

个人信息

姓名：陈欣聪（男）
工作年限：10年以上

出生年月：1990.11
职位：web全栈工程师

联系电话：18116302795
邮箱：STLCXC@163.com

项目经验

普华永道信息技术（上海）有限公司

• PwC Workforce Orchestrator（2019/03-2023/6）

通过提供促进果断行动的定制化数字体验，提升团队生产力与参与度。Workforce Orchestrator 作为智能数字平台，可实时无缝连接人员、信息与应用，营造更高效、动态且具参与感的工作环境。借助移动参与平台，将广泛的沟通转化为有意义的一对一对话，确保员工专注于工作并保持高效。您能向不同员工群体发送个性化消息，同时在统一应用中轻松管理任务与关键活动。此外，通过预先编程的高效自动化聊天机器人，即时响应常见咨询与请求。通过脉冲调查持续收集反馈，洞察实时员工情绪，以便及时调整。另外，在技术部署、组织变革和战略计划等各个阶段，提供清晰且可执行的沟通，为员工提供支持。这种全方位的方法不仅赋予员工能力，还能在组织内培育持续改进的文化。

- 技术栈：Angular,graphql,nodejs,express,typescript,gcp-firebase,gcp-pub-sub,postgresql,google-dialogflow,redis
- 协助修复前端bug
- 使用firebase，实现分布式的服务端推送系统
- 使用pubsub，express和graphql，处理各种后端异步数据处理
- 使用dialogflow，实现智能客服，自然语言处理与action的绑定，在teams以及短信以及网页版实现智能交互

• Model Edge（2023/06-2024/6）

更优的模型治理与验证模式。Model Edge 支持端到端的模型生命周期管理，同时可在单一平台中实现全量模型组合（含 AI 模型）的管理、开发、验证及治理。通过提供必要工具以向内部及外部利益相关方展示模型有效性（及可解释性），Model Edge 简化操作流程，助力您增强对模型项目的信心。

- 技术栈：Angular,graphql,nodejs,express,typescript,gcp-pub-sub,postgresql,redis
- 协助修复前端bug
- 参与设计开发了，问卷创建和答题系统，以及答题后的计算风险逻辑

• 内部poc项目ai flow ceartor（2024/07-2025/05）

基于flowise的ai低代码平台，集成了公司内部系统，以及定制化了前端ai和一些组件，可以基于这个平台轻松构建ai机器人。

- 技术栈：React,nodejs,express,typescript,long-chain,redis
- 参与设计开发了与公司系统的集成，如用户单点登录，用户资源的绑定分组，以及一些组件的内部化后端的ai-component的内部化，以及runtime library。前端导出可用于runtime的结构，以及聊天组件一些插件支持，比如让ai的回答支持图标（plotly.js）

- **Concourse (2024/07-2025/05)**

一个让复杂交互变得简化、智能且透明的交付平台。

- 技术栈：React,nodejs,nest.js,typescript,graphql,redis
- 后端维护各种数据的crud的api
- 前端参与了主页设计和开发
- 编写一些脚本用来维护或者更新一些客户数据

上海秀店网络科技有限公司

1. 服务端

- **某电视台app (2018/10-2019/2)**

包含电视节目查看提醒·商城·玩游戏·抢红包等一系列功能的app

- 技术栈：nodejs,egg.js,typescript,redis
- app对接阶段·对于接口规范参与了讨论设计·统一了整个客户端接口标准
- 开发了锦鲤活动全部逻辑·在设计上对于消息队列的合理运用·解耦了app原生逻辑与活动逻辑的耦合
- 对于抽奖逻辑·开发了一套奖品分发逻辑·用于正太分布奖品
- 在redis的运用上：token存入redis实现了session共享·以及避免jwt对于发放时间不可中途把控的缺点；通过sortedset实现了最常玩的游戏;通过过期key实现了定时任务。
- 参与了阿里云互动直播邀测·设计了部分直播逻辑原型

- **秀店2.0 (2018/8-2018/10)**

电商平台·包括商品·库存等(我参与了推广部分·合伙人规则制定以及分润结算)

- 技术栈：nodejs,egg.js,typescript,redis
- 推广模块app端(对接安卓ios需求出接口)
- 以及后台接口开发(参与一些业务逻辑设计开发)

上海金桥信息股份有限公司

1. 服务端

- **会议平台系统 (2016/10-2018/8)**

基于第三方音视频sdk的一套会议系统。能够建会、约会、控会、直播、录播的会议的基础产品。衍生了教育行业-公开课直播、法院行业-执行值守等定制产品。

- 技术栈：nodejs,express,backbone.js,阿里云,mongodb
- 设计阶段：
 1. 引入认证中心的概念解决了之前项目中·每个项目都要独立开发一套用户系统以及系统间互相间同步的问题；
 2. 设计了统一权限管理和外部应用介入管理·解决了之后开发其他微服务接入配置统一设置、存放、分发的问题；
 3. 推进了使用markdown书写文档的规范·有效的提高了文档书写、发布、迭代、共享的效率

- 开发阶段：

1. 开发了第三方node.js的sdk，方便公司其他同样技术栈的小伙伴接入，对外提供标准的http接口方便其他语言实现接入；
2. 参与了钉钉对接通讯录、短信调用、外部录播直播接口调用的设计和开发，落地了每个服务单一指责的架构形式；
3. 使用mocha.js、nock、suptest实现自动化单元测试，有效提高了敏捷开发过程中快速迭代带来的代码健壮性风险；
4. 使用新版本node.js的async/await进一步改进异步代码的可读性；
5. 引入基于git-flow的开发流程，帮助团队从svn切换到git，解决了基础代码分支出来的项目代码过多互相混乱的问题。还解决了版本发布的规范性，使得开发分支明确，处理线上bug时团队协作清晰；
6. 对于mongodb参与设计、推进了集群的使用，提高了系统的并发性能、以及数据库的稳定性。通过尝试总结优化了数据库设计，使得1000qps的请求时间从6s减少到了3s；
7. 引入redis缓存页面，加快了jade模板页面的渲染效率，省去了重构模板引擎的时间

- 部署阶段：

1. 使用docker、alinode、newrelic管理发布和监控线上以及测试环境。使得团队发现代码设计以及疑难bug的概率大大提升，并且使得性能优化变得更加直观可见
2. 配置nginx与各服务集群，使得平台管理的服务唯一，通过nginx完成同一服务下各个实例的负载均衡

- 排课系统 (2015/5-2016/5)

- 基于老师的排课系统

- 技术栈：nodejs,express,backbone.js,mongodb
 - 在预约系统的基础上，加大了对backbone.js的使用能够开发一些抽象化的组件方便团队复用

- 会议预约系统 (2014/10-2015/12)

- web预约，配合安卓终端放置会议室的预约系统

- 技术栈：nodejs,express,backbone.js,mongodb
 - 替换了公司内部框架，换成了更加开源的express框架，解决了原来框架中数据库只能sqlite以及设计上全部数据内存持久化的不可靠性。
 - 引入mongodb作为主力数据库

- 公告系统 (2013/10-2014/10)

- web界面排版，配合安卓以及windows终端展现的公告服务系统

- 技术栈：nodejs,esp(内部框架),backbone.js,sqlite
 - callback时期，使用async.js管理异步代码的可读性
 - 前段刚接触backbone.js，通过设计文件下载、查看、管理等功能第一次实现了组件化开发。

2. 客户端

- 微信的门禁 (2015/10-2016/10)

- 技术栈：nodejs,express,微信公众号开发

对接企业微信、以及门禁提供厂商，使用扫一扫打开公司的门禁系统

- 尝试了微信对接，解决了有些同事忘带门禁卡无法进入公司拿东西的问题

- html5扫一扫导航（2016/2-2016/6）

html制作的简易导航流程，配合门禁内置微信体系中

- 技术栈：nodejs,express,backbone.js
- 配合当年公司推进的导览机项目，方便一些点位稀少的环境中也可以完成导览

- 开发了基于backbone.js的表格以及分页组件

- 技术栈：nodejs,express,backbone.js

对于多个项目的总结，开发了一些基于backbone.js的可复用组件。如：基于collection自动生成的可定制表格

- 对接钉钉的html5小应用（通讯录同步）

- 技术栈：nodejs,express,mongodb

钉钉的对接，类和微信对接类似

3. 其他

- 写过基于阿里云的短信的npm模块（当时官方还不支持nodejs）
- 写过梦幻西游加点简易版模拟器。结果了当时系统更新年代久远已经没有当前版本可用的加点模拟器问题
- 写过第三方的api封装模块（系统和短信差不多）

专业技能

1. 服务端

- node.js（es6以及async/await）express和nest.js等web框架（语言框架）
- golang语言gin框架
- docker（部署）、linux
- mongodb（数据库）、mysql、postgresql
- rabbitmq（消息队列）、MQTT、socket.io、GCP-pubsub、Azure-service-bus
- nginx（负载均衡）
- redis（缓存）
- graphql（数据传输框架）

2. 客户端

- backbone.js（组件化基础框架）
- react、angular

3. 工具

- git，熟悉gitflow流程
- vscode（友好typescript）

- mocha、Jtest、postman (测试)
- restful api
- markdown (文档)
- github-copilot(ai代码工具)
- NX (Monorepo)

4. 云平台

- gcp
- azure
- aliyun

教育经历

上海第二工业大学

本科 (学士)

软件工程专业

2009.9-2013.6

英语(cet-4)