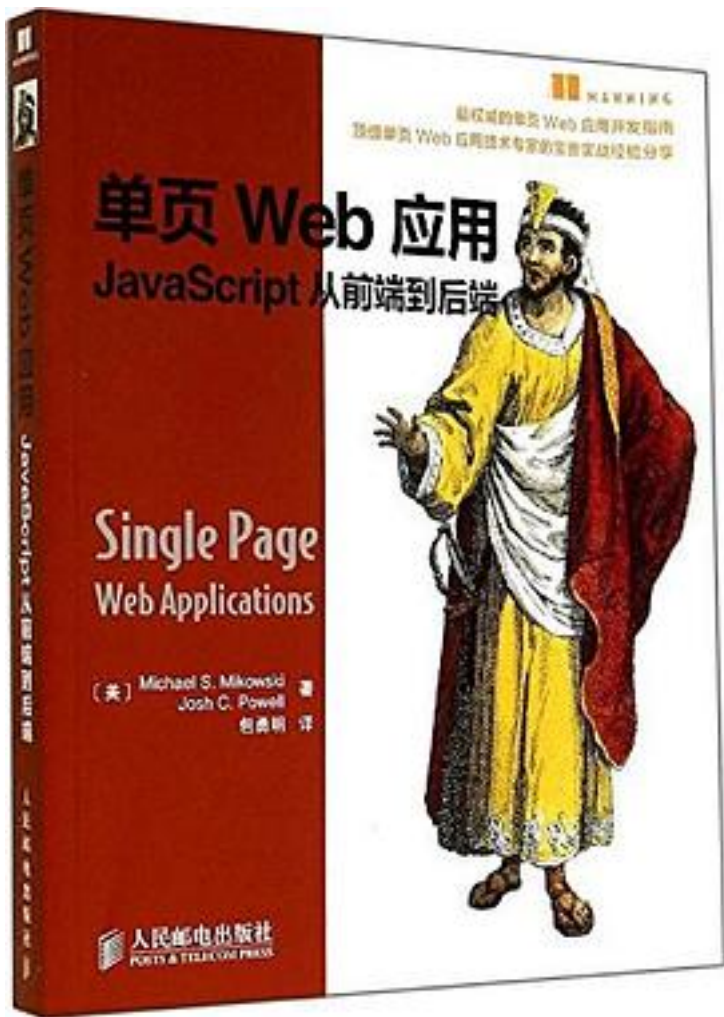


单页Web应用



[单页Web应用_下载链接1](#)

著者:米可夫斯基 (Michael S. Mikowski)

出版者:人民邮电出版社

出版时间:2014-9-1

装帧:平装

isbn:9787115363626

《单页Web应用：JavaScript从前端到后端》是设计和构建大规模JavaScript单页Web

应用(SPA)的宝贵指南，这些应用从前端到后端都使用JavaScript：浏览器端应用、Web服务器和数据库。

《单页Web应用：JavaScript从前端到后端》大约三分之二的內容专门讲解单页Web应用的客户端开发，先定义何为JavaScript单页应用，接着介绍书中使用的单页Web应用的架构，然后依次讲解Shell模块、功能模块、Model模块、Data模块和Fake模块等，其中还专门拿出一整章讲解开发单页Web应用时需要掌握的JavaScript概念和特性；剩下三分之一的內容讲解单页Web应用的服务器端开发和其他与单页Web应用相关的知识，如Node.js、MongoDB、CDN、搜索引擎优化、数据分析、错误日志以及各个层级的缓存等。本书最后的两个附录分别介绍JavaScript的编码规范和单页Web应用的测试。

《单页Web应用：JavaScript从前端到后端》适合Web开发人员、架构师和产品经理阅读，需要读者至少要有些JavaScript、HTML和CSS的开发经验。

专业书评：

“对好几代单页Web应用优化后的深刻见解。”

——摘自Gregory D. Benson撰写的序

“彻底、全面、系统。”

——Mark Ryall，ThoughtWorks公司

“即使你使用了框架，也有必要读一下这本书。”

——Ken Rimple，《Spring Roo in Action》作者

“我强烈推荐本书所展示的技术。”

——Jason Kaczor，SharePoint MVP

“非常棒的指南。”

——Mike Greenhalgh，NHS Wales

作者介绍：

本书作者是架构师和工程管理者。Michael Mikowski已经开发过许多商业级单页Web应用，其中有个平台每年要处理的请求超过1000亿次。Josh Powell搭建过一些流量非常大的网站。

目录: 第一部分 单页应用简介

第1章 第一个单页应用 3

1. 1 定义、一些历史和一些关注点 4

1. 1. 1 一些历史 4

1. 1. 2 是什么导致JavaScript单页应用姗姗来迟 5

1. 1. 3 我们的关注点 8

1. 2 构建第一个单页应用 9

1. 2. 1 定义目标 9

1. 2. 2 创建文件结构 10

1. 2. 3 使用Chrome开发者工具 10

1. 2. 4 开发HTML和CSS	11
1. 2. 5 添加JavaScript	12
1. 2. 6 使用Chrome开发者工具查看应用	17
1. 3 精心编写的单页应用的用户效益	19
1. 4 小结	20
第2章 温故JavaScript	22
2. 1 变量作用域	24
2. 2 变量提升	27
2. 3 高级变量提升和执行环境对象	28
2. 3. 1 提升	28
2. 3. 2 执行环境和执行环境对象	30
2. 4 作用域链	33
2. 5 JavaScript对象和原型链	35
2. 6 函数——更深入的窥探	43
2. 6. 1 函数和匿名函数	43
2. 6. 2 自执行匿名函数	44
2. 6. 3 模块模式——将私有变量引入JavaScript	46
2. 6. 4 闭包	51
2. 7 小结	54
第二部分 单页应用客户端	
第3章 开发Shell	57
3. 1 深刻理解Shell	57
3. 2 创建文件和名字空间	59
3. 2. 1 创建文件结构	59
3. 2. 2 编写应用的HTML文件	60
3. 2. 3 创建CSS根名字空间	61
3. 2. 4 创建JavaScript根名字空间	62
3. 3 创建功能容器	64
3. 3. 1 选取策略	64
3. 3. 2 编写Shell的HTML	64
3. 3. 3 编写Shell的CSS	65
3. 4 渲染功能容器	68
3. 4. 1 将HTML转换为JavaScript	68
3. 4. 2 在JavaScript中添加HTML模板	69
3. 4. 3 编写Shell的样式表	71
3. 4. 4 指示应用使用Shell	73
3. 5 管理功能容器	74
3. 5. 1 编写展开或收起聊天滑块的方法	74
3. 5. 2 给聊天滑块添加点击事件处理程序	76
3. 6 管理应用状态	80
3. 6. 1 理解浏览器用户所期望的行为	80
3. 6. 2 选取一个策略来管理历史控件	81
3. 6. 3 当发生历史事件时，更改锚	82
3. 6. 4 使用锚来驱动应用状态	83
3. 7 小结	89
第4章 添加功能模块	90
4. 1 功能模块策略	91
4. 1. 1 与第三方模块的比较	91
4. 1. 2 功能模块和分形MVC模式	93
4. 2 创建功能模块文件	96
4. 2. 1 规划文件结构	96
4. 2. 2 填写文件	97
4. 2. 3 我们创建了什么	103
4. 3 设计方法API	103

- 4. 3. 1 锚接口模式 104
- 4. 3. 2 Chat的配置API 105
- 4. 3. 3 Chat的初始化API 106
- 4. 3. 4 Chat的setSliderPosition API 107
- 4. 3. 5 配置和初始化的级联 107
- 4. 4 实现功能API 109
- 4. 4. 1 样式表 110
- 4. 4. 2 修改Chat 114
- 4. 4. 3 清理Shell 120
- 4. 4. 4 详细解释执行的过程 125
- 4. 5 添加经常使用的方法 127
- 4. 5. 1 removeSlider方法 127
- 4. 5. 2 handleResize方法 129
- 4. 6 小结 133

第5章 构建Model 134

- 5. 1 理解Model 135
 - 5. 1. 1 我们将要构建什么 135
 - 5. 1. 2 Model做什么 137
 - 5. 1. 3 Model不做什么 137
- 5. 2 创建Model和其他文件 138
 - 5. 2. 1 规划文件结构 138
 - 5. 2. 2 填充文件 139
 - 5. 2. 3 使用统一的触摸——鼠标库 145
- 5. 3 设计people对象 145
 - 5. 3. 1 设计person对象 146
 - 5. 3. 2 设计people对象的API 147
 - 5. 3. 3 给people对象的API编写文档 150
- 5. 4 构建people对象 151
 - 5. 4. 1 创建伪造的人员列表 152
 - 5. 4. 2 开始构建people对象 154
 - 5. 4. 3 完成people对象的构建 157
 - 5. 4. 4 测试people对象的API 164
- 5. 5 在Shell中开启登入和登出的功能 166
 - 5. 5. 1 设计用户登入的体验 167
 - 5. 5. 2 更新Shell的JavaScript 167
 - 5. 5. 3 更新Shell的样式表 169
 - 5. 5. 4 使用UI测试登入和登出 170
- 5. 6 小结 171

第6章 完成Model和Data模块 172

- 6. 1 设计chat对象 172
 - 6. 1. 1 设计方法和事件 173
 - 6. 1. 2 给chat对象的API添加文档 175
- 6. 2 构建chat对象 177
 - 6. 2. 1 先创建chat对象的join方法 177
 - 6. 2. 2 更新Fake以响应chat. join 179
 - 6. 2. 3 测试chat. join方法 181
 - 6. 2. 4 给chat对象添加消息传输功能 182
 - 6. 2. 5 更新Fake, 模拟消息传输功能 187
 - 6. 2. 6 测试chat的消息传输功能 189
- 6. 3 给Model添加Avatar功能 190
 - 6. 3. 1 给chat对象添加Avatar功能 190
 - 6. 3. 2 修改Fake来模拟头像功能 191
 - 6. 3. 3 测试头像功能 192
 - 6. 3. 4 测试驱动开发 193

- 6. 4 完成Chat功能模块 195
- 6. 4. 1 更新Chat的JavaScript 196
- 6. 4. 2 更新样式表 203
- 6. 4. 3 测试Chat UI 207
- 6. 5 创建Avatar功能模块 208
- 6. 5. 1 创建Avatar的JavaScript 209
- 6. 5. 2 创建Avatar的样式表 213
- 6. 5. 3 更新Shell和浏览文档 214
- 6. 5. 4 测试Avatar功能模块 215
- 6. 6 数据绑定和jQuery 216
- 6. 7 创建Data模块 217
- 6. 8 小结 220

第三部分 单页应用服务器

第7章 Web服务器 223

- 7. 1 服务器的作用 223
 - 7. 1. 1 认证和授权 224
 - 7. 1. 2 验证 224
 - 7. 1. 3 数据的保存和同步 225
- 7. 2 Node.js 225
 - 7. 2. 1 为什么选择Node.js 225
 - 7. 2. 2 使用Node.js创建‘Hello World’应用 226
 - 7. 2. 3 安装并使用Connect 229
 - 7. 2. 4 添加Connect中间件 230
 - 7. 2. 5 安装并使用Express 231
 - 7. 2. 6 添加Express中间件 234
 - 7. 2. 7 Express的使用环境 235
 - 7. 2. 8 Express的静态文件服务 236
- 7. 3 高级路由 237
 - 7. 3. 1 用户对象的CRUD路由 237
 - 7. 3. 2 通用CRUD路由 243
 - 7. 3. 3 把路由放到单独的Node.js模块里面 246
- 7. 4 添加认证和授权 249
- 7. 5 Web socket和Socket.IO 251
 - 7. 5. 1 简单的Socket.IO应用程序 251
 - 7. 5. 2 Socket.IO和消息服务器 254
 - 7. 5. 3 使用Socket.IO更新JavaScript 255
- 7. 6 小结 258

第8章 服务器数据库 259

- 8. 1 数据库的作用 259
 - 8. 1. 1 选择数据存储 260
 - 8. 1. 2 消除数据转换 260
 - 8. 1. 3 把逻辑放在需要的地方 261
- 8. 2 MongoDB简介 262
 - 8. 2. 1 面向文档的存储 262
 - 8. 2. 2 动态文档结构 262
 - 8. 2. 3 开始使用MongoDB 263
- 8. 3 使用MongoDB驱动程序 264
 - 8. 3. 1 准备项目文件 265
 - 8. 3. 2 安装并连接MongoDB 265
 - 8. 3. 3 使用MongoDB的CRUD方法 267
 - 8. 3. 4 向服务器应用添加CRUD操作 270
- 8. 4 验证客户端数据 274
 - 8. 4. 1 验证对象类型 274
 - 8. 4. 2 验证对象 276

- 8. 5 创建单独的CRUD模块 283
- 8. 5. 1 组织文件结构 284
- 8. 5. 2 把CRUD移到它自己的模块里面 287
- 8. 6 构建chat模块 292
- 8. 6. 1 开始创建chat模块 293
- 8. 6. 2 创建adduser消息处理程序 295
- 8. 6. 3 创建updatechat消息处理程序 299
- 8. 6. 4 创建disconnect消息处理程序 301
- 8. 6. 5 创建updateavatar消息处理程序 302
- 8. 7 小结 305
- 第9章 单页应用发布准备 306
- 9. 1 单页应用针对搜索引擎的优化 307
- 9. 2 云和第三方服务 310
- 9. 2. 1 站点分析 310
- 9. 2. 2 记录客户端错误 312
- 9. 2. 3 内容分发网络 314
- 9. 3 缓存和缓存破坏 314
- 9. 3. 1 缓存时机 315
- 9. 3. 2 Web存储 316
- 9. 3. 3 HTTP缓存 317
- 9. 3. 4 服务器缓存 320
- 9. 3. 5 数据库查询缓存 325
- 9. 4 小结 326
- 附录A JavaScript编码标准 328
- 附录B 测试单页应用 361
- [\(收起\)](#)

[单页Web应用_下载链接1](#)

标签

JavaScript

前端

前端开发

SPA

web

计算机

JS

编程

评论

用 ES5 + jQuery 做 SPA+后端的书。技术栈在2018年来看比较落伍，但是结构划分的思想依然很值得借鉴参考

虽然这样说有点mean，但作者这一套积累多年的“精心开发SPA”的经验，在今日已经稍显落伍了。首先模块化已经有成熟的工程手段，DOM操作也成了反模式，当然，example 里有关SPA的结构、模型抽象、前后端数据交换和前端工程（缓存策略等）的内容还是有点示范意义的。

从码农到程序员，得一步一步来，这本书无疑是引路者之一。

嗯，很清晰的就理清了思路，通过一个简单应用的构建基本了解了

译者好认真！

这个作者好像也不太喜欢用现成的框架，最近写SPA找了一圈也没发现比较合适又不重的框架，只好自己写了。书里有些地方不大同意

对于新手还是挺难的

内容有点老了，但是对于刚接触SPA的人来说比较有参考价值，可以参考其设计和开发流程，了解前后端用到的一些技术栈。完整地把代码敲一遍，能更好的帮助理解

到node就没什么兴趣了，还是老老实实用框架吧。

作者很幽默

喜欢第二章里对闭包的解释。

一本被现在框架(Vue/React)所替代的书，技术的更新迭代速度太快了。

8012年如何自学前端？——读过时的老书

没读完…作者比较另类…写代码的方式有点点奇怪。

Get one's feet wet.

很落后的技术吧

跟不上时代了，老实用框架吧

挺失望的，本以为能学到多一点的单页面实现原理，结果就讲了一个hash模式，第二部分感觉没有太大的价值去读，换成讲第三方框架都好呀。。

从无到有构建基于nodejs的单页应用，需要一定的js/jquery基础。

spa的书看得很多，这本书绝对力荐。不光因为它的内容，里面的JavaScript书写的规范性，代码优雅的易读性就完全值得一读。

[单页Web应用 下载链接1](#)

书评

不得不说，mikowski真是一个难得负责的作者，他在亚马逊里回复了许多读者的留言，在manning的论坛里也经常出来回答问题，这本书是一本让你快速上手的实战书，里面作者提供了不仅是方法，更多的是他的个人经验，和对新手入门的建议，忘了那些api,那些documents,那些tutorial，跟...

SPA页面 单页面应用，整个页面只有一个HTML页面
以前对于单页面应用的理解很粗浅，就是HTML+CSS+JS写到同一个文件中，其实最核心的技能，关键在于映射关系的设置：
用户初次访问，资源全部加载到浏览器，根据不同的操作展示出不同的页面，前端来维护路由规则，一旦URL改变， ...

Code for most web sites mostly runs on the server. When a user clicks on a link, the site reacts slowly because the browser sends information to the server and the server sends it back again before displaying the results. With near universal availability of...

[单页Web应用 下载链接1](#)