

Payment callback failures

- [Overview SaaS](#)
- [Callbacks management](#)

Overview SaaS

Payment callbacks are notifications sent by [online payment gateways](#) that contain information about the transaction outcome. The platform receives these notifications and depending on the outcome of the transaction either converts temporary order into open order or cancels it. Without a clear response contained in the callback the platform cannot make a decision.

In some cases callbacks can fail during processing. The platform retains all callbacks in their original form thus if failures occur these callbacks are retained in the "unprocessed callbacks" queue.

If the queue is not empty dashboard widget will alert the shop managers to check the "Unprocessed callbacks" queue. Shop managers can review all callbacks that have failed and either: replay them (a retry), replay them forcefully (accept the callback) or delete them.

Callbacks management

Callbacks management section can be accessed via **Operations** section or via quick link on **Unprocessed Callbacks** widget.

Unprocessed callbacks information can be extracted in CSV format for review or can be reviewed one by one on callbacks panel directly. The panel shows all callbacks that have failed. Shop manager can either remove the callback if it is deemed as fake or invalid or can attempt to re-run the callback. If re-try is successful then the order relating to this transaction will be put into processing state. If however re-try fails shop manager has the option to disable validation check. This is the check which is made to the payment gateway to verify the call, which in some cases although being genuine does not pass the validation (e.g. SSL certificate problems, wrong hash, encoding problems). This mode may allow the order to progress. In case of a critical failure the payment callback stays in unprocessed form.

Payment gateway callbacks are never deleted, they are simply marked as: invalid or processed. So even if shop manager accidentally removes the callback it is still possible to recover it from DB by changing its state.

DEMO ENVIRONMENT

Logged in as Yes

Dashboard

Attention

2019-12-04 17:25:02: Node message failure [BasicMessageImpl[source='EDEMO.JAM', targets=[EDEMO.SF0, EDEMO.SF1], subject='AlertDirector.getAlerts', payload=null]] to channel [EDEMO.SF1:http://localhost:8083/api/services/connector] . Exception occurred during ws call

New orders 0
This week 0
This month 0

Manage orders

4
Waiting approval

Manage orders

New customers 0
This week 0
This month 0

Manage customers

Page size: 10
Filter max: 50
No filter max: 100
Language: en

UI settings

0 (+0) %
276 \$
0 %

External PIM

postFinancePaym... 1

Manage PG callbacks

Unprocessed callbacks can also be accessed from the operations section of menu

Unprocessed callbacks widget shows counts of failures by PG

DEMO ENVIRONMENT

Logged in as Yes

Unprocessed Callbacks

Shop code	Payment gateway code	Timestamp
SHOP10	postFinancePaymentGatewayLabel	2019-07-15 21:52:06

Review or delete?

delete

review

Unprocessed Callbacks

Timestamp: 2019-07-15 21:52:06

Shop code: SHOP10

Payment gateway code: postFinancePaymentGatewayLabel

Result: Failure 13

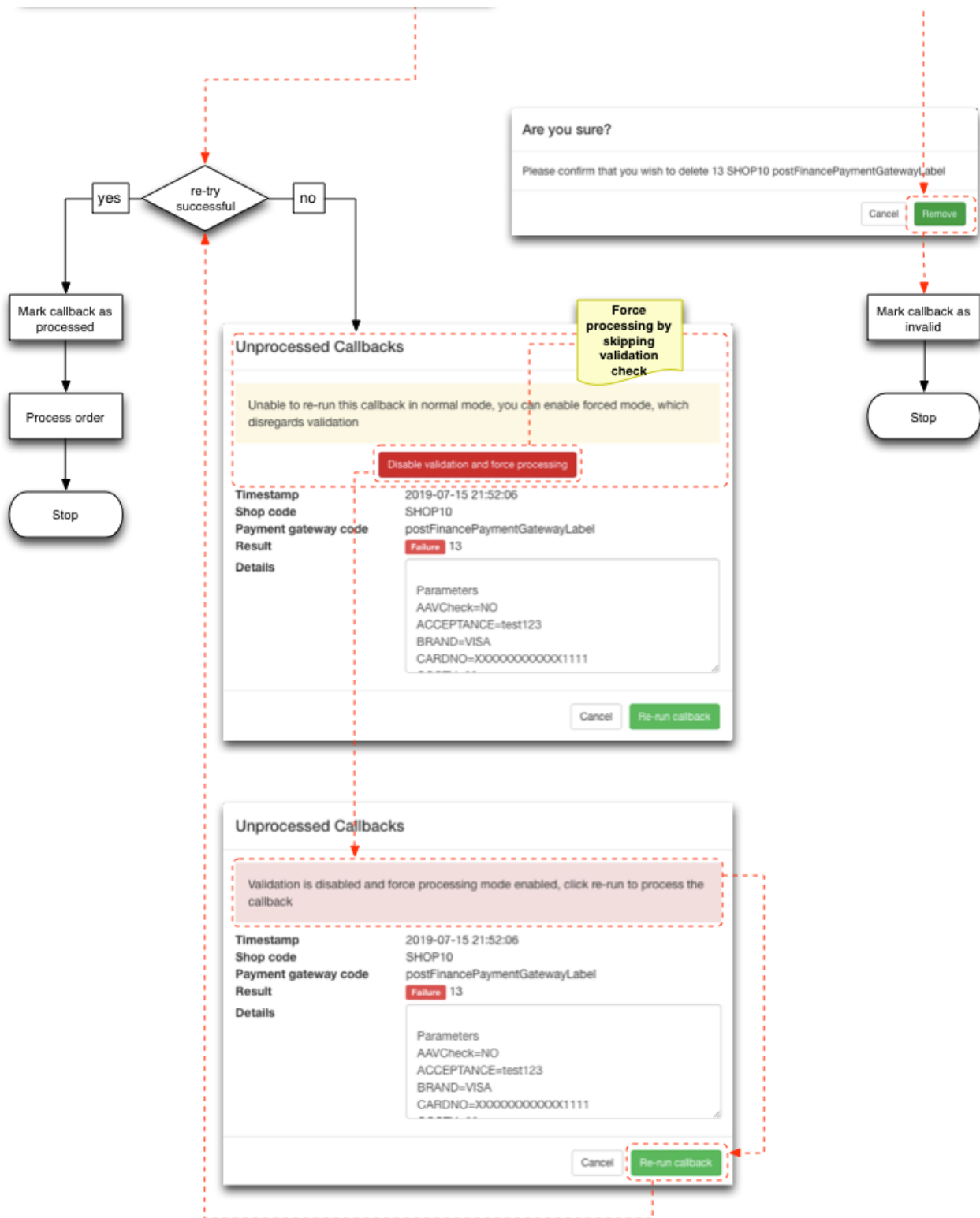
Details:

```
Parameters
AAVCheck=NO
ACCEPTANCE=test123
BRAND=VISA
CARDNO=XXXXXXXXXXXX1111
```

Cancel Re-run callback

Callback information can be reviewed and "re-run" can be attempted

In case if callback is found to be invalid it can be removed (i.e. no processing)



Untitled

about:blank

```
paymentGatewayCallbackId,shopCode,processed,label,requestDump,createdTimestamp,forceProcessing,
13,SHOP10,false,postFinancePaymentGatewayLabel,
Parameters
AAVCheck=NO
ACCEPTANCE=test123
BRAND=VISA
CARDNO=XXXXXXXXXXXX1111
CCCTV=99
```

Callback data can
also be exported as
CSV

```
CN=
CVCCheck=NO
ECI=92
ED=0125
IP=80.5.131.173
IPCTY=GB
NCERROR=0
PAYID=3051768755
PM=CreditCard
SHASIGN=96F89F1AB68EE4DEC4DB581A2669DC4AC74415BA
STATUS=9
TRXDATE=07/15/19
VC=NO
amount=4121.52
currency=EUR
orderId=190715205110-5

Attributes
CharacterEncodingFilter.FILTERED=true
ShoppingCart=org.yes.cart.shoppingcart.impl.ShoppingCartImpl@23eae65
org.springframework.web.context.request.RequestContextListener.REQUEST_ATTRIBUTES=org.apache.catalina.connector.RequestFacade@66e86d
dc
org.springframework.web.context.request.async.WebAsyncManager.WEB_ASYNC_MANAGER=org.springframework.web.context.request.async.WebA
syncManager@62d7e55e

Headers
connection=Keep-Alive
content-length=329
content-type=application/x-www-form-urlencoded
host=edemo.yes-cart.org
user-agent=Mozilla/3.0 (compatible; Indy Library)
,1563223926000,false,
```