

React 脚手架

React 脚手架
简介
使用方法
目录结构
目录图
目录解析
规则
src文件夹
Mock文件夹
components与pages文件夹
bug总结

简介

该脚手架是基于 **create-react-app** 基础上添加修改功能的一个脚手架。现有的功能有：

git 仓库：[git@gitlab.stargt.com.my:frontend_developers_cn/h5-develop-framework.git](https://gitlab.stargt.com.my/frontend_developers_cn/h5-develop-framework.git)

- [eslint](#)（代码规范检查）
- [fetch](#)（[fetch](#)请求数据）
- [mock](#)（接口数据模拟）
- [webpack-dev-server](#)（热更新）
- [babel-plugin-import](#)（模块按需加载）
- [react](#)（前端框架）
- [react-route](#)（前端路由）
- [antd](#)（react ui framework）
- [rc-form](#)（表单验证）
- [scss-loader](#)（编译scss）
- [postcss](#)
 - [autoprefixer](#)（自动补全css）

- url-loader (图片大小控制, 小于10Kb转base64)
-

使用方法

- 克隆完项目之后, 安装依赖包

```
$ yarn install
```

- 常用命令

运行开发者模式

```
$ yarn start
```

运行产品构建模式

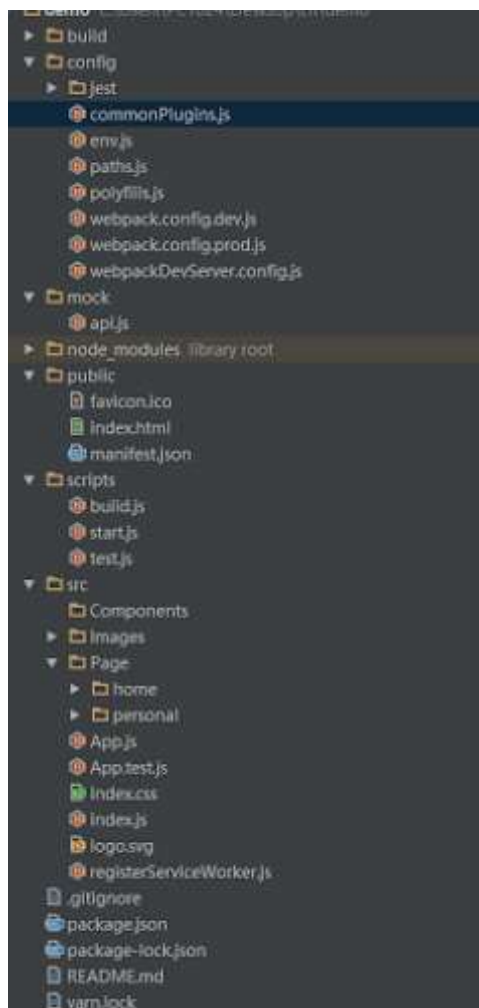
```
$ yarn build
```

运行测试模式

```
$ yarn test
```

目录结构

目录图



目录解析

文件夹	文件夹描述
build	当执行 <code>yarn build</code> 命令后出现，存放编译后文件
config	存放 webpack 的配置文件 commonPlugins.js 配置开发环境与生产环境共用的插件 webpack.config.dev.js 配置开发环境参数 webpack.config.prod.js 配置生产环境参数 webpackDevServer.config.js 配置webpack热更新本地服务器
mock	里面的 <code>api.js</code> 用于生成存放模拟数据
public	存放 index.html
scripts	存放命令启动脚本。 <code>start.js</code> 为开发者模式，与命令 <code>yarn start</code> 对应，以此类推...

文件夹	文件夹描述
src	存放项目资源文件，注意规则
	components 存放项目组件
	images 存放项目图片
	pages 存放项目页面
	App.js 项目的路由
	index.js 项目的主入口文件

规则

src文件夹

只有src中的文件参与进热更新

Mock文件夹

```
var Mock = require('mockjs');

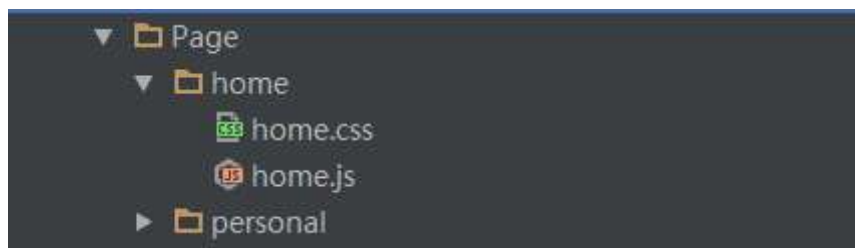
module.exports = {
  'test': {
    a: 2,
    b: 3
  },
  'demo': Mock.mock({
    'demo|1-10': [{
      'demo|+1': 1
    }]
  })
};
```

字段名为 api 名，假如字段名为 test，那么请求的时候请求 /api/test，便能获取到对应的数据

对应的数据可使用自定义的json格式；或者使用Mock.js动态随机生成，具体生成方法参考[Mock.js文档](#)

该文件数据不会加载到生产环境中，只用于开发环境

components与pages文件夹



page规则一致，在目录下再以个独立组件划分文件夹，独立文件夹中存放js和css代码

例如上方截图home页面与personal页面独立划分一个文件夹，里面存放各自需要的js和css代码

bug总结

有待后面慢慢发现~~！

