**通用4.0开发规范**

1. **开发工具：Android Studio**
2. **项目名称：UctCommon4.0**
3. **项目模块：UctCommonApp4.0**
4. **应用名称：com.ptyt.uct.common.UctApplication**
5. **开发模式：MVC**
6. **项目包名：com.xigu.uct.ui(动态获取，不要直接使用该包名)**
7. **编码格式：UTF-8**
8. **SQLite数据库框架：传统**
9. **本地数据存储：** 
   * **xml存储：**

设置：fragment+header?

* + **sdcard存储：**

照片，录像

1. **SharedPreference键命名：**

格式：common\_ + 模块\_ + 数字， 键命名：KEY\_<FIELD>

1. **注释：**
   * **类：**

/\*\*  
 \* Description: 基础Activity类，所有的Activity需要继承该类，包含一些

\* 抽象公有方法，和Activity共用的方法  
 \* Date: 2017/4/24  
 \* Author: ShaFei  
 \* Version: V1.0  
 \*/

* + **方法：**

/\*\*

\* Description: 登录逻辑处理

\* param: String username

\* String password  
 \* return： true 登录成功

\* false 登录失败  
 \*/

* + **修改Bug：**

修改开始：/\* ptyt begin, 修改了登录传参个数的问题\_shafei\_20170424 \*/

修改结束：/\* ptyt end \*/

1. **编码规范：**
   * **包（packages）：**

采用反域名命名规则，全部使用小写字母。一级包名为com，二级包名为xx（可以是公司或则个人的随便），三级包名根据应用进行命名，四级包名为模块名或层级名

|  |  |
| --- | --- |
| **包名** | **描述** |
| com.ptyt.uct.activity | 用来组织Activity类 |
| com.ptyt.uct.fragment | 用来组织fragment类 |
| com.ptyt.uct.dialog | 自定义dialog |
| com.ptyt.uct.preference | 自定义首选项 |
| com.ptyt.uct.widget | 自定义小控件 |
| com.ptyt.uct.adapter | 项目中用到的适配器类 |
| com.ptyt.uct.utils | 工具类，如HttpUtil，ImageUtil，FileUtil，常量类 |
| com.ptyt.uct.service | 服务类，如GetMsgService |
| com.ptyt.uct.ptytinterface | 自定义接口 |
| com.ptyt.uct.common | 调了一些UctLib API，和一些Model管理的类 |
| com.ptyt.uct.exception | 自定义的异常 |
| com.ptyt.uct.dbhelper | 数据库类，如DataBaseHelper，MessageDB |

* + **类（classes）**采用大驼峰命名法，尽量避免缩写，除非该缩写是众所周知的，比如HTML,URL，如果类名称中包含单词缩写，则单词缩写的每个字母均应大写。

|  |  |  |
| --- | --- | --- |
| **类** | **描述** | **例如** |
| activity 类 | Activity为后缀标识 | 欢迎页面类WelcomeActivity |
| Adapter类 | Adp或者Adapte为后缀标识 | 新闻详情适配器NewtDetailAdp或则直接NewDetailAdapter |
| 公共方法类 | Tools或Manager为后缀标识 | 线程池管理类：ThreadPoolManager日志工具类：LogTools |
| 数据库类 | 以DBHelper后缀标识 | 新闻数据库：NewDBHelper |
| Service类 | 以Service为后缀标识 | 时间服务TimeService |
| BroadcastReceive类 | 以Broadcast为后缀标识 | 时间通知TimeBroadcast |
| ContentProvider | 以Provider为后缀标识 |  |
| 直接写的共享基础类 | 以Base开头 | BaseActivity,  BaseFragment |

* + **方法（methods）：**

采用小驼峰命名法例如:onCreate(),run()。

|  |  |
| --- | --- |
| **方法** | **说明** |
| initXX() | 初始化相关方法,使用init为前缀标识，如初始化布局initView() |
| isXX() | checkXX()方法返回值为boolean型的请使用is或check为前缀标识 |
| getXX() | 返回某个值的方法，使用get为前缀标识 |
| processXX() | 对数据进行处理的方法，尽量使用process为前缀标识 |
| displayXX() | 弹出提示框和提示信息，使用display为前缀标识 |
| saveXX() | 与保存数据相关的，使用save为前缀标识 |
| resetXX() | 对数据重组的，使用reset前缀标识 |
| clearXX()、removeXXX() | 清除数据相关的 |
| drawXXX() | 绘制数据或效果相关的，使用draw前缀标识 |

* + **变量（variables）**

采用小驼峰命名法。类中控件名称必须与xml布局id保持一致。

用统一的量词通过在结尾处放置一个量词，就可创建更加统一的变量，它们更容易理解，也更容易搜索。例如，请使用strCustomerFirst和strCustomerLast，而不要使用strFirstCustomer和strLastCustomer。

量词列表：量词后缀说明

First 一组变量中的第一个

Last 一组变量中的最后一个

Next 一组变量中的下一个变量

Prev 一组变量中的上一个

Cur 一组变量中的当前变量

* + **常量（Constants）**

全部大写,采用下划线命名法.例如：MIN\_WIDTH

* + **资源布局文件（XML文件（layout布局文件））：**

全部小写，采用下划线命名法

1)Activity默认布局，以去掉后缀的Activity类进行命名。不加后缀：

功能模块.xml

例如：main.xml、more.xml、settings.xml

或则：activity\_功能模块.xml

例如：activity\_main.xml、activity\_more.xml

2)Dialog命名：dialog\_描述.xml

例如：dlg\_hint.xml

3)PopupWindow命名：ppw\_描述.xml

例如：ppw \_info.xml

4)列表项命名listitem\_描述.xml

例如：listitem\_city.xml

5)包含项：include\_模块.xml

例如：include\_head.xml、include\_bottom.xml

6)adapter的子布局：功能模块\_item.xml

例如：main\_item.xml

* + **资源ID（resourcesid）：**

大小写规范与方法名一致，采用小驼峰命名法。

命名规范为“资源控件的缩写名”+“变量名”。注意：页面控件名称应该和控件id名保持一致。

strings.xml，colors.xml等中的id命名：

命名模式：

activity名称\_功能模块名称\_逻辑名称

或者

activity名称\_逻辑名称

或者

common\_逻辑名称

strings.xml中，使用activity名称注释，将文件内容区分开来。

* + **资源文件（图片drawable文件夹下）：**

全部小写，采用下划线命名法，加前缀区分。

命名模式：activity名称\_逻辑名称

或者

common\_逻辑名称

如果有多种形态如按钮等除外如btn\_xx.xml（selector）

|  |  |
| --- | --- |
| **名称** | **功能** |
| btn\_xx | 按钮图片使用btn\_整体效果（selector） |
| btn\_xx\_normal | 按钮图片使用btn\_正常情况效果 |
| btn\_xx\_press | 按钮图片使用btn\_点击时候效果 |
| bg\_head | 背景图片使用bg\_功能\_说明 |
| def\_search\_cell | 默认图片使用def\_功能\_说明 |
| icon\_more\_help | 图标图片使用icon\_功能\_说明 |

* + **layout中的id命名**

命名模式为：view缩写\_模块名称\_view的逻辑名称

view的缩写详情如下：

|  |  |
| --- | --- |
| 组件名称 | 简写 |
| Button | Btn(btn) |
| RadioButton | Rbtn(rbtn) |
| ImageButton | Ibtn(ibtn) |
| TextView | Tv(tv) |
| ImageView | Iv(iv) |
| ListView | Lv(lv) |
| ProgressBar | Pbar(pbar) |
| EditText | Edtv(et) |
| CheckBox | Chk(chk) |
| RelativeLayout | Rl(rl) |
| LinearLayout | Ll(ll) |
| FrameLayout | Fl(fl) |

1. **Git 路径：**

git@192.168.4.200:android/android\_app/UctCommonApp.git

**14. Bug修改规范，范例（解决何种问题/如何解决\_Bug ID\_修改者\_修改时间）：**

*/\* ptyt begin, 解决通讯录下拉刷新下载失败时一直转圈\_4018\_shafei\_20170906 \*/***mContext**.runOnUiThread(**new** Runnable() {  
 @Override  
 **public void** run() {  
 **if** (**contact\_srl**.isRefreshing()) {  
 **contact\_srl**.setRefreshing(**false**);  
 }  
 }  
});  
*/\* ptyt end \*/*