

**L**-star

https://github.com/bilibiliou

http://bilibiliou.github.io/

**\** 18824588282



前端工程师 (正式工作时长不满两年)



# 教育经历及亮点

**EDUCATION&BRIGHT SPOT** 

2014 ~ 2018

# 黑龙江大学

- 彦 信息管理与信息系统 (全日制本科)
- ▼ 5年技术博文积累
- ▼ github 独立项目"animations 个人前端动画研究集" 89star 29fork (喜欢还望点个star)
- ▼ 早年抱过大漠老师大腿:参与翻译了开源书籍 O'REILLY 《SVG ANIMATIONS》 (已出版,独立翻译前言部分+1、2、9、13、17章内容) 开源地址

#### 团队提效:

#### 独立开发 zhuiyi-analytics

- 1. 实现非侵入式或低侵入式埋点,通过Vue自定义指令、AOC切面等手段提供包括手动埋点、声明式埋点、面向切面埋点等方式供团队开发者使用
- 2. 以类比Vue插件化的方式解决痛点:由于公司各业务线使用的三方埋点平台(如神策、百度统计)不一致,zhuiyi-analytics 将埋点收集到的数据外抛给插件,各业务线只需要开发自己的插件就能完成埋点代码统一,如后续平台到期了,换平台的话,只需要换插件即可
- 3. 统一公司埋点数据字段规范, 明确各类数据由那方进行收集
- 4. 愿景:希望打造公司内部的埋点平台(如腾讯罗盘),能够支持埋点数据分析、错误上报、资源测速、pv, uv统计、页面访问阈值告警、自动埋点、页面热力图生成,帮助降低兄弟团队业务维护成本

#### 维护 zhuiyi-ui

- 1. zhuiyi-ui 基于elementUI 二次开发的UI代码库,为满足业务需求,开发或者修改现有的ElementUI库中的组件并修改Element文档
- 2. 修改代码库脚本,对业务侧输出scss文件,业务侧可针对设计稿风格scss再次定义组件样式(不是我做的,但是我读过源码)
- 3. 为降低代码侵入:业务逻辑使用 mixin 对js逻辑进行合并,样式则采用.el-extra-xxx方式进行覆盖(不是我做的,但是我读过源码)

# 团队影响力:

- 1. 独自设计一种前缀逻辑表达式,及其相关解析和抽象方法,空间复杂度相较于JSON 更轻量(正在走公司专利申请流程)
- 2. 团队分享《前端监控体系建立》和《浅谈交互动画设计》(包括线性插值和间补动画、音频可视化等) PPT 归档

# 团队建设:

- 1. 带过一名实习生,他最后成功拿到Offer
- 2. 维护团队业务文档归档沉淀



# 项目与工作经验

EXPERIENCE

□ 追一科技

2018.07.02~至今

#### Call外呼智能机器人项目组(ToB)

- 🌺 1. 从立项开始从0到1迭代Call外呼机器人项目前端项目,至今项目已在多家企业中正式投产,能够独立承担业务开发工作
- 🌺 2. 维护过多家银行、券商、保险的多个定制化或POC项目多条分支代码和标品主支的合并,并掉了很多头发
- ❖ 3. 处理过前端浏览器处理过较大数据量并发导入的情况(前端解析10W条Excel数据并展示)

2020/4/19 欧阳湘粤 - 个人简历

◆ 4. 独立设计了一种特殊的前缀逻辑表达式及其解析和抽象方法(一种更轻量化的数据结构),并运用到项目中降低了项目中的空间复杂度

- ❖ 5. OEM腾讯中信银行-静默坐席系统, 团队协作完成, 个人独立负责核心工作台模块(目前已经投产)
- ✿ 6. 由于call业务实时性要求较高,项目websocket相关知识及其前端侧接口层设计沉淀博文-- 《前端侧接口层设计》

#### 开发独立内部业务系统

- ❖ 独立开发前后端,默默地给算法组打杂
- ◆ 包括简单的增删改查(用的Sequelize),定时任务,上传文件(支持大文件切片上传、断点续传)

# 独立完成Aiforce官网

♠ 官网网址 Aiforce官网 兼容移动端

■ 腾讯科技IVWeb团队

2017.07.12 ~ 2017.10.20

# 手Q内外渠道相关业务 (实习)

- ✿ 维护、开发内外渠道相关业务
- ♥ 学习部署服务端直出渲染+部署压测服务
- 🌣 维护、开发活动管理内部系统,管理系统实现通过JSONSchma配置,可以自动产出相关活动页,提高产能
- ❖ 学习内部活动系统后台搭建+配置Nginx
- ♥ 学习浏览器相关调试技巧

直 百度金融(FSG)

2017.02.20 ~ 2017.06.01

#### 内部策略平台开发(实习)

- 蜂 复习reactJs + react-router + redux + webpack 学习接触 Ant-desgin 沉淀了关于洋葱模型的理解
- ♪ 阅读flex-box 规范,深入了解flex-box的计算方式、FFC、以及其他细节,并总结博文
- ♪ 负责权限控制模块、效果报告模块的前端开发,准备接手离职同学实验模块
- ♥ 学习git 分支管理, 尝试多人配合开发
- ◆ 学习配置Webpack(建立DII库,搭建Webpack-dev-server,配置转发代理,Chunks分块按需加载,使用各种plugin进行优化)

✓ 个人项目% 源代码

### 阔论--即时评论系统

- ☆ 阔论 免注册评论系统
- ◆ 使用flex布局 + Es6 + react + redux技术
- ◆ 支持评论代码高亮,发送表情,**流光标题**,图片上传(可拖拽)以及 **自拍上传**(使用WebRtc技术调用摄像头进行拍照),**动态表格创建**
- ✿ 独自一人完成**前后端**所有模块
- nodeJs 做后台服务,基于mongodb进行数据存储
- ✿ 使用js-xss做了防Xss处理
- ✿ 伪造CA创建HTTPS连接
- star 28 folk 5

┩ 个人项目% 源代码

#### 我研究的Animations (Svg + canvas)

- ♠ README 上有codepen demo link
- star 89 fork 29

✔ 个人项目

# JQ 插件: 轻量级博客音乐播放器

- ◆ 一个轻量级的JQ 博客音乐播放器
- ☆ 扁平化设计
- ✿ 圆形进度条

┩ 个人项目% 源代码

#### Fox-Listener

- npm i fox-listener
- ❖ 基于Es6语法实现的 发布/订阅 模式 小工具



# 技能

SKILLS

#### → Web 前端

# HTML / CSS

- ★ 能够编写语义化的 HTML,模块化的 CSS,实现较复杂的布局与动画
- ★ 了解已标准化的 HTML5 / CSS3 新特性, 了解例如css clip-Path 之类的新特性
- ★ 关注w3cplus 实验项目中能尝试使用未广泛实现的新标准

# **ECMAScript (JavaScript)**

- ★ 了解 ECMAScript 新标准与相关后处理工具并尝试项目中使用
- ★ 关注MDN Es7 新规范

#### DOM / BOM 及常见 Web API

★ 了解原生 DOM 与 BOM 的使用及相关标准 (W3C / WHATWG)

#### Canvas / SVG

- ★ 关注CodePen 以及 W3cPlus 动画效果, 并愿意了解源代码
- ★ 对营销类的动画效果有一定追求

#### 前端工程实践

- ★ 在项目中使用过 CommonJS / WebPack 等多种模块化解决方案
- ★ 在项目中使用过 Less / Sass / babel 等预处理/后处理工具
- ★ 掌握基本的前端兼容性处理、性能优化知识与技巧,并能在项目中加以实践
- ★ 代码健壮性! 可读可扩展性! 业务逻辑解耦
- ★ 沉淀过前端接口层和websocket通信设计,并总结在博客

# ● Web 后端

# Node.js (不深入)

- ★ 能够使用 Node.js API / Egg.js 搭建简单的后端程序与数据库交互、渲染模板及提供 RESTful API
- ★ 掌握常见的 4种 异步编程 解决方案
- ★ 能编写简单的repl脚本
- ★ 了解和写过puppeteer的Demo,知道是拿来做自动化测试和爬虫脚本的
- ★ 写了个小脚本,通过处理图片灰度值,将图片进行字符化
- ★ 写了个小脚本: 一行代码画一爱心

# ♥ 其他

- ★ 一定的 数据结构 / 软件工程 / 计算机网络 相关知识
- ★ VsCode 使用者
- ★ git基本操作:维护过多个定制化分支迭代和与master的合并,能够基于gitflow workflow维护好多个分支的迭代
- ★ 了解Unix/Linux 基础使用,并不精通地使用Xshell
- ★ 个人缺点:涉猎很多但是不算精通,未来能够找到一个方向,进行深入 | 有些时候自己有点小能耐就想显摆