

梁国华

- 1995/男/在职
- 13312967837/1017118152@qq.com
- 统招本科/华南师范大学/计算机学院/软件工程专业
- 工作年限：2.5 年
- 求职意向：前端工程师

技能清单

- 熟练 JavaScript 基础；熟练 TypeScript 编程语言；熟练 ES6 等语法糖；
- 熟练 React 及其全家桶（Router, Redux, Hook 等）的源码实现原理；
- 熟悉 Webpack 打包工具，了解其构建原理、AST 和常用的优化方案，能根据需求自定义 cli 工具；
- 熟悉 Koa/Express/NestJS 等 Node.js 的服务端框架，了解其实现，能运用中间层转发 API；
- 掌握函数式编程思想，掌握相关的 RxJS 库；
- 掌握 CommonJS、ES Module 等模块化规范；掌握 MVVM/MVC 等常用的设计模式；
- 掌握 HTML/CSS3 等 Web 开发技术，能快速开始网站开发；
- 了解 CSS 预处理器 Sass，增强对 CSS 的管理与维护；
- 了解 NPM, Yarn 等包管理工具，将常用的工具库发布到 NPM；了解 Git 基本命令；
- 了解浏览器渲染机制；了解各种渲染优化指标，有前端性能优化的经验；了解前端异常处理和监控；
- 了解基本的计算机网络概念，HTTP 相关的协议；

工作经历

2017.08-2018.03 广州市纵横数码有限公司 后端开发（实习）

- **岗位职责：**编写和调试后端接口，完成结合后端驱动的网页制作。
- **团队贡献：**很快完成了三个模块，成语测试/成绩统计/语文基础知识测试，协助公司完成当时在香港举办的教育分享会。

2018.07-至今 深圳市台电实业有限公司 前端开发

- **岗位职责：**维护了几个老项目/独立开发了两个设备内置网站/参与开发了两个 PC 端网站。
- **团队贡献：**封装工具库和组件库帮助减少开发工作/重构 Redux 相关的工作流，降低耦合度，易于维护/自定义 cli 工具/打包公共模块，开箱即用/优化了网站性能和渲染效率/帮助新人解决问题和分享技术。

项目经验

8600 主机的内置网站

- **项目背景：**对接主机设备，提供可视化控制设备和调节参数的网站。其中，PC 端根据 WebSocket 与设备端进行通信，通过在 PC 端网站上操作，能设置主机或是分机的设备参数。
- **项目分工：**PC 端由本人独立开发，与嵌入式同事协作合作，他们负责提供数据和协议。推动产品正常交付，协助桌面端的同事完成对应的桌面应用
- **成果和亮点：**

个人简历

1. 引入 Webpack 打包方式，优化代码块体积和自定义插件。

方案：对组件库按需引入；配置提供公共代码，将业务代码、公共模块和第三方模块分开不同文件，最后单个文件不超过 2m；优化打包的流水线，添加插件，解决打包后需要压缩加密文件的无用功；

2. 首屏加载维持在 1s 左右，较之前提升了 200%（以前架构没考虑过优化问题）。

方案：懒加载模块；尽量压缩首屏需要的资源；小图片采用前端缓存；虚拟长列表；减少 JS 操作 DOM 等

3. 提供了 PC 端与设备端在进行 WebSocket 数据交换时，压缩解析和转换部分的代码的解决方案。

方案：利用 TypeScript 装饰器的语法特性，封装了一个公共库，发布成一个 NPM 包。

4. 为此类设备内置网站项目更换了组件库，并且根据公司 UI 设计习惯，制定了一套 Sass 变量表。

- **思考与进步：**在开始项目的时候，面对需求不是机器化模仿以前的开发方式，重复机械性的作业很难让人得以进步。而要多思考有没有更好的方式可以解决问题，如何让程序帮助你完成重复性高的作业。后面回想，能改进的空间依然非常大。这个项目培养了本人独立思考和解决问题的能力，对前端性能有了初步的认识，以及之后深入 Webpack 原理奠定了一定的基础。

智慧教育平台

- **项目背景：**一个多媒体智慧教学、远程教学互动、无线物联管控、音视频点播、能提升教学质量等智能化管理平台。通过平台直接统一接管和调配整个校园的有线和无线设备资源，管理员可以通过制定一系列任务方案进行智能化管理。

- **项目分工：**此项目由三个组合作完成，PC 端网站、服务端以及设备端。其中，PC 端由本人和另外一个同事完成，负责与服务器端进行交互。本人负责了 PC 端网站的大部分业务，封装公共可配置可复用组件，制定配色方案，模块之间的组织和代码规范等内容。

- **成果与亮点：**

1. 重构 Redux 工作流程，将 Redux 状态和更新逻辑与业务代码进行解耦。重构后代码结构更加清晰，且减少了 1/3 的冗余逻辑代码。

方案：利用 TypeScript 类型约束，减少运行时错误；借鉴函数式编程思想，把混乱且不断在不同文件中重复的处理逻辑抽离成一个工具库，封装成 API，使用者只需要知道什么操作调用什么 API 即可。

2. 封装动态模块，开始新项目时可以开箱即用。

方案：抽离公共逻辑，整合成一个动态模块，发布成一个 NPM 包。

3. 封装可配置可复用的组件，能达到状态和逻辑复用的效果。

- **思考与进步：**这一个项目考虑了很多团队合作需要考虑的问题，代码规范，组件复用，注意代码便于维护等。在完成业务逻辑的过程中，每一步都在想会不会有更好的解决方案，为此本人去研究了 TypeScript，研究了 Redux，了解函数式编程，当时刚好在学习 React 全家桶源码，所以就把自己学习到的设计应用到了项目中，并且将重构后的架构分享给了同事，还得到了同事的高度认可。读源码理解别人优秀的设计，对自己的架构思维和码代码的能力会有很大的提升。

自我评价

本人热爱挑战难题，对求知的过程比较痴迷。本人对生活 and 未来充满了期待，非常乐于帮助别人，渴求能与身边人一起进步。为此，本人坚持每天下班后都要学习，巩固基础的同时多阅读优秀的框架源码，逐步培养架构思维。希望以后有能力可以多参与开源项目，参与社区的讨论。