1. 为什么在腾讯做了一年就出来

在腾讯学到了很多东西，高级工程师在做同样的工作，只是分工的不同。比较愿意尝试新鲜事物。外面的世界也很精彩。很多事情都是摸石头过河。不同的路有不同的成长经历。我也不能完全知道选一条会怎样，每一条路都有不一样的经验吧。腾讯：工作驱动的学习，某一方面比较深入。新的单位：有时间系统的看看市面的书。对整个链条有一些了解。不仅体现在安卓上，可以涉及到后台。

1. 为什么想来

我个人觉得，相对中小型公司，微软和腾讯比较像。

小型公司业务不稳定，变化大，对技术的重视程度不够，不会去花时间打磨出来。技术上有进步，对业务帮助不大，高层也看不明白。

市面上深度学习资料和机会比较少。有些问题需要大用户量的验证。

从职业发展的角度来讲，广度是一方面，深度相对更重要些。还是需要精通某一块，能够解决别人解决不了的东西。

学习培训机制。人员交流。

展现自己的价值。

1. 为什么做java后台

对高并发系统比较好奇。

1. 项目总结

**猫来付：**

**需求：收款语音播报。**

用户本身希望长时间接收到。不会主动去杀死进程。

效果：

公司测试反馈：存活时间有很大的提升，除非用户主动杀死。

技术方案：

关键点

**startForeground调用，在服务启动里面调用，可以设置服务为前台。**

扩展：

技术分析：

跟安卓进程优先级有关系

学习资料

保活-》进程生命周期-》看了一些源码

源码：

AMS进程优先级相关代码

updateOom

//TODO 需要细化

扩展：

service生命周期

**金猫银猫：**

产品总体背景：外包开发的，层次简单。所有的逻辑都写在Activity里。网络库HttpClient。网络请求新起一个线程请求，post主线程处理。有的地方，连错误处理都没有。

**需求：经常改版**

**技术方案：**

**MVP实践**

问题：以详情页为例，包含多个请求，加载态的处理。改版以后。整个Activity需要重写。

解决方案：分离变化和不变。变化的是界面，数据的填充，状态的变化。不变的是后台接口，数据格式。类似于css和html的关系，表示和数据的分离。引入MVP模式。

新的问题：MVP各种接口，模板代码太多。单继承，一点用没有。只有在改版的时候才看到效果。

解耦表示和数据。不用接口，能用什么？EventBus。

具体实现：

1. 简单的界面，只要一个Activity实现。
2. 常用的状态通知，统一接口。界面和Presenter一一对应。Presenter接受生命周期通知，控制状态。
3. 界面只有update方法，接受数据用事件。

同一个总线上，有多个界面和Presenter，数据串了。

多个总线，一个核心总线。

给事件加id，多可以写判断逻辑。

展望：

是否能总结到框架中。

**需求：希望用户在没有网络的情况下能够打开**

基本实现：

本地缓存数据、图片

首页数据：http请求缓存，存文件

商品、图片数据：结构化、数据库

有网就更新，没网就取本地数据

展望：

主要是客户端实施的。

有网也可以缓存。HTTP缓存机制

后端支持。

实现思路：

离线缓存-》数据库+文件-》过期？-》代码分层-》

orm-lite, eventbus\Picasso

技术细节：

进入商品页-》Activity->Presenter->DataHelper

离线-》本地

在线-》网络-》缓存

Json数据存数据库。

商品大图存本地文件夹。

缓存清理：保存最近一周的数据。

清理数据库、缓存文件夹。

1. 首页展现出来以后，启动一个线程执行清理动作。

2. 设置，执行清理

数据库：url到本地地址的对应。

实现要点：

picasso加载图片，取出图片：保存一份到文件夹，invalidate。

http

304 机制详细了解

缓存删除机制

Activity 单层=》mvp

从安卓系统来讲：控件树、Activity

从应用开发者来讲：

Handler的机制

Bus源码

广播机制

其他优化：网页缓存。

Eventbus原理

**需求：动态化**

**全名K歌**

Fragment 源码和生命周期

有些插件化框架是基于Activity的，代理模式。相对复杂一点。

采用Fragment就非常好了。

QQ音乐技术情况

Volley库

几个队列？：本地队列，网络队列，分发队列。

提前终止。

停止更新很久了。2015年？

安卓内存

https://www.cnblogs.com/onelikeone/p/7112184.html

说明: images/lagou-division.png

**工作经历**

|  |
| --- |
| * 金猫银猫 移动开发主管、全栈开发 2015.08-2018.1. |

1. **熟悉Java后台，熟悉Mysql应用**

**Java基础扎实**（数据结构；多进程、多线程通信同步；Java内存模型等）

**熟练应用spring、spring security开发**。 对linux代码部署、nginx、cdn有应用。

**熟练应用Eureka、Ribbon等微服务组件、对消息中间件等有了解。**

**熟悉Mysql等数据库**。能够编写复杂的sql语句。对数据库范式、事务等理论有深入了解。

JAVA产品：猫来付聚合支付系统：后台、APP、POS机。应对央行支付检查。

PHP后台产品：参与金猫银猫、银生银后台开发，开发了微信登录、支付、卡券等功能。

**二、精通安卓开发，管理安卓苹果开发发布流程**

1. 精通Android Java层开发。

2. 核心框架的选型和二次开发：网络、图片缓存组件。

3. 自定义UI、复杂UI、动画效果的实现。

4. 接入崩溃组件，控制APP崩溃率1%以下。疑难崩溃的分析与解决。

5. 员工面试、管理指导员工9人。金猫银猫app开发进度管理。

6. APP质量管理：开发、BUG管理、版本管理、内测、公测（灰度发布）。

7. APP发布、运营支持：混淆、渠道打包、接入数据统计组件。关注新增、日活、留存、界面访问等运营指标。

上线APP产品：金猫银猫安卓应用、苹果应用。

未上线：猫来付手机端、猫来付POS机端。

**三、熟悉React前端框架开发，精通微信公众号开发**

1. 熟悉H5、Javascript。

2. 熟悉React框架，熟悉webpack打包、配置不同环境。

3. 熟悉微信前端、后端API。

上线公众号产品：猫券支付、猫豆商城、银生银、金猫银猫CSmall（部分功能）

**四、精通微信小程序开发**

上线小程序产品：金猫银猫CSmall、小目标助理。

说明: images/lagou-division.png

|  |  |
| --- | --- |
| * **腾讯** Android工程师 | 2014.05-2015.08 |

1. 开发全民k歌，见证了全民K歌从日活0到500万的历程。

A. 复杂界面、动画的开发：点击事件的传递，自定义组件。

B. 数据库组件的升级功能。

C. 伴奏、歌词、评分下载管理, 逻辑复杂。

D. 发现QQ空间Fragment框架BUG并修复；发现QQ空间数据库组件BUG并修复；发现QQ音乐歌词控件BUG并修复。（日活几千万）

2. 开发qq音乐。 负责网络组件、Webview组件的接入、开发、和问题解决。