优惠券到期提醒定时、用户地址管理需求

1. 优惠券到期提醒定时

项目背景：

即将到期的优惠券提醒用户来使用；

项目说明：

APP、微信、短信推送优惠券即将到期信息；

1、即将过期：即分别距离有效期倒数6天/4天；

2、距离有效期倒数6天的发送短信提醒，距离有效期倒数4天的发送微信和APP提醒；

3、目前有2种优惠券：红包+加息券，有效期计算方式，分别举例如下

（1） 距离有效期倒数6天：

提醒日=到期日-6天

举例：优惠券到期日是2018-05-18，那么提醒日则是到期日倒数6天，即2018-05-13；

（2） 距离有效期倒数4天：

提醒日=到期日-4天

举例：优惠券到期日是2018-05-18，那么提醒日则是到期日倒数4天，即2018-05-15；

项目设计：

（1）数据查询

优惠券目前分为红包、加息券两类

红包查询，查询未使用的红包数据count\_red

加息券查，查询未使用的加息券数据count\_ticket

未使用的优惠券数量总和 = count\_red + count\_ticket

举例：定时运行的时间2018-05-13 10:00:00，未使用的优惠券数据，查询使用有效期结束时间use\_time\_end [ 2018-05-18 00:00:00, 2018-05-18 23:59:59 ]，状态status=INIT( 未使用 )的优惠券数据，根据查询到的手机号发送短信；

查询优惠券（加息券+红包）数量：

SELECT cc.user\_id, cc.user\_id, SUM(cc.ticket\_count) as ticket\_count

FROM

(

SELECT \* FROM

( SELECT

a.user\_id, b.mobile, COUNT(\*) AS ticket\_count

FROM bs\_interest\_ticket\_info a, bs\_user b

WHERE a.status = 'INIT'

AND DATE(a.use\_time\_end) = '2018-05-18'

AND a.user\_id = b.id AND b.`status` = 1

GROUP BY a.user\_id ) aa

UNION ALL

SELECT \* FROM

( SELECT

a.user\_id, b.mobile, COUNT(\*) AS ticket\_count

FROM bs\_red\_packet\_info a, bs\_user b

WHERE a.status = 'INIT'

AND DATE(a.use\_time\_end) = '2018-05-18'

AND a.user\_id = b.id AND b.`status` = 1

GROUP BY a.user\_id ) bb

) cc

GROUP BY cc.user\_id

1. 发送短信

根据统计出来手机号，发送短信，短信发送服务新增方法，新增短信发送模板TemplateKey；短信记录流水表bs\_sms\_record\_jnl写入记录；

短信模板：

短信：

【币港湾】尊敬的用户，您有X个优惠券将于5天后过期（yyyy-mm-dd为最后可用期限），请尽快使 用。

1. 推送app消息

TicketNotifyTask 卡券信息通知用户定时任务

原有卡券信息通知用户定时任务已经推送过的app消息是否要重复发送？（消息通知状态 msg\_status = 'FINISHED '）

继续发送

推送的优惠券是否筛选时有效期开始时间use\_time\_start 至 有效期结束时间use\_time\_end时间间隔较大的优惠券，发送短信、app微信推送信息？

避免信息频繁发送？

举例：红包、优惠券的使用期限只有5~6天；

**加息券发送的时间间隔不考虑，已跟产品确认；**

APP消息通知表 bs\_app\_message写入记录；

APP用户推送记录表 bs\_app\_pushed\_message 可以批量插入数据；

加息券信息表 bs\_interest\_ticket\_info更新消息通知状态 msg\_status =FINISHED 已发送，update\_time；

举例：定时运行的时间2018-05-13 10:00:00，未使用的优惠券数据，查询使用有效期结束时间use\_time\_end [ 2018-05-16 00:00:00, 2018-05-16 23:59:59 ]，状态status=INIT( 未使用 )的优惠券数据，APP、微信推送信息；

查询优惠券（加息券+红包）数量：

SELECT cc.user\_id, cc.user\_id, SUM(cc.ticket\_count) as ticket\_count

FROM

(

SELECT \* FROM

( SELECT

a.user\_id, b.mobile, COUNT(\*) AS ticket\_count

FROM bs\_interest\_ticket\_info a, bs\_user b

WHERE a.status = 'INIT'

AND DATE(a.use\_time\_end) = '2018-05-16'

AND a.user\_id = b.id AND b.`status` = 1

GROUP BY a.user\_id ) aa

UNION ALL

SELECT \* FROM

( SELECT

a.user\_id, b.mobile, COUNT(\*) AS ticket\_count

FROM bs\_red\_packet\_info a, bs\_user b

WHERE a.status = 'INIT'

AND DATE(a.use\_time\_end) = '2018-05-16'

AND a.user\_id = b.id AND b.`status` = 1

GROUP BY a.user\_id ) bb

) cc

GROUP BY cc.user\_id;

App推送模板：

标题：优惠券过期通知

内容：您有x个优惠券将于3天后过期（yyyy-mm-dd为最后可用期限），请尽快使用。

1. 推送微信消息

数据查询方式同app推送服务的查询；

SendWechatService 微信通知模板新增服务；

推送模板是否要提前申请。。。？

微信模板：

业务通知

标题：券过期提醒

内容：尊敬的用户，您有x个优惠券将于3天后过期（yyyy-mm-dd为最后可用期限），请尽快使用。

1. 定时脚本

TicketReminderNotifyTask

-- 优惠券到期提醒定时，每天10点运行

INSERT INTO `bs\_schedule\_config` (`job\_name`, `job\_group`, `job\_status`, `cron\_expression`, `bean\_class`, `method\_name`, `bean\_id`, `is\_concurrent`, `description`, `create\_time`, `update\_time`) VALUES ('TicketReminderNotifyTaskTime', 'SCHEDULE', 'STOP', '0 0 10 \* \* ?', 'com.pinting.business.dayend.task.TicketReminderNotifyTask', 'ticketReminderNotify', 'TicketReminderNotifyTask', 'NO', '优惠券到期提醒定时', NOW(), NOW());

2、产品详情页添加说明提示

项目背景：

日常优化，产品详情页文案修改；

项目说明：

项目设计：

改动涉及的端币港湾h5，报业渠道、七店h5；币港湾app，钱报app；

定期类产品、分期类产品均需要添加；

产品详情页

micro2.0/regular/regular\_product\_index.vm

H5钱报（报业端）

qianbao178/regular/regular\_product\_index.vm

币港湾APP

micro2.0/regular/regular\_product\_index\_app.vm

钱报APP

qianbao178/app/regular/regular\_product\_index.vm



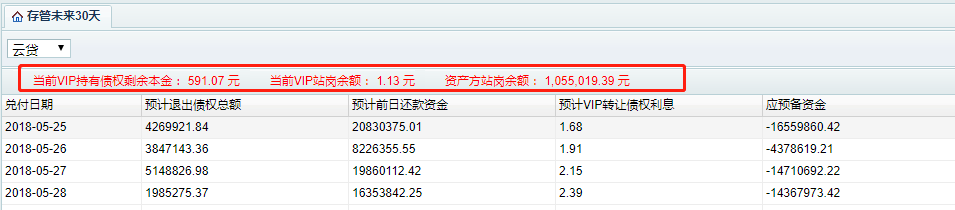
1. 财务管理-存管未来30天兑付查询

项目背景：

日常优化，产品详情页文案修改；

项目说明：

红框中金额统计在手机端查看时，显示不全，页面样式需要调整；



项目设计：

depCash30/index.vm

4、用户地址管理

项目背景：

用户地址管理由于活动逐步增加实物型的奖品，所以客服在回访的过程中需要询问用户的邮寄地址，从长期考虑现需增加一个“客户地址添加存档查询”的功能，以便更方便快捷的联系用户，提升工作效率项目说明：

项目说明：

项目设计：

新加用户地址管理表 bs\_user\_address\_info

-- ----------------------------

-- Table structure for bs\_user\_address\_info

-- ----------------------------

DROP TABLE IF EXISTS `bs\_user\_address\_info`;

CREATE TABLE `bs\_user\_address\_info` (

`id` int(11) NOT NULL AUTO\_INCREMENT COMMENT '编号',

`user\_id` int(11) DEFAULT NULL COMMENT '用户id',

`consignee` varchar(64) DEFAULT NULL COMMENT '收货人',

`consignee\_mobile` varchar(20) DEFAULT NULL COMMENT '收货手机号',

`consignee\_address` varchar(512) DEFAULT NULL COMMENT '收货地址',

`is\_default` varchar(20) DEFAULT NULL COMMENT '是否是默认地址：YES-是 NO-否',

`note` varchar(64) DEFAULT NULL COMMENT '备注',

`create\_time` datetime DEFAULT NULL COMMENT '创建时间',

`update\_time` datetime DEFAULT NULL COMMENT '更新时间',

PRIMARY KEY (`id`),

KEY `FK\_Reference\_215` (`user\_id`),

CONSTRAINT `FK\_Reference\_215` FOREIGN KEY (`user\_id`) REFERENCES `bs\_user` (`id`)

) ENGINE=InnoDB DEFAULT CHARSET=utf8 COMMENT='用户地址管理表';

1. 列表数据

用户地址管理，查询数据的列表只显示默认地址对应的那条记录的数据，支持姓名、手机号模糊查询；

SELECT b.user\_name, b.mobile, a.consignee\_address, a.update\_time

FROM bs\_user\_address\_info a, bs\_user b

WHERE a.user\_id = b.id AND a.is\_default = 'YES'

GROUP BY a.user\_id

ORDER BY a.update\_time DESC

LIMIT 0,20;

1. 批量导入

（1）用户地址管理导入模板下载；

（2）批量导入，读取excel模板中的数据，校验读取到excel数据是否为空；空 返回提示；

（3）有数据，校验注册手机号、姓名，与bs\_user表中已经存在的用户数据是否一致；

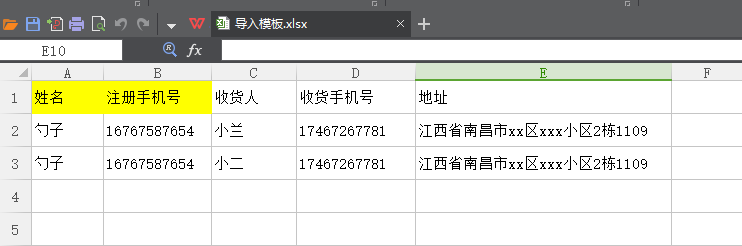
（4）如录入的数据有重复数据，自动去重处理，数据唯一性校验：姓名、注册手机号、收货人、收货手机号、收货地址5个条件一起判断；

举例1：excel导入100个地址信息，有5个地址信息姓名、注册手机号在bs\_user表中不存在，

提示信息：5个用户不存在或用户手机号与姓名不一致，请检查后重新导入；

举例2：excel导入100个地址信息姓名、注册手机号与bs\_user中一致，自动去重，

提示信息：导入成功；



3、新增

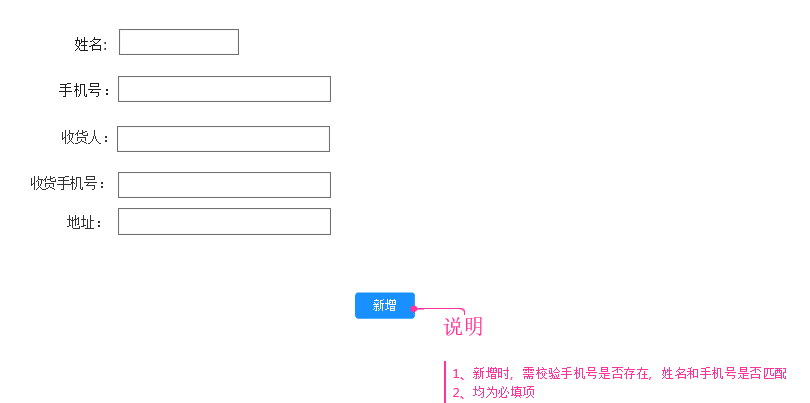
（1）非空校验，均为必填项；

（2）校验姓名和注册手机号是否与bs\_user中数据匹配；

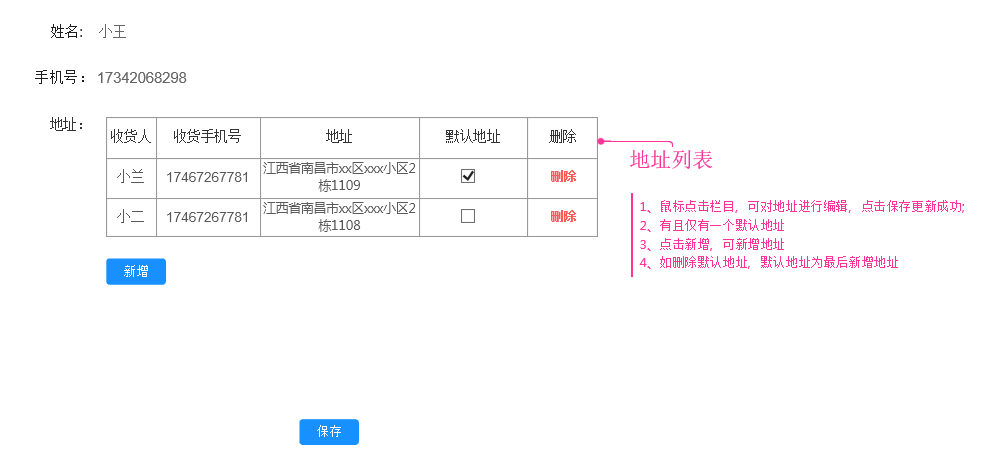
（3）数据唯一性校验：姓名、注册手机号、收货人、收货手机号、收货地址5个条件一起判断；

（4）新增的地址为默认地址，更新库里面旧的默认地址；

（5）提示信息开发时再与产品确认；



1. 查看编辑
2. 所有的地址均可以删除，包含默认地址，默认地址有且只有一个；所有地址都删除，即改注册手机号在库里面的所有地址都删除；
3. 收货人、收货手机号、地址、默认地址均可编辑；
4. 新增，table下新增一列，新增提交时添加非空校验；
5. 如删除默认地址，默认地址为最后新增地址；
6. 提示信息开发时再与产品确认；



5、列表删除按钮

根据注册手机号删除所有地址信息；

删除提示：

