

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
学号	15352441	姓名	钟丹彬
电话	18927386861	Email	2229690353@qq.com
开始日期		完成日期	

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：

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”。

（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”。

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 xml 文件中，加入 TextInputLayout，用于接受用户输入，与 EditText（请输入学号、请输入密码）配合。

```
<android.support.design.widget.TextInputLayout
    android:id="@+id/editText_layout"
    android:layout_width="0dp"
    android:layout_height="80dp"
    app:layout_constraintRight_toRightOf="parent"
    app:layout_constraintTop_toBottomOf="@+id/imageView"
    android:layout_marginTop="10dp"
    android:layout_marginLeft="20dp"
    android:layout_marginRight="20dp"
    app:layout_constraintLeft_toLeftOf="parent"
    app:hintEnabled="true"
    android:layout_marginStart="20dp"
    android:layout_marginEnd="20dp">
    <EditText
        android:id="@+id/editText"
        android:layout_width="match_parent"
        android:layout_height="40dp"
        android:textSize="18sp"
        android:inputType="number"
        android:hint="请输入学号"
    />
</android.support.design.widget.TextInputLayout>
```

使用 TextInputLayout 时，TextInputLayout 和其包含的 EditText 都需要表明自己的 id。并且，把约束布局写在 TextInputLayout 内。

2. 在 MainActivity.java 文件中，写入事件处理代码。首先在 java 文件中使用 findViewById() 引用布局文件中的控件。

```
mImage = (ImageView) findViewById(R.id.imageView);
mUser = (TextInputLayout) findViewById(R.id.editText_layout);
mPassword=(TextInputLayout) findViewById(R.id.editText2_layout);
mRadioG = (RadioGroup) findViewById(R.id.radiogroup);
mStudent= (RadioButton) findViewById(R.id.button1);
mTeacher= (RadioButton) findViewById(R.id.button2);
mLogin = (Button) findViewById(R.id.button3);
mRegist = (Button) findViewById(R.id.button4);
mUser_edit = (EditText) findViewById(R.id.editText);
mPassword_edit = (EditText) findViewById(R.id.editText2);
```

通过 id 获得控件后，就可以在后续操作中，直接使用定义的名字即可。

3. 处理点击图片，弹出对话框的事件。该事件使用了“对话框”、“Toast”。

点击对话框，需要设置事件监听。

```
//图片信息
//创建 AlertDialog.Builder 对象
final AlertDialog.Builder mbuilder = new AlertDialog.Builder(this);
//设置对话框项目
final String[] items = new String[]{"从相册上传","拍照"};
//对 mImage 事件监听
mImage.setOnClickListener(new ImageView.OnClickListener() {
    @Override
    public void onClick(View v){
        mbuilder.setTitle("上传头像")
            .setItems(items,new DialogInterface.OnClickListener(){
                @Override
                public void onClick(DialogInterface dialogInterface,int i){
                    Toast.makeText(MainActivity.this,"您选择了
```

```

"+items[i],Toast.LENGTH_SHORT)
        .show();
    }
})
.setNegativeButton("取消",new DialogInterface.OnClickListener(){
    @Override
    public void onClick(DialogInterface dialogInterface,int i){
        Toast.makeText(MainActivity.this,"您选择了取消",Toast.LENGTH_SHORT)
        .show();
    }
});
mbuilder.create().show();
}
});

```

4. 当点击 “请输入学号” 、 “请输入密码” 时，需要弹出 TextInputLayout 的提示：

```

//输入学号、密码提示
mUser.setHint("请输入学号");
mPassword.setHint("请输入密码");

```

5. 当选择 “学生” 、 “教职工” 按钮时，因为这两个按钮在同一 RadioGroup 里面，所以使用 mRadioGroup.setOnCheckedChangeListener 监听，判断选中的按钮，返回至 int id 中，再判断此 id 属于 “学生” 还是 “教职工” 。注意到后续步骤的 “注册” 按钮事件处理与 “学生” 、 “教职工” 按钮选择有关，所以将其事件一并写入 RadioGroup 内。这里使用了 “Snackbar” 、 “Toast” 。

```

//学生、教职工选择,radioGroup 用 setOnCheckedChangeListener
mRadioG.setOnCheckedChangeListener(new RadioGroup.OnCheckedChangeListener){

```

```

@Override
public void onCheckedChanged(RadioGroup RG,int id){
    if(id == mStudent.getId()){
        Snackbar.make(mRadioG,"您选择了学生",Snackbar.LENGTH_SHORT)
            .setAction("确定",new View.OnClickListener(){
                @Override
                public void onClick(View view){
                    Toast.makeText(MainActivity.this,"Snackbar 的确定按钮被点击了",Toast.LENGTH_SHORT)
                        .show();
                }
            })
            .setActionTextColor(getResources().getColor(R.color.colorPrimary))
            .setDuration(5000)
            .show();

        mRegist.setOnClickListener(new Button.OnClickListener(){
            @Override
            public void onClick(View view){
                Snackbar.make(mRegist,"学生注册功能尚未启用",Snackbar.LENGTH_SHORT)
                    .setAction("确定",new View.OnClickListener(){
                        @Override
                        public void onClick(View view){}
                    })
                    .setActionTextColor(getResources().getColor(R.color.colorPrimary))
                    .setDuration(5000)
                    .show();
            }
        });
    }
}

```

6. 点击登录按钮，依次判断学号是否为空、密码是否为空、学号密码是否正确（123456，6666）。这里需要对 mLogin 进行事件监听，当其被按下时，对其进行事件处理。我们需要得到在 mUser_edit 和 mPassword 上的字符串，通过对字符串进行判断，做出相应的事件反应。比如，当学号输入为空时，mUser 需要显示错误信息“学号不能为空”。

*//点击登录按钮，依次判断学号是否为空、密码是否为空、学号密码是否正确
(123456，6666)*

```
mLogin.setOnClickListener(new View.OnClickListener(){
    @Override
    public void onClick(View view){
        String number = mUser_edit.getText().toString();
        String password = mPassword_edit.getText().toString();
        if(TextUtils.isEmpty(number)){
            mUser.setErrorEnabled(true);
            mUser.setError("学号不能为空");
        }
        else if(TextUtils.isEmpty(password)){
            mPassword.setErrorEnabled(true);
            mPassword.setError("密码不能为空");
            mUser.setErrorEnabled(false);
        }
        else if(number.equals("123456") && password.equals("6666")){
            mPassword.setErrorEnabled(false);
            Snackbar.make(mLogin,"登录成功",Snackbar.LENGTH_SHORT)
                .setAction("确定",new View.OnClickListener(){
                    @Override
                    public void onClick(View view){}}
                )
                .setActionTextColor(getResources().getColor(R.color.colorPrimary))
                .setDuration(5000)
```

```

        .show();
    }
    else{
        mPassword.setErrorEnabled(false);
        Snackbar.make(mLogin,"学号或密码错误",Snackbar.LENGTH_SHORT)
            .setAction("确定",new View.OnClickListener(){
                @Override
                public void onClick(View view){}
            })
            .setActionTextColor(getResources().getColor(R.color.colorPrimary))
            .setDuration(5000)
            .show();
    }
}
});

```

(3) 实验遇到困难以及解决思路

1. 在写 ImageView 的事件监听时，总是会报错。后来发现是自己没有包入正确的头文件 import android.view.View;
2. 在获取 TextInputLayout 时，总是会报错。后来发现定义类型时，要定义在 AppCompatActivity 下面，而不是 onCreate 里面。并且，在标错的地方按下“alt,enter”键即可。
3. 在不熟悉使用 OnClickListener 的情况下，不知道每一个 OnClickListener 都需要对应一个“public void onClick”。
4. 在写 RadioGroup 的事件处理时，错误地使用了 OnClickListener。后面详读课件后才发现错误。应该使用 OnCheckedChangeListener。我们需要在 public void onCheckedChanged(RadioGroup,int)中返回 RadioGroup 中被选中的 button 的 ID。

四、 课后实验结果

(1) 应用启动后看到的第一个界面



(2) 点击图片，弹出对话框如下：



点击“从相册选择”选项，弹出 Toast 信息“您选择了[从相册选择]；



(点击其他选项，显示类似)

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



(5) 点击登录按钮

依次判断学号是否为空

中国电信 55% 13:39

Lab2

中山大学学生信息系统



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

密码是否为空

中国电信 55% 13:39

Lab2

中山大学学生信息系统



请输入学号

1111

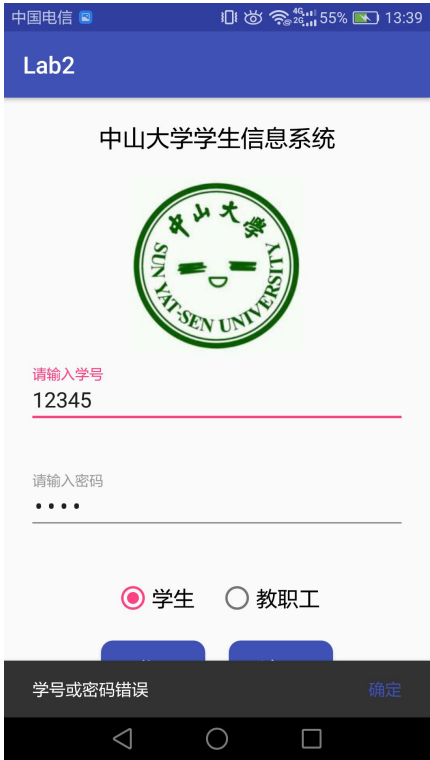
请输入密码

密码不能为空

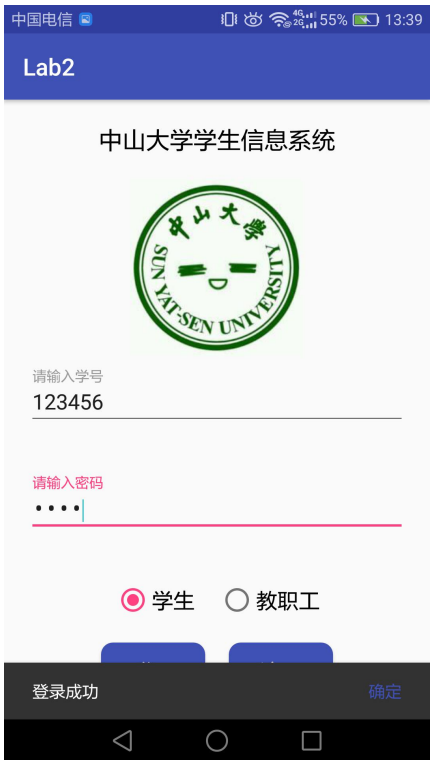
☒ 学生 ☐ 教职工

登录 注册

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”。



(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”。



五、 实验思考及感想

在此次实验中，我们学习在 java 文件中写入相应的事件处理。我主要学习到了事件监听的使用、TextInputLayout、弹出 Toast 信息、简单对话框的使用、弹出 Snackbar 信息。在这次实验中，Android Studio 经常报错，即使我完全按照课件上的步骤写。在查阅许多了相应的博客后，我主要总结了几种报错原因：1. 没有 import 正确的头文件；2. 错误信息残留在缓存中，需要重新 build 或者 clean project 等；3. 因为对 java 语言的不熟悉，会写错格式；4. 在写 ImageView.OnClickListener()时报错了，那个时候还不知道要继续写完 public void onClick 才行，就一直盲目地找报错原因。

总结下来，还是因为自己对这种语言，这种代码方式不熟悉，导致会犯很多令人摸不着头脑的错误。而且，写完实验后，其实还是有许多不明白的地方，只是按照要求写好了代码。为什么要这样写，为什么要写在这个地方，有很多还是不明白。希望在后面的实验中能一一解惑。