

Smart Product Android APP Architecture

Version	Time	Comment	Author
1.0	2018.06.27	First version	chaw

APP 需求及特点

此 APP 是为了公司中高层管理查看公司内部生产、人力等整体状况，特别是了解生产中的效率、达标状态，还需要结合登陆者权限对内容进行甄别显示，以及显示是否已被上级关注，后续可能需要添加紧急消息推送等功能

综合以上，APP 应满足：

- 网络请求，向服务器请求具体数据以显示以及更新服务器数据库关注信息
- 本地显示，应满足操作简单，显示明了，尽可能美观
- 消息通知，获取被关注事件及显示，以及保留后期的推送接口

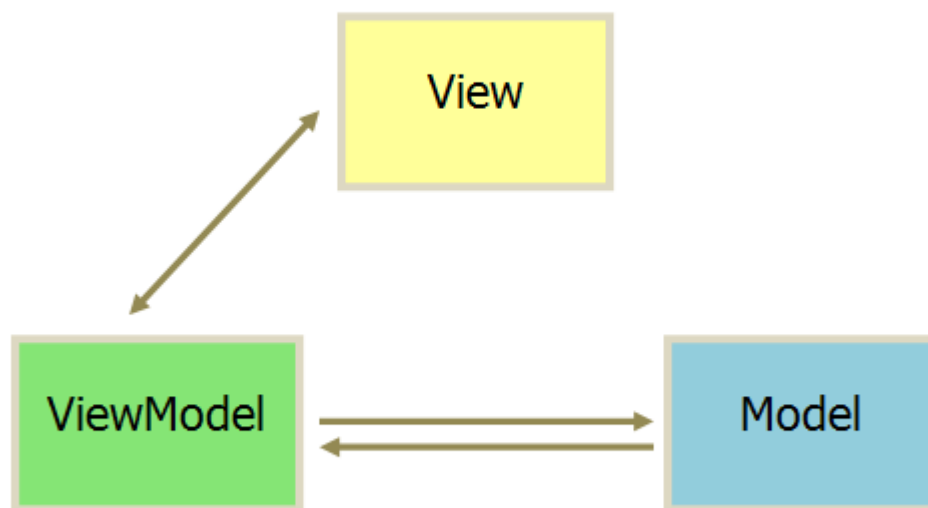
APP 具体技术特点

2018.6.26 会议：

- 远程数据库数据非即时更新，一般为日更

通用 APP 架构

常用的 APP 架构有 MVC、MVP、MVVM，考虑到此 APP 特点（显示界面较多、网络请求不多、从服务器请求的数据种类多、需要根据不同数据显示不同界面），可主要采用简单架构即可满足要求，即分层型简单 MVVM（Model：模型——处理数据，View：视图——界面显示，VM：ViewModel——逻辑业务）



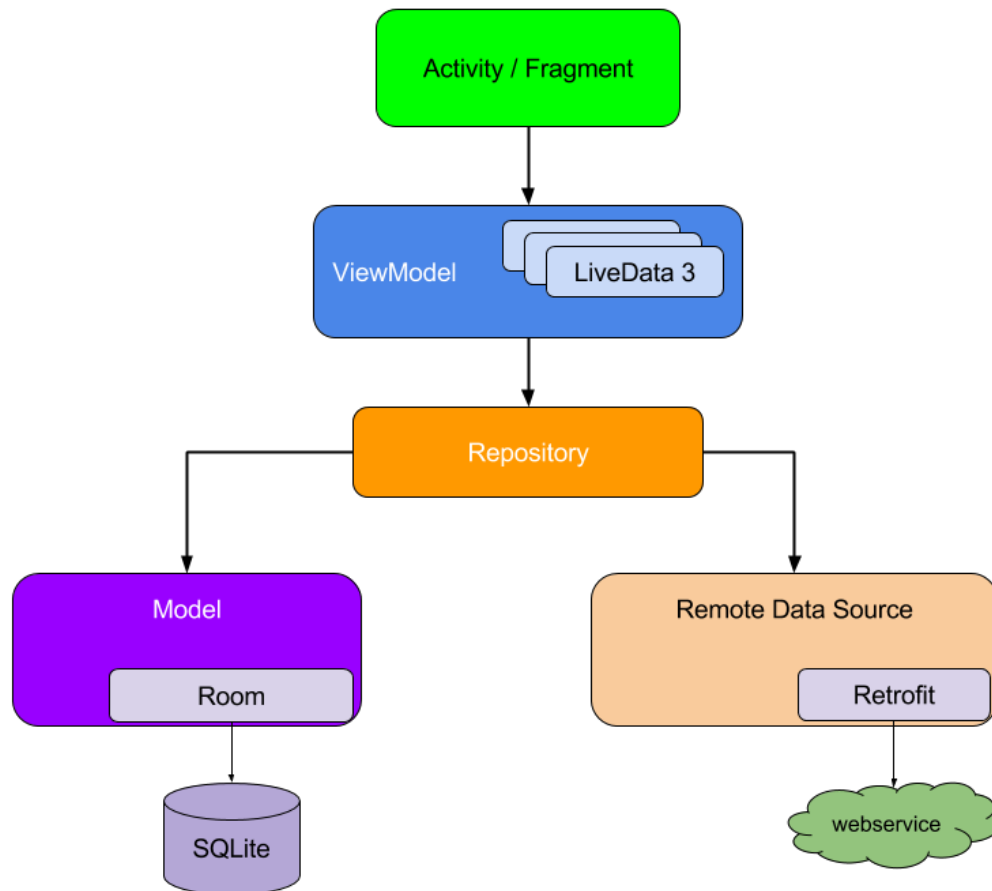
View: 处理所有的界面显示，如效率、效率趋势、指标、关注情况等，但不处理任何的数据逻辑

ViewModel: 界面和数据之间的业务逻辑，不真正操作界面和数据，还包括后续的消息通知及推送

Model: 数据，本地保存的数据，如从服务器获取的消息数据及本地保存的用户名密码等有用消息

架构

此 APP 对应 MVVM 的开发架构如下：



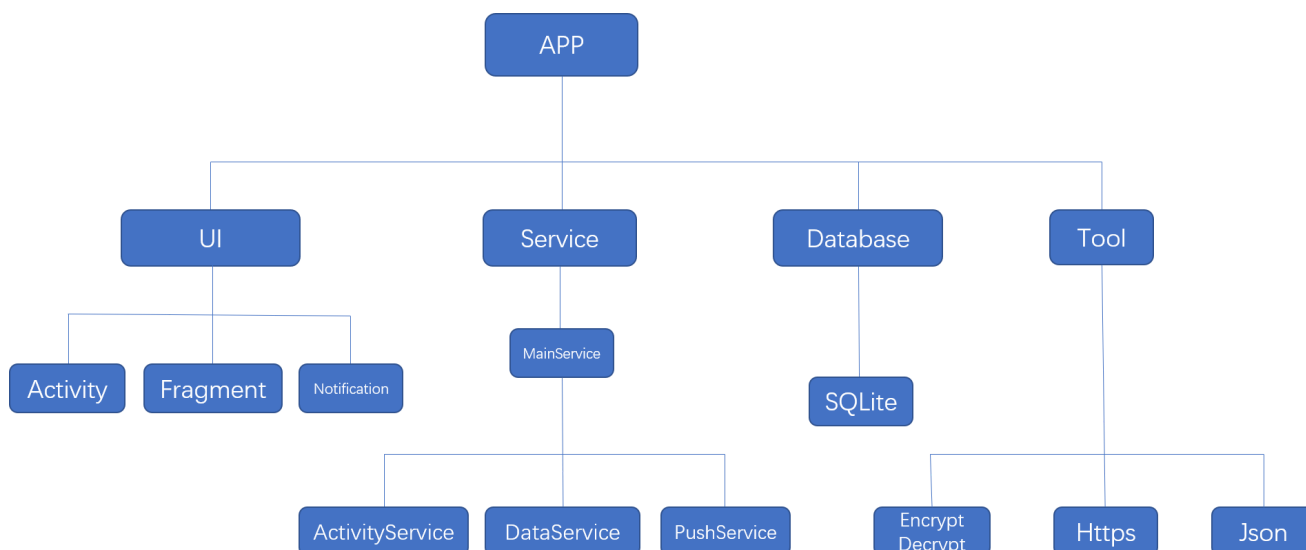
Activity/Fragment：所有界面显示有关，各种数据图表显示，与 **ViewModel** 交互采用回调接口实现；还包括消息通知等 UI 显示；为了更好的显示，所有的界面都继承于 **BaseActivity** 及 **BaseFragment**，根据需要，还可能有 **BaseAdapt** 等

ViewModel：所有不操作 UI 及数据存储的操作，包括网络请求，检测数据库，数据库存储抽象出的接口操作，以及后续的消息通知、推送等处理

Model：本地数据库，包括从远程数据库获取的数据，以及本地需要保存的用户名、密码，和其他 APP 的设置信息

编程角度框架

APP 开发及包基础框架如下:



整体共分为四大部分，由于 APP 必然存在后台服务，以及为了安全方面考虑，将以主 Service 为主线启动 APP 及其他所有功能模块

Service

- MainService
APP 入口，以及守护进程，负责启动其他 Service 以及 Activity，还包括异常捕获处理等
- ActivityService
管理所有 Activity 生命周期，以及更新数据至 UI 显示
- DataService
两类，一类是本地数据库操作，所有 UI 需要的数据均从本地数据库获取，另一类是进行 Http 服务的网络请求，包括读/写，保证 UI 刷新及时以及避免不必要的网络请求，不卡顿的 APP 要求所有计算及 UI 渲染在 16 ms 内完成，还将数据库操作及网络请求隔离开，后续数据库增删操作或网络请求变更仅仅需要更新实现在各自模块即可，在此 Service 仅调用抽象出的接口，将修改内容控制在最小范围内
- PushService
后期消息推送服务，较为常见的有绿盟、小米、腾讯等推送框架，开发时，可考虑选定使用

UI

所有 UI 元素均无法脱离 Activity 与 Fragment，所有 Activity 与 Fragment 均继承于 BaseActivity 与 BaseFragment，其仅包括最基础的 UI 处理，减少重复，尽可能缩减轮子数量，只处理显示有关的业务，避免对数据对网络的处理，将具体的业务逻辑隔离，最好使用接口，保证更改仅需要修改接口实现，而无须改动接口调用

Database

预计所需要的数据，虽繁多，但并不复杂，可考虑使用轻量级的数据库，SQLite，根据网络请求更新，并把更新后的数据传递给 UI，以及如有可能，还会保存登录用户的有关信息，为安全计，保存时进行加密处理，根据实现效果考虑使用 Base64 或更高安全等级的 SHA256 算法

Tool

包括与以上弱耦合的工具类操作，如加解密操作，http 通信，json 消息解析等

附加

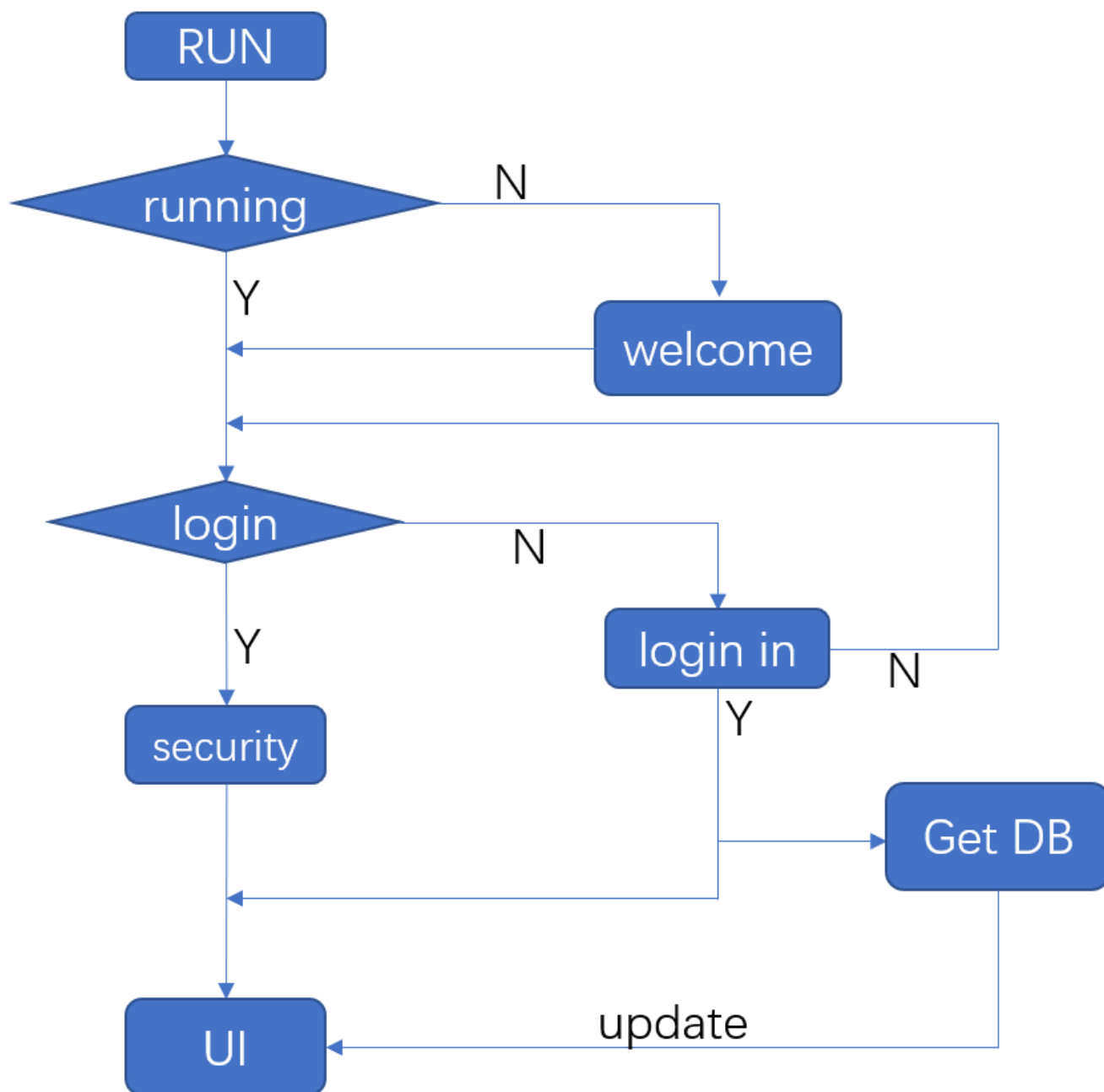
除数据加密外，考虑是否添加界面的安全保护，如类似银行 APP 的超时保护，以及登录保护

即指纹/图案登录，后台运行再次打开时的验证登录等

以上所有，根据开发进程及开发中的进展，进行更进一步的细化更新

APP 流程框架

APP 基本流程框架如下：



当用户点击 APP 启动，

- 首先检测是否在后台运行，若已后台运行，可考虑添加安全验证，如指纹验证/数字图案验证
- 未后台运行，单纯为显示效果，是否添加 欢迎界面，在欢迎界面可启动所必须的后台服务以及 UI 准备工作
- 登录成功，考虑是否添加更换用户登录、直接登录、安全方式登录（指纹、数字图案）

参考

[MVC MVP 和 MVVM 图示](#)

[Android 应用架构](#)

[QuickDevFramework Github](#)

[okhttp](#)

[gson](#)