

# 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	大三	专业（方向）	互联网
学号	15352010	姓名	蔡烨
电话	15989010669	Email	<u>caiy29@mail2.sysu.edu.cn</u>
开始日期	2017/10/3	完成日期	2017/10/9

### 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

### 三、实验结果

(1) 实验截图



(字体为手机默认字体)

## (2) 实验步骤以及关键代码

- ① 首先新建一个新的项目，根据要求写入一个个组件，实现彼此间的依赖关系
- ② 在 res/values/colors.xml 文件中定义需要用到的颜色

```

1  <?xml version="1.0" encoding="utf-8"?>
2  <resources>
3      <color name="colorPrimary">#3F51B5</color>
4      <color name="colorPrimaryDark">#303F9F</color>
5      <color name="colorAccent">#FF4081</color>
6      <color name="black">#000000</color>
7      <color name="white">#FFFFFF</color>
8  </resources>

```

在左边可以预览颜色效果

## ③ TextView: 中山大学学生信息系统

```

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

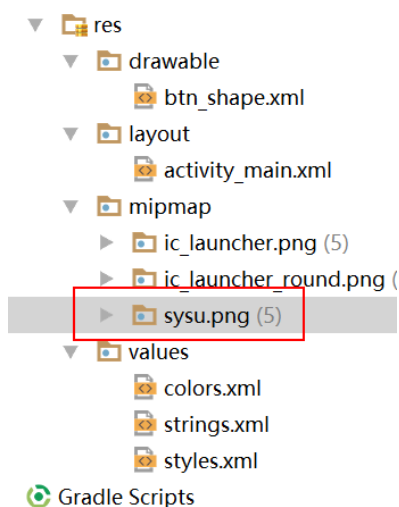
```

使用 layout\_constraint 语句来实现居中，左右两边都以父元素（即整个屏幕）为依赖，则可以居中；

引用上面定义的颜色时，用@color/定义颜色时的名称 即可；  
定义自身的 id 为 head；

#### ④ ImageView

在 res/mipmap 文件夹下添加图片，命名为 sysu



设置好图片的长度和宽度，引用时直接用 src= “@mipmap/定义图片时的命名” 即可，同样用 layout\_constraint 语句将其居中且置于 id 为 head 的组件的下方

```
<ImageView
    android:layout_width="89dp"
    android:layout_height="81dp"
    android:id="@+id/image"
    android:src="@mipmap/sysu"
    app:layout_constraintTop_toBottomOf="@+id/head"
    app:layout_constraintLeft_toLeftOf="parent"
    app:layout_constraintRight_toRightOf="parent"
    android:layout_marginTop="20dp" />
```

#### ⑤ TextView+EditView

将组件的 layout\_width 和 layout\_height 设置为 wrap\_content 时，即是刚好包围内容的大小，如图中的“学号”“密码”，大小刚好包围了字。

而下方设置为 0dp，是为了让组件填充所有的空间，前提是该组件的依赖关系必须定义好。比如此处，右边以父元素屏幕为依赖，距离屏幕右边 20dp，左边是以 id 为 id（即学号）的组件为依赖，左右两边都有依赖，因为设置了 layout\_width 为 0dp，则意味着将左右两边之间的空白自动填充。

下面密码一栏也同理。

```

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:id="@+id/id"
    android:text="学号: "
    android:textSize="18sp"
    app:layout_constraintTop_toBottomOf="@+id/image"
    android:textColor="@color/black"
    android:layout_marginTop="20dp"
    android:paddingLeft="20dp" />
<EditText
    android:layout_width="0dp"
    android:layout_height="wrap_content"
    android:id="@+id/input1"
    android:hint="请输入学号"
    android:inputType="number"
    android:textSize="18sp"
    android:layout_marginTop="20dp"
    android:layout_marginRight="20dp"
    app:layout_constraintTop_toBottomOf="@+id/image"
    app:layout_constraintRight_toRightOf="parent"
    app:layout_constraintLeft_toRightOf="@id/id"
    app:layout_constraintBottom_toBottomOf="@+id/id"/>

```

## ⑥ RadioGroup

两个单选按钮作为一个 radiogroup, orientation=“horizontal” 将两个按钮设置为水平放置, radiogroup 和每个 radio 都必须设置 id, 组件才能正常工作。其中第一个按钮 checked=“true” 默认被选中状态。

```

<RadioGroup
    android:id="@+id/occasion"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:orientation="horizontal"
    app:layout_constraintLeft_toRightOf="parent"
    app:layout_constraintRight_toLeftOf="parent"
    app:layout_constraintTop_toBottomOf="@id/password"
    android:layout_marginTop="20dp">
    <RadioButton
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:id="@+id/student"
        android:text="学生"
        app:layout_constraintRight_toLeftOf="@+id/teacher"
        android:paddingRight="10dp"
        android:checked="true"/>
    <RadioButton
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:id="@+id/teacher"
        android:text="教职工"
        app:layout_constraintLeft_toRightOf="@id/student"
        app:layout_constraintTop_toBottomOf="@id/password"/>
</RadioGroup>

```

## ⑦ Button

```

<Button
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:id="@+id/signin"
    android:text="登陆"
    android:textSize="18sp"
    android:layout_marginTop="20dp"
    android:layout_marginRight="10dp"
    app:layout_constraintTop_toBottomOf="@id/occasion"
    app:layout_constraintRight_toLeftOf="@+id/registre"
    app:layout_constraintLeft_toLeftOf="parent"
    android:textColor="@color/white"
    android:background="@drawable/btn_shape"/>

```

Button 组件的特殊在于其使用了自定义的 background，在 res/drawable/btn\_shape.xml 中定义了按钮的背景样式，包括圆角、颜色、内边距等，使用时直接引用即可。

```

<?xml version="1.0" encoding="utf-8"?>
<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>

```

- ⑧ 以上便写好了布局文件，需要创建 java 文件 MainActivity，然后在该文件中将布局文件引入。然后在注册文件 AndroidManifest.xml 中注册，将该 activity 设置为应用启动时第一个加在的 activity 然后就可以运行了。实际上，创建新的项目时，软件已经自动帮我们做好这些工作，只需要检查各个文件的名称对否对应上即可。

```

package com.example.caiye.lab1;

import ...

public class MainActivity extends AppCompatActivity {

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
    }
}

<activity android:name=".MainActivity">
    <intent-filter>
        <action android:name="android.intent.action.MAIN" />

        <category android:name="android.intent.category.LAUNCHER" />
    </intent-filter>
</activity>

```

#### 四、 实验思考及感想

一开始设置了 margin 的大小但是发现并没有作用，于是用了 padding 代替，后来才了解了 margin 外边距需要有相对作用物才会有作用，也就是说，外边距是指和哪个组件的外边距，需要建立好依赖。而 padding 作为内边距则没有这个问题。

其次是自定义按钮的背景框时，因为没有把 selector 改成 shape 而不起作用。其他则基本上按照实验的 pdf 文档上的指导可以完成。

作业要求：

1. 命名要求： 学号\_姓名\_实验编号，例如 15330000\_林 XX\_lab1。
2. 实验报告提交格式为 pdf。
3. 实验内容不允许抄袭，我们要进行代码相似度对比。如发现抄袭，按 0 分处理。