# Android期末文档

1. **本学期通过这门课学到的知识**

通过本学期选修Android移动开发，使我学到了Android开发所需要的知识，比如最重要的**Activity的生命周期：**

1.启动Activity：系统会先调用onCreate方法，然后调用onStart方法，最后调用onResume，Activity进入运行状态。

2.当前Activity被其他Activity覆盖其上或被锁屏：系统会调用onPause方法，暂停当前Activity的执行。

3.当前Activity由被覆盖状态回到前台或解锁屏：系统会调用onResume方法，再次进入运行状态。

4.当前Activity转到新的Activity界面或按Home键回到主屏，自身退居后台：系统会先调用onPause方法，然后调用onStop方法，进入停滞状态。

5.用户后退回到此Activity：系统会先调用onRestart方法，然后调用onStart方法，最后调用onResume方法，再次进入运行状态。

6.当前Activity处于被覆盖状态或者后台不可见状态，即第2步和第4步，系统内存不足，杀死当前Activity，而后用户退回当前Activity：再次调用onCreate方法、onStart方法、onResume方法，进入运行状态。

7.用户退出当前Activity：系统先调用onPause方法，然后调用onStop方法，最后调用onDestory方法，结束当前Activity。

**Fragment局部刷新：**

Fragment需要嵌套在Activity中使用,当然也可以嵌套到另外一个Fragment中,但这个被嵌套 的Fragment也是需要嵌套在Activity中的,间接地说,Fragment还是需要嵌套在Activity中!! 受寄主Activity的生命周期影响,当然他也有自己的生命周期!并且不建议在Fragment里面嵌套Fragment因为嵌套在里面的Fragment生命周期不可控

Fragment的生命周期和Activity有点类似:三种状态:  
Resumed:在允许中的Fragment可见  
Paused:所在Activity可见,但是得不到焦点  
Stoped: ①调用addToBackStack(),Fragment被添加到Bcak栈②该Activity转向后台,或者该Fragment被替换/删除

在最开始学习Android开发的时候，感觉是前后端开发合在一起了，前端写在res文件夹里面，后端代码写在main里面。在进行实践一作业时，让我学习了Text，EditText，多选框和单选框这些基础组件的运用，并且学会了使用按钮来让输入的数据输出出来，通过监听Button，使得一按按钮就执行按钮放入的函数。实践二作业是让我收获颇丰的作业，我做的是基于杭电助手的课表的接口，来进行解析请求接口来的信息，最后在界面上面展示出每日课表，需要获取个人token来请求杭电助手接口来获取课表信息，再使用Gson库来解析json数据，并且把数据放入sqllite3数据库，为了观察sqllite3里面数据的变化还在网上找了一个可视化数据库的库。用户在修改课表内容修改到一半退出时activity处于Pause时就将写到一半的数据放入sharpreferene里面，当acitvity又处于start时就将数据拿出来。为了使用ListView来显示，需要编写Myadapter来编写ListView中每一条的样式。

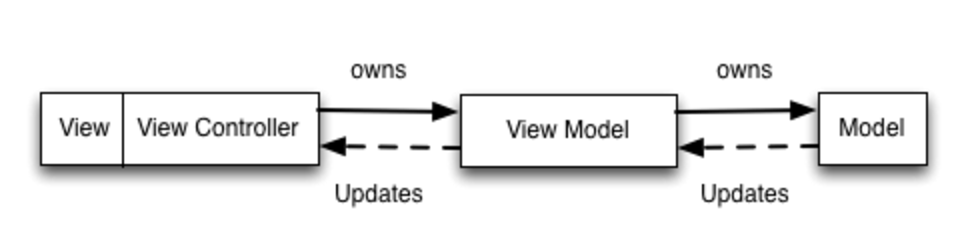
1. **大作业介绍**

最后的实践大作业，我选择了Flutter作为开发语言来开发杭电助手APP，因为Flutter可以一份代码编译出安卓的APK和苹果的app，苹果使用Xcode cloud来持续集成和发版，用testflight发行来为测试者测试。虽然不是上课学到的东西，但是也是本学期大作业学到的，所以讲讲我这个大作业学到了什么东西。最基本的是基于Dart语言的Flutter框架的语法学习，flutter开发一个很好用的功能就是能够实现热重载，秒级呈现修改后的效果。

热重载的流程可以分为扫描工程改动、增量编译、推送更新、代码合并、Widget重建5个步骤：

* 工程改动。热重载模块会逐一扫描工程中的文件，检查是否有新增、删除或者改动，直到找到在上次编译之后，发生变化的Dart代码。
* 增量编译。热重载模块会将发生变化的Dart代码，通过编译转化为增量的Dart Kernel文件。
* 推送更新。热重载模块将增量的Dart Kernel文件通过HTTP 端口，发送给正在移动设备上运行的 Dart VM。
* 代码合并。Dart VM 会将收到的增量 Dart Kernel 文件，与原有的Dart Kernel 文件进行合并，然后重新加载新的Dart Kernel 文件。
* Widget重建。在确认Dart VM 资源加载成功后，Flutter 会将其 UI 线程重置，通知Flutter Framework 重建Widget

还有就是团队合作，使用git作为版本控制工具，github作为代码管理工具，团伙合作是规划好谁做什么部分，防止重复做，每个人做完整功能的话就建一个分支，写好功能和单元测试（比如接口解析成类的测试）然后提pull request，有负责人进行代码审查。



图表 1：MVVM架构

项目结构以MVVM（model-viewModel-View）作为项目结构，Model就是进行解析的承载对象，View是页面展示，model和View进行通信时需要通过viewModel，举个例子是viewModel进行请求接口并且通过model来解析接口信息并且把model数组传递给View并进行展示出来。

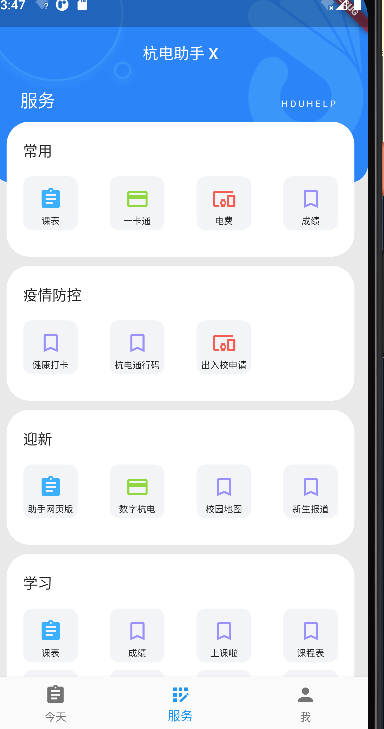
**2.1、功能介绍**

界面修改：根据设计部给出的设计来制作界面，展示老新界面的区别

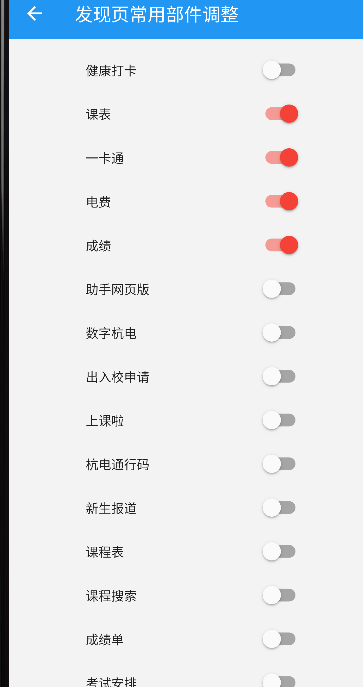


第一是优化了界面的美感，第一行是温度，天气和时间信息，把最重要的时间信息放到中间，下面一行是三个高频使用功能，再下面是今日明日课表，成绩，空教室三个按钮你可以根据需求来看相应的信息，首页请求infostream这个接口来获取全部的信息，获取数据的需求需要全部写出接口形式这样来减轻兼容APP端和web端的压力。

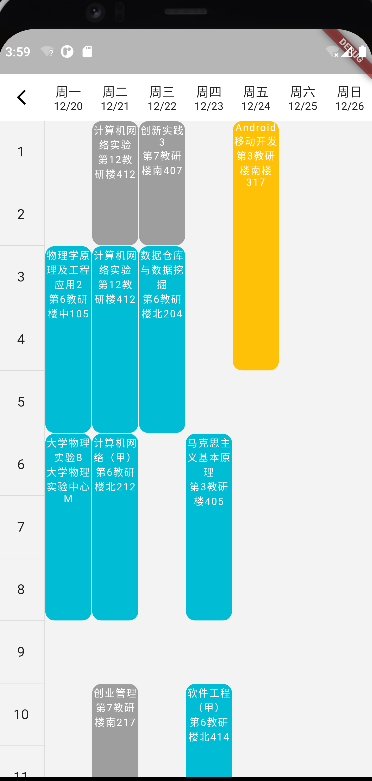
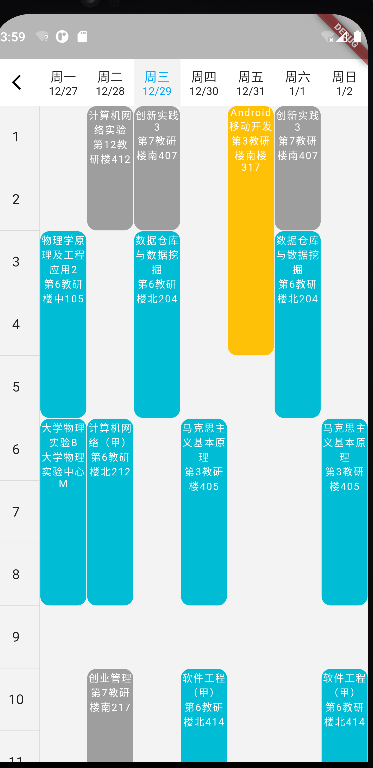
第二是加入了服务界面，服务界面放置了所有杭电助手的应用或者外部应用。



很多界面本质上是一个web链接，点进去是web界面，然后真正高频使用的就写成原生的来增加流畅度。有一个创新点是常用界面，用户可以自己选择自己喜欢的高频应用放入常用界面，虽然这个界面很丑不优雅，但是能够以一种很简单的方式实现个性化常用功能的实现。



第三是我重写了详细课表界面，通过左右滑动获取上一周或者下一周的课表，基本的课表功能实现了，但是美观和显示特定周课表功能还有欠佳，之后还需要继续改进。课表使用的新接口，老接口是一节课一个数据，但是会有一种普遍的情况是好多课是两个老师合作上的，就会有两节课，其他信息相同但是上课老师不同。新课表V2接口就解决了这个问题，把两节课合在一起，然后把两个老师的信息都放入同一节课中。



1. **在实践的过程中遇到的问题及解决办法**

在进行实践二时，导入了Gson和可视化sqllite3数据库两个包，但是这两个包同时导入会引起冲突，因为sqllite3里面已经有了Gson库，如果我导入Gson和可视化sqllite3数据库两个包的话会起冲突，显示两个Gson库版本不同引起冲突，解决办法是引入sqllite3包时不解析它内在的Gson库。

进行flutter开发时需要处理V2的课表接口，使用了一节课集成了多个老师，多个课时信息的课，这样会有一个问题，一节课需要周二上一次，周四上一次，这样如果课想要根据时间排序的话根本不能排序，所以就不考虑上课啦那种只显示一周的一天的那种模式，我们选择了直接展示一周的课的课表展现形式，把接口response解析成class之后，根据时间放入想要的表格里面。还有一个课表判断很复杂的情况是判断是否是本周的课，接口中有每节课的一学期的上课时间周，需要每节课都判断一下是不是属于这周然后再显示出来。

Fluuter请求接口是异步方式请求，通过async开启一个异步操作，通过await等待请求或其他操作的执行，并接收返回值。当数据发生改变时，调用setState方法并更新数据源，Flutter会更新对应的Widget节点视图。使用异步更新会导致一种情况的出现，就是点击界面的时候你还没解析好接口的信息，需要用到接口信息时就会导致报错，解决办法是合理设置解析完成的标准，使用数据前要判断是否解析完成。

服务中每个界面都需要配置免登功能，可以单纯从url上面配置，但是每个界面都需要配置免登就会很麻烦，而且拓展性很差，接入一个新功能都需要重新配置这个服务的免登，助手网页端早有成熟的解决方案，就是请求一个discover接口，接口中会告诉你这个服务是由什么协议来实现免登的，有些是通过oauth，有些是url后面加token来实现。

1. **总结与展望**

这学期选修了赵丽娜老师Android课，使我不仅收获了Android开发所需要的基础知识，比如Andiroid开发最重要的Activity生命周期和Fragement局部刷新，还为我之后使用flutter合作开发杭电助手app垫下了坚实的开发思想和理论基础。

在之后的课程里面，希望赵丽娜老师课讲的越来越好，也希望我自己这个app能够在毕业前合力维护下去，能够根据需求进行更新迭代，更好的服务杭电师生。

凑字数凑字数凑字数凑字数凑字数凑字数凑字数凑字数凑字数凑字数凑字数凑字数凑字数凑字数凑字数凑字数凑字数凑字数凑字数凑字数凑字数凑字数凑字数凑字数凑字数凑字数凑字数凑字数凑字数凑字数凑字数凑字数凑字数凑字数凑字数凑字数凑字数凑字数凑字数凑字数凑字数凑字数凑字数凑字数凑字数凑字数凑字数凑字数凑字数凑字数凑字数凑字数凑字数凑字数凑字数凑字数凑字数凑字数凑字数凑字数凑字数凑字