

# 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	2015 级	专业 ( 方向 )	软件工程 ( 移动互联网 )
学号	15352358	姓名	谢裕强
电话	15766376518	Email	Xieyq26@mail.china.com
开始日期	2017.9.25	完成日期	2017.9.28

### 一、 实验题目

#### 基本 UI 界面设计

### 二、 实现内容

实现一个中山大学学生信息系统的 Android 应用的 UI 界面：

1. 包括标题、图片、输入框、按钮
2. 只用一个 ConstraintLayout 实现整个布局
3. 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；
4. 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；
5. 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；
6. 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
7. 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

### 三、 课堂实验结果

#### (1) 实验截图



## (2) 实验步骤以及关键代码

### a. 图片与标题的间距为 20dp，居中

```
<TextView
    android:id="@+id/text"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:textSize="20sp"
    android:text="@string/textview"
    app:layout_constraintLeft_toLeftOf="parent"
    app:layout_constraintRight_toRightOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    android:layout_marginTop="20dp"
/>
<ImageView
    android:id="@+id/sysu"
    android:layout_width="100dp"
    android:layout_height="100dp"
    android:src="@drawable/sysu"
    app:layout_constraintTop_toBottomOf="@+id/text"
    app:layout_constraintLeft_toLeftOf="parent"
    app:layout_constraintRight_toRightOf="parent"
    android:layout_marginTop="20dp"/>
```

设置居中的办法是将两个模块相对于父容器即屏幕的左右边界对齐即实现了居中的效果

### b. 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；

```
<EditText
    android:id="@+id/number"
    android:layout_width="0dp"
    android:layout_height="wrap_content"
    android:textColor="@color/colorPrimaryDark"
    android:inputType="number"
    android:textSize="18sp"
    app:layout_constraintLeft_toRightOf="@+id/studentid"
    app:layout_constraintTop_toBottomOf="@+id/sysu"
    app:layout_constraintRight_toRightOf="parent"
    android:layout_marginRight="20dp"
    android:layout_marginTop="20dp"
    android:hint="请输入学号"
    android:layout_marginEnd="20dp" />
<TextView
    android:id="@+id/studentid"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:textSize="20sp"
    android:text="学号："
    app:layout_constraintRight_toLeftOf="@+id/number"
    app:layout_constraintTop_toTopOf="@+id/number"
    app:layout_constraintBottom_toBottomOf="@+id/number"
    app:layout_constraintLeft_toLeftOf="parent"
    android:layout_marginLeft="20dp"
    android:layout_marginRight="20dp" />
```

以学号文本框及学号输入框为例，在这里将文本框和输入框互相建立约束条件将文本框和输入框作为一个整体去间距。

- c. 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；

```
<RadioGroup
    android:id="@+id/button0"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:orientation="horizontal"
    app:layout_constraintLeft_toLeftOf="parent"
    app:layout_constraintRight_toRightOf="parent"
    app:layout_constraintTop_toBottomOf="@+id/passnum"
    android:layout_marginTop="20dp">
    <RadioButton
        android:id="@+id/button1"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:textSize="18sp"
        android:text="学生"
        android:checked="true"
    />
    <RadioButton
        android:id="@+id/button2"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="教职工"
        android:textSize="18sp"
        android:layout_marginLeft="10dp"/>
</RadioGroup>
```

在这里设置两个单选按钮，必须给两个按钮设置 id，不然出现的效果就是只有两个按钮，但是出现不了显示信息。（pdf 里是说设置了 id，RadioButton 才能正常工作）

- d. 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

```
<Button
    android:id="@+id/btn2"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    app:layout_constraintTop_toBottomOf="@+id/button0"
    android:background="@drawable/button_style"
    android:layout_marginLeft="5dp"
    android:layout_marginStart="5dp"
    android:layout_marginTop="20dp"
    android:text="注册"
    android:textSize="18sp"
    android:textColor="@color/white"
    app:layout_constraintLeft_toLeftOf="@+id/guideline" />

<android.support.constraint.Guideline
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:id="@+id/guideline"
    android:orientation="vertical"
    app:layout_constraintGuide_percent="0.5"
    tools:layout_editor_absoluteY="0dp"
    tools:layout_editor_absoluteX="192dp" />
```

- e. 在这次实验中使两个按钮居中我使用的办法是设置一条位于中间的引导线，让两个 button 离这条引导线各 5dp。在这里设置了 button 统一的属性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 是在 drawable 文件目录下定义了一个 xml 文件，然后在这里引用。

```
android:background="@drawable/button_style"
```

里面的 xml 定义是

```
<?xml version="1.0" encoding="utf-8"?>
<selector xmlns:android="http://schemas.android.com/apk/res/android">
  <item>
    <shape>
      <solid android:color="#3F51B5"></solid>
      <corners android:radius="10dp"></corners>
      <padding android:bottom="5dp"
        android:top="5dp"
        android:left="10dp"
        android:right="10dp"></padding>
    </shape>
  </item>
</selector>
```

这个方式是当时问舍友，舍友告诉我查找这种方法，最后通过借阅他人的博客学会的，主要是当时查找圆角半径的设置。

(来自 <http://blog.csdn.net/lwcumt/article/details/4700206>)

### (3) 实验遇到困难以及解决思路

1.遇到困难：使用 wrap\_content 或者 match\_parent 无法将输入框的横线填满屏幕的空白部分。

解决办法：当时是想自己定义一个大小如 250dp，但是这样无法自动调整，如果换成别的大小的屏幕，输入框的横线就是固定那个大小。后来想起来老师在上课的时候有说过宽度设置为 0dp，则会填充全部空位。

2.遇到困难：文本框“学号”和输入框“请输入学号”那一部分的 UI 界面不能随着屏幕大小的改变自己调整大小。

解决办法：一开始是没有把文本框和输入框作为一个整体，后来将文本框和输入框互相设置约束条件，以整体去设置距离父容器即屏幕左右两边的距离，再不管怎么改变屏幕大小，这一部分都会自动适应屏幕。

3.遇到困难：不会将最下面登录和注册两个按钮居中后将中间的距离设置为 10dp。

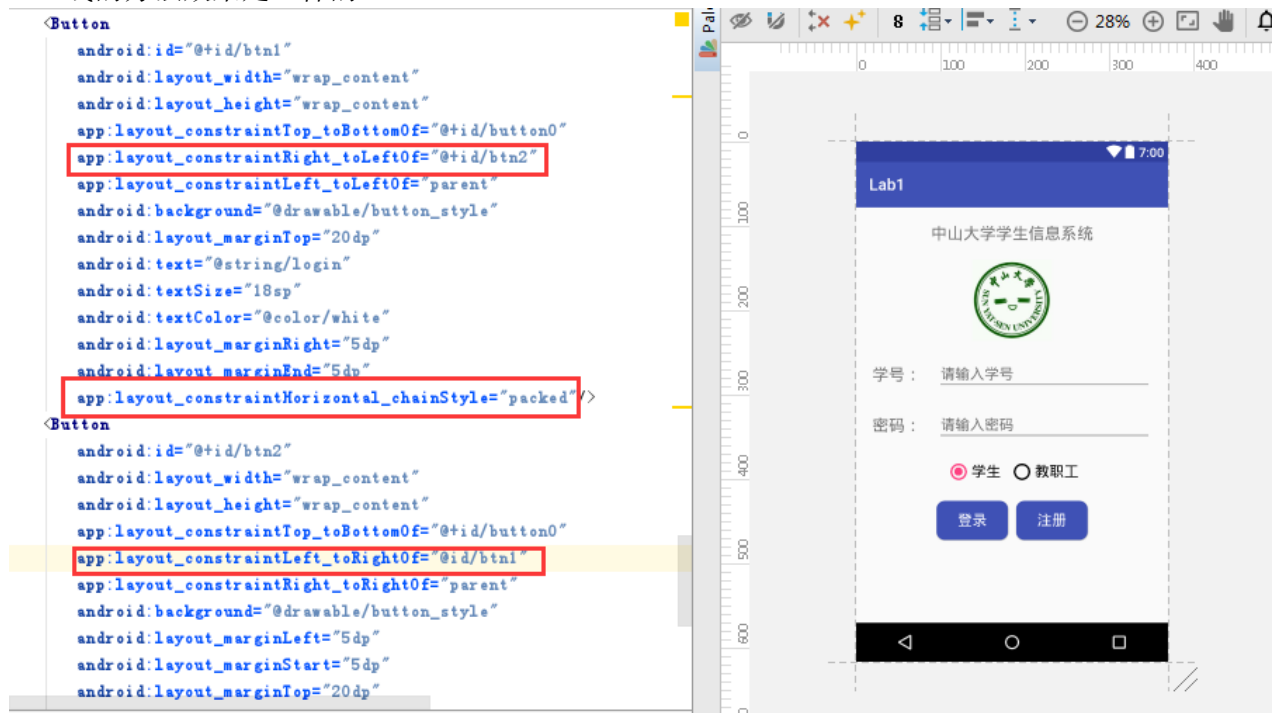
解决办法：一开始是将两个按钮设置约束条件，但是这样中间的距离是自动调整好的，不能达到预期要的效果。后来是使用实验文档中提到的引导线 guideline，将两个按钮设置为在引导线两边各一个且距离都为 5dp。后来在实验课上听 TA 讲，发现原来自己那种方式是缺少了设置链接的样式，后来再尝试之后发现确实和用引导线实现的效果是一样的。

4. 打开项目后 android.support.constraint.ConstraintLayout 前几行定义变成红色有报错，提示 URI is not registered。

解决办法：复制出现红色字体的路径，然后 File - Settings - schemas and Dtds，点击加号。把路径考进去就可以了。但修复了这个问题之后，却直接报错说无法找到约束布局，不过这个问题在重启多次 AS 后自动修复了，我猜这是 AS 本身软件没有加载完或者和当时的网络有关系。

## 四、 课后实验结果

1. 将两个按钮设置约束条件，并且将链接样式设置为 **packed**，效果如下，确实和使用引导线的方法效果是一样的：



2. 增加了点击登录按钮后的效果：

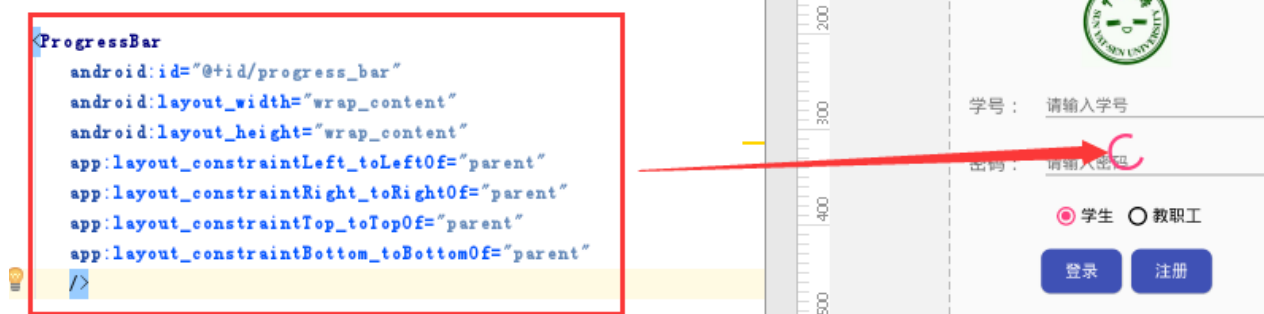


是在 MainActivity.java 使用 AlertDialog 在当前的界面点击了 button 按钮后注册的监听器，弹出一个对话框，这样可以提示一些重要的内容或者警告信息：

```
@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_main);
    Button btn1 = (Button)findViewById(R.id.btn1);
    btn1.setOnClickListener(this);
}

//注册一个监听器
@Override
public void onClick(View v) {
    switch(v.getId()) {
        case R.id.btn1://弹出对话框
            AlertDialog.Builder dialog = new AlertDialog.Builder(MainActivity.this);
            dialog.setTitle("Log in");
            dialog.setMessage("Log in Successfully!");
            dialog.setCancelable(false);
            dialog.setPositiveButton("OK",new DialogInterface.OnClickListener() {
                @Override
                public void onClick(DialogInterface dialog, int which){
                }
            });
            dialog.setPositiveButton("Cancel",new DialogInterface.OnClickListener() {
                @Override
                public void onClick(DialogInterface dialog, int which){
                }
            });
            dialog.show();
        }
    }
```

3. 在使用对话框之前也弄过进度条，是圆形进度条在旋转的形式：



但是设置后发现效果没有使用对话框更好，所以取消这种方式。

## 五、实验思考及感想

界面编程还是有很多种形式的，要用起来还是要多操作才能熟悉，不过常用控件的基础使用还是大同小异的，关键是各个细节部分。写完每次都要及时上传到 GitHub 保存好，养成这个习惯。更重要的是每次写项目或者实验遇到的问题解决后要及时记录下来。