

## 1. 项目阐述

项目名称：嗨起来。

项目简介：整个项目总共实现了 4 个版块，功能分别为拼图，小转盘，图灵机器人聊天，新浪新闻，用于打发无聊时间，以自己和身边的同学为背景开发了一款暴漫系列的 app。

GitHub 地址：<https://github.com/yangxuanxc/JavaHomeWrok>

## 2. 程序的运行环境、安装步骤

(1) 运行环境：JDK 1.8

SDK：编译版本          API 23

最小运行版本      API 19

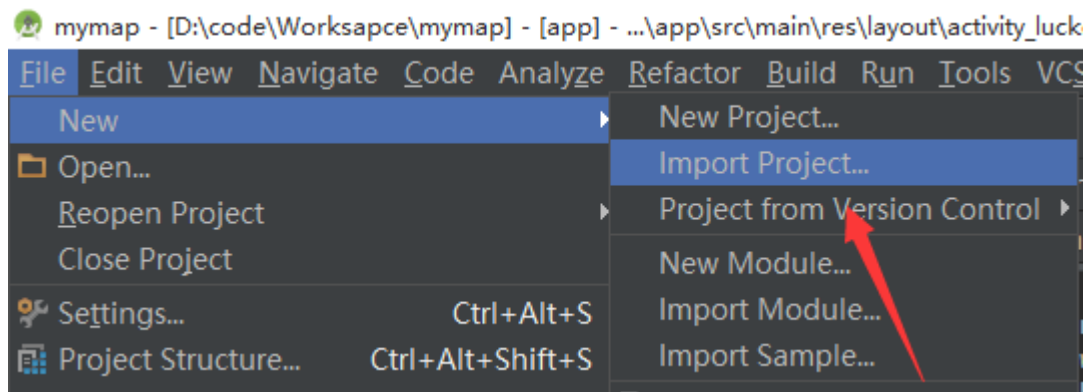
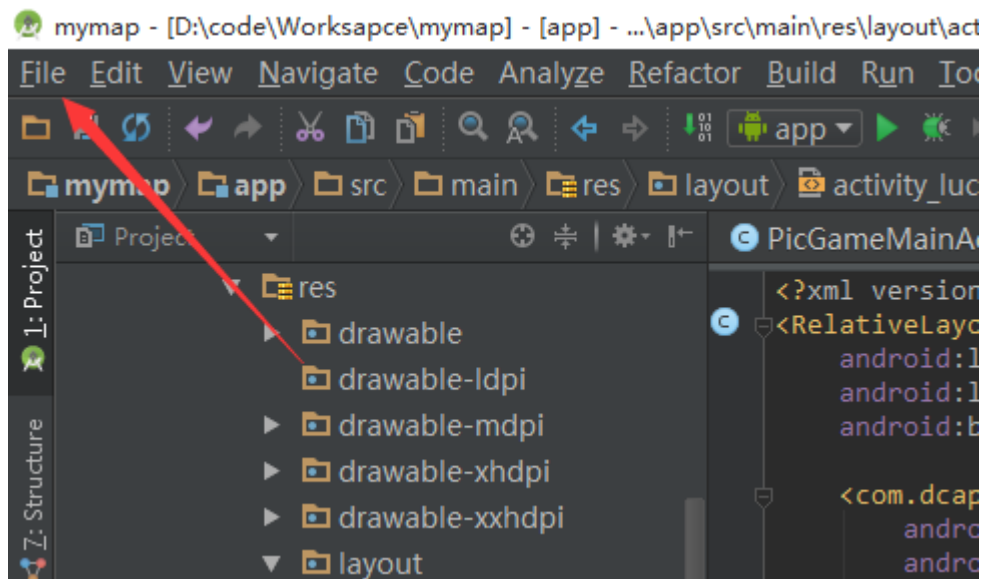
(2) 程序的组成部份：

1. 启动页面和主 Activity 在一个包下，其余 4 个功能分别在四个对应的包下。
2. 其中用到了图片启动时的滑动效果，使用了开源源码 Viewflow，库  
GitHub 库地址：<https://github.com/pakerfeldt/android-viewflow>
3. 用到了 gson-2.2.3.jar，实现对 Json 格式的信息进行转换，附件见程序。

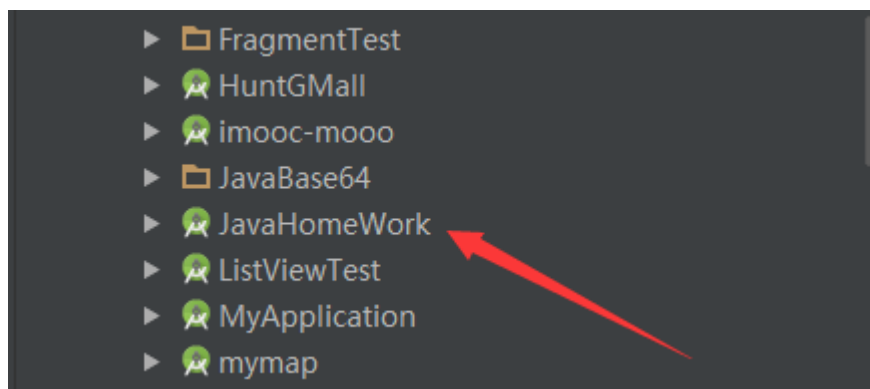
(3) 安装步骤：

- 1) 安装 JRE 1.8。
- 2) 将程序 jar 文件复制到计算机上
- 3) 安装 Android studio 1.5
- 4) 打开 Android studio 1.5.1 如下图所示：

点击 File:



点击：



这样稍等片刻就能导入到本机中。

### 3. 程序开发平台

(1) 代码行数: 5000

(2) 开发环境: windows10 专业版

Android studio 1.5.1

Eclipse 4.51

Genymotion for personal use

(3) 程序功能说明:

#### 3.3.1 启动页面:

启动页有两种显示:

① 第一次安装 APP 后启动: 显示程序启动页和其后的四张功能介绍页

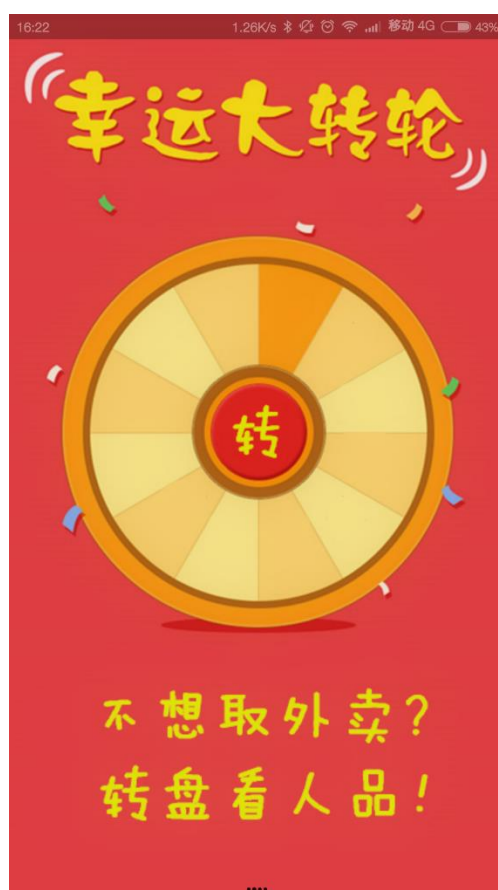
② 非第一次启动: 只显示程序启动页, 不显示功能介绍页

如下图: 图为启动页面, 每次启动 App 都会先显示 2 秒该页面



下面的 4 张图在第一次启动的时候，会起到介绍该 App 功能的作用，向左滑动即可看到下一张图片，这个效果是通过开源源码 **Viewflow** 实现的。

下面按照顺序给出 4 张启动图效果，使用小米 Note 截图：



上述 4 张图在第一次安装 App 的时候会依次滑动出现，分别介绍了几种功能，在滑动到最后一张图片的时候，点击 button [Start] 即可进入程序的主功能界面。

### 3.3.2 进入程序主功能界面即可显示下图：

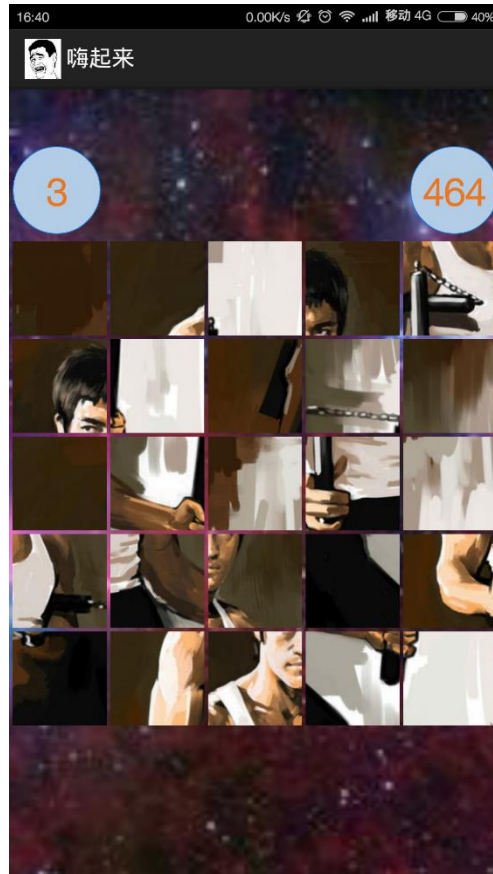
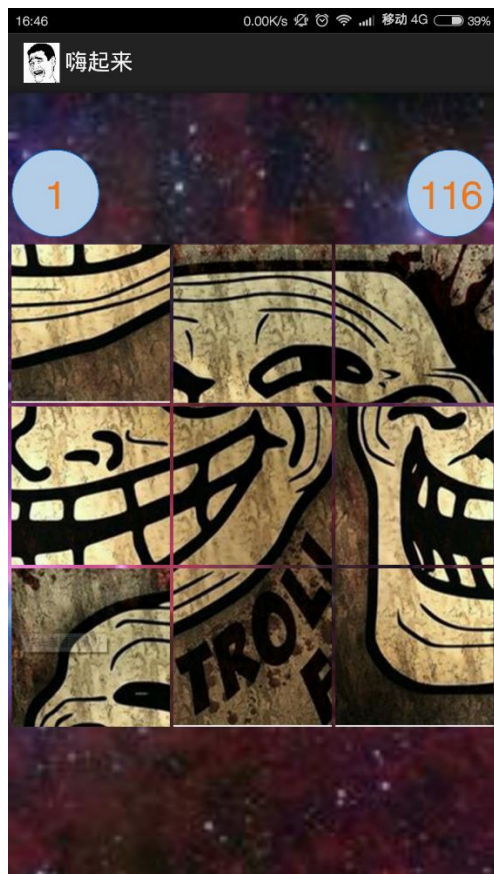


界面下方有 4 个 button，分别点击就可以进入 4 个不同的 Activity，该页面使用了 RadioGroup，里面包含了 4 个 RadioButton。主界面上除了点击 4 个 button 之外就没有其余功能，上面的图画和文字是通过一张背景图实现的。

### 3.3.3 拼图

首先看拼图的版块，如下图所示，其中会随机出现不同的图片，一共设置了 4 张图片，在图片左上角会显示关卡，当前是第几关，右上角显示的是当前关卡剩余的时间，如果在规定时间内不能完成拼图，那么就会提示游戏失败，关卡一共设置了无限的关卡，图片会越切越碎。下面给出 4 种状态下的图



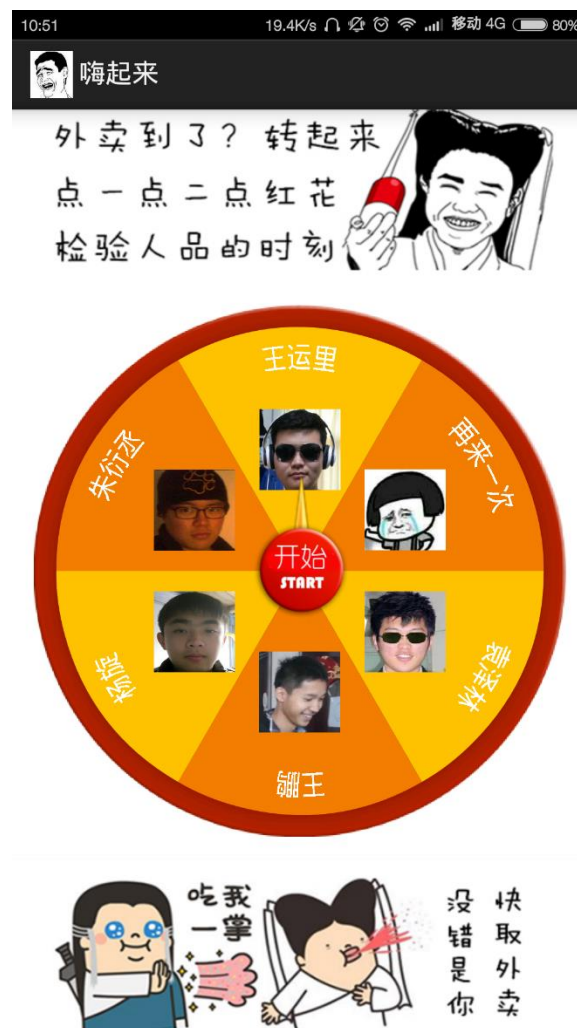




### 3.3.4 取外卖（转盘功能）

这个功能是受到大家点外卖后又懒得出去拿这一现象的启发设计的，我们写了一个转盘，把寝室成员的名字和照片都放了上去。因为我们寝室在 13 号楼的 6 层，所以下楼去拿还是比较辛苦的，以前大家要么谁先接到电话谁下去取，要么丢微信骰子比谁大谁小下去取，现在通过随机数专门写了这个功能，可以公平地决定谁下楼拿外卖。程序有一个接口，传入的是一个 `int`，6 个数据，那么传入 0-5，分别对应 6 个人，那么在点击开始的那一瞬间，结果已经随机出来了，所以可以在随机数上进行概率设定或者范围设定。

```
//传入几就是几，现在产生随机数  
mLuckyPanView.luckyStart((int)(Math.random()*5));
```



### 3.3.5 聊天

因为这学期刚学习了人工智能，并且网上有一个图灵机器人的网站，可以对 App 以及微信提供接口，所以就把第三个版块设置成为了聊天的窗口。其中，这个版块也是相当好玩的版块。机器人的名字设置的是小袁，全名袁泽林，因为袁泽林是我们的同学和室友，所以我们就开玩笑地把名字改成他了。这个机器人可以进行学习，还有很多特定的比如天气预报等功能，也可以自己设定一些特定的关键字，例如下图：



如下是服务器的设置，在服务器上动态更改，那么用户那里进行数据访问就会有更新。



修改头像

机器人昵称:

小袁

机器人性别:

☐ 男 ☒ 女

机器人年龄:

18

机器人住址:

北京市海淀区中关村南大街5号

日调用次数:

5000 次/日

权限升级:

扫描下方二维码关注图灵机器人订阅号，申请权限



开通了如下数据功能:



当然还有自己的 NLP 知识库，可以设定关键词进行特定的回答，在这里就不截图了，因为已经被玩坏了，室友开玩笑加了很多特定的问题。

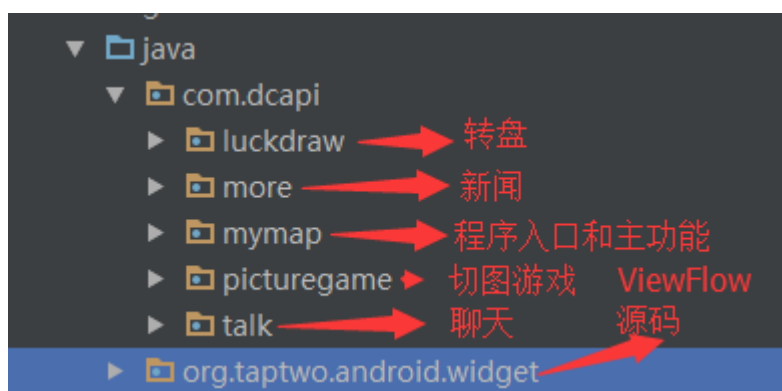
### 3.3.6 新闻

新闻版块使用了 WebView 控件，链接到了新浪新闻，相当于对网页的访问，这个功能最简单。



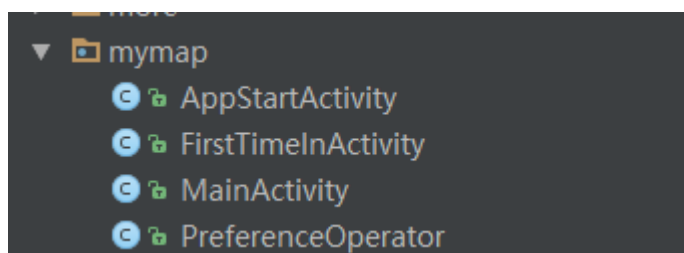
## 4. 程序算法说明及面向对象实现技术方案

由于一开始 Android studio 没有配置好，所以先在 Eclipses 上进行开发，最开始的版块同样是 4 块，准备开发继承了地图，指南针，天气预报和地铁线路图的程序，但是由于开发到地图的时候，百度开发者官网上的 SDK 出了问题，无法完成，后来便决定回退版本到刚刚写好了功能框架的版本，开发了现在的程序，其中先在 Eclipses 上建立了工程，建立了 5 个包，其中程序入口和主功能界面为“mymap”这个包，因为后面想改包名改不了，就没有修改包名。然后其余 4 个包分别对应 4 个版块。如图：



下面分别阐述每个包的结构：

首先看每个程序的入口及主功能包：



4.1 程序最先启动的时候调用的是 AppStartActivity，在该代码中逻辑很简单，直接显示布局文件，然后展示启动图，使用多线程的方式，展示了启动图，然后线程 sleep 2 秒后，创建一个新的线程再跳转到 MainActivity，然后结束线程。

在 Main 中进行判断，首先判断当前是不是第一次启动 App，其中

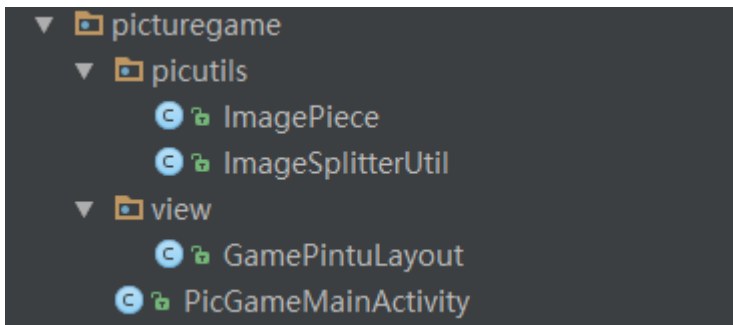
PreferenceOperator 存储了用户数据，那么如果用户是第一次启动 App，那么就执行下述代码：

```
if (sp.getString("first", "").equals("")) {  
    //测试阶段，后一个参数实际中为 yes，现在改为空进行测试  
    editor.putString("first", "yes");  
    editor.commit();  
    startActivity(new Intent(this, FirstTimeInActivity.class));  
}
```

然后 FirstTimeInActivity 中的代码执行依次出现 4 张介绍图，如果不是第一次启动，那么直接进入主功能页，主功能页面的代码也写在 MainActivity 中。

MainActivit 实现了 RadioGroup.OnCheckedChangeListener 接口，来监听点击 4 个 Button 的功能。

4.2 其次再来看 picturegame 包，这对应的是切图游戏，这个功能的实现相对较复杂：



其中 picutils 包中的两个文件是用来对图片进行切割和存储的；

图片用 Bitmap 来进行描述；

ImageSplitterUtil 工具类用来将一张完整的图切成指定方格小块；

ImagePiece 用来存储每张图的信息，用 Bitmap 描述，还有指定的序号 index

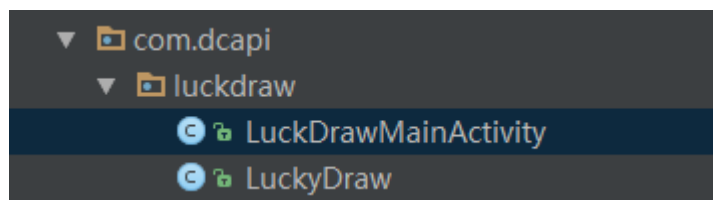
然后在 ImageSplitterUtil 中的方法传入一张图，然后返回一个 ArrayList 进行存储。

View 包中的 GamePintuLayout 对应的是对布局文件以及动画效果等的自定义容器，可以实现对面板的布局，宽度，游戏时间，游戏关卡，是否过关等逻辑等进行控制，其中较为复杂的逻辑都在这个布局文件中进行了编写。



最后 PicGameMainActivity 实现了方法的调用和对过关和游戏结束的时候的弹出框的处理。

#### 4.3 转盘的功能对应了 luckdraw 包



还是遵循了同样的架构模式。在 LuckDrwa 类中实现了较复杂的逻辑。

其中 LuckDrwa 继承了 surfaceView 类，并且实现了 Callback 和 runnable 接口。对转盘进行了绘制，以及启停逻辑的编写。在这当中，为了计算角度范围，进行了一些简单的数学运算。

#### 4.4 talk 包下对应的是聊天功能

其中每个 java 文件的功能就如同它们的命名所描述的一样。

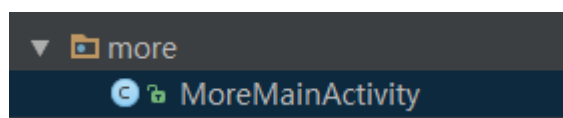
在进行布局 and 编写之前，首先进行通信测试。通过 Test 包下的 TestHttpUtil 工具类进行测试通信，当通信成功后进行编写后续代码。

其中 bean 包下的 ChatMessage 表示的是每一条聊天的信息。包含了发送人，发送内容，发送时间等内容。Result 用于表示从网络获取的数据。

其中 Utils 包下的 HttpUtils 用于进行数据的接受和传送，使用了外部库 gson，使用了 GET 的方式进行传送。

ChatMessageAdapter 的功能是适配器，将数据显示在布局文件。

#### 4.5 more 包下仅一个 Activity，使用了控件 WebView



所以综上所述：

总的项目架构比较清晰，如下所示：

