

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)	软件工程 (移动互联网)
学号	15352396	姓名	袁子豪
电话	13903039982	Email	568608846@qq.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 Android 应用，界面呈现如下效果：



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；

3/11 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；

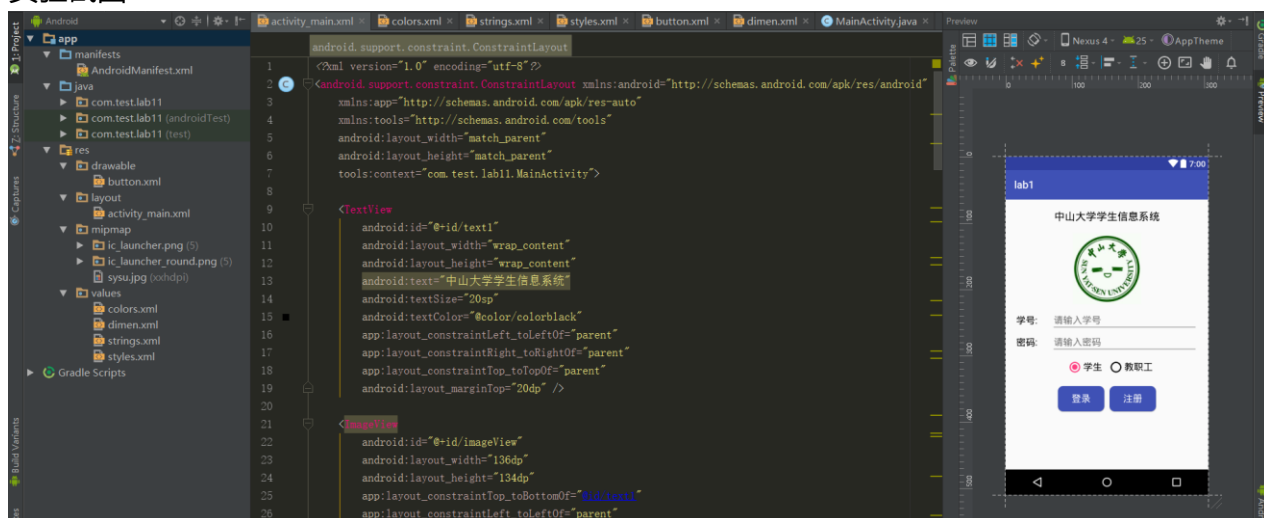
两个单选按钮整体居中，字体大小 18sp，间距 10dp，默认选中的按钮为第一个；

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

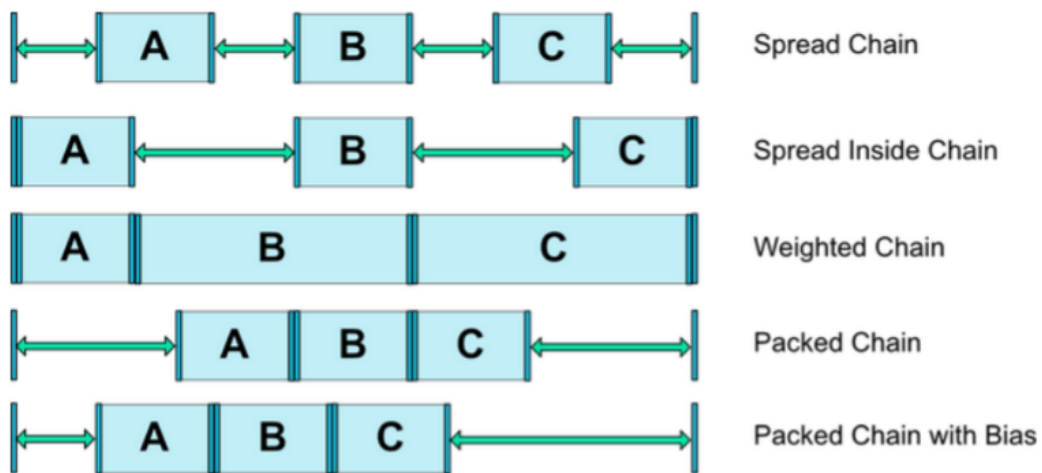
(1) 实验截图



(2) 实验步骤以及关键代码

1. 学习 ConstraintLayout 约束布局相关属性

```
app:layout_constraintBottom_toBottomOf="parent"
app:layout_constraintLeft_toLeftOf="parent"
app:layout_constraintRight_toRightOf="parent"
app:layout_constraintTop_toTopOf="parent" />
```



通过 `app:layout_constraintHorizontal/Vertical_chainStyle` 设置链接的样式。

2. 创建 TextView 控件显示标题信息

```
<TextView
    android:id="@+id/text1"
    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:textColor="@color/colorblack"
    app:layout_constraintLeft_toLeftOf="parent"
    app:layout_constraintRight_toRightOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    android:layout_marginTop="20dp" />
```

中山大学学生信息系统

利用步骤 1 中的相关属性来确定标题的位置，主要在于整体居中（两边与 parent 两连），以及在最上方的位置。然后就是分别用 `text`、`textSize`、`textColor` 来调整文本的内容、大小以及颜色等基本属性。

3. 创建 ImageView 控件显示指定图片

```
<ImageView
    android:id="@+id/imageView"
    android:layout_width="136dp"
    android:layout_height="134dp"
    app:layout_constraintTop_toBottomOf="@id/text1"
    app:layout_constraintLeft_toLeftOf="parent"
    app:layout_constraintRight_toRightOf="parent"
    android:layout_marginTop="20dp"
    android:src="@mipmap/sysu" />
```



根据约束布局确定该控件在于标题下方且有 20dp 的距离。然后利用 src 引用把已经添加到 mipmap 文件夹的 sysu 图片显示出来。

4. 创建学号（密码）输入框

```
<TextView
    android:id="@+id/xh"
    android:layout_height="wrap_content"
    android:layout_width="wrap_content"
    app:layout_constraintLeft_toLeftOf="parent"
    app:layout_constraintTop_toBottomOf="@+id/imageView"
    android:layout_marginLeft="20dp"
    android:layout_marginTop="20dp"
    android:textColor="@color/colorblack"
    android:textSize="18sp"
    android:text="学号:" />

<EditText
    android:id="@+id/edit1"
    android:layout_height="wrap_content"
    android:layout_width="0dp"
    android:textColor="@color/colorAccent"
    app:layout_constraintLeft_toRightOf="@+id/xh"
    app:layout_constraintRight_toRightOf="parent"
    app:layout_constraintTop_toTopOf="@+id/xh"
    app:layout_constraintBottom_toBottomOf="@+id/xh"
    android:layout_marginRight="20dp"
    android:textSize="18sp"
    android:hint="请输入学号"
    android:inputType="number" />
```

学号:

密码:

在布局上，因为要保持两个大小不同的控件的紧密相连，这里需要让后面的 EditText 的 top 和 bottom 都与相对应的前面的 TextView 的 top 和 bottom 相连接，使控件间保持水平上的高度中心对齐。然后把它们左右相连，则会产生紧密相连且整体居中的效果。而对于控件本身属性的设置，这里要考虑到学号只能输入数字，而密码要为小点*，利用其对 inputType 的设定可以完成对应的效果（数字-number，密码-numberpassword）

5. 创建单选按钮

```

<RadioGroup
    android:id="@+id/id0"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginTop="20dp"
    android:orientation="horizontal"
    app:layout_constraintTop_toBottomOf="@+id/mm"
    app:layout_constraintRight_toRightOf="parent"
    app:layout_constraintLeft_toLeftOf="parent">
    <RadioButton
        android:id="@+id/id1"
        style="@style/my_radiobutton_style"
        android:text="学生"
        android:checked="true"/>
    <RadioButton
        android:id="@+id/id2"
        style="@style/my_radiobutton_style"
        android:layout_marginLeft="10dp"
        android:text="教职工"/>
</RadioGroup>

```

☒ 学生 ☐ 教职工

RadioGroup 是一组单选按钮，我们可以直接对其进行控件位置的调整，就可以达到对其下所有单选按钮同时调整的效果。需要注意的是对每一个 RadioButton 都要设置相应的 id，该组单选按钮才能正常的工作。Checked 控制的是默认下的选择对象。具体属性在这里我通过了 style 设置，会在步骤 7 中提到。

6. 创建“登录”、“注册”按钮

```

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

```

```

<Button
    android:id="@+id/button1"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    style="@style/my_button_style"
    app:layout_constraintTop_toBottomOf="@+id/id0"
    app:layout_constraintRight_toLeftOf="@+id/button2"
    app:layout_constraintLeft_toLeftOf="parent"
    android:text="登录"
    app:layout_constraintHorizontal_chainStyle="packed"
    android:layout_marginTop="20dp" />

<Button
    android:id="@+id/button2"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    style="@style/my_button_style"
    app:layout_constraintTop_toBottomOf="@+id/id0"
    app:layout_constraintLeft_toRightOf="@+id/button1"
    app:layout_constraintRight_toRightOf="parent"
    android:layout_marginLeft="10dp"
    android:text="注册" />

```



两个按钮需要自定义背景边框来达到一个蓝色圆角的效果。具体要在 `drawable` 文件夹新建一个 `resource file`，然后通过对一个 `shape` 对象调整其 `padding`, `radius`, `solid` 等属性来设计出一个我们想要的背景效果，然后再通过 `button` 控件 `background` 对其的引用来实现目标按钮。而两个的布局设计，要求整体居中且相距 `20dp`，可以把它们相互连接成链条 `chain`，然后设置其 `chainstyle` 为 `packed` 让两个按钮左右紧紧相连，最后在 `marginLeft` 中让距离保持在 `20dp`。

7. 设置相关 style 及引用

```

<?xml version="1.0" encoding="utf-8"?>
<resources>
    <color name="colorPrimary">#3F51B5</color>
    <color name="colorPrimaryDark">#303F9F</color>
    <color name="colorAccent">#FF4081</color>
    <color name="colorblack">#000000</color>
    <color name="colorwhite">#ffffff</color>
</resources>

<resources>
    <dimen name="nomal_text">18sp</dimen>
</resources>

```

```

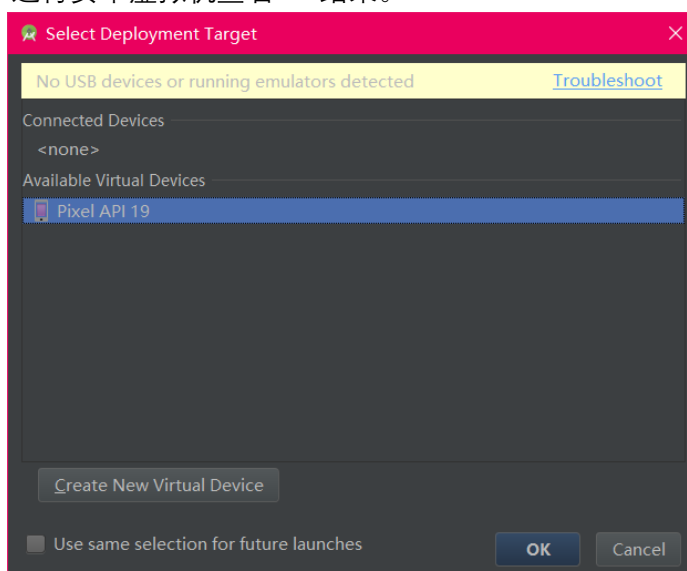
<style name="my_radiobutton_style">
    <item name="android:layout_height">wrap_content</item>
    <item name="android:layout_width">wrap_content</item>
    <item name="android:textColor">@color/colorblack</item>
    <item name="android:textSize">@dimen/nomal_text</item>
    <item name="android:layout_marginLeft">10dp</item>
</style>

<style name="my_button_style">
    <item name="android:layout_height">wrap_content</item>
    <item name="android:layout_width">wrap_content</item>
    <item name="android:textSize">@dimen/nomal_text</item>
    <item name="android:background">@drawable/button</item>
    <item name="android:textColor">@color/colorwhite</item>
    <item name="android:layout_marginTop">20dp</item>
</style>

```

在拓展知识中了解到我们可以在 styles.xml 文件中对指定控件的属性进行一种统一的设置，即定义一个 style 来给相关的控件引用。我在这里对单选按钮和按钮进行了 style 的定义，里面分别对其 textcolor, textsize 甚至是 background 等基础属性进行了设置，而在使用时（在前面的步骤图中），只要直接 style="" 就可以方便地完成重复的设置。

8. 运行安卓虚拟机查看 UI 结果。



(3) 实验遇到困难以及解决思路

1. 对于大小不同的控件，如何在水平上保持中心对齐。如本次实验中的输入框部分，输入框的大小比文本框大，未解决时左右相连会出现文本框在输入框左上位置的情况。

解决：把两个控件的 top、bottom 对应相连。



2. 在给按钮设置链式时，`app:layout_constraintHorizontal_chainStyle="packed"` 语句没有响应

解决：尝试过了重建项目，直接使用 design 设置等方法。最后是 SDK manager 中重新安装了最新的 constraintlayout 文件后，再新建了一个项目中才恢复了正常。

Support Repository		
<input checked="" type="checkbox"/>	ConstraintLayout for Android	Installed
<input checked="" type="checkbox"/>	Solver for ConstraintLayout	Installed
<input checked="" type="checkbox"/>	Android Support Repository	Installed
<input type="checkbox"/>	Google Repository	Update Available: 58

四、 课后实验结果



虚拟机中运行结果如图

五、 实验思考及感想

对于本次的实验，我们学到了一些 UI 的设计，虽然只是皮毛，但是基础也是十分重要的。在整个实验的过程中，可以了解到安卓 UI 编写的模式，创建控件，控制布局，然后利用 android: 语句来对控件属性进行编辑来达到目的的效果，十分简单高效。

这是接触安卓的第一次实验，AS 的使用对我来说还是十分新鲜的。首先作为一个全新的软件，我们在慢慢熟悉它后可以发现，作为一种比较新的系统，其实安卓中的代码编写和操作比起之前来说显

得非常方便。而 Lab1 我们只是简单地学习了一些控件的使用，但也可以感受到安卓能实现的功能的丰富性，AS 中还有很多东西等待我们去挖掘。

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。