

基本信息

- 个人信息：陈世松 | 男 | 1994年
- 学历信息：成都理工大学 | 全日制 本科 | 通信工程
- 工作经验：前端开发 3年+ (2017.2至今)
- 期望职位：前端工程师(中高级) | 成都 | 全职 | 薪资面议
- 联系方式：15608003358 (微信号同该手机号) | 448216252@qq.com

掌握技能

- 扎实的前端开发基础 HTML/CSS/JavaScript(ES5/ES6+)，能够运用设计模式写出结构良好的现代代码。
- 善用 Vue.js 构建高可用、高质量的SPA项目，对其原理有一定深入的了解，知道大部分功能的实现方式。
- 熟悉 小程序 开发及小程序相关优化，善用 uni-app 构建性能良好的 多端小程序 项目。
- 熟悉 webpack，对前端工程化、模块化有一定的理解，有一定改良公司内部脚手架的经验。
- 熟悉 dart，能够使用 Flutter 进行跨终端开发。
- 熟悉 React 及相关生态和部分功能原理，能够使用其进行业务开发。

工作经历

成都虎魄网络科技有限公司(2018.6至今 | 增长组前端负责人)

前端基础建设

企业内部 npm 依赖库(2019.6)

- 技术架构：nexus 作为包管理平台，Monorepo 项目管理模式，lerna 作为包管理工具，@pika/pack 作为打包工具，npm 域 控制私有包的拉取和上传
- 根据业务和技术积累，依赖库主要分成三大类：
 1. 核心库(@hupo/core): 核心工具方法，包含小程序框架核心依赖、常用方法、通信层封装、多端多态协议封装，代码质量得到提升。<https://github.com/ChenXiaoSong12321/hupo-core>
 2. 组件库(@hupo/ui): 商城项目内部UI库，集中化管理，迭代效率得到优化，UI设计的规范化得到显著提升。<https://github.com/ChenXiaoSong12321/hupo-ui>
 3. 编译库(@hupo/compile): 覆盖编译类工具方法，通过配置全局命令实现快速创建基础项目，拉取接口资源等功能，大幅度提高开发人员工作效率。

多页面高性能脚手架(2018.10)

- 主要特点：
 1. 多项目/多页面，可快速启动一个活动页 H5 项目。
 2. 已有cdn资源使用cdn链接，其他打包资源上传阿里oss，充分利用cdn，首次加载速度和二次加载速度得到极大提升。
 3. 通过命令行 快速部署 项目。
- Github: <https://github.com/ChenXiaoSong12321/vue-multiplePages-useCdn>

多端UI组件库(2018.12)

- 主要贡献：协助并结合设计师的UI组件规范，实现了内部小程序组件库大部分组件，后迁移至chameleon和uni-app多端框架。



- Github: <https://github.com/ChenXiaoSong12321/hupo-ui>

主要涉及项目

- 琥珀亲子小程序(gulp+原生 -> chameleon -> uni-app)
 1. 梳理和持续优化维护线上系统, 完成各项需求的前端落地和优化。
 2. 基于 **多端框架** 的部分架构设计和业务(2.0->3.0)迁移, 完成相关的依赖库、组件库的开发。
- 琥珀亲子后台管理系统(Vue + Element)
 1. 基于选型框架并根据项目需求调整和框架优化, 解决标签页缓存等技术难点。
 2. 实现并利用 **dispatch** / **broadcast** 增强Vue的通信方式, 优化表单验证。
 3. 利用 **\$attrs** / **\$listeners** 对Element部分组件进行二次封装, 提升开发效率。
- 商家端
 1. 基于H5完成1.0商家端的业务开发, 完成日历统计等核心功能。
 2. 基于 **Flutter** 完成商家端跨终端(iOS/Android)开发。
- 相关 **H5** 营销活动项目的快速开发上线。
- 制定并持续优化 **代码规范**、**Git提交规范**、**版本号规范** 等团队规范化建设。
- 基于 **Vue** + **Vuepress** 完成文档系统的搭建和模版设计。

成都世纪顶点科技有限公司(2017年2月 ~ 2018年5月)

主要涉及项目

顶点 IPsec VPN 安全网关系统

1. 利用 **JavaScript(ES5)** 面向对象编程, 开发了列表详情等核心模块, 提升新增页面时的效率。
2. 通过 **gulp** 构建, 引入 **ES6+** 与 **sass**, 对系统部分复杂业务的界面进行模块拆分重构, 异步代码采用 **Promise化**, 实现模块化, 代码复用性和可维护性得到提升。
3. 利用 **perl** 进行作为中间件进行前后端联调, 开发上传、下载、数据传输等功能。
4. 利用 **ECharts**, 实现基本系统页面效果与交互、数据可视化等功能。