中山大学移动信息工程学院本科生实验报告

(2017 学年秋季学期)

课程名称: Android 任课教师: 郑贵锋

年级&班级	15M3	专业 (方向)	软件工程 (移动信息工程)
学号	15352399	姓名	詹 巽霖
电话	13676010854	Email	1305699406@qq.com
开始日期	2017/9/25	完成日期	2017/9/25

一、实验题目

基本 UI 界面设计

二、实验内容

【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;

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

【实验过程】

1. 资源的配置:

首先是根据实验要求配置相应的资源,这里的资源即指文本信息,对应颜色,背景 backgroud,风格 style 等等。

colors.xml 里进行颜色的定义:

可以在网上查询到不同的颜色对应的编号,这里我们根据颜色编号定义了五种颜色,建立 project 自带的三种和白色、黑色。

string.xml 里进行文本信息的定义,所定义的文本信息都是 string 类型,这些文本信息都是在之后的布局中需要用到的。虽然可以在布局的时候用 text="XXXXX",但是那是一种不规范的做法,比较规范的是用引用的方式,在需要对某个文本进行修改的时候就不需要一个一个进行修改,修改对应的定义信息即可。具体如下:

```
⟨resources⟩
    ⟨string name="app_name"⟩Lab1⟨/string⟩
    ⟨string name="text_title"⟩中山大学学生信息系统⟨/string⟩
    ⟨string name="student_ID"⟩学号:⟨/string⟩
    ⟨string name="password"⟩密码:⟨/string⟩
    ⟨string name="student"⟩学生⟨/string⟩
    ⟨string name="student"⟩学生⟨/string⟩
    ⟨string name="input_ID"⟩请输入学号⟨/string⟩
    ⟨string name="input_password"⟩请输入密码⟨/string⟩
    ⟨string name="login"⟩登陆⟨/string⟩
    ⟨string name="login">登陆⟨/string⟩
    ⟨string name="login">登陆⟨/string⟩
    ⟨string name="register">注册⟨/string⟩
</resources⟩
</pre>
```

以上两个文件都是定义在 res/values 的文件夹中。

background.xml 里进行背景的定义,这里主要是用在按钮那里,因为按钮的

背景在布局文件中是不能设置出那种效果的,所以我们需要单独定义按钮的背景。

在 res/drawable 创建相应的.xml 文件,将 selector 改成 shape, shape 有多个属性, corners 对应四个边角的圆滑程度, solid 对应颜色, padding 对应在文本基础上扩展出相应的宽度高度。

mipmap.xml 存放相应的照片,这里存放 sysu.jpg (即中山大学的校徽图片)

2. 主布局文件

这里用的是约束布局,约束布局是根据其他元素或视图来确定 View 在屏幕中的位置,受到三类约束,即其他视图,父容器,基准线的约束。用法模板为:

layout_constraint[本源位置]_[目标位置]="[目标 ID]" 例如: app:layout constraintBottom toBottomOf="parent"

一、"中山大学学生信息系统"这个控件的显示:

```
android:id="@+id/text_title"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:text="@string/text_title"
android:textSize="20sp"
android:textColor="@color/colorBlack"
app:layout_constraintLeft_toLeftOf="parent"
app:layout_constraintTop_toTopOf="parent"
app:layout_constraintRight_toRightOf="parent"
android:layout_marginTop="20dp"/>
```

定义好相应的 id, text 内容, 颜色 textColor, 分别用三个链条来约束住这个控件, 水平方向上的居中需要左右两条链条, 上面一条将其约束在顶部, 最后设置距离

顶部 20dp 的距离。

二. 图片的显示:

```
android:id="@+id/imageView"
android:layout_width="100dp"
android:layout_height="100dp"
android:src="@mipmap/sysu"
app:layout_constraintLeft_toLeftOf="parent"
app:layout_constraintRight_toRightOf="parent"
android:layout_marginTop="20dp"
app:layout_constraintTop_toBottomOf="@+id/text_title"/>
```

与上面那个控件基本类似,也是定义 id,设置长度宽度,引入图片,需要注意的是这里的的竖直约束条件是用标题的那个控件来控制的,在它下方 20dp。用 layout marginTop 来控制距离距离。

三、输入学号密码等控件的设计:

学号:	请输入学号	
密码:	请输入密码	

基本的设计方法跟上面的控件是相似的,这里用到的分别是 textView 和 EditVieew 两种控件。

该部分遇到的问题及解决思路:

1. 因为这两个空间的大小并不一样, 所以如果两个控件都与上面的校徽图片对 齐相接的话会导致水平方向上二者是不在同一直线上的, 效果是这样的:



这种情况我刚开始一直在调水平方向的链接,后来意识到这是竖直方向上的问题,所需要的调整是学号的控件顶部约束于请输入学号的顶部,同时底部受约束于请输入学号的底部。这样学号那个控件才能显示得居中。同样的道理密码和请输入密码控件也是一样的操作。

2. 因为密码输入之后要求是小点,所以我们需要设置文本的输入类型,即 Android:inputType="textPassword",为了在输入前显示请输入密码,我们不能 简单的用 text="请输入密码",而是要用 hint="请输入密码"

四、单选按钮的显示:



单选按钮的显示用到的控件是 RadioGroup, 里面包含两个 RadioButton 的按

钮。也就是说我们设计这两个按钮在全局上的布局要用 RaduoGroup 这个控件,但是当涉及到这两个按钮之间的位置关系时我们就需要对 RadioButton 来操作。用 checked 属性来控制哪个按钮是被选上了。

```
RadioGroup
   android: layout_width="wrap_content"
   android: layout_height="wrap_content"
   android:id="@+id/ido"
   app:layout_constraintLeft_toRightOf="parent"
   app:layout_constraintTop_toBottomOf="@id/text_input_password"
   app:layout_constraintRight_toLeftOf="parent"
   android:orientation="horizontal">
   <RadioButton</pre>
       android:id="@+id/student_id"
       android: layout_width="wrap_content"
       android:layout_height="wrap_content"
       android:textSize="18sp"
       app:layout_constraintRight_toLeft0f="@id/staff_id"
       android: layout_marginRight="10dp"
       android:checked="true"
       android:text="@string/student"/>
   <RadioButton</pre>
       android:id="@+id/staff_id"
       android: layout_width="wrap_content"
       android:layout_height="wrap_content"
       android:textSize="18sp"
       app:layout_constraintLeft_toRightOf="@id/student_id"
       android:layout_marginLeft="10dp"
       android:text="@string/staff"/>
</RadioGroup>
```

五、登陆注册按钮的设计:



该部分遇到的问题及解决思路:

由于这两个按钮都在中间,所以我之前用链条左边接右边的做法是不能实现这个效果的,那样的话两个按钮之间的距离很大,所以我是用一个大的约束布局把这两个按钮包含起来,直接将这个大的约束布局放在中间,此时布局里的两个按钮是紧挨着的,所以再分别对这两个按钮进行相应的位置操作,即这两个按钮用链条链接起来然后设置之间的距离。

```
<android. support. constraint. ConstraintLayout</p>
    android:layout_width="wrap_content"
    android: layout_height="wrap_content"
    android:layout_marginTop="20dp"
    app:layout_constraintTop_toBottom0f="@+id/ido"
    app:layout_constraintRight_toRightOf="parent"
    app:layout_constraintLeft_toLeftOf="parent">
        Button
           android:id="@+id/Button1_id"
           android: layout_width="wrap_content"
           android: layout_height="wrap_content"
           android:text="@string/login"
           android:textSize="18sp"
           android:textColor="@color/colorWhite"
           android:background="@drawable/backgroud"
           app:layout_constraintRight_toLeftOf="@id/Button2_id"
           android:layout_marginRight="10dp">
       </Button>
        Button
           android:id="@+id/Button2_id"
           android: layout_width="wrap_content"
           android: layout_height="wrap_content"
           android:text="@string/register"
           android:textSize="18sp"
           android:textColor="@color/colorWhite"
           android:background="@drawable/backgroud"
           android:layout_marginLeft="10dp"
           app:layout_constraintLeft_toRightOf="@id/Button1_id">
        </Button>
</android. support. constraint. ConstraintLavout>
```

三、实验结果

左边是刚开机的界面,右边是输入相应信息之后:





四、实验思考与总结

这次的实验主要内容是进行 Android 的界面设计,全都是通过 XML 代码来实现的,在这次实验中,由于我试了几种布局,所以体会到了几种布局的不同之处和各自的优缺点,同时也使用了许多控件,例如 Button, EditText 等等,对如何使用 XML 语言对 Android 界面进行排版有了一些比较基础的了解。dimens, strings 和 styles 三个 XML 可以看成是一些变量的库,我们将变量保存在这些库里,使用的时候就可以直接调用他们,在修改的时候也可以在这些 XML 文件里修改,这样,代码的风格更好,可读性更强,降低了代码的耦合性。

在这次实验中,也体会到了 Android 开发需要兼容不同手机的这个因素,例如 TableLayout 这种布局就对系统的兼容性更高。以后还需要多加练习,提高对控件属性的熟悉度。