



吴睿 - 前端开发工程师

自我介绍

- 男，24 岁
- 中山大学（2015.08 - 2019.06），本科，软件工程。
- 邮箱 - wurui7@mail3.sysu.edu.cn
- QQ/微信 - 261857860
- 在线简历 - <https://modyqyw.github.io/resume/>，方便地查看以下各个链接。
- Github - <https://www.github.com/ModyQyW>，展示了我的一些开源项目。
- 个人站 - <https://modyqyw.github.io>，展示了我个人的学习路径和一些知识归纳，也可以在语雀 <https://www.yuque.com/modyqyw> 查看。
- 对过往业务相关较为熟练，注重归纳总结并探索同一业务更好的实现和处理方式。
- 理解前端常见的开发技术，包括但不限于 `jquery`，`vue`，`react` 以及其它常用周边库等。
- 拥有良好的编程习惯，注重文档、代码规范和必要的注释。
- 学习能力较强，能在短时间内了解未接触过的技术并运用，乐意接触前沿新鲜技术。
- 对语言和框架底层的理解有待提高。

工作期望

- 正式岗位（非人力外包岗位）
- 佛山禅城/南海工作
- 10-12k，双休，五险一金
- 良好的个人和公司发展空间
- 和谐、自由、包容的公司和团队氛围
- 有计划性的工作安排

过往工作

广东砺云教育科技有限公司 2019.06 - 2021.04

担任前端开发工程师一职，下面列写持续 1 个月以上的工作内容。

都城快餐 2019.06 - 2021.04

公司早期外包项目，为都城快餐提供电子化支持，包括桌面端后台管理网页系统、微信支付宝小程序和移动端应用等。

本人负责前端选型、开发、维护和重构工作，技术栈涉及

vue，uni-app，react-native，taro 等。

1. 接手遗留项目，主要涉及维护（修复遗留问题、补全开发规范和文档）和重构（基于 vue 和 uni-app，全面拥抱 vue 生态），保证线上项目稳定运行。
 - 基于 angular 1 的后台管理系统。
 - 基于 wepy 1 的微信小程序。
 - 基于 expo + react-navigation + mobx + typescript 的移动端应用。
 - 自管理更新 self-hosting updates。
2. 领导开发在线订餐系统，主要涉及选型、设计和协同开发。
 - 与产品、后端三方协调，了解项目全貌。
 - 基于 vue 2 + element-ui + vue-router + vuex 的后台管理系统。
 - 多角色支持。
 - 统一布局，配置式侧边栏。
 - 修改 vue-cli 配置以自定义，包括通过 sass-loader 注入自定义的 sass 变量，自动去除 console，增加路径别名等，分块构建，增加 stylelint 校验等。
 - 基于 axios 封装请求，包括请求/响应拦截器，自动错误处理，格式化数据，基于环境变量请求不同服务器等。
 - 结合 np 和 shell 文件，一条命令快速部署。
 - 基于 taro 2 + mobx + typescript 的微信支付宝双端点餐小程序。
 - 微信小程序名 都城送，需要切换地址到广州，完整功能的体验时间：11:00 - 13:00，17:30 - 19:00。支付宝小程序内测中。

- 根据要求实现样式，参考 `tailwindcss 1` 实现项目内原子化样式，后续封装成库。
- 基于 `taro.request` 封装请求。
- 使用 `uni-app` 完全重构。



MTU 互动学习平台 2020.07 - 2020.11

公司自研项目，助力教学智能化，涉及包括桌面端后台管理网页系统和微信小程序。

本人负责桌面端后台管理网页系统的选型和开发工作，技术栈涉及 `vue`。

1. 基于 `vue 2` + `element-ui` + `vue-router` + `vuex` 的后台管理系统，高校使用的教学资源平台。
 - 查看教学资源时，使用 `EventBus` 处理侧边栏、面包屑和页面主体三个组件之间的数据流动，区分开 `vuex`。
 - 封装 `vue-simple-uploader`，针对不同文件格式显示不同图标，自定义添加和上传完成逻辑以控制上传行为。
 - 轻度修改 `element-ui` 的面包屑，支持点击最后一个面包屑项，以刷新页面。
 - 轻度修改 `element-ui` 的树，支持懒加载失败时可以重新加载。

基础建设

- 前端规范 <https://millcloud.github.io/frontend-standard/> - 聚合社区内现有的很多优秀实践而来的规范，旨在降低新成员融入的成本，提高开发、研发、协作效率，避免开发中常见的问题，覆盖学习规范、编码规范、工作流规范、异常处理和监控规

范、测试规范、文档规范、前后端协作规范等，也涉及提高方面的建议。

- boilerplate-vue <https://github.com/millcloud/boilerplate-vue> - 在 `vue-cli` 创建的模版基础上，结合过往项目经历，加入开发常用配置的模板，包括组件库、`vuex`、`vue-router`、`lodash`、`dayjs`、`idb`、`axios` 等，使用 `vue` 做桌面端网页、移动端网页和桌面端应用开发，支持 `vue 2` 和 `vue 3`，目前 `vue 3` 部分尚不稳定。
- boilerplate-uni-app <https://github.com/millcloud/boilerplate-uni-app> - 和上一个模板类似，使用 `uni-app` 做小程序和移动端应用开发，没有使用 `vue-router`，请求使用了 `uni-ajax`，只支持 `vue 2`。

开源项目

- @modyqyw/fabric <https://github.com/modyqyw/fabric#readme> - 不同前端项目的可共享规范，涉及 `editorconfig`、`prettier`、`eslint`、`stylelint`、`markdownlint`、`commitlint`、`lint-staged`、`husky`、`vscode` 等配置。
- @modyqyw/utils <https://github.com/modyqyw/utils#readme> - 不同前端项目的可共享工具方法库，主要参考 `lodash-es`，加入了一些其它常用方法。
- @modyqyw/mp-scss <https://modyqyw.github.io/mp-scss> - 一个基于 `flexbox` 的小程序 `scss` 库，用于快速实现自定义设计，定位位于 `tailwindcss` 和 `bootstrap` 之间，既提供了原子化的功能类，也提供了用于快速开发的布局类和组件类。
- @modyqyw/dsa <https://github.com/modyqyw/dsa#readme> - 用 `typescript` 实现的部分数据结构和算法，期望用于个人学习和实际开发。使用 `jest` 测试，使用 `rollup` 构建。
- uni-helper <https://github.com/modyqyw/uni-helper#readme> - 增强 `vscode` 对 `uni-*` 的支持。

Demo

- vue2-todo-list-demo <https://github.com/ModyQyW/vue2-todo-list-demo> - 一个简易的待办清单应用，使用 `vue 2` + `vuetify` 制作，在线体验 <https://vue2-todo-list-demo.vercel.app/>。
- next-mine-sweeper-demo <https://github.com/ModyQyW/next-mine-sweeper-demo>

- 一个简易的扫雷游戏，使用 `next 10` + `tailwindcss 2` 制作，在线体验 <https://next-mine-sweeper-demo.vercel.app/>。
- next-fetch-demo <https://github.com/ModyQyW/next-fetch-demo> - 一个简易的获取 Github 仓库的示例，使用 `next 10` + `tailwindcss 2` 制作，在线体验 <https://next-fetch-demo.vercel.app/>。