

# 如何打造一个“逼格”的web前端项目

## 前端准备篇

前端代码规范：制定前端开发代码规范文档。

PS：重中之重，为了下一步实现前端工程化。

编码风格有很多。团队代码规范一定要统一。便于维护web项目。

接口文档规范：制定RESTful架构接口规范文档。

PS：果断使用apiDoc构建API文档。美观好看，易于维护。

还用Word或者Excel写文档，太low了。

前后端分离：简单理解为前端代码不影响后端代码，后端代码不影响前端代码。

PS：个人感觉就是有点像C/S架构，web前端 == apk || ipa。

前端工程化（模块化）：简单理解为前端代码不再零散，有组织，有规范的管理起来。

PS：个人感觉就是把后端mvc模式中v与c层都交了前端处理。

前端渐进式（PWA）：依靠缓存，Web Workers等现代技术打造的渐进式web应用。

PS：有点像hybrid app，但无需原生代码，可离线，可推送web应用。

## 前端代码篇

html：当然采用最新的html5标签。

PS：值得一提的是，不要滥用html5的标签，比如section与div标签。

css：使用CSS Modules，让css局部模块化，保证css全局唯一性。

PS：建议配合webpack一起使用。

使用sass预处理，让css模块化，便于维护与管理css。

PS：我确实有点抵触css预处理，觉得没什么用，后来使用了css预处理自后，感觉真的很方便，

我选用sass是因为很多css框架都有集成sass，便于使用而已。

使用PostCSS后处理，让css标准化，写出高质量的css。

PS：后处理器基本是把sublime text里面的前端插件都搬过来了，完全实现前端自动化。

js：使用html5新api。

PS：html5 api更容易操作dom，实现一些新功能，比如拖拽。

使用ES6/ES7/ES8的语法。

PS：js终于变成真正的javascript了，语法与java相似度更高了。

使用jquery库。

PS：jquery永远不会过时，因为jquery有强大的插件。

使用js mvvm框架Angular.js或vue.js

PS：mvvm框架让web前端更易于提高web性能。当然啦，具体问题具体分析，慎用mvvm框架。

据个人感觉，vue.js比较适合移动端web项目，体积小，性能高。

Angular.js比较适合PC端大型项目，功能强大。

## 前端构建工具篇

nodejs：一个从Chrome浏览器提取出来的强大的js解析器（js运行环境）。

PS：与java的JRE类似，一个高性能的运行环境。

webpack：一个出色的前端打包构建工具。

PS：webpack真的是个神奇的打包工具，实现前端模块化的神器。

补充：现在一个前端的开发主流工具就使用脚手架（命令行工具），比如：vue的vue-cli。  
能迅速构架web应用项目，搭建开发环境与发布项目。