

个人信息

- 翟浩然/男/1992
- 中国农业大学(985)/本科
- 工作年限：4.5年
- Github： <https://github.com/zhaihaoran>

联系方式

- 手机：13552232913
- Email：13552232913@163.com
- 微信号：zhai359118915

工作经历

滴滴出行有限公司 2018.6 ~~ 至今

网约车事业部 - 治理平台用户体验组-安全方向 高级研发工程师

工作亮点

- 19年绩效A、小组副leader，虚线管理2人小团队。
- 由于作为治理平台唯一的前端团队，横向支撑各个方向业务。同时作为前端团队绝对主力，横跨参与多个核心项目，覆盖多种场景。具体包含了
 - 复杂表单的，业务场景复杂、交互繁多的各类业务中台系统；
 - Hybrid应用中webApp需求场景，具备联调、优化能力；
 - 前端组件脚手架、组件工具库，SSR文档站点研发
 - 基于node后端server服务，提供中间层服务
 - 内部工具市场研发（基于业务组件场景下的抽离方案）
- 熟练掌握规范化流程操作，熟练接入公司各项内部技术平台。包含了部署、编译、埋点统计、错误上报、链路查询、域名接入、文档管理等等，经验丰富，大型公司上手零成本
- 投身于技术基础建设中，有大量理论及实践经验
- 有稳定性建设相关实践经验。有前端监控实践经验

项目介绍

明镜（安全处置一体化平台）

职能主要帮助多类二线客服，提供安全方面工单响应、处置、流转的能力，项目特点主要为多场景、多流程、多客服交互的，整体表单结构复杂，大量多级联动、动态表单信息，同时项目迭代较长，历史包袱重。

个人主导团队对webpack进行升级优化、提升响应编译速度、热环境速度。研发配置化表单方案，降低复杂度，提升效率。同时支持RD控制渲染的能力。

组件库研发、静态站点

组件库主要为各个常用业务组件进行npm包发布能力支持，能力主要包括编译、热更新、测速、发布，编译能力包含单组件发布编译和多组件发布编译，支持babel-plugin-import异步加载。静态站点基于nextjs实现同构渲染的能力，主要用于组件文档、demo的展示。

作为发起人，独自设计实现组件库核心功能和框架，降低维护发布成本。带领团队提升大组研发效率。提升组件复用率，扩大影响力。

天眼server端

天眼是一套帮助客服同学还原复现问题场景、流量回放的流程，本身包含研发平台、devtool、服务端三部分。个人主要负责设计并实现服务端，提供数据上报，信息查询等基础后端接口服务，包含了基础的权限认证、日志切割归集统计、数据存储服务。项目整体技术栈为egg + typescript + sequelize + mysql，对node后端服务有实践经验。

脚手架

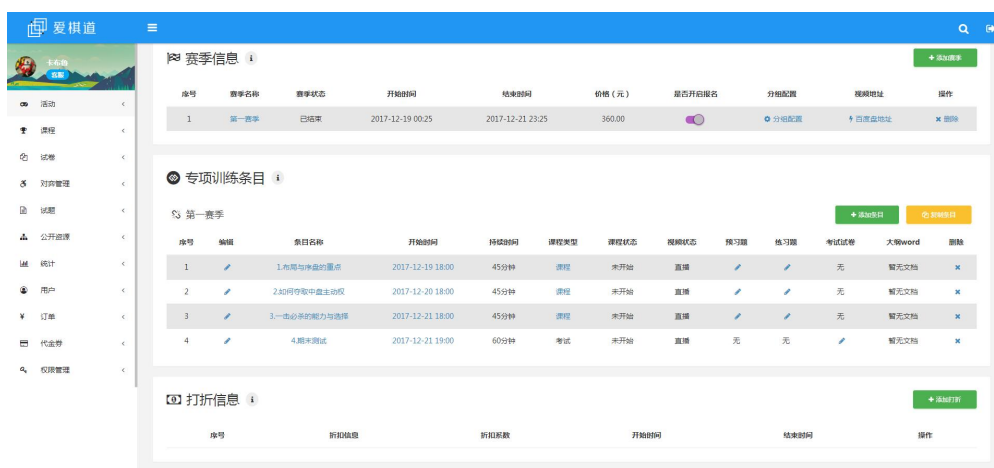
类cra,主要目的帮助团队提供成熟的项目模板，支撑业务快速迭代，类型包含了vue、react、pc、mobile、node等各类场景。个人主要参与其中功能流程设计，部分模板的研发工作。

途梦公益教育 2018.2 ~~ 2018.6

途梦前端整体架构搭建、实现

- 独立使用ES6 + bootstrap4 + webpack + scss + ajax + smarty 设计主站前端架构和实现，功能集成了分页、视频播放、百度分享等功能，兼容性为IE9+,firefox,chrome,页面均为响应式，基于淘宝响应式设计，SEO表现良好。主站代码风格上命名规范、语义化、注释清晰。
- 基于vue-cli基础上使用 vue + vue-router + vuex + ElementUI 及周边组件独立搭建、实现嘉宾、学校、运营三个后台前端部分。功能集成增删改查、视频点播、聊天（留言）等功能。
- 技术特色上，通过配置webpack中commonchunks、dllReference、external 分离第三方组件库以提升打包性能，extract实现样式分离；配置alias、proxy反向代理，通过apimock平台模拟数据接口，实现前后端分离。提升开发效率。
- 组件方面，使用配置vue-router懒加载进一步减少打包体积，提升第一次加载性能，axios统一封装后调用，使用mixin、filters抽取重复方法、避免模板使用复杂逻辑，vuex按module划分，层次清晰，易于维护和调整。

爱棋道文化传播有限公司 2016.2~~2018.1



工作内容：

负责公司运营后台前端重构、业务功能开发；公司活动落地页设计与实现；研发教师上课工具；

- 运营后台主要面向所有运营、教务、客服等工作人员进行日常维护，业务涵盖了公司所有产品相关的数据支持，包括课程、活动、用户、试卷、题库等等方面。具体功能点包含了基本的信息展示、增删该查、权限管理、数据可视化、数据导出、视频文档和课件上传、新闻专栏富文本编辑等。
- 渲染方式为后端服务器渲染+传统多页面的研发方案，技术稳定且成本较低，易于迭代。核心技术栈为 RequireJs + ES5 + Gulp + Less + Documentation。
- 前端整体基于AMD模块化开发,多页面通用组件统一声明,自制工具类util,二次封装组件等工作以支持页面js的工作,抽象度较高,逻辑结构清晰。
- 后端方面技术栈为 golang + iris + mysql,熟练使用go语言进行路由编写，部分接口编写；

hexo博客 -- 卡布鲁 | 杂食堆
