生活Plus V.3.12.5

<ID1014241前端发放代金券时，段时间（10秒）内后台没有返回成功数据，会提示发放失败，需对接口进行优化>

设计文档

*<2021.01.28>*

**文档历史**

|  |  |  |  |
| --- | --- | --- | --- |
| **版本** | **日期** | **作者** | **内容** |
| 1.0 | 20210128 | 吴扬帆 | 初始版本 |

**目录**

[1 需求分析 4](#_Toc10885)

[1.1 背景信息 4](#_Toc26048)

[1.2 原始需求 4](#_Toc21416)

[1.3 需求分析 4](#_Toc32280)

[2 解决方案 4](#_Toc27007)

[3 详细设计 5](#_Toc20882)

[3.1 用户场景（User Case） 5](#_Toc6961)

[3.2 流程 5](#_Toc30400)

[3.3 模块 6](#_Toc16638)

[3.4 数据结构 6](#_Toc30960)

[3.4.1 MySQL 6](#_Toc17612)

[3.4.2 Redis 7](#_Toc7956)

[3.4.3 MQ 7](#_Toc19417)

[3.5 配置 7](#_Toc29474)

[3.6 性能 7](#_Toc4910)

[3.7 可扩展性 7](#_Toc21826)

[4 遗留问题 7](#_Toc15503)

[4.1 主要缺点 7](#_Toc377)

[4.2 问题 7](#_Toc30711)

[附录A. 参考资料 7](#_Toc25533)

[附录B. 功能测试报告 8](#_Toc5598)

# 需求分析

## 背景信息

【行家端、后端】前端发放代金券时，段时间（10秒）内后台没有返回成功数据，会提示发放失败，需对接口进行优化。

现有代金券批量发放接口返回较慢，超时后无数据返回，发放过程中遇到错误就会终止后续用户发放，用户体验较差。

## 原始需求

【ID1014241】【行家端、后端】前端发放代金券时，段时间（10秒）内后台没有返回成功数据，会提示发放失败，需对接口进行优化。

需求地址：

<https://www.tapd.cn/53440148/prong/stories/view/1153440148001014241?url_cache_key=eb976a2d416699f23f2239f98bf68047&action_entry_type=story_tree_list>

用户批量发放代金券后可查看发放进度以及发放失败信息，遇到错误记录错误信息跳过不影响后续用户发放。

## 需求分析

对于有大量数据代金券需要发放时，原来采用同步的方法，当后端收到发放代金券的请求后，需要一条条数据去处理，如果数据量太大，很有可能会超过10s。这个时候hangjia是收不到反馈的。

如果采用异步方式去做，这个问题可以解决。当后端收到请求后，告诉行家该请求正在处理，同时行家可以去查询后台处理该请求的状态。

# 解决方案

行家用户点击批量发放代金券，直接返回提示“代金券正在发放”。然后用户可在代金券发放记录中查询代金券发放进度。发放完成后可导出发放失败记录。

我们可以把这个请求作为一个task，创建一个表来存储这些task。task的重要信息有状态，已经处理了多少条，总共多少条，错误多少条。这样行家可以要想知道当前状态，直接从该表查。

当后端收到请求后，一边处理，以便更新上面的task状态。

同时，我们要考虑到异常处理，如果一张表里有几十条，上百条记录发放不成功，我们需要有一张表专门来保存这些错误记录。该表至少包括：task id（用来下载错误列表），发放的代金券id，发给谁的，错误信息等。

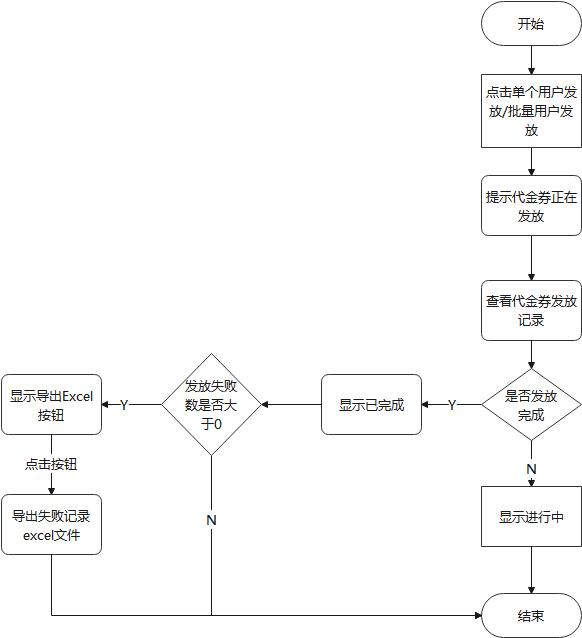
这样，后端至少需要提供2个API，一个是请求API，一个是查询处理状态的API。

当然，也可以hangjia直接根据task id去数据库查询状态。

# 详细设计

## 用户场景（User Case）

## 流程



1. 用户点击单个用户发放/批量用户发放按钮
2. 填完信息后点击代金券发放按钮
3. 立即提示代金券正在发放
4. 用户查看代金券发放记录
5. 如果发放完成显示已完成，并显示发放结果，发放总数/成功数/失败数。
6. 如果失败数大于0显示到处excel按钮。
7. 点击按钮导出代金券发放失败的用户手机号以及原因。

## 模块

FHGL-USER

## 数据结构

### MySQL

用户代金券发放记录表

* **tb\_voucher\_send\_record**

|  |  |  |  |
| --- | --- | --- | --- |
| **字段名称** | **数据类型** | **是否为NULL** | **描述** |
| id | int(11) | N | Id |
| voucherId | int(11) | Y | 代金券id |
| sendTime | datetime | N | 发放时间 |
| sendType | tinyint(1) | Y | 1:单个发放，2:批量发放 |
| sendCount | int(11) | Y | 发放总数 |
| successCount | int(11) | Y | 成功个数 |
| failCount | int(11) | Y | 失败个数 |
| sendState | tinyint(1) | Y | 0:进行中，1:已完成 |

代金券发放失败记录表

* **tb\_voucher\_send\_fail\_record**

|  |  |  |  |
| --- | --- | --- | --- |
| **字段名称** | **数据类型** | **是否为NULL** | **描述** |
| id | int(11) | N | Id |
| voucherSendId | int(11) | N | 用户代金卷发放记录id |
| phone | varchar(50) | Y | 发放失败手机号 |
| failReason | varchar(255) | Y | 失败原因 |
| createTime | datetime | N | 创建时间 |

### Redis

### MQ

## 配置

## 性能

## 可扩展性

# 遗留问题

## 主要缺点

## 问题

# 附录A. 参考资料

|  |  |
| --- | --- |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

# 附录B. 功能测试报告

该部分包括本文档需求的功能测试报告:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Test Case** | **用例目的** | **输入** | **输出** | **期望结果** | **测试结果** | **说明** |
| Case 1 |  |  |  |  |  |  |
| Case 2 |  |  |  |  |  |  |