开发文档及规范

修订历史记录：

|  |  |  |  |
| --- | --- | --- | --- |
| 日期 | 版本 | 说明 | 作者 |
| 2013/9/13 | 1.0 | 客户标签开发文档及规范 | 杨卫争 |
|  |  |  |  |
|  |  |  |  |

# 项目目的

客户标签系统是为移动营销提供服务的客户筛选系统，为营销活动提供指导的作用！

# 项目背景

业务运营时代的来临，市场竞争形势的日趋严峻，省公司运营策略的调整，对市场运营提出了更高要求.

我们的思考: 如何更好迎接全业务时代的来临，实现公司的精细化运营？

我们的尝试：通过各种方式分析、收集客户的行为特征，为每个客户打上相应的分类标签，针对不同类型的客户开展精细化营销。

# 数据字典

描述： 整理项目中用到的词语及相应的英语

样例：

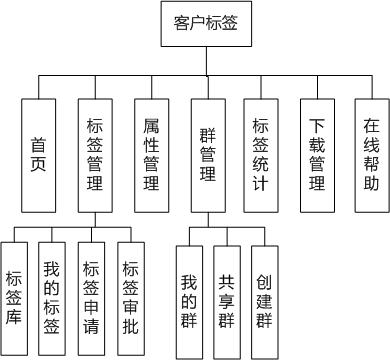
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 名称 | 英语名称 | 描述 | 添加人/修改人 | 添加时间 |
| 客户标签 | CustermTag | 客户标签是对各类客户属性、特征、行为的认知结果，反映客户的需求和偏好，主要包括客户身份标签、内容偏好标签、流量标签、终端信息标签、基础信息标签等。 | 杨卫争 | 2013-09-04 |
| 标签 | Tag |  | 杨卫争 | 2013-09-04 |
| 属性 | Attr |  | 杨卫争 | 2013-09-04 |
| 管理 | Mage |  | 杨卫争 | 2013-09-04 |
| 下载 | DownLoad |  | 杨卫争 | 2013-09-04 |
| 群 | Group |  | 杨卫争 | 2013-09-04 |
| 排行 | Order |  | 杨卫争 | 2013-09-04 |
| 城市 | City |  | 杨卫争 | 2013-09-04 |
| 区县 | County |  | 杨卫争 | 2013-09-04 |
| 申请 | ApplyFor |  | 杨卫争 | 2013-09-04 |
| 审批 | Approve |  | 杨卫争 | 2013-09-04 |
| 我的标签 | MyTag |  | 杨卫争 | 2013-09-04 |
| 在线帮助 | OnlineHelp |  | 杨卫争 | 2013-09-04 |
| 月 | Month |  | 杨卫争 | 2013-09-04 |
| 日 | Date |  | 杨卫争 | 2013-09-04 |
| 创建 | Create |  | 杨卫争 | 2013-09-04 |
| 标签库 | TagLib |  | 杨卫争 | 2013-09-04 |
| 共享 | Share |  | 杨卫争 | 2013-09-04 |
| 应用 | Apply |  | 杨卫争 | 2013-09-04 |
| 访问 | Access |  | 杨卫争 | 2013-09-04 |
| 构成条件 | FormCondition |  | 杨卫争 | 2013-09-04 |
| 统计 | Statis |  | 杨卫争 | 2013-09-04 |
| 添加 | Add |  | 杨卫争 | 2013-09-04 |
| **数据库相关操作 动词+名词 例如：修改标签 updateTag** | | | | |
| 修改 | Update |  | 杨卫争 | 2013-09-04 |
| 删除 | Del |  | 杨卫争 | 2013-09-04 |
| 获得 | Get |  | 杨卫争 | 2013-09-04 |
| 设置 | Set |  | 杨卫争 | 2013-09-04 |
| 装载 | Load |  | 杨卫争 | 2013-09-04 |
| 查找 | Find |  | 杨卫争 | 2013-09-13 |
| 添加 | Add |  | 杨卫争 | 2013-09-13 |
| 列表 | List |  | 杨卫争 | 2013-09-13 |

# 四、任务概述

1.目标

阐述系统要实现的功能模块及各模块之间的关系，例如：

客户标签系统模块关系图



2.项目运行环境

操作系统：

运行环境：weblogic10

数据库：DB2

3.数据描述

列举相应的模块中设计到的表结构

表名：TB\_CUSTOMGROUP\_INFO 说明：客户群信息表 创建人：杨卫争 日期：2013/9/13

|  |  |  |  |
| --- | --- | --- | --- |
| 列名 | 中文 | 数据类型 | 备注 |
| TAG\_ID |  | VARCHAR(32) | 客户群标识 |
| TAG\_NAME |  | VARCHAR(64) | 客户群名称 |
| TAG\_TYPE |  | VARCHAR(32) | 客户群类型 |
| TAG\_SERV\_TYPE |  | VARCHAR(32) | 客户群业务类型 |
| TAG\_CLASS |  | VARCHAR(32) | 客户群小类 |
| CREATE\_TIME |  | VARCHAR(24) | 客户群创建时间 |
| END\_TIME |  | VARCHAR(24) | 客户群结束时间 |
| PROFILE |  | VARCHAR(1000) | 客户群简介 |
| COUNT\_SUBS |  | INTEGER | 客户群统计人数 |
| TAG\_STATEMENT |  | VARCHAR(1000) | 客户群口径 |
| TAG\_CREATOR |  | VARCHAR(32) | 客户群创建人 |
| TAG\_CREATOR\_ID |  | VARCHAR(32) | 客户群创建人ID |
| TAG\_REGION |  | VARCHAR(32) | 客户群地市 |
| IS\_SHARE |  | VARCHAR(4) | 是否共享：1是，0否 |
| TAG\_STATUS |  | VARCHAR(4) | 客户群状态：0 客户清单下载中 1 已下载 |
| TAG\_TEC\_STAMT |  | VARCHAR(1000) | 客户群技术口径（SQL） |
| CUSTLIST\_PATH |  | VARCHAR(1000) | 目标客户清单存储路径 |
| TYPE\_FLAG |  | INTEGER | 客户群标识 |
| TAG\_TEC\_STAMT\_ADVANCE |  | VARCHAR(500) | 客户群名称 |

表名：TB\_CUSTOMTAG\_REMARK 说明：客户标签备注表 创建人：杨卫争 日期：2013/9/13

|  |  |  |  |
| --- | --- | --- | --- |
| 列名 | 中文 | 数据类型 | 备注 |
| TAG\_ID |  | VARCHAR(32) | 客户群标识 |
| TAG\_PRIORITY |  | Integer | 客户标签优先级 |
| TAG\_REMARK |  | VARCHAR(32) | 客户标签备注 |
| TAG\_VERSION |  | Integer | 客户标签版本号 |

# 程序模块说明

1. 包说明

包的命名规则，例如：客户标签系统

com.miapsoft.customtag.dao 数据库层

|  |  |  |  |
| --- | --- | --- | --- |
| 包名 | 作用/范围 | 创建人 | 创建/修改日期 |
|  |  |  |  |
|  |  |  |  |

1. 类说明

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 类名 | 所在包 | 作用/范围 | 创建人 | 创建/修改日期 |
| IBusClassTag.java | Dao.Idao | 标签库业务类，业务类二级标签Dao层接口 | 杨卫争 | 2013-09-13 |
| BusClassTag.java | Module | 业务类二级标签JavaBean | 杨卫争 | 2013/9/13 |
| BusClassTagImpl.java | Dao | 业务二级标签实现类 | 杨卫争 | 2013/9/13 |
| ICustomTagNew.java | Dao.Idao | 新建标签接口 | 杨卫争 | 2013/9/13 |
| CustomTagNewImpl.java | Dao | 新建标签实现类 | 杨卫争 | 2013/9/13 |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

1. 方法说明

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 方法名 | 所属类名 | 作用/范围 | 创建人 | 创建修改日期 |
|  |  |  |  |  |
|  |  |  |  |  |

# 附录：

# 注释

1. 类、接口注释

在类、接口定义之前当对其进行注释，包括类、接口的目的、作用、功能、继承于何种父类，实现的接口、实现的算法、使用方法、示例程序等，在作者和版本域中使用CVS标记自动跟踪版本变化等，具体参看附件《注释模板》中相关部分。

/\*\*

\*<p>字符串实用类。</p>\*

\*定义字符串操作时所需要用到的方法，如转换中文、HTML标记处理等。\*

\***@author** $Author:l\_walker$

\***@version** $Revision:1.2$ $Date:2003/05/1502:10:27$

\*/

1. 方法注释

依据标准

JavaDoc

规范对方法进行注释，以明确该方法功能、作用、各参数含义以及返回值

等。复杂的算法用/\*\*/在方法内注解出。

参数注释时当注明其取值范围等

返回值当注释出失败、错误、异常时的返回情况。

异常当注释出什么情况、什么时候、什么条件下会引发什么样的异常

/\*\*

\* 作用：执行查询。 \*

\* 描述：该方法调用Statement的executeQuery(sql)方法并返回ResultSet 结果集。

\* **@param** sql 标准的SQL语句

\* **@return** ResultSet结果集，若查询失败则返回null \* **@throws** SQLException 当查询数据库时可能引发此异常 \*/

public ResultSet executeQuery**(**String sql**)** **throws** SQLException **{**

//Statement和SQL语句都不能为空

**if(null** **!=** stmt **&&** **!**StringUtil**.**isEmpty**(**sql**)){** //返回查询执行结果

**return** stmt**.**executeQuery**(**sql**);**

**}**

**return** **null;**

**}**//end executeQuery()

1. 其他注释

应对重要的变量加以注释，以说明其含义等。

应对不易理解的分支条件表达式加注释。不易理解的循环，应说明出口条件。过长的方法实现，应将其语句按实现的功能分段加以概括性说明。

对于异常处理当注明正常情况及异常情况或者条件，并说明当异常发生时程序当如何处理。

# 命名规则

5.1 基本原则

规范的命名能使程序更易阅读，从而更易于理解。它们也可以提供一些标识功能方面的信 息，有助于更好的理解代码和应用。

5.1.1  使用可以准确说明变量/字段/类/接口/包等的完整的英文描述符。例如，采用类

似 firstName，listAllUsers 或 CorporateCustomer 这样的名字，严禁使用汉语拼音及不 相关单词命名，虽然Java支持Unicode命名，但本规范规定对包、类、接口、方法、变 量、字段 等不得使用汉字等进行命名。

5.1.2  采用该领域的术语。如果用户称他们的“客户” (clients) 为“顾客”

(customers)，那么就采用术语 Customer 来命名这个类，而不用 Client。

5.1.3  采用大小写混合，提高名字的可读性。一般应该采用小写字母，但是类和接口的

名字的首字母，以及任何中间单词的首字母应该大写。包名全部小写。

5.1.4  尽量少用缩写，但如果一定要使用，当使用公共缩写和习惯缩写等，如实现

（implement）可缩写成impl，经理（manager）可缩写成mgr等，具体参看附录之《常 用缩写简表》，严禁滥用缩写。

5.1.5  避免使用长名字（最好不超过 25 个字母）。

5.1.6  避免使用相似或者仅在大小写上有区别的名字。

5.1.7  避免使用数字，但可用2代替to，用4代替for等，如：go2Jsp。

5.2 文件、包

5.2.1  文件名当与其类严格相同，所有单词首字母大写。

5.2.2  包名一般以项目或模块名命名，少用缩写和长名，一律小写。

5.2.3  基本包：org.skyinn，所有包、文件都从属于此包。

5.2.4  包名按如下规则组成：

                        [基本包].[项目名].[模块名].[子模块名]...                    如：

                        org.skyinn.quasar

                        org.skyinn.skyhome.dao.hibernate

5.2.5  不得将类直接定义在基本包下，所有项目中的类、接口等都当定义在各自的项目

和模块包中。

5.3 类、接口

所有单词首字母大写。使用能确切反应该类、接口含义、功能等的词。一般采用名词。 接口可 带I前缀或able、ible、er等后缀。

5.4 字段

4.4.1 常量

采用完整的英文大写单词，在词与词之间用下划线连接，如：DEFAULT\_VALUE

4.4.2 变量和参数

对不易清楚识别出该变量类型的变量应使用类型缩写作其前缀，如字符串使用 strXXX,boolean使用isXXX，hasXXX等等。除第一各个单词外其余单词首字母大写。

对私有实例变量可使用\_前缀，但在其存取方法中则应该将其前缀去掉。

4.4.3 组件/部件

应采用完整的英文描述符命名组件（接口部件），遵循匈牙利命名法则 如：btnOK， lblName。

4.4.4 集合

一个集合，例如数组和矢量，应采用复数命名来表示队列中存放的对象 类 型。命名应采用完整的英文描述符，名字中所有非开头的单词的第一个字母应大写，适当使 用集合缩写前缀。如：

     Vector vProducts = new Vector(); //产品向量

Array aryUsers = new Array();    //用户列表

4.4.5 神秘的数

我们在程序里经常会用到一些量，它是有特定的含义的。例如，现在我们写一个薪金统计 程序，公司员工有50人，我们在程序里就会用50这个数去进行各种各样的运算。在这里，50 就是"神秘的数"。当别的程序员在程序里看到50这个数，将很难知道它的含义，造成理解上的 困难。

在程序里出现"神秘的数"会降低程序的可读性、可维护性和可扩展性，故规定不得出现此 类"神秘的数"。避免的方法是把神秘的数定义为一个常量。注意这个常量的命名应该能表达该数 的意义，并且应该全部大写，以与对应于变量的标识符区别开来。例如上面50这个数，我们可 以定义为一个名为NUM\_OF\_EMPLOYEES的常量来代替。这样，别的程序员在读程序的时候就 可以容易理解了。

  4.4.6 其他

命名时应使用复数来表示它们代表多值。如：orderItems。

4.5 方法

方法的命名应采用完整的英文描述符，大小写混合使用：所有中间单词的第一个字母大写。方法名称的第一个单词常常采用一个有强烈动作色彩的动词。

取值类使用get前缀，设值类使用set前缀，判断类使用is(has)前缀。

例： getName()

setSarry()

isLogon()

        方法参数建议顺序：(被操作者，操作内容，操作标志，其他…)

    例：public void replace(String sourceStr,  //源字串

String oldStr,     //被替换字串

 String newStr){    //替换为字串

}

4.6 异常

异常类名由表示该异常类型的单词和Exception组成，如ActionException。

异常实例一般使用e、ex等，在多个异常时使用该异常名或简写加E，Ex等组成，如：

            SQLEx

 ActionEx

4.7 命名约定表

Table7 命名约定表：

|  |  |  |
| --- | --- | --- |
| 操作项 | 命名约定 | 示例 |
| 参数 | 使用传递值/对象的完整的英文描述符。 | userID |
| 字段/属性 | 字段采用完整的英文描述，第一个字母小写，任何中间单词的首字母大写。 | firstName |
| 布尔型的获取成员函数 | 所有的布尔型获取函数必须用单词 is（has）做前缀。 | isString()  hasMoney() |
| 类 | 采用完整的英文描述符，所有单词  的第一个字母大写。 | Customer |
| 编译单元文件 | 使用类或接口的名字，或者如果文件中除了主类之外还有多个类时，加上前缀java来说明它是一个源码文件。 | Customer.java |
| 部件/组件 | 使用完整的英文描述来说明组件的用途，将组件类型使用匈牙利命名法则作其前缀。 | btnOK,cboTypeList |
| 构造函数 | 使用类名 | Customer() |
| 异常类名 | 由表示该异常类型等的单词和  Exception组成 | SQLException ActionException |
| 异常实例名 | 通常采用字母 e、ex表示异常。多个异常时使用异常名或其简写加E、Ex等构成 | E  SQLEx |
| 静态常量字段（常量） | 全部采用大写字母，单词之间用下划线分隔。采用静态常量获取成员  函数。 | DEFAULT\_NAME |
| 获取成员函数 | 被访问字段名的前面加上前缀  get。 | getUserName() |
| 接口 | 采用完整的英文描述符说明接口封装，所有单词的第一个字母大写。使用I前缀，其后使用able, ible 或者 er等后缀，但这不是必  需的。. | IRunnable IPrompter |
| 局部变量 | 采用完整的英文描述符，第一个字  母小写，但不要隐藏已有字段。例  如，如果有一个字段叫firstName，不要让一个局部变量叫 firstName。 | strName  totalMoney |
| 循环计数器 | 通常采用字母 i，j，k 或者  counter，index | i,j,k,count,index |
| 包 | 采用完整的英文描述符，所有单词  都小写，多个单词以下划线相连。  所有包都属于org.skyinn | org.skyinn.quasar  org.skyinn.skyhome.dao |
| 成员函数 | 采用完整的英文描述说明成员函数功能，第一个单词尽可能采用一个生动的动词，除第一个单词外，每个单词第一个字母小写。 | openFile()  addUser() |
| 设置成员函数 | 被访问字段名的前面加上前缀  set。 | setName()  setPower() |