

中山大学移动信息工程学院本科生实验报告

（2017年秋季学期）

课程名称：移动应用开发 任课教师： 郑贵锋

2017 年 10 月 9 日

年级	15级	专业(方向)	移动信息工程专业
学号	15352044	姓名	陈姝睿
电话	18107346633	Email	chenshr8@mail2.sysu.edu.cn
开始日期	2017/9/24	结束日期	2017/10/1

目录

1	实验题目	2
2	实现内容	2
3	课堂实验结果	2
3.1	实验截图	2
3.2	实验步骤以及关键代码	3
3.3	实验遇到困难以及解决思路	8
4	课后实验结果	8
5	实验思考及感想	10

1 实验题目

熟悉Android Studio开发工具操作并且熟悉Android基本UI开发,并进行UI基本设计。

2 实现内容

要求实现一个应用界面，其启动后为第一个出现的界面。

3 课堂实验结果

3.1 实验截图

在模拟器中运行应用，出现的界面为以下界面，在编辑框中，输入学号和密码，并切换按钮，可以看到界面元素功能基本实现。



图 1: 实验结果效果图

在AS的design中，图形化界面和相关元素的边框约束关系如图：

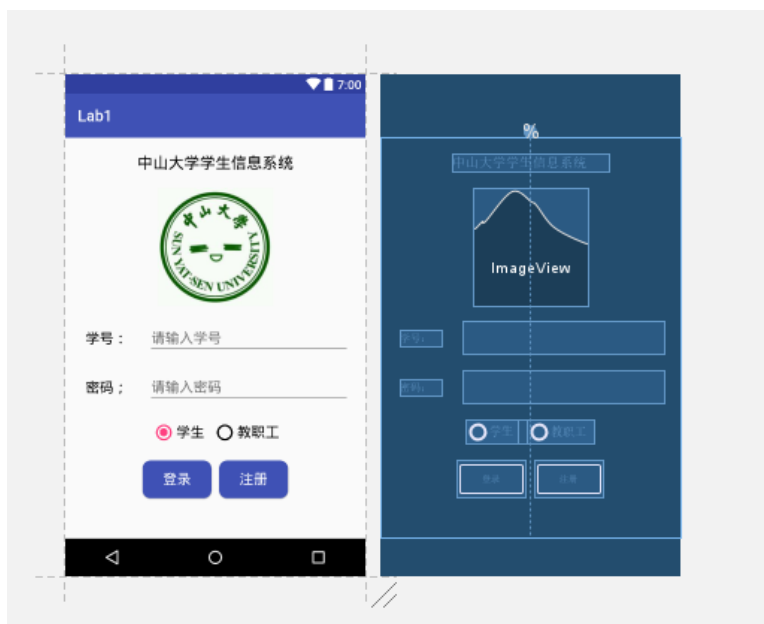


图 2: UI界面整体设计图

3.2 实验步骤以及关键代码

首先设置文本框，其文字大小通过textsize设置为20sp，然后将文本的左边与父边框左边建立约束，右边与父边框右边建立约束，顶部与父边框顶部建立约束且边距需设为20dp。文字颜色需要在资源包中的colors.xml文件中添加黑色，然后在文本中调用资源，将字体设为黑色。

```
<TextView
    android:id="@+id/textView2"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="中山大学学生信息系统"
    android:textColor="@color/colorBlack"
    android:textSize="20sp"
    app:layout_constraintLeft_toLeftOf="parent"
    app:layout_constraintRight_toRightOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    android:layout_marginTop="20dp" />
```

图 3: 中山大学信息系统文本框

校徽图片右边与父边框的右边建立约束，左边与父边框左边建立约束。然后其与它之上的中山大学信息系统需要建立约束，且距离已规定为20dp，所以图片的顶部与信息系统的底部建立约束，边距为20dp。

```
<ImageView
    android:id="@+id/imageView"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    app:srcCompat="@mipmap/sysu"
    android:layout_marginLeft="8dp"
    app:layout_constraintLeft_toLeftOf="parent"
    android:layout_marginRight="8dp"
    app:layout_constraintRight_toRightOf="parent"
    app:layout_constraintHorizontal_bias="0.5"
    android:layout_marginTop="20dp"
    app:layout_constraintTop_toBottomOf="@+id/textView2" />
```

图 4: 中山大学校徽

学号的文本大小需要设置为18sp，然后颜色为黑色，之后其顶部与输入编辑框顶部，底部与输入编辑框底部，建立约束，使文本框与它之后的输入编辑框上下平行对齐，其左边与父边框的左边建立约束，右边与输入编辑框左边建立约束。整个学号文本框的设置就完成了。

```
<TextView
    android:id="@+id/textView"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="学号: "
    android:textSize="18sp"
    android:textColor="@color/colorBlack"
    app:layout_constraintRight_toLeftOf="@+id/editText"
    app:layout_constraintBottom_toBottomOf="@+id/editText"
    app:layout_constraintTop_toTopOf="@+id/editText"
    app:layout_constraintLeft_toLeftOf="parent" />
```

图 5: 学号文本框

密码文本与上面的学号文本设置完全相同，其与密码输入框建立好上述的约束，以及与父边框的约束。

```
<TextView
    android:id="@+id/textView3"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="密码:"
    android:textSize="18sp"
    android:textColor="@color/colorBlack"
    app:layout_constraintRight_toLeftOf="@+id/editText2"
    app:layout_constraintBottom_toBottomOf="@+id/editText2"
    app:layout_constraintTop_toTopOf="@+id/editText2"
    app:layout_constraintLeft_toLeftOf="parent" />
```

图 6: 密码文本框

输入学号的编辑框与它上面的元素建立约束即其顶部与图片底部，边距为20dp，其右部与父边框右边建立约束，边距为20dp，提示文字大小为18sp，允许输入的类型需要设置为数字类型，在design中的inputtype界面选择number。输入框的允许输入长度可自由控制，但是为了美观，学号框与密码框需尽量相同，所以ems设置为12位，较为合适。

```
<EditText
    android:id="@+id/editText"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:ems="12"
    android:inputType="number"
    android:hint="请输入学号"
    android:textSize="18sp"
    android:layout_marginRight="20dp"
    app:layout_constraintRight_toRightOf="parent"
    android:layout_marginTop="20dp"
    app:layout_constraintTop_toBottomOf="@+id/imageView" />
```

图 7: 输入学号的编辑框

输入密码的编辑框约束方式的建立与输入学号的完全相同，输入选择textpassword，长度ems也设为12位，字体大小18sp，边距也与输入学号编辑框相同。

```

<EditText
    android:id="@+id/editText2"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:ems="12"
    android:hint="请输入密码"
    android:inputType="textPassword"
    android:textSize="18sp"
    android:layout_marginRight="20dp"
    app:layout_constraintRight_toRightOf="parent"
    android:layout_marginTop="20dp"
    app:layout_constraintTop_toBottomOf="@+id/editText" />

```

图 8: 输入密码的编辑框

我们需要radiogroup将两个按钮合并为一个元素，约束的建立，在radiogroup这个元素上设立。经过百度到的博客，首先将内部元素的orientation选项设为水平horizontal，元素的左右边分别与父边框左右边建立约束，居中，其顶部与其上的密码编辑框建立约束边距也为20dp。然后内部的学生和教职工两个按钮之间的距离，可以之间通过margin来设置，在学生里设置右边距或教职工中设置左边距。

```

<RadioGroup
    android:id="@+id/radioGroup"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:orientation="horizontal"
    app:layout_constraintLeft_toLeftOf="parent"
    app:layout_constraintRight_toRightOf="parent"
    android:layout_marginTop="20dp"
    app:layout_constraintTop_toBottomOf="@+id/editText2"
>
    <RadioButton
        android:id="@+id/radioButton2"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="学生"
        android:layout_marginRight="10dp"
        android:textSize="18sp"
        android:checked="true"/>

    <RadioButton
        android:id="@+id/radioButton"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="教职工"
        android:textSize="18sp"/>
</RadioGroup>

```

图 9: 学生或教职工二选一选择按钮

最后是两个按钮，登录和注册按钮，我们首先设立一个居中的引导线。引导线的方向为垂直，且为居中。居中通过Guidepercent来设置为0.5，即在父边框水平方向的0.5份额的位置。

```
<android.support.constraint.Guideline
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:id="@+id/guideline"
    app:layout_constraintGuide_percent="0.5"
    android:orientation="vertical" />

</android.support.constraint.ConstraintLayout>
```

图 10: 引导线

登录和注册按钮，登录按钮的右边需要与引导线建立约束，注册按钮的左边与引导线建立约束，然后两者都与其上的二选一按钮组建立顶部到底部的约束，边距为20dp，此时两者的位置已经设置好。接下来是圆角形状以及背景色的设置。

```
<Button
    android:id="@+id/button"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:background="@drawable/myshape"
    android:text="登录"
    android:textColor="@android:color/background_light"
    android:textSize="18sp"
    app:layout_constraintRight_toLeftOf="@+id/guideline"
    android:layout_marginRight="5dp"
    android:layout_marginTop="20dp"
    app:layout_constraintTop_toBottomOf="@+id/radioGroup" />
```

图 11: 登录按钮

```
<Button
    android:id="@+id/button2"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:background="@drawable/myshape"
    android:text="注册"
    android:textColor="@android:color/background_light"
    android:textSize="18sp"
    android:layout_marginLeft="5dp"
    app:layout_constraintLeft_toLeftOf="@+id/guideline"
    android:layout_marginTop="20dp"
    app:layout_constraintTop_toBottomOf="@+id/radioGroup" />
```

图 12: 注册按钮

由上我们可以看到，我们引入了一个新的资源包，myshape。在这里面设置了corner边角的圆弧弧度，还有背景色的设置，其中背景色设置与之前的字体颜色设置调用相同。

```
shape
<?xml version="1.0" encoding="utf-8"?>
<shape xmlns:android="http://schemas.android.com/apk/res/android">
    <corners
        android:radius="10dp"/>
    <solid android:color="@color/colorBlue" />
</shape>
```

图 13: 资源包myshape

至此，整个实验就完成了。

3.3 实验遇到困难以及解决思路

一：在设计登录和注册两个按钮时，想要使它们按照屏幕中央对齐且上下对齐，上下对齐可利用top-to-top，bottom-to-bottom的约束使它们对齐，然后如何实现整体居中呢？

在TA给出提示前，我的思路是左边按钮左边与边框的约束，右边按钮右边与边框的约束，按钮之间的约束来实现，这个过程需要慢慢手动调整边距，使两个按钮的距离适中，不会隔太远。之后使用引导线，两个按钮分别与位于屏幕中央的垂直引导线建立约束条件，按钮的设计就更加美观且方便。引导线的用法也百度了相关的经验，在之后的UI设计中，引导线应该能够方便我们的设计。

4 课后实验结果

上课看见老师对直接调用UI模板进行操作，于是在想可不可以像C语言中自己写的库一样，将某些经常需要用的UI元素封装起来，自己在某些活动中引用这个资源库，经过百度以及实践，发现是可行的。

具体步骤如下：

- (1) 在layout资源夹下新建一个界面资源文件，命名为reall。

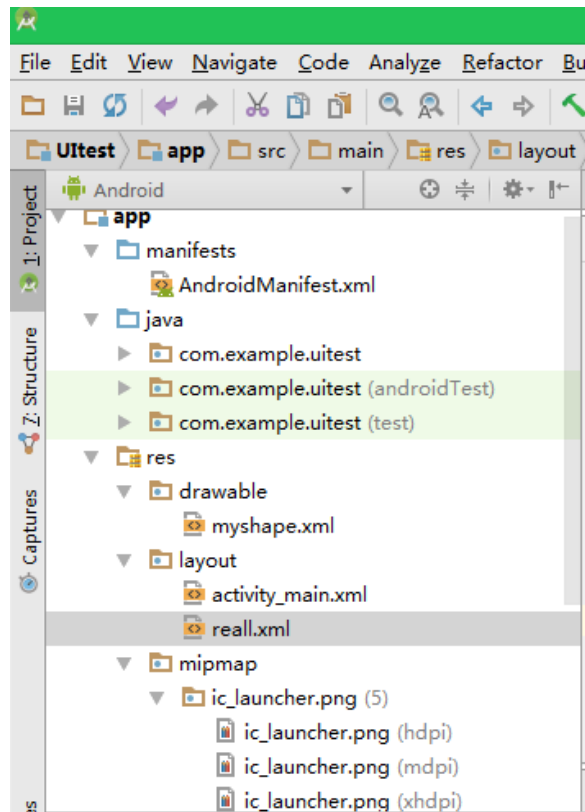


图 14: 新建资源文件

- (2) 在reall文件中，写好我们的界面，在本次实验中，利用之前完成的实验任务，直接复制代码。
- (3) 在activity-main中引用这个已编写好的资源。

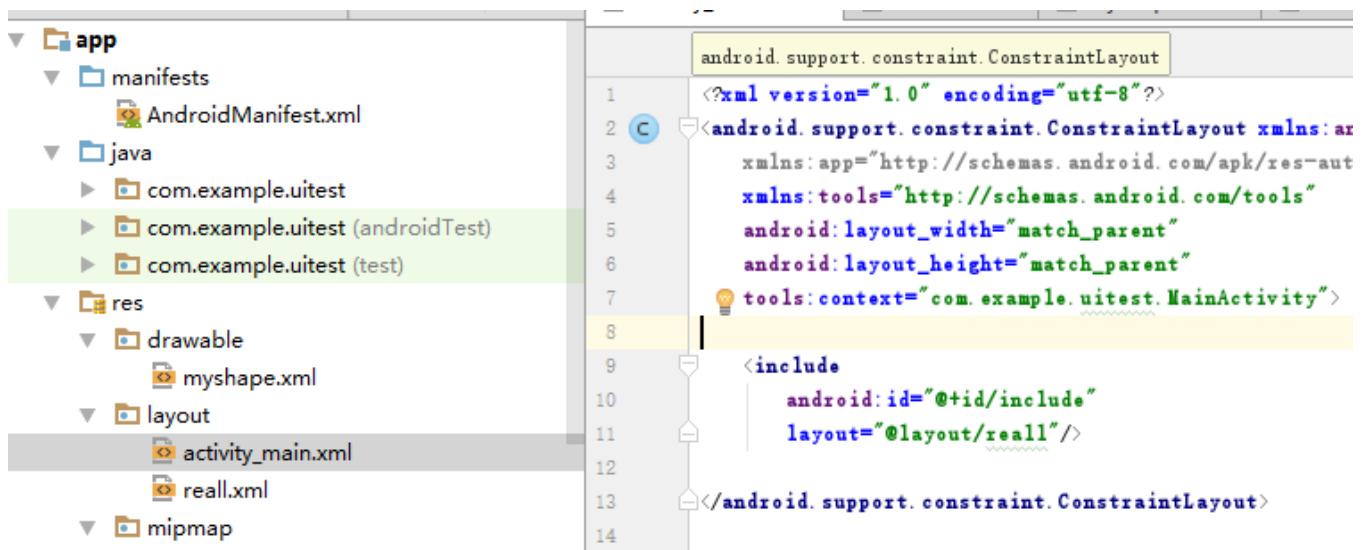


图 15: 调用资源文件

可以看到，运行后效果与直接写的界面完全一样，此时调用不需要在activity-main里面写布局，因为元素单一。真正使用时，若有多种元素和布局，则需要对这个文件的元素进行布局。

5 实验思考及感想

Android Studio的UI界面设计可以通过控件的拖拽，相连等方式进行设计，即图形化界面设计，这让枯燥的纯手写代码的工作变得轻松愉悦。我们可以在design界面，把控件先设计好，约束和位置大致摆放整齐，然后再在text界面，用代码方式进行细致的调整。

另外，安卓开发是一个时间较长也较热门的方向，我们在实验过程中遇到的问题，可以在各种各样的论坛中找到答案。有一些自己觉得比较奇怪的bug，通常都是粗心导致的，所以在实验过程中尽量做到细致，这样能够少花冤枉时间。