

SVG

Canvas

Threejs

WPO

CSS3

HTML5

APIs

Data

Device

JavaScript

jQuery

Zepto

Cocos2D

Createjs

指尖上行

移动前端开发进阶之路

腾讯互动娱乐TGideas 编著



中国工信出版集团



人民邮电出版社
POSTS & TELECOM PRESS



指尖上行

移动前端开发进阶之路

腾讯互动娱乐TGideas©编著

人民邮电出版社
北京

图书在版编目（C I P）数据

指尖上行：移动前端开发进阶之路 / 腾讯互动娱乐
TGideas编著. — 北京：人民邮电出版社，2017.9
ISBN 978-7-115-46458-3

I. ①指… II. ①腾… III. ①移动通信—技术开发
IV. ①TN929.5

中国版本图书馆CIP数据核字(2017)第186667号

内 容 提 要

本书基于 TGideas 团队多年的研究、应用及沉淀，旨在向各位读者介绍在移动前端开发的各类技术知识，帮助读者在移动前端领域更上一层楼。

全书从基础的移动页面布局和常见前端框架，到进阶的移动页面动画技术、Web API 及性能优化，再到各类实战案例剖析等，详细讲解了技术层面的各类知识和心得。另外，书中还介绍了常用的数据分析方法，帮助读者验证项目效果。

本书适合网页重构和 Web 前端开发岗位的从业人员使用，也适合有一定的移动页面开发基础的读者阅读。更多延伸内容，我们会给出官方文档地址，方便读者进一步了解。

-
- ◆ 编 著 腾讯互动娱乐 TGideas
责任编辑 孙 媛
责任印制 陈 犇
 - ◆ 人民邮电出版社出版发行 北京市丰台区成寿寺路 11 号
邮编 100164 电子邮件 315@ptpress.com.cn
网址 <http://www.ptpress.com.cn>
北京市雅迪彩色印刷有限公司印刷
 - ◆ 开本：787×1092 1/16
印张：16.25
字数：401 千字
印数：1—3 000 册
-

定价：79.00 元

读者服务热线：(010)81055410 印装质量热线：(010)81055316

反盗版热线：(010)81055315

广告经营许可证：京东工商广登字 20170147 号

目录

CONTENTS

01

移动页面开发 /007

1.1 页面布局 /008

- 1.1.1 Viewport /008
- 1.1.2 布局形式 /014
- 1.1.3 Media Queries /018
- 1.1.4 屏幕适应 /022
- 1.1.5 内容排布技巧 /034

1.2 页面调试 /037

- 1.2.1 Chrome开发者工具 /037
- 1.2.2 Mac电脑调试 /039
- 1.2.3 自研工具 /040

1.3 常用库和框架 /042

- 1.3.1 jQuery Mobile /042
- 1.3.2 Zepto /055
- 1.3.3 Cocos2d /069
- 1.3.4 CreateJS /079

02

技术创意形式 /087

2.1 动画形式 /088

- 2.1.1 CSS3 /088
- 2.1.2 帧动画 /094
- 2.1.3 Canvas /100
- 2.1.4 SVG /105
- 2.1.5 Three.js /110

2.2 移动设备Web API详解 /114

- 2.2.1 视频 (Video) /114
- 2.2.2 音频 (Audio) /115
- 2.2.3 媒体流 (getUserMedia) /119
- 2.2.4 Web Speech /120
- 2.2.5 Web Audio API /121
- 2.2.6 地理定位 (Geolocation API) /123
- 2.2.7 陀螺仪 /126
- 2.2.8 设备震动 (Vibration API) /128
- 2.2.9 电池状态 (Battery API) /129
- 2.2.10 环境光 (Ambient Light) /130
- 2.2.11 网络信息 /132
- 2.2.12 平台JSSDK /133

2.3 WebVR /134

- 2.3.1 实现步骤 /134
- 2.3.2 常用WebVR解决方案 /135

2.4 创意点 /142

- 2.4.1 基于微信录音接口 /142

- 2.4.2 基于微信语音识别 /144
- 2.4.3 基于摄像头和相册 /145
- 2.4.4 基于人脸识别 /147
- 2.4.5 基于陀螺仪 /148
- 2.4.6 基于手势 /150
- 2.4.7 基于Websocket /151

03

页面性能优化 /153

3.1 优化原理 /154

- 3.1.1 为什么优化 /154
- 3.1.2 优化思路 /154

3.2 资源优化 /155

- 3.2.1 图像 /155
- 3.2.2 音频 /162
- 3.2.3 视频 /166
- 3.2.4 代码 /168

3.3 加载优化 /170

- 3.3.1 加载原理 /170
- 3.3.2 浏览器分析 /175
- 3.3.3 加载优化实战 /179

3.4 脚本优化 /181

- 3.4.1 执行效率 /181
- 3.4.2 Web Worker /188

04

页面效果验证 /191

4.1 为何验证 /192

4.2 数据埋点 /193

- 4.2.1 什么是数据埋点 /193
- 4.2.2 什么数据需要埋点 /193
- 4.2.3 数据分析平台 /197

4.3 分析数据 /200

- 4.3.1 数据收集 /200
- 4.3.2 数据整理 /200
- 4.3.3 数据分析 /202

4.4 数据参考 /210

- 4.4.1 资源投放数据参考 /210
- 4.4.2 H5用户行为数据参考 /211
- 4.4.3 转化率数据参考 /220

05

大型项目实战 /221

5.1 UP+2014邀请函 /222

5.1.1 设计风格定稿 /222

5.1.2 前端体验定位 /223

5.1.3 单屏滑动 /224

5.1.4 CSS3动画的应用 /226

5.1.5 SVG的应用 /230

5.2 TGA城市拉力赛 /232

5.2.1 地理位置的使用 /232

5.2.2 音频的作用 /233

5.3 全民突击明星队长 /235

5.3.1 需求创意形式和技术点分析 /235

5.3.2 执行中的问题和解决方法 /236

5.3.3 微信视频自动播放问题 /240

5.4 全民超神明星皮肤 /242

5.4.1 需求创意形式和技术点分析 /242

5.4.2 执行中的技术点和实践 /242

5.4.3 执行中常见的兼容性问题和解决方法 /246

5.5 龙之谷手游WebVR项目 /248

5.5.1 程序与用户共同控制摄像头 /248

5.5.2 多重蒙版贴图 /249

5.5.3 自适应长度文字提示 /253

5.5.4 Unity地形导出 /255

5.5.5 3ds Max动画导出问题 /255

附录 /257

1. 微信JSSDK /257

2. 手机QQ JSSDK /258



指尖上行

移动前端开发进阶之路

腾讯互动娱乐TGideas©编著

人民邮电出版社
北京

图书在版编目（C I P）数据

指尖上行：移动前端开发进阶之路 / 腾讯互动娱乐
TGideas编著. — 北京：人民邮电出版社，2017.9
ISBN 978-7-115-46458-3

I. ①指… II. ①腾… III. ①移动通信—技术开发
IV. ①TN929.5

中国版本图书馆CIP数据核字(2017)第186667号

内 容 提 要

本书基于 TGideas 团队多年的研究、应用及沉淀，旨在向各位读者介绍在移动前端开发的各类技术知识，帮助读者在移动前端领域更上一层楼。

全书从基础的移动页面布局和常见前端框架，到进阶的移动页面动画技术、Web API 及性能优化，再到各类实战案例剖析等，详细讲解了技术层面的各类知识和心得。另外，书中还介绍了常用的数据分析方法，帮助读者验证项目效果。

本书适合网页重构和 Web 前端开发岗位的从业人员使用，也适合有一定的移动页面开发基础的读者阅读。更多延伸内容，我们会给出官方文档地址，方便读者进一步了解。

-
- ◆ 编 著 腾讯互动娱乐 TGideas
责任编辑 孙 媛
责任印制 陈 犇
 - ◆ 人民邮电出版社出版发行 北京市丰台区成寿寺路 11 号
邮编 100164 电子邮件 315@ptpress.com.cn
网址 <http://www.ptpress.com.cn>
北京市雅迪彩色印刷有限公司印刷
 - ◆ 开本：787×1092* 1/16
印张：16.25
字数：401 千字
印数：1—3 000 册
-

定价：79.00 元

读者服务热线：(010)81055410 印装质量热线：(010)81055316

反盗版热线：(010)81055315

广告经营许可证：京东工商广登字 20170147 号

序言

PREFACE

近年来,作为移动页面的一种表现形式,H5在朋友圈的表现受到整个行业的追捧。TGideas作为业内顶尖的创意设计团队,也制作出了不少被广泛传播的H5作品。对于任何项目,优秀的创意都是吸引用户浏览并产生传播的核心之一,而扎实的技术手段则是保证优秀创意得以实现的基石,缺少了技术的保障,再好的创意也是空中楼阁。

在和业界同行的交流过程中,我们发现,由于技术问题无法解决而导致创意调整的情况比比皆是,其最终效果自然会大打折扣。因此在启动一个项目的时候,提前做好技术评估,充分考虑可实现性,对于项目的最终成功会有很大帮助。本书的目的就是帮助技术人员全面掌握包括H5在内的各种类型移动页面的开发要点,将优秀的创意通过技术手段完美呈现出现。

第1章,介绍移动端页面的布局方式和技巧、页面调试方法,以及常见的移动端库和框架,这可以帮助开发人员快速完成移动页面的基础架构。

第2章,讲解一些常用的动画技术形式、移动端设备API、最近非常热门的VR及常见的技术创意组合形式,可以让移动页面实现更多的互动效果。

第3章,介绍页面性能的优化原理,如如何进行资源优化、加载优化及脚本优化,这可以保证较快的访问速度,可以更好地增加用户体验,减少用户流失。

第4章,整理监控移动页面的办法及我们总结的数据参考,让开发人员可以通过数据分析更好地评估作品和发现问题。

第5章,整理我们团队产出的一些优秀项目案例,教大家如何将这些知识运用到实战中及如何解决遇到的问题。

相信大家在完成对本书的学习之后,对于包括H5在内各种类型移动页面的技术实现环节,将更有信心。当然,互联网是一个快速变化的行业,技术的发展日新月异,本书或许可以帮助大家解决眼下移动页面技术上的大部分问题,但谁也无法预计未来会出现什么样的颠覆性技术,因此,我们需要保持对行业的持续关注。

欢迎大家关注TGideas的博客(<http://tgideas.qq.com>)及扫描下面的二维码关注TGideas的微信公众平台账号。我们也会经常更新一些技术文章及实战案例,帮助大家持续提升自己的技术能力。



本书尽可能地将最新的技术内容呈现给大家,但由于操作系统、浏览器内核的持续迭代及各类规则的调整,在本书出版时,可能部分API的兼容性已经有了很大的改观,书中的部分内容也可能在未来的技术革新中渐渐成为历史。请各位读者在具体使用时,做好预研和测试工作。由于技术能力有限,书中难免出现纰漏;另外由于篇幅限制,书中无法将所有内容都讲得特别详细,望各位谅解。

读者在阅读时需要注意,本书部分的CSS代码及JavaScript代码采用了左右的排列方式。

最后感谢每一位为本书撰写内容的小伙伴,感谢人民邮电出版社,希望本书能为在这个行业奋斗的同行人及准备进入这个行业的年轻人提供一点帮助。

目录

CONTENTS

01

移动页面开发 /007

1.1 页面布局 /008

- 1.1.1 Viewport /008
- 1.1.2 布局形式 /014
- 1.1.3 Media Queries /018
- 1.1.4 屏幕适应 /022
- 1.1.5 内容排布技巧 /034

1.2 页面调试 /037

- 1.2.1 Chrome开发者工具 /037
- 1.2.2 Mac电脑调试 /039
- 1.2.3 自研工具 /040

1.3 常用库和框架 /042

- 1.3.1 jQuery Mobile /042
- 1.3.2 Zepto /055
- 1.3.3 Cocos2d /069
- 1.3.4 CreateJS /079

02

技术创意形式 /087

2.1 动画形式 /088

- 2.1.1 CSS3 /088
- 2.1.2 帧动画 /094
- 2.1.3 Canvas /100
- 2.1.4 SVG /105
- 2.1.5 Three.js /110

2.2 移动设备Web API详解 /114

- 2.2.1 视频 (Video) /114
- 2.2.2 音频 (Audio) /115
- 2.2.3 媒体流 (getUserMedia) /119
- 2.2.4 Web Speech /120
- 2.2.5 Web Audio API /121
- 2.2.6 地理定位 (Geolocation API) /123
- 2.2.7 陀螺仪 /126
- 2.2.8 设备震动 (Vibration API) /128
- 2.2.9 电池状态 (Battery API) /129
- 2.2.10 环境光 (Ambient Light) /130
- 2.2.11 网络信息 /132
- 2.2.12 平台JSSDK /133

2.3 WebVR /134

- 2.3.1 实现步骤 /134
- 2.3.2 常用WebVR解决方案 /135

2.4 创意点 /142

- 2.4.1 基于微信录音接口 /142

- 2.4.2 基于微信语音识别 /144
- 2.4.3 基于摄像头和相册 /145
- 2.4.4 基于人脸识别 /147
- 2.4.5 基于陀螺仪 /148
- 2.4.6 基于手势 /150
- 2.4.7 基于Websocket /151

03

页面性能优化 /153

3.1 优化原理 /154

- 3.1.1 为什么优化 /154
- 3.1.2 优化思路 /154

3.2 资源优化 /155

- 3.2.1 图像 /155
- 3.2.2 音频 /162
- 3.2.3 视频 /166
- 3.2.4 代码 /168

3.3 加载优化 /170

- 3.3.1 加载原理 /170
- 3.3.2 浏览器分析 /175
- 3.3.3 加载优化实战 /179

3.4 脚本优化 /181

- 3.4.1 执行效率 /181
- 3.4.2 Web Worker /188

04

页面效果验证 /191

4.1 为何验证 /192

4.2 数据埋点 /193

- 4.2.1 什么是数据埋点 /193
- 4.2.2 什么数据需要埋点 /193
- 4.2.3 数据分析平台 /197

4.3 分析数据 /200

- 4.3.1 数据收集 /200
- 4.3.2 数据整理 /200
- 4.3.3 数据分析 /202

4.4 数据参考 /210

- 4.4.1 资源投放数据参考 /210
- 4.4.2 H5用户行为数据参考 /211
- 4.4.3 转化率数据参考 /220

05

大型项目实战 /221

5.1 UP+2014邀请函 /222

5.1.1 设计风格定稿 /222

5.1.2 前端体验定位 /223

5.1.3 单屏滑动 /224

5.1.4 CSS3动画的应用 /226

5.1.5 SVG的应用 /230

5.2 TGA城市拉力赛 /232

5.2.1 地理位置的使用 /232

5.2.2 音频的作用 /233

5.3 全民突击明星队长 /235

5.3.1 需求创意形式和技术点分析 /235

5.3.2 执行中的问题和解决方法 /236

5.3.3 微信视频自动播放问题 /240

5.4 全民超神明星皮肤 /242

5.4.1 需求创意形式和技术点分析 /242

5.4.2 执行中的技术点和实践 /242

5.4.3 执行中常见的兼容性问题 and 解决方法 /246

5.5 龙之谷手游WebVR项目 /248

5.5.1 程序与用户共同控制摄像头 /248

5.5.2 多重蒙版贴图 /249

5.5.3 自适应长度文字提示 /253

5.5.4 Unity地形导出 /255

5.5.5 3ds Max动画导出问题 /255

附录 /257

1. 微信JSSDK /257

2. 手机QQ JSSDK /258

01

MOBILE WEB PAGES
DEVELOPMENT

移动页面开发

- 页面布局
- 页面调试
- 常用库和框架

1.1 页面布局

1.1.1 Viewport

移动Web开发自然要从Viewport说起，那什么是Viewport呢？字面意思为视图窗口，通俗地讲，就是指移动设备上能用来显示页面的那部分区域。

默认情况下，为了在移动设备上正常显示那些为PC端浏览器设计的页面，部分移动设备上的浏览器会把自己默认的Viewport设为980像素（当然也可能是其他值，各设备之间存在差异）。但是这样带来的后果是浏览器可能会出现横向滚动条（这个和PC端浏览器出现横向滚动条的原理是一样的），而且页面内容会被缩小，需要用户手动放大，浏览体验会很糟糕。那么这个Viewport该如何设置呢？我们从几个基本概念开始讲解。

■ 设备像素

对于设备来说，有两个设备像素：物理像素和独立像素。

设备物理像素是指设备显示屏中使用的最小显示单元，也就是屏幕分辨率，例如，iPhone5的屏幕分辨率是640像素×1136像素，iPhone6的屏幕分辨率是750像素×1334像素。

设备独立像素是指Web编程中的逻辑像素，也就是CSS像素。其实对于前端开发者，最感兴趣的是这个像素值，上面提到的iPhone5的CSS像素是320像素×568像素，在竖屏情况下，如果将一个div的宽度设置为320像素，那么它看起来正好是百分之百的宽度。

■ 像素密度（PPI）

PPI（Pixels Per Inch）是用来表示设备每英寸所拥有的物理像素数目。这个数值越高，我们看到的屏幕越清晰。PPI的计算公式如下。

$$\text{PPI} = \frac{\sqrt{\text{屏幕分辨率宽}^2 + \text{屏幕分辨率高}^2}}{4}$$

如iPhone5，通过上面的公式，可以算出iPhone5的PPI大约等于326。

当显示屏的PPI超过某一数值时，人的肉眼就无法分辨其中的单独像素了，这就是Retina显示屏。根据Apple提出的定义，电脑显示屏PPI高于200、手持平板显示屏PPI高于260及移动电话显示屏PPI高于300的屏幕都称为Retina屏幕，这种屏幕最早出现在iPhone4上。但是我们可以通过另一个数值来判断是否是Retina显示屏，那便是设备像素比。

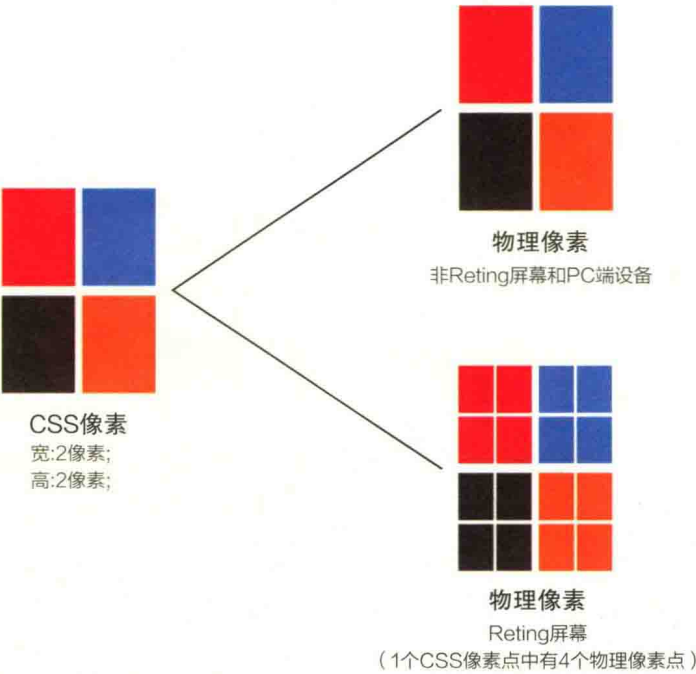
■ 设备像素比 (DPR)

DPR (Device Pixel Ratio) 是指物理像素和CSS像素的比例。
在JavaScript中，可以通过window的devicePixelRatio属性来获取当前设备的DPR，如下。

```
<script>
    var dpr = window.devicePixelRatio;
</script>
```

在CSS中，可以通过device-pixel-ratio、min-device-pixel-ratio及max-device-pixel-ratio媒体查询针对不同像素比的设备进行特殊化的适配。

在前端工作中，CSS像素是常用的尺寸单位。对于常规显示屏来说，物理像素和CSS像素的比值是1:1，1个CSS像素就等于1个屏幕分辨率，但是在Retina屏幕设备中，一个CSS像素可能等于多个物理像素，如下图所示。



例如，iPhone6，设备物理像素是750像素×1334像素，CSS像素是375像素×667像素，计算得到DPR等于2。不同设备的像素比值都不一样，iPhone从4代开始基本都是2（iPhone Plus除外），而Android设备，碎片化很严重，没有一个定值。

关于设备像素、像素密度及设备像素比等具体的数据可以到网站<http://screensiz.es>上查看。

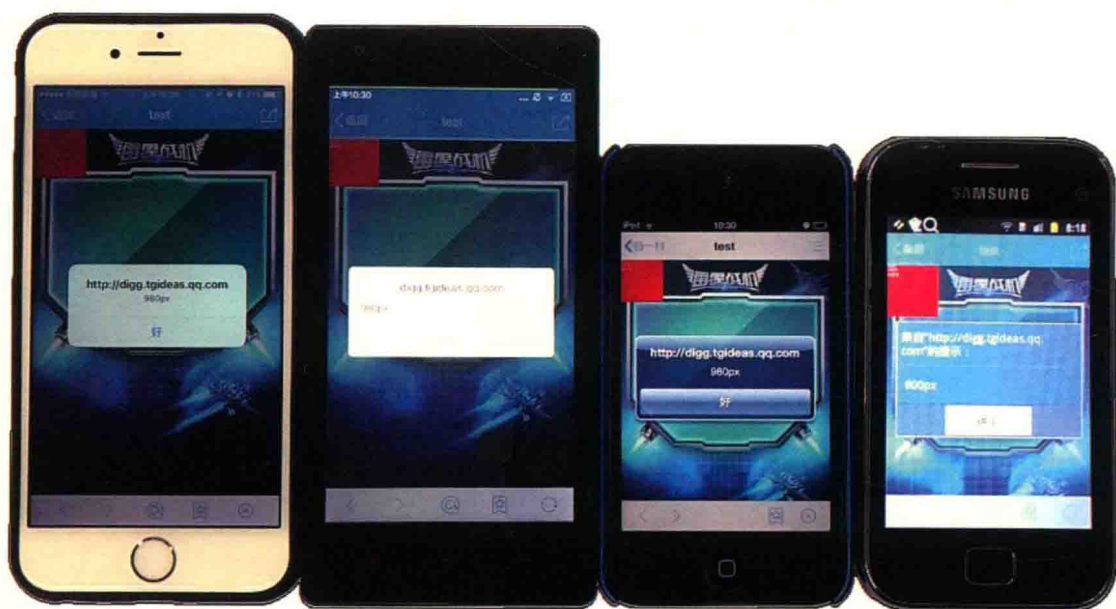
■ 3个Viewport

<1> Layout Viewport

前面提到，移动设备为了不让桌面端页面因为Viewport太窄而出现页面被遮盖或错乱的情况，会默认把Viewport设为一个较宽的值，如980像素等，这样便如同在980像素分辨率的显示器中打开页面一样，如下图所示。



我们将这个默认的Viewport叫作 Layout Viewport，可以通过JavaScript的 `document.documentElement.clientWidth`和`document.documentElement.clientHeight`获取，如下图所示。



<2> Visual Viewport

在浏览器或者App的Webview中的可视区域，称为Visual Viewport，可以通过JavaScript的 window.innerWidth和window.innerHeight获取。它相当于在计算自身的1像素可以显示多少像素的页面内容，因此当用户放大和缩小页面时，它的度量值会随之改变，如下图所示。



<3> Ideal Viewport

Viewport是一个理想中的、抽象的视图。在这个视图下，图片和文字无论在什么设备和分辨率下，看起来都会保持差不多的大小。

Ideal Viewport的宽度并没有一个固定的尺寸，不同的设备之间存在差异。目前大部分机器的宽度如下表所示。

设备	竖屏下宽度
iPhone4/4s/5/5s	320像素
iPhone6/6s	375像素
iPhone6 Plus/6s Plus	414像素
Android（大部分）	360像素

然而，Layout Viewport的宽度都是大于浏览器可视区域的宽度，因此我们为了不让用户去缩放页面就能正常查看网站内容及确保页面中不会出现横向滚动条，需要将Layout Viewport的宽度设置为Ideal Viewport的宽度，如下图所示。