个人信息

1992/05/07

18609669205

hxtheone@gmail.com

github.com/MrHuxu

xhu.me

教育经历

计算机科学学士学位
2010/09 - 2014/06, 哈尔滨工业大学

技术评价

- HTML & CSS
- React.js & Redux
- Node.js
- Ruby on Rails
- Go
- Docker

胡　旭

高级开发工程师

工作经历

高级开发工程师 - FreeWheel

2014/07 至今

在 UI 团队从事全栈开发的职位.

· 在 Ruby on Rails 和 jQuery 的基础上对公司 UI 项目进行持续开发和维护.

· 使用公司内部的 SparkUI 完整重构一个项目的前端代码.

· 深入挖掘工作痛点, 通过开发各种自动化工具, 提升团队的工作效率.

· 在纽约出差三个月, 跟美国工程师进行技术交流, 并且为 Service Team 提供技术支持.

· 熟练使用 Go 重写现有项目, 并且在团队使用 SOA 架构之后独立负责一个公共基础模块的研发.

项目经验

Workflow Service - 为公司业务提供流程控制的基础服务组件

Go · gRPC

· 独立完成项目的技术 design, 注重与服务之间的解耦, 并且适当暴露接口以便于优化和扩展.

· 代码采用清晰的分层结构, 下层方法通过接口被上层调用, 适当解耦的同时也便于编写单元测试.

· 每一层都贯彻 '单一职责原则', 提高方法的可复用性.

· 代码的单元覆盖率达到 85% 以上

HyLDA - 线性广告和数字广告混合投放系统的 UI 部分.

Ruby on Rails · jQuery · Redis · MySQL

性能问题:

· 采用 Activerecord 中的 eager load 操作避免了加载数据时可能出现的 n+1 问题

· 将部分非常耗时的操作使用 Resque 的任务队列进行管理, 提升页面的整体响应效率

· 给数据库中常用的外键搜索字段加上索引, 使查询升级操作由表锁变为行锁, 提升项目的并发性能

质量控制:

· 完善的 Tech Design, Case Design, 以及充足的单元测试和自动化测试流程

· 建立完善的监控机制: 使用 RoR 的 hook 机制无侵入的实现自定义 log, 学习使用 ELK 建立了相应的 dashboard 以可视化用户的使用情况

SparkUI Adoption - 带领一个团队使用内部的 SparkUI 框架对 HyLDA 模块进行完整的前端替换.

React.js · Redux

项目设计:

· 项目初期进行详细的 design, 特别是和后端的 API 设计, 在保证功能的情况下尽可能使用统一格式, 方便今后的后端升级.

· 项目初期确定测试框架: 采用单元测试(mocha)测试数据 + 行为测试(cucumber)测试页面组件的方案, 避免在测试方式上出现重复.

· 使用 Agouti 编写了一个小型的性能测试工具, 并且使用 ECharts 可视化对比替换前后测试结果, 确保了产品的稳定上线.

代码重构:

· 使用柯里化的方式编写功能类似的函数的 generator, 避免了大量重复代码的产生.

· 优化数据模型结构, 使得组件之间的耦合变小, 通过给组件自定义 shouldComponentUpdate 方法避免页面大规模重绘.

· 在2的基础上, 把大的数据模型拆分成小的文件, 在使用 Redux 框架的前提下, 每个文件专注于一个功能的 action/actionType/reducer, 方便后续维护和升级.

Bug Bash Tool - 内部的一个统计特定 Jira ticket 的工具.

Ruby on Rails · React.js · Redux · Express.js · Go · MongoDB

· 初始版本使用 Ruby on Rails 和 React.js 实现了完整的前后端分离开发.

· 将后端使用 Node.js 进行了重写, 实现数据的并行获取, 了解了 ES6 原生的 Promise 规范, 并对前端代码进行了部分重构.

· 在 Jira 速度太慢的前提下, 采用 redis 作为数据缓存, 提升了页面整体响应速度.

· 使用自己写的脚手架用 Go 重写后端代码, 使用 MongoDB 持久化数据.