

个人信息

姓名: 邵梓阳 年龄: 21 手机: 15708406604 邮箱: 1666930632@qq.com

求职方向: 前端开发 英语等级: 通过CET-6

教育经历: 2020/09 --2024/06 西南交通大学 | 本科

技术栈

- 熟悉JavaScript, ES6+, 如闭包、事件循环、Proxy、Promise等, 熟悉TypeScript
- 熟悉计算机网络, HTTP网络协议, 操作系统等计算机基础
- 熟悉数据结构与算法, 线性表, 树, 图, 堆, 递归, 排序, 查找, 动态规划, 贪心等
- 熟悉React、Redux, 了解React Fiber架构及其执行原理
- 熟悉TailwindCSS、UnoCSS等原子化CSS框架
- 熟悉Vite工具链, pnpm包管理器, 了解Vitest单元测试框架
- 了解性能优化, 如防抖, 节流, 懒加载
- 了解Next.js, SSG, SSR, ISR
- 了解Vue、Composition Api、Reactivity原理
- 后端了解node.js、Java、MySQL, 能编写基本的RESTful API

项目经历

react-routerfly

- 地址:
在线playground: friendly-paletas-0b8746.netlify.app
github: <https://github.com/Frankcaozas/react-routerfly>
npm: <https://www.npmjs.com/package/react-routerfly?activeTab=readme>
- 项目介绍: 一个实现了在React项目中实现跨路由动画的组件库, 支持定位布局, 也支持流式布局。技术栈包括React+TypeScript。
- 项目亮点:
 1. 使用代理组件获取并维护真实组件的位置, 大小, props信息, 并将组件实例传递给RouterFlyCraft, 将真实组件提升至根组件
 2. "起飞","降落"的过程中保存组件的状态。因为使用闭包维护一个Map(单例), 会导致React每次都重新渲染组件 (React更新需要新的数据)。于是封装了keepalive组件缓存组件状态(借助createPortal实现)
 3. 借助onTransitionEnd事件确定使用createPortal"降落"的时机
 4. 借助useLayoutEffect和getBoundingClientRect在画面渲染前更新组件布局
 5. 快速部署: 使用Netlify绑定Github仓库自动部署
 6. 优化开发者使用体验。
 1. 根据收集到的组件信息自动创建RouterFlyCraft(负责"起飞","降落"的组件), 避免重复手动创建
 2. 可配置动画时间, keepalive是否开启

mini-frameworks

- 地址: <https://github.com/Frankcaozas/mini-frameworks>
- 项目介绍: React, Vue/Reactivity的迷你版实现。技术栈为TypeScript, Vitest, pnpm
- 项目亮点:
 1. Vue/Reactivity:
 1. 实现reactivity函数时需要根据对象是普通对象还是集合实现不同的handler。
 2. 依赖与effect双向链接实现响应式依赖清理, 清理时set.forEach()中delete和add会出现死循环, 需要复制set。
 3. 用栈实现effect嵌套。
 2. React:
 1. 用Fiber数据结构和requestIdleCallback实现concurrent mode
 2. 支持函数组件和state
 3. pnpm管理monorepo
 4. 使用Vitest测试框架效果, 保证代码质量