# 页面介绍

# Image-upload:（vue-upload-component）文件上传组件

接收参数：

url：上传接口地址

txt：组件展示的文本内容，非必需，默认为“上传”

# Pages（视图页面）：

### 标准查看：StandardView.vue

页面功能：

1. 增删改查，查询通过路由跳转到stdModelInfo.vue(参数：model\_id)
2. 上传，使用封装的Image-upload实现，其中参数包括：
3. <url:上传的接口路径>
4. @on-complete：上传成功的回调
5. @on-error：上传失败的回调

### 标准审核：StandardModify.vue

页面功能：

1. 查询列表：功能类型：funType字段等于1为标准审核，2为码表审核
2. 查看审核
3. 审核通过、审核拒绝：传参需要传递funType区分功能类型

### 标准检测：StandardDetection.vue

页面功能：

1. 检测：模型目录和标准目录进行字段比较，返回三种状态：符合、相似、不符合
2. 保存：保存最后一次检测记录

### 码表管理：StandardCode.vue

增删改查（新增、修改、删除会进入到标准审核列表中）

### 模型列表：StandardModels.vue

1. 查询列表：查询所有的模型
2. 查询检测历史：路由跳转到StandardHistory.vue（传参：id、name）

### 统计管理：StandardStatistical.vue

查询现有模型数、已检测模型数、现有标准模型数、现有标准字段数

# 文件夹结构

api： 公用方法及外部js

assets：静态资源

components：封装的公用模板

pages：视图页面

plugins：配置axios请求

router：配置路由(name和path)

App.vue：首页（包括sideMenu、顶部导航栏和route-view(所有视图页面显示的区域)）

main.js：入口js

package.json：所有依赖的配置、运行命令的配置等

vue.config.js：配置代理，用来解决一部分跨域问题

### 一般添加页面步骤：

pages下添加.vue文件

router/path.js下引入文件并配置name及path

api/menu.js 下配置sideMenu，注意name需要和path.js下对应的name相同

# 本地运行步骤

安装node: https://nodejs.org/en/

安装脚手架：终端运行 npm install –g vue-cli

终端打开项目文件（src文件夹同级）：运行 npm install 安装相关依赖，会生成node\_modules文件夹

npm run serve 运行项目

浏览器打开：[http://localhost:8080](http://localhost:8080/)

# nginx

1. npm run build 打包命令运行后生成dist文件夹
2. 复制dist下的全部文件到nginx的html文件夹下
3. 修改nginx.conf

location /stdglprj {

include uwsgi\_params;

proxy\_pass http://10.3.0.18:8080/stdglprj;

}

<http://10.3.0.18:8080>：部署服务器的ip及端口号