

2023 年夏季 Java 小学期大作业

方政 2022012141

1 代码结构

该项目的目标是实现一个用于新闻浏览的新闻 app。其中，后端需求的新闻获取、搜索等功能由清华大学计算机系知识工程实验室进行，因此，本人主要通过提供的网络接口实现前端功能。本项目通过 android studio，采用 Java 语言进行开发。开发过程中参考了部分课上所学内容以及互联网获取的其他 android 开发相关知识。

从功能上来说，本项目主要可以分为五个模块：首页新闻列表，新闻详情展示，新闻分类以及类别管理，新闻搜索，用户管理、新闻收藏和历史记录。其中不同模块之间部分功能需求类似，比如新闻按列表展示和按详情展示，在主页面、搜索页面、收藏和历史记录页面均有使用，因此这部分代码可以进行复用，只需要依据不同功能的需求进行适当改动即可。下面将进一步叙述各个模块的结构以及大致的实现逻辑。

首先是最为核心的首页新闻列表模块。该模块可以拆解为前端的控件视图展示和后端的新闻数据获取两部分，同时，我也使用了一系列其他人开发的成熟框架以减小工作量。在前端控件的选取上，我使用了 SlidingTabLayout 布局嵌套 ViewPager，ViewPager 下嵌套多个自定义的 NewsFragment 以实现不同的类别的新闻列表的展示，同时每个 NewsFragment 中通过加载 RecyclerView 以实现两种新闻 item 的展示。在后端的新闻数据获取上，我使用了 Retrofit 网络框架来向接口申请数据。为此，我自定义了相应的 api 类和接口方便在开发过程中修改申请参数。在对应的 ui 类中，我通过自定义 getNewsList 函数并定义 call.enqueue 方法实现网络请求和数据处理，最后再通过自定义 adapter 类将数据加载到视图中。此外，我还使用了 SmartRefreshLayout 框架以实现刷新和拉取功能。

在新闻详情展示模块中，我实现了两套 layout 布局，分别对应没有视频的新闻和有视频的新闻，并通过在 activity 初始化时判断新闻有无视频来绑定相应的 xml 文件。新闻详情内容设置在 scrollview 中以实现内容的滚动浏览，并定义收藏按钮以便于收藏功能的实现。视频则使用 google 官方提供的 videoview 进行加载。

在新闻分类以及类别管理模块中，我在主页面中定义了两个 list 分别存储展示和未展示的分类。此外，我自定义了继承于 BottomSheetDialogFragment 的 Fragment 类以实现类

别管理系统。同时，我也使用了 `FlowLayout` 布局以创建不同的类别按钮，并在点击时进行相应的处理。至于具体的类别展示，在第一个模块中已有阐释，在此不再赘述。

在新闻搜索模块中，我定义了四个 `edittext` 视图以接受用户输入的不同参数，并通过复用首页展示新闻列表和详情的方法来获取和展示搜索结果。这一部分的实现逻辑大部分与首页相同，整体较为简单。

在用户管理、新闻收藏和历史记录模块中，我首先额外实现了用户登录和注册系统，使 app 的功能更贴近于市面上的新闻 app。用户数据则储存在数据库中，为了便于操纵，我自定义了 `MyDataBaseHelper` 类来对数据库进行操作。其中展示新闻列表和详情的方法依然沿用之前实现的 `NewsActivity` 和 `NewsAdapter`，区别在于显示的时候需要现在数据库中查找而非网络申请，并且我还增加了一键删除功能，便于使用。

最后，我使用了 `tablayout` 将三个相对独立的 `Fragment` 插入 `HomeActivity` 中，实现不同页面的切换。

2 具体实现

基于上述的项目结构，下面我将详细阐述整个项目的细节和实现难点。

从文件分类上来说，为了代码的维护性和可读性，我将其分为了 `adapter`，`bean`，`ui`，`utils` 四个部分，`adapter` 用于实现相关的 `adapter` 类，`bean` 则用于存放实体的数据类，`ui` 则是各个 `activity` 和 `fragment` 的实现，`utils` 则封装了一系列工具类，如数据库操作、状态栏设置等工具。

在首页新闻列表模块中，首要解决的问题就是新闻数据的获取。首先，为了便于数据的管理，我创建了 `bean` 并实现了 `NewsData` 类用于封装每条新闻的各个参数。而在实现网络请求的过程中，通过收集资料和学习，我了解到传统网络请求方法的弊端，并选取了较为实用和热门的 `Retrofit` 框架进行网络请求，`Retrofit` 框架封装的一系列方法省去了许多繁复的步骤，并且规避了网络请求过程中多线程的问题，减小了程序崩溃的可能性。而主页实现的第二个难点就是如何将不同类别的新闻列表展示出来，实现类似今日头条的切换类别的功能。我采用的是另一个热门的框架 `slidingtablayout`，这个框架集成了 `scrollview` 和 `tablayout` 的功能，再用 `viewpager` 将不同的 `fragment` 插入其中就能实现新闻类别的切换效果。主页新闻的刷新和拉取依旧使用了现成的 `smartrefreshlayout` 框架，再设置相应的监听器用于

实现刷新功能即可。此外，为了用户体验良好，我在加载 viewpager 时设置了自动刷新动画，这样能防止每次创建主页时没有加载的反馈。

新闻详情页总体的功能较为简单，其中的一个难点是实现视频播放功能。由于 videoview 在默认状态下难以隐藏，因此，我使用了两套 layout 布局，在加载布局使先判断 newdata 的 video 参数是否为空，即新闻是否包含视频，再选择用哪一套布局。视频播放器的实现则较为简单，无需赘述。另一个难点是新闻数据如何传递。我使用了 intent 的 serializable 方法向 intent 中加入数据，再进行传递。

新闻分类及类别管理模块的难点除了上面讲的不同新闻列表的切换，还有新闻类别的增删以及动态特效。我参照今日头条的结构，实现了两个 flowlayout 用于分别储存显示和不显示的类别，在点击的时候实现两个 list 之间数据的变化，最后即使重新加载即可实现动态特效，在退出分类管理对话框的时候将类别 list 传递给主页面，最后再让主页面更新即可。此外，为了美观和实用性考虑，我没有重新创建一个 activity，而是在主页面中插入了一个 BottomSheetDialogFragment，这样以弹窗的形式实现界面，显得较为合理且美观。

得益于实现主页时较为良好的代码结构，在实现搜索功能时，我直接复用了新闻主页的部分方法，只需要将申请参数改为用户输入的参数即可。

本地管理系统涉及到数据存储的问题。一方面，我主要采用了 sqlite 数据库以及 DataBaseHelper 来实现数据库相关的操作；另一方面，我采用了 sharedpreference 来存储用户的登录状态，避免重复登录。历史记录和收藏的总体结构大致相同，在用户点击新闻的时候向数据库的 History 表插入数据，在点击收藏时向 Collection 表增删数据即可。而阅读过的标题变灰只需在显示之前判断是否在数据库中即可，由于不涉及图片和视频的下载，整体逻辑也较为简单。

在基础功能之外，我还做了许多额外功能和优化工作，让 app 使用体验更为良好。在整体布局上，我采用了一致的色调与简约的布局，使得 app 整体较为美观。此外，我加入了动态启动动画，进一步改善了用户体验。在用户界面，我实现了账号和头像的设置，并且支持保存用户登录状态，每一次重新启动 app 使自动调用，避免重新登录。在历史记录和收藏界面，我添加了删除按钮，用户可以清除历史记录和收藏列表。在新闻浏览时，我设置了每次拉取新闻使会根据上一条新闻的时间自动获取不同的新闻，避免了新闻列表拉取时反复获取重复新闻，提升用户体验。同样的，在搜索页面我也进行了如上设置。

3 总结与心得

新闻 app 的开发过程是十分复杂的,在小学期不算很长的时间里完成上述工作还是相当有难度的。在项目开发的初期,我对于安卓软件的开发过程不甚了解,对于虚拟机的安装,android studio 的各项功能使用也不清楚,没有注意到 locate 窗口中能够查看 app 运行信息。因此,在前期遇到闪退、程序崩溃的问题时,我常常难以找到错误的原因。在编写前端 UI 的过程中,我经常不了解各个控件的用法,需要不断在网上寻找教程和资料。但是随着开发和学习的不断深入,我逐渐熟悉了 Java 语言和安卓编程,开发进度不断加快。

总体而言,我从小学期的项目开发过程中得到了以下几点经验:首先是要对项目的整体结构和开发顺序有个整体的规划,避免在开发过程中项目难以推进而被迫重写一些功能。其次,要善于利用网上的教程以及现有的成熟框架,减小不必要的时间和经历消耗。最后,要经常性地测试项目的各个功能,避免开发过程中软件闪退时难以找到问题的根源。

最后,感谢老师和助教对 Java 课程小学期的付出。你们的付出,让我们收获了一次宝贵的 Android 开发体验,也让我们锻炼了各项技能,学到了很多知识。