

修改人	修改时间	修改内容
冯帅	2018年4月2日	初稿

一、概述

融讯通H5插件应用是利用html5插件技术，将应用静态文件预置在App中，可以极大地提高html5应用的响应速度，也为HTML5开发提供了大量的JS内置接口，给html5开发提供了大量原生操作能力。

二、环境配置和前期准备

1.安装nodejs

vue插件应用工程是通过npm构建的，所以需要安装nodejs，安装后就可以使用npm了，安装相关问题请参考网上的安装教程。

官网地址：<https://nodejs.org/en/>

注：nodejs版本需6.0以上

2.安装IDE

推荐使用 **Visual Studio Code** (如何评价VSCode：<https://www.zhihu.com/question/29984607>)开发，VSCode不仅使用方便，里面有minxing（敏行）开发插件，后期调试、打包等都会用到，安装相关问题请参考网上的安装教程或者参考：

<https://www.minxing365.com/docs/html5/ide/VSCode/setupSoft.html>

3.建立工程

- 已有工程

svn或git拉取工程后执行：

```
npm install
```

npm会自动下载各个依赖包，然后执行：

```
npm run dev
```

成功后在浏览器中输入 localhost:8080 查看效果。

- 新建工程

复制TestDemo工程生成新工程，然后执行：

```
npm install
```

npm会根据 `package.json` 中的配置自动下载各个依赖包，下载成功后执行：

```
npm run dev
```

成功后在浏览器中输入 localhost:8080 查看效果。`npm run dev` 可实时监控代码变更实现自动刷新页面。

注：

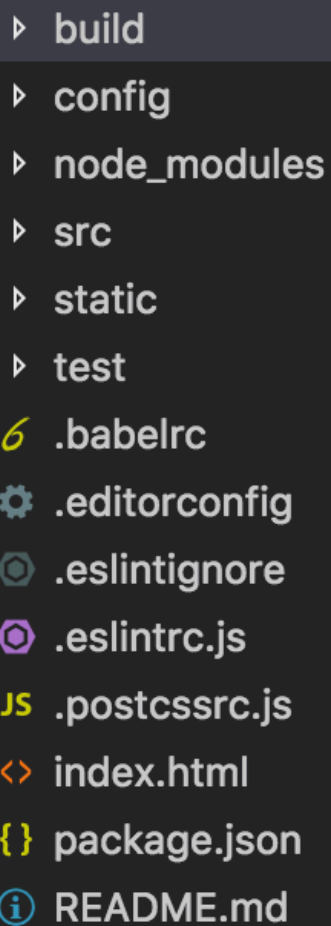
所有依赖包和相关模块都在 `package.json` 文件中（包括vue、webpack、等等），开发时可根据自身项目需要修改依赖。

在融讯通发布H5插件应用需要遵循融讯通平台规则，所以尽量使用TestDemo模板（或minxing插件）新建工程，以防上传到平台后无法识别。

三、vue工程说明

以TestDemo为例

1.项目结构



- build
- config
- node_modules
- src
- static
- test
- 🔗 .babelrc
- ⚙️ .editorconfig
- 🔒 .eslintignore
- 🔗 .eslintrc.js
- 📄 .postcssrc.js
- 📄 index.html
- 📄 package.json
- 📖 README.md

- **index.html**

是整个工程的入口文件

- **build**

build文件夹存放的是构建和编译相关的文件

- **config**

config文件夹存放的是参数配置文件

- **node_modules**

此文件夹为 `npm install` 自动生成，可忽略

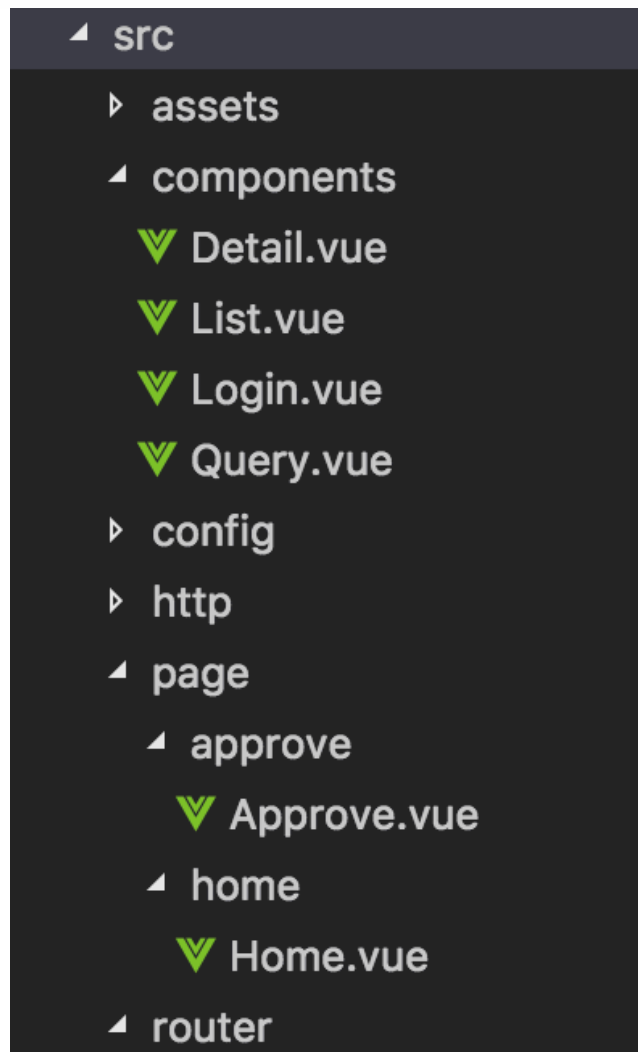
- **src**

所有功能相关的文件都存放在此文件夹内，页面放在page中；组件放在components中；assets为资源文件夹；

- **package.json**

npm依赖和模块配置文件

2.命名规范



文件夹小写命名，页面采用首字母大写驼峰式命名

3.路由配置

```
export default [{
  path: '/',
  component: App, //顶层路由, 对应index.html
  children: [ //二级路由。对应App.vue
    //地址为空时跳转home页面
    {
      path: '',
      redirect: '/home'
    },
    //首页
    {
      path: '/home',
      component: home
    },
    {
      path: '/approve',
      component: approve
    }
  ]
}]
```

所有通过路由的页面都需要在router.js中注册才能push

4.文件引入

如需引入bootstrap等文件可以参考如下方式：

bootstrap

在index.html中引入，引入bootstrap时注意加入标签实现响应式，未加此标签时，可能会出现手机模式时，响应式无法实现。

jQuery

在当前项目的目录下（就是package.json），运行命令 `cnpm install jquery --save-dev` 若是运行报错，则运行 `cnpm install jquery`（cnpm和npm都可以）这样就将jquery安装到了这个项目中。然后修改webpack.base.conf.js(在build文件下)两个地方：

1.加入

```
var webpack=require('webpack')
```

2.在module.exports的里面加入

```
plugins:[
  new webpack.optimize.CommonsChunkPlugin('common.js'),
  new webpack.ProvidePlugin({
    jQuery: "jquery",
    $: "jquery"
  })
]
```

最后在main.js中加入 `import $ from 'jquery'`，完成jquery的引入

5.JS API

客户端提供了一些JS接口来调用native，如：相机、网络请求等等

https://www.minxing365.com/docs/html5/js_dsk_description.html

其它相关调试请参

考：<https://www.minxing365.com/docs/html5/ide/VSCode/debugPrepare.html>

6.打包

① 首先需要修改本次打包的版本号，在 `plugin.properties` 文件中修改 `version_code` 和 `version_name` 字段；然后执行：

```
npm run build
```

然后就能看到自动生成的 `dist` 文件夹

② 打开minxing插件新建html5项目工程，新建工程后目录结构为

```
|-- plugin.properties
|-- /www
    |-- index.html
    |-- config.properties
    |-- /image
    |-- /css
    |-- /js
```

删除www目录下的其它文件，保留 `config.properties` 文件，删除后的目录结构为：

```
|-- plugin.properties
|-- /www
    |-- config.properties
```

③ 把dist目录下的index.html文件和static文件夹拷贝到新工程的www目录下，拷贝后的文件目录为：

```
|-- plugin.properties
|-- /www
    |-- index.html
    |-- config.properties
    |-- /static
```

④ 修改 `plugin.properties` 里的版本号，然后执行minxing插件打包，会自动生成zip文件，此文件为最终发布文件。

7.部署

登录部署平台 仿真地址：http://10.80.38.238/accounts/sign_in 生产地址：http://10.80.36.241:1443/accounts/sign_in#/login

添加应用

选择 `应用管理` -> `添加应用` 修改相关设置，然后把上一步的zip包上传

更新应用

找到要更新的应用，选择 `修改` -> `版本管理` 然后上zip包上传

注：vue工程不支持直接打包，所以需要间接打包

三、其它

融讯通插件应用支持所有H5框架开发，但须满足以下目录结构

```
|-- plugin.properties      --敏行插件配置文件
|-- /www                  --敏行插件资源目录
    |-- index.html --html5应用入口文件
    |-- config.properties --html5应用配置文件
    |-- /images --图片目录（可选）
    |-- /styles --CSS样式目录（可选）
    |-- /scripts --JS脚本目录（可选）
```

打包需要minxing的打包插件，手动打包可能由于操作系统编码不同，输出的文件敏行平台无法识别。

学习参考资料

1.vue官网: <https://cn.vuejs.org/>

2.vux参考: <https://doc.vux.li/zh-CN/>

3.webpack参考: <https://segmentfault.com/a/1190000006178770>