

1 软件名称

完成人：钟昌宏

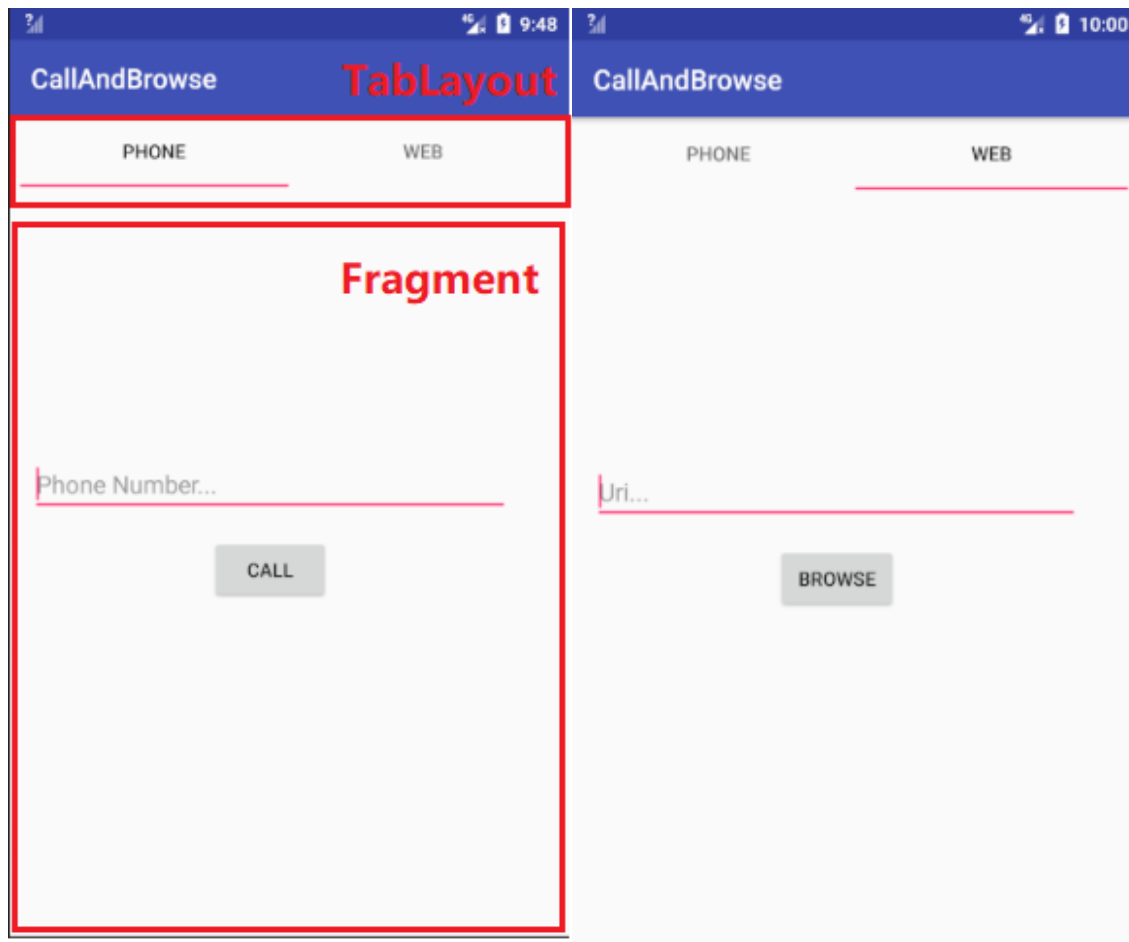
学号：20152100127

完成时间：2017/11/17

2 软件内容简介

实现一个 App，可以打电话和浏览网页。（调用系统自带的打电话功能实现拨打电话，用 WebView 实现在本 App 内浏览网页）

3 界面设计



基本设计思想是顶部一个 TabLayout，让用户滑动选择拨打电话和浏览网页；下面是一个 ViewPager，ViewPager 里嵌套着 Fragment，分别设计两个 Fragment 用于打电话和浏览网页，最后将 TabLayout 和 ViewPager 关联起来。

4 代码设计

4.1 界面

```
tabs = (TabLayout) findViewById(R.id.tabLayout);
pages = (ViewPager) findViewById(R.id.viewPager);

titles = new String[] {getString(R.string.phone), getString(R.string.web)};

fragmentList = new ArrayList<>();
fragmentList.add(new PhoneCallPage());
fragmentList.add(new BrowseWebPage());

adapter = new MyAdapter(getSupportFragmentManager());
pages.setAdapter(adapter); //嵌套
tabs.setupWithViewPager(pages);
```

1) 这个用到了国际化的设计思想，在 XML 文件中可以直接用@string/来表达，在.java 的文件中需要用 getString(R.string.xxx)来获取。

2) Myadapter 是自己定义的继承 FragmentPagerAdapter 的类，里面重载了 getItem, getPageTitle 等等函数，用户获取 Fragment 和其标题。

3) ViewPager.setAdapter 用于设置适配器，TabLayout.setupWithViewPager 用于关联 ViewPager。

4.2 权限

```
<!--添加可以向外拨打电话的权限 -->
<uses-permission android:name="android.permission.CALL_PHONE"/>
<!--上网权限 -->
<uses-permission android:name="android.permission.INTERNET"/>
```

打电话和浏览网页都涉及 App 权限问题，所以要在 AndroidManifest.xml 里加入这两行来授予权限。由于打电话是危险权限，在这里加了还不够，在执行代码的时候还要进行一次判断，判断本 App 是否已经获取了权限，如果没有则弹出对话框请求获取，如果已获取则继续执行动作。

4.3 打电话

```
myBtnCall.setOnClickListener((v) → {  
    if(ContextCompat.checkSelfPermission(getContext(),  
        Manifest.permission.CALL_PHONE)  
        != PackageManager.PERMISSION_GRANTED) { //if have no permission, request it.  
        ActivityCompat.requestPermissions(getActivity(),  
            new String[] {Manifest.permission.CALL_PHONE},  
            MY_PERMISSIONS_REQUEST_CALL_PHONE);  
    } else { //have permission  
        String number = myEditTextPhoneNumber.getText().toString();  
        if(number.equals(""))  
            return;  
        Uri uri = Uri.parse("tel:"+number);  
        Intent intent = new Intent(Intent.ACTION_CALL,uri);  
        startActivity(intent);  
    }  
}); //setOnClickListener
```

1) 一开始先判断 `Manifest.permission.CALL_PHONE` 这个权限有没有被授予，如果没有则进入 `if` 语句执行获取权限代码；如果有则进入 `else` 语句执行打电话代码。

2) 此处是主动获取权限代码，执行该代码会弹出小窗口让用户选择是否允许，`new String` 里放的是要申请的权限，此处是 `CALL_PHONE`，如果要申请多个权限，比如读取存储、获取位置等等，可以放多个进去。

3) `MY_PERMISSIONS_REQUEST_CALL_PHONE` 是一个 `constant` 型的整型数，自己随意定义。这个参数的意义在于，可以在重载的 `onRequestPermissionsResult` 函数里，根据获取的权限，执行相应的代码。

4) 新建一个 `Intent`，执行自带的事件 `Intent.ACTION_CALL`，第二个参数是 `Uri` 类型的数据。

4.4 浏览网页

```
myBtnBrowse.setOnClickListener(new View.OnClickListener() {  
    @Override  
    public void onClick(View v) {  
        String web = myEditTextUri.getText().toString();  
        if(web.equals(""))  
            return;  
        String http = web.substring(0, 4);  
        if(!http.equals("http"))  
            web = "https://" + web;  
        Intent intent = new Intent(getActivity(), WebViewPage.class);  
        intent.putExtra("uri", web);  
        startActivity(intent);  
    }  
});  
}
```

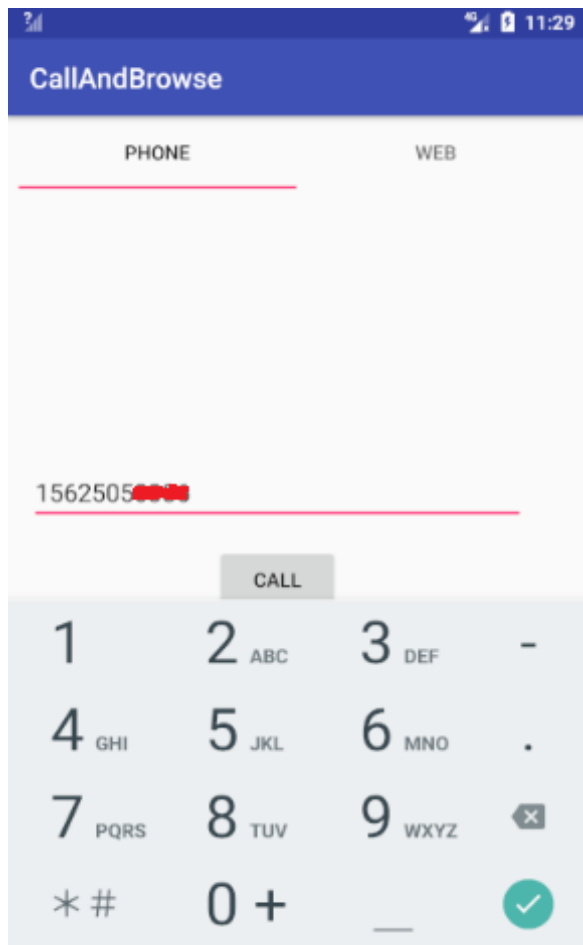
在本 App 中，我用 WebView 实现在本 App 内浏览网页，为啥要这么做呢？因为小组大作业里有类似的功能要实现，现在先练练手。

- 1) 一开始先判断用户输入的网址是否以 http 开头，WebView 浏览的网址需要以 https 或 http 开头。
- 2) 新建一个 Intent 跳转到另一个界面，里面只放了一个 WebView 控件。
- 3) 传数据，要把用户输入的网址传过去。

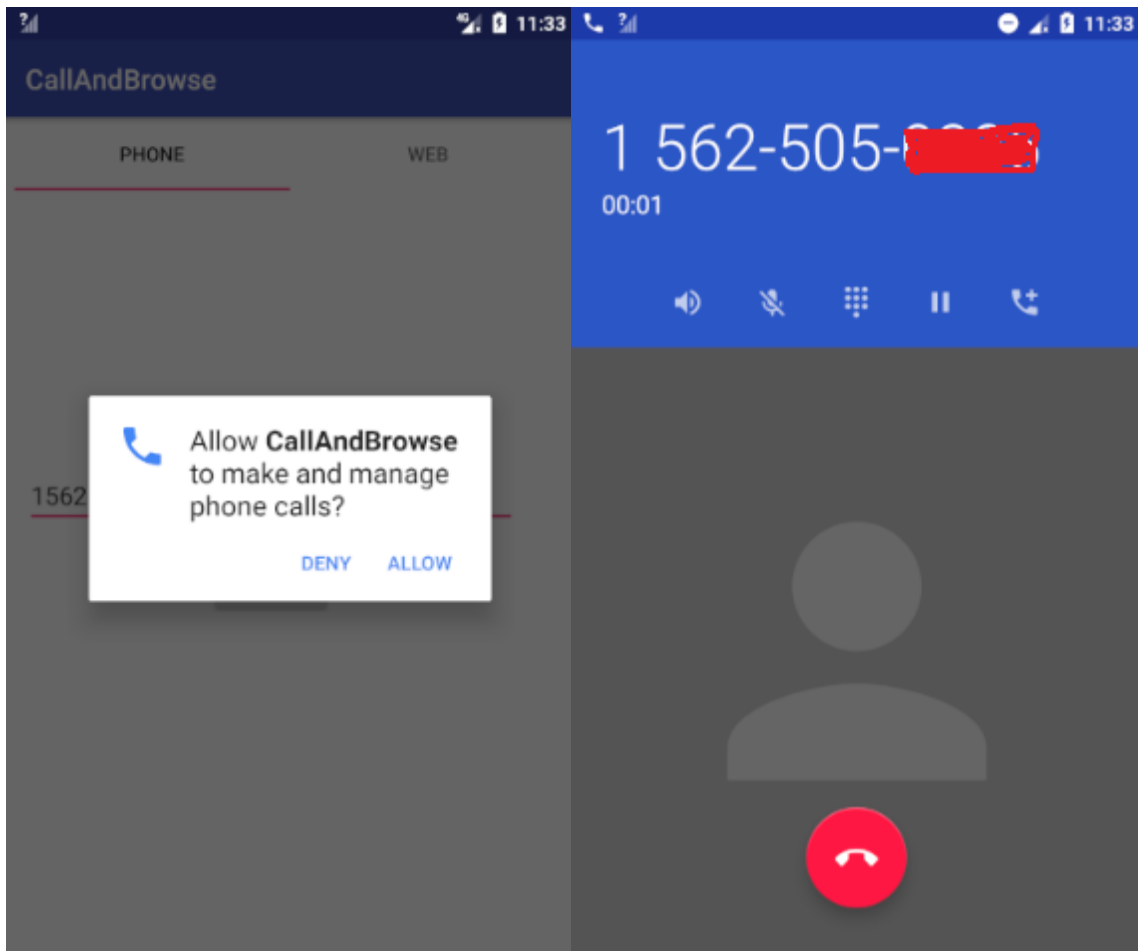
```
Intent intent = getIntent();  
Bundle bundle = intent.getExtras();  
String uri = bundle.getString("uri");  
Toast toast = Toast.makeText(this, uri, Toast.LENGTH_LONG);  
toast.show();  
webView.setWebViewClient(new WebViewClient() {  
    @Override public boolean shouldOverrideUrlLoading(WebView view, String url) {  
        webView.loadUrl(url);  
        return true;  
    }  
});  
webView.loadUrl(uri);
```

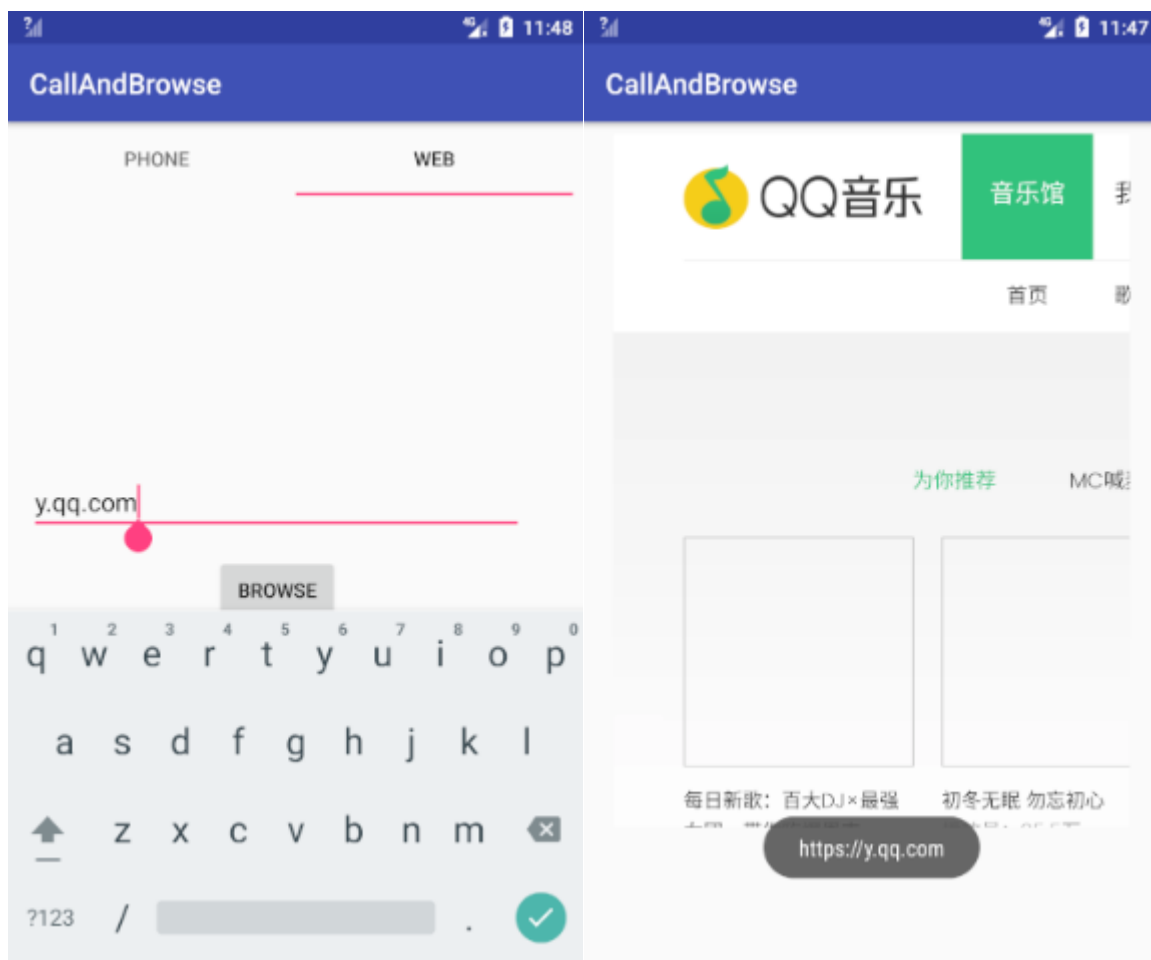
- 1) 获得传来的数据。
- 2) 设置 WebViewClient, 重载 shouldOverrideUrlLoading 函数, 这样做的目的是使得网页显示在 WebView 里, 而不是调用手机的浏览器显示。
- 3) 加载 uri。

5 软件操作流程



由于在 xml 文件里设置了 `inputType`，所以出来的小键盘是数字键盘。









国际化设计展示

6 难点和解决方案

6.1 危险权限的获取

危险权限不仅要在 Manifest 里写入，还要在程序执行的时候，也要判断 App 是否已经获取。

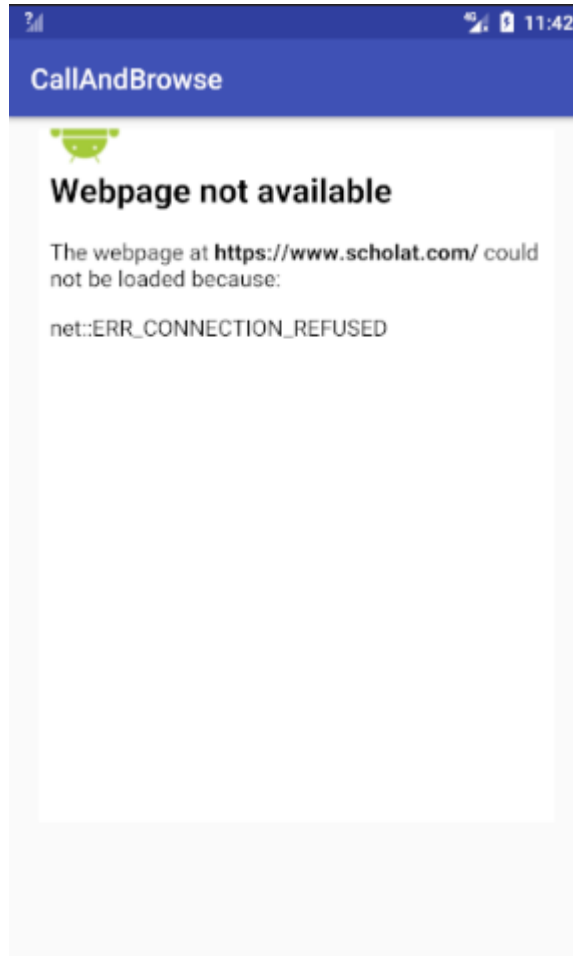
6.2 WebView 的使用

WebViewClient 是用于使得网页显示在 WebView 中，而不是调用本地浏览器，其中有很多函数需要重载。本 App 只是简单地加载，还有很多函数是控制 JS 的加载，控制图片的加载，优先加载，按键处理等等。

7 不足之处和今后设想

7.1 有的网页不以 https 开头

本 App 中判断如果前缀没有 http，自动加上 https，但是有的网站不用 https 协议，比如学者网，如果输入 <http://www.scholat.com> 能够浏览，但是如果输入 www.scholat.com，那么 App 自动加上 https，则无法浏览。



7.2 WebViewClient 的完善

由于是通过 WebViewClient 来浏览网页，所以效果肯定没有浏览器来好，有的图片加载不出来，因为 WebViewClient 还有很多函数可以重载，用来处理图片加载、JS 加载等等。本 App 中都没有实现这些函数，只是简单地加载网页。