国航积分兑换平台

用户行为分析设计

**文档信息**

|  |  |  |  |
| --- | --- | --- | --- |
| **项目名称：** | 国航商业模式创新一期 | **文档名称：** | 用户行为分析 |
| **文档版本号：** | 0.1 | **文档版本日期：** |  |
| **作者：** |  | **起草日期：** | 2017-08-22 |

**文档变更历史**

| **版本号** | **版本日期** | **修改人** | **变更描述** |
| --- | --- | --- | --- |
| 0.1 | 2017-08-22 | 魏晓锋 | 创建文档 |

目录

[第1章 引言 4](#_Toc494208681)

[1.1 编写目的 4](#_Toc494208682)

[1.2 读者对象 4](#_Toc494208683)

[1.3 缩略词 4](#_Toc494208684)

[1.4 参考资料 5](#_Toc494208685)

[第2章 需求概述 5](#_Toc494208686)

[2.1 前提与假设 5](#_Toc494208687)

[2.2 边界说明 5](#_Toc494208688)

[2.3 Actors 5](#_Toc494208689)

[2.4 场景说明 5](#_Toc494208690)

[第3章 用例图 6](#_Toc494208691)

[第4章 数据模型设计 6](#_Toc494208692)

[第5章 实现设计 6](#_Toc494208693)

[5.1 会员登录 6](#_Toc494208694)

[5.2 会员搜索商品： 7](#_Toc494208695)

[5.3 会员浏览商品： 7](#_Toc494208696)

[5.4 加入购物车 8](#_Toc494208697)

[5.5 公共部分代码 8](#_Toc494208698)

[第6章 日志 9](#_Toc494208699)

[第7章 接口 9](#_Toc494208700)

[7.1 内部接口 9](#_Toc494208701)

[7.2 外部接口 9](#_Toc494208702)

[第8章 异常 9](#_Toc494208703)

[8.1 异常 9](#_Toc494208704)

[8.2 错误编码 9](#_Toc494208705)

# 引言

## 编写目的

## 读者对象

本文档供以下相关人员阅览：

* 本项目管理人员
* 本项目的需求分析人员
* 本项目的系统设计人员
* 本项目的配置开发人员
* 本项目的界面开发人员
* 本项目的接口开发人员
* 本项目组的测试人员

## 缩略词

本项目主要采用的审计术语约定如下：

|  |  |
| --- | --- |
| 术语 | 解释 |
| FFP | Frequent Flyer Program飞行常客计划，本文中特指国航常旅客系统 |
| CRM | 国航CRM（Customer Relationship Management）系统 |
| B2C | Business-to-Consumer，网络直销电子商务 |
| 中航信 | 中国民航信息网络股份有限公司 |
| TTL | 中国民航信息网络股份有限公司 |
| TravelSky | 中国民航信息网络股份有限公司 |
| PNR | 乘客订座记录，即Passenger Name Record的缩写，它反映了乘客的航程，航班座位占用的数量及乘客信息 |
| IBE接口 | E-Build 基础API 接口（IBE接口）是航信提供的航空公司电子商务网站解决方案中的底层基础服务接口 |
| PhoenixMiles | “凤凰知音”是国航常旅客里程奖励计划名称。 |
| 订单号 | 系统中用来唯一标识一笔订单的编号。 |
| 票号 | 票号是航信系统中用来标识乘客客票信息的编号，由主机生成，被记录在PNR中。 |
| 银行支付订单号 | 银行支付订单号是系统为一次支付生成的一个编号，该编号会被支付平台记录，并作为后续处理环节中对该支付的唯一标识。 |
| 供应商 | 特指为国航知音商城提供商品的合作企业 |
| 奖励机票 | 指航空公司预留的会员可使用里程兑换的机票类型，舱位和积分规则与正常舱位不同。星空联盟的通用舱位名称为O、I、X舱位，分别对应头等、公务、经济舱位。 |
| 可销售机票 | 指使用里程兑换的正常销售客票，舱位和使用、积分规则与正常客票规则一致。 |

## 参考资料

# 需求概述

## 前提与假设

## 边界说明

会员在积分商城的行为分析

## Actors

积分平台

APP

航信

## 场景说明

会员登录

记录会员卡号、IP、登录时间、

会员搜索商品：

记录会员卡号、会员搜索条件

会员浏览商品：

记录会员卡号、商品id

加入购物车

记录会员卡号、商品id、购买数量、商品规格id

下单（此部分不做记录，积分平台已经入库）

# 用例图

# 数据模型设计

# 实现设计

## 会员登录

事件Id: event1

记录会员id、IP、登录时间

1.修改login.js，登录成功后加入代码：

pt.setVar(1,"events","event1");

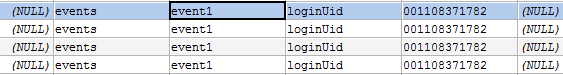
pt.setVar(2,"loginUid",$("#loginUid").val());

pt.track();

2.修改login.ftl，增加如下代码

<script src="${base}/res/js/ptracker.js" /></script>

<script src="${base}/res/js/ptracker\_code.js" /></script>



## 会员搜索商品：

事件Id: event2

记录会员卡号、会员搜索条件

修改header.ftl

1. 引入js库

<script src="${base}/res/js/ptracker.js" /></script>

<script src="${base}/res/js/ptracker\_code.js" /></script>

1. 修改按钮click事件 658行

$("#button").click(function(){

//用户行为

if(trim($("#keyword").val()) == ""){

return;

}

pt.setVar(1,"events","event2");

pt.setVar(2,"keyword",$("#keyword").val());

pt.setVar(3,"loginUid","${principal.crmMemberNumber}");

pt.track();

window.location.href="${base}/search/goodsSearch?searchType=keywordSearch&keyword="+encodeURIComponent($("#keyword").val());

})



## 会员浏览商品：

事件Id: event3

记录商品id

修改product-detail.ftl

1. 引入js库

<script src="${base}/res/js/ptracker.js" /></script>

<script src="${base}/res/js/ptracker\_code.js" /></script>

1. 头部增加代码

<@shiro.authenticated>

<#assign principalTag =newTag("memberPrincipalTag")>

<#assign principal =principalTag()>

</@shiro.authenticated>

1. 尾部增加脚本

<script>

pt.setVar(1,"events","event3");

pt.setVar(2,"goodsid","${goodsId}");

pt.setVar(5,"loginUid","${principal.crmMemberNumber}");

pt.track();

</script>



## 加入购物车

事件Id: event4

记录商品sku

修改product-detail.ftl

1. 修改函数buy(type)

在if(specId !=''){后加入代码992行

//用户行为分析

pt.setVar(1,"events","event4");

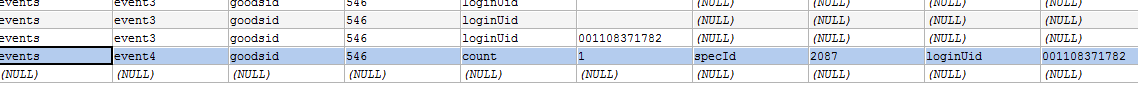
pt.setVar(2,"goodsid",goodsId);

pt.setVar(3,"count",goodsNum);

pt.setVar(4,"specId",specId);

pt.setVar(5,"loginUid","${principal.crmMemberNumber}");

pt.track();



## 公共部分代码

**var** pt = {};

pt.tracker = Piwik.getAsyncTracker("//10.9.245.48/piwik/piwik.php", "1");

pt.setVar = **function**(seq,varName,value){

**this**.tracker.setCustomVariable(seq,varName,value,"page");

}

pt.track = **function**(){

**this**.tracker.trackPageView();

**this**.tracker.enableLinkTracking();

}

# 日志

# 接口

## 内部接口

## 外部接口

# 异常

## 异常

## 错误编码