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(移动信息工程) |
|------|-------------|-----------|----------------------------|
| 学号 | 15352033 | 姓名 | 陈黄胤 |
| 电话 | 18588741286 | Email | chenhy98@mail2.sysu.edu.cn |
| 开始日期 | 2017.9.28 | 完成日期 | 2017.10.8 |

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;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

首先创建一个 textview 控件,其内容为"中山大学学生信息系统",并且按要求设置 text 的格式。然后设置控件的布局,首先与 parent 建立约束,设置该控件在界面的居中,并设置与 parent 的顶部距离 20dp。

```
Android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:text="@string/title_name"
android:textSize="20sp"
android:textStyle="bold"

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

再创建一个 ImageView 控件,设置好与 parent 的约束,并让控件的顶部与上面的 textView 控件底部对齐,并设置好图片路径

```
ImageView
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:src="@drawable/s"
android:contentDescription="@string/title_name"

app:layout_constraintLeft_toLeftOf="parent"
app:layout_constraintRight_toRightOf="parent"
app:layout_constraintTop_toBottomOf="@+id/textView"

android:layout_marginTop="18dp"
android:id="@+id/image" />
```

再创建两个 TextView+EditText 的搭配控件,对它们设置好彼此的约束条件,然后再设置对于上面图片的约束条件。最后设置好它们本身的内容样式。

```
<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:textSize="18sp"
    android:layout_marginStart="30dp"
    android:layout_marginTop="30dp"
    android: id="@+id/text2"
    app:layout_constraintTop_toBottomOf="@+id/image"
    app:layout_constraintLeft_toLeftOf="parent"
ŒditText
   android:layout_width="wrap_content"
    android:layout_height="wrap_content"
   android:textSize="18sp"
    android:inputType="number"
    android:layout_marginTop="18dp"
    android:ems="13"
    android: id="@+id/ed1"
    app:layout_constraintTop_toBottomOf="@+id/image"
    app:layout_constraintLeft_toRightOf="@+id/text2"
```

再创建单选控件。首先创建一个单选组控件 RadioGroup,设置为横向排列 horizontal,然后设置好对 parent 的约束,保证居中,以及和上面输入框的约束条件。然后在里面再建立两个 RadioButton 控件,设置好彼此之间的距离约束

```
Radi oGroup
   android:layout_height="wrap_content"
   android:layout_width="wrap_content"
   android:layout_marginTop="20dp"
   android: orientation="horizontal"
    android: checkedButton="@+id/id1"
   app:layout_constraintLeft_toLeftOf="parent"
   app:layout_constraintRight_toRightOf="parent"
   app:layout_constraintTop_toBottomOf="@id/text3">
   RadioButton
       android:id="@+id/id1"
       android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:textSize="18sp"
        android:layout_marginEnd="10dp"
   RadioButton
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:textSize="18sp"
        android:layout_marginStart="10dp"
</RadioGroup>
```

最后是两个按钮控件的设置,使用 packed chain 来关联两个按钮,设置好居中以及两者之间的 距离。按照 packed chain 的要求设置好

```
Button
    android:layout_width="88dp"
    android:layout_height="wrap_content"
    android:layout_marginTop="20dp"
    android:layout_marginEnd="10dp"
    android:textSize="18sp"
    android:background="@drawable/my_button"
    android:id="@+id/sign_in"
    app:layout_constraintHorizontal_chainStyle="packed"
    app:layout_constraintLeft_toLeftOf="parent"
    app:layout_constraintTop_toBottomOf="@+id/id0"
    app:layout_constraintRight_toLeftOf="@+id/sign_up"/>
Button
    android: layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginTop="20dp
    android:layout_marginStart="10dp"
    android:textColor="@color/White"
    android:textSize="18sp"
    android:background="@drawable/my_button"
    app:layout_constraintRight_toRightOf="parent
    app:layout_constraintTop_toBottomOf="@+id/id0"
    app:layout_constraintLeft_toRightOf="@+id/sign_in"/>
```

同时对 backgroud 进行设置:

实现了对按钮形状颜色的编辑

(3) 实验遇到困难以及解决思路

一开始对两个按钮进行布局的时候遇到了困难,设置了两个按钮的约束情况,包括彼此之间的约束以及与 parent 的约束,但是两个按钮就是无法居中显示,而且距离也没法调整到

需要的值。向同学请教之后得知要把 chain_style 设置成 packed。但是设置之后没法通过 gradle 的编译。最后发现要安装更高版本的 constrainStyle 库,设置好之后终于成功了。

四、 课后实验结果



五、 实验思考及感想

初次接触到 Android Studio 并初步接触到 APP 的布局设置,难免会遇到各种各样的问题。而遇到问题应该多查阅官方手册,因为很多问题都能在官方的使用手册中找到详细的解决方法。虽然 Google 为了降低 Android APP 开发的门槛,在 Android Studio 中加入了大量的可视化操作,但我觉得不仅要理解原理,更应该有读写相关代码的原理,不能完全依赖于这些可视化操作。