

# 把 "格子衫" 改造得更时尚 | Kotlin & Jetpack 最佳实践技巧

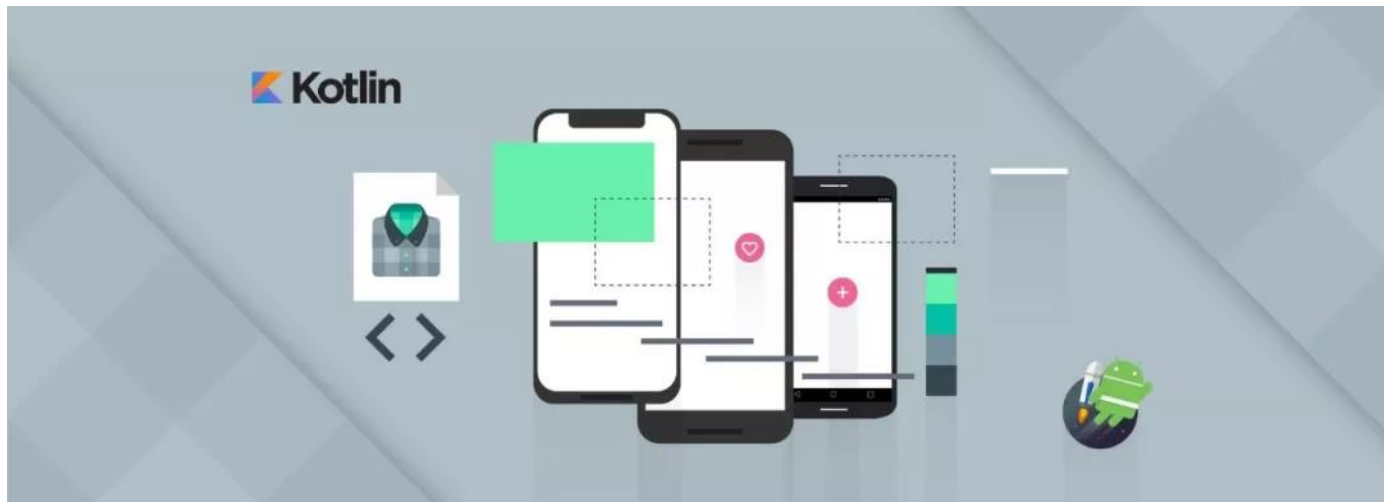


谷歌开发者

已认证的官方帐号

已关注

10 人赞同了该文章



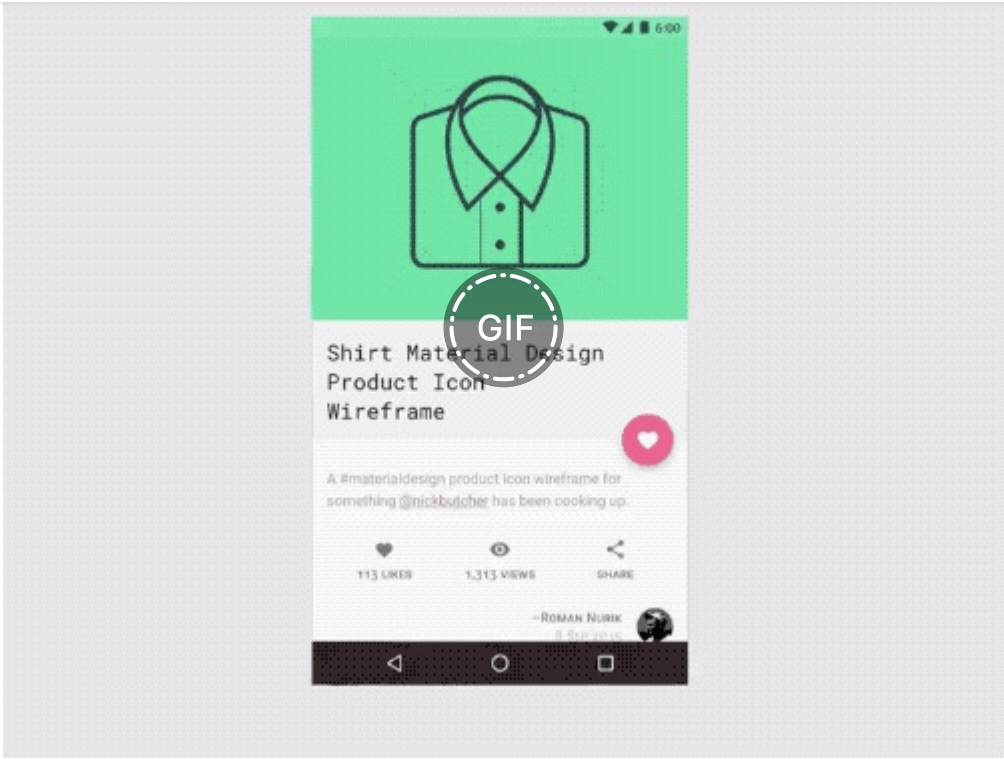
Plaid 是格子图案的意思，也是一款在 2015 年开源的 Material Design 样例应用。它通过 API 调用聚合了 Dribbble、Designer News 和 Product Hunt 三个网站的内容数据，并通过丰富的 UI 交互体验呈现给用户。

重构 Plaid 的目标不仅是修复内容源 API 接口的失效问题，更重要的是应用最新的 Android Jetpack、Kotlin 语言特性和 Android 应用架构指南，因此 Plaid 2.0 将不仅仅是 Material Design 的示例应用，同时还将是使用了 Kotlin 的符合 Android 应用架构指南的示例应用。

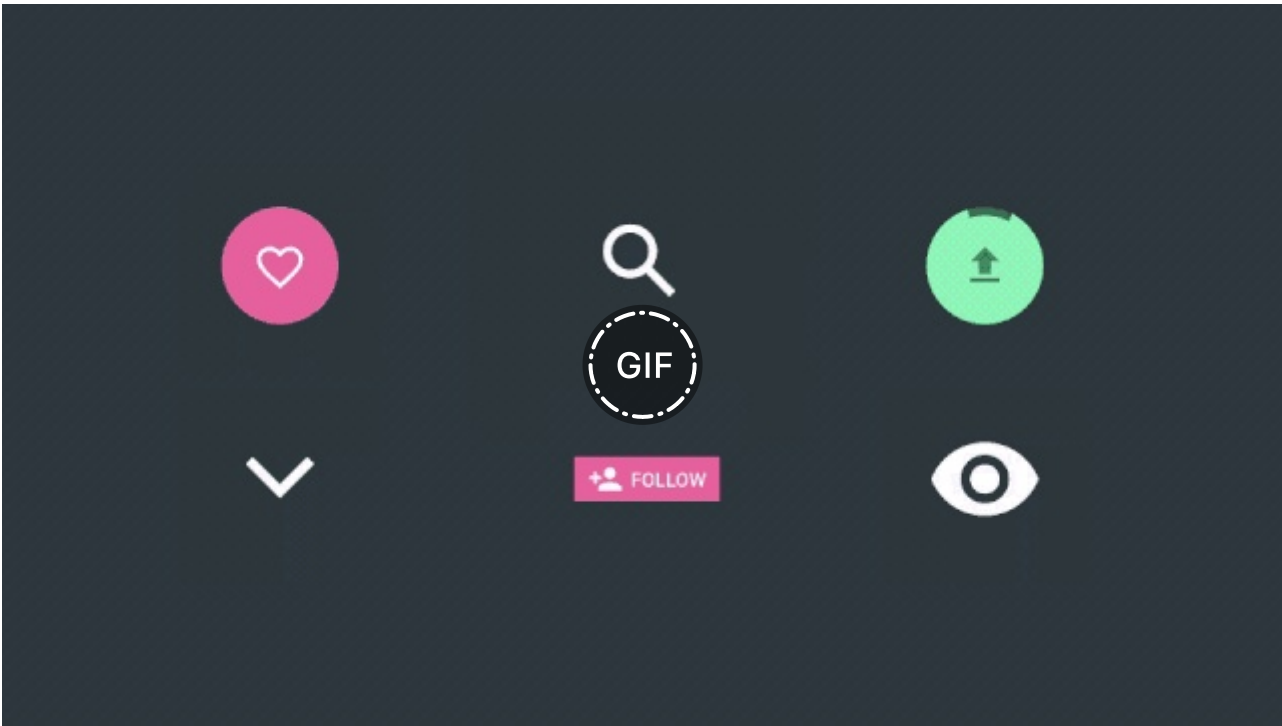
本文将为大家分享和总结 Android 团队工程师们在重写 Plaid 应用时的实践经验，前面部分将着重分享应用的分层设计，后面部分是具体使用 Kotlin 实现的视频，扩展阅读部分提供了更多 Plaid 重构以及 Kotlin 相关知识的文章。

## Plaid 1.0

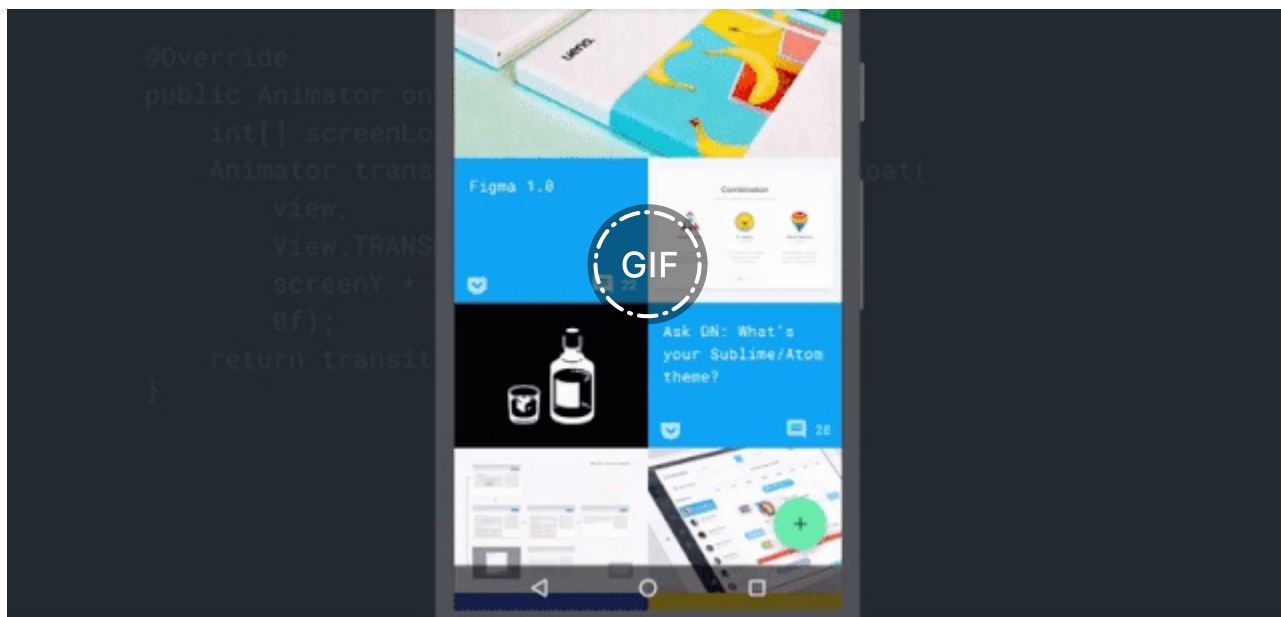
起初，我们定位 Plaid 将是一款 Material Design 样例应用，我们希望通过它展示 Material Design 的可以为交互体验带来的改进。



△ 最初的 Plaid 被设计为 Material Design 样例应用



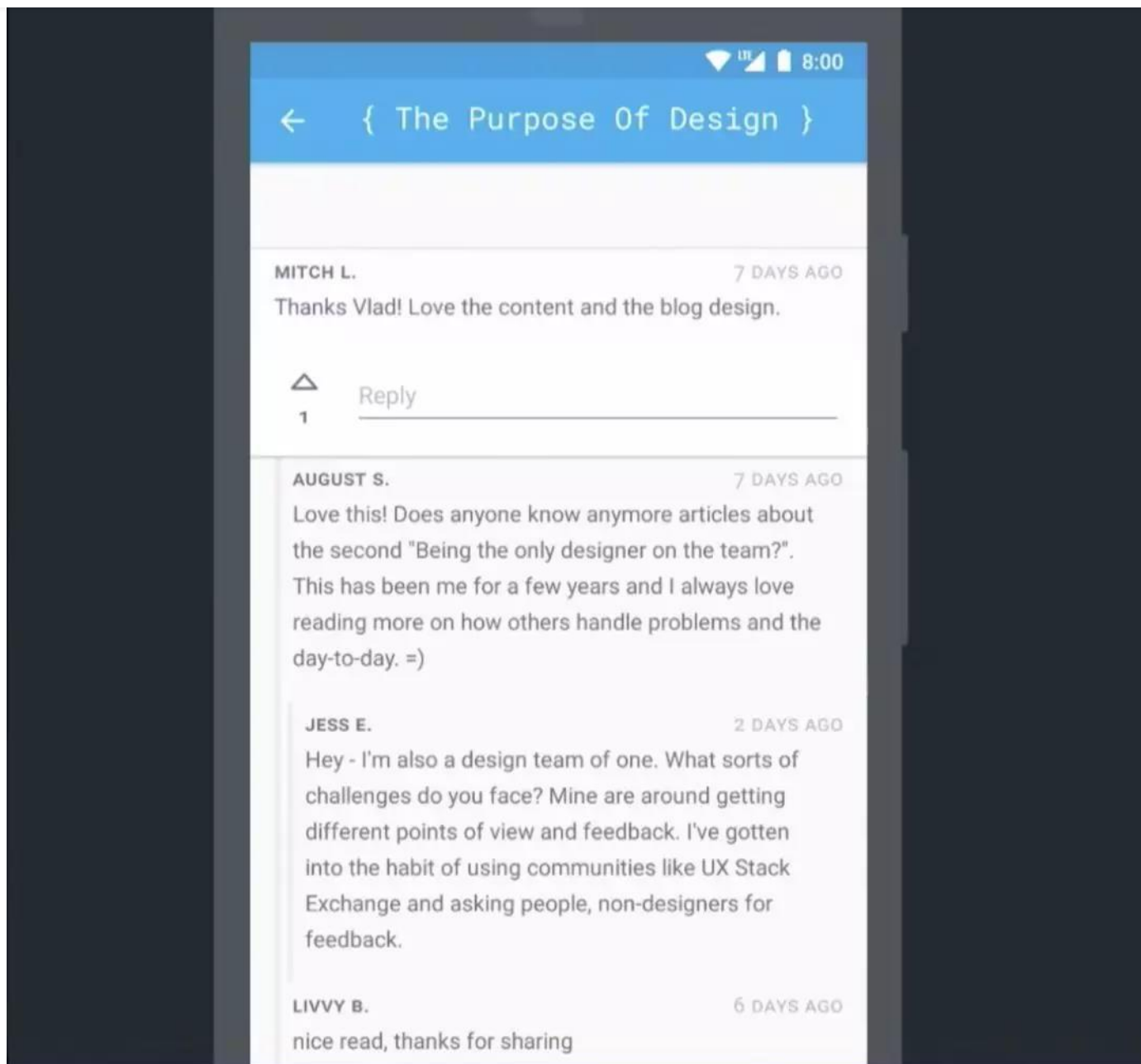
△ Plaid 里实现的图标动画效果



△ Plaid 里构建了我们称之为 Reflow Text 的动画效果



△ 让相同文字在位置转换的时候有一个酷炫的「走位」



△ Plaid 在 Designer News 模块实现了评论嵌套显示和当前评论突出显示的功能

2017 年，这些内容源的部分 API 发生了改动: Dribbble 的 Shots API 里不再返回评论的内容，Shot 点赞和用户关注的接口失效。类似这样的 API 改动，也同样出现在了 Designer News 和 Product Hunt 这两个内容源。

不过，我们有失去也有得到: Kotlin 作为 Android 官方的首要支持编程语言这一重大的消息在 17 年 Google I/O 时公布。同年 11 月，Android Jetpack 系列内的架构组件 (Architecture Components) 1.0 版正式发布，用以帮助开发者构建高质量的应用，这份指南包含了移动应用用户体验的说明，常见的架构原则，推荐的应用架构说明和最佳实践等。

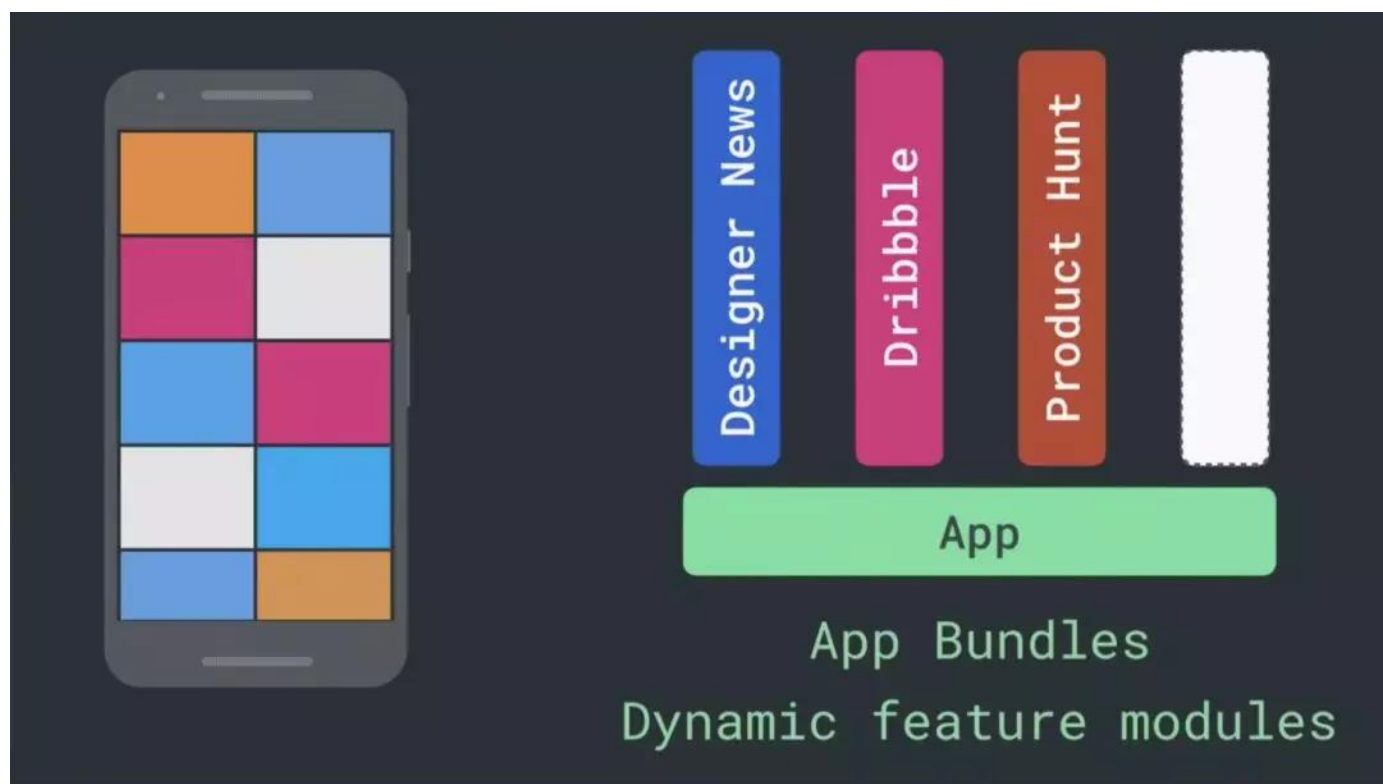
因此，在决定修复 Plaid 应用 Bug 的同时，我们希望为这个项目加入 Jetpack 架构组件和引入 Kotlin 语言，使其成为一个更 "时尚" 的应用。



Plaid 重构的目标 / 要求有以下几个:

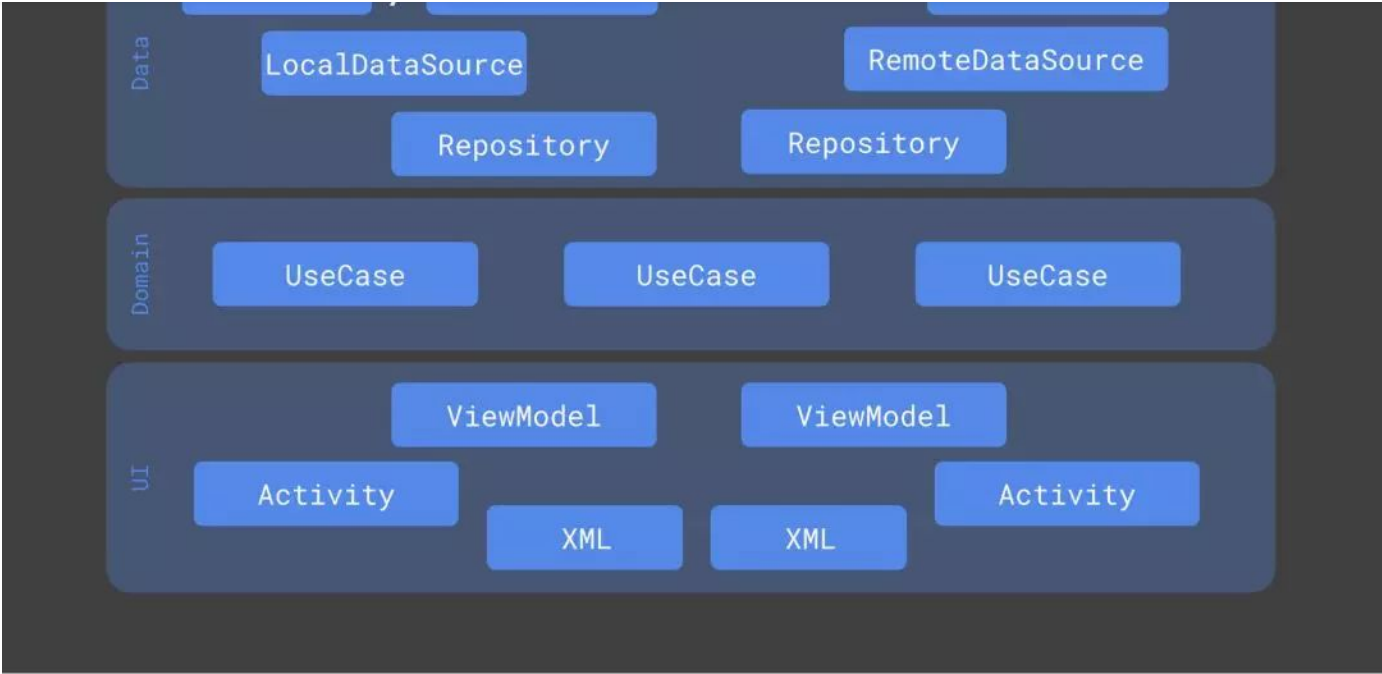
- 解决因内容源的 API 失效而导致的功能缺失问题
- 构建模块化、可扩展的应用架构，规范依赖引用和注入
- 引入和使用 Kotlin 重构现有功能

随着 Android 平台不断的改进以及 Kotlin 在全球开发者社区的迅速流行，达到这个目标变得更容易了。比如可以通过使用 Android App Bundles 动态增添新的新闻源，使用新的 AndroidX 库，使用 Jetpack，遵循 Android 应用架构指南构建应用等。



我们在应用里使用了 Android App Bundles 的动态模块功能为新的新闻源模块接入提供可能。此外，我们为 Plaid 规划了三层，分别是 Data 层、Domain 层、UI 层，并设计了一些主要的类如下图所示：





接下来我们一层一层的分享一些主要的类和它们的作用：

首先是 Data 层，这一层我们会关注数据的交互和持久化存储，我们设计了 RemoteDataSource 和 LocalDataSource 两个类，其中 RemoteDataSource 将用来与 API 服务交互，构造请求数据，接收响应数据；而 LocalDataSource 则会负责将数据存储在本机，存储的方式可以通过数据库，或者 SharedPreferences。

### RemoteDataSource

角色：  
构造与 API 通信的请求数据  
从 API 服务获取数据

依赖：  
API 服务

输入：  
请求信息

输出：  
响应信息

### LocalDataSource

角色：  
在磁盘上存储数据

依赖：  
SharedPreferences 或  
数据库

输入：  
需要被存储的数据

输出：  
数据存储

为了衔接和使用 RemoteDataSource 和 LocalDataSource 这两个类，我们设计了一个名为 Repository 的类，这个类将用于获取和存储数据，也可以把数据缓存到内存里。



如果 Repository 类只依赖于一个数据源，每次都从后端获取数据而不做任何临时存储的话，在用户离开当前页面重新返回时，应用将必须重新获取数据，即使数据未发生更改也是如此，这将会浪费宝贵的网络带宽，并且可能迫使用户等待页面完成加载。更多关于数据获取的架构设计和推荐，请参考《[Jetpack 应用架构指南](#)》。

实际的业务逻辑通常会比较复杂，我们设计了一些小型的轻量级并且可以复用的 UseCase 类，这些类将基于实际的业务逻辑来处理数据，由这些 UseCase 类构成了应用架构的第二层: Domain 层。特别要提到的是，每个 UseCase 类将只负责完成一个单独的任务，比如回复评论等：



第三层是 UI 层，首先我们设计了一个 ViewModel 类，它的目标将是为界面的显示提供数据，以及

## ViewModel

角色:

为 UI 显示提供数据

根据用户的操作触发不同的响应

依赖:

UseCase 类

输入:

id (可选)

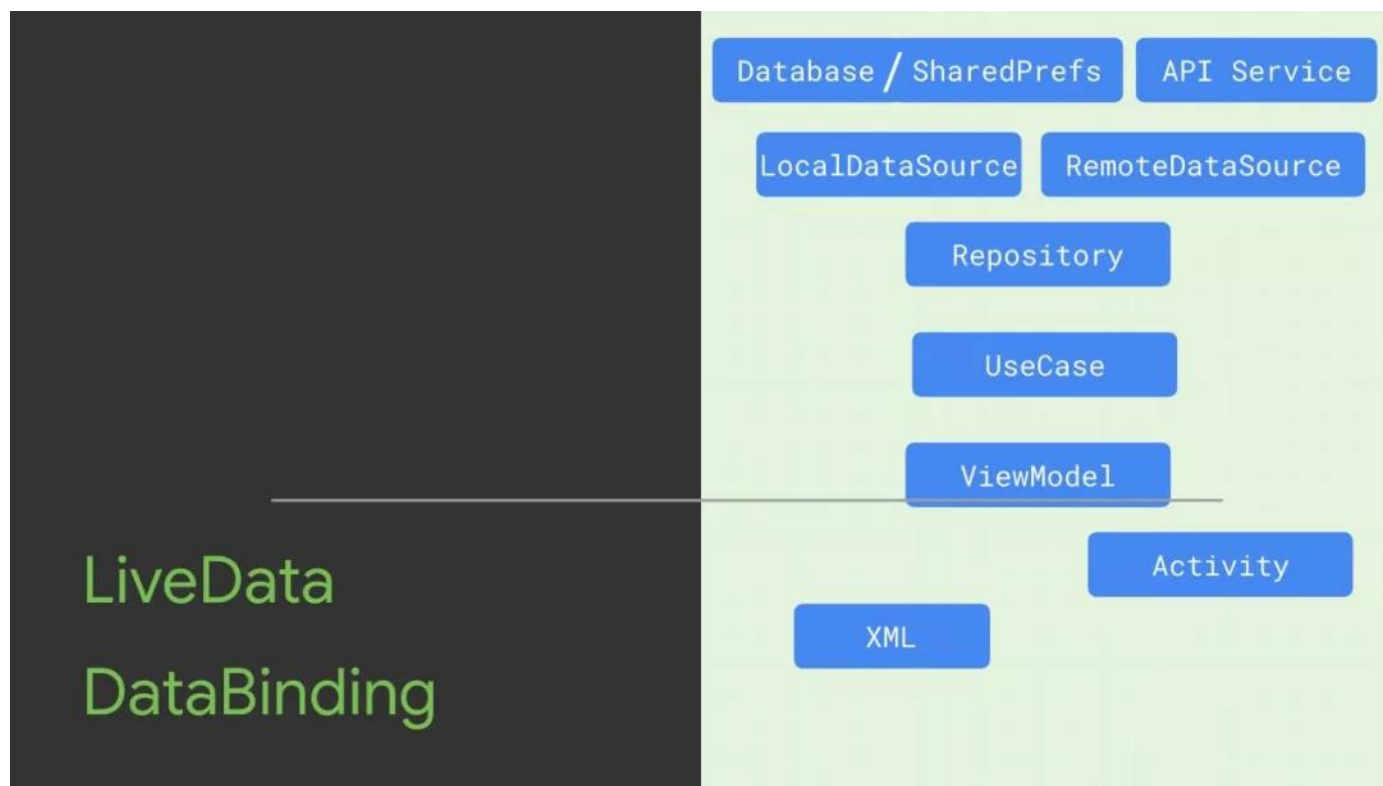
用户操作

输出:

LiveData<T>

ViewModel

在 UI 层使用 Activity 和 XML 显示界面以及将用户的操作转发给 ViewModel。



LiveData 可以很好的跟 Activity 和 Fragment 配合使用，而且与 DataBinding 结合，可以直接将数据与 XML 绑定，所以在 UI 层，这里我们选择使用 LiveData 和 DataBinding。

## 使用 Kotlin (视频)

Kotlin 语言有诸多特性 (如协程的 suspend 函数、扩展函数和高阶函数等) 可以帮助开发者更好的构建应用。比如，我们在重构 Plaid 应用时具体的利用扩展函数来提高 "when" 表达式的可读性。



Kotlin 语言帮助开发者更好的构建应用。\_腾讯视频

[v.qq.com](https://v.qq.com)



- Bilibili 视频链接

<https://www.bilibili.com/video/av70762038/>

[www.bilibili.com](https://www.bilibili.com)



## 扩展阅读

本文仅针对 Plaid 的应用架构和 Kotlin 实现进行了简单描述，更多关于 Plaid 相关文章，包括 Android App Bundles、AndroidX 重构，以及多模块的依赖注入实践等，请参看我们通过掘金翻译计划的社区内容项目发布的文章：

- Plaid 的 Android App Bundle 重构实践：  
[juejin.im/post/5c7fba65...](https://juejin.im/post/5c7fba65...)
- 将 Plaid 迁移到 AndroidX  
[juejin.im/post/5cb567cc...](https://juejin.im/post/5cb567cc...)
- Plaid 应用中使用 Dagger 管理依赖和在多模块里的依赖注入  
[juejin.im/post/5cee7ae6...](https://juejin.im/post/5cee7ae6...)

Plaid 2.0 应用在 GitHub 开源，查看更详细的代码实现，或参与本开源项目，请查看：  
[github.com/android/plaid](https://github.com/android/plaid)

关于 Kotlin 的更多知识，可以参考我们之前发布的内容 [《31 天，从浅到深轻松学习 Kotlin》](#)，或 [点击这里](#) 访问 Android 开发者文档网站。

# android

发布于 2019-10-11

推荐阅读



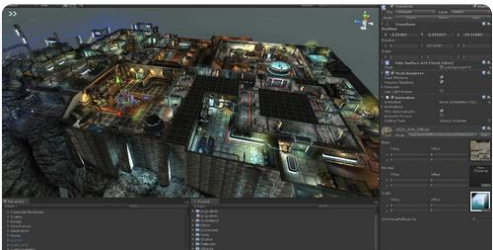
用 Kotlin 开发现代 Android 项目 Part 2

KenChoi



作为一个程序员怎么通过 android 开发赚钱

室内设计二... 发表于室内设计



Android 开发环境搭建（干货）

宁古分享家 发表于 Unity...

还没有评论

写下你的评论...

