

# 王超

求职岗位: Web 前端开发工程师

性别: 男 电话: 18571351438  
户籍: 湖北随州 年龄: 24  
邮箱: 18571351438@163.com



## 求职意向



意向岗位: 前端开发工程师

期望薪资: 面议

## 相关技能



1. 熟练掌握 HTML、CSS 和 JavaScript,包括 DOM 操作、事件处理、异步编程等;
2. 熟练使用 React、Vue 等主流前端框架,能够快速构建复杂的单页应用程序;
3. 熟悉 TypeScript 和 ECMAScript 的语法和特性,并在实际项目中使用过;
4. 熟悉 W3C 的标准,能够编写符合标准的代码;
5. 熟悉 Web Components 技术,能够使用自定义元素、影子 DOM 和 HTML 模板构建可重用的 Web UI 组件;
6. 熟练掌握小程序开发(微信小程序和 uni-app 框架)、H5 页面开发和混合开发技术;
7. 熟悉前端工程化构建工具,如 Vite、Webpack、Babel 等,能够优化 Web 应用程序的性能和代码质量;
8. 熟悉 Node.js、Express 等后端技术,能够实现一定程度的全栈开发;
9. 熟悉 MongoDB、MySQL 数据库的增删查改等操作;
10. 熟练使用 Mock、ApiFox、Postman 等工具来进行接口测试和调试;
11. 熟悉 Git 等版本控制工具,能够熟练使用团队开发流程;
12. 熟悉 Linux 系统命令和 shell 脚本,能够完成项目的部署和维护以及个人网站的搭建;
13. 熟悉的 UI 组件库有 Semantic、Vant、Element、Antd、Naive;了解常用的数据结构与算法。

## 实习经历



2022.8-2023.3

武汉朋悦科技有限公司

前端开发实习

- 负责公司项目中 Web 页面制作 (HTML+CSS) 和实现页面上各种交互效果、切换效果;
- 持续优化现有前端技术体系、前端用户体验及页面响应速度, 保证适配和兼容性;
- 通过 Ajax、axios 与后台数据进行交互, 与后端工作有效对接, 优化网站前端性能;



2022.12-至今

房屋租赁网站

**使用技术:** Vue3 + TypeScript + Vue-Router + Naive UI + Pinia + Axios + Echarts + Vite + Node.js + Express

**项目简介:** 本项目主要实现房屋租赁网站的后台数据管理,主要的功能模块有房屋租赁管理、房屋信息管理、用户管理、公告管理等。

**项目描述:**

1. 项目通过 Vite 脚手架搭建,各个组件通过 Vue-router 注册实现跳转,利用 Pinia 来管理全局的数据状态,以及 LocalStorage+pinia-plugin-persist 插件实现数据持久化。
2. 首页模块使用 Echarts 以图表的形式展示常用的房屋租赁相关的数据,配合 excel.js 包使用导出 Excel 文件。
3. 注册时使用 Naive UI 中的表单组件,其中在密码输入和确认密码时利用了表单自带的 validateField 方法,给密码添加 onblur 事件来监听这个方法,用来验证表单中具体的某个字段,防止用户在输入确认密码后再次修改密码导致两次密码不一致而验证通过的问题。
4. 用户权限部分面向的分别是管理员,屋主,租客,在登录成功后会判断用户的权限,利用 v-if 控制显示导航的板块,例如:屋主只有首页界面,公告列表界面,房源信息界面等。
5. 嵌入子组件,点击按钮弹窗创建公告。此处需要上传图片附件到公告列表,使用 Naive UI 的 upload 上传,由于图片有上传时间,需要等待图片上传提示完成才能提交公告,所以这里加一个上传成功的条件在点击创建公告信息前判断一下。

**项目难点:**

- 通过路由守卫的使用做权限管理。前端会有一份路由表,每一个路由只有对应可访问的权限。当用户登录之后,通过 token 获取用户的 role,动态根据用户的 role 算出其对应有权限的路由,再通过 router.addRoutes 动态挂载路由。
- 使用 exceljs 包来导出数据。首先我们需要引入依赖包,定义导出 Excel 文件需要的数据类型,然后使用依赖包定义一个对象,后期使用该对象设置参数和导出数据,设置好表格的 header 和表格顶部的信息,以及表格的主要数据部分,做好数据美化。

2022.9-2022.11

移动端购物商城

**使用技术:** Vue + Vue-Router + Vant UI + Pinia + Axios

**项目描述:**

1. 针对移动端使用 lib-flexible 和 postcss-px2rem(淘宝适配)方案对移动端页面进行相关排版与设置样式。
2. 使用 Pinia 状态管理库搭配 pinia-plugin-persist 插件,实现在客户端的数据持久化存储。
3. 使用 VueRouter 对所有页面进行相关路由配置,设置首页重定向,404 页重定向,引导客户进行页面浏览。
4. 使用 Vant 相关组件对页面功能布局,包括弹出层,表单,下拉菜单,详情展示,购物引导,城市选择等页面的互动,增强用户交互感与体验感。
5. 使用 swiper 封装能够反复利用的轮播组件,在首页和详情页进行展示。
6. 对 Axios 的相关请求进行封装,设置请求及响应拦截器,使用 vant 的 toast 组件进行提示,封装请求成自用的 ajax,对不同类型的请求设置不同的方法及参数和请求头。
7. 用户在登陆完成后,可以对自己的点赞,收藏,购物车进行管理,可以查看自己的订单及操作订单相关状态,与后台交互,完成订单的整套流程,向服务器发送相关处理。
8. 在分类页如果点击蓝色加号,可以将商品立即添加入购物车,可对商品类别进行选择,进入详情页同样可以点击加入购物车,完成对商品添加。

9. 在对订单处理时,用户在详情页点击立即购买时,将会立即生成订单,如果用户若放弃当前订单,生成的订单将暂存到用户的待支付订单。用户可以对商品进行搜索,提供价格筛选及类别筛选,同样可以使用关键字搜索。

#### 项目难点:

- 项目在对数据进行请求时,遇到相关跨域(CROS)问题,尝试使用设置请求头,无效仍然产生,查阅发现可能需要使用反向代理(Proxy),尝试使用后,能正常获取数据。
- 在项目中遇到过 mounted 钩子函数中请求数据导致页面闪屏问题,最后把请求数据,放在 created 里解决了该问题,如果在页面挂载完之前请求完成的话,就不会看到闪屏了。

2023.3-至今

出入库管理系统

**使用技术:**uniApp + uView UI + Node.js + Express+Echarts + MongoDB

**项目简介:**本项目是一款移动端出入库管理系统,旨在实现出入库流程的数字化管理以及便捷性。

#### 项目描述:

1. 多端兼容性:基于 uniapp 框架,可以同时打包成小程序、H5、安卓和 IOS 原生应用,适应不同用户群体的使用习惯。
2. 界面简洁易用:应用的界面设计简洁美观,用户操作流畅,降低了使用难度。
3. 数据安全可靠:使用 Token 验证机制保证用户登录安全,数据存储使用了加密算法进行保护。
4. 出入库登记:可以记录每个商品的出入库情况,包括进货、销售、退货等,以便跟踪库存情况。
5. 仓库商品数量可视化:可以通过可视化的方式展示仓库中不同商品的数量,方便管理人员了解库存情况。
6. 产品管理:可以管理仓库中的所有产品信息,包括产品名称、型号、价格、生产日期等,以便更好地管理库存。
7. 供应商管理:可以管理供应商的信息,包括名称、联系方式、地址等,以便及时地采购物品。
8. 出入库记录查询:可以查询每个商品的出入库记录,包括数量、时间、出入库类型等信息,以便了解商品的流动情况。

#### 项目难点:

- Echarts 图表在微信小程序上的使用难点:需要利用 canvas 发布来绘制图表,还要考虑新规范的配置问题,解决办法是对 Echarts 的初始化配置进行组件封装,然后对绘制方式进行调整和优化。
- 后端接口设计上的难点:在数据库文档结构设计、接口设计、接口的安全性和稳定性上,花费了大量时间进行优化和调试。

2022.10-2022.12

云在线校园后台管理系统

**使用技术:**React+TypeScript+Mobx+Antd+Mongo+Echarts +Umi

**项目简介:**该项目主要用于校园信息的录入管理的后台系统,主要从用户与管理员的角度分别 进行需求分析的工作, 实现高效、安全、合理的管理模式,方便对校园信息的管理。主要模块为校园用户管理、学生意见、公告管理、教务管理以及学生成绩的录入。

#### 项目描述:

1. 项目整体采用 Umi+AntDesign 搭建, 使用 Mobx 作为状态管理库, TypeScript 进行数据的强类型定义, 使用 Yarn 作为软件包管理器。
2. 路由表根据登录用户的权限, 从接口获取数据然后使用 patchClientRoutes()方法进行动态添加。
3. 项目中的网络请求使用@umijs/max 的内置方法 request,并进行请求和响应拦截器的二次封装。
4. 针对 PC 使用 Antd 相关组件对页面进行相关排版与设计
5. 根据项目需求, 使用 react-router-dom 实现页面的重定向以及侧边栏路由的多层嵌套。
6. 封装侧边栏组件, 使用路由递归, 动态实现根据不同权限展示不同的页面模块。
7. 通过 React.lazy 实现路由组件的懒加载, 提高模块的响应效率。

8. 通过使用 react-router-dom 里面的 useNavgate, useLocation , useParams 来进行路由跳转及传参。
12. 加入 token, 后端接受 token 才返回数据, 如果 token 过期则为不返回, 拦截 axios 响应后的请求, 如果
13. token 过期则跳回登录界面。
14. 使用 echarts 插件中折线图, 饼图实现各部门人员分布数据及公告分类数据的可视化。

#### 项目实现及问题:

- 登录注册运用 antd 中的 form 表单,登录后采用 token 的方式来检验用户信息,将 token 存入 sessionStorage 中,在请求时通过拦截器 interceptors 设置 token 来携带请求信息,当 token 失效后就重定向到登录界面;
- 登录成功后,用户可以在头部组件中管理自己的基本信息,包括头像的上传,修改密码,和 用户退出功能;
- 权限管理模块则是通过 antd 中的表格组件展示已有的权限,管理员可以通增加,删除修改等操作实现权限角色的设置侧边导航通过存入路由信息和权限信息组成的数组,然后进行筛选递归,将匹配用户权限的板块渲染出来,从而形成多级嵌套的侧边导航菜单;
- 异步问题,在使用 echarts 时刷新页面导致数据消失,解决办法通过 async+await 获取到数据后再执行 echarts 的初始化函数进行渲染;
- 在 hooks 中,修改状态是通过 useState 返回的函数来实现的,而 useState 返回的函数修改状态存在异步,无法直接设置值并使用,通过 useEffect 去监听值的改变;

#### 教育经历



2017.9-2023.6

武汉工程大学邮电与信息工程学院

计算机应用技术

#### 自我评价



- 计算及专业背景,基础知识扎实,对前端技术有浓厚的兴趣,喜欢并不断学习新技术,逻辑清晰,有阅读文档的自学能力;
- 熟悉前端工程化与模块化开发,理解并熟练运用组件化开发思想,并了解 Node.js 后端开发语言;
- 有较强的逻辑思维能力,善于分析、归纳、解决问题的能力;
- 具备良好的沟通能力和团队意识,有较强的抗压能力不怕吃苦,能够很快适应各种工作环境,能合理规划时间;