|  |
| --- |
| 密级：机密 |
| 总页数：91页 |



**IWT前端框架概要设计**

**概要设计**

**(V 0.1)**

**英泰伟业信息技术股份有限公司**

**应用业务开发部**

**2016年11月**

**版本信息**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **日期** | **版本** | **描述** | **作者/修改人** | **备注** |
| 2016-11-7 | 0.1 | 初稿 | 苏晓燕 |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

# IWT前端框架概要设计说明书

# 1引言

## 1.1编写目的

随着前端技术的不断发展，可以说每天前端技术都在变化着。从最初石器时代的简单的表单提交、再到铁器时代Ajax的出现以及简单的框架流行，随着时间的推移迎来了农业时代模块加载（AMD/CMD）,再到工业化时代MV\*的到来,可见前端的发展的速度有多么的快，编写可重用，加载速度快,与前后端分离的框架技术已迫在眉睫。

编写IWT前端框架的本质是一种管理，对代码的管理。此框架的目的是可以快速的配合开发人员完成开发。达到：加载快，渲染快，执行快，开发快的目的。

## 1.2背景

待开发的前端框架名称：IWT前端开发框架v1.0

项目的任务提出者：田大昕、刘钊

项目开发者：田大昕、刘钊 苏晓燕

用户：IWT的项目工作人员

## 1.3定义

**1、**JavaScript：JavaScript一种直译式脚本语言，是一种动态类型、弱类型、基于原型的语言，内置支持类型。它的解释器被称为JavaScript引擎，为浏览器的一部分，广泛用于客户端的脚本语言，最早是在HTML（标准通用标记语言下的一个应用）网页上使用，用来给HTML网页增加动态功能。

2、HTML:HTML（Hyper Text Mark-up Language ）即超文本标记语言，是 WWW 的描述语言，由 Tim Berners-lee提出。

3、BootStrap: Bootstrap,是目前最受欢迎的前端框架。

4、Node.js：Node.js是一个Javascript运行环境(runtime)。

5、Npm: nodejs的包管理和分发工具。

6、Cooking: Cooking是用于前端项目构建的构建工具。

7、Vue.js：Vue.js（读音 /vjuː/, 类似于 view）是一个构建数据驱动的 web 界面的渐进式框架。

## 1.4参考资料

* 【1】徐涛 译 人民邮电出版社 JavaScript设计模式 第一版
* 【2】司徒正美 编 人民邮电出版社 JavaScript框架设计
* 【3】amaze UI:<http://amazeui.org/getting-started>
* 【4】ZUI: <http://easysoft.github.io/>
* 【5】MUI: <http://www.dcloud.io/mui.html>
* 【6】重置样式表：<http://necolas.github.io/normalize.css/>
* 【7】后台UI：<http://element.eleme.io/>
* 【8】Vue mui: <http://mui.yaobieting.com/docs/index.html>
* 【9】Cooking: <http://cookingjs.github.io/zh-cn/intro.html>

# 2总体设计

## 2.1技术体系

1、HTML重构：

定义：将设计稿转换成WEB页面的过程称为重构。

做法：栅格系统+css3.0规范进行网站HTML代码的重构。

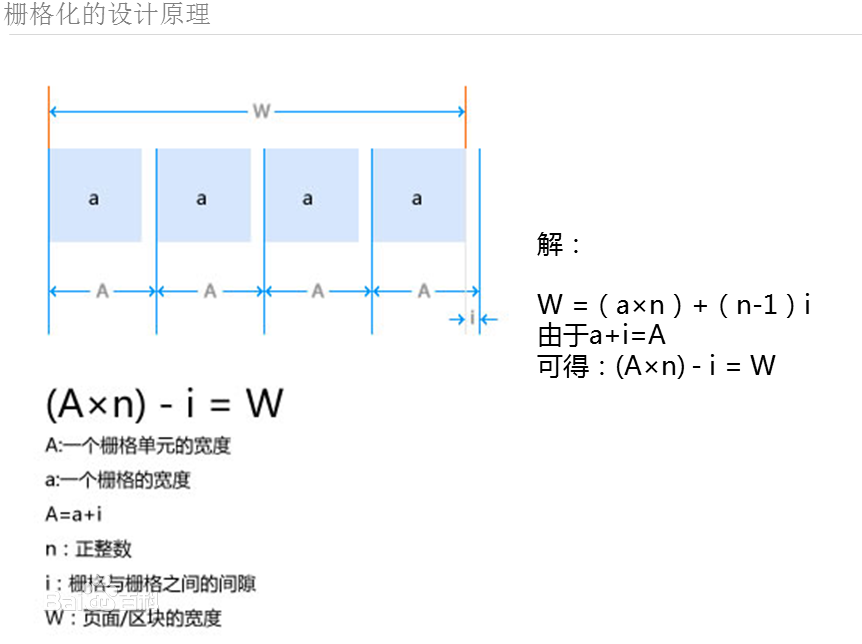
1. 理解的定义：

栅格系统的页面重构：栅格系统英文为“grid systems”，也有人翻

为“网格系统”，运用固定的格子设计版面布局，其风格工整简洁。

1. 计算方法：在网页设计中，我们把宽度为“W”的页面分割成n个网格单元“a”，每个单元与单元之间的间隙设为“i”,此时我们把“a+i”定义“A”。他们之间的关系是：W =（a×n）+（n-1）i

可得到：**(A×n) – i = W**



2、编写适合公司发展的js组件：

1、我们以Vue.js作为核心的基础库，进行二次开发。它是一个小型的MVVM的构建数据驱动的web界面的库。

2、我们构建这个项目用到的构建工具是：cooking，它可以快速搭建项目雏形。

3、项目的运行平台

本项目需要借助：Node.js：一个Javascript运行环境(runtime)。

Npm: nodejs的包管理和分发工具。我们可以快速安装一些需要的工具。

## 2.2运行环境

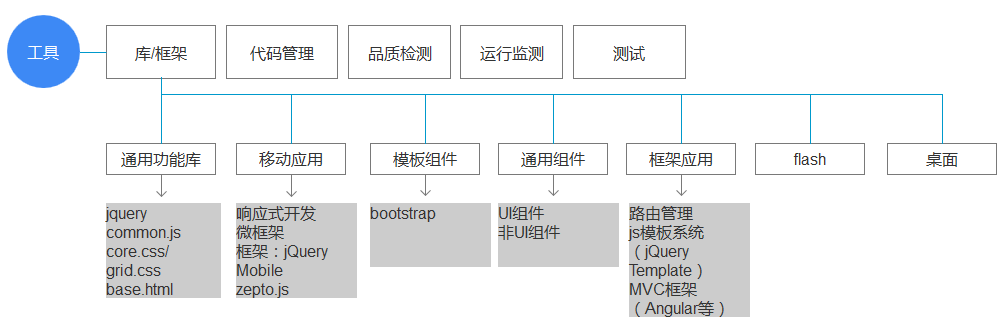
1、硬件运行环境：

1. Intel®Core™ i5-4460 CUP @3.20GHz 3.20GHz 内存8.00GB 64位操作系统

2、软件运行环境：可运行在IE9+以上的浏览器上。

## 2.3基本设计概念

开发IWT前端框架，我们需要了解现有项目的需求，对现有项目的需求进行分析，经过PS的分析，并且进行前端成熟框架的技术选型。最终确定1-2个比较合适IWT的前端框架作为核心框架。



1. **商品演示类web开发技术选型**

可以考虑：jQuery+Bootstrap+RequireJS的组合方式。RequireJS的作用不单单是提供了一个按需加载机制，它还能让我们组织起更为庞大的代码。

1. **后台系统技术选型**

可以考虑一些比较好的EXT、EasyUI、YUI等这些重大的UI库方案。模块组织、打包、数据可视化，它都已经可以完全胜任。

1. **移动端开发和重交互得web网站选型**

MVVM最擅长做这些重交互的产品。Vue.js是最近几年比较流行的MVVM的移动端框架。

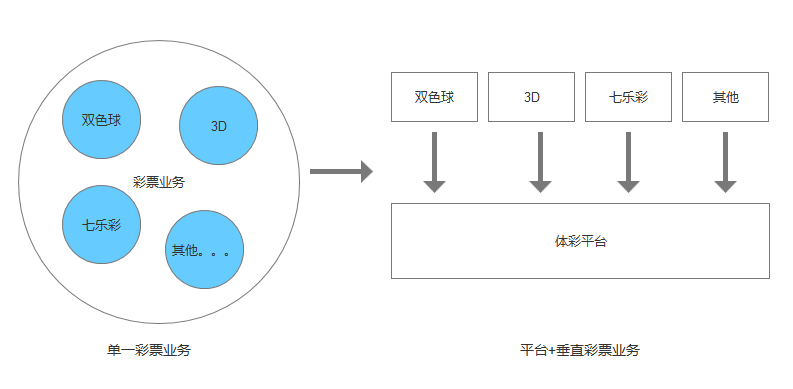
综上所述，我们的需求是开发后台管理和移动端app，交互操作比较多。因此我们选择MVVM重交互的产品，Vue.js就是这样的核心框架。他的优势是：

1、Vue.js 是尤玉溪老师写的一个用于创建Web交互界面的库，是一个精简MVVM框架。它主要目的是分离视图（view）和（模型）。

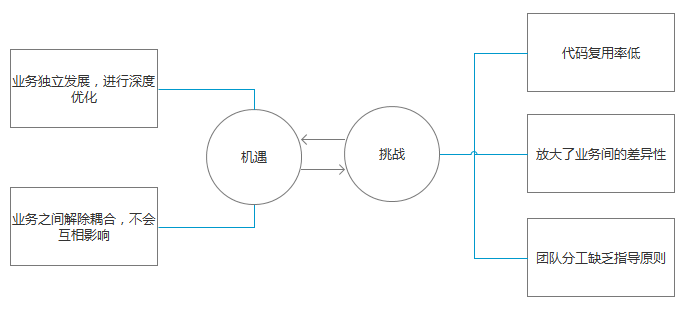
2、目前Vue.js有开源的UI组件库。不需要我们重复造轮子，我们可以在此基础上进行适合IWT的开发。节约了开发成本、节省了开发时间，提高开发效率。

## PS系统管家前端业务分析

### PS系统架构的演化



### 2.4.2 PS系统面临的机遇与挑战

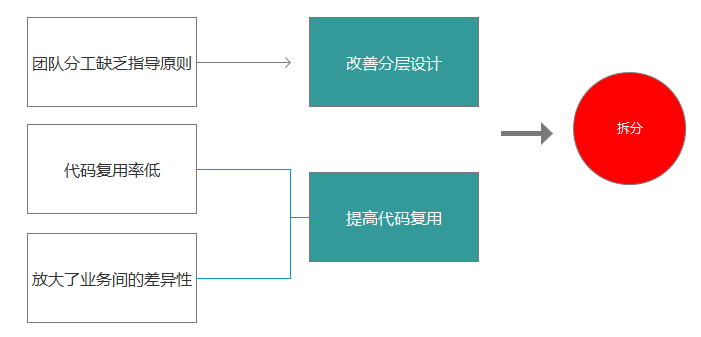


挑战：

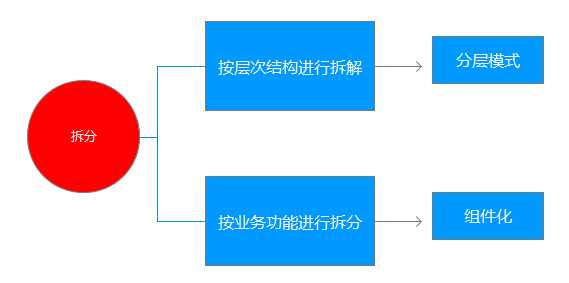
* + - 各业务之间频繁进行代码拷贝，App的包大小显著增大。
    - 从“相同”变为“相似”。缺乏对差异性的定制能力，导致很多东西要从头再来。
    - 新业务在初期，表现的相对艰难，可重用的东西偏少。
    - 不同级别的工程师分工缺乏有效的指导。

### 2.4.3 对业务模块的拆分

对系统进行有效拆分：改善分层、提高代码的复用率。因此在前端设计了前端UI库，里面写好了，UI层面上的可以用到的所有的样式代码。



拆分方案：



#### 2.4.3.1分层模式

概念了解：MVC模式和MVVM模式

1、什么是MVC模式？

MVC定义：即Model-View-Controller，把一个应用的输入、处理、输出流程按照Model、View、Controller的方式进行分离，这样一个应用被分成三个层，即模型层、视图层、控制层。

**模型（Model）：**封装的是数据源和所有基于对这些数据的操作。在一个组件中，Model往往表示组件的状态和操作状态的方法。

**视图（View）：**封装的是对数据源Model的一种显示。一个模型可以由多个视图，而一个视图理论上也可以同不同的模型关联起来。

**控制器（Control）：**封装的是外界作用于模型的操作。通常，这些操作会转发到模型上，并调用模型中相应的一个或者多个方法。一般Controller在Model和View之间起到了沟通的作用，处理用户在View上的输入，并转发给Model。这样Model和View两者之间可以做到松散耦合，甚至可以彼此不知道对方，而由Controller连接起这两个部分。

1. 什么是MVVM模式？

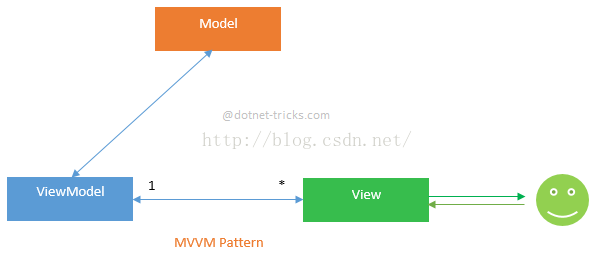
MVVM即Model-View-View Model。

这个模式提供对View和View Model的双向数据绑定。这使得View Model的状态改变可以自动传递给View。典型的情况是，View Model通过使用obsever模式（观察者模式）来将View Model的变化通知给model。

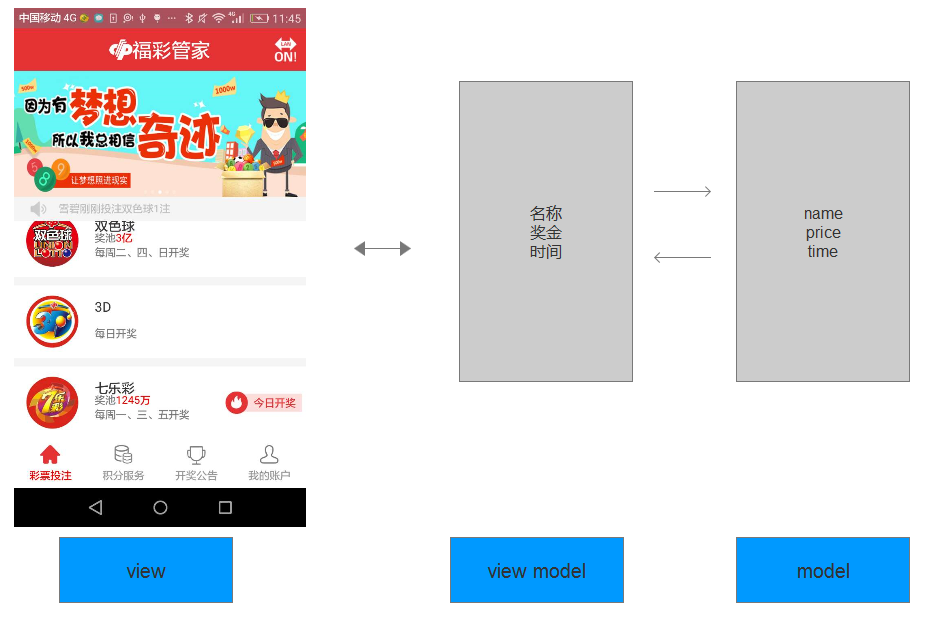
* Model层代表了描述业务逻辑和数据的一系列类的集合。它也定义了数据修改和操作的业务规则。
* View代表了UI组件，像CSS，JQuery，html等。他只负责展示从Presenter接收到的数据。也就是把模型转化成UI。

**（Presenter**是从Model中获取数据并提供给View的层**）**

* View Model负责暴露方法，命令，其他属性来操作VIew的状态，组装model作为View动作的结果，并且触发view自己的事件。



举例说明：



#### 2.4.3.2分层的优势



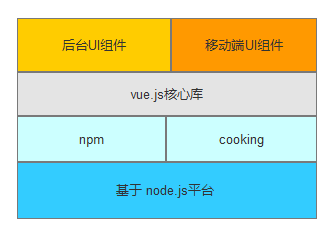
#### 2.4.4.3 组件化：按功能组件化



总结：

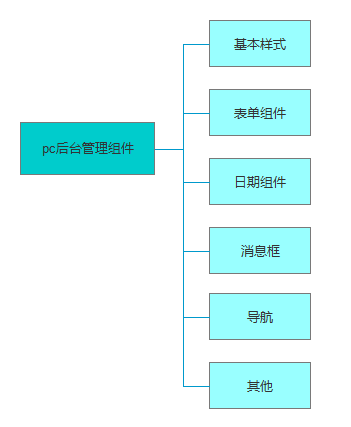
1. 拆：MVVM的分层模式；组件化的分解方式
2. 组装：将组件的View/viewMode依赖关系配置完成组装，即可实现业务的需要
3. 行动起来：对业务进行“拆分”，通过“组装”完成开发。最终实现业务组件的复用，和开发效率的提升。

# IWT前端框架结构



## PC后台管理UI组件开发

1、功能模块结构图：

****

#### 一、配置文件：新建项目，项目结构为

|- src/ --------------------- 项目源代码

|- App.vue --------------

|- main.js -------------- 入口文件

|- .babelrc ----------------- babel 配置文件

|- index.html --------------- HTML 模板

|- package.json ------------- npm 配置文件

|- README.md ---------------- 项目帮助文档

|- webpack.config.json ------ webpack 配置文件

#### 二、Basic基础

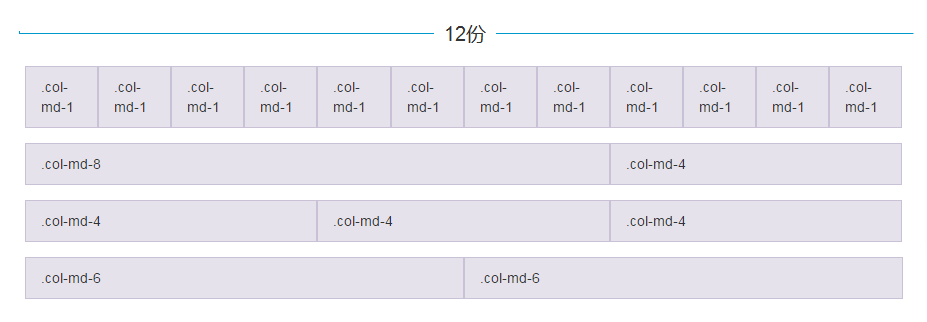
##### 1、Layout布局

栅格布局：通过基础的24分栏，迅速简便的创建布局。

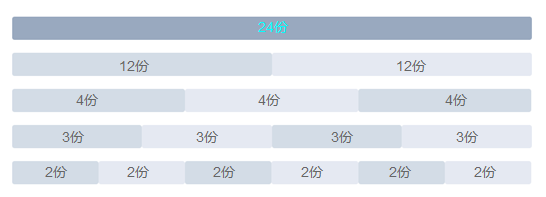
什么是栅格系统定义：

以规则的网格阵列来指导和规范网页中的版面布局以及信息分布。

最古老的栅格系统来自于报纸的布局，国外提出960网格系统就是IWT前端布局的前身。它将屏幕分成均等的12份。来进行布局。



1.1基础布局



1.2分栏间隔

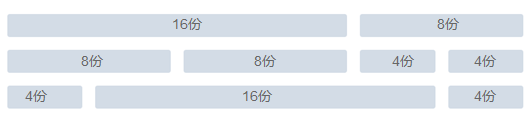
C:\Users\admin\AppData\Local\Temp\AE1.tmp.png

分栏布局的例图：



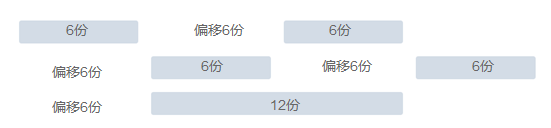
1.3混合布局

通过基础的 1/24 分栏任意扩展组合形成较为复杂的混合布局。



1.4分栏偏移

支持偏移指定的栏数。



1.6行的说明：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 参数 | 说明 | 类型 | 可选值 | 默认值 |
| Gutter | 栅格间隔 | number |  | 0 |
| type | 布局模式，可选 flex，现代浏览器下有效 | string |  |  |
| justify | flex 布局下的水平排列方式 | string | start/  end/  center/  space-around/  space-between | start |
| align | flex 布局下的垂直排列方式 | string | top/middle/bottom | top |

1.7列的说明

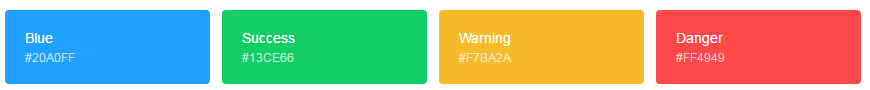
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 参数 | 说明 | 类型 | 可选值 | 默认值 |
| Span | 栅格占据的列数，**必选参数** | number |  | - |
| offset | 栅格左侧的间隔格数 | number |  | - |
| Push | 栅格向右移动格数 | number | start/  end/  center/  space-around/  space-between | - |
| Pull | 栅格向左移动格数 | number | top/middle/bottom | - |

##### 2、Color配色方案：根据公司UI设计进行一下仅供参考

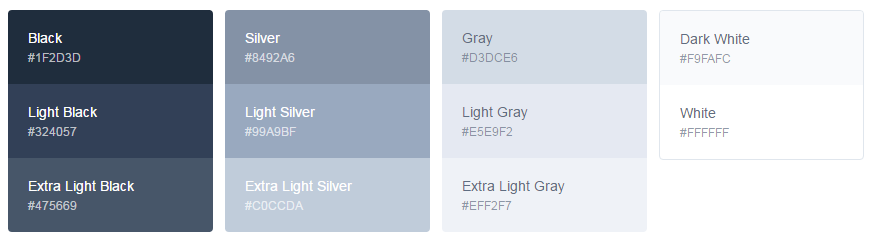
主颜色：



辅助颜色



中性颜色



##### 3、Typography 字体方案：

我们对字体进行统一规范，力求在各个操作系统下都有最佳展示效果。

3.1中文字体：

PingFang SC\ Microsoft YaHei\ Hiragino Sans GB

3.2英文字体：

Helvetica Neue\Helvetica\Arial

3.3 Font-family代码

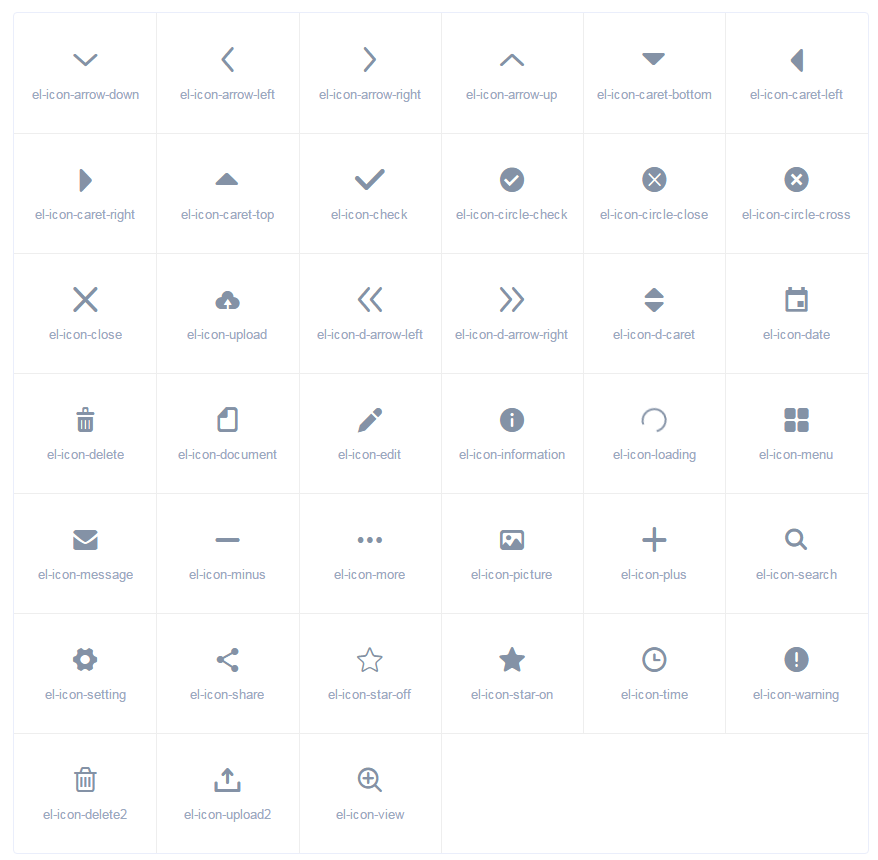
font-family: "Helvetica Neue",Helvetica,"PingFang SC","Hiragino Sans GB","Microsoft YaHei","微软雅黑",Arial,sans-serif;

3.4字体使用规范：

|  |  |  |
| --- | --- | --- |
| 主标题 | 快速搭建页面 | **20px Extra large** |
| 标题 | 快速搭建页面 | **18px Extra large** |
| 小标题 | 快速搭建页面 | **16px Extra large** |
| 正文 | 快速搭建页面 | **14px Extra large** |
| 正文（小） | 快速搭建页面 | **13px Extra large** |
| 辅助文字 | 快速搭建页面 | **12px Extra large** |

##### Icon图标的定义

C:\Users\admin\AppData\Local\Temp\ED56.tmp.png



##### 5按钮

5.1 基础用法



5.2禁用状态

C:\Users\admin\AppData\Local\Temp\6B51.tmp.png

5.3 有颜色倾向



5.4图标按钮

C:\Users\admin\AppData\Local\Temp\5D13.tmp.png

5.5 按钮组

C:\Users\admin\AppData\Local\Temp\9B6C.tmp.png

5.6 记载中



5.7不同尺寸的按钮



#### 三、Form 表单

1. Radio 单选表单

基础用法：

C:\Users\admin\AppData\Local\Temp\1A13.tmp.png

禁止状态：

C:\Users\admin\AppData\Local\Temp\3C16.tmp.png

单选组：

C:\Users\admin\AppData\Local\Temp\C6D9.tmp.png

表单样式：

C:\Users\admin\AppData\Local\Temp\346C.tmp.png

1. Checkbox 多选框

基本用法：

C:\Users\admin\AppData\Local\Temp\E381.tmp.png

禁用状态：

C:\Users\admin\AppData\Local\Temp\8129.tmp.png

多选框组：

C:\Users\admin\AppData\Local\Temp\E7D9.tmp.png

3、Input输入框

基本用法：

C:\Users\admin\AppData\Local\Temp\4372.tmp.png

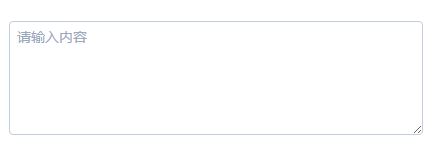
禁用状态：



带图标的输入框：

C:\Users\admin\AppData\Local\Temp\9F1C.tmp.png

文本区域：可调整大小



复合型输入框：



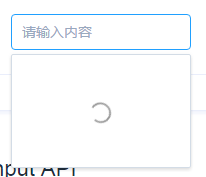
尺寸：



带输入建议：



远程搜索：



4、Input Number 计数器

基本用法：

C:\Users\admin\AppData\Local\Temp\14E6.tmp.png

禁止状态：

C:\Users\admin\AppData\Local\Temp\C0DB.tmp.png

设置步数：

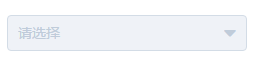


5、Select 选择器

基础用法：



禁用状态：



可清空单选：

C:\Users\admin\AppData\Local\Temp\49F2.tmp.png

基础多线：

C:\Users\admin\AppData\Local\Temp\9F8.tmp.png

自定义模板：

C:\Users\admin\AppData\Local\Temp\7DC2.tmp.png

6、Switch 开关

基本用法：

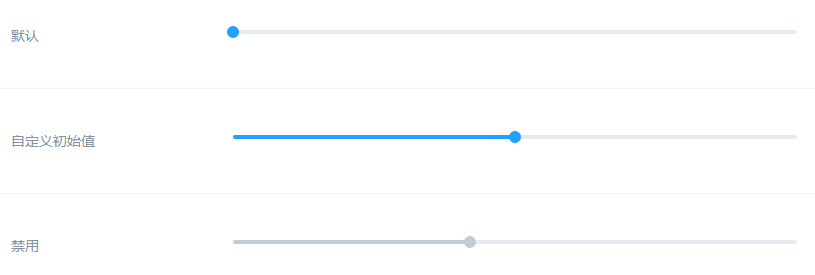
C:\Users\admin\AppData\Local\Temp\D91F.tmp.png

禁用状态：

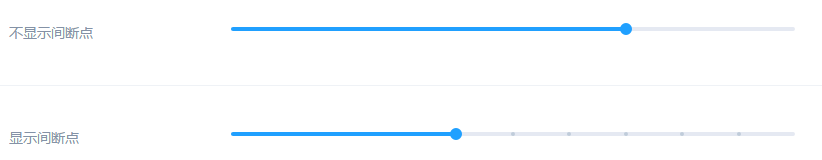
C:\Users\admin\AppData\Local\Temp\3035.tmp.png

7、Slider 滑块

基础用法：



离散值：

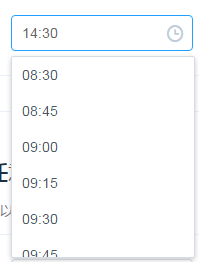


带有输入框：

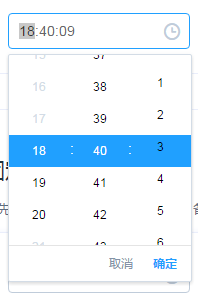


8、Time Picker 时间选择器

固定时间点：



任意时间点：



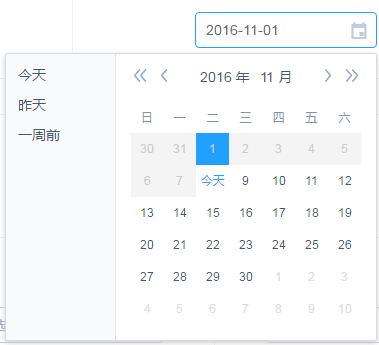
任意时间范围



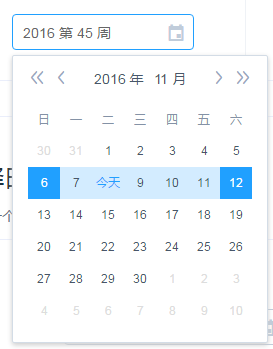
9、Date Picker 日期选择器



带快捷向：



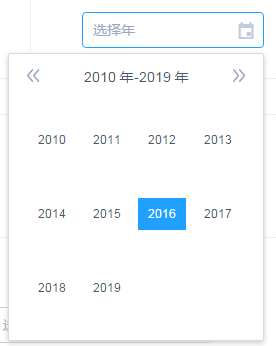
选择周：



选择月：



选择年：



选择日期范围：



10、Datetime Picker 日期时间选择器



11、Upload 上传

点击上传:



拖拽上传：



12、Form 表单



表单验证：



13、Rate 评分

基本用法：



辅助文字：

C:\Users\admin\AppData\Local\Temp\DB88.tmp.png

#### 四、Data数据

1. Table 表格

基本用法：



带斑马纹的表格：



带边框表格：



带状态表格：



固定表头：



固定列：



带固定列和表头：



多选：



排序：



筛选：



1. Tag 标签

基础用法：

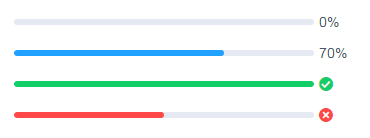
C:\Users\admin\AppData\Local\Temp\46E3.tmp.png

可移除标签：

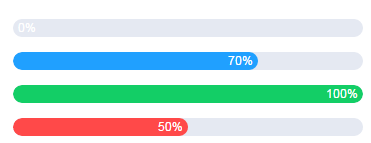
C:\Users\admin\AppData\Local\Temp\8A3B.tmp.png

1. Progress 进度条

线性进度条：



百分比内显：

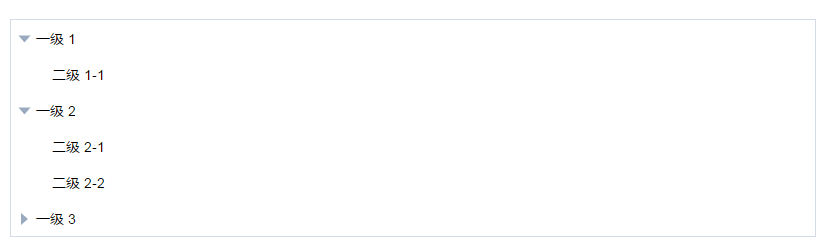


环形进度条：

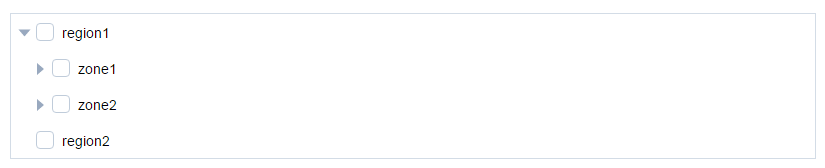


1. Tree 树形控件

基本用法：



可选择：



1. Pagination 分页

基础用法：

C:\Users\admin\AppData\Local\Temp\4A1C.tmp.png

小型分页：

C:\Users\admin\AppData\Local\Temp\FB34.tmp.png

附加功能：显示总页数

C:\Users\admin\AppData\Local\Temp\ABA0.tmp.png

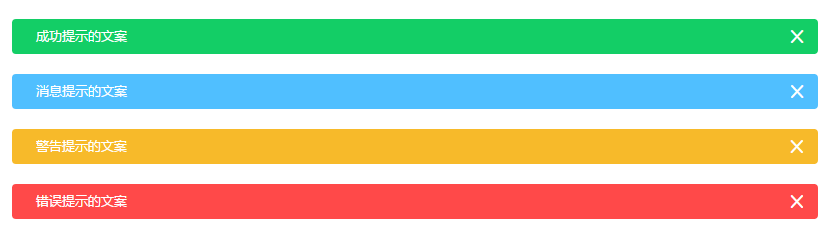
1. Badge 标记

C:\Users\admin\AppData\Local\Temp\8A3D.tmp.png

#### 五、Notice 消息框

1. Alert 警告

基本用法：



1. Loading 加载

区域加载：



整页遮罩加载：

1. Message 消息提示

C:\Users\admin\AppData\Local\Temp\D2B8.tmp.png



1. Message Box 弹框



1. Notification 通知



#### 六、NAV导航

* + - 1. NavMenu 导航菜单



* + - 1. Tabs 标签页



* + - 1. Breadcrumb 面包屑

C:\Users\admin\AppData\Local\Temp\9278.tmp.png

* + - 1. Dropdown 下拉菜单



* + - 1. Steps 步骤



#### 七、其他

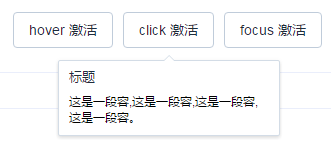
* + - 1. Dialog 对话框



* + - 1. Tooltip 文字提示



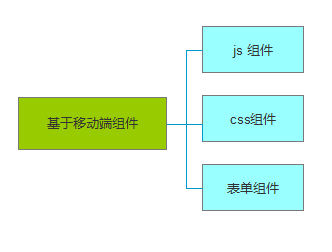
* + - 1. Popover 弹出框



* + - 1. Card 卡片

## 3.2移动端UI框架

结构图：



#### 一、配置文件：新建项目，项目结构为

|- src/ --------------------- 项目源代码

|- App.vue

|- main.js -------------- 入口文件

|- .babelrc ----------------- babel 配置文件

|- index.html --------------- HTML 模板

|- package.json ------------- npm 配置文件

|- README.md ---------------- 项目帮助文档

|- webpack.config.json ------ webpack 配置文件

#### 二、Js 组件

##### 1、Toast：

简短的消息提示框，支持自定义位置、持续时间和样式。

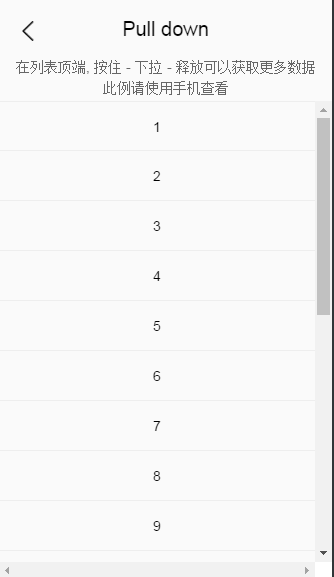


##### 2、Indicator:

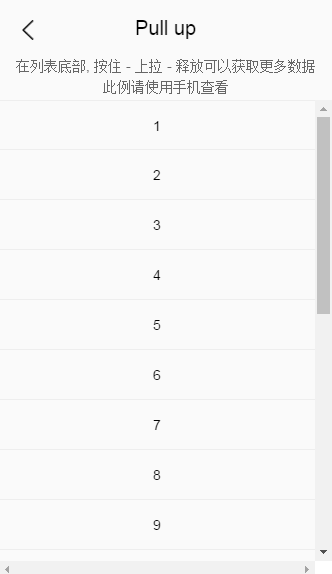
加载提示框，支持自定义文本和加载图标。



##### Loadmore



##### Infinite scroll



##### Message box



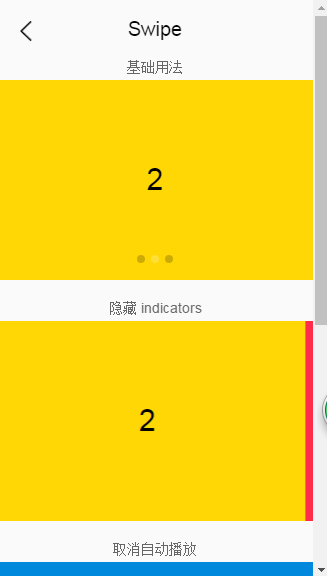
##### Action sheet



##### Popup



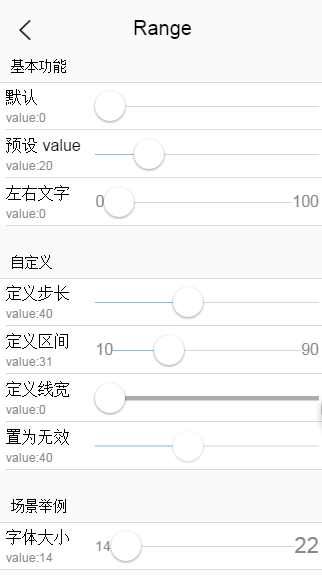
##### Swipe



##### Lazy load



##### Range



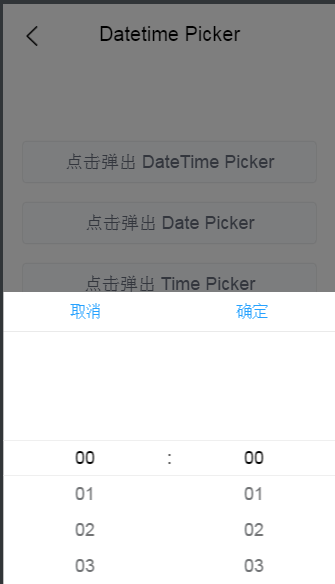
##### Progress



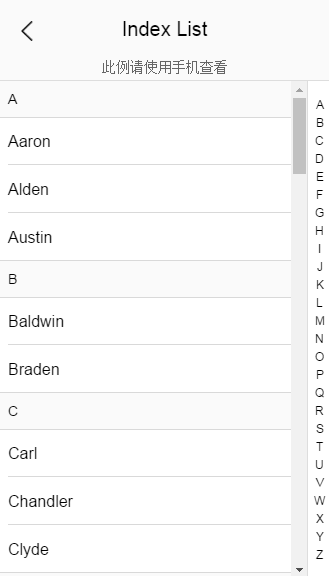
##### Picker



##### Datetime Picker



##### Index List



##### Palette Button (2.0)



#### 三、css组件

##### 1、Header



##### Tabbar



##### Navbar



##### Button



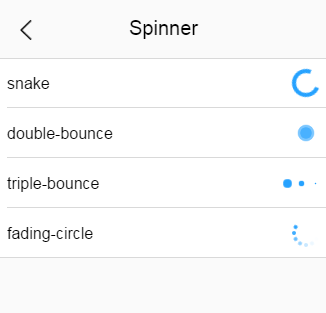
##### Cell



##### Cell Swipe



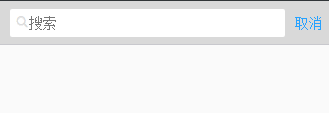
##### Spinner



##### TabContainer

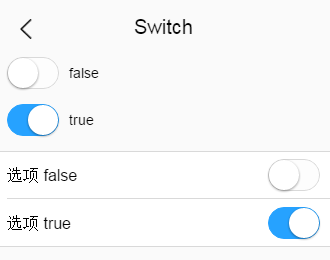


##### Search

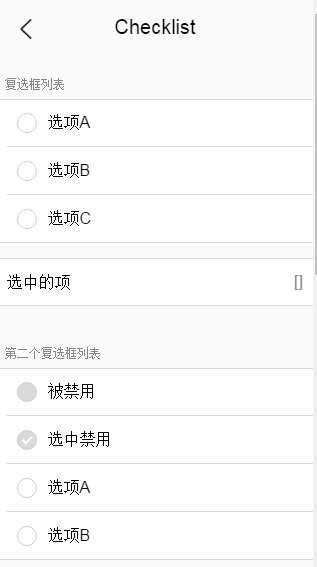


#### 四、Form组件

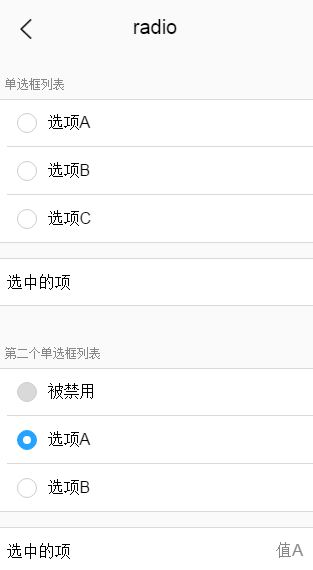
##### 1、Switch



##### Checklist



##### Radio



##### Field



##### 4、Badge



# 4、总结

1、感谢同事们对我的帮助，特别感谢我的领导田大昕和刘钊，给我这个为大家服务的机会。让我实现我想做而还没做过的web前端架构工作。

2、我会努力在项目中尽可能的实现，做好web前端架构的情况下，利用这套框架在项目中运用。我也会将项目放在github上，供大家分享。谢谢。