

北京理工大学计算机学院

《Android 技术开发基础》课程

班级__07111701__ 学号__1120171185__ 姓名__张静怡__

1 App 的运行与开发环境

- (1) **运行环境：** Android 9.0 的 Android 手机/平板
- (2) **部署方法：** 直接安装 app_release.apk 即可

2 程序开发平台

- (1) **开发环境：** Android Studio 3.3.1
- (2) **手写代码行数：** java: 3817 行, xml: 1866 行

3 App 功能说明

3.1 登录界面

本软件设置登录功能，包括保存账号密码的功能自动登录的功能。同时若邮箱或者密码输入错误，自动弹出错误提示并且清空输入栏，自动保存也自动取消。登入后进入当前用户专属记录状态。

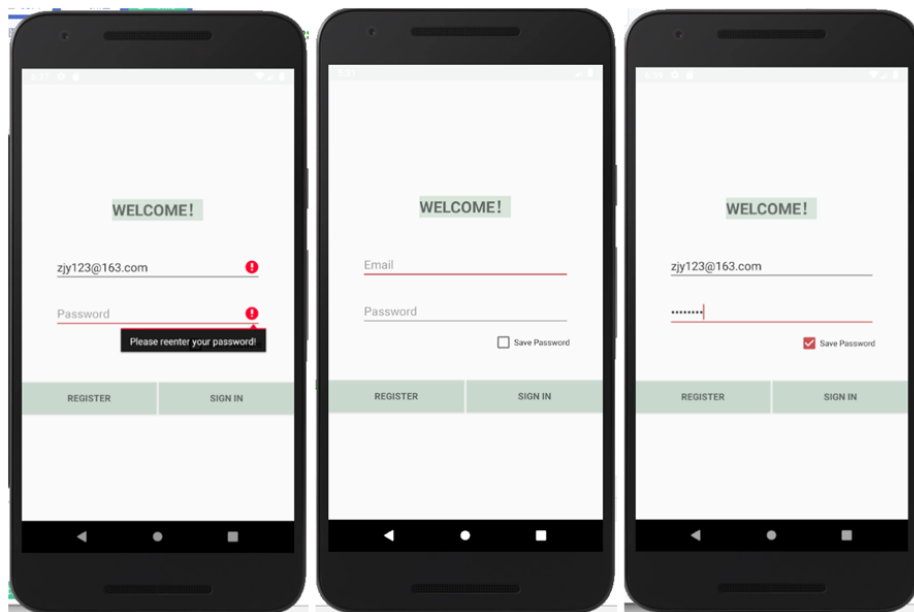


图 3.1.1 登录界面、登录报错及自动登录

3.2 注册界面

在登录之前，若未注册，则需要先注册。其中需要填入的信息有：

1. 注册邮箱
2. 密码（两次输入确认）
3. 手机号（发送验证码功能待完善，但已经加入权限）
4. 验证码（看图生成验证码，验证码图片为自动生成），点击图片切换新的验证码



图 3.2.1 注册界面及两次输入不一致产生的重新输入提醒

3.3 主页界面

主界面氛围上下两部分，上部分之中又有三部分：体重记录块、当日饮食记录块、运动块。

下部分则为导航栏，选项为：主页、当日饮食记录添加、体重记录。

上部分的功能详细如下：

1. 体重记录块

1. 显示用户当前体重，以及用户设定的体重目标。
2. 视图的最中间部分为一个自定义的弧形 ProgressBar，用来显示体重改变目标的完成百分比。
3. 点击该块，直接设定目标体重，同时自动以当前体重为起始体重，改变体重改变计划，主界面的信息也会即时更新。

2. 饮食记录块

1. 当前饮食计划下，每日自动设置的饮食摄入标准。根据每日三餐饮食摄入记录，自动计算摄入卡路里的剩余值
2. 三个 ProgressBar 分别记录了每一餐的摄入，根据自动设置的每一餐推荐摄入标准，显示该餐是否摄入过量
3. 点击该块，可以跳转至饮食记录显示页。
4. 饮食记录每天清零

3. 运动块

1. 按钮转入运动视频下载界面

导航栏功能如下：

1. 主页
 1. 停在主页
2. 记录当日饮食
 1. 转入饮食记录添加页面
3. 体重记录
 1. 弹出底部弹窗，弹窗中有标尺，用于记录用户当前体重。并且记录之后主界面的显示即时更新。



图 3.3.1 主页上部分三个主模块，以及底部导航栏



图 3.3.2 用户当前体重设定以及目标体重设定

3.4 饮食选择页

上部分为一个食物列表，每一个食物 Item 之中包括食物的名字、每 100g 的热量，以及根据热量显示的热量的高低值评估显示——红黄绿三个级别。点击一个 Item，即弹出底部弹窗，用标尺选择食物摄入量，将其加入用户的饮食记录之中。

底部为选择栏。选择当前食物是加入早中晚的哪一餐。

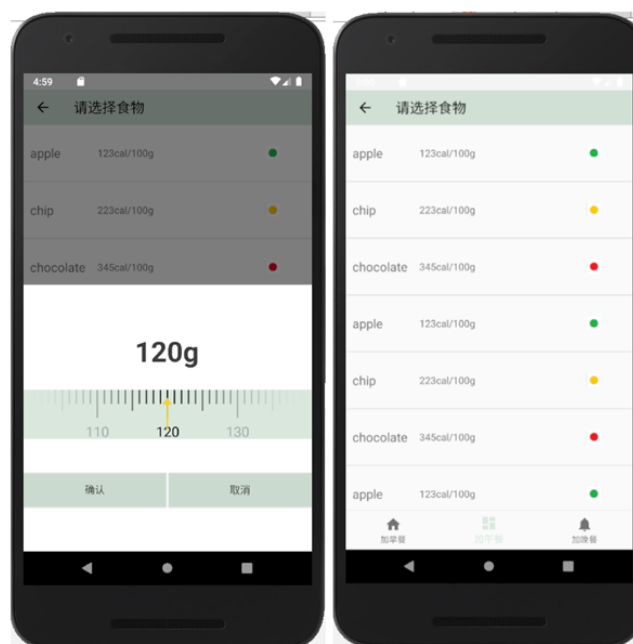


图 3.4.1 饮食记录添加底部弹窗以及食物库

3.5 饮食记录显示页

该部分分为上半部分的 listview 显示和一个悬浮按钮。

上部分的饮食记录显示页会分别显示三餐摄入的食物与摄入量。悬浮按钮则直接跳转至食物选择页，使用户继续添加饮食记录。

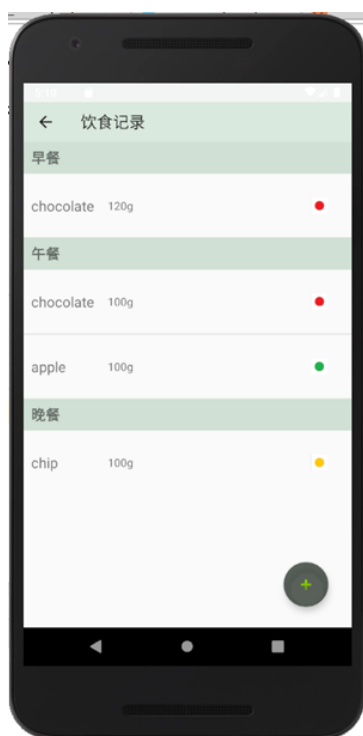


图 3.5.1 饮食记录页

3.6 运动视频下载页

该页由一个视频列表组成。列表中的每一个 Item 包括该视频的头图、视频的名称，另有两个按钮分别对应视频开始下载和取消下载。下载路径为特定网址。

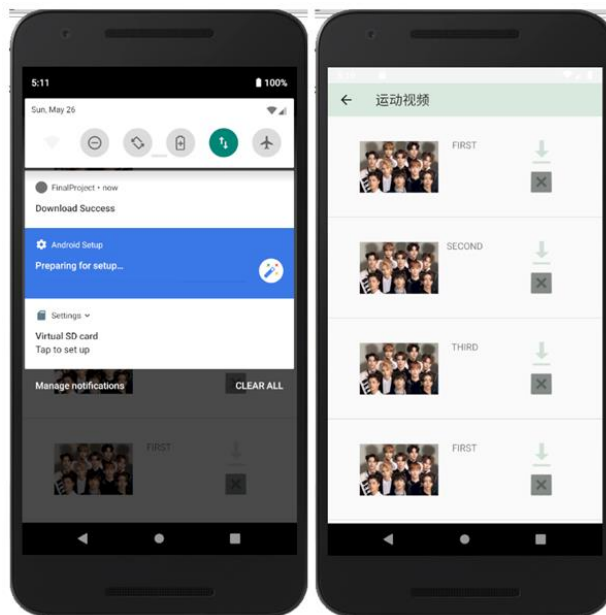
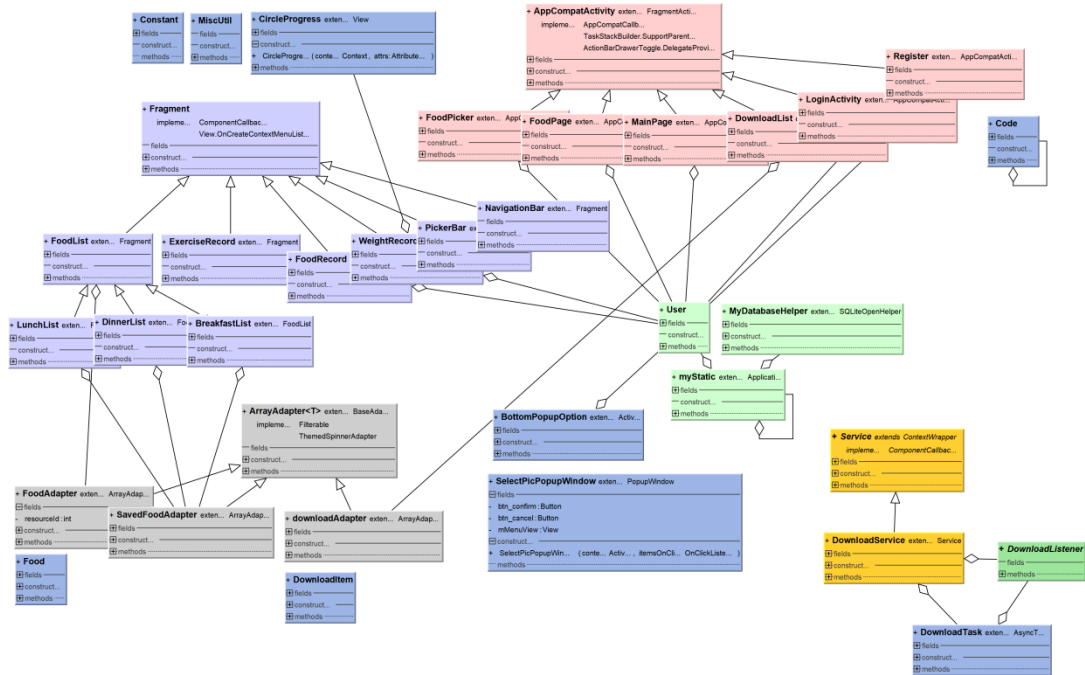


图 3.6.1 下载界面以及后台下载提醒

4 App 架构设计及技术实现方案

整个文件分成了 6 个 package:

1. Activity: 包含所有活动界面
2. Data: 存储常量和全局变量
3. Download: 下载相关类
4. Database: 数据库相关
5. Fragment: 所有碎片的组件
6. Module: 自定义控件



4.1 主功能结构上的实现：继承

由图可见，主要运用了三个继承 Fragment 类，AppCompatActivity 类，ArrayAdapter 类来实现，其中 FoodList 类有有有三个子类。

Fragment 实现控件

1. ExcerciseRecord, FoodRecord, WeightRecord 为主页的三个部分的碎片
2. FoodList 为一个带有标题的食物列表，可在三餐的记录中进行复用
3. PickerBar, NavigationBar 为底部导航控件

AppCompatActivity 实现界面

包括登录注册界面，主页，食物记录页 FoodPage，下载页 DownloadList，食物选择页 FoodPicker

ArrayAdapter 实现 ListView

三个子类分别用于实现食物选择 List，食物记录 List，以及下载内容 List

4.2 数据上的实现

由图可见，以下三个类组成了数据实现的中心。

User 类

用于记录当前用户的全部信息，包括登录信息，饮食记录信息，体重信息。

在软件运行过程中较为独立地与各界面与控件进行信息交互。

myStatic 类

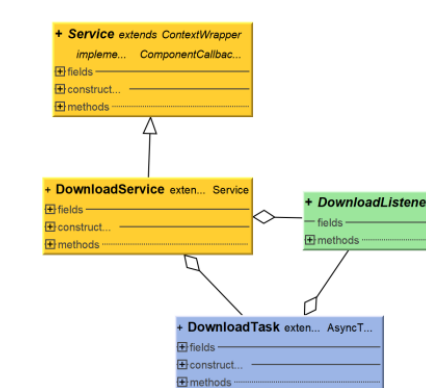
继承 Application，用于保存所有全局变量，为整个软件数据的存储核心，其中以 User 类对象 tester 为主。

myDatabaseHelper 类

进行数据库的连接和管理。

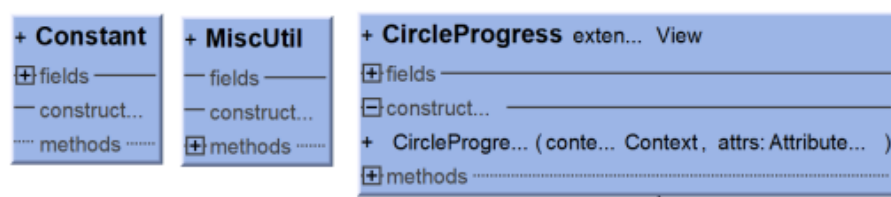
4.3 特定功能的实现：下载功能

DownloadService 实现后台，DownloadTask 创建下载任务，其中用 DownloadListener 进行监听

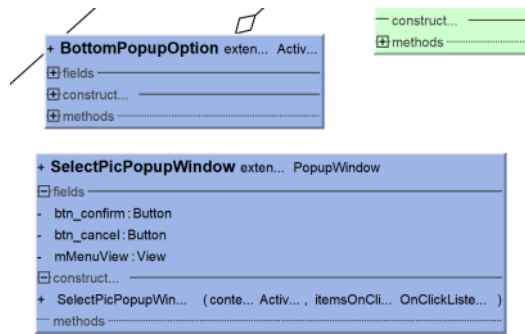


4.4 细节和多方式显示实现：自定义控件

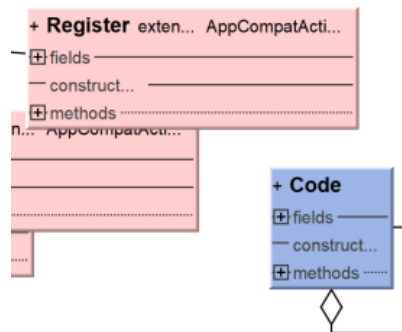
1. CircleProgress 实现弧度进度条显示，Constant 和 MiscUtil 类动态设定显示样式



2. BottomPopupOption 实现底部弹窗，其中 SelectPicPopupWindow 进行动态设置样式



3. Code 实现验证码功能辅助于注册界面



5 技术亮点、技术难点及其解决方案

5.1 本程序的亮点

1. UI 界面设计简洁，清新
2. 使用了多组 progressBar 来使数据清晰化，更直观
3. 将健康管理多方面综合，达到全方位的健康监督和体重管理功能
4. 管理配合登录账户，可多人使用

5.2 本程序的技术关键点

1. 应用了 MVC 模式来设计整个程序的架构
2. 使用 SQLite 数据库存储诸多用户多日基本信息记录和饮食记录，并采用多条件查询及添加操作实现诸多功能
3. 使用 SharedPreferences 实现自动登录记录密码功能
4. 使用 fragment 来实现功能复用
5. 用自定义控件来实现多方面交互和直观显示
6. fragment、自定义控件和活动之间的接口来实现活动总界面与多个控件之间的信息交互和同步
7. 全局变量的应用使得数据使用更方便
8. AsyncTask 和 Service 的应用实现下载任务以及后台运行提醒

- 9. ListView、Container 的单独、混合使用实现多方面列表
- 10. ListView 自定义视图高度显示
- 11. 活动生命周期的重点理解和应用
- 12. 多种 View 控件的使用

5.3 遇到的技术难点及对应的解决方案

1. 装配问题

解决方式：大量查阅资料、进行实践。

2. 用户信息保存问题

解决方式：

对于整个程序运行过程，存在一个 User 类对象 tester。选择用全局变量保存其信息，将其放入一个自定义的 myStatic 类之中，最初出现空指针的问题，后用 singleton 解决，使得当前用户信息可以在所有活动和控件之间共享。

后与登录注册功能联合，实现多用户使用的功能，并且带有保存当日饮食记录，隔日即清零重记的功能，最终用用户基本信息 UserInfo 表与 Breakfast、Lunch、Dinner 三张表记录每个用户在每个特定日期下的饮食记录。在登录后把当前登入用户信息赋给全局变量 tester 方便其在当前应用中使用，例如在数据显示、数据计算上使用，但是当数据库表中信息被修改时也要同步更新数据库信息。

最后选择了用 SharedPreferences 技术实现密码记录功能。

3. 在 RulerView 和 CircleProgressBar 之中的信息不能同步在主页的体重记录块之中更新

解决方式：利用自定义控件和 Activity 之间的回调方法解决

4. ListView 不能默认达成展开的状态

解决方式：自定义函数设定适应该 list 的 item 个数的 ListView 大小

6 简要开发过程

2019.4.25-4.27

做基本设想，并作出界面、功能、界面跳转、控件跳转的设计图

2019.4.28-5.1

基本实现饮食记录 ListView 点击添加功能，BottomPopupWindowView 弹出实现，进行标尺条的操作，实现饮食记录加入计算以及取消回收操作，弹窗界面修改

2019. 5. 2-5. 5

建立视频下载 listview, 进行少量修改, 完成 downloadlist, 并对 navigationBar 进行基本设置, 同时统一整个 app 的 theme

2019. 5. 13

饮食记录页和底部弹窗相关界面实现

2019. 5. 15

学习 activity 声明周期相关, 并实现界面信息状态保存

2019. 5. 16

切换动画的设定, 尝试滑动切换但后来失败, 尝试加入计步功能, 但 AVD 本身不支持计步功能没有相关的传感器, 后放弃

2019. 5. 17-5. 18

实现多个控件的 UI 设计

2019. 5. 20

实现了体重调整之后即时更新的功能。饮食记录块也实现即时更新, UI 设计完善。学习 fragmentmanager 相关知识

2019. 5. 21 -5. 23

数据库连接以及 UI 设计完善, 细节优化

2019. 5. 23-5. 26

细节修改, 写文档

7 个人小结、不足及建议

此 app 界面设计美观、简洁, 信息显示直观, 功能也包括: 体重管理、饮食管理、运动辅助以及注册登录, 较为全面。在实现功能的方面, 也应用了较多方面的不同技术、fragment、listview、AsyncTask、Service、数据库与本地的数据库存储技术、网络下载技术、自定义控件交互工具和信息显示工具等等, 最终达到了如其名所指——对用户健康状况的多方面管理。

当然, 其不足也有许多, 首先 UI 界面设计因为本人没有经验而做的不够规范。并且由于时间和精力缘故, 可查询非当日的饮食记录、生成当月、当周的体重记录图表或者曲线、软件具体内容的丰富、手机发送验证码功能未实现, 同时由于虚拟机问题未能实现计步功能。这些都是日后可以改进的地方。

而未达到最好的效果, 主要有三个原因。本学期一开始金老师上课就强调了几点学习和开发的注意事项, 当时因为对于安卓开发完全不熟悉, 所以有些

不以为意，也不理解老师的用意。只有等到自己亲自动手去做这个项目，不断碰壁之后，我才有所体会。

首先，课上老师讲了很多原理性的东西，比如环境配置 gradle 里面的参数之类都是什么含义，活动、碎片等等的活动周期，这些都是比较原理性的，如果没有清楚掌握，看起来对于做一个软件没什么影响，但是很多时候问题就出在这些小地方，尤其安卓开发之中，我经常因为环境配置问题被卡主，因而这学期的学习让我更在意这些原理性的东西。

第二是一开始我没有经验，就经常遇到几个功能之间出现问题的情况，没有注意逐个去实现功能再整合。后来我就每次先用一个项目完成一个功能，测试好后再加入主项目之中，开发效率才逐渐提高。

第三是我一开始没有慢下来细致地学习每一个控件，先逐个摸清其用法和功能再去做项目。当时因为学的很基础，又有《第一行代码》这样很方便的参考书，让我想要“加快进度”，直接在做项目时，想实现什么功能直接去查该功能如何实现。但是实际上没有扎实的基础知识，让我后续的开发经常碰壁，很多细小的问题，都有其严格的规则，一点不对就会出现 bug 并且很难找出，只有有基础扎实的支撑，并且脑子里对于现在正在对哪一层面的哪一功能、哪一控件进行操作很清楚，才能顺利写出正确的代码和顺利地找出错误。而静下心来学习和掌握基本的原理虽然看起来费时间，但是实际上它能使得后续的开发少走弯路。我就是一开始没懂这一点，才导致开发时间被大大加长，反倒是后期经过学习之后，开发速度才越来越快。

在以后的开发过程之中，我会吸取教训，努力改进。

这一学期非常感谢金老师分享给我们很多经验，这些经验只有我自己亲自开发时，才明白其重要性。并且安卓开发课程继续了我上学期选修的 java 课程，两个学期学下来，我也对面向对象的编程有了更深刻的理解。

再一次感谢金老师的悉心讲授！