

# 《iOS Web应用开发》

## 图书基本信息

书名：《iOS Web应用开发》

13位ISBN编号：9787115323216

出版时间：2013-8-1

作者：皮基 (Andrea Picchi)

页数：358

译者：罗晴明

版权说明：本站所提供下载的PDF图书仅提供预览和简介以及在线试读，请支持正版图书。

更多资源请访问：[www.tushu000.com](http://www.tushu000.com)

## 内容概要

本书介绍了如何使用Web标准技术来为iPhone和iPad制作Web应用。书中利用最前沿的Web和移动技术，演示了如何使用HTML5来完成繁重的基础工作，如何使用CSS3来制作外观，以及如何使用JavaScript来为移动网站或Web应用添加程序逻辑。

通过阅读本书，读者可以掌握面向移动的项目的开发流程。作者逐章递进，引导读者了解iOS设计与开发的各个步骤。读者可以学习到如下知识：

设计针对iPhone、iPad和iPod touch而优化的网站；

在苹果的主要平台上使用Web标准技术开发专业的网站，并了解这对网页开发者及设计师的重要性；

充分利用苹果的多点触控屏幕技术，为Web应用实现最佳的用户体验及可访问性；

为移动网站和Web应用执行可用性测试；

定义为移动开发而优化的敏捷项目流程；

UI设计中认知方法的运用，如感知规律、色彩心理学以及正负空间所蕴含的理念等。

本书将帮助读者交付丰富的、面向触摸的移动用户体验。掌握了本书所介绍的概念和技术后，读者就可以在设计师或工程师的道路上更进一步，创造出精彩的iOS移动项目！

# 《iOS Web应用开发》

## 作者简介

作者简介：

Andrea Picchi

拥有心理学（帕多瓦大学）和计算机科学（比萨大学）双学位，从2007年第一台iPhone面世时起便开始为它设计Web应用。在2008年Apple SDK发布首个版本之后，他开始使用Objective-C进行本地应用的开发。

他曾和SimBin的开发团队AB合作开发过视频游戏RACE07——世界房车锦标赛的官方游戏项目。此外还为iPhone iUI项目的开发者、iPhoneWebDev网站以及斯坦福大学的公开课“iPhone应用程序开发”提供过支持。近些年在欧洲举行的许多重要会议上发表过演讲。

译者简介：

罗晴明

曾任职于网易产品技术中心、搜狐视频移动部门，现就职于百度垂直搜索部，任Web前端工程师。

## 书籍目录

## 目录

第1章 移动触摸设备的新思路	1
1.1 为什么要选择移动互联网	1
1.1.1 从桌面端到移动端	1
1.1.2 移动市场	3
1.1.3 为何现在就进军移动领域	4
1.1.4 面向移动的做法	4
1.1.5 面向移动的指南	5
1.2 Apple的移动设备硬件	6
1.3 小结	6
第2章 为iOS设备打造的敏捷项目	7
2.1 实现移动信息架构	7
2.1.1 信息架构及其重要性	8
2.1.2 遵循移动策略的黄金法则	16
2.1.3 纯内容标注法	17
2.1.4 使用站点导航图来表现信息架构	18
2.1.5 使用线框图草拟信息架构	21
2.1.6 通过原型将交互可视化	22
2.2 移动设计的系统方法	24
2.2.1 苹果设备的可访问性	25
2.2.2 iOS设备的可用性	28
2.2.3 iPhone页面模型	32
2.2.4 iPhone用户界面	33
2.2.5 iPad块模型	35
2.2.6 iPad用户界面	37
2.2.7 用于苹果移动设计的工具	39
2.3 小结	39
第3章 针对iOS设备的Web开发	41
3.1 Web开发工具	41
3.1.1 开发框架	42
3.1.2 移动网站	48
3.2 移动应用	49
3.3 Web开发模型	50
3.3.1 Web开发模型的优缺点	50
3.3.2 开发Web应用的四种方法	54
3.3.3 Web应用和原生应用在用户眼中的区别	56
3.3.4 Web标准：HTML、CSS及JavaScript	59
3.3.5 浏览器对标准的支持情况：WebKit和Safari	86
3.4 SDK开发模型	92
3.4.1 苹果的Objective-C、Cocoa Touch以及Xtools模型	93
3.4.2 我是一台Mac电脑	95
3.4.3 用于苹果移动开发的工具和框架	96
3.5 小结	96
第4章 针对iOS设备的用户界面设计	97
4.1 用户界面设计	97

4.1.1	什么是界面	98	
4.1.2	万物皆界面	100	
4.1.3	负空间	105	
4.1.4	色彩心理学	110	
4.1.5	界面层次	117	
4.1.6	阅读模式	118	
4.1.7	用户界面设计流程	125	
4.2	兼容iPhone和iPad的用户界面设计	127	
4.2.1	调研	128	
4.2.2	结构	130	
4.2.3	美工	133	
4.2.4	交互	133	
4.2.5	可交付结果	134	
4.3	iPad类原生用户界面设计	135	
4.3.1	调研	135	
4.3.2	结构	135	
4.3.3	美工	136	
4.3.4	交互	137	
4.3.5	可交付结果	138	
4.4	iPhone类原生用户界面设计	138	
4.4.1	调研	138	
4.4.2	结构	138	
4.4.3	美工	139	
4.4.4	交互	140	
4.4.5	可交付结果	141	
4.4.6	用于用户界面设计的工具	141	
4.5	小结	142	
第5章	iPhone用户界面设计：简化思维	143	
5.1	草绘用户界面	143	
5.2	简化思维	143	
5.2.1	iPhone是随身携带的设备	144	
5.2.2	iPhone页面模型的本质	144	
5.2.3	iPhone的局限性	144	
5.2.4	用户认知资源的本质	144	
5.3	草绘剖析	145	
5.4	使用工具来设计	147	
5.4.1	探索Balsamiq Mockups界面	148	
5.4.2	描绘关联性	149	
5.5	使用Adobe Fireworks来设计	150	
5.5.1	创建画布	151	
5.5.2	整理图层	152	
5.5.3	布局设计	152	
5.5.4	界面设计	153	
5.5.5	复用设计	158	
5.5.6	用于用户界面设计的工具	160	
5.6	小结	160	
第6章	iPad用户界面设计：逆向思维	161	
6.1	草绘用户界面	161	
6.2	逆向思维	161	

6.2.1	逆向简化	162
6.2.2	删减与优选	163
6.2.3	隐藏与改造	164
6.2.4	收缩与分组	165
6.2.5	“简单?复杂”悖论的关键点	166
6.2.6	草绘用户界面	166
6.3	使用工具进行设计	168
6.4	使用Adobe Fireworks来设计	174
6.4.1	iPad兼容版	174
6.4.2	iPad类原生版	183
6.4.3	用于用户界面设计的工具	190
6.5	小结	191
第7章	WebKit所支持的Web标准：充分发挥Mobile Safari的能力	192
7.1	iPhone和iPad的Web表现对比	192
7.2	HTML5	192
7.2.1	HTML5标记语法	193
7.2.2	HTML5重新定义过的标签	194
7.2.3	HTML5语义	195
7.2.4	HTML5媒体	197
7.3	CSS3	217
7.3.1	前缀	217
7.3.2	圆角边框	218
7.3.3	边框图片	218
7.3.4	渐变	219
7.3.5	盒子尺寸标准	219
7.3.6	盒子阴影	220
7.3.7	轮廓	220
7.3.8	背景尺寸	221
7.3.9	背景起点	221
7.3.10	多重背景	222
7.3.11	文字阴影	222
7.3.12	文本溢出	223
7.3.13	单词换行	223
7.3.14	线上字体	224
7.3.15	轻触高亮	224
7.3.16	多列布局	225
7.3.17	横跨多列	225
7.3.18	过渡	226
7.3.19	变换	226
7.3.20	动画	228
7.3.21	关键帧	230
7.3.22	倒影	230
7.4	JavaScript	231
7.4.1	在网页上添加JavaScript	232
7.4.2	JavaScript结构	233
7.4.3	数据类型分类	234
7.4.4	保留字	235
7.4.5	变量	236
7.4.6	运算符	238

7.4.7	条件语句	239
7.4.8	循环语句	241
7.4.9	函数	242
7.4.10	变量作用域	243
7.4.11	数组	244
7.4.12	字符串	245
7.4.13	对象	246
7.4.14	浏览器对象模型	247
7.4.15	文档对象模型	248
7.4.16	对比DOM和HTML结构	249
7.4.17	操作DOM	250
7.4.18	一些JavaScript的最佳实践	253
7.4.19	与Web标准有关的资源	253
7.5	小结	254
第8章	原生iOS环境开发	255
8.1	配置环境	255
8.2	视口的定义	256
8.3	全屏模式应用	258
8.4	添加主屏幕图标	259
8.5	应用启动图片	260
8.6	应用重定向	261
8.7	配置Head标签部分	262
8.8	原生链接模拟	263
8.9	原生文本模拟	263
8.10	原生元素模拟	264
8.11	原生滚动模拟	265
8.12	原生iOS服务交互	265
8.12.1	电话应用	265
8.12.2	邮件应用	266
8.12.3	短信应用	267
8.12.4	地图应用	267
8.13	触摸事件和手势交互	268
8.13.1	触摸事件模式：触摸并不是点击	269
8.13.2	原生及自定义的触摸事件处理	272
8.13.3	创建可触摸的设计元素	273
8.13.4	设备朝向变更事件	274
8.13.5	屏幕朝向变更的媒体查询	275
8.14	扩展iOS开发框架	276
8.15	用于编写代码的资源	278
8.16	小结	278
第9章	原生iOS设计实现	279
9.1	iPhone页面模型的实现	279
9.2	实现类原生的页面结构	279
9.3	iPhone原生界面模拟	280
9.3.1	顶部栏部分	280
9.3.2	页面标题元素	282
9.3.3	面包屑导航栏	284
9.3.4	宣传内容区	285
9.3.5	菜单区	288

9.3.6	页脚部分	290
9.4	小结	292
第10章	优化iOS Web应用	293
10.1	iPad和iPhone兼容性	293
10.2	性能优化	293
10.2.1	代码优化	294
10.2.2	图片优化	297
10.2.3	应用压缩	300
10.2.4	可用性优化	301
10.3	离线可用的Web应用	305
10.4	移动SEO	308
10.4.1	搜索引擎的结构	309
10.4.2	面向搜索引擎的设计	309
10.4.3	可用于优化和SEO的工具	315
10.5	小结	316
第11章	测试iOS Web应用	317
11.1	Web开发生命周期	317
11.2	Web应用测试	318
11.3	敏捷测试	318
11.4	热图测试	320
11.5	组织一次测试	321
11.5.1	创建用例	321
11.5.2	创建物料	324
11.6	执行测试	325
11.6.1	纸面原型图	326
11.6.2	电子原型	328
11.7	评估测试	328
11.8	评估的变量和反馈	329
11.8.1	点触次数	330
11.8.2	错误次数	330
11.8.3	预计完成时间	331
11.8.4	收集反馈	332
11.8.5	评估技术	333
11.9	测试所需资源	335
11.10	小结	335
第12章	深入挖掘iOS Web应用市场	337
12.1	利用你的移动策略	337
12.2	如何推广你的Web应用	337
12.2.1	利用Beta测试邀请测试者	338
12.2.2	利用新闻发布	339
12.2.3	创建Web应用的宣传网站	339
12.2.4	使用邮件营销	340
12.2.5	创建YouTube视频教程	341
12.2.6	提交到苹果Web应用门户	341
12.2.7	提交到其他Web应用门户网站	342
12.2.8	利用社交网络的传播力	344
12.3	让Web应用创造收益	345
12.4	Web应用市场上的资源	346
12.5	小结	347



第13章	从移动互联网展望普适计算	348
13.1	移动设备、无线网络及蜂窝通信的爆发	348
13.2	下一代触屏和多点触控技术的用户体验	349
13.3	新技术、新用途以及新机遇	349
13.4	多点触控屏幕革命会为下一代计算机带来怎样的改变	350
13.4.1	从家用到普适计算与环境智能	351
13.4.2	通信与普适计算资源	353
13.5	小结	354
索引		355

## 精彩短评

- 1、没意思. 浅薄且混乱....
- 2、如译者序所说，第二章中的移动项目流程图和敏捷优化信息架构流程才是本书最有价值的地方。
- 3、以ios的整个开发流程的为索引，介绍了ios开发的环境与设计、开发（相关的属性、事件）、测试、优化，以及传统应用与web应用的区别。可作为入门级了解ios设计以及开发的相关内容，内容理论多于实践。具体深入还是需要一些专业性强的书籍。
- 4、没什么干货

## 章节试读

### 1、《iOS Web应用开发》的笔记-第337页

“依靠天赋可以赢得比赛，但只有团队合作加上智慧才能赢得冠军。” ----迈克尔·乔丹

### 2、《iOS Web应用开发》的笔记-第304页

平均只需18名测试用户即可发现85%的可用性问题的。

### 3、《iOS Web应用开发》的笔记-第294页

法则1：使用遵循Web标准的代码使用遵循HTML5、CSS3和JavaScript标准的代码。除了使用“纯净”的HTML5语法，这还意味着应该尽量将样式表放入页面的<head>标签中，而将JavaScript脚本放在页面的底部（除了iWebkit框架文件的链接）。这样做是因为将样式表放在页面头部可以极大地提高加载速度。另一方面，将JavaScript脚本放置于页面底部可以避免JavaScript代码阻塞HTTP请求，因为在下载JavaScript脚本的时候，浏览器不会同时下载其他资源，哪怕这些文件位于不同的域名上。这一法则可以使浏览器的解析器工作得更快，并缩短整体的渲染延时。

### 4、《iOS Web应用开发》的笔记-第295页

法则2：精简代码移除不必要或代码的冗余部分，在不需要的场合避免使用tab或是空格。如果能用其他方式达到目的，就尽量不要使用CSS表达式。CSS表达式会以远超我们想象的频率反复计算CSS样式规则，并因此拖累网页的整体性能表现。在为注释、CSS的ID和class、JavaScript变量及函数命名时应尽量简短而有意义。在样式表内尽量合并CSS规则，但保持合理的结构分离。减少网页整体大小，自然也减少了渲染和加载的延时。

### 5、《iOS Web应用开发》的笔记-第163页

简化理论是一种认知理论，旨在解释人类与环境的某些交互对人类思维产生的吸引力。该理论认为有趣的情形要比观察者所预期的更为简单。该理论有一个非常知名的实例，即奥卡姆剃刀原理。

## 版权说明

本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问:[www.tushu000.com](http://www.tushu000.com)