**青训营结营实践——汇报文档——上学人组**

1. 项目介绍

本项目为基于Android studio开发的简易版抖音

项目服务地址 <https://github.com/xiangmingP1/qxy_sxr>

1. 项目分工

|  |  |
| --- | --- |
| 团队成员 | 主要贡献 |
| 张合嘉 | 编写答辩汇报文档 |
| 刘湘铭 | 负责简易版抖音的框架搭建 |
| 陈明耀 | 主动和队长沟通提供技术思路 |
| 朱帅然 | 发表意见提供思路 |
| 沈彤 | 无安卓基础但认真求教 |
| 刘雨函 | 不会安卓开发无法参与项目搭建 |
| 罗红云 | 没有学过安卓无法参与项目搭建 |
| 陈冲 | 不懂安卓没有参加项目搭建 |

1. 项目实现

### 3.1 技术选型

布局主要采用ViewPage2+fragment实现电影榜单的页面切换。只做两个Fragment，然后复用RecyclerView，

### 3.2 架构设计

MVP架构：MVP是单词Model View Presenter的首字母的缩写，分别表示数据层、视图层、发布层，它是MVC架构的一种演变。作为一种新的模式，MVP与MVC有着一个重大的区别：在MVP中View并不直接使用Model，它们之间的通信是通过Presenter (MVC中的Controller)来进行的，所有的交互都发生在Presenter内部，而在MVC中View会直接从Model中读取数据而不是通过 Controller。

 Model: 对于Model层也是数据层。它区别于MVC架构中的Model，在这里不仅仅只是数据模型。在MVP架构中Model它负责对数据的存取操作，例如对数据库的读写，网络的数据的请求等。

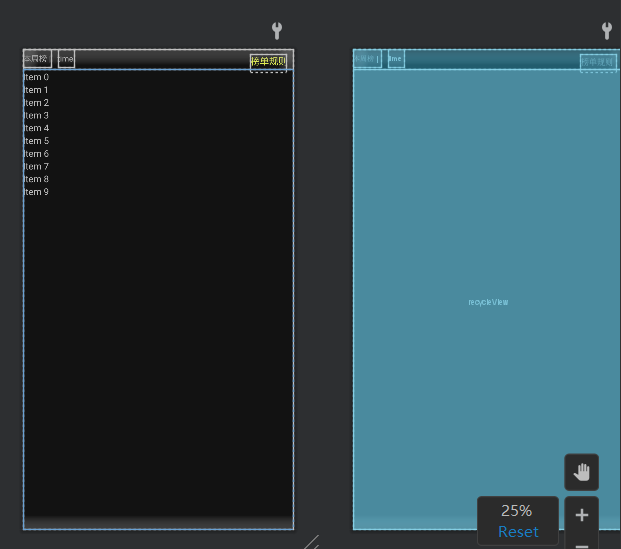
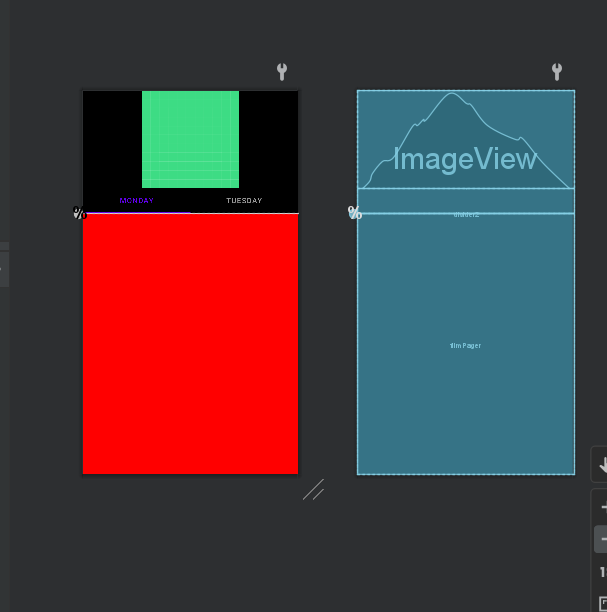
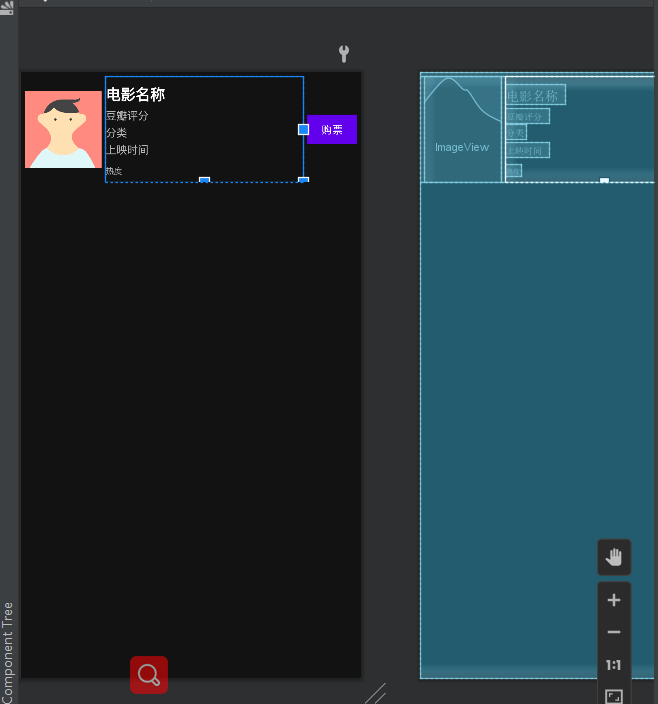
 View: 是显示数据（model）并且将用户指令（events）传送到presenter以便作用于那些数据的一个接口。View通常含有Presenter的引用。在Android开发中通常将Activity或者Fragment作为View层。

 Presenter:对于Presenter层他是连接View层与Model层的桥梁并对业务逻辑进行处理。在MVP架构中Model与View无法直接进行交互。所以在Presenter层它会从Model层获得所需要的数据，进行一些适当的处理后交由View层进行显示。这样通过Presenter将View与Model进行隔离，使得View和Model之间不存在耦合，同时也将业务逻辑从View中抽离。

### 3.3 项目代码介绍

访问GitHub公共仓库即可查看：<https://github.com/xiangmingP1/qxy_sxr>

1. 测试结果



1. 演示demo

因负责开发项目框架的队友家中老人去世，无法进行演示demo，其他队友又均不具备安卓开发技术，所以无法演示demo，对此我们深感遗憾。

1. 项目总结与反思

1.拿到access\_token 怎么利用这个token 请求数据？

目前Access\_token的有效期通过返回的expire\_in来传达，目前是7200秒之内的值。中控服务器需要根据这个有效时间提前去刷新新access\_token。在刷新过程中，中控服务器可对外继续输出的老access\_token，此时公众平台后台会保证在5分钟内，新老access\_token都可用，这保证了第三方业务的平滑过渡；Access\_token的有效时间可能会在未来有调整，所以中控服务器不仅需要内部定时主动刷新，还需要提供被动刷新access\_token的接口，这样便于业务服务器在API调用获知access\_token已超时的情况下，可以触发access\_token的刷新流程。