设计文档

17301087-覃脍

17301081-肖剑

一．功能介绍与技术分析：

1. 启动页

功能分析：当打开APP时，首先会进入到启动页，启动页参考老师提供的示例，进行了修改，一共有4页，每一页的下方也设置了Next与Skip按钮，点击Next可以进入下一页启动页，也可以点击Skip跳过，此处我们设置了是每次打开都需要进入这个启动页的介绍动画，也可以通过代码的修改，设置只是第一次进入此APP会进入这一启动页。

技术分析：

这部分主要采取的就是要新建一个welcomeactivity,并且设置它，让它在mainactivity之前启动，就可以在进入app的时候，首先进入启动页，剩下的就是设计n个不同的xml布局即可。



1. 登录与注册：

功能分析：

在这部分，用户可以进行登录与注册，两者功能的实现都是在修改老师提供的服务器的基础上实现的，我们的登录注册验证都是在服务器端进行的。

注册方面，输入昵称与密码，可以进行注册，用户信息可以存储在数据库中，登录部分，正确输入用户名以及密码后，可以登录，并且登录后，会跳转到主页面。

技术分析：

登录时用户输入用户名和密码就能直接登录，核心就是访问了后台的登录接口，后台验证登录成功后，将后台返还的用户信息通过SharedPreferences保存到本地，以便于后续页面获取用户信息以及记录用户登录状态。

用户注册同样是调用后台的注册接口，将后台返还的数据就能成功注册，将后台返还的用户信息通过SharedPreferences保存到本地，以便于后续页面获取用户信息以及记录用户登录状态。



1. QQ登录

功能分析：

我们也可以通过QQ进行登录，在这引入了第三方登录，可以获取头像与昵称等信息。

技术分析：

通过申请QQ第三方登录接口，并且本地导入QQ Login的SDK（libs/open\_sdk\_r2973327\_lite.jar），通过将返还的QQ用户数据利用后台的QQ登录接口登录并注册上用户信息，存储用户的头像信息。将接口返还的用户信息通过SharedPreferences保存到本地以便后续页面获取用户信息以及记录用户登录状态。



1. 个人主界面（实现记住登录状态）

功能分析：在登陆后，个人主界面可以显示用户的头像以及昵称等信息，在这部分，我们实现了记住用户的登录状态，用户如果不选择退出登录，下次进入到APP时，还是会进入到该登陆状态，用户可以选择退出登录，进行新账号的登录。

技术分析：

将登录和注册获取到的数据从SharedPreferences提取出来，放入用户的头像和昵称。并提供退出用户的登录状态的接口，退出登录后能将存储的SharedPreferences清空并清除页面栈然后转回登录页。



1. 课程列表界面：

功能分析：在这个界面，可以查看课程的信息，在这个界面，我们在考虑之后，只引入了课程名，课程简介，开课时间等信息，并通过不同的Item进行展示，比如推荐课程的item,就只介绍了课程的图片以及课程名，即将开课课程界面，则引入了课程名，课程介绍，课程开课时间等信息，最后的正常课程列表界面，则介绍了课程名与课程信息，课程图片等。

技术分析；这个界面，采用了RecycleView中引入不同的Item布局，在这，我们引入了三个不同的item布局，分别在xml布局文件中进行设计，同时，自定义了一个抽象类TypeAbstractViewHolder,并且分别对于三种布局，设计了不同的ViewHolder,并且，自己设计了CourseAdapter,根据不同的type，对于Item进行不同的操作。最后在课程列表界面对应的Fragment中，加载不同的Item到界面中。



1. 详细课程界面：

功能分析：详细课程界面，则对界面进行了简单的布局与划分，最上面展示了课程的图片，课程名称，开课时间，这里的截图稍微有些错位是因为设计的时候，是在虚拟机上设计的，但是截图是在真机测试时截图，屏幕的宽窄有区别。接下来展示了老师的信息，包括头像，名称，电话，邮箱等。最下面是相关资源，点击下载可以进行PDF文件的下载，会在之后介绍，点击分享也可以分享到QQ，会在之后介绍。点击播放与暂停，可以进行视频的播放与暂停，这部分也会在之后进行说明。

技术分析：界面的实现相对较为简单，知识考察了不同的布局的使用，难点在于一些功能性模块的实现，我会在后面进行说明。



1. 加载资源的线程异步：（从服务器获取资源）

功能分析：在每个界面进入之前，会向服务器发送请求，请求资源，这个功能是线程异步的。

技术分析：使用了Google推荐的集成Okhttp的Volley，使用这个依赖进行的Http请求默认就是异步的。

1. 播放视频文件:

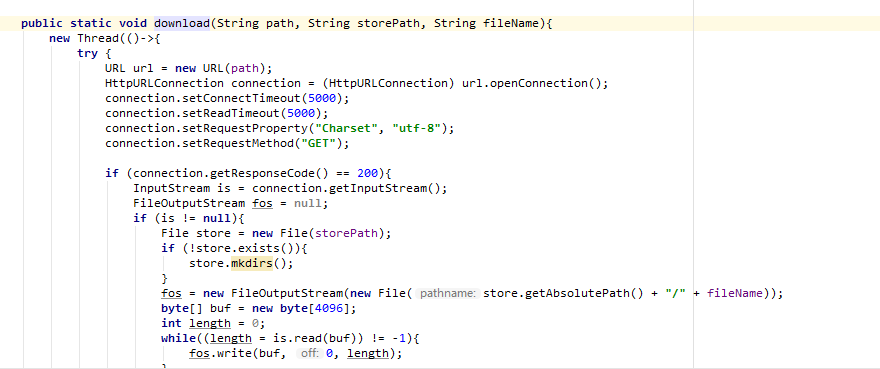
功能分析：这一部分，我们可以播放视频，也可以选择暂停视频播放。

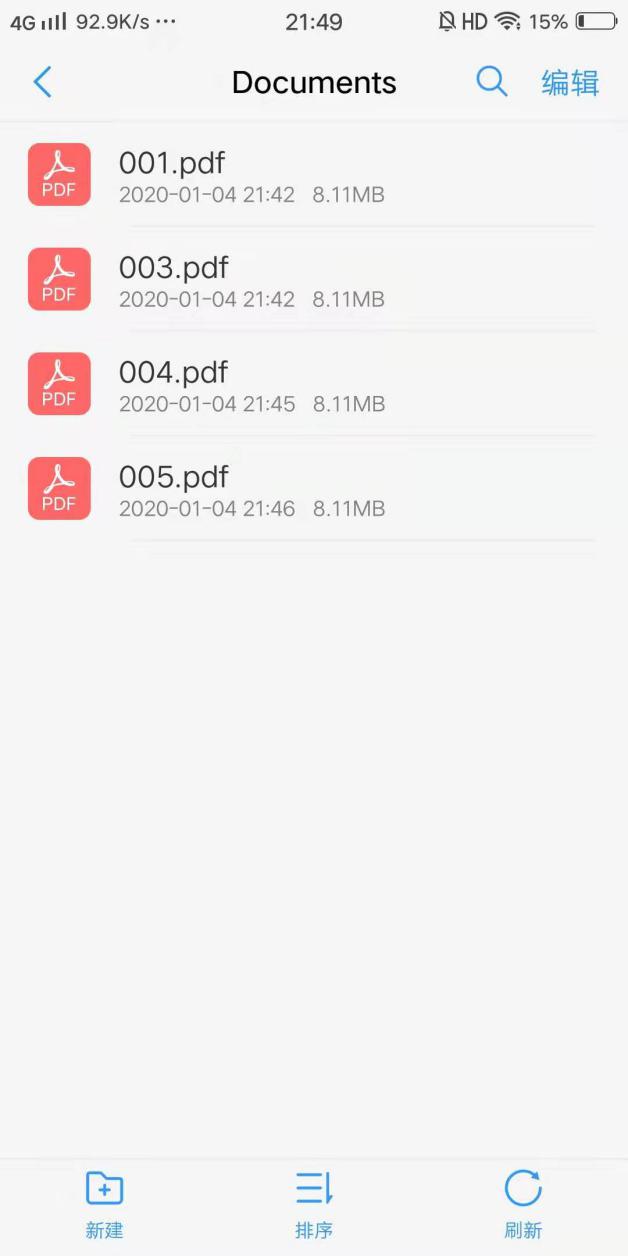
技术分析：在这部分，我们采用了BILIBILI的IjkMediaPlayer进行视频播放，这一组件，相对于原生的VideoView，更轻量级，也更易于操作，但是在视频播放中，其实发生了一个问题，困扰了我比较久，正常的视频传输是靠流进行的，我记得老师说过，我们的服务器不是通过流进行传输的，这就造成了我在使用IjkMediaPlayer时，会造成丢帧的问题，而且这个问题，每次丢帧的多少是不一定的，就会造成视频卡顿，甚至视频不能播放的问题，我在网上查询之后，都说是因为服务器的问题，但是由于修改服务器难度太大，我们就没有进行服务器的修改。

1. 下载PDF功能：

功能分析：在这部分，点击课程详情页的下载后，可以下载课程对应的PDF文件，可以在手机中找到。

技术分析：通过Java原有类HttpURLConnection获取后台接口的PDF资源对应的InputStream，然后将输入流输出到指定的外部存储路径。





1. 通知功能：

功能分析：点击每一门课，进入详情页时，会发一个通知，用户可以在通知栏中查看。

技术分析：首先确认用户的Android版本，将26以上的通知分离出来，在26以下的直接使用Notification建立通知。在26及以上的通知先建立Channel，之后将Channel注入到Notification中建立通知。





1. 分享功能：

功能分析：点击分享按钮，可分析内容至QQ。

技术分析：同样是通过申请QQ第三方登录接口，并且本地导入QQ Login的SDK（libs/open\_sdk\_r2973327\_lite.jar）实现基本的QQ分享功能





1. 过渡动画进行导航：

功能分析：点击课程列表页的我的按钮，可以进入到个人主页，在这使用了动画，自定义了动画类型。详细演示在我们的视频中。

技术分析：这部分的实现，与正常的给Acitivity之间，与Fragment之间添加动画还不一样，由于我们的导航是采取的安卓的新组件，JetPack中的Navigation,这个组件很好的集成了底部导航栏的功能，但是由于引入了Fragment，其实在跳转的动画处理上，难度提升了一些，最后，在Navigation的xml文件中，首先将两个Fragment连接在了一起，然后引入了anim中的动画，实现了动画的跳转，底部导航栏按钮点击的动画跳转，我进行了尝试，但是没有具体的实现，以后会进一步探索与尝试。

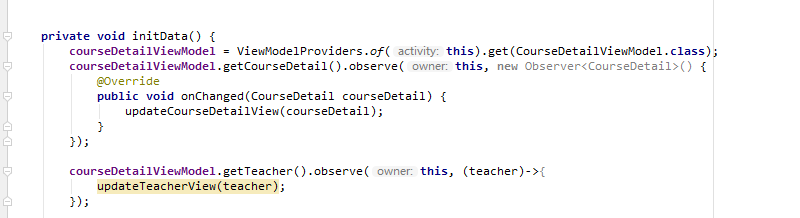
1. MVVM重写课程资源模块：

功能分析：

通过MVVM将数据的绑定从原有的绑定中分离出来，使用LiveData配置View的数据。

技术分析：

利用Android自带的ViewModel类和MutableLiveData<?>实现基本的MVVM配置



1. 单元测试：

功能分析：对登录模块（LoginActivity）进行简单测试

技术分析：简单地将登录后返还的JSON字符串的解析进行了测试



二．完成情况总结：

在我们两人的合作下，我们基本的完成了这个项目的开发，在开发过程中，有许多难点去克服，安卓的更新较快，会导致版本控制的一些问题，这在合代码，导入库，以及同步方面，引起了一些问题，在功能的设计上，我们尽量的按照老师的要求，完成了功能的实现，虽然有很多不尽如人意的地方，也有许多地方的设计略显不足，但我们今后会继续学习，去开发出更好的安卓项目。

通过这些assignment的实现，首先，让我对于一个安卓项目的基本要求有了大概的了解，我们更加切实的感受了一个较为成熟的安卓项目需要满足用户的哪些需求，比如记住登录状态，缓存等问题，也切实的学习了如何去实现这些方面的功能，这种体系化的assignment作业，对我们来说，是一个挑战，但是也较为完整的让我们了解了安卓项目，学习到了许多安卓开发方面的知识。