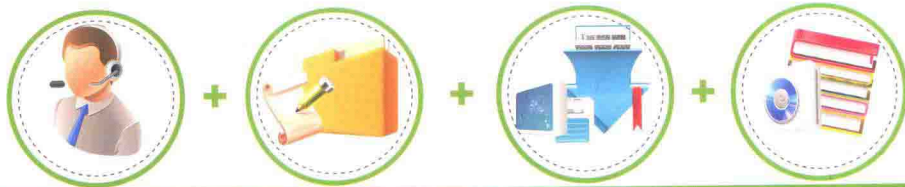




Web客户端开发

——HTML 5+CSS+JavaScript实例教程

王 黎 主 编
张希文 段炬霞 副主编
刘军玲 郭洪荣



- ◆ 以基础知识—实用技术—案例实训为主线
- ◆ 理论知识与实例合理结合，内容结构清晰，易学易懂
- ◆ 内容新颖，将新知识、新技术和新案例引入教程
- ◆ 配备免费教学资源——电子课件、实例源代码、习题答案等



全国高等院校应用型创新规划教材·计算机系列

Web 客户端开发—— HTML 5 + CSS + JavaScript 实例教程

王 黎 主 编

张希文 段炬霞
副主编

刘军玲 郭洪荣

清华大学出版社
北 京

内 容 简 介

HTML 5、CSS 3 和 JavaScript 是网站前端开发的主要应用技术,本书以结合实例和小型案例实训的形式,逐一详细讲解这三大核心技术的基础知识,包括 HTML 5 网页设计的文档结构、常用标记、表单的使用及 HTML 5 新增的标记和属性,CSS 在网页中的应用、CSS 3 新增的功能,以及 JavaScript 语言基础、内置对象、对象编程、JavaScript 操作 HTML 5 + CSS 实现网页设计的方法和技巧。

本书结构安排合理,实用性强,既可作为应用型本科或高职高专计算机专业 Web 客户端开发课程的教材,又可作为自学人员学习网站前端开发的技术参考资料或培训教材。通过对本书实例和小型案例的学习和操练,读者能够很快掌握书中的知识,提高网站前端开发的实战能力。

本书还提供了实例和小型案例的源代码,供读者直接查看和调用;同时还提供了教学课件,便于教师授课使用,也为学生自学带来了方便。

本书封面贴有清华大学出版社防伪标签,无标签者不得销售。

版权所有,侵权必究。侵权举报电话:010-62782989 13701121933

图书在版编目(CIP)数据

Web 客户端开发——HTML 5+CSS+JavaScript 实例教程/王黎主编. —北京:清华大学出版社,2017
(全国高等院校应用型创新规划教材·计算机系列)

ISBN 978-7-302-48232-1

I. ①W… II. ①王… III. ①超文本标记语言—程序设计—高等学校—教材 ②网页制作工作—高等学校—教材 ③JAVA 语言—程序设计—高等学校—教材 IV. ①TP312 ②TP393.092

中国版本图书馆 CIP 数据核字(2017)第 203093 号

责任编辑:汤涌涛

封面设计:杨玉兰

责任校对:宋延清

责任印制:杨 艳

出版发行:清华大学出版社

网 址: <http://www.tup.com.cn>, <http://www.wqbook.com>

地 址:北京清华大学学研大厦 A 座 邮 编:100084

社总机:010-62770175 邮 购:010-62786544

投稿与读者服务:010-62776969, c-service@tup.tsinghua.edu.cn

质量反馈:010-62772015, zhiliang@tup.tsinghua.edu.cn

课件下载: <http://www.tup.com.cn>, 010-62791865

印 装 者:北京密云胶印厂

经 销:全国新华书店

开 本:185mm×260mm 印 张:17 字 数:356千字

版 次:2017年9月第1版 印 次:2017年9月第1次印刷

印 数:1~2000

定 价:42.00元

产品编号:072524-01

前 言

从目前国内互联网的发展趋势来看, Web 前端开发已经成为一门广泛应用的技术, 各行各业对网站的要求越来越高, 对网页设计开发人才的需求也大大增加。Web 标准和 CSS 技术已经成为一种潮流和趋势。

作者结合自己多年积累的相关课程教学经验和 Web 开发经验, 编写了本书。本书是为计算机相关专业的学生以及对网站前端开发感兴趣的读者编写的, 旨在培养读者的网站开发能力, 以适应网络社会对这方面人才的需求, 让读者通过学习, 成为一名精通 Web 前端开发的能手。

本书以作者参与的院级科研项目“校园电子商务网站开发研究”为案例参考背景, 将构建商务网站时需要的典型应用作为书中的案例, 引入网站前端开发所需要的关键技术和开发语言。

本书合理安排了基础知识和实战训练的比例, 基础知识以“够用”为度, 用简明清晰的语言阐述, 结合图表来表达, 符合学生的学习特点和认知规律。

本书重点放在专业技能的训练上。几乎每个重要的知识点都有实例供读者参考练习, 并且采用的是人们熟悉的电商网站的精彩部分作为案例。

本书包含三大核心技术, 分别是 HTML 5、CSS 3 和 JavaScript。在内容组织方面, 共分为 9 个教学模块和 1 个综合应用案例模块。

第 1 章 介绍网站前端设计需要的知识, 包括 Web 基础知识、网站开发的基本流程及关键技术、网站开发人才需求, 以及网站开发工具。

第 2 章 属于 HTML 5 部分, 介绍 HTML 5 的新功能、新增标记和属性、废弃标记, 以及 HTML 5 文档中的常用标记、表单元素和 HTML 5 新增的结构化元素。

第 3~5 章 属于 CSS 部分, 主要介绍 CSS 基础、布局方法和样式属性, 对主流浏览器都支持的也比较成熟的 CSS 3 的部分属性做了讲解。

第 6~8 章 属于 JavaScript 部分, 主要介绍 JavaScript 语法基础、函数及其应用、常用的内置对象、常用的文档对象、常用的窗口对象、事件处理等内容。

第 9 章 利用 JavaScript 实现 Canvas 功能, 包括使用 Canvas 绘制基本图形、变换图形及绘制文本等。

第 10 章 综合应用案例, 主要通过商品购物车功能设计, 系统地介绍 HTML 5、CSS 样式和 JavaScript 脚本编程三项技术的综合运用。

本书由王黎任主编, 张希文、段炬霞、刘军玲、郭洪荣任副主编, 其中王黎编写第 2、3、4、7、8、9、10 章, 张希文编写第 1 章, 刘军玲编写第 5 章, 段炬霞编写第 6 章, 邱冬副教授、吕殿基副教授审稿, 编写过程中参考了很多相关技术资料及经典案例, 吸取了许多同仁的宝贵经验, 在此深表谢意!

由于作者水平有限, 虽然对内容进行了反复斟酌和修改, 但书中不足与疏漏之处在所难免, 恳请各位专家和广大读者批评指正。作者 E-mail: wangli@biem.edu.cn。

编 者

第1章 网站前端设计知识准备.....1

1.1 Web 基础.....2

1.1.1 Web 的基本概念.....2

1.1.2 了解“Web 标准”.....2

1.1.3 静态网页.....4

1.1.4 动态网页.....4

1.2 网站开发.....4

1.2.1 网站开发的基本流程.....4

1.2.2 网站开发的人才需求.....5

1.2.3 网站开发的主要技术.....6

1.3 Web 前端工程师需要掌握的技能.....6

1.3.1 Web 前端工程师的工作内容.....6

1.3.2 Web 前端工程师需要掌握的技术.....6

1.4 网站开发工具介绍.....8

1.4.1 原型设计工具.....8

1.4.2 开发工具.....9

1.4.3 调试工具.....9

1.4.4 代码托管工具.....10

本章小结.....10

习题.....11

第2章 HTML 5 基础.....13

2.1 HTML 简介.....14

2.1.1 了解 HTML.....14

2.1.2 HTML 元素.....15

2.2 初识 HTML 5.....16

2.2.1 了解 HTML 5.....16

2.2.2 HTML 5 的新功能.....17

2.2.3 HTML 5 的废弃标记.....18

2.2.4 HTML 5 的新增标记.....18

2.2.5 HTML 5 的新增属性.....19

2.3 HTML 5 的结构.....20

2.3.1 HTML 5 的基本结构.....20

2.3.2 编写第一个符合 W3C 标准的

HTML 5 网页..... 21

2.4 HTML 5 文档的常用标记..... 22

2.4.1 文本段落的相关标记..... 22

2.4.2 图像标记

2.4.3 超链接..... 24

2.4.4 列表..... 26

2.4.5 表格..... 27

2.4.6 HTML 5 的音频和视频..... 29

2.5 HTML 5 的表单元素..... 32

2.5.1 创建表单..... 32

2.5.2 input 输入类型控件..... 32

2.5.3 列表框(select)..... 39

2.5.4 多行文本输入框(textarea)..... 39

2.5.5 表单控件综合示例..... 40

2.6 HTML 5 语义化结构性元素..... 41

2.6.1 新增的主体结构元素..... 41

2.6.2 新增的非主体结构元素..... 46

2.7 小型案例实训：使用结构元素进行

网页布局..... 47

本章小结..... 49

习题..... 49

第3章 CSS 基础..... 51

3.1 CSS 概述..... 52

3.1.1 什么是 CSS..... 52

3.1.2 CSS 的发展史..... 52

3.1.3 CSS 3 简介..... 53

3.2 CSS 的组成..... 53

3.2.1 基本语法规则..... 53

3.2.2 选择符的分类..... 54

3.3 在 HTML 中使用 CSS 的方法..... 61

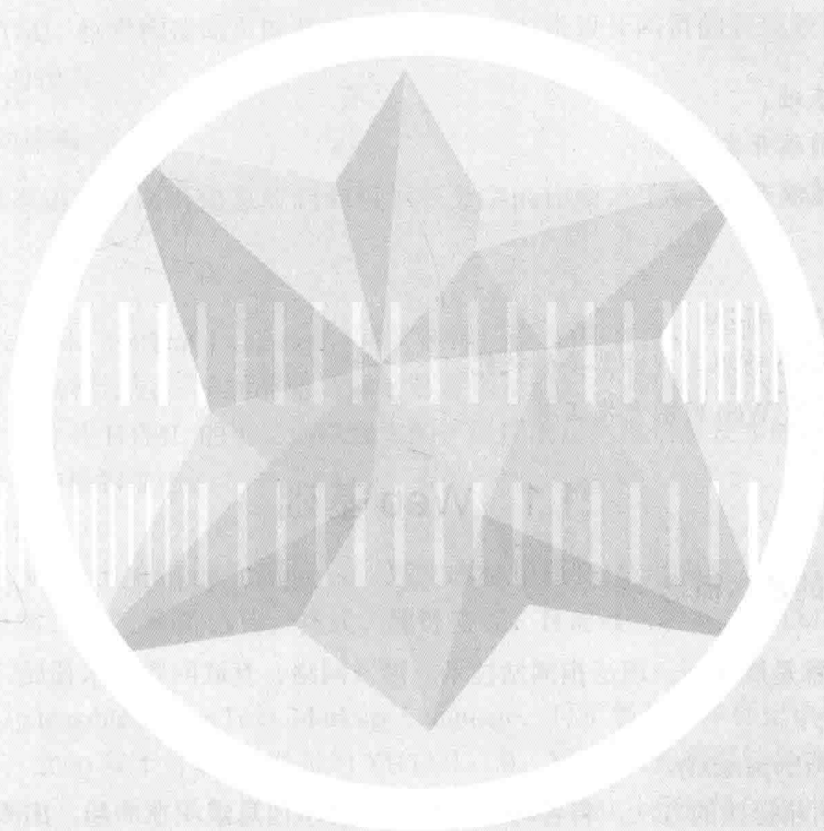
3.3.1 行内样式..... 61

3.3.2 内部样式..... 61

3.3.3 外部样式.....	62	第 5 章 CSS 样式属性.....	103
3.3.4 CSS 的优先级.....	63	5.1 CSS 3 字体相关属性.....	104
3.3.5 常用的 CSS 3 属性前缀.....	63	5.1.1 设置字体属性.....	104
本章小结.....	63	5.1.2 设置字体属性的综合示例.....	106
习题.....	64	5.1.3 CSS 3 新增字体属性.....	106
第 4 章 CSS 布局.....	65	5.2 CSS 控制文本的样式.....	107
4.1 CSS 的盒模型.....	66	5.2.1 文本属性.....	107
4.1.1 盒模型的结构.....	66	5.2.2 设置文本属性的综合示例.....	109
4.1.2 盒模型的元素类型.....	67	5.2.3 CSS 3 新增的文本属性.....	110
4.1.3 使用 DIV.....	69	5.3 使用 CSS 控制背景.....	114
4.1.4 外边距、内边距与边框的 CSS 设置.....	70	5.3.1 背景属性.....	114
4.1.5 CSS 3 对盒模型边框的完善.....	74	5.3.2 背景设置综合示例.....	116
4.2 网页元素的定位.....	78	5.3.3 CSS 3 新增的与背景相关的 属性.....	116
4.2.1 定位属性 position.....	78	5.4 使用 CSS 设置列表样式.....	121
4.2.2 float 浮动定位.....	81	5.4.1 CSS 列表属性.....	121
4.2.3 其他 CSS 布局定位方式.....	83	5.4.2 列表属性的综合实例.....	122
4.3 DIV + CSS 常用的布局方式.....	84	5.5 小型案例实训.....	123
4.3.1 单列水平居中布局.....	85	5.5.1 案例 1: 商品信息展示.....	123
4.3.2 浮动的布局.....	87	5.5.2 案例 2: CSS 制作二级导航 下拉菜单.....	125
4.3.3 div 嵌套布局.....	89	本章小结.....	128
4.3.4 CSS 3 多列布局.....	92	习题.....	128
4.4 列表元素布局.....	94	第 6 章 JavaScript 入门.....	131
4.5 小型案例实训——布局电商网站 首页(制作盒模型).....	95	6.1 JavaScript 概述.....	132
4.5.1 布局网页的总体结构.....	95	6.1.1 认识 JavaScript.....	132
4.5.2 <header></header>区域的 结构分析及布局.....	96	6.1.2 JavaScript 的特点和作用.....	132
4.5.3 <div class="content"></div> 内容区域的设计.....	98	6.1.3 在网页中使用 JavaScript.....	134
4.5.4 <footer></footer>区域的 设计.....	100	6.1.4 JavaScript 代码规范.....	135
本章小结.....	100	6.2 变量、数据类型.....	136
习题.....	101	6.2.1 变量的声明和使用.....	136
		6.2.2 JavaScript 的基本数据类型.....	138
		6.3 表达式与运算符.....	141
		6.3.1 表达式.....	141
		6.3.2 运算符.....	141

6.4 流程控制语句	148	7.4.2 案例 2: 限制多行文本域 输入的字符个数	202
6.4.1 分支结构	148	本章小结	204
6.4.2 循环结构	151	习题	205
6.5 JavaScript 的函数	154	第 8 章 事件处理	209
6.5.1 函数的定义	154	8.1 了解 JavaScript 事件	210
6.5.2 函数的调用	154	8.1.1 JavaScript 的常用事件	210
6.5.3 使用函数的返回值	157	8.1.2 调用事件处理程序的方法	211
6.5.4 函数的嵌套	158	8.2 常用事件在网页中的应用	213
6.5.5 内置函数	158	8.2.1 鼠标事件	213
6.6 小型案例实训——制作简易 计算器	160	8.2.2 表单事件	216
本章小结	162	8.2.3 页面相关事件	221
习题	162	8.2.4 键盘事件	222
第 7 章 JavaScript 中的对象	165	8.3 小型案例实训: JavaScript 实现 广告图像轮播	223
7.1 JavaScript 的常用内置对象	166	本章小结	227
7.1.1 数组对象	166	习题	227
7.1.2 字符串(String)对象	170	第 9 章 利用 JavaScript 实现 Canvas 功能	229
7.1.3 日期(Date)对象	172	9.1 创建 Canvas 元素	230
7.1.4 数学(Math)对象	174	9.2 绘制基本图形	231
7.2 常用文档对象	177	9.2.1 绘制直线	232
7.2.1 文档对象模型(DOM 对象)	177	9.2.2 绘制圆形	232
7.2.2 文档对象的节点树	178	9.2.3 绘制矩形	233
7.2.3 文档对象(document)	185	9.2.4 绘制多边形	234
7.2.4 表单及其控件对象	188	9.3 图形的变换	235
7.2.5 style 对象	190	9.3.1 保存与恢复 Canvas 状态	235
7.3 常用窗口对象	192	9.3.2 移动坐标位置	236
7.3.1 屏幕对象	192	9.3.3 缩放图形	237
7.3.2 window 窗口对象	194	9.4 使用特效	237
7.3.3 浏览器信息对象	196	9.5 绘制文本	239
7.3.4 网址对象	197	9.6 小型案例实训: 用 Canvas 绘制 时钟	240
7.3.5 历史记录对象	198	9.6.1 绘制静态时钟	240
7.4 小型案例实训	200		
7.4.1 案例 1: 将英文单词首字母 改成大写	200		

9.6.2 制作动态时钟.....	244	10.3 实施过程.....	249
本章小结.....	244	10.3.1 购物车的 HTML 结构.....	249
习题.....	245	10.3.2 购物车的样式设计.....	251
第 10 章 综合案例——购物车功能的实现	247	10.3.3 利用 JavaScript 实现购物车功能.....	256
10.1 案例介绍.....	248	参考文献	262
10.2 设计思路.....	248		



第 1 章

网站前端设计知识准备

本章要点

- Web 基础。
- Web 前端开发技术。
- Web 前端开发工具。

学习目标

- 了解 Web 标准。
- 掌握 Web 前端工程师需要的技能。
- 了解常用 Web 前端开发工具。

1.1 Web 基础

1.1.1 Web 的基本概念

Web 的本意是蜘蛛网，现泛指网站技术，涉及网络、互联网等技术领域。该领域中有三个重要的概念。

(1) 超文本(hypertext)。

超文本是用超链接的方法，将各种不同位置的文字信息组织在一起，成为网状文本。通过关键字建立链接，使信息得以用交互的方式进行搜索。

(2) 超媒体(hypermedia)。

超媒体是一种采用非线性网状结构对块状多媒体信息(包括文本、图像、视频等)进行组织和管理的技术。超媒体是超文本和多媒体在信息浏览器环境下的结合。利用超媒体，用户不仅能从一个文本跳到另一个文本，而且可以激活一段声音、显示一个图形，甚至播放一段动画。

(3) 超文本传输协议(HTTP)。

HTTP(Hypertext Transfer Protocol, 超文本传输协议)是在 Internet 中进行信息传送的、被浏览器默认使用的协议。它可以使浏览器更加高效，能使网络传输减少。不仅可用于保证计算机正确快速地传输超文本文档，还可用于确定传输文档中的哪一部分，以及哪些内容首先显示。例如，当用户在浏览器的地址栏中输入“www.biem.edu.cn”时，浏览器会自动使用 HTTP 协议来搜索 <http://www.biem.edu.cn> 网站的首页。

1.1.2 了解“Web 标准”

Web 标准是一个复杂的概念集合，它是由一系列标准组成的，这些标准大部分由 W3C 起草与发布。为了遵循 Web 标准，在进行网页设计时，要从三个方面入手：结构、表现和行为，对应的语言分别是 HTML、CSS 和 JavaScript，它们是网页前端设计的三种基本语言。

其中, HTML 负责构建网页的基本结构, CSS 负责设计网页的表现效果, JavaScript 用来控制网页的行为。

1. 网页的结构

结构化标准语言对网页信息能起到组织和分类的作用, 主要包括 XML、HTML 和 XHTML。

(1) XML。

XML(Extensible Markup Language, 可扩展标记语言)和 HTML 一样, 来源于标准通用标记语言。可扩展标记语言和标准通用标记语言都是能定义其他语言的语言。XML 最初设计的目的, 是弥补 HTML 的不足, 以强大的扩展性满足网络信息发布的需要, 后来逐渐用于网络数据的转换和描述。

(2) HTML。

HTML(HyperText Markup Language, 超文本标记语言)是用来编写网页的语言, 用于声明信息(如文本、图像等)的结构、格式、超链接等。目前最新版本是 HTML 5.0。

(3) XHTML。

XHTML(Extensible HyperText Markup Language, 可扩展超文本标记语言)目前推荐遵循的是 W3C 于 2000 年 1 月 26 日发布的 XHTML 1.0。XML 虽然数据转换能力强大, 完全可以替代 HTML, 但面对无数已有的站点, 直接采用 XML 还没有合适的环境。因此在 HTML 4.0 的基础上, 用 XML 的规则对其进行扩展, 得到了 XHTML。简单地说, 建立 XHTML 的目的, 就是实现 HTML 向 XML 的过渡。

2. 网页的表现

表现标准语言主要用来对网页信息的显示进行控制, 即规定如何修饰网页信息的显示样式。表现标准语言主要有 CSS(Cascading Style Sheets, 层叠样式表)。W3C 创建 CSS 标准的目的是, 用 CSS 取代 HTML 表格, 以控制布局、帧和其他表现。CSS 布局与结构化 HTML 相结合, 能够实现外观和结构的分离, 使站点的访问和维护更加容易。

3. 网页的行为

行为标准语言主要对网页信息的结构和显示进行逻辑控制, 也就是动态地控制网页信息的结构和显示, 实现网页的智能交互。行为标准语言主要有 DOM 和 ECMAScript 等。

(1) DOM。

DOM(Document Object Model, 文档对象模型), 根据 W3C DOM 规范所定义, 是一种与浏览器及平台无关的语言接口, 使得我们可以访问页面中其他的标准组件。

简单地说, DOM 解决了 Netscape 的 JavaScript 和 Microsoft 的 JScript 之间的冲突, 给 Web 设计师和开发者提供了一个标准的方法, 以访问站点中的数据、脚本和表现层对象。

(2) ECMAScript。

ECMAScript 是 ECMA 制定的标准脚本语言(JavaScript)。

1.1.3 静态网页

网页是构成网站的基本元素，是承载各种网站应用的平台。网站就是由网页组成的，网页是一个文件，它可以存放在世界某个角落的某一台计算机中，采用超文本标记语言格式(文件扩展名为.html 或.htm)。网页由网址 URL 来标识和访问，再通过浏览器解释，最后显示给用户。

在网站设计中，纯粹 HTML 格式的网页通常被称为静态网页。静态网页是标准的 HTML 文件，可以包含文本、图像、声音、动画、客户端脚本和 ActiveX 控件及程序。

静态网页是网站建设的基础。静态网页没有后台数据库，不包含开发的程序且不可交互。静态网页制作完成后，页面的内容和显示效果就确定了，除非修改页面代码。因此，静态网页更新起来相对比较麻烦，适用于更新较少的展示型网页。

1.1.4 动态网页

动态网页能对用户请求做出动态响应，可以为不同的用户提供个性化的服务。动态网页文件以.aspx、.asp、.jsp、.php、.perl、.cgi 等形式为后缀，并且在动态网页的网址中，经常会出现一个标志性的符号“?”。

动态网页一般以数据库技术为基础，可以与后台数据库进行交互并进行数据传递，可以大大降低网址维护的工作量。采用动态网页技术的网站能够实现更多的功能，如用户注册、用户登录、在线调查、用户管理、订单管理等。动态网页实际上并不是独立存在于服务器上的静态网页文件，只有用户请求时，服务器才返回一个完整的网页。

1.2 网站开发

1.2.1 网站开发的基本流程

为了加快网站开发的速度和减少失误，应该采用一定的制作流程来策划、设计、制作和发布网站。网站项目开发流程一般包括 5 个步骤。

1. 需求分析

分析主要用户的特点、网站功能、网站的风格、网站的类型、网站的版块、网站的域名及空间等。

2. 平台规划

(1) 内容规划：确定网站要经营的内容如何在网站中得到体现。然后把内容包装成栏目；再结合网站的主题进行风格策划和版式设计，包括全局、导航、核心区、内容区、广告区、版权区及板块设计。绘制网站内容板块草图。

(2) 网站功能规划：主要是管理功能和用户功能。

3. 设计阶段

根据网站草图，由美工制作成效果图。

4. 项目开发阶段

前端和后台可以同时进行。前端开发工程师根据设计效果，负责制作网站静态界面；程序开发工程师根据网站功能，进行数据库设计和代码编写。

5. 测试和发布

对网站进行系统测试，包括兼容性测试、功能测试、性能测试、安全测试等，并根据测试解决问题。

测试完成后，通过服务器发布网站，包括申请域名、部署实施。

网站发布并正式运行后，需要对网站的内容持续进行更新，对网站中出现的错误持续进行修改和完善，提升网站的安全性及性能等。

1.2.2 网站开发的人才需求

网站开发是一个系统工程，需要多种人员协调配合，共同完成。通常，开发网站的过程中，需要有下列人员。

(1) 项目经理。进行项目的管理和协调，合理分配和使用资源，保证项目按计划顺利进行。

(2) 内容编辑。Web 的重要特征是媒体特征，通过网站，将信息更为有效地传达给最终用户，就是网站内容编辑角色的主要职责。网站内容编辑要具有较好的写作能力，并且对网站要传达的内容以及客户的心理都要有充分的理解和把握。

(3) 网站结构规划。将信息内容进行合理的编排，使用户可以方便地找到需要的信息，并且设计用户访问的工作流程。

(4) 美工设计。主要负责 Logo、按钮等的设计，图片的创意与设计，色彩的搭配，以及菜单、表格等的合理使用。同时，还要负责网站有关多媒体动画或音视频应用功能的实现，与前端开发人员共同完成网页的创建、维护和更新工作。

(5) 主页制作。内容与结构都已经确定，需要由专业的 Web 前端工程师使用专业的制作工具进行主页等内容的制作。

(6) 软件程序开发。软件开发工程师主要负责与 Web 相关的基于网络数据库系统的应用软件开发工作。

(7) 系统管理。对服务器进行管理和维护。

(8) 文档管理。对项目中的所有文档进行编辑和管理。

(9) 质量测试。对开发的产品进行测试，发现其中的技术问题和错误，及时地向项目团队报告情况，并监督相关人员解决技术问题。



1.2.3 网站开发的主要技术

网站开发技术分为 Web 前端技术和 Web 后台技术。

(1) HTML: 用于静态页面开发, 是网站建设的基础。

(2) CSS: 对网页内容的显示样式进行控制。

(3) JavaScript: 脚本语言, 为静态页面增添动态交互功能。

(4) PHP: HTML 内嵌式语言, 一种在服务器端执行的嵌入 HTML 文档的脚本语言。

(5) JSP: 是一种动态页面技术。用 JSP 开发的 Web 应用可以跨平台运行。

(6) Python: 是一种面向对象的解释型计算机程序设计语言。

(7) ASP.NET: 是 .NET Framework 的一部分, 是一种使网页中的脚本可由因特网服务器执行的服务器端脚本技术。

1.3 Web 前端工程师需要掌握的技能

1.3.1 Web 前端工程师的工作内容

Web 前端开发是指利用 HTML、CSS、JavaScript、DOM 等各种 Web 技术进行产品的界面开发。其工作目标是制作标准优化的代码, 并增加动态交互功能, 同时, 结合后台开发技术实现整体应用目标, 通过技术改善用户体验。

前端开发技术发展得越来越成熟, 且适用范围更广, 如 HTML 5 可以替代原生 APP, JavaScript 能够用于数据库操作, NodeJS 能让 JavaScript 在服务器端运行等。

Web 前端工程师目前已经成为业界很普遍的工作岗位, 有较大的市场需求, 在职业发展中也逐步形成了体系。Web 前端工程师的职业方向大致有两种: Web 前端工程师和 Web 架构师。Web 前端工程师通过积累和对产品、项目的深入理解, 及对技术的进一步研究和理解, 将能更好地规划和设计 Web 架构的应用服务, 并逐步成长为 Web 架构师。

Web 前端工程师的工作内容主要包括以下几点。

(1) 为网站上的产品和服务实现一流的 Web 页面, 优化代码并保持良好的兼容性。

(2) 负责产品整体前端框架的搭建。

(3) 参与产品的前端开发, 与后台工程师协作, 优质高效地完成产品的数据交互、动态信息展示。

(4) 使用 JS 等编写封装良好的前端交互组件, 维护及优化网站前端页面的性能。

(5) 研究和探索创新的开发思路, 以及应用最新的前端技术。

1.3.2 Web 前端工程师需要掌握的技术

1. HTML

HTML(HyperText Markup Language, 超文本标记语言)是一个网页的骨架, 无论是静

态网页还是动态网页，最终返回到浏览器端的都是 HTML，浏览器将 HTML 代码解释并渲染后呈现给用户。HTML 并不是一种编程语言，而是一种标记语言，利用标记来识别和描述网页结构和内容，如标题、段落等。网页中所有定义的色彩、文字、表格、音视频等元素的相关代码，都是编写在 HTML 文件中的。

要成为 Web 前端工程师，必须掌握 HTML 语言，不仅要编写 HTML，更需要对 HTML 的工作原理和各种属性、性能有深入的理解。学习 HTML 最好的方法就是动手编写，而不是单纯记忆 HTML 标记和属性。

2. CSS

CSS(Cascading Style Sheets, 层叠样式表)用来描述网页内容如何显示，是能够真正做到网页表现与内容分离的一种样式设计语言。页面的字体、色彩、背景、行间距、页面布局等展示都是由 CSS 控制的，目前 CSS 3 还具有页面绘图、动画等功能。

Web 前端工程师的重要工作之一就是设计开发美观、清晰、易于阅读的网页，因此，对 CSS 的掌握是必要的。学习 CSS 的最好采用案例学习的方法，通过反复学习、研究、操作各种案例，对 CSS 的属性熟练掌握，并能灵活应用。

3. JavaScript

JavaScript 是应用最为广泛的脚本语言，在网页中，用来添加交互和行为。例如验证表单输入，以确保输入正确有效；更换一个元素或整个网站的风格；使浏览器记住有关用户的资料，方便下一次访问等。

JavaScript 也是最常用的操作网页元素或某些浏览器窗口功能的语言。使用 JavaScript 可以访问并展开网页元素的标准列表。例如通过 JavaScript 可以增加 HTML 标记、隐藏 HTML 区块等。

Web 前端工程师必须掌握交互技术，能够熟练使用 JavaScript 进行动画设计、交互设计开发。学习 JavaScript 的方法也是要通过案例学习，来掌握 JavaScript 的语法、方法等，并能很好地应用到网页中。

4. 后台开发技术

目前大多数网站都是动态网站，虽然 Web 前端工程师不需要进行大量的动态网站程序开发，但却经常性地需要与程序开发人员进行配合和做业务衔接，因此掌握一定的动态网站开发技术，对于前端工程师来说也是非常必要的。

Web 前端工程师应该掌握一种动态网站开发技术，例如 ASP.NET、PHP 等，并能具备一定的开发能力，能够理解动态网站开发语言的工作原理。

Web 前端工程师除了具备最基础、最核心的技术外，还需要掌握和具备其他技术和能力，主要是以下几个方面。

(1) 计算机专业知识。包括编译原理、计算机网络、操作系统、算法原理、软件工程、软件测试等专业计算机知识，这些知识能够帮助开发者更好地理解 and 掌握 Web 前端开发技术。



- (2) 知识管理、总结分享的能力。
- (3) 沟通技巧，团队协作开发、需求管理、项目管理的能力。
- (4) 代码模块化开发的基本方法和技术。
- (5) 代码版本管理的技术。
- (6) 交互设计、可用性、可访问性的原理和技术。

1.4 网站开发工具介绍

随着 Web 应用的发展，人们对 Web 应用系统和 Web 应用程序的开发技术提出了更高的要求，同时，Web 前端开发也面临着越来越复杂的开发环境。在这种背景下，如何高效地创建稳定、可靠和安全的 Web 应用程序，是 Web 前端工程师面临的重要挑战。选择合适的开发工具，能够帮助 Web 前端开发人员更高效地实施 Web 前端开发。

Web 前端开发工具根据开发的阶段和用途不同，可以分为 Web 设计工具、Web 开发工具、Web 管理与维护工具。为保证代码质量，提高开发效率，在 Web 开发过程中还需要用到 Web 调试工具，对代码进行调试。Web 前端开发常用的工具如表 1-1 所示。

表 1-1 Web 前端开发常用的工具

开发阶段	工 具
原型设计	Axure RP
	Microsoft Office Visio
技术开发	Adobe Dreamweaver
	Oracle NetBeans
	Microsoft Visual Studio
Web 调试	Firefox
	Google Chrome
	Internet Explorer
代码托管	GitHub
	SCN
项目管理	Microsoft Project
	Collabtive

下面简单介绍一下前几种工具，感兴趣的读者可以查阅相关的资料进行深入学习。

1.4.1 原型设计工具

在进行需求分析时，用户的口头描述和想法有时并不一致，而在实际工作中，用户对于图形化的沟通交流更容易理解。原型设计是将页面的模块、元素、人机交互的形式，利用线框描述的方法，将功能更具体、生动地表达出来。原型设计是交互设计师与客户、产

品经理、网站开发工程师沟通的最好工具。

Axure 是目前最受关注的原型设计工具, Axure 借鉴 Office 的界面, 为用户提供了丰富的组件样式, 网站或软件设计师可以通过组件的方式快速建立带有注释的原型(流程图、线框图)。Axure RP 软件可以通过官方网站(<http://www.axure.com>)下载试用。

1.4.2 开发工具

Web 开发工具主要用于对 HTML、CSS 和 JavaScript 程序的编写, 将好的设计思路更好地呈现出来。

1. Adobe Dreamweaver

Adobe Dreamweaver 是第一个针对专业网页设计师特别开发的可视化网页开发工具, 可用来设计并部署网站和 Web 应用程序。

利用 Dreamweaver, 可以很容易地制作出跨越平台、跨越浏览器限制的网页, 可提供强大的编程环境及基于标准的“所见即所得”的设计界面。

2. Oracle NetBeans

NetBeans 是一个为软件开发者而设计的自由、开放的 IDE(集成开发环境)。

NetBeans 可以帮助开发人员编写、编译、调试和部署 Java 应用, 并将版本控制和 XML 编辑融入其众多的功能之中。NetBeans 可以非常方便地安装于多种操作系统平台。

NetBeans 拥有功能全面的 Web 应用开发环境, 开发者可通过页面检查、CSS 样式编辑器和 JavaScript 编辑器、调试器等工具来提升开发效率。

3. Microsoft Visual Studio

Visual Studio 是微软公司开发的一个丰富的集成开发环境, 可用于创建 Windows、Android、iOS 应用程序以及 Web 应用程序。使用 .NET Framework 的功能, 为开发人员提供了可简化 ASP Web 应用程序和 XML Web Services 开发的关键技术。

1.4.3 调试工具

相对来说, Web 前端开发的调试要简单一些。在 Web 应用开发过程中, 开发人员通常需要借助浏览器等工具来了解程序的执行情况, 从而修正语法错误和逻辑错误, 以确定程序的正确性、安全性和稳定性。目前, 支持各种浏览器的 Web 调试工具比较丰富, 并且各浏览器都默认地内置了开发调试工具。

1. Internet Explorer

Internet Explorer 是微软推出的一款网页浏览器, 是所有新版本的 Windows 操作系统的组成部分。

Internet Explorer 可以在浏览器中交互地突出显示被选择的网页元素, 可以查看 style