

移动 Web 应用的前端工程化实现

蔡 兵,王啸楠

(安徽信息工程学院计算机与软件工程学院,安徽 芜湖 241000)

摘 要:随着 Web 前端应用功能日趋复杂,以移动 Web 为代表的前端项目参考软件工程的思想,提出模块化、组件化的要求,规范项目的开发、构建、测试等环节,并结合流行的技术框架设计出前端工程化的实现方案.移动 Web 项目按照前端工程化、页面布局及功能规范开展,能够提升研发效率,保障软件产品质量.

关键词:移动 Web; 前端工程化; Web 前端开发; 模块化

中图分类号:TP311.5 **文献标识码:**A **文章编号:**1008-4681(2020)05-0048-04

随着 Web 前端技术的发展,越来越多的移动应用倾向于使用 HTML、CSS 和 JavaScript 等技术开发.相对于原生 Android、iOS 应用,移动 Web 应用的技术栈通用程度高、易于学习,可以跨平台,有着更好的兼容性和便于推广等特点,已经成为中小型团队开发首选,得到了开发者的普遍青睐.

2014 年 9 月,HTML5 的发布为 Web 前端领域带来了新生^[1],该标准提供了移动应用所需的特性,如手势拖拽、地理定位、音视频、本地存储等,使得移动 Web 应用替代原生应用成为可能.2015 年 6 月,ECMA Script 6.0 发布,让 JavaScript 语言具备了企业级应用的开发能力,可以编写复杂的大型应用程序^[2].由于移动 Web 应用广阔的应用场景和生命力,前端框架和技术规范层出不穷,如 Web pack、Grunt、Gulp、AMD、CMD 等,它们为前端项目的开发提供了更多选择,但也面临着如何快速稳定地构建移动 Web 应用的问题.

1 前端工程化的必要性

前端项目主要包括 SPA 项目、HTML 模板项目、大前端项目三种形式^[3].当前 Web 开发中采取的主要方式是前后端分离,前端项目作为独立结构,会经历开发、测试、部署等阶段.根据用户的习惯,比如商城、资讯、社交类等应用,移动 Web 端的使用频次已经远超 PC 端.移动端的项目要求更高,需要界面交互友好、功能丰富、体验流畅才能吸引用户.另

外,在项目开发中会面临需求经常变化、时间要求紧急的情况,随着第三方插件引入和团队多人协作的进行,项目越来越复杂,难以维护.为了有效应对这些问题、保证项目有序展开,前端工程化以提升开发效率、提高产品质量、增强项目的可维护性为目标,规范了项目的开发方式,在行业内逐渐形成了一套行之有效的技术体系.

2 前端工程化的特点

2.1 模块化

模块化思想是将大型项目按照功能拆分成独立的模块^[4].每个模块作为功能较为单一的部分,开发难度明显降低;拆分模块具有高内聚、低耦合的特点,能够在项目中实现复用.模块整合和管理通过版本控制工具(如 Git)完成,因此可以让多个研发人员并行开发,缩短项目研发时间,提升开发效率和软件质量.在后期的产品迭代中,由于模块相对独立,便于快速定位问题,找出缺陷代码,做到风险可控.

2.2 组件化

组件化与模块化的差异主要体现在颗粒度上.组件相对于模块的划分更窄一些,它是一种特定的对象集合,并不能直接实现某项功能.

在 Web 前端项目中,组件化将页面视为一个容器.通常页面的组成包括 Header 头部、导航栏、Banner 栏、内容区域、侧边栏、Footer 底部等,这些部分独立完成后拼接成完整的页面,如图 1 所示.

收稿日期:2020-07-15

作者简介:蔡兵,男,湖南岳阳人,安徽信息工程学院计算机与软件工程学院中级软件工程师.

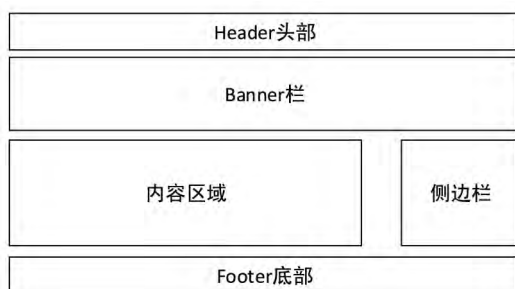


图 1 页面的组件化构成

由于组件独立性的特点,它可以是一个单独的文件.对于内容较为复杂的组件,还可以对应一个工程目录,为方便维护,其所需的资源都放在这个目录中.在使用上,组件可以自由组合,如图 1 中内容区域部分,实际开发中会由多条内容子项(Item)组合而成.当页面需要修改或者维护时,组件结构能够极大地提升操作效率,直接删除或替换对应文件即可完成.

2.3 规范化

项目的运行效率和稳定可维护需要规范约束,相关要求包括结构配置、编码和部署三个方面.

(1) 结构配置规范

正式的前端项目中除了自身编写的代码外,需要引入许多资源和第三方库文件.为了有效管理和组织各种代码、资源、文件,项目要制定清晰的目录结构;同时依赖的插件通过包管理其管理,并使用配置文件记录,便于使用与移除.为了使目录结构与配置的规范顺利执行,项目研发中可以借助与开发环境配套的工具执行.

(2) 编码规范

编码规范主要体现在命名规范和内容格式规范,要求 HTML、CSS、JavaScript 等 Web 前端语言合理组织、使用,通常各项目团队会存在细微的差异,此处列举约定俗成的编码规范要求:

第一,文件(夹)的命名均不允许包含空格.

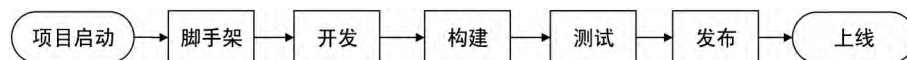


图 2 前端工程化简化流程

3.1 脚手架工具

脚手架是一种项目搭建工具,可以集成配置项目所需的 ES6、Stylus 等开发环境,用来减少重复性工作.在项目研发中,开发人员主要关注的是业务逻辑,而对插件或工具的配置使用可以通过“傻瓜式”

第二,所有文件,包括 .html、.css、.js 文件,全部使用无 BOM 头的 UTF-8 编码.

第三,项目中所有源码使用同样的缩进,使用 4 或者 2 个 spaces 视项目成员而定,但是必须都保持一致,推荐使用 2 个 spaces.

ESLint 是 JavaScript 代码中识别和报告模式匹配的工具,应用在项目中可以保证编码规范的严格执行.安装 ESLint 后使用配置文件设定规则的等级,修改.eslintignore 文件过滤不需要规范验证的文件,设置较为灵活.

(3) 部署规范

在 Web 前端项目部署前,需要对源代码进行一系列处理,使得发布的代码体积和数量减少,提高页面的加载速度和安全性,具体的操作如下.

精简:删掉代码注释、空白内容,缩短语句,从而减少代码体积,让页面的加载速度加快,在一定程度上提升用户体验.

混淆:针对逻辑操作代码(如 JavaScript),将变量和函数名替换无意义的标识(如 a、b),降低代码的可读性,防止通过代码逻辑窥视系统结构.

压缩:base64 编码是常用代码压缩方式,可以减少 50%的代码体积.服务器的 gzip 也能对代码有效压缩.

编译:在前端项目的编译包括 CSS 预处理器(如 Stylus、Less、Sass)、ES6 及以上版本,许多低版本的浏览器无法直接运行,需要通过特定的工具编译成标准的 CSS、JavaScript 语法.

3 Web 应用的工程化实现

根据前端工程化的要求进行 Web 应用的开发,其工作流程从脚手架工具开始到上线发布运行,需要经历开发、构建、测试等环节^[4].前端工程化的简化流程如图 2 所示.

的脚手架完成.

脚手架通常是基于 Node.js 环境,需要依赖 npm 工具进行包管理,其中典型的是 Vue CLI(Vue.js 框架的命令行工具).使用 Vue CLI 创建项目时,开发人员根据命令行的提示选择模板、启动 ESLint 规则

验证、使用本地服务器等操作,从而避免了 webpack 复杂配置,降低学习成本,极大地提升了开发效率。

3.2 项目开发构建

在编写源代码的过程中,为了验证功能效果是否达标,需要通过浏览器调试检验。引入工程化方式编写前端代码,其代码源文件(如 ES6、Stylus 等)常常无法直接在浏览器运行,需要编译(构建)成符合运行标准的 JavaScript、CSS 形式。如果每修改一次代码,都要手动编译执行,显然非常影响工作效率。因此,在实际的项目中,会使用本地服务器与构建系统结合,监听源代码的修改并动态构建,同时控制浏览器刷新加载,从而帮助开发人员流畅地调试。

在 Node.js 环境下,标准的本地服务器使用 webpack-dev-server 搭建,它提供了动态构建(编译)、模块热替换(HMR)等功能。webpack-dev-server 编译后的文件不会写入磁盘,它能够像真实文件一样挂载到服务器的根路径,从而加载到内存中,加快调试效率。在 webpack 的配置文件可以定义本地服务器加载的目录、端口号、编译完成后自动打开浏览器等,以下是典型的 devServer 的配置:

```
devServer: {
  contentBase: './dist',
  port: 9000,
  open: false
}
```

3.3 项目测试

测试是软件研发必不可少的一环,它能够运用各种技术手段,充分覆盖用户使用的场景,尽可能多地找出软件缺陷,从而保证项目上线后稳定运行。Web 前端的测试方法种类繁多,包括 UI 测试、功能测试、兼容性测试等等,这里仅探究与开发人员结合较为紧密的单元测试。

单元测试(Unit Testing)是指通过编写程序测试源代码的最小可测试单元,一次检查一个独立、明确的功能是否正常。当前广泛使用的前端单元测试框架是 Jest,它包含断言、Mock 工具集、生成覆盖率报告等测试工具,可支持 Vue、React 等框架的前端项目测试。通过 npm 命令可以安装 Jest 到项目中:

```
npm install --save-dev jest
```

然后在 package.json 配置符合 Jest 支持的文件

格式,即可编写测试代码进行单元测试:

```
"jest": {
  "testRegex": "/test/. *.test.jsx? $ "
}
```

3.4 部署上线

项目开发完成后需要将编译可运行的代码部署到指定的位置,会根据实际情况选择不同的策略。对于单人负责的小规模项目,在指定服务器上搭建项目运行环境后,使用 FTP 文件传输的方式,直接将代码复制到服务器对应文件目录。多人协作的项目中,开发人员使用代码管理工具(如 Git)在 dev 分支上编码,功能完成后合并代码,并由专人进行代码审查及部署。在功能复杂或安全性要求高的项目中,还需要使用权限控制的方式,对开发人员和审查人员分工,避免开发人员随意修改服务器上的代码导致的环境崩溃问题。

4 移动 Web 应用的技术探究

移动 Web 应用除了遵循上述工程化规范和要求外,还要根据用户习惯进行布局 and 功能的适配操作,保证在移动端有较好的体验。

4.1 页面布局的实现

由于手机屏幕的分辨率不同,前端代码需要采用响应式布局方式,让各种屏幕下显示效果与设计稿一致。通常在移动端页面中使用 Viewport 属性,使其不可缩放,页面的宽度与浏览器宽度相同、比例保持固定。设置代码如下:

```
<meta name="viewport" content="width=device-width, initial-scale=1.0, maximum-scale=1.0, user-scalable=no">
```

页面元素单位推荐使用 rem,替代原始的 px。rem 是一种相对长度单位,参考基准是 font-size,能根据屏幕大小成比例地调整成实际的像素,让内容元素在不同的设备上效果一致。rem 与 px 的换算公式为:

$$1 \text{ rem} = \text{屏幕宽度} / \text{设计图宽度} \times 100 \text{ px}$$

为了提升开发效率,项目中常常会选用成熟 UI 框架进行移动端的布局,如 ElementUI、Ant Design 等。它们能提供统一且美观的样式,包括按钮、图标、字体等,以及功能丰富的组件,如上拉刷新、下拉加

载、对话框等。

4.2 常用功能实现

移动端常用功能包括地理定位、拍照、多媒体等基础功能,以及内容分享、支付等高级功能。基础功能的实现可以使用 HTML5 技术,如 Geolocation API 的 `getCurrentPosition`(用于获取用户的位置信息,得到经纬度值),还能用于百度地图等地图软件可视化展示。高级功能需要第三方库或接口的支持,如在实际应用中使用微信支付,需要调用对应的接口,按照其业务流程操作。

5 结束语

本文给出了 Web 前端工程化的规范特点和实

现方式,结合当前流行的移动 Web 应用进行技术探究。相比传统的前端开发方式,按照工程化的要求开发移动 Web 应用,能够明显提高项目质量,让页面加载速度更快,应用更稳定、安全,在技术层面上保证项目有序进行。

参考文献:

- [1] 盛婷婷.基于 HTML5 技术的移动 Web 前端设计与开发分析[J].电子制作.2016(19):46-47.
- [2] 阮一峰.ES6 标准入门[M].第 3 版.北京:电子工业出版社,2017.
- [3] 周俊鹏.前端工程化:体系设计与实践[M].北京:电子工业出版社,2018.
- [4] 周伟,郑世珏.Web 前端工程化解决方案研究[J].信息技术.2018(8):44-47.

Front-end Engineering Implementation of Mobile Web Applications

CAI Bing, WANG Xiaonan

(Department of Computer Science and Software Engineering, Anhui Institute of Information Technology,
Wuhu Anhui 241000, China)

Abstract: As web front-end application functions become more and more complex, front-end projects represented by mobile web refer to the ideas of software engineering, propose modular and componentized requirements, standardize project development, construction, testing and other links, and combined with popular technology framework to design the front-end engineering implementation plan. Mobile web projects are carried out in accordance with front-end engineering, page layout and functional specifications, which can improve R & D efficiency and ensure the quality of software products.

Key Words: mobile Web; front-end engineering; front-end web development; modular

(责任编辑:王平)