

# 个人信息

- 基本信息：陈世松 | 男 | 1994
- 学历信息：成都理工大学 | 全日制本科 | 通信工程
- 联系方式：15608003358(微信号同该手机号) | 448216252@qq.com
- 工作年限：3年+(2017.2至今)
- 期望职位：前端工程师(中高级) | 成都 | 全职 | 13-18k
- Github: <https://github.com/ChenXiaoSong12321>

## 掌握技能

- 熟练使用 JavaScript(ES6+)，熟悉 JavaScript 面向对象及函数式编程，能够运用设计模式写出结构良好的现代代码。
- 熟练使用 Vue 全家桶及相关生态，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，熟悉 uni-app、chameleon 等小程序多端方案。
- 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

- 主要业务：亲子OTA(Online Travel Agency)平台，涉及游玩、剧场、酒店、餐饮等场景及其他OTA平台的对接产品。
- 2018年9月 ~ 2020年4月 | 增长组前端负责人

### 前端基础建设

#### 企业内部 npm 依赖库(2019.6)

- 问题：随着业务项目的推进，前端项目数量越来越多，通用性代码大多通过复制粘贴的形式，代码变得繁杂臃肿且不易维护，需找到合适的代码管理方式。
- 任务：建立企业内部私有npm依赖库
- 分析：根据目前的困境，将需要重复利用和管理的代码主要分成三类，编译库，组件库，以及核心库。
- 结果：根据在 npm文档、github、掘金等资源进行调研，结合公司内部资源，采用 nexus 作为包管理平台，lerna 作为 编译库(compile)、组件库(UI)、核心库(core) 三个核心包的管理工具，通过 npm域 来控制私有包的拉取和上传，大幅度增加开发人员工作效率和代码复用性，代码管理得到显著提高。

#### ■ 说明

- 编译库(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。
- 组件库(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>
- 核心库(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, 做到一处修改, 各端更新。 <https://github.com/ChenXiaoSong12321/hupo-core>

#### 小程序UI组件库(2018.12)

技术架构: 基于原生小程序 -> 基于 chameleon 框架 -> 基于 uni-app框架

- 主要贡献: 根据借鉴 vant、Element、mint及其他优秀类库, 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, 考虑在 npm 上进行开源, 但后续由于人力、框架及私有依赖库的建立没有在开源的路上继续维护, 已支持的端有微信小程序、支付宝小程序、微信内外H5、头条小程序。



#### 基于 vue-cli2 的 vue 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): <https://github.com/ChenXiaoSong12321/vue-multiplePages-useCdn>

#### 主要涉及项目



## 琥珀亲子小程序

技术架构：gulp+原生(目前线上) -> chameleon框架(过渡) -> uni-app框架(内测)

### ■ 主要贡献：

1. 基本业务逻辑及营销活动的前端支撑。
2. 梳理和持续优化维护线上系统。
3. 基于多端框架的部分架构设计和业务迁移。

## 琥珀亲子小程序后台管理系统

1. 基于选型框架并根据项目需求调整和框架优化。
2. 利用 `dispatch/broadcast` 增强Vue的通信方式，优化表单验证。
3. 利用 `$attrs/$listeners` `provide/inject` 对 Element 部分组件进行二次封装，提升开发效率。