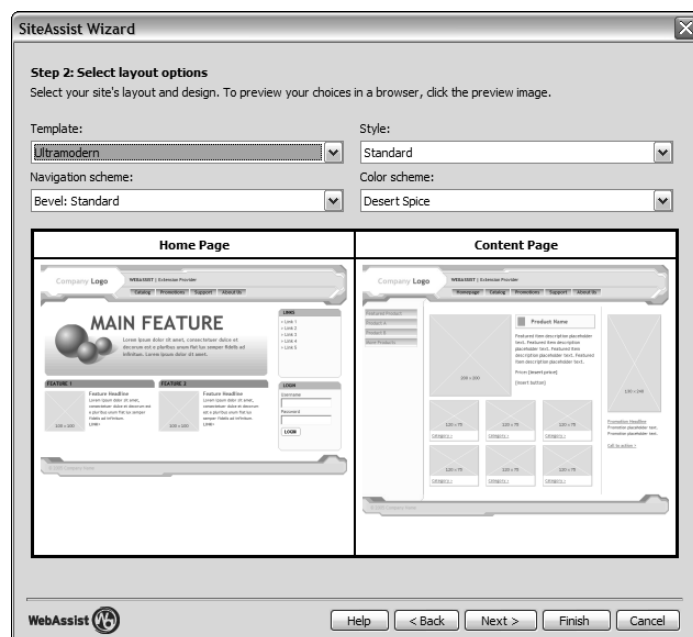


Рис. 2.10. На сайте WebAssist доступны инструменты, которые помогут быстро и эффективно создать сложный Web-сайт

Проверка качества

Для разработки сайта проделано много работы, и, конечно, где-то вы допустили ошибку. Планируйте оставить немалое время в конце работы над проектом для проверки его качества. Для тестирования и отладки крупных коммерческих проектов мне требуется целый месяц работы (а иногда и этого недостаточно). И все равно после того, как сайт выставлен в Интернете, мы еще дорабатываем последние штрихи.

Кроме проверки на наличие грамматических ошибок, пропущенных картинок, неработающих ссылок и т.д., обязательно проверьте работу сайта на различных платформах (Windows и Mac) и в браузерах (Internet Explorer и Mozilla), чтобы убедиться в корректности отображения сайта. Еще предстоит проверка на быстроту загрузки всех страниц сайта при разных скоростях интернет-соединения. Роскошные эффекты, например Flash-анимация, видео, звук, или недостаточно сжатые изображения могут сильно затормозить загрузку сайта даже при высоких скоростях.

Обратите внимание на небольшое отличие в качестве изображений, представленных на рис. 2.11, и огромную разницу в размерах файлов и, как следствие, в скорости загрузки.

День запуска

При проектировании и тестировании Web-сайта вы не дадите никому взглянуть на него даже хоть одним глазком. До презентации сайта в Интернете вы будете прятать его от посторонних глаз, размещая на разных серверах, используя *IP-блокировщики* (специализированное программное обеспечение, позволяющее только определенным компьютерам иметь доступ к серверу) или просто разрабатывая сайт *локально* (только на своем компьютере). Итак, в день запуска Web-сайта вы должны снять все ваши блокировки для того, чтобы пользователи Интернета получили возможность просмотреть ваш Web-сайт.

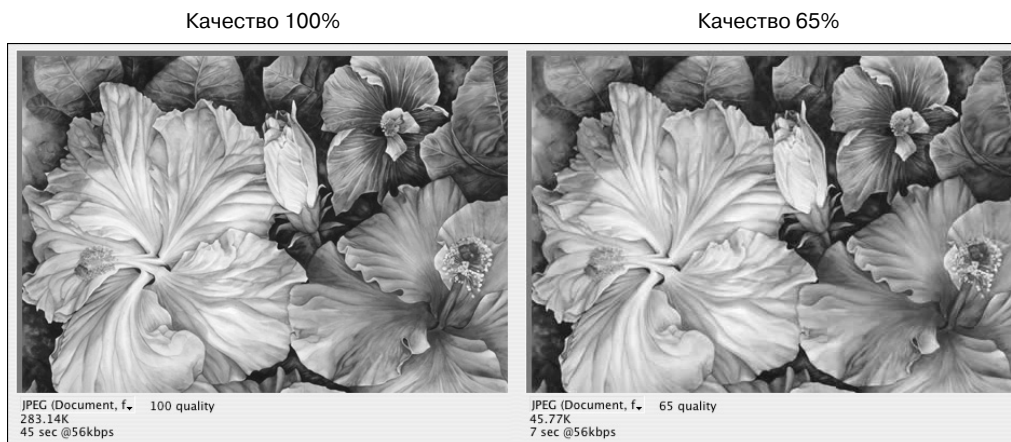


Рис. 2.11. При 65%-ном JPEG-сжатии изображения картинка загружается за 7 с, в отличие от 45 с загрузки несжатого изображения при той же скорости 56 Кбит/с



Следует создать план работ, которые необходимо выполнить до запуска сайта, для себя и команды разработчиков. Работа по получению нового URL-адреса (Web-адрес, например `www.companyname.com`) может занять несколько дней, что надо обязательно предусмотреть при составлении плана. Удаление IP-блокировщика и установка *переедресаций* (перенаправление пользователей на другой URL-адрес после того, как они ввели данный адрес) — это задачи, на выполнение и проверку которых уходит время, что также надо учесть при составлении плана запуска сайта.

Этап 5. Поддержка

После того как сайт открыт в Интернете, начинается этап работ, связанных с его поддержкой. Как только Web-сайт будет размещен на сервере, может возникнуть целый ряд проблем. Я обычно внимательно наблюдаю за сайтом в течение первых двух недель после его размещения на новом месте в Интернете. На всякий случай не отпускайте далеко своих разработчиков!

Кроме того, вам нужно обновлять содержимое сайта: регулярно изменять картинки и текст и исправлять ошибки, не выявленные или не исправленные на стадии проверки качества. Теперь можно заняться доработкой тех функций, которые не удалось реализовать в проекте из-за нехватки времени или денежных средств, а затем добавить их на сайт как обновление. Обновления и дополнения к сайту — это новые Web-проекты, для успешного осуществления которых вам придется повторить пять этапов сначала.