**购物平台用户行为**

**大数据分析系统设计书**

项目组组长：邹圣栋

项目组成员：李林轩、陆于炀、姜振楠、葛奔奔、蔡嘉隆

日期：2019.12.4

目录

[一、大数据分析需求说明 1](#_Toc8226)

[1.1 热门商品离线统计 1](#_Toc29527)

[1.2 订单信息的统计 1](#_Toc8015)

[1.3 实时统计活跃人群信息 1](#_Toc16080)

[二、项目要求 1](#_Toc31102)

[2.1编码规范和编码严谨性 1](#_Toc13200)

[2.2技术选择要求 1](#_Toc26104)

[2.3项目管理 2](#_Toc4117)

[三、项目流程图 2](#_Toc25652)

[3.1总体业务流程图 2](#_Toc27937)

[3.2购物平台流程图 3](#_Toc3181)

[3.3用户注册子流程 4](#_Toc25217)

[3.4实时数据处理流程图 5](#_Toc24015)

[四、项目功能 6](#_Toc26360)

[4.1购物平台子系统 6](#_Toc16175)

[4.1.1注册 6](#_Toc27812)

[4.1.2登录 6](#_Toc20076)

[4.1.3商城主页 6](#_Toc1357)

[4.1.4商品详情页 6](#_Toc15338)

[4.1.5购物车 6](#_Toc30343)

[4.2大数据后台管理子系统 6](#_Toc6850)

[4.2.1热门商品统计 6](#_Toc9103)

[4.2.2订单信息统计 7](#_Toc5021)

[4.2.3实时统计活跃人群信息 7](#_Toc9024)

[五、数据库表设计 7](#_Toc5983)

[5.1 Mysql表业务设计 7](#_Toc19102)

[5.1.1用户表 user\_info 7](#_Toc22524)

[5.1.2商品表 goods\_info 7](#_Toc30916)

[5.1.3商品图片表 goods\_picture 8](#_Toc16692)

[5.1.4订单信息表 order\_info 8](#_Toc5946)

[5.1.5订单详情表 order\_detail 9](#_Toc14355)

[5.1.6购物车表 shopping\_car 9](#_Toc6653)

[5.2 Hive表设计 9](#_Toc24404)

[5.2.1商品下单记录表 goods\_order\_record(按天分区) 9](#_Toc25788)

[5.2.2商品下单次数表 goods\_order\_count(不分区) 10](#_Toc20523)

[5.2.3订单表 order\_info（分区） 10](#_Toc26396)

[5.3 MySql报表设计 10](#_Toc29177)

[5.3.1热门商品表 hot\_goods\_info （饼状图显示） 10](#_Toc4099)

[5.3.2订单统计表 order\_count （两张折线图显示） 11](#_Toc12529)

[5.3.3活跃人群统计表 active\_count（折线图显示） 11](#_Toc30769)

[六、详细设计 11](#_Toc4808)

[6.1热门商品离线统计 11](#_Toc26290)

[6.2订单信息的统计 12](#_Toc25045)

[6.3实时(每隔一小时)统计活跃人群信息 14](#_Toc9720)

**一、大数据分析需求说明**

**1.1 热门商品离线统计**

该模块主要实现每天统计出各个区域的top3热门商品。然后使用Oozie进行离线统计任务的定时调度；使用Echarts进行数据可视化的报表展示。该模块可以让企业管理层看到公司售卖的商品的整体情况，从而对公司的商品相关的战略进行调整。

**1.2 订单信息的统计**

总的订单个数、成功支付的订单个数、退款订单个数;

迄今为止，总的订单交易额和总的退款金额等。

**1.3 实时统计活跃人群信息**

生成按小时统计信息的统计表,得出每天24小时的活跃人群信息。

**二、项目要求**

**2.1编码规范和编码严谨性**

(1)编码规范：包括建表规范，java/scala/spark编写规范；

(2)命名规范：包括变量名、方法名、函数名、类名等。数据库字段使用下 划线区分单词（user\_info），代码中使用驼峰命名法(userName)。

(3)特别强调：不允许在项目期间的任何命名出现拼音，不允许写错英语单 词。

**2.2技术选择要求**

(1)在本项目期间Web阶段使用HTML5代替JSP。

(2)使用Spring及Spring MVC、SpringBoot、Spring Security、Spring Cloud等代替Servlet。

(3)鼓励使用数据库封装技术：Mybatis、Spring JDBC；不允许使用Hibernate、JPA等。

(4)鼓励使用Spring Transaction进行事务管理。

(5)鼓励使用胖客户端项目开发形式。鼓励使用js框架(JQuery)、流行的前端页面框架(Bootstrap)、及报表框架,页面展现鼓励使用ElementUI等。

(6)采用Spark代替MapReduce。

(7)大数据实时处理阶段开发鼓励使用：Kafka+SparkStreaming，不建议使用Storm。

**2.3项目管理**

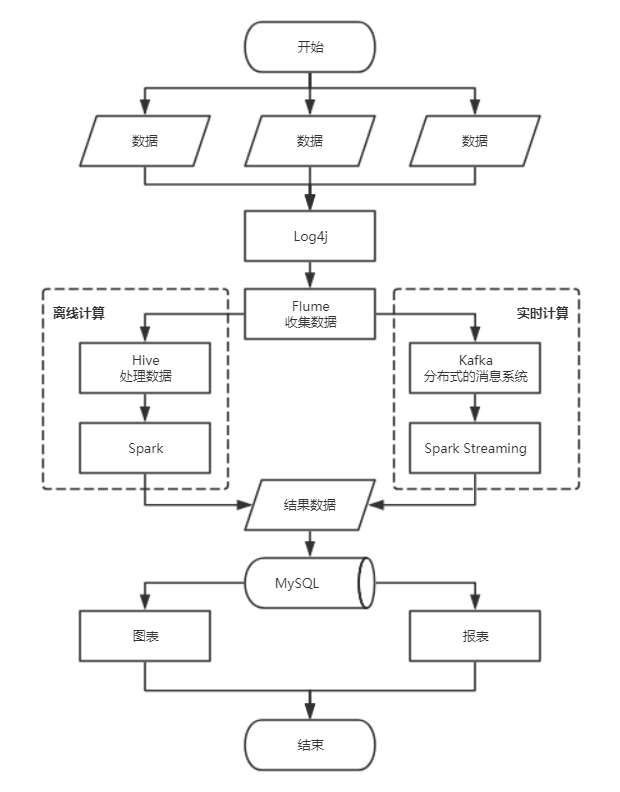
(1)Web开发阶段采用Maven工程

(2)团队开发，管理工具采用Git及Github

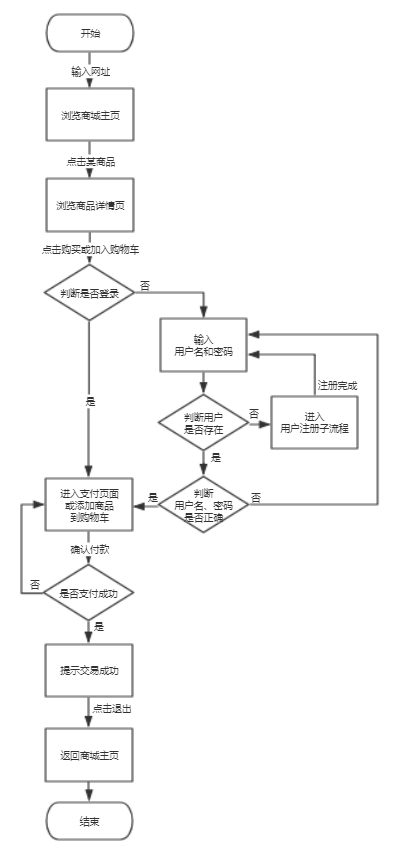
(3)项目期间最终成果物提交到SVN

**三、项目流程图**

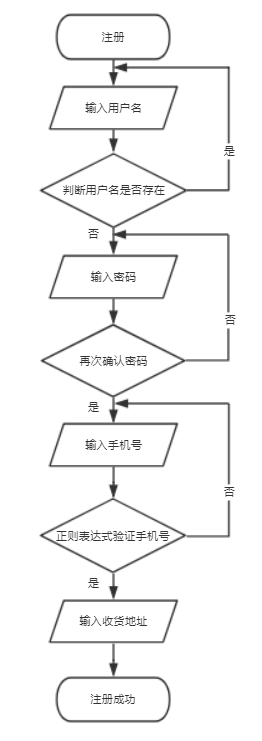
**3.1总体业务流程图**



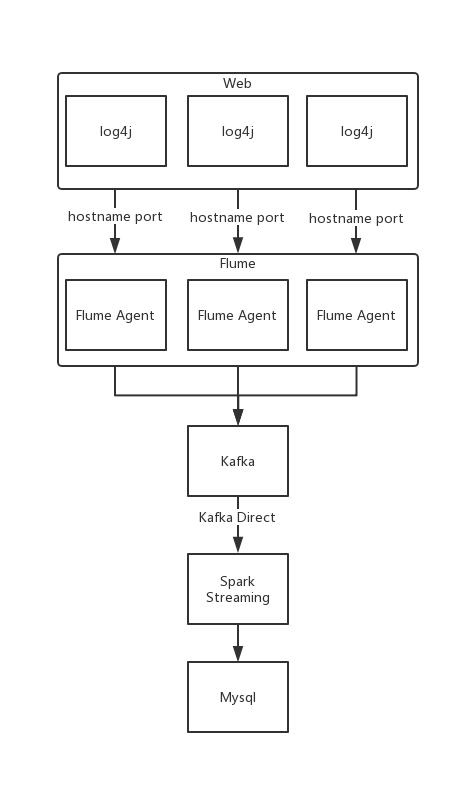
**3.2购物平台流程图**



**3.3用户注册子流程**

****

**3.4实时数据处理流程图**



**四、项目功能**

**4.1购物平台子系统**

**4.1.1注册**

用户进入商城之后，点击注册按钮后，跳转到注册页面，在注册页面填写用户名、密码等信息来进行注册，产生出来的用户信息，利用sql语句添加到用户表中。

**4.1.2登录**

用户通过输入对应的用户名和密码来登录商城，以进行下单的操作，如果用户没有登录，也可以浏览商城，查看商品的基本信息，一旦涉及到加入购物车与购买的操作，系统将自动跳转到登录页面，如果用户没有账号，可以点击注册按钮来跳转到注册页面进行注册。

**4.1.3商城主页**

商城主页分为几个模块：导航栏、搜索栏、热门商品推荐等，导航栏主要用于商品的分类，用户在导航栏中点击对应的分类，即可进入对应的分类去选购对应商品，如果用户已经有确定的商品，就可以通过搜索栏进行搜索，来定位到相应的商品，用户也可以通过主页上的商品推荐，点击进入对应热门商品的详情介绍，来选购相应的商品。

**4.1.4商品详情页**

用户在找到自己想要的商品时，可以点击对应商品的词条来进入详情页查看商品的具体信息，同时在详情页上来进行对商品的购买或将其添加入购物车。

**4.1.5购物车**

用户可以直接在商品的详情页，将自己想要的商品添加入购物车，方便以后的购买。添加入购物车的商品，其对应的商品信息与操作用户的信息将被添加到购物车表中。用户点击我的购物车可查看添加到购物车中的商品，勾选某几个商品后可以显示出商品的合计金额。

**4.2大数据后台管理子系统**

**4.2.1热门商品统计**

利用sql语句将各分类中每日的下单次数top3的热门商品给查询出来，然后使用Oozie进行离线统计任务的定时调度，使用Echarts进行数据可视化报表的展示

**4.2.2订单信息统计**

所有用户的每一笔交易信息都将被存放入订单表中，每日进行一次统计，统计的数据包括今日付款成功与退款成功的订单数，今日所有订单的交易总额与退款总额。最后可以查看当天的订单信息（添有日期字段），还可以查看一周内每天订单数的变化情况，以及一周内订单交易总额和退款总额的汇总情况。

**4.2.3实时统计活跃人群信息**

生成按小时统计信息的统计表,每隔一小时扫描一次这段时间内登录过的用户，从而得出每天24小时的活跃人群信息。

**五、数据库表设计**

**5.1 Mysql业务表设计**

**5.1.1用户表 user\_info**

|  |  |  |  |
| --- | --- | --- | --- |
| **字段名** | **类型** | **描述** | **约束** |
| **user\_id** | int | 用户编号，自增 | 主键 |
| **user\_name** | varchar(30) | 用户名 |  |
| **user\_password** | varchar(32) | 密码，使用MD5加密 |  |
| **user\_phone** | bigint | 手机 |  |
| **user\_address** | varchar(255) | 收货地址 |  |
| **login\_status** | char(1) | 登录状态：1（登录）、0（退出） |  |
| **active\_status** | char(1) | 每小时活跃状态：  1(在线)、0(离线) |  |

**5.1.2商品表 goods\_info**

|  |  |  |  |
| --- | --- | --- | --- |
| **字段名** | **类型** | **描述** | **约束** |
| **goods\_id** | int | 商品编号，自增 | 主键 |
| **goods\_name** | varchar(50) | 商品名 |  |
| **goods\_price** | decimal(8,2) | 价格 |  |
| **goods\_store** | int | 库存 |  |
| **goods\_detail** | text | 商品描述 |  |
| **goods\_shop** | varchar(50) | 店铺名 |  |
| **goods\_category** | varchar(30) | 商品分类 |  |

**5.1.3商品图片表 goods\_picture**

|  |  |  |  |
| --- | --- | --- | --- |
| **字段名** | **类型** | **描述** | **约束** |
| **picture\_id** | int | 图片编号，自增 | 主键 |
| **goods\_id** | int | 商品编号 | 外键 |
| **picture\_path** | varchar(100) | 图片路径 |  |

**5.1.4订单信息表 order\_info**

|  |  |  |  |
| --- | --- | --- | --- |
| **字段名** | **类型** | **描述** | **约束** |
| **order\_id** | varchar(50) | 订单编号  （由下单时间或购物车结算时间  和user\_id拼接而成，例：20191207-103000-1） | 主键 |
| **user\_id** | int | 用户编号 | 外键 |
| **consignee\_name** | varchar(30) | 收货人姓名 |  |
| **order\_sum** | decimal(8,2) | 订单合计金额 | null |
| **order\_status** | char(1) | 订单状态：  1(下单成功)  2(付款成功)  3(退款成功) |  |
| **create\_time** | timestamp | 订单创建时间 |  |
| **update\_time** | timestamp | 订单状态修改时间 |  |

点击直接购买或购物车结算时，跳转到订单生成页，生成订单编号。上面有一个输入框，用于输入收货人姓名。收货地址默认值来自用户表中，若改变收货地址则需要更新用户表。

点击付款时，弹出一个确认框：是否确认付款？点击否：订单状态为1；点击是：订单状态为2。先不插入订单合计金额，将其余数据插入订单信息表。

**5.1.5订单详情表 order\_detail**

一对多关系设计：一个订单可以对应多个商品

|  |  |  |  |
| --- | --- | --- | --- |
| **字段名** | **类型** | **描述** | **约束** |
| **order\_detail\_id** | int | 订单表id，自增 | 主键 |
| **order\_id** | varchar(50) | 订单编号 | 外键 |
| **goods\_id** | int | 商品编号 | 外键 |
| **goods\_number** | int | 购买数量 |  |
| **goods\_sum** | decimal(8,2) | 商品总金额 |  |

生成订单信息表后，接着生成订单详情表（使用商品编号查询出商品单价后，算出商品总金额）。订单详情表生成完后，通过一个订单编号查询出多个商品总金额，合计后更新订单详情表的订单合计金额字段。

**5.1.6购物车表 shopping\_car**

一对多关系设计：一个用户可以将多个商品添加进购物车

|  |  |  |  |
| --- | --- | --- | --- |
| **字段名** | **类型** | **描述** | **约束** |
| **shopping\_car\_id** | int | 购物车id，自增 | 主键 |
| **user\_id** | int | 用户编号 | 外键 |
| **goods\_id** | int | 商品编号 | 外键 |
| **goods\_number** | int | 购买数量 |  |

购买数量使用Ajax随前台的变化而更新。

当选中购物车中的某几个商品，点击结算时，在前台获取每一条记录的shopping\_car\_id，放进一个集合，传入后台，后台遍历集合逐一操作：根据shopping\_car\_id将用户编号传递给订单表，将商品编号和购买数量传递给订单详情表。

**5.2 Hive表设计**

**5.2.1商品下单记录表 goods\_order\_record(按天分区)**

|  |  |  |
| --- | --- | --- |
| **字段名** | **类型** | **描述** |
| **goods\_id** | int | 商品编号 |
| **goods\_name** | string | 商品名字 |
| **goods\_category** | string | 商品分类 |
| **one** | int | 固定为1，用于次数的统计 |
| **day** | string | 按天分区，该字段自动生成 |

**5.2.2商品下单次数表 goods\_order\_count(不分区)**

|  |  |  |
| --- | --- | --- |
| **字段名** | **类型** | **描述** |
| **goods\_id** | int | 商品编号 |
| **goods\_name** | string | 商品名字 |
| **goods\_category** | string | 商品分类 |
| **sum** | int | 下单次数合计（热度值） |
| **day** | string | 当天日期（年月日） |

**5.2.3订单表 order\_info（分区）**

|  |  |  |
| --- | --- | --- |
| **字段名** | **类型** | **描述** |
| **order\_id** | string | 订单编号 |
| **orders\_sum** | decimal(8,2) | 合计金额 |
| **user\_name** | string | 用户名 |
| **consignee\_name** | string | 收货人姓名 |
| **order\_time** | timestamp | 修改时间 |
| **order\_status** | char(1) | 订单状态 |
| **day** | date | 按天分区，该字段自动生成 |

**5.3 MySql报表设计**

注：各个报表的详细信息通过多表查询获得。

**5.3.1热门商品表 hot\_goods\_info （饼状图显示）**

|  |  |  |  |
| --- | --- | --- | --- |
| **字段名** | **类型** | **描述** | **约束** |
| **hot\_goods\_id** | int | 热门商品表id，自增 | 主键 |
| **goods\_id** | int | 商品编号 | 外键 |
| **goods\_hot** | int | 热度（按下单量，默认值为0） |  |
| **goods\_date** | date | 当前日期 |  |

**5.3.2订单统计表 order\_count （两张折线图显示）**

|  |  |  |  |
| --- | --- | --- | --- |
| **字段名** | **类型** | **描述** | **约束** |
| **order\_count\_id** | int | 订单统计表id，自增 | 主键 |
| **order\_sum\_count** | int | 总的订单个数 |  |
| **order\_pay\_count** | int | 成功支付的订单个数 |  |
| **order\_refund\_count** | int | 退款的订单个数 |  |
| **order\_trade\_sum** | decimal(10,2) | 订单交易总额 |  |
| **order\_refund\_sum** | decimal(10,2) | 订单退款总额 |  |
| **order\_count\_date** | date | 当前日期 |  |

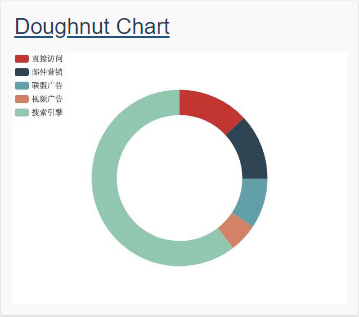
**5.3.3活跃人群统计表 active\_count（折线图显示）**

|  |  |  |  |
| --- | --- | --- | --- |
| **字段名** | **类型** | **描述** | **约束** |
| **active\_count\_id** | int | 活跃人群统计表id，自增 | 主键 |
| **per\_hour** | timestamp | 时间 |  |
| **active\_count** | int | 每小时活跃人数 |  |

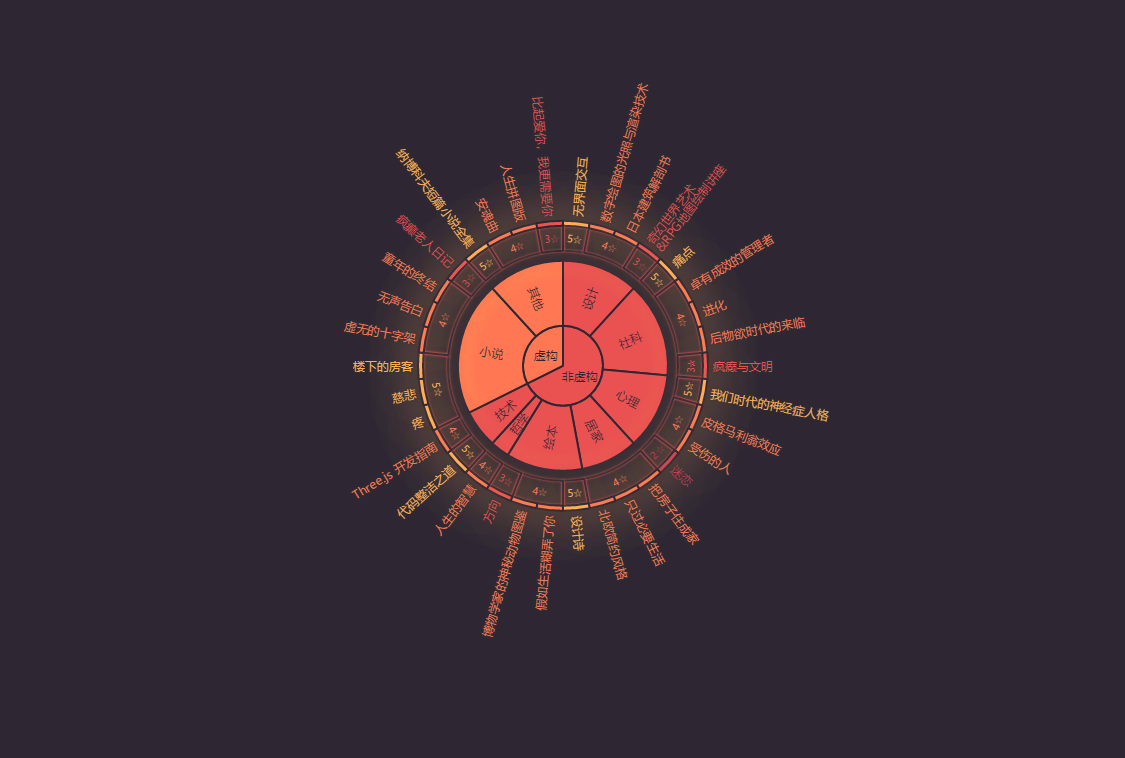
**六、详细设计**

**6.1热门商品离线统计**

每当有一个用户下单成功后，就把该商品信息通过日志传入到Flume中接着导入到Hive的商品下单记录表，Hive表按天做分区存储（格式为<商品编号，1，分区时间>）。Spark获取到Hive中的数据进行操作，先用Spark Core分组统计各个商品下单次数，结果存入商品下单次数表，Spark Core 从商品下单次数表取出数据，排序后，取出各个分类的最热门的3个商品，并用Spark Sql将每天统计的结果追加到Mysql的热门商品表中（在商品表的基础上新增热度和时间字段）。将Spark Sql程序打成Jar包，用Oozie工作流实现每天零点调度一次任务。最后使用Echarts的显示数据报表。



Top3热门商品饼图



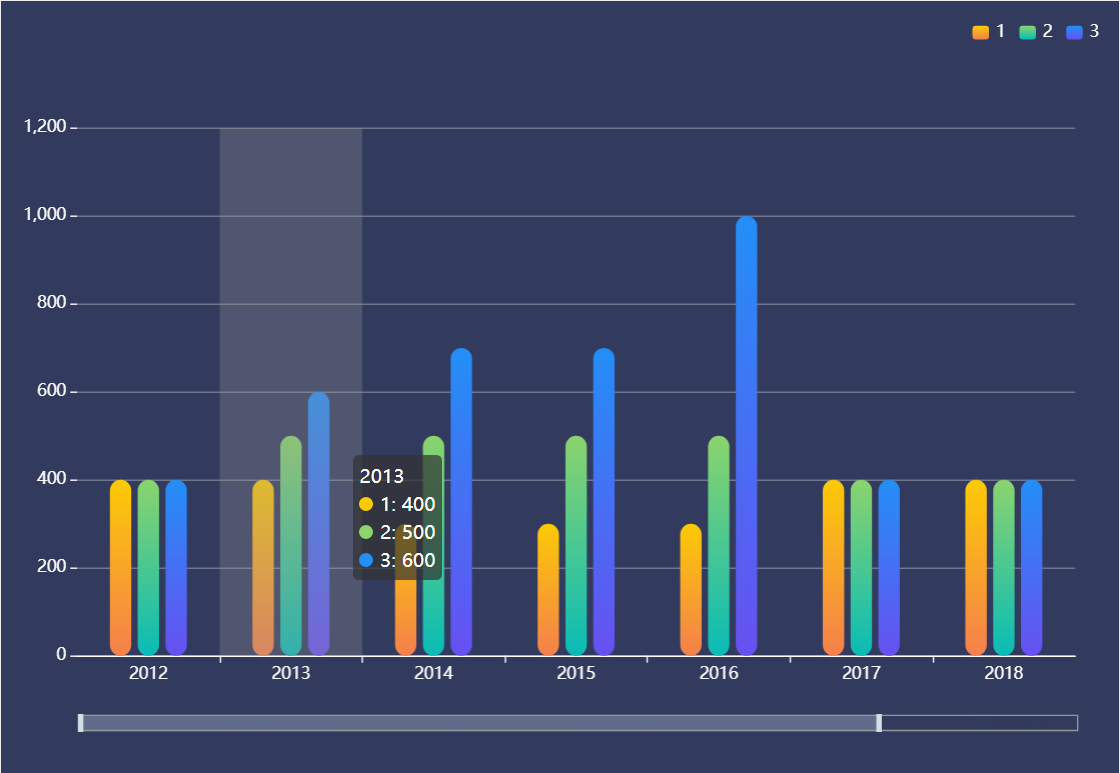
旭日图显示Top3热门商品(内层圆为分类，外层扇形为比例)

**6.2订单信息的统计**

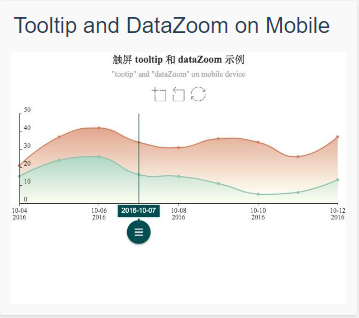
订单表中添加订单的状态信息，订单的状态分为下单成功、付款成功、退款成功三种，订单的状态随着用户的操作进行改变（update）。

每当有用户对订单进行操作后，就将订单的信息通过日志传入到Flume中接着导入到Hive的订单表（日志中的订单信息对应Hive订单表中的每一个字段），并且将操作订单的时间放入Hive订单表的修改时间字段，通过最新的时间确定订单的状态（开窗函数），从而进行订单信息的统计。

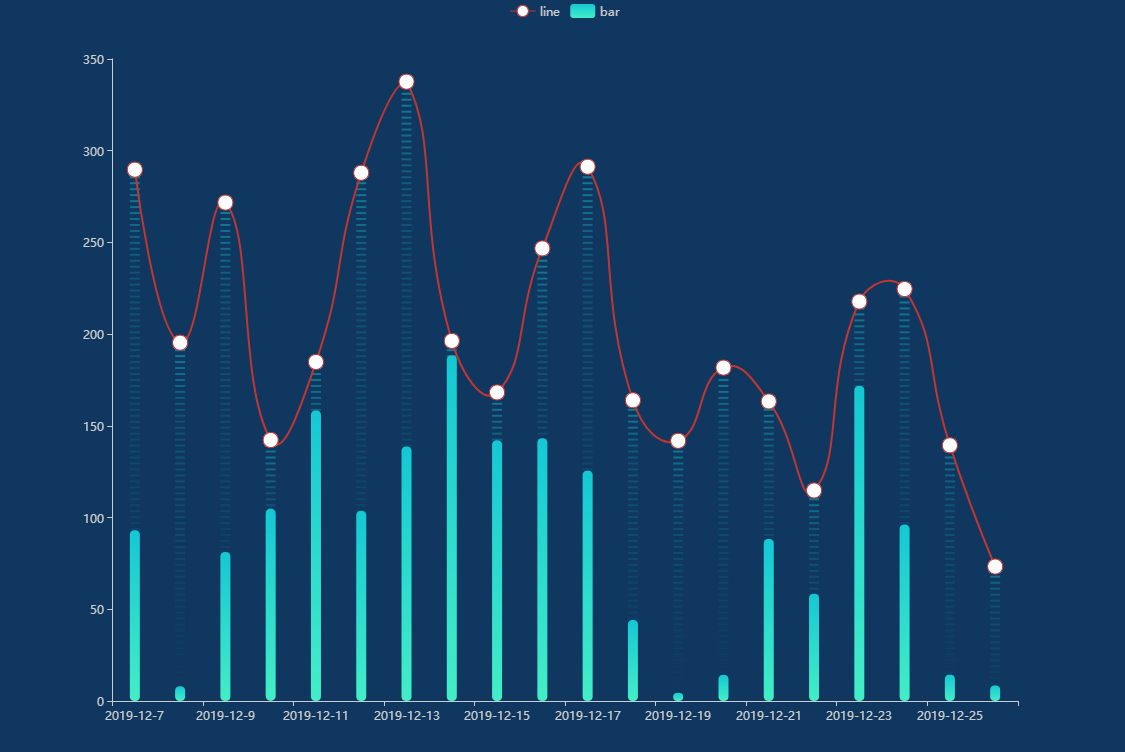
利用Spark SQL进行订单信息的统计，其中包括今日付款成功的订单数与退款成功的订单数，今日订单的交易总额与退款总额，最后使用Spark SQL将统计出来的所有数据放入MySQL的订单统计表中，并加入当天的日期，这样就能查看一周内每天订单数的变化情况（使用折线图反映），以及一周内订单交易总额和退款总额的汇总情况。



每日订单总个数、成功支付个数、退款成功个数柱状图（可移动）



一周交易总额和退款总额折线图



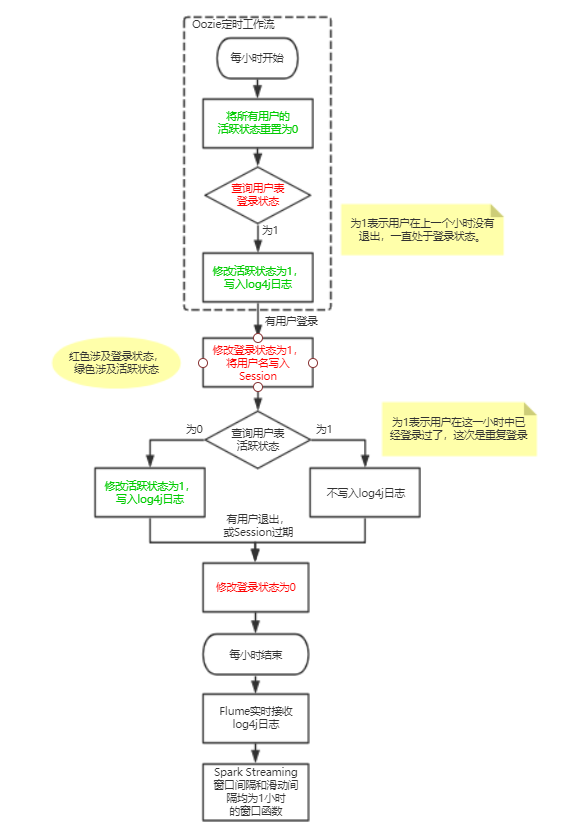
一周交易总额和退款总额象形柱图

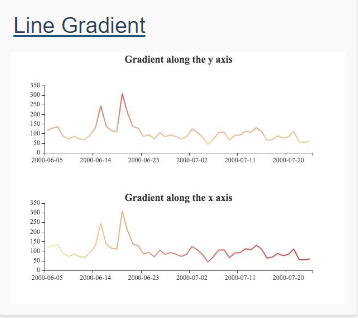
**6.3实时(每隔一小时)统计活跃人群信息**

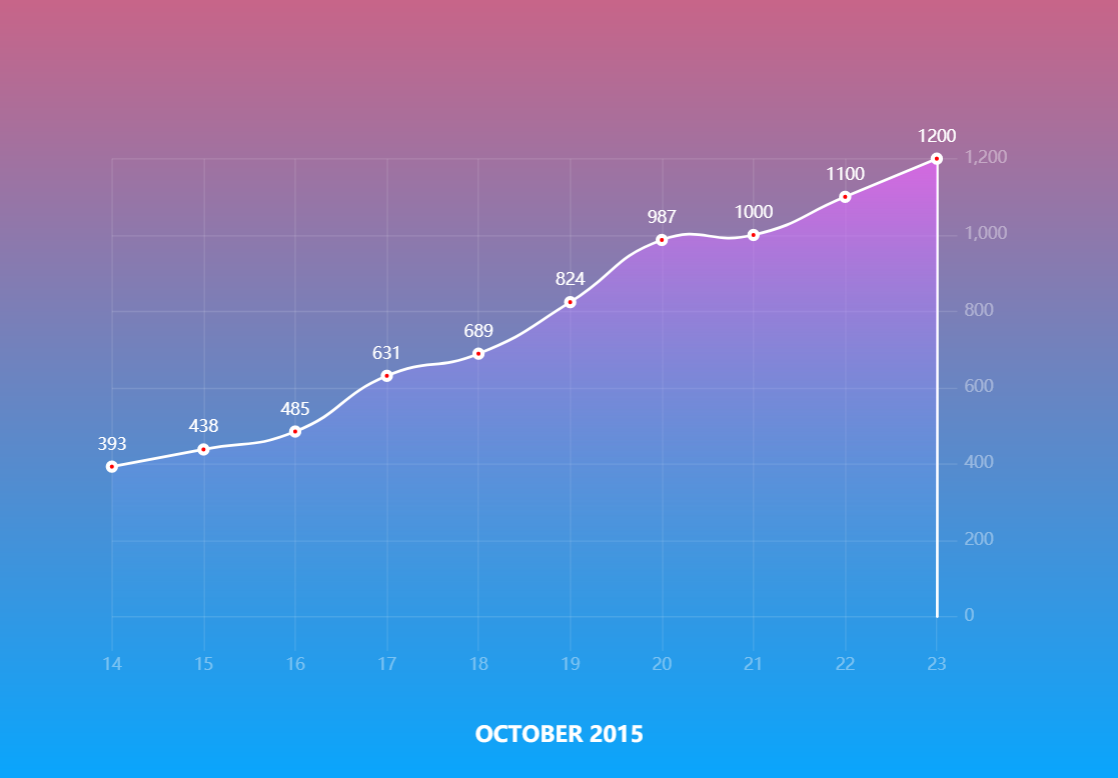
用户表添加登录状态和活跃状态字段，登录状态为0时表示退出，为1时表示登录；活跃状态为0时表示一小时之内没有登录过，为1时表示一小时之内登录过或者超过一小时但仍然保持登录状态。

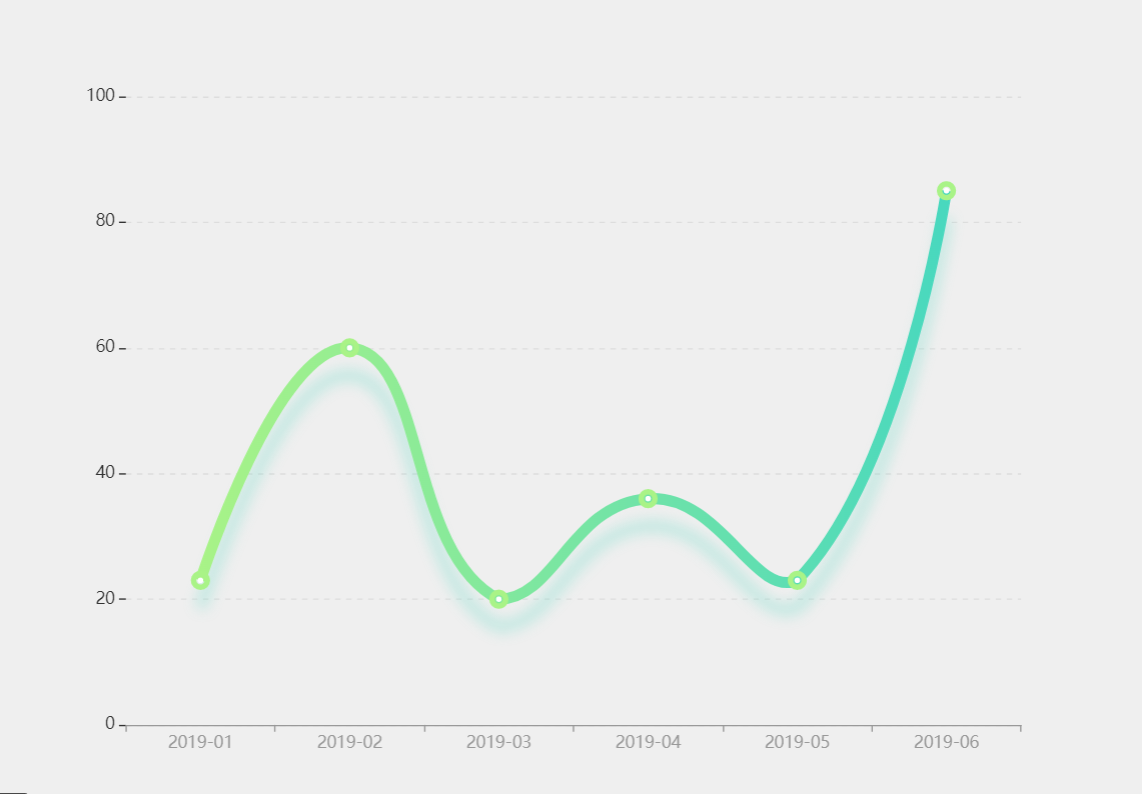
每当用户登录时，将登录状态和活跃状态同时修改为1；在一小时内，退出登录只修改登录状态，不修改活跃状态。同时考虑到并不是所有用户都会正常退出登录，所以需要监听登录状态的Session信息，当Session会话失效时触发事件，将登录状态改为0。

使用Spark Streaming中窗口间隔和滑动间隔都为1小时的窗口函数累加每小时的数据，统计出每天24小时活跃人群信息。具体流程见下图：









24小时活跃人群折线图