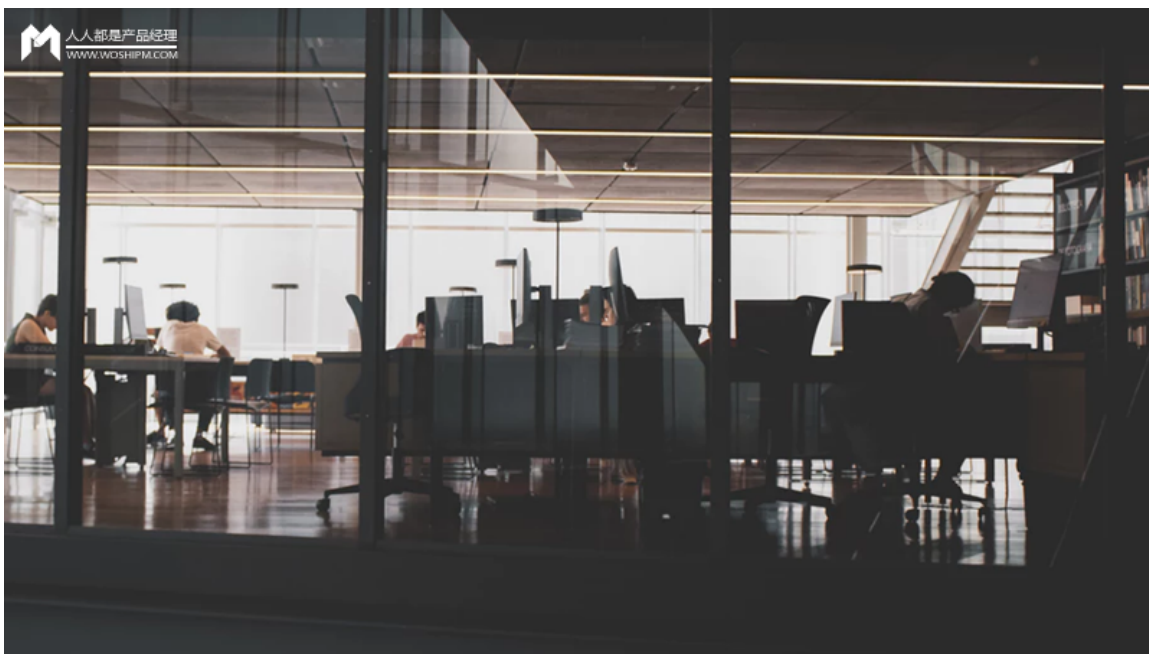


一个好的收银系统需要满足易操作、快速收银、支付体验佳的效果，本文将为大家分享一些减少支付异常率的小技巧。



零售行业发展至今，无论是大型还是小型商户，线下商超还是连锁店铺，高频低额还是低频高额的交易场景，一个好的 ERP 收银系统是整个交易环节不可或缺的元素之一，易操作、快速收银、支付体验佳等均是衡量一个收银系统是否为一个好产品的标准。

而在支付过程中，难免遇到一些系统或操作导致的异常从而导致交易阻塞甚至是客诉，那么尽量避免系统发生异常或发生异常时的解决策略，也是我们在设计产品时需要考虑的。

如何设计一个收银产品这里就不多说了，感兴趣的同学可以搜索一下相关资料，下面和大家简单讨论下如何通过一些小技巧减少支付异常率。

目前线下收银系统多采用 CS 设计模式——即客户端/服务端分离，使用时需安装独立客户端，交易过程中和第三方支付系统交互的多为商户系统的服务端，我们可分别在客户端/服务端做改动从而实现我们的需求。

一、支付前校验/支付后判重

支付前校验：在创建完待付款订单后，服务端调用支付 API 前，对该订单进行校验，是否为待支付订单。若是待支付订单，则继续进行支付，若否，则直接对该订单进行处理。

支付后判重：商户系统在收到支付结果后，同样根据业务参数对此订单进行判断是否为重复订单或异常订单而做出对应处理。

做好支付前校验和支付后判重，能有效解决同比订单重复支付、重复生效等相关异常问题。同样在创建订单或订单支付成功生效后也可以做校验订单状态的动作从而达到更好的过滤效果。

二、部分系统参数配置化

API / 支付API / 统一收单交易支付接口 / 公共请求参数

参数	类型	是否必填	最大长度	描述	示例值
app_id	String	是	32	支付平台侧给开发者的应用ID	2014072300007148
method	String	是	128	接口名称	allpay.trade.pay
format	String	否	40	仅支持JSON	JSON
charset	String	是	10	请求使用的编码格式。如utf-8,gbk,gb2312等	utf-8
sign_type	String	是	10	商户生成签名字符串所使用的签名算法类型。目前支持RSA2和RSA，推荐使用RSA2	RSA2
sign	String	是	344	商户请求参数的签名串，详见 签名	详见示例
timestamp	String	是	19	发送请求的时间，格式“yyyy-MM-dd HH:mm:ss”	2014-07-24 03:07:50
version	String	是	3	调用的接口版本，固定为：1.0	1.0
notify_url	String	否	256	支付宝服务窗主动通知商户服务器指定的页面http/https路径。	http://api.test.allpay.net/etinterface/receive_notify.htm
app_auth_token	String	否	40	详见 应用授权概述	
biz_content	String	是		请求参数的集合，最大长度不限，除公共参数外所有请求参数都必须放在这个	

三、提供主动撤销的入口

目前业内第三方支付服务商在在交易时返回支付结果多采用主动通知或提供查询服务让商户系统自行查询，但即使商户系统同时接入这两种服务或做了轮询和异常场景主动撤销的机制，仍存在多种原因导致无法获取支付结果，如网络异常，接口异常，参数异常等，从而造成用户扣钱而订单未成功引起不必要的纠纷。

遇到此种场景不要慌张，多数支付系统会提供撤销的服务，我们只需在商户系统中加入可以主动撤销的入口，手动撤销此订单，便可很轻松解决此问题，但入口和操作权限控制也需深思。

API / 支付API / 统一收单交易支付接口 / 公共请求参数

参数	类型	是否必填	最大长度	描述	示例值
app_id	String	是	32	支付三分割端开发者的应用ID	2014072300007148
method	String	是	128	接口名称	alipay.trade.pay
format	String	否	40	仅支持JSON	JSON
charset	String	是	10	请求使用的编码格式。如 utf-8,gbk,gb2312 等	utf-8
sign_type	String	是	10	商户生成签名字符串所使用的签名算法类型。目前支持RSA2和RSA，推荐使用RSA2	RSA2
sign	String	是	344	商户请求参数的签名串，详见 签名	详见示例
timestamp	String	是	19	发送请求的时间，格式“yyyy-MM-dd HH:mm:ss”	2014-07-24 03:07:50
version	String	是	3	调用的接口版本，固定为：1.0	1.0
notify_url	String	否	256	支付宝服务端通知商户服务器指定的页面http/https路径。	http://api.test.alipay.net/v1interface/receive_notify.htm
app_auth_token	String	否	40	详见 应用授权概述	
biz_content	String	是		请求参数的集合，最大长度不限，除公共参数外所有请求参数都必须放在这个	

四、提供手工再次发起查询的入口

上文提到要留主动撤销的入口以主动解决异常订单，但对于商户来说损失一笔订单毕竟不是一个上等决策，幸好大多数支付系统会提供查询服务供商户系统查询订单状态。

因此我们在收银系统中可预留一个再次发起查询订单状态的入口及时挽回损失，但此场景仅适用于支付成功、商户未获取到支付结果而导致的支付异常场景。

五、要留再次发起退款的入口

商户系统在退款时也同样会存在退款失败的问题，如商户系统退款完成而并未调用支付系统的退款服务或调用退款服务时失败，且有些第三方服务商也不会同步返回退款结果，从而造成在商户系统中退款成功而实际并未给用户退钱。

若此时加入可以再次发起退款的入口直接调用退款服务完成退款，可有效解决部分退款异常问题。

以上观点仅供参考，小伙伴们在运用时一定要结合实际业务场景进行使用，切勿照猫画虎，有些异常可通过定时任务解决的就不用开放手动处理的入口了，而有些功能一定要注意权限的控制。

用好了能提升产品的使用体验，降低异常发生的频率，用不好会引起大问题！

欢迎各位评论区留言讨论！

题图来自 Unsplash，基于 CC0 协议