

# 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 )	移动信息工程
学号	15352124	姓名	黄浩伦
电话	13928606193	Email	<u>564945308@qq.com</u>
开始日期	2017.10.15	完成日期	2017.10. 15

### 一、 实验题目

事件处理

### 二、 实现内容

实现一个 Android 应用，界面呈现与实验一基本一致，要求：

(1) 该界面为应用启动后看到的第一个界面



(2) 输入学号和密码的控件要求用 `TextInputLayout` 实现

(3) 点击图片，弹出对话框如下图：



点击“拍摄”选项，弹出 Toast 信息“您选择了[拍摄]”；

点击“从相册选择”选项，弹出 Toast 信息“您选择了[从相册选择]”；

点击“取消”按钮，弹出 Toast 信息“您选择了[取消]”。

（4） 切换 RadioButton 的选项，弹出 Snackbar 提示“您选择了 xx”；

例如从选项“学生”切换到选项“教职工”，则提示“您选择了教职工”；



点击 Snackbar 上的“确定”按钮，则弹出 Toast 信息“Snackbar 的确定按钮被点击了”

（5） 点击登录按钮

依次判断学号是否为空，密码是否为空，用户名和密码是否正确（正确的学号和密码分别为“123456”，“6666”）；不正确则给出错误信息，如学号和密码都正确则提示“登陆成功”，如图：

请输入学号

学号不能为空

请输入密码

☐ 学生 ☒ 教职工

登录 注册

请输入学号

1234567

请输入密码

密码不能为空

☐ 学生 ☒ 教职工

登录 注册

请输入学号

1234567

请输入密码

.....

☐ 学生 ☒ 教职工

登录 注册

学号或密码错误 确定

请输入学号

123456

请输入密码

.....

☐ 学生 ☒ 教职工

登录 注册

登录成功 确定

(6) 点击注册按钮

如果切换选项时，RadioButton 选中的是“学生”，那么弹出 Snackbar 信息“学生注册功能尚未启用”，如果选中的是“教职工”，那么弹出 Toast 信息“教职工注册功能尚未启用”。

☒ 学生 ☐ 教职工

登录 注册

学生注册功能尚未启用 确定

### 三、课堂实验结果

(1) 实验截图



## (2) 实验步骤以及关键代码

1. 在 Android Studio 的 Layout 文件的代码文件中找到原本 editText 的地方改成使用 TextInputLayout，即在每个 editText 的地方加上一个 TextInputLayout 作为它的 parent 将它住，且一个 TextInputLayout 只能有一个 editText 作为它的子元素

```
<android.support.design.widget.TextInputLayout
    android:id="@+id/textinput1"
    android:layout_width="287dp"
    android:layout_height="wrap_content"
    android:layout_marginLeft="8dp"
    android:layout_marginRight="8dp"
    android:layout_marginTop="25dp"
    app:layout_constraintHorizontal_bias="0.508"
    app:layout_constraintLeft_toLeftOf="parent"
    app:layout_constraintRight_toRightOf="parent"
    app:layout_constraintTop_toBottomOf="@+id/imageView">

    <EditText
        android:id="@+id/edittext1"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_weight="1"
        android:textColor="@color/colorDefault"
        android:hint="请输入学号"
        android:inputType="number"
        android:paddingBottom="10dp" />

</android.support.design.widget.TextInputLayout>
```

2. 要让点击图片弹出一个 AlertDialog 框，则要为点击图片定义一个 onclick 的监听器

```
View mImage;  
mImage = (ImageView) findViewById(R.id.imageView);  
mImage.setOnClickListener(new clickImage());
```

由于弹出的 AlertDialog 框是以列表形式展示内容的，则要用 setItems 给它设置内容并且给内容设置一个点击的 onclick 监听器，当点击列表中的一项时，根据点击的列表项弹出含有相应内容的 toast

```
AlertDialogBuilder.setItems(items, new DialogInterface.OnClickListener() {  
    @Override  
    public void onClick(DialogInterface dialogInterface, int i) {  
        switch (i) {  
            case 0:  
                Toast.makeText(MainActivity.this, "您选择了[拍摄]", Toast.LENGTH_SHORT).show();  
                break;  
            case 1:  
                Toast.makeText(MainActivity.this, "您选择了[从相册中选择]", Toast.LENGTH_SHORT).show();  
                break;  
            default:  
                Toast.makeText(MainActivity.this, "您选择了[拍摄]", Toast.LENGTH_SHORT).show();  
                break;  
        }  
    }  
});
```

并为这个 AlertDialog 设置一个“取消”按钮和这个按钮的 onclick 监听器，点击“取消”按钮后可以弹出含有相应内容的 toast

```
AlertDialogBuilder.setNegativeButton("取消", (dialogInterface, i) -> {  
    Toast.makeText(MainActivity.this, "您选择了[取消]", Toast.LENGTH_SHORT).show();  
});
```

3. 为 radiogroup 设置一个 onCheckedChangeListener 的监听器，监听当选择的 radio 按钮改变时做出的反应

```
RadioGroup mRadioGroup = (RadioGroup) findViewById(R.id.button0);  
mRadioGroup.setOnCheckedChangeListener(new changeRadio());
```

当选中一个 radio 按钮后，通过 getId 判断两个单选哪个被选中，并且通过 getText() 函数获取 radio 按钮中的值，然后根据这些值弹出含有相应内容的 snackbar

```

public void onCheckedChanged(RadioGroup group, int checkedId) {
    String temp = null;
    RadioButton rb1 = (RadioButton) findViewById(R.id.radioButton);
    RadioButton rb2 = (RadioButton) findViewById(R.id.radioButton2);
    if(rb1.getId() == checkedId){
        temp = rb1.getText().toString(); //取得单选文本
    }
    if(rb2.getId() == checkedId){
        temp = rb2.getText().toString(); //取得单选文本
    }

    temp = "您选择了"+temp; //设置文本信息
    final Snackbar snack = Snackbar.make(group, temp, Snackbar.LENGTH_SHORT);
    snack.setAction("确定", (v) -> { snack.dismiss(); })
        .setActionTextColor(getResources().getColor(R.color.colorPrimary))
        .setDuration(5000)
        .show();
}

```

4. 然后再给“登录”按钮设置一个 onclick 的监听器

```

Button btn1 = (Button) findViewById(R.id.button);
btn1.setOnClickListener(new clickLogin());

```

点击登录后，需要做三重判断，先判断学号那里的 editText 的文本是否为空，再判断密码的 editText 是否为空，最后都不为空则判断学号密码是否正确，正确的话则弹出登录成功的 snackbar（注意这里字符串做比较判断时，要用 isEmpty 和 equals，不能直接用==）

```

if(TextUtils.isEmpty(str1)){
    temp = "学号不能为空";
    TextInputLayout t = (TextInputLayout) findViewById(R.id.textinput1);
    t.setErrorEnabled(true);
    t.setError(temp);
}
else{
    if(TextUtils.isEmpty(str2)){
        temp = "密码不能为空";
        TextInputLayout t = (TextInputLayout) findViewById(R.id.textinput2);
        t.setErrorEnabled(true);
        t.setError(temp);
    }
    else {
        TextInputLayout t = (TextInputLayout) findViewById(R.id.textinput2);
        t.setErrorEnabled(false);
        TextInputLayout t1 = (TextInputLayout) findViewById(R.id.textinput1);
        t1.setErrorEnabled(false);
        if(str1.equals("123456") && str2.equals("6666")) {
            temp = "登录成功";
        }
        else {
            temp = "学号或密码错误";
        }
        final Snackbar snack = Snackbar.make(v, temp, Snackbar.LENGTH_SHORT);
        snack.setAction("确定", (v) -> { snack.dismiss(); })
            .setActionTextColor(getResources().getColor(R.color.colorPrimary))
            .setDuration(5000)
            .show();
    }
}
}

```

5. 相类似地，给“注册”按钮设置一个 onclick 监听器

```
Button btn2 = (Button) findViewById(R.id.button2);  
btn2.setOnClickListener(new clickSign());
```

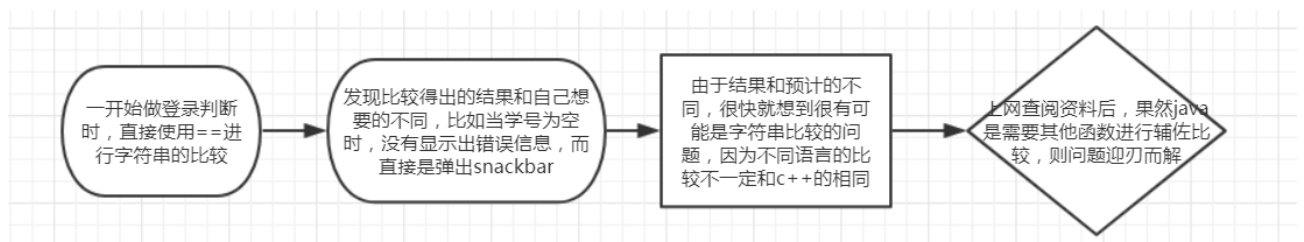
然后利用 `getCheckedRadioButtonId` 函数获取当前被选择的 radio 按钮，再通过 `getText` 函数获取其中的文本，然后弹出含有相应内容的 snackbar

```
RadioButton radioButton = (RadioButton) findViewById(rg1.getCheckedRadioButtonId());  
String selectText = radioButton.getText().toString();  
  
temp = selectText + "注册功能尚未启用";  
if(selectText.equals("学生")){  
    TextInputLayout t = (TextInputLayout) findViewById(R.id.textinput1);  
    final Snackbar snack = Snackbar.make(v, temp, Snackbar.LENGTH_SHORT);  
    snack.setAction("确定", (v) -> { snack.dismiss(); })  
        .setActionTextColor(getResources().getColor(R.color.colorPrimary))  
        .setDuration(5000)  
        .show();  
}  
else{  
    Toast.makeText(MainActivity.this, temp, Toast.LENGTH_SHORT).show();  
}
```

### (3) 实验遇到困难以及解决思路

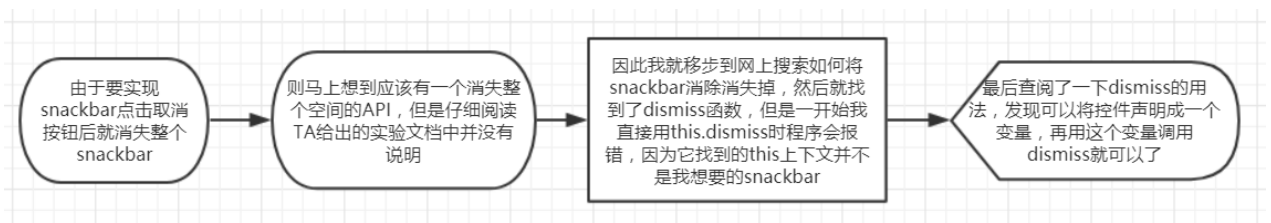
**困难：**字符串进行比较时，直接用了==号进行比较，没有用 equals 函数和 isEmpty 函数

**解决思路：**



**困难：**不知道使用 dismiss 函数将弹出来的 snackbar 直接进行删除隐藏

**解决思路：**



## 四、 课后实验结果



## 五、 实验思考及感想

本次的实验相比于上次进行页面布局的实验要难的多，首先就是要进行代码的编写和理解，并且首次接触 Android Java 的编写，对很多函数和类名会有点陌生。特别在进行添加控件的监听器时，经常会不理解添加时声明类的作用和含义是什么，后来在网上翻看资料，并且尝试不用匿名函数的方式进行声明，而将一个实现事件响应的函数写成一个单独的类后，就开始有点理解这整个设置监听器并处理响应的过程了。然后在整个代码的编写过程中，比较大的阻碍就是对语法还不是特别熟悉，特别是一下细节的地方，如 final 变量的声明、字符串的比较判断，等等一些需要语法等基本功的东西并不是特别的熟悉。但在实验中，通过一直百度查阅资料，翻看博客等对函数的介绍，使我的自学能力和对 android 的理解逐渐地加深，并且在实验中不断自我琢磨和思考理解的过程也让我慢慢地找到了学习安卓的兴趣，希望日后可以更进一步，在期末项目中发挥自己的实力。