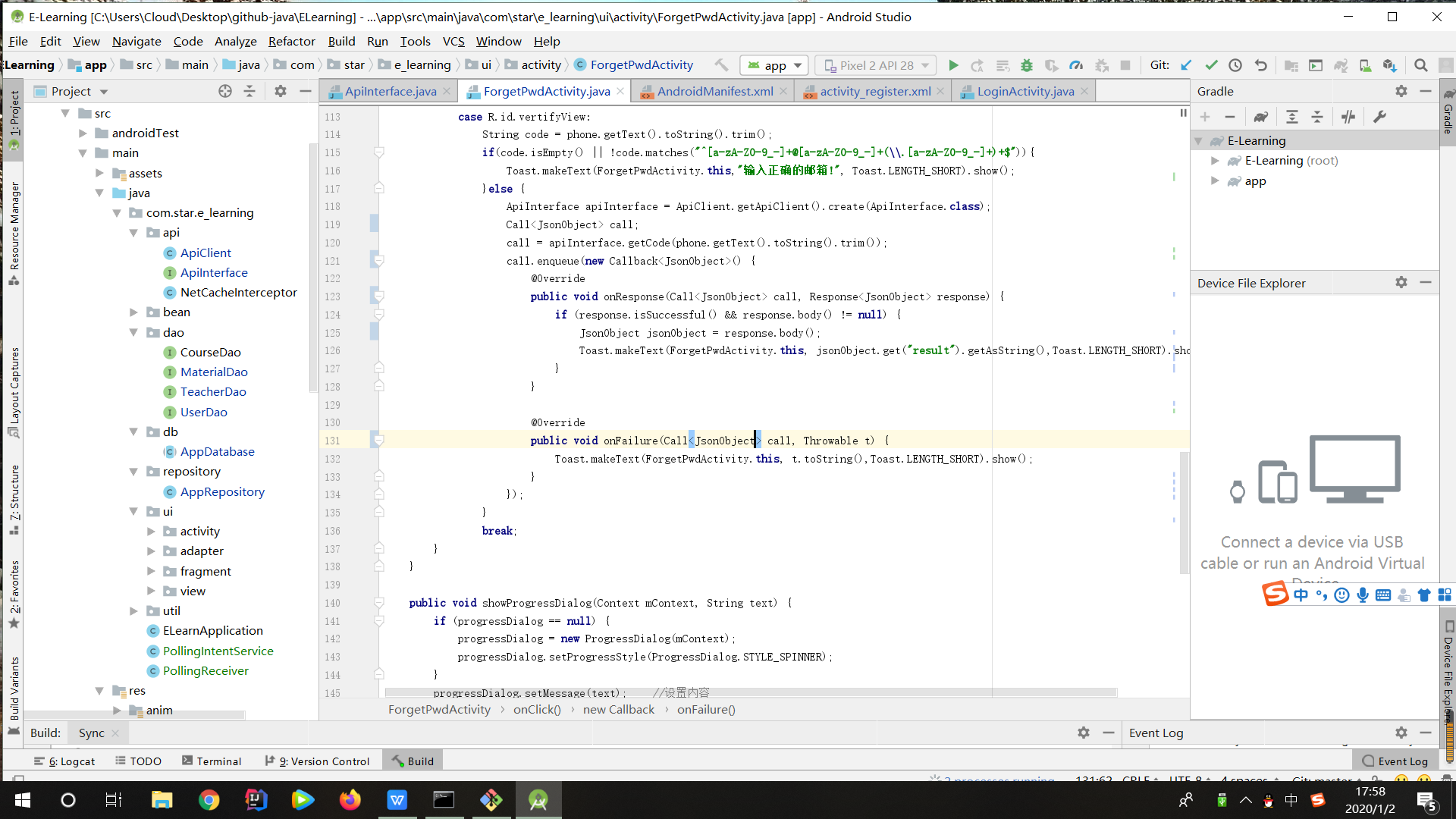
**注：老师，您好，由于之前没搞懂要求，把文档都写一块了，时间来不及重写文档，望老师见谅，作业要求该有的我都有，除此之外我还添加了user表，增加了额外的个人信息简单的修改，在我的shortVideos文件夹下有几个简短的视频，展示了multitype的recyclerview和一些视频材料的播放。**

**除此之外，服务器我也改了一些，用的IDE跑的。**

**项目大体结构如下：**

****

** app logo**

1.Background

随着当今社会经济发展，各种教育机构越来越多，人们对各类学习软件的需求也越来越明显，本产品是一个学习类的app，主要针对学生学习，让选择参加课程，观看视频，相互交流进行学习。

角色与定位：

|  |  |
| --- | --- |
| **用户角色** | **目标人群定位** |
| **学生** | 查看课程、浏览课程、更改信息、注册账户、修改密码、用户登录、查看课程老师、观看课程材料（包括视频、图片、音频、pdf等文件） |

技术路线：

|  |  |  |
| --- | --- | --- |
| 技术路线 | 客户端 | 客户端主要是针对安卓进行开发，包括用户界面的设计实现，以及界面间的联系、跳转。 |
| 前后端交互 | 客户端通过 HTTP 请求服务器接口，服务器接口根据对应参数，返回 JSON 格式数据。客户端端解析 JSON 数据，进行数据显示。 |
| 后台服务器 | 使用 Spring boot 框架，严格遵守接口分离原则。Dao 层向Services层提供数据库操作相关接口，由Controller  层调用 Services 层接口完成业务逻辑操作，并将数据  以 Json 格式返回给客户端。 |

1. Main Functions

Functional demands:

|  |  |  |
| --- | --- | --- |
| 功能需求 | 学生 | 登录  注册  找回密码  课程列表查看  课程详细信息查看  课程视频观看  接收重要事项通知 |

1. UI design

一、欢迎页面



1. 我的Introduction slide page，一张图片
2. Splashactivity，同时使用handler两秒后根据 shared preferences获取当前登录状态，根据登录情况跳转至登录界面或主界面，若已登录，则继续通过 shared preferences获取当前登陆者邮箱，继而从本地数据库中获取用户具体信息，若未登录，则跳转至登录界面。

二、登录、注册、找回密码

  **使用mobsdk进行qq授权登录**

登录界面

1.通过邮箱密码登录，密码采用inputtype=textpassword,输入时不可见

2.使用awsomeavalidation添加了正则表达式验证

1. 可以点击qq图像通过mob sdk第三方登录（第三方登录未关联用户，故不能修改个人信息，只能查看课程模块）
2. 该界面可以跳转至注册界面以及忘记密码的找回密码界面。
3. 登录时会弹出加载框，同时使用retrofit请求服务器进行登陆操作，操作完成时会使用toast显示结果。
4. 若登陆成功，则会更改 shared preferences的登录状态(通过调用自己写的Utlis.putValue()方法)，更改当前登用户邮箱，想本地数据库用room api插入该用户数据。

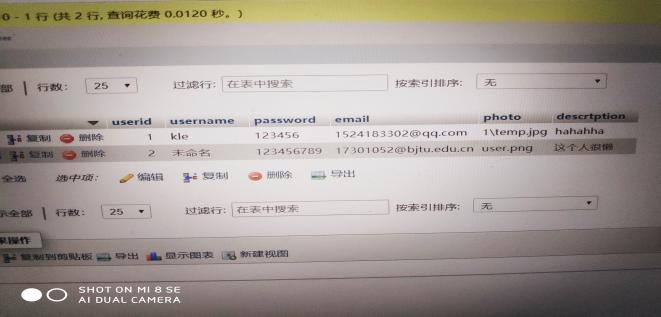
 

验证码 注册成功

注册界面

1. 同样采用awsomeavalidation正则表达验证
2. 采用邮箱注册，通过服务器调用一个我自己的邮箱发送验证码

 密码更改为123456789

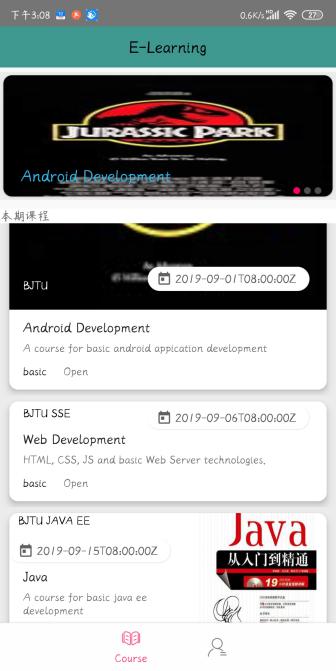
忘记密码界面

1.同样采用awsomeavalidation正则表达验证

2.采用邮箱验证，通过服务器调用一个我自己的邮箱发送验证码

1. 课程界面

由于主界面的multitype的recyclerview不好截长图，请老师观看shortVideos文件夹下的视频



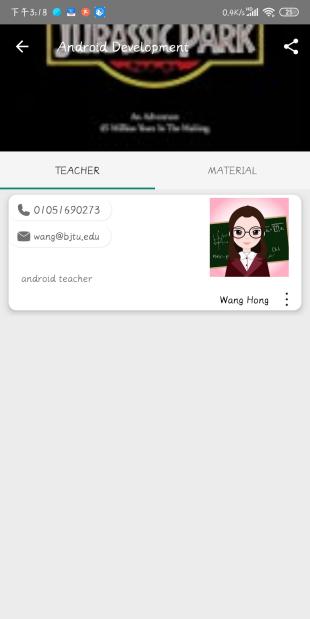
主界面

1.上面是一个recyclerview做的轮播，下面是一个recyclerview做的multitype，上面可以 左右滚动，下面可以上下滚动。

2.主界面可以下拉刷新，同时会向网络请求，每次请求加载完毕后，都会向数据库中插入实体类，向手机中存入本地课程图片，没有网络时，会显示本地缓存的数据没有网络时，会显示本地缓存的数据（room与local file）。

3.采用livedata，当数据改变时ui会变。

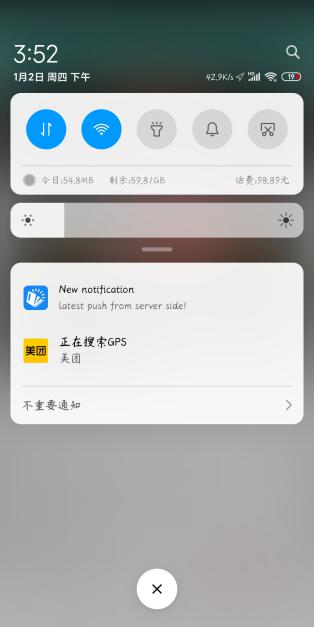
4.若没有网络也没有缓存数据，则会显示errorlayout，由于这个实在不好测试（逻辑设计是登陆进去立刻就请求数据的），就没有截图，但逻辑里是有的（防止意外发生）。



然后点击一个卡片进入后会出现详细介绍

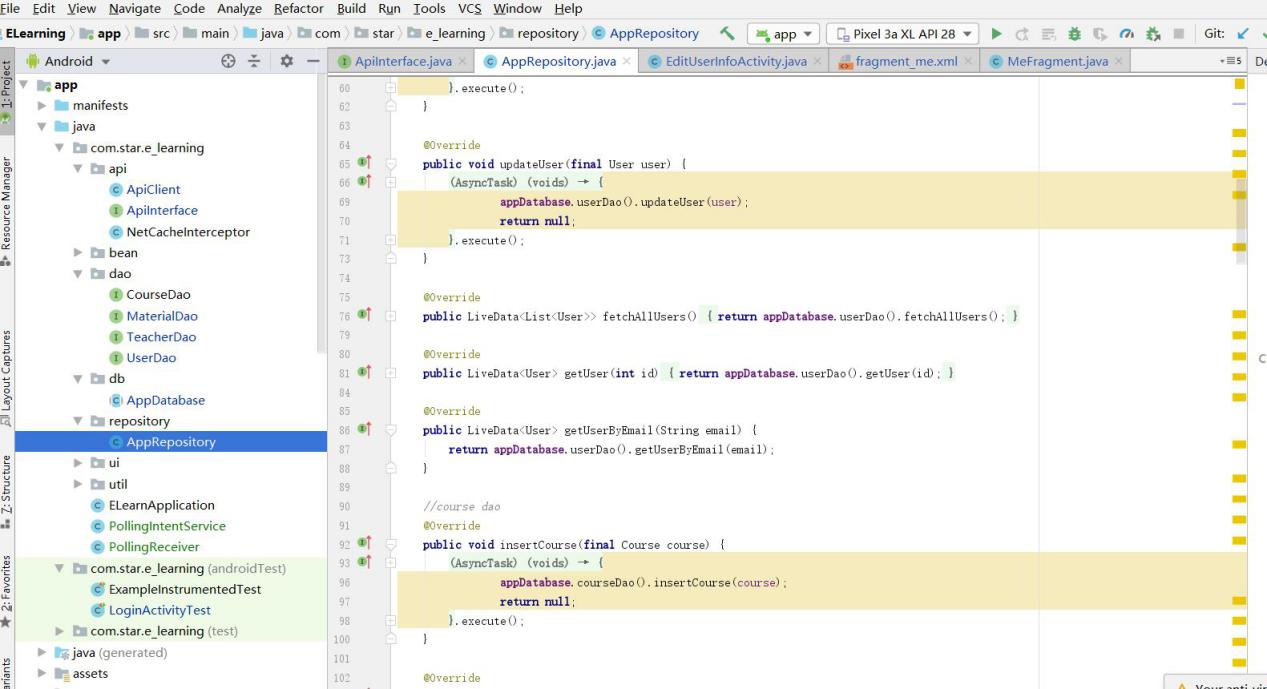
1. 这个界面采用collapsingtoolbarlayout以及下面的tablayout，分别控制两个fragment，左边展示的是与该课程有关的老师列表（recyclerview），右边是材料列表，右边采用multitype的recyclerview,材料支持视频、音频、文件、图片等多种格式，可以进行播放
2. 类似课程，可以刷新，同时会保存实体类以及视频、音频等文件
3. 该界面也可以进行分享（点击右上角）
4. 没有网络时，会显示本地缓存的数据（room与local file）
5. 采用livedata，当数据改变时ui会变



写了一个轮询intentservice（PollingService），定时从服务器获取课程并广播，brocastreceiver接收(PollingReceiver)，跳转至主界面更新ui。

四、Mvvm设计

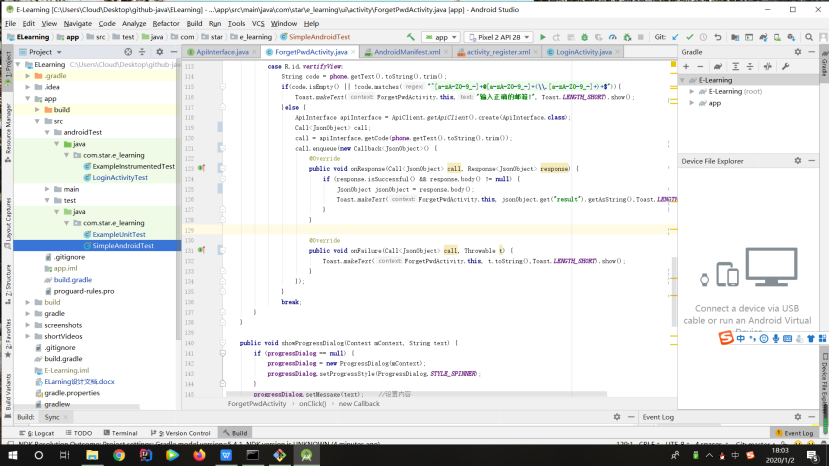
采用roomapi+livedata实现，具体实现见repository,dao,db三个包



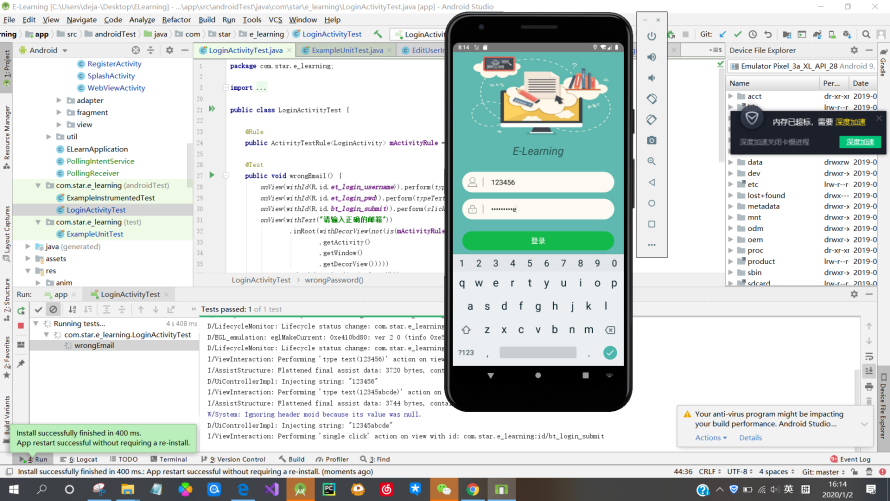
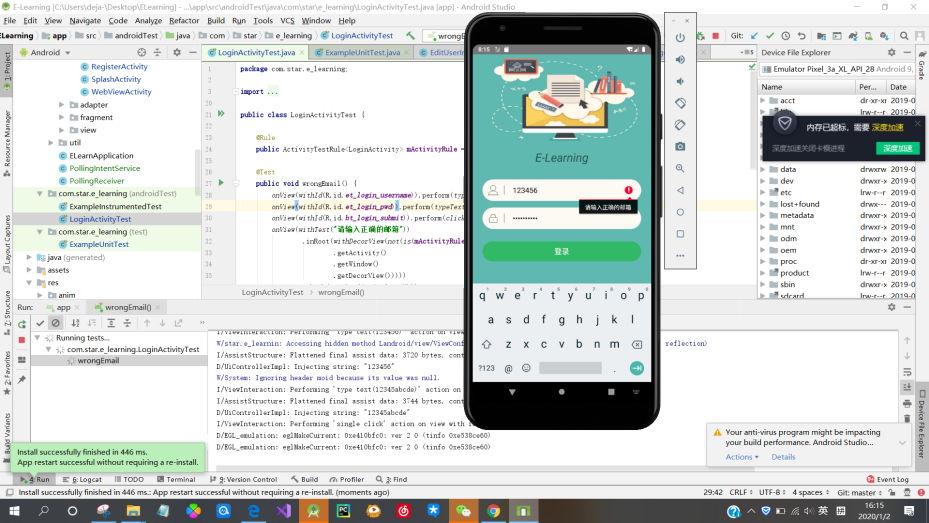
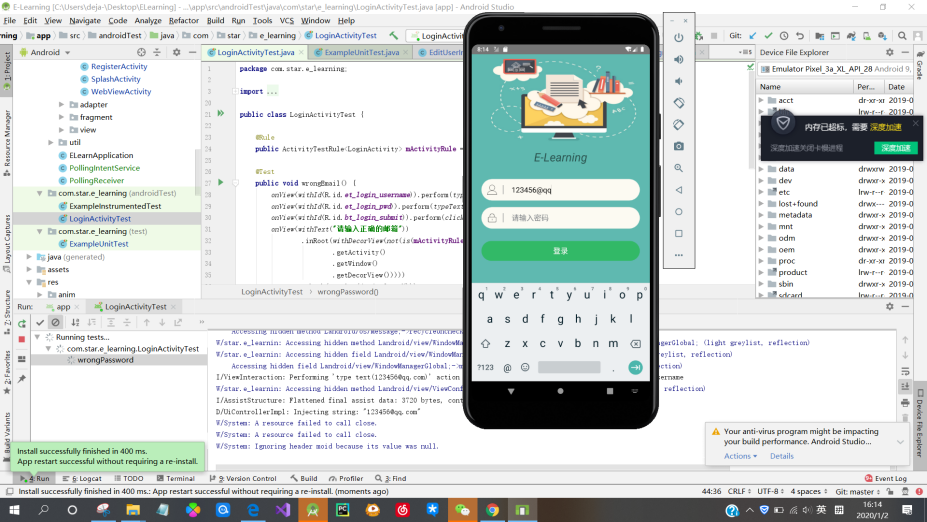
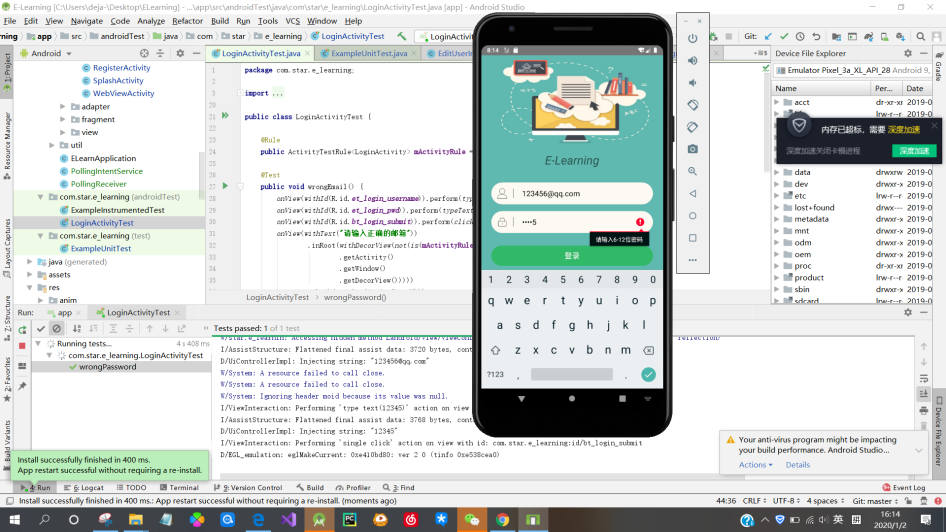
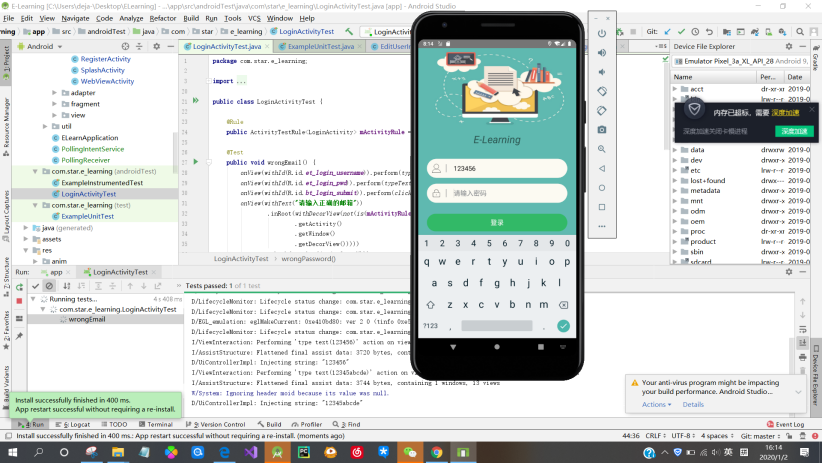
五、测试（由于真机不能做测试，此部分用的模拟机）：

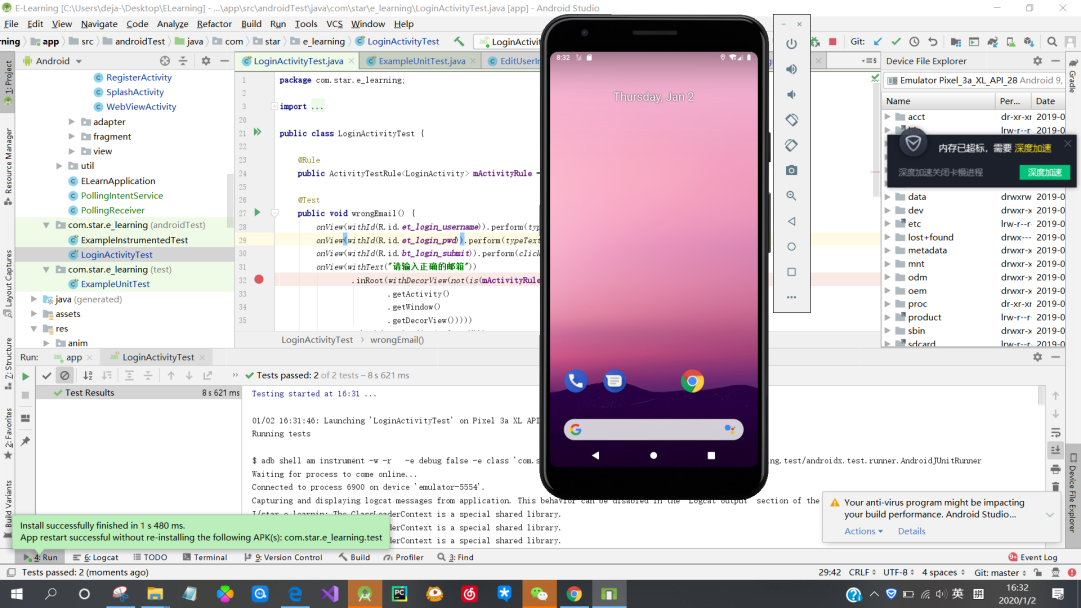
两个UI测试+1个简单的unit test

LoginActivityTest + SimpleAndroidTest

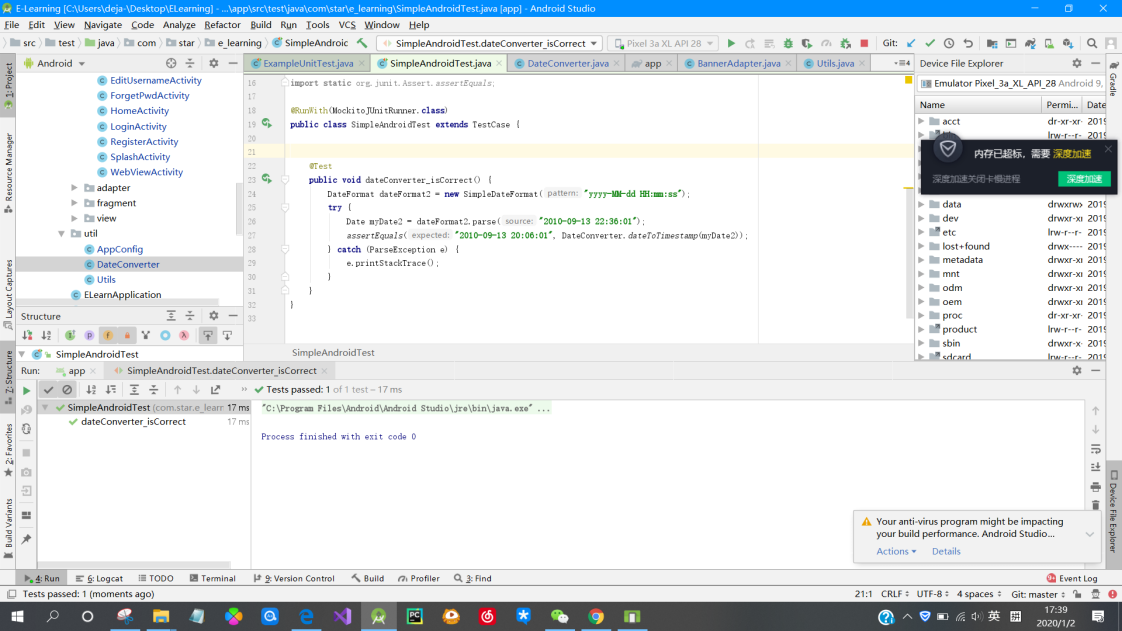


分别测试输入错误的邮箱和错误的密码是，提示的信息是否与预期一致



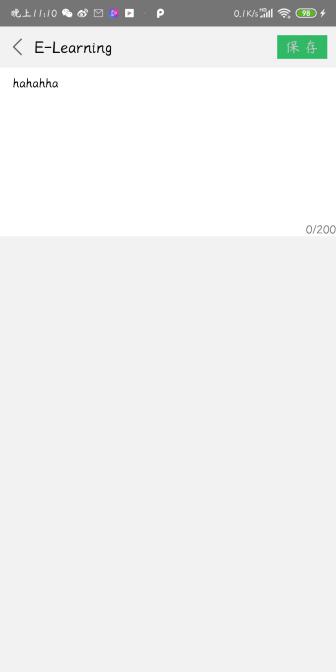
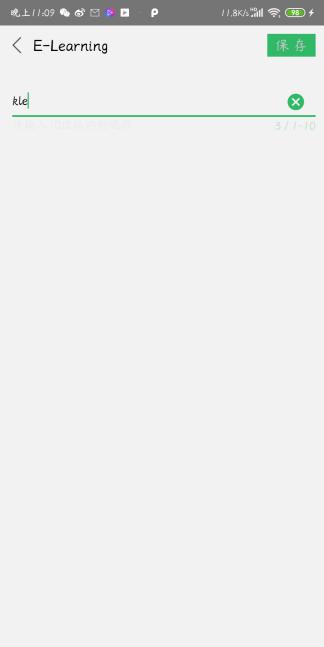
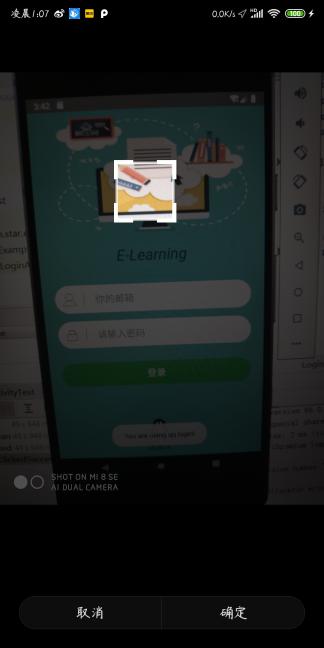
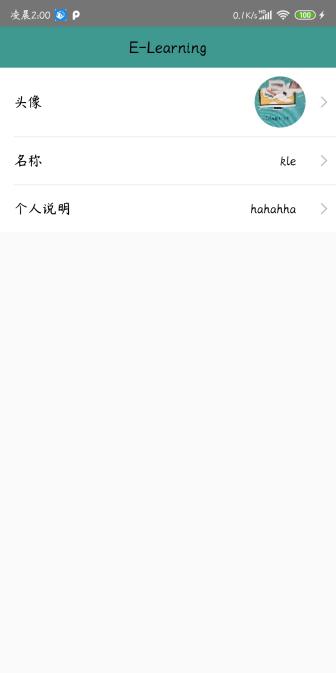


测试日期转换是否和预期一致



六、额外功能：

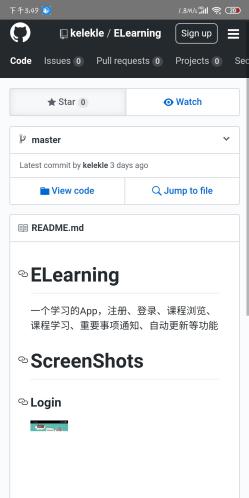
1. 个人信息修改

可以修改个人描述，头像以及用户昵称

1. 关于我们

会打开一个webviewactivity跳转至本项目的url



1. 退出

更改当前登录状态，结束当前活动，跳转至登录界面

1. Package and class design

下表是本系统的的各个包的设计说明：

|  |  |
| --- | --- |
| **包名** | **设计说明** |
| **com.star.e\_learning.ui.adapter** | **该包主要存放控件的适配器。** |
| **com.star.e\_learning.ui.activity** | **主要存放所有具体活动** |
| **com.star.e\_learning.ui.fragment** | **该包中主要存放了所有的碎片** |
| **com.star.e\_learning.bean** | **存放了所有的实例对象** |
| **com.star.e\_learning.ui.view** | **自身实现的简单的view控件** |
| **com.star.e\_learning.api** | **网络请求** |
| **com.star.e\_learning.util** | **工具类** |
| **com.star.e\_learning.dao** | **dao接口** |
| **com.star.e\_learning.db** | **room数据库类** |
| **com.star.e\_learning.Repository** | **我的viewmodel，通过数据库异步获取数据** |