



基于 taro 的跨端组件库

贾磊

目录



一、背景

二、项目介绍

三、根据命令看项目

四、参考

一、背景



随着本地业务的发展，逐渐开发了【58本地招工】、【58本地招工 | 商家版】多个小程序项目。这些小程序之间，有些相同功能、高度相似的模块，总是需要重新开发或是粘贴拷贝、涂涂改改。开发和维护工作都比较麻烦。

单一项目的组件封装不能快速引用到其他项目中，所以需要将 these 模块整合起来。

于是产生了开发组件库的想法，落地了基于 taro 的组件库。

二、项目介绍



仓库地址: <https://igit.58corp.com/hrg-fe/zp-tongzhen/location-ui>

预览步骤:

- ① git clone [git@igit.58corp.com:hrg-fe/zp-tongzhen/location-ui.git](https://igit.58corp.com/hrg-fe/zp-tongzhen/location-ui.git)
- ② pnpm i (node版本 >14.6)
- ③ pnpm run build:taro:vue
- ④ pnpm dev:taro:weapp (如果缺少依赖需要先 cd src/sites/mobile-taro/ && yarn, 再返回项目根运行)

开发步骤:

- ① pnpm run add, 执行添加新组件脚本。该脚本会自动生成该组件目录、`config.json` 内该组件的配置信息等
- ② 完善src/packages/ 下组件的功能、文档、测试用例等, 在src/sites/mobile-taro/src小程序项目下完善demo
- ③ pnpm run dev:taro:weapp, 执行开发环境预览 taro 小程序文档脚本。该脚本会生成小程序文档路由, 打开微信开发者工具预览

三、根据命令看项目

添加组件

pnpm run add 执行添加新组件命令，实际上是执行 node 脚本 node ./scripts/createComponentMode.js 。

- 收集组件信息
- 生成 .taro.vue 文件
- 生成 .md 中英文文档
- 生成 .scss 文件
- 生成测试文件
- 更新 config 文件 src/config.json

```
> __tests__
└─ doc.en-US.md
└─ doc.md
└─ index.scss
└─ index.taro.vue
```

三、根据命令看项目



预览组件

需要在 `src/sites/mobile-taro/src` 小程序项目下完善对应的路由页面及 页面内容 demo

`pnpm run dev:taro:weapp` 执行命令拆解:

- `pnpm createTaroConfig =>` 执行 `node` 脚本 `node ./scripts/generate-taro-route.js` 更新小程序 `app.config.ts` 配置
- `pnpm checked:taro:vue =>` 执行 `node` 脚本 `node ./scripts/generate-ui-taro-vue.js` 生成入口文件
- `cd src/sites/mobile-taro/ && pnpm dev:weapp taro` 本地开发微信小程序
- 微信开发者工具打开小程序 `dist/miniProgram` 目录预览

三、根据命令看项目

构建打包

pnpm run build:taro:vue 执行命令拆解:

- pnpm checked:taro:vue => node ./scripts/generate-ui-taro-vue.js 入口文件
- vite build --config vite.config.build.taro.vue.ts
- vite build --config vite.config.build.taro.vue.disperse.ts
- pnpm generate:types:taro => node ./scripts/generate-types-taro.js
- pnpm generate:themes => node ./scripts/generate-themes.js 样式
- vite build --config vite.config.build.css.ts 主题样式
- vite build --config vite.config.build.locale.ts 本地化
- pnpm attrs => node ./scripts/createAttributes.js 属性

三、根据命令看项目



预览官网

官网分为两部分：

一部分是官网 PC 部分，包括官网头部、左侧导航栏、中间文档示例；

另一部分为 mobile 部分，即 mobile-taro 项目 build 的 h5，通过 iframe 嵌入到 PC 部分的右侧。切换 PC 路由时，同时更新 iframe 的 url。

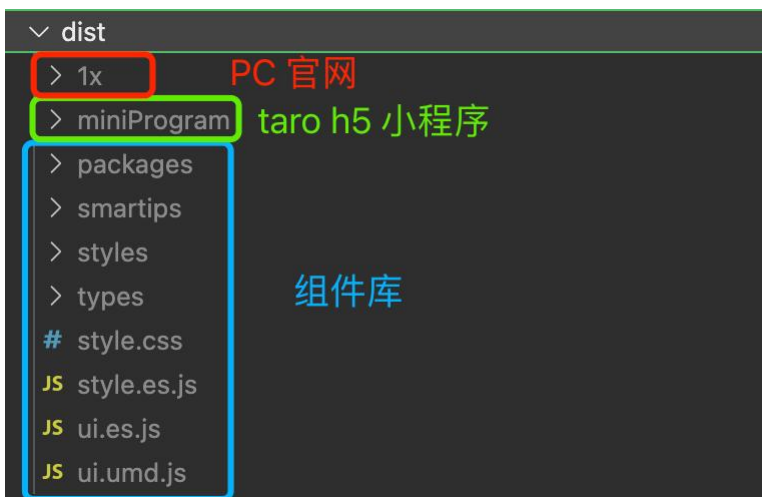
pnpm run dev 执行命令拆解：

- pnpm run build:taro:h5
 - pnpm createTaroConfig => node ./scripts/generate-taro-route.js 更新 mobile-taro 的 app.config.ts
 - pnpm checked:taro:vue => node ./scripts/generate-ui-taro-vue.js 生成组件库入口文件
 - cd src/sites/mobile-taro/
 - pnpm i
 - pnpm build:h5 生产环境 taro h5 build
- pnpm run checked
- vite --open --force vite 开发环境预览

三、根据命令看项目

构建官网

构建官网前部分跟【预览官网】一样，就是最后一步 `vite --open --force` 改为 `vite build` 。



四、参考

- [nutui源码探究——taro组件打包篇（一）](#)
- [nutui源码探究——taro组件开发篇（二）](#)
- [nutui源码探究——taro组件开发篇（三）](#)