

XXX

XXX/男

XXXXX/信息工程学院（2020-2023）

☎ 133XXXXX675

🏠 <https://uiuing.blog.csdn.net>

🌐 <https://github.com/uiuing>

✉ uiuing@foxmail.com

专业技能

- 熟练使用Vue2、Vue3及其相关生态技术，熟悉React框架，并配合实施SSR、SSG、CSR方案；
- 有参与一线前端UI框架建设的经验，能针对UI进行深入定制化开发；
- 熟悉Rollup、Vite等构建打包工具，对前端工程化有自己的理解与实践；
- 了解Electron、uni-app、Taro跨平台开发与产品迭代；
- 了解性能优化、用户体验优化，通过CI/CD工具管理产品版本迭代；
- 了解多个Server端开发（Node.js、SpringBoot），负责过团队音视频领域、神经网络模型定制与集成开发。

个人优势

- 有重点参与或负责的成熟项目，从需求分析、技术选型、软件开发、到部署运维均有接触，对产品的实现与发展具有积极的主观能动性；
- 对开源有丰富热情，在XXXXX、XXXXX等开源仓库进行过多次贡献，有版本管理与国际化经验；
- 进行过产品基建工作，注重标准化和规范化，并在技术团队内部得到执行；
- 喜欢分享，期待探索新技术，在CSDN/掘金等平台积极分享技术文章，与广大读者共同探讨和学习新技术，在工作团队内进行过技术与开发经验分享。

工作经历

XXXXX科技有限公司 - 全栈开发工程师（2023.08-2024.03）

担任全栈工程师期间，重点参与了WEB前端和小程序开发，同时涉及后端及AI技术集成。致力于提升应用性能和用户体验，加强系统的实时通信和高并发处理能力。成功实现了项目的快速迭代和技术优化，为公司技术发展和产品改进作出贡献。

XXXXX有限公司 - 前端开发工程师（2023.01-2023.07）

先从事WEB前端开发，后升任产品跨平台负责人。主导了产品优化、跨平台落地（小程序/iOS）、以及视频流兼容性解决方案。通过技术重构与性能提升，显著改善了用户体验。引入现代开发流程和技术栈，提高了开发效率和团队协作，加快产品迭代，维持高代码质量。

项目经历

XXX - XXXXX平台（2023.08 - 2024.03）

TypeScript React SpringBoot Taro

负责岗位：WEB前端开发（2023.08-2023.10），小程序产品负责人（2023.10后）

负责内容：WEB前端客户端工作台开发、小程序产品落地。

工作难点：

- 性能优化：** 面临大量图片实时渲染和数据处理的性能瓶颈，需优化首屏加载时间，同时保持页面的交互流畅性；
- 小程序架构设计：** 设计小程序架构时，考虑到平台特性与限制，需精巧地处理图片质量与加载速度的平衡，同时实现高效的数据管理与用户交互；
- 高并发功能开发：** 在保证实时性和稳定性的前提下，开发高并发的图片上传与直播功能，考虑到后端资源的高效利用和网络条件的多样性。

工作成果：

- 首屏加载优化：** 成功应用Virtual Scrolling技术与React的Virtual DOM，实现了动态渲染策略，仅渲染进入用户视觉范围内的图片，显著提升了首屏加载速度（从15秒优化到2-3秒）和页面的交互响应速度；
- 小程序性能提升：** 在小程序端，结合Web Workers对图片数据进行后台预处理和懒加载策略，有效减轻了主线程的计算压力，优化了用户的浏览体验，并实现了快速的照片流程操作，如自动化修图和品质控制，提升了工作效率和内容质量；
- 实时通信与高并发处理：** 负责集成SpringBoot与WebSocket，促进了前后端的高效实时通信。在支撑照片直播功能方面，与团队共同确保了系统对高并发的弹性应对，通过微服务架构和负载均衡提升了数据处理能力；
- 内容管理与用户体验改进：** 设计并实施了一套完整的照片内容管理流程，包括自动审核、AI辅助修图和质量控制，大幅度提高了内容审核的准确性和效率。通过深入分析用户行为和业务需求，优化了用户界面和交互设计，显著提升了用户满意度和平台活跃度。

XXXXXXX - XXXXXX应用（2023.08 - 2024.03）

TypeScript Taro React SpringBoot Python Kafka

负责岗位：全栈开发

负责内容：小程序界面设计与开发、后端主要通信逻辑开发、AI视频生成模型集成与优化。

工作难点：

- 数据同步与实时状态更新：** 开发前后端高效数据同步和实时状态更新机制，针对AI模型执行过程，优化用户等待体验；
- AI模型集成与调用：** 实现AI模型的高效集成和调用，包括输入图片的预处理、执行结果的实时拦截，以及模型功能的HTTP API化，优化模型响应时间和资源消耗；
- 视频内容上传与预览优化：** 面对大空间视频内容的上传和实时预览，优化处理机制，确保用户体验的流畅性。

工作成果：

- **用户界面与上传优化：** 利用Taro和React开发了直观易用的用户界面，尤其针对视频上传和实时预览进行了专门优化，实现分片上传机制，将大视频文件切割成多个小片段并行上传，减少单次传输大小，提高上传成功率和速度；
- **AI视频生成模型集成与优化：** 在SpringBoot框架中集成AI视频生成模型，使用Python进行模型调用流程的优化；对输入图片执行预处理操作，包括尺寸调整和格式转换，以适配模型要求，提高处理效率；结果实时拦截和格式化处理，确保输出数据的一致性和可用性。通过RESTful API封装模型服务，提高了服务的可访问性和灵活性；
- **视频生成流程性能优化：** 对AI视频生成流程进行了深入的性能优化，包括逻辑的精简和算法的调整，显著减少了视频处理所需时间。同时，通过SpringBoot封装的RESTful API，实现了模型服务的高效调用；
- **实时通信与任务管理：** 开发基于WebSocket的实时通信机制，实现视频处理进度的实时反馈给前端用户界面，引入Kafka消息队列处理视频生成任务的排队和状态管理，通过优化算法和任务调度策略，实现高并发处理和资源的有效分配。

XXXX - XXXX管理平台（2023.01 - 2023.07）

TypeScript uni-app React Vue Node.js

负责岗位： WEB前端开发（2023.01-2023.04），产品跨平台负责人（2023.04后）

负责内容： WEB产品优化、产品跨平台落地（微信小程序/iOS）、WebView通用模块开发、直播视频流中转站开发。

工作难点：

- **代码维护难题：** 需要维护错误混合了TypeScript和JavaScript、效率低下且难以管理的WEB客户端，该客户端经过多年多人迭代，存在维护难、性能低效的问题；
- **GIS SDK集成挑战：** 处理多个第三方GIS SDK的离线和在线引用，以及线上体验的差异问题；
- **资源与团队限制下的快速开发：** 在研发资源有限，和团队成员水平不均的情况下，需要快速开发出微信小程序和iOS客户端；
- **视频流格式兼容问题：** 解决iOS设备不支持原FLV格式视频流的问题，尤其是在设备推流服务商不支持格式定制的情况下。

工作成果：

- **WEB基础设施升级：** 采用React18与Webpack5升级WEB基础设施，引入thread-loader、tree-shaking等性能优化措施，显著提升了运行与热更新速度（初始启动时间从1分43秒缩短至20秒，热更新时间从6秒缩短至2秒内）；
- **业务逻辑与代码质量提升：** 核心业务逻辑用TypeScript重构，复杂计算函数采用JSDoc约束，结合Jenkins实施CI/CD，提高了开发效率和业务逻辑的迭代速度，加速了发布周期；
- **跨平台开发与团队协作优化：** 通过uni-app快速开发小程序与iOS客户端的通用功能，核心GIS功能通过独立的Vue项目开发，并整合至WebView中。采用pnpm workspace进行monorepo管理，ESLint和Stylelint保证代码质量，实现了高效的团队协作和快速的产品迭代（初版一月内完成测试并上线）；
- **视频流兼容性解决方案：** 构建了基于nginx的RTMP流媒体服务，增加HLS直播流配置，并通过Node.js服务管理设备指令及FFmpeg转流，实现了原FLV流的iOS设备兼容性，无延迟增加的同时保证了性能。

XXXXX - XXXXUI框架 (2022.08 - 2022.10)

TypeScript React

负责岗位： Feature开发

负责内容： 负责Typography和Progress组件的特性开发和，专注于提升组件用户体验，挖掘组件的潜力。

工作难点：

- **代码质量维护：** 确保Pull Request的影响最小化，同时保持代码的简洁、可读性和性能稳定性；
- **渲染过程定制：** 通过拦截和定制virtual DOM的渲染过程，实现渲染结果的灵活调整，包括节点内容的变更及其样式和格式的统一计算，满足复杂界面的渲染需求；
- **组件兼容性维护：** 维护组件props的兼容性，确保内部逻辑在异常情况下容易追踪，快速定位和解决问题。

工作成果：

- **代码规范实施：** 在monorepo架构下，采用Stylelint、ESLint和Prettier工具约束代码风格，并通过Jest进行单元测试。在代码提交阶段引入Git Hooks，确保代码一致性和符合质量标准；
- **Typography组件优化：** 对Typography组件内实现了一个数值处理特性，采用深度优先搜索（DFS）算法，优化了virtual DOM的渲染逻辑。通过细致的DOM结构调整和节点优化，显著减少了渲染时间并优化了内存使用，确保即便在数据密集的大屏应用中也能实现快速渲染（不超过50毫秒）；
- **Progress组件功能增强：** 为Progress组件引入了颜色和渐变功能，通过动态样式计算和CSS预处理技术，增强了视觉效果丰富性和自然感。此改进不仅提升了用户界面的吸引力，也为开发者提供了丰富的UI定制选项；

其他

个人相关作品

- **作品名称：** VARBook - 适合中文程序员的变量命名助手
代码仓库：github.com/uiuing/varbook
- **作品名称：** InGop - 一站式学习Go+语言解决方案
代码仓库：github.com/uiuing/ingop
- **作品名称：** 视障资讯-无障碍新闻资讯平台
代码仓库：<https://github.com/uiuing/visual-impairment-information>
- **作品名称：** 宝贝回家 404 - 走失儿童大数据信息传播平台
代码仓库：<https://github.com/uiuing/Findbaby>

荣誉证书

Alibaba Cloud Certified Professional - Data Analyst, HUAWEI HCIP-BigData Developer v2.0 , 2023年泛珠三角+大学生计算机作品赛三等奖, 阿里云“乘风者计划”专家博客, 工业互联网平台开发工程师中级, 第十四届蓝桥杯Web应用开发大学组省一等奖