|  |  |
| --- | --- |
| 文档总页数 | 11 |
| 文档编号 | 移动轻应用H5前端开发规范文档V1.0 |

**移动轻应用**

**H5前端开发规范**

**（1.0）**

**乐视网信息技术（北京）科技有限公司**

**2016-12-17**

关于本文档

|  |  |  |  |
| --- | --- | --- | --- |
| 修改历史 | | | |
| **日 期** | **版 本** | **作 者** | **修改内容** |
| 2016-12-17 | V1.0 | 汪瑶 | 创建该文档，移动轻应用H5前端开发规范初稿 |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

**目录**

[1 概况 4](#_Toc469923928)

[1.1 背景 4](#_Toc469923929)

[1.2 宗旨、适用范围 5](#_Toc469923930)

[1.3 本规范的内容 5](#_Toc469923931)

[1.4 基本概念 5](#_Toc469923932)

[1.5 内容大纲 5](#_Toc469923933)

[2 协作流程 6](#_Toc469923934)

[2.1 确定业务需求 6](#_Toc469923935)

[2.2 确定UI交互 6](#_Toc469923936)

[2.3 确定前端H5技术架构及接口服务 7](#_Toc469923937)

[2.4 确定UE设计 7](#_Toc469923938)

[2.5 研发 7](#_Toc469923939)

[2.6 测试-UAT-上线 7](#_Toc469923940)

[3 技术原则 7](#_Toc469923941)

[3.1 模块划分 7](#_Toc469923942)

[3.2 层次划分 8](#_Toc469923943)

[3.3 代码规范 8](#_Toc469923944)

[3.4 项目目录结构 9](#_Toc469923945)

[4 移动轻应用H5前端开发验收标准 10](#_Toc469923946)

# 概况

## 背景

业务方与项目实施方沟通，业务方并不能清晰描述所有需求及系统要求。实施方未能引导业务方，挖掘业务方的真实需求去研发系统，往往的结果会是提供的系统并不是业务方想要的系统。

项目实施方内部不同职能岗位间沟通信息产生误差，导致与业务沟通时候什么能做什么不能做，以及怎么去给业务方去做不明确。在项目快节奏的压力下，虽然不够明确，但研发人迫于无赖投入研发，导致后期产生大量调整的工作量，最终导致系统未能很好的支持业务。——协作流程问题

产品在新的阶段提出了新的需求，然而上一阶段的项目，并不能或很难去改造满足产品后续的需求。——基础技术架构问题

项目成员发生变动后，无法或者需要很长时间才能熟悉该项目。——代码风格、规范问题

## 宗旨、适用范围

旨在明确开发需求、提高前端项目的质量、前端项目的持续开发、开发人员间的合作及使其基础架构可支撑产品的持续进化。为提高团队沟通、协作、研发的效率，确保项目结构清晰、扩展性良好，保障项目的可读性、可维护性，特制订此文档。

本规范文档适用于移动轻应用H5前端开发。本规范文档一经确认，前端开发人员必须按本文档规范进行前端H5开发，本文档如有不对或者不合适的地方请及时提出，经讨论决定后方可更改。

## 本规范的内容

本文档制定了移动轻应用H5前端项目研发中的协作流程(Cooperation Procedure)、基础技术架构(H5 Tech Architecture)、及HTML/CSS、JAVASCRIPT、ANGULARJS的编写格式和风格规则。

如下：

1. 制定不同职能岗位人员之间协作完成移动轻应用的流程——Cooperation Procedure
2. 制定移动轻应用项目应满足的基础技术架构——H5 Tech Architecture
3. 制定HTML/CSS编码规范——HTML/CSS Guide
4. 制定JAVASCRIPT编码规范——JAVASCRIPT Guide
5. 制定ANGULARJS代码风格——ANGULARJS Guide

Cooperation Procedure作用：提高沟通、协作、研发的效率。

H5 Tech Architecture作用：确保项目结构清晰、扩展性良好。

HTML/CSS Guide、JAVASCRIPT Guide作用：确保代码风格的一致性，保障项目的可读性、可维护性

ANGULARJS Guide作用：用于采用ANGULARJS开发的前端H5项目

## 基本概念

前端开发：UE/UI设计师和后台程序协作，根据设计图稿完成页面(HTML+CSS+JS)代码编写制作后通过后台程序返回的数据接口进行数据展现；维护及优化软件产品前端性能，提高用户体验；保证浏览器和终端兼容性，提供浏览器和移动终端主流版本的适应能力，包括chrome、Firefox、safari、opera、IE9-Edge、IOS和Android等。

## 内容大纲

|  |  |
| --- | --- |
| 协作流程(Cooperation Procedure) | 1. 确定业务需求 2. 确定UI交互 3. 确定前端H5技术架构及接口服务 4. 确定UE设计 5. 研发 6. 测试-UAT-上线 |
| 技术原则(Tech Rule) | 架构：   1. 分模块 2. 分层次 3. 代码规范 4. 项目目录结构 |
| HTML/CSS规范 | 见附件1 |
| JAVASCRIPT规范 | 见附件2 |
| ANGULARJS规范 | 见附件3 |
| 验收标准 | 1. 必须是前端H5轻应用项目——single page application 2. 必须满足代码的基本质量要求 3. 尽可能符合HTML/CSS规范、JAVASCRIPT规范 |

# 协作流程

## 确定业务需求

与业务方沟通需求：

1. 明确使用者角色(如：系统管理员、普通用户、无账号用户)
2. 各用户拥有的行为(如:管理员有：发文、收文行为，普通用户有收文行为)
3. 数据(长度范围、是否必须内容、数据之间的关联关系)

以上几步可帮助分析需求，编写需求说明书

## 确定UI交互

1. 根据需求，设计完整的UI交互
2. 与业务方核对是否能够满足业务方的所有需求

确定后编写UI交互说明书

## 确定前端H5技术架构及接口服务

1. 分析UI交付和前端操作功能
2. 搭建前端架构
3. 模拟数据接口服务
4. 向后端业务系统提出明确的接口需求

编写前端架构说明书、数据接口需求说明书

## 确定UE设计

1. 与业务确定UE，动效，交互行为
2. 切图

UE高保真设计资料

## 研发

1. 根据UE抽象功能组件和行为
2. 编写前端界面交互与接口服务

项目工程代码

## 测试-UAT-上线

1. 测试功能BUG
2. 测试交互BUG

编写测试报告

# 技术原则

## 模块划分

当前移动轻应用功能复杂，前端整合了大量业务，前端H5项目功能越来越复杂项目也越来越大，项目必须进行模块划分。良好模块划分让项目结构清晰，是团队人员合作开发的基础。不同模块也可多人员参与合作开发，也能更好的定位问题提高研发效率和代码的质量。

## 层次划分

对于数据、结构与表现相分离，最早是在软件开发架构理论中提出来的。UI设计师设计出页面模版，程序员负责内容数据的嵌入，数据可能是从数据库中提取出来，也可能是静态写入的提示性文字，最后形成一个新的页面展示给软件操作者。模版文件的结构利用HTML＋XHTML标签来定义，而所有涉及表现的东西通通剥离出来，把它放到一个单独的文件里，这个单独的文件就是CSS。

数据、结构与表现相分离的好处是：

研发员不需要过多的思考页面显示问题，而只需要根据模版效果把数据放入模版相应的位置。界面的结构和表现由UI设计师负责。填入结构的数据自觉套用设计好了的表现效果。最后呈现一个实现功能的完整界面。

表现层的分离保持整个软件界面视觉的一致性，使改版也变得非常简单，修改样式表就可以了；

由于结构清晰，数据层相对独立，对数据的集成、更新、处理和再利用也更加方便灵活；

最简单的前端分层大致也有：视图（View）——界面显示部分、控制器（Controller）——控制页面逻辑和界面跳转流程、数据模型（Model）——提供数据，3层。MVC是最基本的要起，与之类似的前端设计模式有，MVP、MVVM

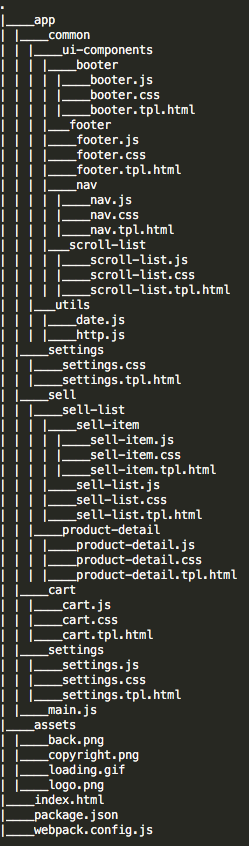
项目具体是采用哪种模式得根据情况而定，但是不管采用哪种前端分层，均可让代码层次分明，结构清晰，易维护。

## 代码规范

对于前端H5项目，使用到了html/css、javascript，所以通常前端的规范都是指：HTML/CSS规范、JAVASCRIPT规范。并有一些开源框架本身支持MVVM、MVP，使用这种框架就应该遵从开源框架的规范。我们以goolge 的 angularjs开源框架为例，在基于angularjs框架的项目应该参考：ANGULARJS规范

附件提供了HTML/CSS规范、JAVASCRIPT规范、ANGULARJS规范，来自Google的规范文档。

## 项目目录结构



该结构只是一个列子，说明目录接口视例：

左侧项目目录简单描述：

./:项目的根目录

app:放整个程序的js代码、html模版、模版使用的css样式

package.json:node项目管理配置

webpack.config.js:webpack配置管理

assets:放整个项目的静态资源文件

index.html:程序单页面入口

app/main.js:主程序入口

app/common:程序公共组件和功能

项目可根据自身条件和规模适度规划目录结构，使结构清晰，层次分明。易于开发、合作、功能维护及结构调整。

# 移动轻应用H5前端开发验收标准

1. 该验收标准仅使用于前端H5项目
2. 明确什么是一个前端H5项目:

前端H5项目特点：靠HTML/CSS、JAVASCRIPT完成界面的展示操纵、使用Ajax与服务端进行数据交互。对用户体验要求较高的项目通常使用ajax、JAVASCRIPT进行界面操作较多。

JAVA Web项目的特点:同样采用Html/CSS、JAVASCRIPT完成界面的展示，但页面的跳转和数据交互使用java后端技术实现。

在PC端仍然有很多项目采用JAVA Web、PHP等后端服务方式开发，JAVA Web开发具有成本低、开发相对简单、基本后台人员都会等优势。前端H5技术可完全替代JAVA Web界面的开发工作，但具体用JAVA Web还是H5前端开发，视项目成本、规模和对用户体验定。

Html JAVASCRIPT和单页面的理论、技术、框架发展讯息，各种技术的发展使移动H5前端开发的项目在移动设备上拥有良好的体验。

1. 简单的判断项目基本质量方法

1. 页面布局、Javascript特效和json数据交互对Google、Safari、Firefox、Android和IOS等主流浏览器的兼容。产品测试必须测试的浏览器为： Google、Safari、Firefox、Android和IOS。

判断方法：浏览器兼容性测试

2. HTML代码结构是否清晰。

判断方法：查看源文件，是否符合页面标准。各个模块是否清晰可辨或是有注释说明。

3. HTML代码结构是否复杂。

判断方法：每个模块 (x)html代码的嵌套层次简单，代码简洁。

4. HTML代码是否和CSS混杂在一起。

判断方法：是否存在大量在标签中使用Style定义的样式。

5. HTML代码是否和大量Javascript混杂在一起。

判断方法：Javascript没有集中的放在<head>中或是放在页面的底部，而是混杂在模板中。

6. HTML代码是否大量出现不被推荐使用的标签。

判断方法：不被推荐使用的标签见第二部分的列表，如<font>、<b>等。

7. HTML代码是否书写规范，是否结构化布局。

判断方法：依照结构化代码规范和具体的语义性检查。

8. CSS书写是否规范。

判断方法：查看CSS代码部分CSS命名是否标准，单个定义是不是过长，是否出现过多重复的定义。

9. HTML代码的格式是否混乱。

判断方法：格式是否整齐、干净。

10. JS代码是否规范

判断方法：格式是否整齐、注释是否准确、各模块间代码是否清晰。

11. 页面是否存在性能问题。

判断方法：观察CPU的占用情况，和页面的打开速度。页面的下载速度明显感觉很慢的不行。