

朱心熹

性别：男 | 年龄：22 | 联系方式：18558612103 | 邮箱：853539461@qq.com

教育经验

常熟理工学院 2018.09 ~ 2023.06

软件工程 本科 1% ~ 10%

个人技能

前端：掌握 HTML、CSS、JS、TS 以及前端常用工具的使用；熟悉 Vue2(Vue3)、Pinia、Vuex、Vue Router、ElementUI(ElementPlus)、Node.js 的使用；了解 React、React Router、Redux Toolkit、Ant Design、Electron 的基本用法以及 Webpack 的基本配置；了解一定的前端性能优化

后端：熟悉 Spring Boot、Mybatis Plus、Mysql 的基本使用；了解 Spring Security、Redis、Elasticsearch 的基本用法

实习经历

苏州科达科技股份有限公司 2021.07 ~ 2021.11

行业应用开发部 | Web前端开发工程师

一、苏州针眼：面向苏州多家政府机关单位，提供日常执勤数据的信息化管理

- 1、基于 Vue2 + Vue Router + Vuex + 公司 ui 库 (ElementUI) 对系统“模板导入”等模块进行前端开发
- 2、针对“模板导入”模块动态表格的可行性进行深入分析，实现了同张表格不同表头特殊情况下数据的修改与展示
- 3、review 项目代码并进行性能优化，提升系统的交互体验

二、新视图库（数据中台）：面向全国多家政府机关，对数据进行集中管理及可视化展示

- 1、基于 Vue2 + Vue Router + Vuex + ECharts + 公司 ui 库 (ElementUI) 对系统中数据管理模块、部分数据展示模块进行开发
- 2、针对图表复用问题与导师进行探讨，完成常用图表的组件封装

项目经历

盒子音乐 2022.04 ~ 至今

全栈开发

结合所学知识，设计并实现一款能够个性化推荐音乐、播放音乐、下载音乐、导入本地音乐并读写音乐元数据的桌面端音乐播放器

1、桌面端播放器：

- ① 基于 Vue3 + Vue Router + Pinia + Vite + ESLint + TS + ElementPlus + Electron + FFmpeg 进行开发
- ② 通过 Vue3 技术栈，配合使用 Pinia 作为状态管理工具，并使用 ESLint 作为代码规范工具，对前端基础模块进行组件化、规范化开发，实现登录、注册、主页推荐、歌曲歌单分类、专辑收藏、喜欢的歌单、音乐下载、播放列表、播放页面、滚动歌词、播放控制等基础页面功能

- ③ 结合 **Electron** 中主进程与渲染进程通讯，配合监听 **Electron** 的 **will-download** 事件对音乐下载进度进行管理，实现 **批量下载、暂停下载、继续下载、取消下载** 以及当异常退出程序后的 **下载恢复** 等基础功能
- ④ 利用 **FFmpeg** 对下载完成的音乐进行歌曲标题、专辑信息、歌手信息、歌词、歌曲封面等 **音乐元数据的写入**，使 PC 端、移动端等默认播放器可读取该音乐的元数据，解决了在一些大众音乐平台所下载的音乐中只有音乐而无音乐基本信息的问题；同时可对本地音乐的元数据信息进行读取、解析
- ⑤ 配合 **Electron** 对项目进行打包，生成 Windows 端 **.exe** 安装包

2、Web 端音乐后台：

- ① 基于 **React18 + React Router + Redux Toolkit + Vite + ESLint + TS + Ant Design** 进行开发
- ② 通过 **React18** 技术栈，配合使用 **Redux Toolkit** 作为状态管理工具，并使用 **ESLint** 作为代码规范工具，对项目页面进行函数式组件化、规范化开发，实现了管理员注册、管理员登录、个人中心、用户管理、角色管理、权限管理、音乐管理、歌手管理、分类管理、专辑管理、歌单管理、收藏管理等系统功能
- ③ 结合 **React Router** 与后端基于 **rbac** 的权限控制系统配合，实现前端 **动态路由** 以及 **按键级** 的权限控制，如 **动态导航栏、动态按钮** 等

3、音乐爬虫：

- ① 基于 **Node.js + Koa2 + TS + Mysql + 网易云音乐 API** 进行开发
- ② 利用 **Node.js、Koa2** 对整体项目框架进行 MVC 模式的爬虫接口搭建
- ③ 通过以 **网易云音乐 API** 作为数据来源，对音乐数据进行结构化爬取
- ④ 结合 **Mysql** 对音乐数据进行 **数据入库**，并结合 **Node.js** 的 **fs** 模块进行音乐数据下载，实现音乐数据 **本地化存储**

4、后端系统：

- ① 基于 **Spring Boot + Spring Security + JWT + MyBatis Plus + Mysql + Redis + Elasticsearch + Logstash + Kibana** 进行开发
- ② 通过 **Spring Boot** 结合 **Spring Security、JWT** 对系统进行 **动态鉴权** 并基于 **rbac** 模式进行 **权限设计**，同时利用 **Redis** 对 **token** 进行存储与管理
- ③ 结合 **Redis** 重写 **MyBatis Plus 二级缓存**，**大幅** 减少了数据库的查询压力，并提高了接口 **50%** 的访问速度
- ④ 利用 **Elasticsearch** 结合 **Logstash** 实现 **Mysql** 数据 **全量同步** 至 **Elasticsearch**，并通过 **Elasticsearch** 的 **IK 分词** 索引查询实现音乐数据的检索

获奖情况

校二等奖学金 2021.10

三好学生 2021.10

优秀班干部 2021.10

自我评价

热爱大前端，生命不息，coding 不止