

# 移动应用开发实验报告

## lab1:基本UI界面设计

姓名	学号	班级	电话	邮箱
王若曦	15352319	1514	15354222552	1511330303@qq.com

### 实验目的

- 1. 熟悉Android Studio开发工具
- 2. 熟悉Android基本UI开发，并进行UI基本设计

### 实验内容

实现一个Android应用，界面呈现如下效果：



（具体要求在下面实验过程中一一描述）

### 实验过程

UI设计主要在layout中的activity\_lab1.xml下进行（实验要求只用一个ConstraintLayout实现整个布局）。

首先，UI的设计可以通过在text界面写代码或者通过在design界面拖动控件实现。为了将UI界面设计的更准确，我们同时利用两种方法进行设计实现。

## 1. 标题字体大小20sp，与顶部距离20dp，居中

**text:** 利用"textSize"属性设置字体大小；利用与顶部的相对位置"layout\_constraintTop\_toTopOf"设置与顶部距离；利用左右边界与父容器对

齐"layout\_constraintLeft\_toLeftOf="parent""layout\_constraintRight\_toRightOf="parent""设置居中。

```
<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"
    app:layout_constraintLeft_toLeftOf="parent"
    app:layout_constraintRight_toRightOf="parent"
    android:id="@+id/textView"
    app:layout_constraintTop_toTopOf="parent"
    android:layout_marginTop="20dp" />
```

**design:** 点击标题text框上部的圆点，连接到界面顶端，设置间距即可设计与顶部距离；同样将左边和右边的圆点连接到界面左右两端，即可设置居中；在design界面也可通过properties设置对应控件的属性，如这里的字体大小textSize（后面不再赘述）。

## 2. 图片与标题间距为20dp，居中

图片的导入有多种方法：

1. 找到project文件位置，直接把图片复制粘贴到drawable文件夹内
2. 右键drawable->New->Image Asset->选择image，找到想要导入的图片即可

**text:** 创建图片控件"<ImageView.../>"；与上面同样方法设置与上方控件的距离并居中，这里要注意与上方控件的相对位置设置是"layout\_constraintTop\_toBottomOf="@+id/textView",即该控件顶部对齐上方控件（上方控件id）底部，"；图片大小可以手动通过"layout\_width"和"layout\_height"设置。

```
<ImageView
    android:id="@+id/imageView3"
    android:layout_width="100dp"
    android:layout_height="0dp"
    app:srcCompat="@drawable/sysu"
    app:layout_constraintTop_toBottomOf="@+id/textView"
    android:layout_marginTop="20dp"
    app:layout_constraintRight_toRightOf="parent"
    app:layout_constraintLeft_toLeftOf="parent"
    app:layout_constraintDimensionRatio="1:1"
/>
```

**\*\*design:\*\***在design界面可以简单地通过拖动调节图片的大小，但是可能会导致图片的长宽比不是1，此时可以在text里写"layout\_constraintDimensionRatio="1:1""同时把图片的高或宽设为0dp来使得图片的长宽比保持为1:1。

### 3. 输入框整体距屏幕右边间距20dp，上下两栏间距20dp，内容字体大小18sp

设置输入框时可以先将默认值输入，即这里的“请输入学号”和“请输入密码”，这样可以通过设置输入框高度为"wrap\_content"来匹配18sp的内容。

**text:** 与上方控件的距离和与右边屏幕的间距设置与上面的方法相同，但是要注意的是设置两输入框间距时这里通过设置密码框与学号框的距离来实现，所以当拖动学号框是密码框会跟着移动，但是拖动密码框时学号框不会跟着动，当然可以通过设置chain来连接，在后面的button设置；这里通过inputType属性设置输入框输入数据的属性，学号框type为number，密码框type为textPassword，此属性使得输入数据后显示为 "."。

```
<EditText
    android:id="@+id/editText"
    android:layout_width="251dp"
    android:layout_height="wrap_content"
    android:layout_marginRight="20dp"
    android:layout_marginTop="20dp"
    android:hint="请输入学号"
    android:inputType="number"
    android:textSize="18sp"
    app:layout_constraintRight_toRightOf="parent"
    app:layout_constraintTop_toBottomOf="@+id/imageView3"
    android:layout_marginEnd="20dp" />
```

```
<EditText
    android:id="@+id/editText1"
    android:layout_width="251dp"
    android:layout_height="wrap_content"
    android:layout_marginRight="20dp"
    android:layout_marginTop="20dp"
    android:hint="请输入密码"
    android:inputType="textPassword"
    android:textSize="18sp"
    app:layout_constraintRight_toRightOf="parent"
    app:layout_constraintTop_toBottomOf="@+id/editText"
    android:layout_marginEnd="20dp" />
```

**design:** 这里输入框的宽度我们通过在design界面拖动输入框调节，可以在text代码中看到拖动调节的宽度为251dp。

这里还要设置两个TextView来显示“学号：”和“密码：”，设置方法和标题TextView类似，不再赘述，要注意的是可以通过与输入框的上下边界对齐来实现和输入框的内容水平对齐。

### 4. 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

一组单选按钮通过RadioGroup设置，一个RadioGroup控件可以设置多个单选按钮。

**text:** 居中方法同上；通过"orientation="horizontal""设置单选按钮的排布方式为水平并列排布；每个单选按钮控件要记得设置id；通过"checked="true""设置默认选择；这里同样通过学生按钮与教职工按钮

的相对位置设置距离10dp。

```
<RadioGroup
    android:id="@+id/radioGroup"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginTop="10dp"
    android:orientation="horizontal"
    app:layout_constraintLeft_toLeftOf="parent"
    app:layout_constraintRight_toRightOf="parent"
    app:layout_constraintTop_toBottomOf="@id/editText1">

    <RadioButton
        android:id="@+id/radioButton2"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginRight="10dp"
        android:checked="true"
        android:text="学生"
        android:textSize="18sp"
        app:layout_constraintRight_toLeftOf="@+id/radioButton" />

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

**design:** 这里可以通过学生按钮的properties中的checked属性设置它默认被选中。

## 5. 圆角半径10dp，背景色为#3F51B5

这里我们先配置按钮的背景：在drawable文件夹下新建一个Drawable resource file,填写file name,然后把自动生成的selector标签改为shape，利用corners和solid属性设置按钮的圆角半径和背景色。

- corners属性用于设置按钮四个角圆角的大小，默认0dp就是直角，大于0dp就会有圆角的效果。
- solid属性用于设置背景色，这里的color通过红绿蓝三原色设置，如这里的3F表示红的值，51表示绿的值，B5表示蓝的值，合起来表示一种由三原色混合生成的颜色。这里的值都是16进制表示的。
- shape的其余几个属性(<http://blog.csdn.net/sysukehan/article/details/52022307>)

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

在button中设置background为@drawable/加上刚刚填写的文件名即可引用这个自定义的背景。

**6. 两个按钮整体居中，与上方控件间距20dp，按钮间间距10dp，文字大小18sp，按钮背景左右边框与文字间距10dp，上下边框与文字间距5dp**

**text:** 居中、与上方控件距离、字体大小同上；

背景的设置上面已经介绍；

按钮间距这里我们通过链样式将两个button双向链接在一

起"layout\_constraintHorizontal\_chainStyle="packed""，此时我们两个button可以相互建立连接

即"layout\_constraintRight\_toLeftOf="@+id/button""和"layout\_constraintLeft\_toRightOf="@+id/button2""，此时拖动任何一个button，另一个button都会随着移动；

内间距通过padding设置，padding有上下左右四个属性，即题目要求的边框与文字间距。

```
<Button
    android:id="@+id/button2"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginLeft="8dp"
    android:layout_marginRight="10dp"
    android:layout_marginTop="20dp"
    android:background="@drawable/back"
    android:paddingBottom="5dp"
    android:paddingLeft="10dp"
    android:paddingRight="10dp"
    android:paddingTop="5dp"
    android:text="登录"
    android:textColor="@android:color/background_light"
    android:textSize="18sp"
    app:layout_constraintLeft_toLeftOf="parent"
    app:layout_constraintRight_toLeftOf="@+id/button"
    app:layout_constraintTop_toBottomOf="@+id/radioGroup"
    app:layout_constraintHorizontal_chainStyle="packed" />

<Button
    android:id="@+id/button"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginLeft="10dp"
    android:layout_marginRight="8dp"
    android:layout_marginTop="20dp"
    android:background="@drawable/back"
    android:paddingBottom="5dp"
    android:paddingLeft="10dp"
    android:paddingRight="10dp"
    android:paddingTop="5dp"
    android:text="注册"
    android:textColor="@android:color/background_light"
    android:textSize="18sp"
    app:layout_constraintLeft_toRightOf="@+id/button2"
    app:layout_constraintRight_toRightOf="parent"
    app:layout_constraintTop_toBottomOf="@+id/radioGroup" />
```

**design:** 这里可以通过在按钮的properties中选择文字的颜色。

# 实验心得

---

本次实验中遇到的问题：

1. **design**界面设计不稳定。一开始进行设计的时候，在**design**界面拖动控件进行界面设计，但是控件总是不稳定，比如把图片拖到预想的位置后，鼠标再点一下空白处图片就会再次移向鼠标的位置，所以后来基本都是通过在**text**界面写代码进行的设计。
2. **Android Studio**环境配置的问题。不知道是我一开始配置**AS**的时候不完善还是更新的原因，一些包好像用不了，总是得不到我想要的效果，然后TA帮忙改了**build gradle**里的1.0.2的**constraint-layout**就可以了。但是后来又有一次报错，不知道为什么。
3. 控件间的连接。如果两个控件只用一条连接的话，比如上面的两个输入框使用一条连接设置了间距，这样不是很稳定，因为当拖动连接者时被连接者是不会动的。还有就是链样式中在**design**界面有一个锁链一样的按钮，点一下两个**button**的间距就会变大，不知道是为什么，想着下次问问TA。
4. 居中的方法。控件居中的方法有多种，比如通过和父容器对齐来居中，设置引导线来居中，还可以直接在**design**界面右键居中。本次实验通过简单的UI界面设计，熟悉了**Android Studio**的使用，掌握了一些UI设计的方法和技巧，比如居中，链接之类的。对以后的安卓开发有很大帮助。感觉自己对这方面比较感兴趣，以后会多花一些时间来研究。