

# 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

年级	15 级	专业 ( 方向 )	移动互联网
学号	15352071	姓名	戴斯铭
电话	13727023544	Email	908660116@qq.com
开始日期	2017.9.26	完成日期	2017.9.28

### 一、 实验题目

基本 UI 界面设计

### 二、 实现内容

实现一个 Android 应用，界面呈现如下效果：

Lab1

中山大学学生信息系统



学号：

密码：

☒ 学生 ☐ 教职工

登录

注册

要求：

( 1 ) 该界面为应用启动后看到的第一个界面。

( 2 ) 各控件的要求如下：

要求只用一个 ConstraintLayout 实现整个布局；

标题字体大小 20sp，与顶部距离 20dp，居中；

图片与标题的间距为 20dp，居中；输入框整体距屏幕右边间距 20dp，上下两栏间距 20dp，内容（包括提示内容）如图所示，内容字体大小 18sp；

学号对应的 EditText 只能输入数字，密码对应的 EditText 输入方式为密码；

两个单选按钮整体居中，字体大小 18sp，间距 10dp，默认选中的按钮为第一个；

两个按钮整体居中，与上方控件间距 20dp，按钮间的间距 10dp，文字大小 18sp。按钮背景框左右边框与文字间距 10dp，上下边框与文字间距 5dp，圆角半径 10dp，背景色为 #3F51B5。

（3）使用的布局和控件：ConstraintLayout, TextView, EditText, Button, ImageView, RadioGroup, RadioButton。

### 三、课堂实验结果

#### （1）实验截图



#### （2）实验步骤以及关键代码

1. 在 layout 文件夹中创建新的 xml 文件，并且将 Root element 修改为 "android.support.constraint.ConstraintLayout"，即可创建出以 ConstraintLayout 为主体的布局。其对应代码如下：

```
<android.support.constraint.ConstraintLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent">

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

2. 下面开始在 xml 文件的 design 页面中添加需要的控件，首先添加的是 TextView。根据要求，“标题字体大小 20sp，与顶部距离 20dp，居中”，很明显可以直接生成标题与顶部，左侧，右侧的 constraint，再在代码中进行具体要求的修改，其关键代码如下：

```
<TextView
    android:id="@+id/textView"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="中山大学学生信息系统"
    android:textSize="20sp"
    android:layout_marginTop="20dp"
    android:textColor="#ff000000"
    app:layout_constraintRight_toRightOf="parent"
    app:layout_constraintLeft_toLeftOf="parent"
    app:layout_constraintTop_toTopOf="parent" />
```

要实现标题字体大小 20sp，直接设置 textSize=" 20sp" 即可；要实现居中，只需要添加其与左侧，右侧的 constraint，如上代码的倒数 2 至 3 行；而要求与顶部距离为 20dp，在添加了该标题与顶部的 constraint 之后，再设置 marginTop=" 20dp" 即可。

3. 在 drawable 文件夹中添加需要的“中大校徽”图片，然后在 design 页面汇总拖入 ImageView，选择 src 为该图片。由于图片本身比较大，因此随后在 Preview 框中调整其大小。要求为“图片与标题的间距为 20dp，居中”，因此只要设置其与左侧，右侧，以及与标题的 constraint 即可。关键代码如下：

```
<ImageView
    android:id="@+id/imageView"
    android:layout_width="110dp"
    android:layout_height="103dp"
    android:src="@drawable/sysu"
    app:layout_constraintLeft_toLeftOf="parent"
    app:layout_constraintRight_toRightOf="parent"
    android:layout_marginTop="20dp"
    app:layout_constraintTop_toBottomOf="@+id/textView" />
```

再将其 marginTop 设置为" 20dp" 即可。

4. 实现两个学号与密码输入框。要求是输入框整体距屏幕右边间距 20dp，上下两栏间距 20dp，内容（包括提示内容）如图所示，内容字体大小 18sp；学号对应的 EditText 只能输入数字，密码对应的 EditText 输入方式为密码。

先实现“学号”文本。由于仅仅对其字体大小有要求，因此需要自己添加 constraint，为了使其能稳定，因此我添加了顶层与左侧对它的 constraint，其对应代码如下：

```
<TextView
    android:id="@+id/textView2"
    android:layout_width="0dp"
    android:layout_height="0dp"
    android:text="学号:"
    android:textSize="18sp"
    android:textColor="#ff000000"
    app:layout_constraintLeft_toLeftOf="parent"
    android:layout_marginLeft="20dp"
    android:layout_marginTop="25dp"
    app:layout_constraintTop_toBottomOf="@+id/imageView" />
```

接下来是“密码”文本。其除了有字体大小要求为 18sp，还需要距“学号”文本 20dp，因此需要添加“学号”与左侧对它的 constraint。其对应代码如下：

```
<TextView
    android:id="@+id/textView3"
    android:layout_width="0dp"
    android:layout_height="0dp"
    android:text="密码:"
    android:textSize="18sp"
    android:textColor="#ff000000"
    app:layout_constraintLeft_toLeftOf="parent"
    android:layout_marginLeft="20dp"
    android:layout_marginTop="20dp"
    app:layout_constraintTop_toBottomOf="@+id/textView2" />
```

接下来是提示“请输入学号”的输入框 EditText。对这个输入框的要求主要是其内容为 18sp，提示内容只能输入数字，同时需要放置在合适的位置，右侧距屏幕右边间距 20dp。首先，设置 textSize=" 18sp" ;同时要求只能输入数字，inputType=" number" 。同时添加上方图片，左侧“学号”和屏幕右侧对它的限制。

```
<EditText
    android:layout_width="0dp"
    android:layout_height="0dp"
    android:textSize="18sp"
    android:inputType="number"
    android:hint="请输入学号"
    app:layout_constraintRight_toRightOf="parent"
    app:layout_constraintLeft_toRightOf="@+id/textView2"
    app:layout_constraintTop_toBottomOf="@+id/imageView"
    android:layout_marginLeft="30dp"
    android:layout_marginTop="13dp"
    android:layout_marginRight="20dp"
    app:layout_constraintHorizontal_bias="0.0"
    android:id="@+id/editText" />
```

可以看到，我所设置的 marginRight=" 20dp" ，marginLeft=" 30dp" ，这样既确定了其右侧距屏幕的距离，也确定了其长度。其次，我设置了其 marginTop=" 13dp" ，这个数值是结合左边的“学号”文本来进行调整的。

最后实现密码输入框 EditText。其要求内容只能为密码形式，同时距上一个输入框 20dp。其他要求则与学号输入框一致，具体位置根据其布局来调整。关键代码如下：

```
<EditText
    android:layout_width="0dp"
    android:layout_height="0dp"
    android:textSize="18sp"
    android:inputType="textPassword"
    android:hint="请输入密码"
    app:layout_constraintRight_toRightOf="parent"
    app:layout_constraintLeft_toRightOf="@+id/textView3"
    app:layout_constraintTop_toBottomOf="@+id/editText"
    android:layout_marginLeft="30dp"
    android:layout_marginRight="20dp"
    android:layout_marginTop="5dp"
    android:id="@+id/editText2" />
```

可以看到，inputType=" textPassword" ，其余与学号输入框基本相同。

5. 下面是实现两个单选按钮。由于要使两个按钮互斥，因此要使用 RadioGroup 控件。将 RadioGroup 控件拉入界面后，我是再在代码里添加了两个 RadioButton 控件，这样才能使他们两个联系在一起。由于对这两个按钮的要求是两个单选按钮整体居中，字体大小 18sp，间距 10dp，默认选中的按钮为第一个。因此实现的方式如下：

两个单选按钮整体居中，即使得整个 RadioGroup 居中，我的做法是在界面正中间设置一条竖直的 guideline，然后使 RadioGroup 的左右都被其约束，这样就可以实现居中了。相关代码如下：

```
<RadioGroup
    android:id="@+id/radioGroup"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginTop="20dp"
    android:orientation="horizontal"
    app:layout_constraintLeft_toLeftOf="@+id/guideline"
    app:layout_constraintRight_toLeftOf="@+id/guideline"
    app:layout_constraintTop_toBottomOf="@+id/editText2">
```

Guideline 的对应代码为：

```
<android.support.constraint.Guideline
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:id="@+id/guideline"
    app:layout_constraintGuide_begin="192dp"
    android:orientation="vertical"
    tools:layout_editor_absoluteY="0dp"
    tools:layout_editor_absoluteX="192dp" />
```

接下来的就比较简单，只要针对两个 RadioButton 进行修改就好了。由于两者间距 10dp，因此需要各自添加对方左右侧的 constraint，之后便改动一下对应文字的大小即可。对应代码如下：

```
<RadioButton
    android:id="@+id/radioButton4"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginRight="5dp"
    android:text="学生"
    android:checked="true"
    android:textSize="18sp"
    app:layout_constraintRight_toLeftOf="@+id/radioButton5" />
<RadioButton
    android:id="@+id/radioButton5"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginLeft="5dp"
    android:text="教职工"
    android:textSize="18sp"
    app:layout_constraintLeft_toRightOf="@+id/radioButton4" />
```

从上图可以看到，每个 RadioButton 和 ButtonGroup 都需要设置 id，这样他们才能正常工作。同时，由于默认选中按钮为第一个，因此其对应的 checked 属性设为 true。

6. 接下来是设置登录和注册按钮。要求是与上方控件间距 20dp，按钮间的间距 10dp，文字大小 18sp。直接在 design 界面将两个 Button 控件拖入界面后，分别添加其与上方的 constraint。设置 `marginTop="20dp"`，

由于要居中，因此我同样使用 guideline 来设置。接着，两个按钮都与 guideline 有 constraint，设置距离为 5dp，这样两个按钮既能居中，也能保持间距为 10dp。基本控件的位置已经确定，现在需要修改按钮的样式。

按照实验课课件，首先在 drawble 文件夹下新建一个 Drawblresourcefile，填写 filename 为 button，然后把自动生成的 selector 标签改为 shape。根据要求，按钮背景框左右边框与文字间距 10dp，上下边框与文字间距 5dp，圆角半径 10dp，背景色为 #3F51B5。在 <http://blog.csdn.net/sysukehan/article/details/52022307> 的这个教程中，我知道了如何按照要求对按钮进行设置，其对应代码如下：

```
<shape xmlns:android="http://schemas.android.com/apk/res/android">
    <corners
        android:radius="10dp"/>
    <solid
        android:color="#3F51B5"/>
    <padding
        android:top="5dp"
        android:bottom="5dp"
        android:left="10dp"
        android:right="10dp"/>
</shape>
```

而整体的两个 button 的代码则分别为：

```
<Button
    android:id="@+id/button1"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:background="@drawable/button"
    android:text="登录"
    android:textColor="#FFFFFF"
    android:textSize="18sp"
    app:layout_constraintRight_toLeftOf="@+id/guideline"
    android:layout_marginRight="5dp"
    android:layout_marginTop="20dp"
    app:layout_constraintTop_toBottomOf="@+id/radioGroup" />

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

需要注意的是，两个按钮的 background 均为我之前所添加的 button.xml 文件。



### (3) 实验遇到困难以及解决思路

1. 一开始对 AndroidStudio 的使用连入门程度都不到，连打开个代码界面都找半天。并且之后，ConstrainLayout 的代码我都是直接手打的，而不是在那个 design 界面上直接进行拖拉，连 constraint 也是手打。后来在实验课上 TA 的简单介绍后，才知道比较简单的方法，在之后未完成的部分便使用 design 来进行拖拉了。

2. 对实现两个控件居中功能，一开始不知道怎么实现。后面听别人说可以使用 guideline 来居中，思考了一下，发现只要对 guideline 来添加 constraint 就可以设置居中了，原理和设置与界面边界的 constraint 一致。

## 四、 课后实验结果

课后思考了一下，我添加的主要功能比较简单，是通过对页面里的控件点击后实现页面跳转。由于一开始并不了解，因此这个过程是在百度中得知的。

首先，假设我已经在布局文件中绘画了跳转控件。随后，需要在关联的类中声明一个私有控件名称，接着在类中的 onCreate 方法内执行以下操作。(下面以我的其中一个控件为例)：

(1)给该私有控件赋值，即设置布局文件中的控件 id 进行关联;

```
button = (Button)findViewById(R.id.button1); //登录
```

(2)给该控件绑定点击事件，同时给该控件添加点击响应时间;

```
button.setOnClickListener(new View.OnClickListener() {  
    @Override  
    public void onClick(View v) {  
        Intent intent = new Intent();  
        intent.setClass(Lab1.this,secondActivity.class);  
        startActivity(intent);  
    }  
});
```

其中，Intent 是一种运行时绑定机制，它能在程序运行过程中连接两个不同的组件。而 startActivity 即为启动，意味着点击后启动该组件。

接下来，就展示一下我实现的页面跳转，很简陋，望见谅。

主页面启动后：



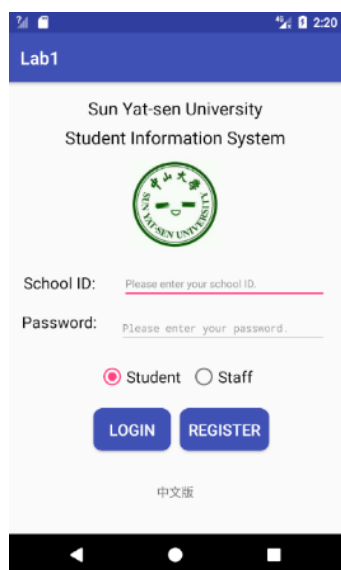
输入学号和密码后点击登录按钮:



又或者点击注册按钮:



同时，从主页面那个可以看到下方有“English version”的字样，点击其即可跳转到英文版界面：





而英文版界面下面又有“中文版”的字样，点击其即可返回中文版登录界面：



## 五、 实验思考及感想

这次实验要求我们使用 ConstrainLayout 来完成布局，而实际上这个布局方式十分简单，除了一开始不清楚怎么使用以外，其实熟练了应该还是能够较好地完成。而在做课后实验结果的时候，由于自己对原理也并不是很懂，因此也废了一点劲。希望今后能够更加高质量地完成创新功能。