|  |  |  |
| --- | --- | --- |
| 文件状态：  [ ] 草稿  [√] 正式发布  []正在修改 | 文件标识： | SYZH1104-TYPT-05 |
| 当前版本： | V1.0 |
| 作者： | 邓虹玉 |
| 完成日期： | 2018/2/8 |

北京慧博科技有限公司

集客CRM项目

详细设计说明书

**北京慧博科技有限公司**

**2018年2月**

修订历史版本

|  |  |  |
| --- | --- | --- |
| **日期** | **修订内容** | **修订人** |
| 2018-02-08 | 创建文档版本。 | 邓虹玉 |
| 2018-02-08 | 审核文档 | 张亚情 |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

目录

第一章 引言 1

1.1 文档目的 1

1.2 预期读者 1

1.3 术语解释 1

1.4 参考资料 1

第二章 系统概述 2

2.1.1 集客CRM 2

2.1.2 积分系统 2

2.1.2.1 积分系统后台 2

2.1.2.2 积分系统客户端 2

第三章 系统数据结构设计 4

3.1 说明 4

3.2 应用程序表（ApplicationInfo） 4

3.3 应用订单表(ApplicationOrderInfo) 4

3.4 后台用户表（BackGroundUserInfo） 5

3.5 店铺信息(TaobaoShopInfo) 6

3.6 卖家用户信息表(SellerUserInfo) 7

3.7 店铺动态评分信息(TaobaoShopScoreInfo) 8

3.8 淘宝受权信息(TaobaoTOPInfo) 9

3.9 店铺签名可用（AppointedShopInfo） 10

3.10 店铺多签名表(MoreSignForShopNickInfo) 10

3.11 主、子店铺关联关系表(ShopRelevantToInfo) 11

3.12 流程任务表(FlowTaskInfo) 11

第四章 系统接口详细设计 13

4.1 工作方式 13

4.2 字符编码 13

4.3 调用示例 13

4.3.1 生成协议 13

4.3.1.1 返回 13

4.3.1.2 示例代码 14

4.3.2 查询协议 16

4.3.2.1 返回 16

4.3.2.2 示例代码 17

1. 引言
   1. 文档目的

本软件系统设计说明书是针对集客CRM系统分析与设计，整理而成的一份详细说明系统设计的文档。该文档将是指导开发人员具体工作的重要文档。

* 1. 预期读者
* 研发组、测试组、设计组等相关项目干系人；
* 集客CRM项目相关成员。
  1. 术语解释
* 商家：指使用集客CRM的用户。
* 客户：指在商家的电商平台进行过交易的买家。
  1. 参考资料
* 集客CRM系统建设方案；
* 集客CRM需求规格说明书；

1. 系统概述

集客CRM包含集客CRM和积分系统两部分。除支持原有系统功能外，还需进行功能的扩展，最重要的一部分功能为积分系统。

* + 1. 集客CRM

集客CRM是与淘宝平台对接的一个客户关系管理系统。淘宝平台将数据传送过来后，集客CRM负责把数据通过一些功能页面进行展示出来。支持订单全程跟进关怀，及对会员进行短信、邮件、优惠券等方式进行营销。并可将订单数据、商品数据、会员数据、物流数据从不同维度进行分析。

* + 1. 积分系统

积分系统是嵌入在集客CRM内的一个忠诚度管理系统，以积分为媒介，将会员存留引导二次复购。支持按会员等级设置积分。提供多种赚积分和花积分的互动方式，并可对会员积分进行管理及对会员积分的趋势、来源、获取、消费等进行分析。历史订单也可进行积分计算。

* + - 1. 积分系统后台

用户为淘宝店铺商家，简称商家。

商家从集客CRM中进入积分系统后台。

商家在积分系统中设置积分等级规则、生效规则、过期规则并开启。

商家设置订单赚积分规则，还可选择性设置评价赚积分、生日下单赚积分及互动玩法赚积分的规则。还需选择性设置积分兑换商品及花积分玩游戏。

商家将积分系统装修到淘宝平台，也可将独立的某系互动玩法装修到淘宝平台。

如后期想更换规则可重新设置保存。

* + - 1. 积分系统客户端

用户为淘宝平台的消费者，简称买家。

买家在淘宝平台下单并付款就可获得积分。买家也可参与互动活动来获得积分，也可花积分来参与互动活动，还可将积分兑换礼品。积分记录将在个人中心查看。

1. 系统数据结构设计
   1. 说明

集客CRM数据库字典分别为基础库、流程库、订单中心库、效果库、应用库1、应用库2六个数据库（不包括会员库、订单库、中间库）。以下简介基础库、流程库、订单中心库、效果库的表明和字段含义。

* 1. 应用程序表（ApplicationInfo）

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段 | 字段类型 | 默认值 | 是否为空 | 备注 |
| Id | BIGINT(8) | 0 | false | 应用ID |
| AppName | VARCHAR(20) |  | true | 应用名称 |
| Description | VARCHAR(100) |  | true | 应用描述 |
| appICON | VARCHAR(255) |  | true | 图标 |
| PaterId | BIGINT(8) | 0 | true | 归属业务ID |
| Uri | VARCHAR(100) |  | true | 界面地址 |
| OrderBy | BIGINT(4) | 0 | true | 排序字段 |
| UpdateTime | TIMESTAMP | ON UPDATE CURRENT\_TIMESTAMP | false | 最后更新日期 |

* 1. 应用订单表(ApplicationOrderInfo)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段名称 | 字段类型 | 默认值 | 是否为空 | 备注 |
| Id | BIGINT(8) | 0 | false | 应用订单ID |
| OrderTime | TIMESTAMP | ON UPDATE  CURRENT\_TIMESTAMP | false | 订购日期 |
| AppID | BIGINT(8) | 0 | true | 应用ID  关联ApplicationInfo表的Id |
| Deadline | TIMESTAMP | 0000-00-00 00:00:00 | false | 到期时间 |
| Status | BIGINT(4) | 0 | true | 状态（1可用） |
| Autosub | BIGINT(4) | 0 | true | 是否自动续约 |
| Expire\_notice | BIGINT(4) | 0 | true | 是否到期提醒 |
| UserID | BIGINT(8) | 0 | true | 用户Id  关联BackGroundUserInfo表的Id |
| Versions | VARCHAR(20) |  | true | 购买版本(基础班、高级版)  关联VersionsInfo表的Id |
| UpdateTime | TIMESTAMP | 0000-00-00 00:00:00 | false | 最后更新时间 |
| TaobaoID | BIGINT(32) | 0 | true | 淘宝用户 |

* 1. 后台用户表（BackGroundUserInfo）

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段名 | 字段类型 | 默认值 | 是否为空 | 备注 |
| Id | BIGINT(8) | 0 | false | 用户ID |
| Domain | VARCHAR(100) |  | true | 域名 |
| Name | VARCHAR(100) |  | true | 用户名 |
| Nickname | VARCHAR(100) |  | true | 名字昵称 |
| Password | VARCHAR(100) |  | true | 密码 |
| Number1 | VARCHAR(21) |  | true | 手机号1 |
| Number2 | VARCHAR(21) |  | true | 手机号2 |
| Number3 | VARCHAR(21) |  | true | 手机号3 |
| Number4 | VARCHAR(21) |  | true | 手机号4 |
| Number5 | VARCHAR(21) |  | true | 手机号5 |
| Telphone | VARCHAR(21) |  | true | 电话 |
| Email | VARCHAR(100) |  | true | 邮件 |
| CreateTime | TIMESTAMP | CURRENT\_TIMESTAMP | false | 创建时间 |
| Company | VARCHAR(50) |  | true | 公司名称 |
| Address | VARCHAR(200) |  | true | 公司地址 |
| Post | VARCHAR(10) |  | true | 公司邮编 |
| PaterId | BIGINT(8) | 0 | true | 父节点ID |
| RoleId | VARCHAR(50) |  | true | 角色id |
| YQZ\_RoleId | VARCHAR(50) |  | true | 云千载角色id |
| UsePeople | VARCHAR(50) |  | true | 使用人 |
| CreateUserId | BIGINT(8) | 0 | true | 创建用户 |
| UpdateTime | TIMESTAMP | 0000-00-00 00:00:00 | false | 最后更新时间 |

* 1. 店铺信息(TaobaoShopInfo)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段名 | 字段类型 | 默认值 | 是否为空 | 备注 |
| Id | BIGINT(8) | 0 | false | 主键ID |
| UserID | BIGINT(8) | 0 | true | 用户ID  关联BackGroundUserInfo表的Id |
| Sid | BIGINT(32) | 0 | true | 店铺编号,shop+sid.taobao.com即店铺地址，如shop123456.taobao.com |
| Cid | BIGINT(32) | 0 | true | 店铺所属的类目编号 |
| Nick | VARCHAR(1024) | 0 | true | 卖家昵称 |
| Title | VARCHAR(200) | 0 | true | 店铺标题 |
| Description | VARCHAR(4096) | 0 | true | 店铺描述 |
| Bulletin | VARCHAR(4096) | 0 | true | 店铺公告 |
| Pic\_path | VARCHAR(200) | 0 | true | 店标地址 |
| IsOpenAuthority | INT(1) | 0 | true | 店铺是否开启权限功能 |
| Created | TIMESTAMP | ON UPDATE CURRENT\_TIMESTAMP | false | 开店时间 |
| Modified | TIMESTAMP | 0000-00-00 00:00:00 | false | 最后修改时间 |
| Shop\_score\_id | BIGINT(8) | 0 | true | 店铺动态评分信息 |
| Status | BIGINT(4) | 0 | true | 授权状态 |
| UpdateTime | TIMESTAMP | 0000-00-00 00:00:00 | false | 最后更新时间 |

* 1. 卖家用户信息表(SellerUserInfo)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段名 | 字段类型 | 默认值 | 是否为空 | 备注 |
| Id | BIGINT(8) | 0 | false | ID主键 |
| UserId | BIGINT(8) |  | true | 用户数字ID |
| Nick | VARCHAR(50) |  | true | 店铺昵称  关联TaobaoShopInfo表的Nick |
| Sex | VARCHAR(2) |  | true | 性别 |
| Level | BIGINT(8) | 0 | true | 信用等级 |
| Score | BIGINT(8) | 0 | true | 信用总分 |
| TotalNum | BIGINT(8) | 0 | true | 收到的评价总条数 |
| GoodNum | BIGINT(8) | 0 | true | 收到的好评总条数 |
| type | VARCHAR(2) |  | true | 用户类型（B 天猫 C 淘宝） |
| HasMorePic | INT(1) | 0 | true | 是否购买多图服务 |
| ItemImgNum | BIGINT(8) | 0 | true | 可上传商品图片数量 |
| ItemImgSize | BIGINT(8) | 0 | true | 单张商品图片最大容量 |
| PropImgNum | BIGINT(8) | 0 | true | 可上传属性图片数量 |
| PropImgSize | BIGINT(8) | 0 | true | 单张销售属性图片最大容量 |
| AutoRepost | VARCHAR(2) |  | true | 是否受限制 |
| PromotedType | VARCHAR(2) |  | true | 有无实名认证 |
| Status | VARCHAR(2) |  | true | 状态 |
| AlipayBind | VARCHAR(2) |  | true | 有无绑定 |
| ConsumerProtection | INT(1) | 0 | true | 是否参加消保 |
| Avatar | VARCHAR(300) |  | true | 用户头像地址 |
| LiangPin | INT(1) | 0 | true | 是否是无名良品用户 |
| SignFoodSellerPromise | INT(1) | 0 | true | 卖家是否签署食品卖家承诺协议 |
| HasShop | INT(1) | 0 | true | 用户作为卖家是否开过店 |
| IsLightningConsignment | INT(1) | 0 | true | 是否24小时闪电发货(实物类) |
| HasSubStock | INT(1) | 0 | true | 表示用户是否具备修改商品减库存逻辑的权限 |
| IsGoldenSeller | INT(1) | 0 | true | 用户是否是金牌卖家 |
| VipInfo | VARCHAR(10) |  | true | 用户的全站vip信息 |
| MagazineSubscribe | INT(1) | 0 | true | 是否订阅了淘宝天下杂志 |
| VerticalMarket | VARCHAR(10) |  | true | 用户参与垂直市场类型 |
| OnlineGaming | INT(1) | 0 | true | 用户是否为网游用户 |
| UpdateTime | TIMESTAMP | ON UPDATE CURRENT\_TIMESTAMP | false | 最后更新时间 |

* 1. 店铺动态评分信息(TaobaoShopScoreInfo)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段名 | 字段类型 | 默认值 | 是否为空 | 备注 |
| Id | BIGINT(8) | 0 | false | 主键ID |
| Item\_score | VARCHAR(4096) |  | true | 商品描述评分 |
| Service\_score | VARCHAR(4096) |  | true | 服务态度评分 |
| Delivery\_score | VARCHAR(4096) |  | true | 发货速度评分 |
| UpdateTime | TIMESTAMP | ON UPDATE CURRENT\_TIMESTAMP | false | 最后更新时间 |

* 1. 淘宝受权信息(TaobaoTOPInfo)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段名 | 字段类型 | 默认值 | 是否为空 | 备注 |
| Id | BIGINT(8) | 0 | false | 主键类别ID |
| Ts | BIGINT(8) | 0 | true | 当前时间戳,单位：毫秒 |
| Iframe | VARCHAR(100) |  | true | 应用输出是否在IFRAME中 |
| Visitor\_Id | VARCHAR(50) |  | true | 当前用户ID |
| Visitor\_Nick | VARCHAR(100) |  | true | 店铺昵称  关联TaobaoShopInfo表的Nick |
| Visitor\_Role | VARCHAR(50) |  | true | 当前用户角色 |
| Expires\_In | BIGINT(8) | 0 | true | Session\_key 失效时间 |
| Refresh\_Token | VARCHAR(100) |  | true | 淘宝授权token |
| Access\_Token | VARCHAR(100) | 0 | true | Session Key |
| Token\_Type | VARCHAR(100) |  | true | Token类型 |
| Re\_Expires\_In | BIGINT(8) | 0 | true | refresh token 失效时间 |
| Sub\_Visitor\_Id | VARCHAR(100) |  | true | 子帐号ID |
| Sub\_Taobao\_User\_Nick | VARCHAR(100) |  | true | 子帐号nick |
| R1\_Expires\_In | BIGINT(8) | 0 | true | r1级别API |
| R2\_Expires\_In | BIGINT(8) | 0 | true | r2级别API |
| W1\_Expires\_In | BIGINT(8) | 0 | true | w1级别API |
| W2\_Expires\_In | BIGINT(8) | 0 | true | 授权到期时间 |
| UpdateTime | TIMESTAMP | CURRENT\_TIMESTAMP | false | 最后更新时间 |

* 1. 店铺签名可用（AppointedShopInfo）

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段名 | 字段类型 | 默认值 | 是否为空 | 备注 |
| Id | BIGINT(8) | 0 | false | ID主键 |
| ShopNick | VARCHAR(50) |  | true | 店铺昵称  关联TaobaoShopInfo表的Nick |
| AllowedToOmitSignPrefix | Int(1) | 0 | true | 允许忽略签名前缀（0 false 1 true）  店铺签名可用 |
| UpdateTime | TIMESTAMP | ON UPDATE CURRENT\_TIMESTAMP | false | 最后更新时间 |

* 1. 店铺多签名表(MoreSignForShopNickInfo)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段名 | 字段类型 | 默认值 | 是否为空 | 备注 |
| Id | BIGINT(8) | 0 | false | ID主键 |
| ShopNick | VARCHAR(100) |  | true | 店铺昵称  关联TaobaoShopInfo表的Nick |
| SignContent | VARCHAR(100) |  |  | 签名内容 |
| IsOpen | Int(1) | 0 | true | 是否开启（0 false 1 true） |
| UpdateTime | TIMESTAMP | ON UPDATE CURRENT\_TIMESTAMP | false | 最后更新时间 |

* 1. 主、子店铺关联关系表(ShopRelevantToInfo)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段名 | 字段类型 | 默认值 | 是否为空 | 备注 |
| Id | BIGINT(8) | 0 | false | ID主键 |
| UserId | BIGINT(10) | 0 | false | 主店铺数字ID  对应BackgroundUserInfo表中Id |
| ParentNick | VARCHAR(50) |  |  | 主店铺昵称 |
| Nick | VARCHAR(50) |  |  | 子店铺昵称 |
| IsDelete | INT(1) | 0 |  | 是否解绑，0：否 1:是 |
| Description | VARCHAR(100) |  |  | 描述 |
| UpdateTime | TIMESTAMP | ON UPDATE CURRENT\_TIMESTAMP | false | 最后更新时间 |

* 1. 流程任务表(FlowTaskInfo)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段名 | 字段类型 | 默认值 | 是否为空 | 备注 |
| Id | BIGINT(8) | 0 | false | 任务ID |
| UserId | BIGINT(8) | 0 | true | 用户Id  关联BackGroundUserInfo表的Id |
| WorkFlowID | BIGINT(8) | 0 | true | 工作流程ID  关联WorkFlowInfo表的Id |
| StartTime | TIMESTAMP | CURRENT\_TIMESTAMP | false | 开始时间 |
| EndTime | TIMESTAMP | 0000-00-00 00:00:00 | false | 结束时间 |
| Status | VARCHAR(50) |  | true | 状态 |
| ExecuteType | VARCHAR(10) |  | true | 执行类型 |
| CurrentNodeID | VARCHAR(1000) |  | true | 当前执行节点  关联ProcessDesignDescInfo表的Id |
| UpdateTime | TIMESTAMP | 0000-00-00 00:00:00 | false | 最后更新日期 |

1. 系统接口详细设计
   1. 工作方式

客户程序通过HTTP协议、POST方式提交数据。服务端程序在完成鉴权、屏蔽字过滤等处理后，将提交结果做为响应输出。

客户程序需保证数据的合法性、有效性，每次发送请求都有响应(响应返回格式为纯文本)，安全验证目前采用用户名和密码的方式。

合法性是指请求数据符合HTTPS协议，有效性是指请求数据包含必要的参数。

* 1. 字符编码

请求与响应，均为UTF8编码。

* 1. 调用示例
     1. 生成协议

接口地址：http://t.tb1.cn:8080/api/short-url/create

通信方式：HTTP POST。

|  |  |
| --- | --- |
| 参数名称 | 参数说明 |
| token | (appkey/secret) appkey帐号，secret密码 由北京慧博分配 |
| time | 系统当前时间毫秒级(System.currentTimeMillis()) |
| data | {"sourceUrl":""} sourceUrl需要生成短链的原链接 |

* + - 1. 返回

如果参数正确，返回如下：

{

"result":{

"message" : "success",

"status" : "200"

},

"data":{

"shortUrl" : "….",

"sourceUrl" : "…."

}

}

shortUrl : 返回生成短链

sourceUrl : 原链接

如果参数有误，返回如下：

{

"result":{

"message" : "invalid appkey secret",

"status" : "401"

}

}

* + - 1. 示例代码

**public** **class** ShortUrlDemo {

**private** **static** **final** Logger *logger* = Logger.*getLogger*(ShortUrlDemo.**class**);

**private** **static** **final** String *HB\_URL\_PREFFIX* = "http://t.tb1.cn:8080";

@Test

**public** **void** shortUrl() **throws** Exception {

HashMap<String, Object> dataMap = **new** HashMap<String, Object>();

dataMap.put("sourceUrl", "www.baidu.com");

String appkey = "appkey";

String secret = "secret";

String assessKey = appkey+"/"+secret;

Map<String, Object> postParamMap = **new** HashMap<String, Object>();

postParamMap.put("token", assessKey);

postParamMap.put("time", System.*currentTimeMillis*());

postParamMap.put("data", dataMap);

String result = *doPost*(postParamMap,"/api/short-url/create",assessKey);

System.*out*.println(result);

}

**private** **static** String doPost(Map<String,Object > paramMap,String uri,String assessKey){

**try** {

String URL\_PREFIX = *HB\_URL\_PREFFIX*;

Date startDate = **new** Date();

HttpClient httpClient = **new** DefaultHttpClient();

httpClient = *wrapClient*(httpClient);

HttpPost httpPost = **new** HttpPost(URL\_PREFIX+uri);

httpPost.addHeader("Encoding","UTF-8");

//设置参数

*logger*.info("uri:" + uri);

*logger*.info("参数:");

*logger*.info(JSONObject.*fromObject*(paramMap).toString());

KeyModel keyModel = AESUtil.*getKeyModel*(assessKey);

paramMap.put("token", assessKey);

paramMap.put("time", keyModel.getTime());

StringEntity entity = **new** StringEntity(JSONObject.*fromObject*(paramMap).toString(),"UTF-8");

httpPost.setEntity(entity);

HttpResponse response = httpClient.execute(httpPost);

String result = "";

**if**(response != **null**){

HttpEntity resEntity = response.getEntity();

**if**(resEntity != **null**){

result = EntityUtils.*toString*(resEntity);

*logger*.info("返回原串: " + result);

**if**(result.trim().startsWith("{")){

result = JSONObject.*fromObject*(result).toString();

}**else**{

result = JSONArray.*fromObject*(result).toString();

}

}

}

*logger*.info("结果:");

*logger*.info(result);

Date endDate = **new** Date();

**return** result;

} **catch** (Exception e) {

**throw** **new** RuntimeException("接口调用失败!");

}

}

**private** **static** HttpClient wrapClient(HttpClient base) {

**try** {

SSLContext ctx = SSLContext.*getInstance*("TLS");

X509TrustManager tm = **new** X509TrustManager() {

**public** **void** checkClientTrusted(X509Certificate[] xcs,String string) {

}

**public** **void** checkServerTrusted(X509Certificate[] xcs,String string) {

}

**public** X509Certificate[] getAcceptedIssuers() {

**return** **null**;

}

};

ctx.init(**null**, **new** TrustManager[] { tm }, **null**);

~~SSLSocketFactory~~ ssf = **new** SSLSocketFactory(ctx);

ssf.setHostnameVerifier(~~SSLSocketFactory~~.*ALLOW\_ALL\_HOSTNAME\_VERIFIER*);

~~ClientConnectionManager~~ ccm = base.~~getConnectionManager~~();

~~SchemeRegistry~~ sr = ccm.getSchemeRegistry();

sr.register(**new** ~~Scheme~~("https", ssf, 443));

**return** **new** DefaultHttpClient(ccm, base.~~getParams~~());

} **catch** (Exception ex) {

ex.printStackTrace();

**return** **null**;

}

}

}

请求数据格式示例:

{"data":{"sourceUrl":"www.baidu.com"},"time":"1498813581000","token":"appkey/secret"}

* + 1. 查询协议

接口地址：http://t.tb1.cn:8080/api/short-url/query

通信方式：HTTP POST。

|  |  |
| --- | --- |
| 参数名称 | 参数说明 |
| token | appkey帐号，secret密码 由北京慧博分配 |
| time | 系统当前时间毫秒级(System.currentTimeMillis()) |
| data | {"data":{"shortUrl":"…","pageSize":"0","currentPage":"10"}} shortUrl需要查询的短链 pageSize:每页多少条 currentPage:当前页(0,1,2….) |

* + - 1. 返回

如果参数正确，返回如下：

{

"result":{

"message" : "success",

"status" : "200"

},

"data":{

"dataList":[

{"system":"PC端","click":"2017-06-30 16:54:39"}

{"system":"PC端","click":"2017-06-30 16:54:39"}

],

"totalCount":6

}

}

dataList: 查询页数据(返回点击时间以及点击终端类型)

totalCount: 总记录数

如果参数有误，返回如下：

{

"result":{

"message" : "invalid appkey secret",

"status" : "401"

}

}

* + - 1. 示例代码

{"data":{"shortUrl":"t.tb1.cn/Ni2q1Y","pageSize":"0","currentPage":"10"},"time":"1498813002000","token":"appkey/secret"}