

联系方式

- 手机: 18978140011
 - Email: z_dadi@163.com
 - QQ: 3098737275
-

个人信息

- 张达棣/男/1990
 - 本科/广西科技大学鹿山学院
 - 工作年限: 8年
 - 技术博客: <http://zhangdadi.github.io>
 - Github: <https://github.com/zhangdadi>
-

##我能带给贵公司的

- 我能组建和带领技术团队, 并按公司的发展战略完成开发需求。
 - 我能对现有项目和新项目搭建一个好的架构, 一个好的架构就是一个项目的骨骼, 拥有一个好的骨骼可以让开发更快捷更顺利, 让程序更健壮。
 - 我能对团队成员提交的代码进行代码审查, 审查后会一对一的跟团队成员交流, 帮助团队成员提高代码技术能力。
 - 我能负责产品的定位、制定产品开发计划。
-

技能专长

- 精通Objective-C和UIKit, 具有已上线的完整iPhone项目的开发经验
 - 精通iOS平台UI设计, 能够熟练编写自定义控件和常用动画效果
 - 精通iOS系统及常用的framework, 对内存管理、对象生命周期有清晰的认识
 - 精通json及xml数据解析, 曾开发过xml的通用解析库
 - 熟悉常见的网络协议, 有http、socket和webSocket等网络编程经验
 - 熟悉XMPP, 使用其开发过聊天APP, 对其重构后性能提高50倍
 - 使用runtime机制做过自动序列化赋值等功能
 - 熟悉iOS应用的发布流程, 有独立在iTunes应用商店上递交过作品
 - 熟悉swift, 用swift开发过http请求库
 - 了解ThinkPHP框架, 用其开发过服务端
 - 了解flutter跨平台开发
-

工作经历

南宁市欢乐果科技有限公司 (2019年6月 ~ 至今)

职位：APP端负责人

负责公司的iOS部门和android部门，负责技术统筹，技术的解决方案，领导技术团队，培养技术人才。

项目名称：悦玩

主要功能描述：

该项目是集合的IM和游戏的APP。

主要工作内容：

- **情境**：项目是是直接购买第三方公司的代码，代码年代久远，有8年之久，IM一直是问题多多，耗能严重，拉取500条离线消息就直接发烧，卡不动，时间久等严重问题。

任务：推进的IM底层的重构，IM底层数据库的设计。

行动：刚开始主要负责iOS部门的事项，当时推进的IM底层的重构，老板让iOS组长和android组长都设计底层数据库，最终选取的我的方案进行，我总结的项目中不好的地方，结合我们的业务难处理的难点，在数据库的设计中就优化的这些方面，采用的内存处理后再批量插入数据库的方法，以前的IM底层通知到处乱用，我采用的多点委托的方法规避的通知乱飞的情况，由IM框架控制界面层的数据展示，IM的所有逻辑都在框架内实现，比如离线消息，漫游消息，退群等业务逻辑，大大方便的界面层的调用。

结果：性能上重构后拉取离线一万条上限时间只2秒不到，并且在iphone6上都不发热不卡，性能提高50倍；开发效率上重构前iOS部门一直是全公司开发速度最慢，bug最多的部门，经过我重构和带领后iOS部门提高到公司开发速度最快，BUG最少的部门。

南宁软媒科技有限公司 (2018年8月 ~ 2019年3月)

职位：技术经理

负责公司的技术部门，根据公司的发展战略规划技术部门的发展战略，领导技术团队，培养技术人才。

项目名称：优洁

主要功能描述：

该项目是一个给香港地区的公司提供清洁办公室等服务的产品。

主要工作内容：

- 制定项目的开发计划，评估开发时间，给合适的人安排合适的任务。
- 完成项目的搭建。
- 主要负责后台的开发。
- 指导其它组员的开发搭建方案。
- 解决项目中的疑难问题。
- 编写合适的SDK给公司使用，提高整体的开发效率。

广西爱店汇网络科技有限公司 （2014年6月 ~ 2018年6月）

职位：项目副经理

入职的三个月内由我负责的产品一直都是以全公司最快的速度开发完成并保证质量而多次得到同事、上级领导和老板的肯定和表扬，在入职后从负责iOS端开发升职为副经理；主要协助经理完成技术部的管理工作并负责APP端的需求分析、需求审核和制定APP组的工作计划。

项目名称：爱店汇

APP下载地址：

<https://itunes.apple.com/cn/app/ai-dian-hui/id838674750?mt=8>

主要功能描述：

该项目为一个同城O2O应用。

主要工作内容：

- **情境**：项目是由外包开发的，外包的开发不规范，代码的结构性不好，并且经常性闪退等BUG。
- 任务**：开发新功能的同时自发的对项目进行重构。
- 行动**：对整个项目分成展现层、业务层、网络层、本地数据层，四层之间互不影响，只由接口调用；网络层和本地数据层都做成动态库，和业务层比较彻底的分开，整个项目四层间高内聚低耦合，网络层管理网络请求，业务层负责界面显示，业务层请求数据唯一要做的就是组织参数，发起请求，显示结果就可以，不用做多余的处理；网络层的接口参数尽量少的，比如：翻页的时候，当前页码对网络层来说是充要条件。但对于业务方来说，这些充要条件还能够再缩减；其实业务方并不需要记录当前页号，网络层给他暴露一个loadNextPage这样的方法就够了。
- 结果**：重新构架的项目闪退问题大量减少，并且项目更好维护，在新构架上完成的接口数据缓存机制。从中我学习到的一个项目拥有一个好构架的重要性，在架构过程中对自己以前的所学、所思、所感进一步总结，在这其中提高自己的代码思维能力。

广东一一五科技有限公司 （2012年7月 ~ 2014年3月）

项目名称：115网盘

APP下载地址：

<https://itunes.apple.com/cn/app/115wang-pan/id647500047?mt=8>

主要功能描述：

该项目为一个集合的文件管理功能、新闻资讯、看电影、听歌、聊天等功能的应用。

主要工作内容：

- 完成听歌模块的功能。
 - 实现备份相片上传、管理、大图浏览等功能。
 - 和同事一起完成我聊模块的功能。
 - 因朋友圈的进度滞后，帮助朋友圈的同事解决他们碰到的难题并指导他们完成朋友圈的开发。
-

开源项目

- swift语言封装网络请求框架：<https://github.com/zhangdadi/SwiftHttp>
- 很方便的聊天工具栏：<https://github.com/zhangdadi/HDChatTextView>
- iOS开发之多表视图滑动切换示例（仿"头条"客户端）：<https://github.com/zhangdadi/HDSliderTabBarExample>

技术文章

- delegate不保留计数的多点回调：<https://zhangdadi.github.io/blog/multipleCallback.html>
- Block在ARC和MRC下的不同点：<https://zhangdadi.github.io/blog/blockDifferentia.html>