

# 宋 学

男 | 1989 年 5 月 | 5 年工作经验 | 籍贯 | 湖南 | 怀化 | 现居住于广东深圳

联系方式: 13418129051

E-mail: [willsame@163.com](mailto:willsame@163.com)

## 求职意向

期望从事职业: 移动端 APP 开发

期望从事行业: 金融相关及其他

期望职位薪资: 面议

---

## 主要技能

- 熟悉 Shell 脚本和 Gradle 构建。
- 熟悉使用 Git 代码管理工具, 熟练命令行操作。
- 熟悉多线程开发及线程池的使用。
- 熟悉第三方支付, 如集成微信、支付宝、银联支付等等。
- 熟悉 Android 的 IPC 机制, 如 Binder 机制, 熟悉 AIDL 的使用及跨进程数据读写。
- 熟悉 Http 通讯协议, 熟悉使用常见的网络请求框架: OkHttp 并了解其内容原理。
- 熟悉 Handler 机制及其源码。
- 熟悉 View 的绘制流程及 Touch 事件的传递机制, 能根据项目中的具体需求自定义控件。
- 熟悉常见的开源框架的使用如: OkHttp, Glide, EventBus, RxBus, greenDAO 等。
- 熟悉模块化, 组件化, 多渠道打包, 热修复, APK 瘦身, 代码混淆等项目中常用的技术。
- 熟悉 OOM 异常和 ANR 异常产生原因及处理方法。
- 有一定的屏幕适配和内存优化经验。
- 熟悉敏捷开发流程。
- 了解 React Native 和 Flutter, Ant Design Pro。

---

## 教育经历

- 2009.09 -- 2013.06 毕业学校: 湘南学院 | 全日制四年本科 |

## 工作经历

- 2019.06 – 2020.08 深圳市多元世纪信息技术有限公司 | Android 开发工程师 |
  - 2016.11 – 2019.05 深圳市智汇云信科技有限公司 | Android 开发工程师 |
  - 2014.06 – 2016.10 广州九尾信息科技有限公司 | Android 开发工程师 |
-

## 项目经历

### 2019.06 – 2020.08 港股通 期货通

开发工具: AS + SVN

项目描述: 港股通, 期货通为多元世纪信为其他金融合作伙伴开发的港股期货交易软件, 项目采用的是 TCP 和 Socket 来实现,数据的传输格式是 protobuf。使用 OkHttp 来实现 Https 方式的请求, 整个项目采用的是 MVP 的思想来架构设计的。

个人职责: 1、港股通, 期货通项目旧版本的维护和新版本的开发和迭代; 2、配合产品, 新需求的开发, 如: 港股通, 期货通交易模块新模式的开发等; 3、出入金线上线下, 以及线上线下出入金历史记录的功能的开发等。

### 2018.12 – 2019.05 GTS2 BTCCGlobal

开发工具: AS + Git | 版本迭代

项目描述: BTCC Global 是 BTCC 集团旗下的数字资产合约交易所。主要为全球投资者提供比特币、莱特币、以太币等主流数字资产合约交易服务。涉及数字金融生态的各个领域, 包括数字资产矿池、钱包、现货交易平台、合约交易平台等。

个人职责: 1、首页, 行情, 交易, 资讯, 我的模块的完成, APP 整体 UI 的重构, 精简 UI 层级, 优化嵌套的过度渲染; 2、梳理 JNI Native 接口, 根据产品需求, 优化和新增 http 网络接口; 3、http 更换为 https 请求, 添加证书, 配合服务端和后台联调; 4、客服系统的更换, 更换为逸创云 KF5 客服; 5、优化登陆注册和开户流程; 6、配合 JS 前端 H5 的开发, app 内各种 H5 页面的混合开发; 7、交易模块交易流程的优化, 解决遗留卡顿问题; 8、Jira Bug 的修复和版本的迭代。

### 2018.10 – 2018.11 BTCC RN 管理后台

开发工具: WebStorm + Git | 首版开发及后续迭代

项目描述: BTCC React Native 版本的 BO 管理后台, 可视化操作, 数据直观展示, 方便运营。整体项目基于阿里的 Ant Design Pro, Ant Design Pro 是一个企业级中后台前端/设计解决方案, 其技术栈基于 ES2015+、React、UmiJS、dva, 快速响应后台产品原型的开发。

个人职责: 1、使用 Ant Design Pro 搭建项目的架构, 首版开发; 2、配置 Umi 的路由方案, 统一方便的管理路由和页面的关系, 使整个后台管理项目的操作直观清晰, 层次分明, 简化操作, 易于上手; 3、与服务器约定网络请求, 修改 request 请求方法, 修改为符合实际业务接口请求的数据处理方式; 4、协定接口协议, 根据接口文档, 同步开发, 本地 mock 数据, 完善和优化接口协议, 响应式敏捷开发; 5、封装项目中常用的业务组件, 统一管理, 独立稳定, 避免冗余; 6、根据 UI 组件交互操作, 开发编写 model 的 effect 以及编写统一管理的 service 请求函数, 和封装的 request.js, 使得发送请求, 获取服务端返回, 然后调用 reducer 改变

state, 更新 model 的流程清晰, 便于管理和排查问题; 7、构建和发布项目, Bug 回归测试, 版本迭代。

## **2018.08 – 2018.10 BTCC RN 版本**

开发工具: WebStorm + Git | 版本重构和后续迭代

项目描述: BTCC 创立于 2011 年, 是目前全球运营历史最悠久、最安全、最专业的数字资产交易的平台, 数字货币 BTCC 交易软件, 致力于汇聚全球优质区块链资产, 为全球用户群提供值得信赖的数字资产投资机会。

个人职责: 1、使用 ReactNative 搭建项目的架构, 实时修复 bug 及在线发布新版本。2、使用 ES6 箭头函数, 异步函数 Promise 的链式调用, Fetch 请求和后台数据交互。3、使用 Flex CSS 进行前端布局。4、使用 React-Redux 作为数据流搭建项目框架, store 存放全局状态, Action 作为组件输入输出控制, reducer 处理业务逻辑部分。5、使用微软的 Codepush 进行实时更新版本。6、React-native-swipeout 侧滑控件的使用和封装。7、首页行情的模块的编写, 取币, 提币, 银行卡管理, 订单管理模块的编写。8、用 mode 封装 reactnative 中涉及到的弹窗。9、Bug 修复和版本迭代。

## **2018.03 – 2018.08 BTCC 数字货币**

开发工具: AS + Git | 首版开发及后续迭代

项目描述: 基于 IX 云信交易系统的数字虚拟货币交易平台, 提供比特币、莱特币、以太坊等数字资产合约交易服务, 以及为合作公司提供 app 服务。

个人职责: 1、云信交易系统 SDK 的封装和抽离, 基于 SDK 快速开发; 2、首页和我的、登陆, 注册模块的开发; 3、配合 gradle 动态加载各个公司的配置文件, 实现不同公司的动态切换; 4、App 内一些自定义控件, UI 动画效果的开发。5、Bug 修复和版本迭代。

## **2016.11 – 2018.02 GTS2 外汇贵金属交易软件 – 云信交易、GWFX Global**

开发工具: AS + Git | 首版开发及后续迭代

项目描述: 金道环球投资专注于移动交易开发, 特别推出 GTS 的升级加强版 GTS2 手机综合投资软件。速度更快, 资讯更多, 功能更贴心! GTS2 是针对亚洲区投资者精心打造的移动客户端交易软件, 是一款集在线开户、高效操盘交易、即时报价、环球金融资讯和分析师点评于一体的外汇、贵金属交易软件, 让投资者不受时间及地域的限制, 随时随地进行交易! 云信交易项目专注 B 端客户, GWFX Global 专注的是国际 C 端客户。项目采用的是自定义 TCP 层协议, 采用 Mina 来实现, 编码器和解码器的编写, 数据的传输格式是 protobuf, 协议格式是头部 10 个字节+Body 构成, 头部分字段为 cmd+seq+crc+len。

个人职责: 1、参与 GTS2 的 Android 项目框架的搭建, 整个项目采用的是 MVP 的思想来架构设计的, 避免了传统的 MVC 模式中, 导致在 Activity 和 Fragment 界面层, 既是界面的展示, 也是逻辑的控制导致, Activity 或者 Fragment 过于臃肿, 采用 MVP 后, Activity 或者 Fragment 界面只关心界面相关的交互, 比如页面的跳转, 颜色的改变, 文字的显示等等, 至于控制的逻辑, 点击的逻辑全部抽离到 Present 中, 网络请求的逻辑抽离到 Model 中, 三个层次之间采用接口进行通讯, 降低耦合; 2、参与 TCP 同步和异步请求方式构建, 具体是采用 synchronized 结合 wait 和 notify 实现; 3、使用 RSA+AES 自定义 SSL 协议, 实现数据传输的安全性; 4、参与数据库, 以及数据部分框架的搭建及后续完善, 数据库采用的是 GreenDao3.0, 项目中表的创建, 增加, 修改, 删除, 数据库升级。整个项目的数据采用的 uuid 方式增量更新; 5、交易软件涉及到行情订阅取消的实现, 首页行情模块高频率刷新的编写, 使用局部更新来应对行情一秒 4 口价格的高频率刷新, 行情涉及到涨幅, 涨跌幅的点差等计算的实现; 6、使用 ReentrantLock 解决交易模块持仓盈亏 保证金佣金的计算, 因为多线程并发读写导致行情汇率读写错误, 盈亏计算错误, 持仓增删异常等问题; 7、重连机制, 主线路和备用线路的切换, 当主服务器连接不上的时候, tcp 切换到备用线路; 8、DataBinding 的使用, 避免重复的 findById, 同时支持事件的绑定; 9、使用 google 的百分比库解决适配, 这样避免传统的那种分多 values 下进行适配导致的在特定的机型上适配可能不是很完美的缺点; 10、推送部分的对接, 推送采用的混合推送, 直接对接主流的手机厂商, 增加存活率, 同时订单成交实时同时弹窗的消息队列编写; 11、使用 OkHttp 来实现 http, https 方式的请求, 并封装调用, 整个项目中主要是出金、入金, 消息部分是 http, https 方式进行通讯的, 其他都是 tcp 方式实时回调更新的; 12、针对不同 B 端客户的需求, 定制化 UI, 夜间模式, 皮肤方案的实现; 13、项目 gradle 的管理构建, 配合编写 shell 脚本实现符合项目需求的自定义的多渠道打包, 并与服务端, 前端配合部署自动化打包系统, 摆脱频繁的由客户端打包的处境; 14、配合 gradle, 项目模块化开发; 15、每周五, 写周报, 项目进度的交流沟通及头脑风暴, 对于遗留问题的解决。

## 2015.03 - 2016.10 小九和小九商家版

开发工具: AS + Git | 开发周期: 迭代开发

责任描述: 首页模块, 我的模块部分功能的完成; 小九商家版的首版开发

使用技术: 1、项目采取 MVP 模式, 整体框架采用 RadioGroup 控制不同页面的切换, 并对自定义的下拉刷新功能进行了抽取。首页模块采用 WebView; 2、对 OkHttp 网络框架的封装抽取, 并配合服务端完成对页面 json 数据的缓存; 3、使用了 UIL 框架实现了图片的异步加载, 并完整配置框架, 按照业务需要实现不同的配置; 4、解决 SwipeLayout 与 ViewPager 的滑动冲突问题, 并自定义 ScrollView 解决与 ListView 的嵌套问题; 5、用 LeakCanary, TraceView, 静态代码分析工



具等对应用的内存和运行效率进行检测优化； 6、对屏幕适配方案的测试和解决； 7、第三方 SDK 登陆集成和分享集成； 8、通过 sl4j 建立了一套完整的日志系统，包括日志上报，crash 日志等等。 9、小九商家版首版开发，负责网络框架的搭建，网络协议的二次封装，网络请求是采用的 OkHttp 开源框架，并对其进行二次封装，将 Json 解析和 OkHttp 的请求统一封装在请求的工具类中。 10、小九商家版使用第三方框架 Picasso 加载网络或本地图片并自动缓存处理，减少内存消耗。 11、使用单例模式，将登陆成功后的用户信息存储在全局单例的 User 类中，创建 Fragment 的过程中，采用工厂模式。

## 2014.06 – 2015.02 兼职猫

开发工具：Eclipse + SVN | 开发周期：迭代开发

责任描述：框架的搭建、网络协议的封装，兼职猫、全部兼职模块以及后续版本的迭代更新

使用技术： 1、项目采用 MVC 模式，整体框架采用 RadioGroup 控制 Fragment 的切换。首页模块采用自定义下拉刷新的 ListView 来显示周边兼职信息详情。 2、分类模块及诸多二级界面采用 ListView 配合 Fragment 的切换来显示的。 3、网络请求框架采用 Volley,并自己对其进行了封装，将 Json 解析封装在请求中。 4、图片加载框架使用 Picasso，组件间的通讯使用 EventBus，并分别对使用的开源框架进行二次封装，便于以后进行框架的更换、也能减少代码量。 5、负责项目整体的优化，写布局时使用 include、merge 标签和 ViewStub 来优化布局，使用 MAT 工具检查内存泄漏。

---

## 自我评价

- 团队协作能力强，有一定的自学能力。
- 能独立思考判断问题，并有迅速准确的调试错误代码能力。
- 有良好的独立思考能力和编程习惯。
- 性格沉稳、乐观，积极向上，对技术岗位有激情，对加班不排斥。
- 英语阅读能力较强，能通过阅读英文文档和浏览国外网站学习新的开发技术。
- 在业务开展中，能够合理评估开发时间，合理分析问题，并进行任务分解与开发。