**《软件编码规范说明书》**

团队名称： 软件工程四人小团队

指导教师： 代祖华

完成时间： 2022年6月19日

团队成员: 刘温元 闫雨馨

刘转弟 李瑞婷

目录

[一.引言 3](#_Toc11308681)

[1.1编写目的 3](#_Toc11308682)

[1.2 背景 3](#_Toc11308683)

[二.页面代码书写规范 4](#_Toc11308684)

[2.1规范目的： 4](#_Toc11308685)

[2.2通用规范： 4](#_Toc11308686)

[2.3命名规范： 4](#_Toc11308687)

[三.代码书写规范 8](#_Toc11308688)

[3.1 术语说明 8](#_Toc11308689)

[3.2 文件名 8](#_Toc11308690)

[3.3文件编码： UTF-8 8](#_Toc11308691)

[3.4 特殊字符 8](#_Toc11308692)

[3.4.1 空格字符 8](#_Toc11308693)

[3.4.2 特殊转义字符串 8](#_Toc11308694)

[3.5 源 码文件结构 8](#_Toc11308695)

[3.6 格 式 8](#_Toc11308696)

3.7 注释......................................................................................................... ............................ 8

# 一.引言

1.1编写目的

在前一阶段的《软件系统详细设计说明书》中，已解决了实现该系统的程序模块设计问题，包括系统的模块化、模块之间传递的信息，以及数据库、数据结构的设计等。此文档主要是描述软件架构和编码的技术规范。详细地说明所确定的整个框架所涉及到的技术的外部行为，以及表现形式。另外，涉及到具体编写代码时，类、方法的命名规范。

1.2 背景

A.待开发的软件系统的名称：；

B.本项目的任务提出者：；

C.开发者：；

D.用户：所有手机使用者；

# 页面代码书写规范

# 2.1规范目的：

统一的编码规范可以使程序可读性增强，使程序更易于被理解，并降低维护成本。

2.2通用规范：

1. 尽量使用完整的英文描述符
2. 采用适用于相关领域的术语
3. 采用大小写混合使名字可读
4. 尽量少用缩写，但如果用了，要明智地使用，且在整个工程中统一
5. 避免使用长的名字（小于 15 个字母是个好主意）
6. 避免使用类似的名字，或者仅仅是大小写不同的名字
7. 避免使用下划线（除静态常量等）
8. 注释应该增加代码的清晰度
9. 保持注释的简洁

2.3命名规范：

命名原则:

1. 使用完整的英文名进行命名，慎用字母缩写，如果使用，请按照缩写规范用

例: No代表number、ID代表identification

2. 避免命名过长(20个字符内比较好)

3.避免相似的命名

例: shopCar 和shopCARAnjoyo命名规范:

1. 包(packages) :采用反域名命名规范，全部使用小写字母。一级包名为com; 二级包名anjoyo,三级包名根据应用进行命名，四级包名根据类的具体功能类命名

|  |  |
| --- | --- |
| 包名 | 说明 |
| com. anjoyo.应用. activity | 此包中包含:页面用到的Activity类。 |
| com. anjoyo.应用. handler | 此包中包含:解析服务器端返回信息的Handle类 |
| com. anjoyo.应用.bean | 此包中包含:程序中要用到的JavaBean类 |
| com. anjoyo.应用. adapter | 此包中包含:进行页面的数据适配的Adapter类 |
| com. anjoyo.应用. info | 此包中包含:程序中要用到的非JavaBean的其它信息类，  例如: Constants. java |
| com. anjoyo.应用. tools | 此包中包含:公共工具方法类 |
| com. anjoyo.应用. db | 此包中包含:数据库操作类 |
| com. anjoyo.应用. custom | 此包中包含:自定义的view等类 |
| com. anjoyo.应用。interfaces | 此包中包含:所有接口 |
| com. anjoyo.应用. service | 此包中包含: Service 服务 |
| com. anjoyo.应用. broadcast | 此包中包含: Broadcast服务 |

2.类(classes): 名司，尽量避免缩写。首字母大写，非首字母全部小写。类名中包括多个单词时，采用驼峰标识方式。

|  |  |  |
| --- | --- | --- |
| 类名 | 规范写法 | 备注 |
| Activity类 | Aty后缀标识。欢迎页面类: WelcomeAty |  |
| Adapter类 | Adp后缀标识。商品详情适配器类:ProductDetailAdp |  |
| Bean类 | Bean后缀标识。安卓越Bean命名:AnjoyoPopBean |  |
| Constants类 | Constants后缀标识。 |  |
| 解析类 | HIr后缀标识。首页解析类: HomePosterHlr |  |
| 公共方法类 | Tools或Manager后缀标识。  线程池管理类: ThreadPoolManager  日志工具类: LogTools |  |
| 数据库类 | DBHelper后缀标识。  城市数据库: CityDBHelper |  |
| 自定义View类 | 以功能名字成结尾: CustomListView |  |
| Service类 | Service后缀标识 |  |
| BroadcastReceive类 | Broadcast后缀标识 |  |

1. 接口(interface ): 命名规则与类的命名规则-致。able或ible结尾。  
   4.方法(methods):动词或动名词。第一个单词首字母小写。方法名中包括多个单词时，第一个单词之后的所有单词采用驼峰标识方式。  
   例如: onCreate（）  
   run（）

|  |  |  |
| --- | --- | --- |
| 方法名 | 规范写法 | 备注 |
| initXX () | 初始化相关方法使用init前缀标识 |  |
| adapterXXX0 | 适配相关使用adapter 前缀标识 |  |
| isXXX( checkXXX 0 | 方法返回值为boolean型的请使用is 或check前缀标识 |  |
| getXXXO | 返回值为基本类型或自定义类型的，使用get前缀标识 |  |
| processXXO | 对数据进行处理的，尽量使用process 前缀标识 |  |
| resetXXXO | 对数据重组的，使用reset前缀标识 |  |
| displayXXXO | 弹出提示框和提示信息，使用display 前缀标识 |  |
| saveXXX () | 与保存数据相关的，使用save前缀标识 |  |
| clearXXXOremoveXXX0 | 清除数据相关的，使用clear 或remove前缀标识. |  |
| drawXXX 0 | 绘制数据或效果相关的，使用draw前缀标识 |  |

5.变量(variables)大小写规范与方法名一致。类中控件名称必须与xml布局id保持一致。

例如: filePath、 fileName、textTitle、editUsername

6.常量(constants)全部大写，单词之间用下划线隔开。

例如: MIN WIDTH

7. XML 文件(布局文件):统一用小写加下划线组合

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

2. Activity 子布局，父布局名称开始，追加item和子布局功能说明。

例如: Activity 默认布局: homeposter. xml

子布局为homeposter\_ item poster. xm1

8.资源文件(图片):统一用小写加下划线组合，加前缀区分。  
规范写法：

btn\_ 1ogin\_ normal：按钮图片使用btn. 功能\_说明

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

seg\_ list\_ line：具有分隔特征的图片使用seg.功能说明

sel\_ ok：选择图标使用sel. \_功能说明

9.动画文件（anim包）：统一小写加下划线组合，加前缀区分。

规范写法：

click\_ head\_ left：点击背景切换动画使用click前缀标识

bg\_ shape\_ rectangle：背景自定义图形使用bg前缀标识

show\_ shopcar\_ add：小动画效果使用show前缀标识

10.资源你ID（resourses id）：大小写规范与方法名一直。命名规范为“资源控制的缩写名”+“变量名”。

规范写法：

ImageView：imgXXX

TextView：textXXX

EditText：edi tXXX

Button：btnXXX

LinearLayout：linXXX

RelativeLayout：relXXX

FrameLayout：fraXXX

ScrollView：scrXXX

ListView：listXXX

GridView：gri dXXX

Gallery：ga1XXX

Spinner：spinnerXXX

11.项目编码必须为“utf-8”。

# 三.代码书写规范

3.1 术语说明

本文档除非特殊说明，否则：

class（类）统指普通的 class 类型、 enum 枚举类型、 interface 类型和 annotation 类型。

comment （注释）总是指 implementation comments （实现注释， /\* \*/ ）。我们不使用“文档注释”这样的说法，而会直接说 javadoc。

3.2 文件名

源码文件名由它所包含的顶级 class 的类名（大小写敏感），加上 .java 后缀组成。

3.3文件编码： UTF-8  
源码文件使用 UTF-8 编码。

3.4 特殊字符

### 3.4.1 空格字符

除了换行符外， ASCII 水平空白字符（ 0x20）是源码文件中唯一支持的空格字符。这意味着：

其他空白字符将被转义。

Tab字符不被用作缩进控制。

### 3.4.2 特殊转义字符串

任何需要转义字符串表示的字符（例如 等），采用这种转义字符串的方式

表示，而不采用对应字符的八进制数（例如 \012）或 Unicode 码（例如 \u000a）表示。

3.5 源 码文件结构

**3.5.1 import 语句**  
字段应该定义在文件开头，或者紧挨着使用这些字段的方法之前。  
**3.5.2 不使用通配符 import**  
不应该使用通配符 import ，不管是否是静态导入。

3.6 格 式  
术语说明：块状结构（ block-like construct ）指类、成员函数和构造函数的实现部分（花括  
号中间部分） 。注意数组初始化，所有的数组初始化都可以被认为是一个块状结构（非强制）。  
**3.6.1 花括号**

花括号在需要的地方使用。  
花括号一般用在 if, else, for, do, 和 while 等语句。 甚至当它的实现为空或者只有一句话时，也需要使用。

**3.7 注释**

**3.7.1文件注释内容**

版权说明、描述信息、生成日期、修改记录、作者、生成日期，格式如下：

/\*\*

Copyright © 2015ktc. All rights reserved.

@Title: LogcatUtil.java

@Description: 调试信息工具文件

@author: shan.zhang

@date: 2015-6-4

@modifier:

@version: V1.0

\*/

**3.7.2类和接口的注释**

类功能描述、作者、生成日期

/\*\*

@ClassName: LogcatUtil

@Description: 用于程序中调试信息输出的工具类

@author: shan.zhang

@date: 2015-6-4

\*/

**3.7.3方法注释**

功能描述、输入参数、返回值

/\*\*

\*

@Title: getInstance

@Description: 调试工具类实例获取方法

@return: LogcatUtil

\*/

public static LogcatUtil getInstance() {

if(null == mLogcatUtil) {

mLogcatUtil = new LogcatUtil();

}

return mLogcatUtil;

}

3.4成员变量、常量注释

对应的功能定义

/\*\*

是否输出打印信息开关

true：输出打印信息

false：不输出打印信息

/

private final static boolean DEBUG\_ENABLE = false;

/\*

默认输出log信息的Tag标识

\*/

private final static String DEBUG\_TAG = “k\_debug”;