

TECHNICAL SPECIFICATIONS

Web Service

yellow rewards program support



Euronet Merchant Services Payment Institution Single Member S.A.
1 Sachtouri & Poseidonos Ave., 176 74 Kallithea, Athens, Greece
Authorised as a Payment Institution by the Bank of Greece under Law 4537/2018

www.epayworldwide.gr
Tel.: +30 210 38 98 954



History of Changes

Date	Version	Changes
17/11/2017	1.0	Original version
22/11/2021	1.0.1	<ul style="list-style-type: none">▪ Section 6: Update in order to support 3D Secure version 2.▪ Section 7: The value of the MerchantReference parameter has now to be unique / different at each call to the Transaction Web Service▪ Section 7: Addition of new parameters to the request message of the Transaction Web Service:<ul style="list-style-type: none"><u>Request Message:</u><ul style="list-style-type: none">▪ Protocol▪ DsTransID▪ RecurringInd▪ TraceID<u>Response Message:</u><ul style="list-style-type: none">▪ TraceID▪ Section 9: Update of the test cases for the Transaction Web Service with new test cards▪ Section 11: Update with Visa Secure and Mastercard IdentityCheck logos (3D Secure v2)▪ Section 13: Update of the Implementation Checklist
16/03/2022	2.0	Service rebranding to epay eCommerce



Contents

1.	Introduction	2
2.	General Architecture	4
3.	Details for the Creation of a Test Account	5
4.	LoyaltyMemberBalance Service	6
5.	Introduction of Information for Redemption of yellows	9
6.	Strong Customer Authentication ("3D Secure")	10
7.	Transaction Web Service	12
8.	Merchant Application Action Flow	31
9.	Test Cases	35
10.	Security Requirements	61
11.	Use of Icons	62
12.	Tips	64
13.	Implementation Checklist	66
	Annex 1	68
	Annex 2	74
	Glossary	76



1. Introduction

Piraeus Bank yellow Loyalty Program rewards Bank customers who enroll and become yellow Members with yellows, for the acquisition and use of certain products and services offered by the Bank. Customers can redeem yellows by making purchases at Program Partners. This document describes the technical implementation that shall be carried out by businesses that will support on-line redemption and earning of yellows through card transactions on Euronet Merchant Services e-payment system epay eCommerce.

The process of redemption/earning yellows and of any card debit is carried out through a call to the proper SOAP Web Services described in the following sections. It shall be highlighted that the redemption and/or earning yellows takes place immediately, upon the call to the Web Services.

In the following sections, a detailed description is given for all the steps required for transactions to be made with the ability of redemption of yellows:

- **Section 2 → General Architecture:**
Description of the general architecture of the service
- **Section 3 → Details for the Creation of a Test Account:**
The details required to be sent to Euronet Merchant Services for the creation of a test account in order for test transactions to be carried out.
- **Section 4 → LoyaltyMemberBalance Service**
Description of the LoyaltyMemberBalance Service through which the business is informed on whether the card holder is a member of the yellow program.
- **Section 5 → Introduction of Information for Redemption of yellows**
Description of the screen that will be displayed and the information that will be requested by the user when the card holder is a member of the yellow program and there are yellows available for redemption.
- **Section 6 → Strong Customer Authentication ("3D Secure"):**
Reference to the Strong Customer Authentication to be performed as part of each online transaction through a website carried out by the card holder..
- **Section 7 → Transaction Web Service:**
Description of the "Transaction Web Service" used for the details of a transaction to be sent to epay eCommerce. Through the same call, real time redemption and earning of yellows is made.
- **Section 8 → Flow of Actions of the Business's Application:**
Presentation in the form of a diagram of the algorithm that has to be implemented by the business's application in order for a transaction to be executed and its result to be checked.
- **Section 9 → Test Cases:**
Description of the test cases that must be implemented within the test calls to the LoyaltyMemberBalance Service and the Transaction Web Service.
- **Section 10 → Security Requirements:**

Description of the security requirements

- **Section 11 → Use of Icons:**

Material regarding the mandatory and optional icons that must be posted on the business's website is included.

- **Section 12 → Tips:**

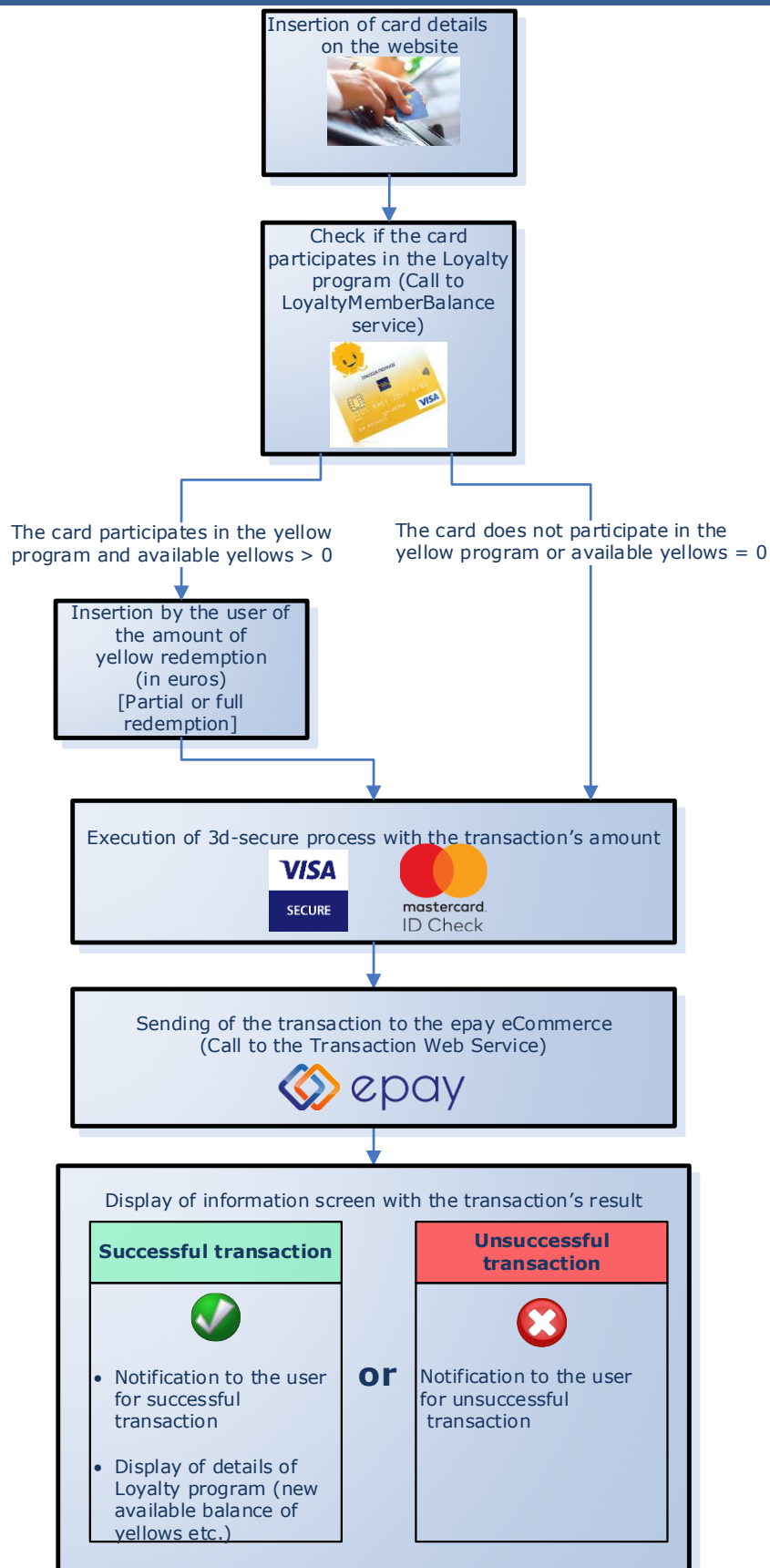
Tips and remarks regarding various points that must be taken into account.

- **Section 13 → Implementation Checklist:**

List of the actions that must be implemented by the technical manager in order for the collaboration with the business to be completed.



2. General Architecture





3. Details for the Creation of a Test Account

Information that must be sent to Euronet Merchant Services, so that the necessary technical information (test account) is provided for test transactions to be made, is as follows (all mandatory):

- **Details of the technical manager:**
 - Name and surname of the technical manager
 - Contact number of the technical manager
 - Email address of the technical manager
 - Company to which the technical manager belongs
- **Details of the business to which the system belongs:**
 - Distinctive title of the business to which the system belongs
 - Tax Registration Number of the business to which the system belongs
 - Domain name of the live web site of the business (provided that the transactions will be made through a website)
- **Technical details:**
 - **IP address:** The IP address of the server from which the "Transaction Web Service" of Euronet Merchant Services will be called.
 - **Support of installments:** It is declared whether installments will be used or not in the test transactions.

The details of the test account given by Euronet Merchant Services, provided that the above information is sent, are the following, which are used both upon implementation of the 3D Secure process (for eCommerce transactions – call to the "Wrapper3DSecure service") and upon sending the transaction (call to the "LoyaltyMemberBalance service" and of the "Transaction Web Service"):

- AcquirerID
- MerchantID
- PosID
- User
- Password
- ChannelType

Information on the usefulness of these details is included in the following sections.



4. LoyaltyMemberBalance Service

The LoyaltyMemberBalance service is a SOAP Web Service through which it is checked whether a card participates in the yellow rewards program and, if so, information is returned regarding the yellows available.

The URL is the following:



<https://paycenter.piraeusbank.gr/Services/LoyaltyMemberBalanceService.asmx>




Attention!


- The response timeout is 60 sec.
- The Web Service call shall be made through a Server. No scrip-based cross-origin HTTP requests are allowed.

The information required for the call to the "LoyaltyMemberBalance service" is the following:

REQUEST PARAMETERS		
Parameter name	Description	Type
MerchantId	Merchant code (merchant id). Provided by Euronet Merchant Services.	Integer
Username	Username. Provided by Euronet Merchant Services.	String
Password	Password <u>encrypted with the MD5 hashing algorithm</u> . Provided by Euronet Merchant Services (in its non-encrypted form).	String
CardNumber	The customer's card number or the token corresponding to the card and acquired through the "Tokenization" mechanism.	String

The parameters sent in the response, are the following:

RESPONSE PARAMETERS		
Parameter name	Description	Type
SupportReferenceID	Request reference id. It has a different value for each request.  Note: It is necessary to save its value in order, if needed, to be used as a reference point upon communication with Euronet Merchant Services.	Long integer

StatusFlag	<p>Declares whether there was a successful result. Possible values:</p> <ul style="list-style-type: none"> ▪ Success: The card check is successful and the value of the "MemberStatus" parameter will then have to be checked. ▪ Failure: An error was returned, information for which can be found in ResultCode and ResultDescription. 	String
ResultCode	<p>The request's result code that declares whether a technical issue emerged upon processing the transaction. More specifically:</p> <ul style="list-style-type: none"> ▪ Value equal to 0: There was no problem. <u>The "MemberStatus" parameter shall then be checked.</u> ▪ Value other than 0: There was a technical problem. The "ResultDescription" parameter contains a description of the problem that occurred. 	String
ResultDescription	<p>The description corresponding to the value of the "ResultCode" parameter.</p> <div>  Note: This information is not recommended to be shown to the user. </div>	String
MemberStatus	<p>The parameter value expresses whether the card participates in the yellow program. The possible values are as follows:</p> <ul style="list-style-type: none"> ▪ 0: The card holder does not participate in the yellow program. ▪ 1: The card <u>participates</u> in the yellow program. Values will be returned for the parameters regarding the available yellows (described below). ▪ 2: The process shall be continued without the possibility of redemption of yellows. (Values may be returned for the parameters regarding the available yellows, but the customer is not allowed to use them). ▪ 3: The process shall be continued without the possibility of redemption of yellows. 	String

	<ul style="list-style-type: none"> ▪ 4: The process shall be continued without the possibility of redemption of yellows. ▪ 5: The process shall be continued without the possibility of redemption of yellows. ▪ 6: The process shall be continued without the possibility of redemption of yellows. 	
AvailableAmount	If the card holder is a member of the yellow program, it contains the monetary value of yellows the card holder has, based on the current yellows to euros exchange rate.	Decimal
AvailableBalance	If the card holder is a member of the yellow program, it contains the total available yellows the customer has earned.	Integer
PartnerExchangeRate	If the card holder is a member of the yellow program, it contains the yellow/euro exchange rate applicable for the specific business.	Decimal
ProgramCode	For future use.	String
RedeemFloorLimit	For future use.	String



5. Introduction of Information for Redemption of yellows

If the below apply:

- The card participates in the yellow program (i.e., when MemberStatus = 1) and
- There are yellows available for redemption (i.e., AvailableBalance ≠ 0)

then the below steps follow:

- 1.** The available yellows and the corresponding amount in euros are displayed. Question to the customer whether they wish to redeem yellows.
 - 1.1.** If they do not want a redemption, step 2 follows (confirmation screen is displayed)
 - 1.2.** If they want a redemption, the customer shall enter the amount in euros that they wish to be redeemed and then step 2 follows (confirmation screen is displayed)
- 2.** A confirmation screen is displayed with the below information:
 - The total amount of the transaction
 - The amount with which the card is debited
 - If yellows are to be redeemed, the amount in euros that corresponds to the yellows to be redeemed is displayed
- 3.** After the user confirms the transaction, the process continues with the next step (implementation of 3d-secure process – see next section)

If the card does not participate in the yellow program (that is when MemberStatus ≠ 1) or the card participates, but has no yellows available for redemption, the process shall continue with the next step (implementation of 3d-secure process – see next section).



6. Strong Customer Authentication ("3D Secure")

Sale or preauthorization eCommerce transactions (see section 7 regarding the types of transactions) that are initiated by the card holder using a Visa, Mastercard or Maestro card, have to be preceded by strong customer authentication ("3D Secure" protocol, "Visa Secure" and "Mastercard Identity Check" by Visa and Mastercard respectively). The technical specifications of this process are included in a separate document.



Attention!

- Card holder authentication refers to Visa, Mastercard and Maestro card transactions.
- The amount, currency and MerchantReference used in the 3D Secure process (PurchAmount, Exponent, Currency, MerchantReference parameters of the "Wrapper3DSecure service") should match those to be used in the Transaction Web Service (Amount, CurrencyCode, MerchantReference parameters).
- The **MerchantReference** (reference code of the transaction originating in the merchant's system) should have a **unique/different value for each transaction**. This means that if a transaction fails and a new 3D Secure process is initiated for a new attempt, the MerchantReference (both in the Wrapper3DSecure service and in the Transaction Web Service) should have a different value (compared to the previous attempt).
- Only **sale** or **preauthorization** transactions are preceded by the authentication process. It is not applied in any other transactions (e.g. refunds, settlements, etc.) (For transaction types, see section 5.)
- In transactions carried out through systems where the holder provides their card details to a third person (e.g. a Call Center agent) no card holder authentication is required.

After the completion of the 3D Secure process, values are returned for the below parameters that shall be transferred to the corresponding fields in the Transaction Web Service call:

Parameter from Wrapper3DSecure	Parameter in the Transaction Web Service
Eci (*)	Eci
Cavv	Cavv
Xid	Xid
Protocol	Protocol
DsTransID	DsTransID

(*) If the Wrapper3DSecure service call does not return a value to ECI (e.g. in the case of a technical issue) and the merchant decides to send the transaction, the Transaction Web Service call should include the following default values in ECI:

- **In the case of a Visa card: ECI=07**
- **In the case of a Mastercard or a Maestro card: ECI=00**



7. Transaction Web Service

After the completion of the 3d-secure process, a call to the “Transaction Web service” follows. It is a SOAP Web Service through which all the following will be carried out **with one call only**:

- 1) Debit of the card through epay eCommerce
- 2) Redemption of yellows (if the user selected redemption)
- 3) Earning yellows (depending on the amount with which the card will be debited)

The URL is the following:




<https://paycenter.piraeusbank.gr/services/paymentgateway.asmx>



Attention!




- The response timeout is 60 sec.
- The Web Service call shall be made through a Server. **No scrip-based cross-origi**n HTTP requests are allowed.



The information required for the call to the “Transaction Web service” is as follows:


REQUEST PARAMETERS		
Parameter name	Description	Type
AcquirerID	The acquirer id. Provided by v.	String (up to 5 characters)
MerchantID	Merchant code (merchant id). Provided by Euronet Merchant Services.	Integer
PosID	Pos number (Pos id). Provided by Euronet Merchant Services.  Note: It is possible that a NULL value is sent to the PosID. In this case, the epay eCommerce will carry out the transaction with one of the available PosIDs. This is recommended in cases where a big number of simultaneous transactions will be sent, so Euronet Merchant Services must be notified in order to create more than one PosIDs.	Integer
User	Username. Provided by Euronet Merchant Services.	String (up to 50 characters)



Password	Password <u>encrypted with the MD5 hashing algorithm</u> . Provided by Euronet Merchant Services (in non-encrypted form).	String (up to 50 characters)
ChannelType	Terminal Channel type. Provided by Euronet Merchant Services.	String (up to 11 characters)
RequestType	<p>The type of transaction that will be executed. Possible values:</p> <ul style="list-style-type: none"> ▪ SALE: <u>Sale</u> → Transaction that will be directly cleared in the current package. ▪ AUTHORIZE: <u>Pre-authorization</u> → A reservation of the amount will be made and later the pre-authorization shall be completed (either through the AdminTool management tool or through the transaction with RequestType = "SETTLE") in order for it to be cleared. <div style="background-color: #ffffcc; padding: 10px; margin: 10px 0;"> <p> Attention! The pre-authorization is available under conditions. Communication and approval by Euronet Merchant Services are required, if the business is interested in using this type of transaction.</p> </div> <ul style="list-style-type: none"> ▪ SETTLE: <u>Preauthorization settlement</u> → It concerns the completion of a preauthorization in order to settle the transaction in the current package. ▪ VOIDREQUEST: <u>Preauthorization voiding</u> → Voiding of a preauthorization which is not settled. ▪ REFUND: <u>Sale cancellation/refund</u> → Refund of a sale or preauthorization that has been settled. ▪ FOLLOW_UP: <u>Transaction follow-up</u> → The data of an executed transaction with a specific "MerchantReference" value, are returned (provided that a cancellation/refund has not been carried out for that transaction). ▪ ISAVAILABLE: It is returned if epay eCommerce is available to receive transactions. <div style="background-color: #ffffcc; padding: 10px; margin: 10px 0;"> <p> Note: At the end of the section, a diagram is included with the order in which the transactions can be used.</p> </div>	String (up to 20 characters)


RequestMethod	The "SYNCHRONOUS" value should always be sent.	String (up to 12 characters)
MerchantReference	<p>Transaction reference code It is generated by the merchant system and identifies uniquely each successful transaction (e.g. order number, contract number, etc.).</p> <ul style="list-style-type: none"> ▪ "MerchantReference" accepts max. 50 Greek and Latin uppercase and lowercase alphanumeric characters, space and the following special characters: /:_().,+ - ▪ It should have a different value for each transaction. <div style="background-color: #ffffcc; padding: 10px; margin-top: 10px;"> <p>⚠ Attention!</p> <ul style="list-style-type: none"> ▪ If the card holder authentication process is used ("3D Secure"), the value of the "MerchantReference" parameter in sale/preauthorization transactions should be <u>identical</u> to the value of the corresponding parameter when the Wrapper3DSecure service is called. ▪ If an online sale/preauthorization transaction is unsuccessful and needs to be resent, the 3D Secure process should be repeated with a <u>"MerchantReference value which should be different"</u> from that in the previous transaction. ▪ Where a sale or preauthorization transaction has been approved, even if it has been cancelled/refunded, its "MerchantReferece" cannot be used in a subsequent transaction. </div>	String (up to 50 characters)
TransactionReferenceID	<p>Used only in the following transactions:</p> <ul style="list-style-type: none"> ▪ Preauthorization settlement (RequestType = "SETTLE") ▪ Preauthorization voiding (RequestType = "VOIDREQUEST") ▪ Sale refund (RequestType = "REFUND") <p>It is the transaction id ("TransactionID" parameter in response message) of the</p>	Integer

	<p>transaction requested to be settled / refunded.</p> <p> Note: If refund is requested for a preauthorisation that has been settled, the "TransactionID" of the settlement (not preauthorization) is submitted to this parameter.</p>	
EntryType	<p>The way the card details were entered. Possible values:</p> <ul style="list-style-type: none"> ▪ KeyEntry: The card details were typed by the card holder. ▪ CardOnFile: The card details were stored and did not have to be entered by the card holder. <p>If no value is sent, KeyEntry is considered the default value.</p>	String (up to 8 characters)
CurrencyCode	<p>The code of the transaction currency. It is 978 for debits in Euros.</p> <p> Attention!</p> <ul style="list-style-type: none"> ▪ For every different currency, Euronet Merchant Services will provide a different MerchantID and PosID. ▪ If the card holder authentication process is used ("3D Secure"), the value of the "CurrencyCode" parameter in sale/pre-authorisation transactions should be identical to the value of the corresponding parameter (Currency) when the Wrapper3DSecure service is called. <p> Note: Supported currency codes are listed in Annex 2.</p>	Integer
Amount	<p>The total amount of the transaction with 2 decimal places. In a sale transaction, it is the amount that will be collected by the business, and in a refund transaction, it is the amount that will be deducted from the business's account (irrespective of whether part of the amount is covered from the card and part of it from yellows). The following apply for the different types of transactions:</p> <ul style="list-style-type: none"> ▪ Preauthorization settlement (RequestType = "SETTLE") The amount can be lower than or equal to the initial transaction amount. 	Decimal with 2 decimal places

	<ul style="list-style-type: none"> ▪ Preauthorization voiding (RequestType = "VOIDREQUEST"): The amount must be equal to the initial transaction amount. ▪ Sale cancellation/refund (RequestType = "REFUND") The amount may be lower than or equal to the initial transaction amount. <div>  Attention! <ul style="list-style-type: none"> ▪ If the card holder authentication process is used ("3D Secure"), the value of the "Amount" parameter in sale/preauthorization transactions should correspond to the same amount expressed by the PurchAmount and Exponent parameters when the Wrapper3DSecure service is called. ▪ <u>Partial refund in installment transactions</u> will be carried out as one-off (without installments). </div>	
Installments	<p>The number of installments of the transaction.</p> <ul style="list-style-type: none"> ▪ For installments to be supported, the business must declare this to Euronet Merchant Services. ▪ For a transaction without installments, the value 0, 1 or NULL must be sent. <div>  Note: Euronet Merchant Services provides the "BIN Web Service" through which one can check if a card supports installments or not without carrying out a transaction. In case of interest, the technical specifications shall be requested from Euronet Merchant Services. </div>	Integer
ExpirePreauth	<p>Concerns only pre-authorization transactions (RequestType = "AUTHORIZE"). It is the number of days within which the pre-authorization can be completed. Maximum value: 30 days</p>	Short Integer
TipAmount	For future use. NULL or zero shall be sent.	Decimal with decimal places 2

Bnpl	For future use. NULL shall be sent.	Unsigned Byte
SessionKey	For future use. NULL shall be sent.	String (up to 50 characters)
CardType	<p>The card type.</p> <p>There are two possibilities:</p> <p>1) The user is not requested to enter their card type. In this case, the "CardType" parameter must be sent with an "UNKNOWN" value and epay eCommerce will decide on the card type of the transaction.</p> <p>2) The user is requested to enter their card type. In this case, the possible values for the "CardType" parameter are the following:</p> <ul style="list-style-type: none"> ▪ VISA: VISA card ▪ MasterCard: MasterCard card ▪ Maestro: Maestro card. Maestro cards can be used <u>only if the 3d-secure process is implemented</u>. ▪ DinersClub: DinersClub or Discover card ▪ AMEX: American Express card <p> Note:</p> <ul style="list-style-type: none"> ▪ In order for Diners/ Discover or American Express cards to be supported, the business shall contact Euronet Merchant Services in order to be informed about the necessary business process. ▪ Transactions with a Diners/ Discover or American Express card are sent with <u>a different MerchantID and PosID</u> compared to Visa / Mastercard / Maestro transactions and <u>with "null" value to the "AuthInfo" element</u> (Cavv, Eci, Xid, etc. parameters are included – see below). 	String (up to 20 characters)
CardNumber	The card number of the transaction. The maximum number of digits of the card can be 19.	String (up to 19 numerical digits)
ExpirationMonth	The card's expiration month.	Short integer

ExpirationYear	The card's expiration year.	Short integer
Cvv2	<p>The card verification code (CVV2 or CVC) usually found on the back of the card.</p> <p> Note: In the case of website transactions, characters should not be visible in the CVV2 field as they are typed by the user (e.g. replaced by asterisks).</p>	String (up to 4 numerical digits)
CardHolderName	<p>The card holder's full name as printed on the card.</p> <p> Attention! The full name should be sent using uppercase Latin characters.</p>	String (up to 100 characters)
Aid	Not used, NULL shall be sent.	String
Emv	Not used, NULL shall be sent.	String
PinBlock	Not used, NULL shall be sent.	String
Track1	Not used, NULL shall be sent.	String
Track2	Not used, NULL shall be sent.	String
Cavv	Concerns only on-line transactions through a website using a Visa, Mastercard or Maestro card. It contains the value for the " Cavv " parameter returned in the 3D Secure process (see section 6).	String (up to 48 characters)
Eci	Concerns only on-line transactions through a website using a Visa, Mastercard or Maestro card. It contains the value for the " Eci " parameter returned in the 3D Secure process (see section 6).	String (2 numerical digits)
Xid	Concerns only on-line transactions through a website using a Visa, Mastercard or Maestro card. It contains the value for the " Xid " field returned in the 3D Secure process (see section 6).	String (up to 40 characters)
Enrolled	No value needs to be sent	String
PAResStatus	No value needs to be sent	String
SignatureVerification	No value needs to be sent	String
Protocol	Concerns only on-line transactions through a website using a Visa, Mastercard or	String

	Maestro card and expresses the version of the 3D Secure protocol used for authentication. It contains the value for the " Protocol " field returned in the 3D Secure process (see section 6). Possible values: <ul style="list-style-type: none"> 1 (for 3D Secure version 1) 2 (for 3D Secure version 2 or EMV 3D Secure) 	
DsTransID	Concerns only on-line transactions through a website using a Visa, Mastercard or Maestro card. It contains the value of the " DsTransID " parameter returned in the 3D Secure process (see section 6).	String
RecurringInd	Used if the transaction concerns a recurring payment, that is when there is an agreement between the card holder and the business for recurring debits (e.g. standing order). Possible values: <ul style="list-style-type: none"> R: In the case of a recurring transaction executed at regular intervals C: In the case of a recurring transaction not executed at regular intervals. 	String
TraceID	In the case of recurring payments and following the second recurrence, it includes the value of the Trace ID of the first transaction, that was returned to the merchant upon the Transaction Web Service call for that first transaction.	String
TaxCardNumber	No value shall be sent	String
	Attention! <ul style="list-style-type: none"> In sale transactions (RequestType=SALE) <u>with redemption of yellows</u>, values shall be sent to all 4 parameters that follow: ClearCardAmount, RedemptionAmount, PointsBurned, Rate. In sale transactions (RequestType=SALE) <u>without redemption of yellows</u>, the ClearCardAmount, RedemptionAmount, PointsBurned, Rate parameters shall be sent with a NULL value. In all other transaction types, the ClearCardAmount, RedemptionAmount, PointsBurned, Rate parameters shall be sent with a NULL value. 	
	ClearCardAmount	For a sale transaction (RequestType=SALE) with redemption of

	<p>yellows, the amount with which the card was debited (in euros) is sent.</p>	
RedemptionAmount	<p>For a sale transaction (RequestType=SALE) with redemption of yellows, the amount (in euros) corresponding to the yellows to be redeemed is sent.</p>	
PointsBurned	<p>For a sale transaction (RequestType=SALE) with redemption of yellows, the amount of yellows to be redeemed is sent.</p>	
Rate	<p>For a sale transaction (RequestType=SALE) with redemption of yellows, the yellow/euros exchange rate is sent as returned from the call to the LoyaltyMemberBalance Service (Rate parameter).</p>	









Note:


With the exception of the "CardHolderName" parameter, the use of spaces in any other String-type parameter is not allowed.

The parameters sent to the response are the following:

RESPONSE PARAMETERS		
Parameter name	Description	Type
RequestType	<p>The transaction type sent with the request. Possible values:</p> <ul style="list-style-type: none"> ▪ SALE: Sale ▪ AUTHORIZE: Preauthorization ▪ SETTLE: Preauthorization settlement ▪ VOIDREQUEST: Preauthorization voiding ▪ REFUND: Refund of a sale or preauthorization that has been settled ▪ FOLLOW_UP: Data of a transaction already executed with a specific "MerchantReference" (provided that a cancellation/refund has not been carried out for that transaction) ▪ ISAVAILABLE: epay eCommerce availability check 	String (up to 20 characters)
MerchantID	Merchant code (merchant id) sent in the request.	Integer
PosID	Pos number (Pos id) sent in the request.	Integer
User	Username sent in the request.	String (up to 50 characters)
ChannelType	Terminal channel type sent in the request.	String (up to 11 characters)
ResultCode	<p>The request result code indicating whether there was any technical problem in the transaction processing. Specifically :</p> <ul style="list-style-type: none"> ▪ Value = 0: There was no problem; the transaction was executed. <u>Then, the «StatusFlag» parameter must be checked to verify that the transaction was approved.</u> ▪ Value ≠ 0: There was a transaction data problem or a technical problem at epay eCommerce, so no transaction was executed. The «ResultDescription» parameter contains the problem description. <p>The following applies specifically to transactions where RequestType= «ISAVAILABLE»:</p> <ul style="list-style-type: none"> ▪ Value = 0: epay eCommerce may accept transactions. 	Integer

	<ul style="list-style-type: none"> ▪ Value ≠ 0: epay eCommerce may not accept transactions. <div>  Note: The most frequent values of the "ResultCode" are included in Annex 1. </div>	
ResultDescription	<p>The description corresponding to the value of the "ResultCode" parameter.</p> <div>  Note: <ul style="list-style-type: none"> ▪ This information is not recommended to be shown to the user. ▪ If the request is rejected due to anti-fraud checks (ResultCode= 7001, see Annex 1), the «ResultDescription» parameter contains the code of the rule that was fired-up. <u>The zero value (0) means that the card number is included in a black list.</u> If special anti-fraud rules have been agreed with the merchant, Euronet Merchant Services will provide the relevant rule codes that may be returned. </div>	String (up to 1024 characters)
SupportReferenceID	<p>Reference id of the request made. It has a different value for each request (even if no transaction was carried out).</p> <div>  Note: It is necessary to save its value so that, if needed, it can be used as a reference point upon communication with Euronet Merchant Services. </div>	Long integer
StatusFlag	<p>The value of the parameter declares whether the transaction was approved. Possible values:</p> <ul style="list-style-type: none"> ▪ Success: The transaction was approved. ▪ Failure: The transaction was not approved. 	String (up to 12 characters)
ResponseCode	<p>In case a transaction has been carried out, it contains the response code. The response codes for an approved transaction are the following: 00, 08, 10, 16.</p> <div>  Note: The most frequent "ResponseCode" values are included in Annex 1. </div>	String (2 characters)

ResponseDescription	<p>The description corresponding to the value of the "ResponseCode" parameter.</p> <p> Note: This information is not recommended to be shown to the user.</p>	String (up to 120 characters)
TransactionID	<p>If a transaction has been made, it contains a unique transaction number generated from epay eCommerce.</p> <p> Note: This value is required in the "TransactionReferenceID", if the following transactions are to be used:</p> <ul style="list-style-type: none"> ▪ Preauthorisation settlement (RequestType = "SETTLE") ▪ Preauthorization voiding (RequestType = "VOIDREQUEST") ▪ Sale cancellation/refund (RequestType = "REFUND"), <p>Therefore, if the above transactions are to be used, the parameter value of the initial transaction must be stored (i.e. the preauthorization, settlement or sale "TransactionID").</p>	Integer
MerchantReference	The reference code of the transaction sent in the request.	String (up to 50 characters)
ApprovalCode	If a successful transaction has been executed (i.e. when ResultCode=0 and StatusFlag=Success), it takes the transaction approval code.	String (up to 6 characters)
PackageNo	If a transaction has been executed (i.e. when ResultCode=0), it takes the number of the package that includes this transaction.	Integer
RetrievalRef	If a transaction has been executed (i.e. when ResultCode=0), it takes the Retrieval Reference Number generated by the acquiring system.	String (up to 12 characters)
TransactionDateTime	If a transaction has been executed (i.e. when ResultCode = 0), the transaction execution date and time are included.	DateTime
SessionKey	For future use. NULL is sent.	String (up to 50 characters)
TransactionTraceNum	If a transaction has been executed (i.e. when ResultCode = 0), the transaction serial number is included in the package it belongs to.	Integer

TraceID	Transaction reference code generated by Visa/Mastercard; it is recommended that this code be stored by the merchant's system. <u>Usefulness in recurring transactions:</u> If the transaction is the first in a series of recurring payments preceded by the 3D Secure process, the value of this parameter should be stored so as to be included in the request (TraceID) in each subsequent recurrence (where 3D Secure is not used).	String (up to 50 characters)
	Attention! If a sale transaction is made with redemption of yellows and the card is successfully debited, but a problem occurs upon redemption/earning yellows, the response message will contain the data of the successful transaction (e.g. StatusFlag=Success) <u>without values in the parameters:</u> <ul style="list-style-type: none">▪ AvailableAmount▪ AvailablePointNumber▪ BurnedPointBalance▪ EarnedPointBalance▪ LoyaltyTransactionId The redemption and earning yellows will automatically take place once this is possible <u>without the need of further actions from the part of the business.</u>	
AvailableAmount	Has a value only in a successful transaction. It contains the monetary value of the yellows owned by the card holder based on the current exchange rate between yellow and euros after the completion of the transaction.	Decimal
AvailablePointNumber	Has a value only in a successful transaction. It contains the total available yellows of the customer after the completion of the transaction	Integer
BurnedPointBalance	Has a value only in a successful transaction. In a sale transaction it contains the number of yellows redeemed. In a refund it contains the number of yellows returned to the customer (value with a minus sign).	Integer
EarnedPointBalance	Has a value only in a successful transaction. In a sale transaction it contains the number of yellows earned by the customer. In a refund it contains the number of yellows taken from the customer (value with a minus sign).	Integer
LoyaltyTransactionId	Has a value only in a successful transaction. A unique id regarding the system managing the member's yellows.	String

**Note:**

- In a refund concerning a sale during which yellows were redeemed, steps are taken so that, together with the return of the amount to the card, the yellows redeemed are returned and the yellows earned are cancelled. If the refund is partial, Euronet Merchant Services ensures that the amount that will be returned to the card and the yellows returned/cancelled, arise in the same proportion between the amount of the refund and the total amount of the initial charge.
- In a sale transaction with redemption of yellows, the "Amount" parameter includes the total amount of the transaction that is allocated to the ClearCardAmount and RedemptionAmount parameters (that is $\text{Amount} = \text{ClearCardAmount} + \text{RedemptionAmount}$).
- The "**SupportReferenceID**" and "**MerchantReference**" parameter values of all transactions must be stored in the merchant system and be available to the merchant responsible person(s).
- It is recommended that the "**TraceID**" parameter be returned with the response be also stored. For now, this value is useful to merchants supporting recurring transactions.
- If some of the preauthorization settlement, preauthorization voiding and/or refund transactions are used, then the "**TransactionID**" should also be stored.
- Of the remaining parameters, it is recommended to also store the "**ResultCode**", "**ResultDescription**", "**StatusFlag**", "**ResponseCode**", "**ResponseDescription**", "**ApprovalCode**", "**PackageNo**" parameter values.
- The transaction decline or technical error message ("**ResultDescription**" or "**ResponseDescription**") should not appear as such on the user page.

In the following table, the parameters of the "Transaction Web Service" that need to be used in the request of every type of transaction are shown. In the sale transactions ("SALE") and pre-authorization ("AUTHORIZE"), the parameters used depend on the value of the "ChannelType" ("3DSecure", "eCommerce", "MOTO") provided by Euronet Merchant Services.

REQUEST PARAMETERS	SALE (SALE)		AUTHORIZE (PRE-AUTHORIZATION)		SETTLE (PRE-AUTHORIZATION SETTLEMENT)	VOIDREQUEST (PRE-AUTHORIZATION CANCELLATION)	REFUND (SALE CANCELATION/ REFUND)	FOLLOW_UP (TRANSACTION FOLLOW UP)	ISAVAILABLE (epay eCommerce AVAILABILITY CHECK)
	3D Secure	MOTO or eCommerce	3D Secure	MOTO or eCommerce	For all Channel Type values	For all Channel Type values	For all Channel Type values	For all Channel Type values	For all Channel Type values
AcquirerID	✓	✓	✓	✓	✓	✓	✓	✓	✓
MerchantID	✓	✓	✓	✓	✓	✓	✓	✓	✓
PosID	✓	✓	✓	✓	✓	✓	✓	✓	✓
User	✓	✓	✓	✓	✓	✓	✓	✓	✓
Password	✓	✓	✓	✓	✓	✓	✓	✓	✓
ChannelType	✓	✓	✓	✓	✓	✓	✓	✓	✓
RequestType	✓	✓	✓	✓	✓	✓	✓	✓	✓
RequestMethod	✓	✓	✓	✓	✓	✓	✓	✓	✓
MerchantReference	✓	✓	✓	✓	✗	✗	✗	✓	✗
TransactionReferenceID	✗	✗	✗	✗	✓	✓	✓	✗	✗
EntryType	✓	✓	✓	✓	✗	✗	✗	✗	✗
CurrencyCode	✓	✓	✓	✓	✓	✓	✓	✗	✗
Amount	✓	✓	✓	✓	✓	✓	✓	✗	✗
Installments	(1)	(1)	(1)	(1)	✗	✗	✗	✗	✗
ExpirePreauth	✗	✗	✓	✓	✗	✗	✗	✗	✗
TipAmount	✗	✗	✗	✗	✗	✗	✗	✗	✗

Bnpl	✗	✗	✗	✗	✗	✗	✗	✗	✗
SessionKey	✗	✗	✗	✗	✗	✗	✗	✗	✗
CardType	✓	✓	✓	✓	✗	✗	✗	✗	✗
CardNumber	✓	✓	✓	✓	✗	✗	✗	✗	✗
ExpirationMonth	✓	✓	✓	✓	✗	✗	✗	✗	✗
ExpirationYear	✓	✓	✓	✓	✗	✗	✗	✗	✗
Cvv2	✓	✓	✓	✓	✗	✗	✗	✗	✗
CardHolderName	(2)	(2)	(2)	(2)	✗	✗	✗	✗	✗
Aid	✗	✗	✗	✗	✗	✗	✗	✗	✗
Emv	✗	✗	✗	✗	✗	✗	✗	✗	✗
PinBlock	✗	✗	✗	✗	✗	✗	✗	✗	✗
Track1	✗	✗	✗	✗	✗	✗	✗	✗	✗
Track2	✗	✗	✗	✗	✗	✗	✗	✗	✗
Cavv	(3)	✗	(3)	✗	✗	✗	✗	✗	✗
Eci	✓	✗	✓	✗	✗	✗	✗	✗	✗
Xid	(3)	✗	(3)	✗	✗	✗	✗	✗	✗
Enrolled	✗	✗	✗	✗	✗	✗	✗	✗	✗
PAResStatus	✗	✗	✗	✗	✗	✗	✗	✗	✗
SignatureVerification	✗	✗	✗	✗	✗	✗	✗	✗	✗
Protocol	(3)	✗	(3)	✗	✗	✗	✗	✗	✗
DsTransID	(3)	✗	(3)	✗	✗	✗	✗	✗	✗
RecurringInd	(4)	(4)	(4)	(4)	✗	✗	✗	✗	✗
TraceID	(5)	(5)	(5)	(5)	✗	✗	✗	✗	✗

TaxCardNumber	×	×	×	×	×	×	×	×	×
ClearCardAmount	(6)	(6)	×	×	×	×	×	×	×
RedemptionAmount	(6)	(6)	×	×	×	×	×	×	×
PointsBurned	(6)	(6)	×	×	×	×	×	×	×
Rate	(6)	(6)	×	×	×	×	×	×	×

EXPLANATION OF SYMBOLS	
Symbol	Explanation
✓	A value must be sent
×	No value must be sent
(1)	A value is sent in case of a transaction with installments
(2)	Optional information
(3)	A value is submitted depending on the outcome of the card holder authentication process ("3D Secure" process) – see section 6.
(4)	A value is sent in case of a recurring transaction.
(5)	A value is submitted after the second recurrence of a recurring transaction.
(6)	A value is sent only in sale transactions with redemption of yellows.

In the diagram that follows, the order in which the various types of transactions can be used is shown. More specifically, as shown in the diagram, the following apply:

- A preauthorization (RequestType="AUTHORIZE"), may either be settled (RequestType="SETTLE") for an amount lower than or equal to the preauthorization amount, or voided (RequestType="VOIDREQUEST") for an amount equal to the preauthorization amount.
- A preauthorization that has been settled, may be refunded (RequestType="REFUND") for an amount lower than or equal to the preauthorization settlement amount. **Attention!** Partial refund in installment transactions will be carried out as one-off (without installments).
- A sale transaction may be refunded (RequestType="REFUND") for an amount lower than or equal to the sale amount. **Attention!** Partial refund in installment transactions will be carried out as one-off (without installments).
- A "RequestType="FOLLOW_UP" transaction may be used at any time and returns the details of a request already sent, provided no cancellation/refund has been performed.



Note:

In the following transactions, a value must be filled in the "**TransactionReferenceID**" parameter:

- Preauthorization settlement (RequestType = "SETTLE")
- Preauthorization voiding (RequestType = "VOIDREQUEST")
- Sale or settlement cancellation/refund (RequestType = "REFUND")

In any case the transaction id ("TransactionID" parameter in response message) of the preceding transaction is submitted. For example, for the preauthorization settlement, the preauthorization "TransactionID" is used, while for the settlement refund, the settlement "TransactionID" is used.

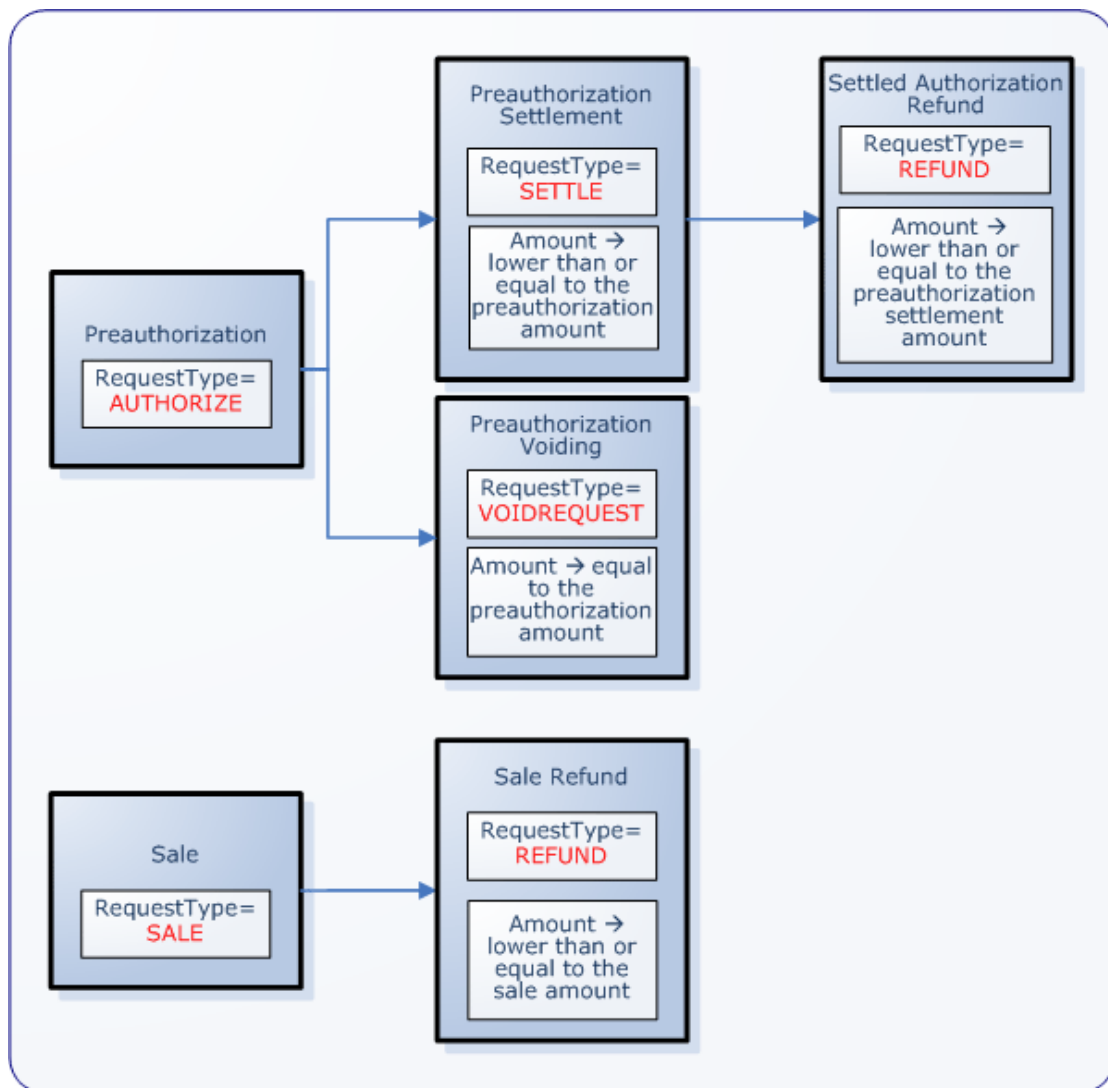


Diagram 1: Transaction sequence



8. Merchant Application Action Flow

Following an analysis of all individual process modules to be implemented (strong customer authentication and dispatch of transaction to epay eCommerce), the diagram shows the flow of actions to be performed by the merchant's application in collaboration with epay eCommerce as required for a transaction execution.

It is important to use the proposed algorithm, so that all cases are taken into account and no problems occur during the application productive operation.

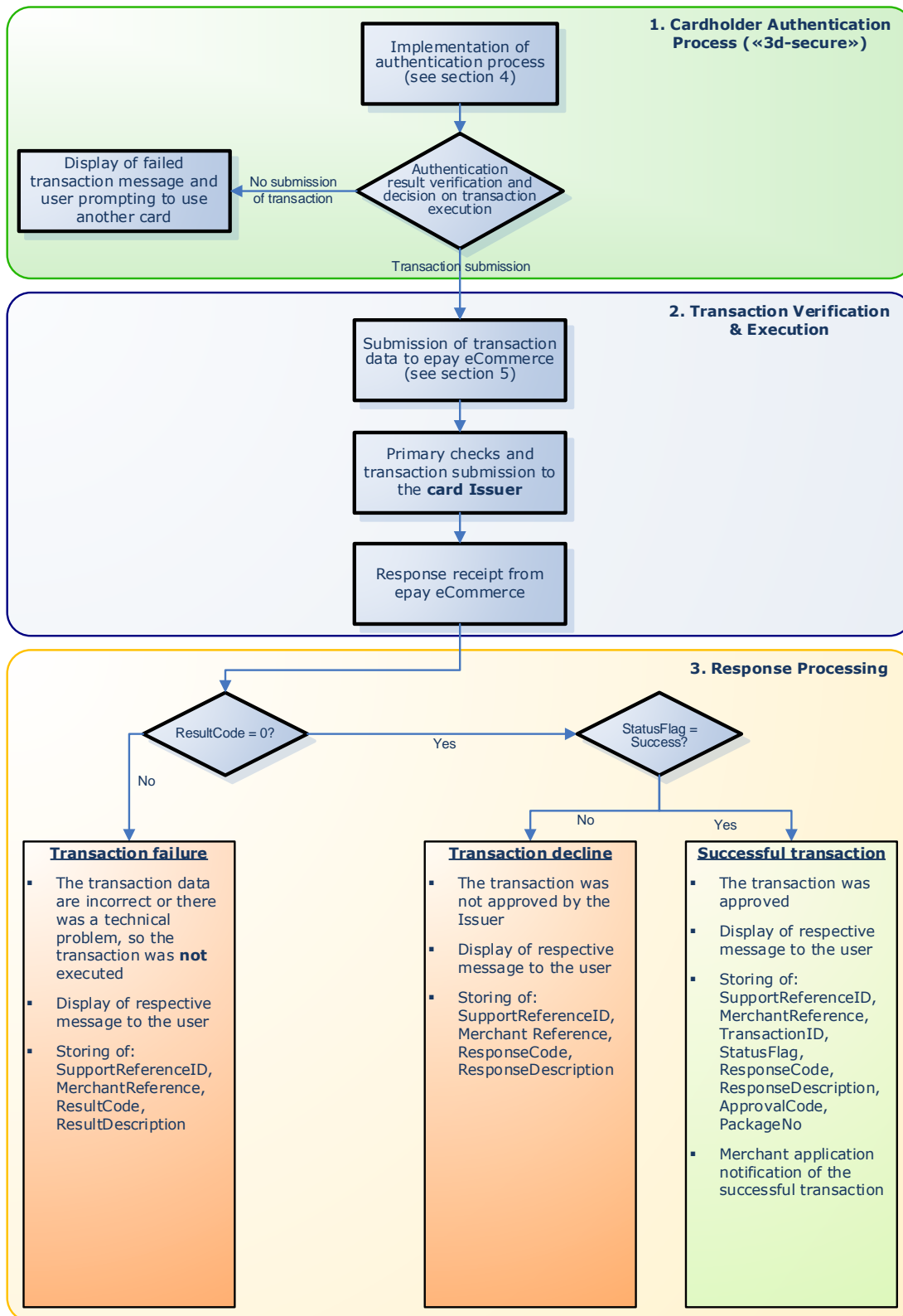


Diagram 2: Merchant application actions

As shown in the diagram, the total process consists of the following 3 phases:

1. Strong Customer Authentication ("3D Secure") process

In the case of sale or preauthorization transaction executed via a website (online transactions) using a Visa, Mastercard or Maestro card, the strong customer authentication process is carried out (see section 6). Depending on the authentication outcome, the merchant's application determines whether to submit the transaction to epay eCommerce or not.



Attention!

The authentication process is only used for online **sale** or **preauthorization** transactions with Visa, Mastercard or Maestro cards.

2. Transaction Verification & Execution

The merchant system uses the «Transaction Web Service» to submit the transaction data to epay eCommerce (see section 7). epay eCommerce runs primary checks to the submitted data and, if correct, the transaction data are submitted to the card Issuer. Then, a response is sent to the merchant system.

3. Response Processing

The merchant system must check the response parameters, to verify whether the transaction is successful. Specifically:

- **If ResultCode ≠ «0»**, then there was either a problem with transaction data, or some **technical problem**, thus the transaction was not executed. A problem description is contained in the «ResultDescription» parameter (not to be displayed on the user page). If necessary, the details of the technical problem (SupportReferenceID, MerchantReference, ResultCode, ResultDescription) are stored in the merchant system.
- **If ResultCode = «0»:**
 - If StatusFlag ≠ «Success», then the transaction was executed but not **approved by the card Issuer**. If necessary, the details of the unsuccessful transaction (SupportReferenceID, MerchantReference, ResponseCode, ResponseDescription) are stored in the merchant system.

- If `StatusFlag = «Success»`, then **the transaction was successful**, thus the transaction information, such as `SupportReferenceID`, `MerchantReference`, `TransactionID`, `StatusFlag`, `ResponseCode`, `ResponseDescription`, `ApprovalCode`, `PackageNo` should be stored and the merchant system notified of the successful transaction. Moreover, if it is a sale transaction, it is recommended to show to the user data regarding the yellows redeemed/earned and the new available number of yellows (`AvailableAmount`, `AvailablePointNumber`, `BurnedPointBalance`, `EarnedPointBalance`, `LoyaltyTransactionId` parameters of the response).

**Attention!**

The «`SupportReferenceID`» value should always be stored so that it may be used as reference in the communication between the merchant and Euronet Merchant Services.

**Note:**

- It is suggested that the transaction approval code («`ApprovalCode`») be indicated and/or sent on a transaction confirmation email from the merchant to the user.
- It is recommended that the transaction decline or technical error messages (`ResultDescription`, `ResponseDescription`) should not appear as such on the user page.



9. Test Cases

Below, the test cases that can be executed in the test environment of the epay eCommerce service have been recorded (call to the LoyaltyMemberBalance Service and to the Transaction Web Service). It is mandatory that test transactions are executed for all test cases with the indication "MANDATORY". From the optional ones, those that are considered to apply in the implemented system can be executed.

Since the use of the 3D-secure process is required, this must precede using any case of section 4.

TEST CASES FOR THE LoyaltyMemberBalance SERVICE



Test Case 1: CARD NOT SUPPORTING LOYALTY

MANDATORY



Input parameters:

Parameter	Value
CardNumber	4273101111111116



Response parameters:

Parameter	Value
StatusFlag	Failure
ResultCode	9113
ResultDescription	Card does not participate in loyalty program
MemberStatus	0
AvailableAmount	-
AvailableBalance	-
PartnerExchangeRate	-
ProgramCode	-
RedeemFloorLimit	-



Test Case 2: CARD SUPPORTING LOYALTY (MemberStatus=1)

MANDATORY



Input parameters:

Parameter	Value
CardNumber	4020680000000114



Response parameters:

Parameter	Value
StatusFlag	Success
ResultCode	0
ResultDescription	-
MemberStatus	1
AvailableAmount	5839,69
AvailableBalance	2919845
PartnerExchangeRate	0,002
ProgramCode	PB-BWL-01
RedeemFloorLimit	0



Test Case 3: CARD THAT RETURNS MemberStatus=2

MANDATORY



Input parameters:

Parameter	Value
CardNumber	4410292222222225



Response parameters:

Parameter	Value
StatusFlag	Success
ResultCode	0
ResultDescription	-
MemberStatus	2
AvailableAmount	-
AvailableBalance	-
PartnerExchangeRate	-
ProgramCode	-
RedeemFloorLimit	-



Test Case 4: CARD THAT RETURNS MemberStatus=3

MANDATORY



Input parameters:

Parameter	Value
CardNumber	4410293333333331



Response parameters:

Parameter	Value
StatusFlag	Success
ResultCode	0
ResultDescription	-
MemberStatus	3
AvailableAmount	-
AvailableBalance	-
PartnerExchangeRate	-
ProgramCode	-
RedeemFloorLimit	-



Test Case 5: CARD THAT RETURNS MemberStatus=4

MANDATORY



Input parameters:

Parameter	Value
CardNumber	4410294444444447



Response parameters:

Parameter	Value
StatusFlag	Success
ResultCode	0
ResultDescription	-
MemberStatus	4
AvailableAmount	-
AvailableBalance	-
PartnerExchangeRate	-
ProgramCode	-
RedeemFloorLimit	-



Test Case 6: CARD THAT RETURNS MemberStatus=5

MANDATORY



Input parameters:

Parameter	Value
CardNumber	4410295555555558



Response parameters:

Parameter	Value
StatusFlag	Success
ResultCode	0
ResultDescription	-
MemberStatus	5
AvailableAmount	-
AvailableBalance	-
PartnerExchangeRate	-
ProgramCode	-
RedeemFloorLimit	-



Test Case 7: CARD THAT RETURNS ANOTHER VALUE AT THE MemberStatus

MANDATORY



Input parameters:

Parameter	Value
CardNumber	4410296666666664



Response parameters:

Parameter	Value
StatusFlag	Success
ResultCode	0
ResultDescription	-
MemberStatus	6
AvailableAmount	-
AvailableBalance	-
PartnerExchangeRate	-
ProgramCode	-
RedeemFloorLimit	-



Test Case 8: CARD THAT RETURNS ERROR

MANDATORY



Input parameters:

Parameter	Value
CardNumber	4410297777777770



Response parameters:

Parameter	Value
StatusFlag	Failure
ResultCode	100
ResultDescription	Authentication Error
MemberStatus	-
AvailableAmount	-
AvailableBalance	-
PartnerExchangeRate	-
ProgramCode	-
RedeemFloorLimit	-

TEST CASES FOR THE TRANSACTION WEB SERVICE

A brief list of the test cases for the Transaction Web Service is the following:

S/N	TITLE	MANDATORY
Test case 1	APPROVED TRANSACTION WITH PARTIAL REDEMPTION (VISA)	YES
Test case 2	APPROVED TRANSACTION WITH FULL REDEMPTION (VISA)	YES
Test case 3	APPROVED TRANSACTION - ONLY EARN (VISA)	YES
Test case 4	APPROVED TRANSACTION - UNAVAILABLE LOYALTY MANAGEMENT (VISA)	YES
Test case 5	APPROVED TRANSACTION – NON-PARTICIPATING CARD IN YELLOW PROGRAM (VISA)	YES
Test case 6	DECLINED TRANSACTION	YES
Test case 7	RECHARGE ATTEMPT	YES
Test case 8	COMMUNICATION ERROR	YES
Test case 9	INVALID CARD NUMBER	YES
Test case 10	UNDER-PROCESS TRANSACTION WAS RE-SENT	YES
Test case 11	BATCH IS CLOSING	YES
Test case 12	GENERAL ERROR	YES
Test case 13	APPROVED TRANSACTION WITH INSTALLMENTS	NO
Test case 14	APPROVED TRANSACTION (MASTERCARD)	NO
Test case 15	APPROVED TRANSACTION (DINERS)	NO
Test case 16	APPROVED TRANSACTION (DISCOVER)	NO
Test case 17	APPROVED TRANSACTION (AMERICAN EXPRESS)	NO
Test case 18	APPROVED TRANSACTION (GBP)	NO
Test case 19	APPROVED TRANSACTION (USD)	NO

In all test cases the following apply:

- The values of the "AcquirerID", "MerchantID", "PosID", "User", "Password" parameters are provided by Euronet Merchant Services.
- The "RequestType" parameter is filled depending on the type of transaction (see section 7) .
- The "Amount" parameter may include any valid value (see section 7).
- The values of the "Installments", "CurrencyCode", "ExpirePreauth", "CardType", "CardNumber", "ExpirationMonth", "ExpirationYear", "CVV2" parameters are filled with the values referred to in the test cases.

**Note:**

- It is reminded that the pre-authorization is a transaction with which an amount is reserved, and the completion of the pre-authorization must be made by the business (either through the epay eCommerce AdminTool, or through a call to the Transaction Web Service) within the days set through the "ExpirePreauth" parameter, in order for the transaction to be cleared.
- In the test transactions of pre-authorizations, the "ExpirePreauth" parameter must have a value exactly 30, but in the live environment, it can have a value between 2 and 30 (if pre-authorizations are used).



Test Case 1: APPROVED TRANSACTION WITH PARTIAL REDEMPTION (VISA)

MANDATORY

Scenario: Successful sale (without installments) with partial redemption of yellows (that is part of the amount of the transaction will be covered from the charge of the card and part of it from the redemption of yellows)



When it applies:

When ResultCode=0 and StatusFlag=Success



Input parameters:

Parameter	Value
RequestType	SALE
ExpirePreauth	0
Currency	978
Installments	0
CardType	VISA or UNKNOWN
CardNumber for purchase	4020680000000114
ExpirationMonth	02
ExpirationYear	Any future
CVV2	123
Amount	292,87
ClearCardAmount	287,87
RedemptionAmount	5
PointsBurned	2500
Rate	0,002



Response parameters:

Parameter	Value
ResultCode	0
ResponseCode	00
StatusFlag	Success
AvailableAmount	5839,69
AvailablePointNumber	2919845
BurnedPointBalance	2500
EarnedPointBalance	2878
LoyaltyTransactionId	1-1YYFIAN



Merchant application actions:

- Display of a message of transaction approval to the user, as well as information regarding the yellows
- Saving of the values of the SupportReferenceID, MerchantReference, TransactionID, StatusFlag, ResponseCode, ResponseDescription, ApprovalCode, PackageNo parameters
- Update of the merchant's application on the successful transaction



Test Case 2: APPROVED TRANSACTION WITH FULL REDEMPTION (VISA)

MANDATORY

Scenario: Successful sale (without installments) with full redemption of yellows (that is the entire amount of the transaction will be covered from the redemption of yellows)



When it applies:

When ResultCode=0 and StatusFlag=Success



Input parameters:

Parameter	Value
RequestType	SALE
ExpirePreauth	0
Currency	978
Installments	0
CardType	VISA or UNKNOWN
CardNumber for purchase	4020680000000114
ExpirationMonth	03
ExpirationYear	Any future
CVV2	123
Amount	40,02
ClearCardAmount	0
RedemptionAmount	40,02
PointsBurned	20010
Rate	0,002



Response parameters:

Parameter	Value
ResultCode	0
ResponseCode	00
StatusFlag	Success
AvailableAmount	5799,67
AvailablePointNumber	2899835
BurnedPointBalance	20010
EarnedPointBalance	0
LoyaltyTransactionId	1-28Y8QAM



Merchant application actions:

- Display of a message of transaction approval to the user, as well as information regarding the yellows
- Saving of the values of the SupportReferenceID, MerchantReference, TransactionID, StatusFlag, ResponseCode, ResponseDescription, ApprovalCode, PackageNo parameters
- Update of the merchant's application on the successful transaction



Test Case 3: APPROVED TRANSACTION - ONLY EARN (VISA)

MANDATORY

Scenario: Successful sale (without installments) without redemption of yellows (that is a number of yellows will be earned through the transaction without a redemption)



When it applies:

When ResultCode=0 and StatusFlag=Success



Input parameters:

Parameter	Value
RequestType	SALE
ExpirePreauth	0
Currency	978
Installments	0
CardType	VISA or UNKNOWN
CardNumber for purchase	4020680000000114
ExpirationMonth	04
ExpirationYear	Any future
CVV2	123
Amount	108,43
ClearCardAmount	-
RedemptionAmount	-
PointsBurned	-
Rate	-



Response parameters:

Parameter	Value
ResultCode	0
ResponseCode	00
StatusFlag	Success
AvailableAmount	5801,83
AvailablePointNumber	2900919
BurnedPointBalance	0
EarnedPointBalance	1084
LoyaltyTransactionId	1-28YXNCP



Merchant application actions:

- Display of a message of transaction approval to the user, as well as information regarding the yellows collected
- Saving the values of the SupportReferenceID, MerchantReference, TransactionID, StatusFlag, ResponseCode, ResponseDescription, ApprovalCode, PackageNo parameters
- Update of the merchant's application on the successful transaction



Test Case 4: APPROVED TRANSACTION - UNAVAILABLE LOYALTY MANAGEMENT (VISA)

MANDATORY

Scenario: Successful sale (without installments) during which there was a problem upon management of yellows (that is it was sent with redemption of yellows but there was a problem in the yellow management system, and thus the redemption/earning of yellows will take place at a subsequent time)



When it applies:

When ResultCode=0 and StatusFlag=Success



Input parameters:

Parameter	Value
RequestType	SALE
ExpirePreauth	0
Currency	978
Installments	0
CardType	VISA or UNKNOWN
CardNumber for purchase	4020680000000114
ExpirationMonth	05
ExpirationYear	Any future
CVV2	123
Amount	77,59
ClearCardAmount	74,31
RedemptionAmount	3,28
PointsBurned	1640
Rate	0,002



Response parameters:

Parameter	Value
ResultCode	0
ResponseCode	00
StatusFlag	Success
AvailableAmount	-
AvailablePointNumber	-
BurnedPointBalance	-
EarnedPointBalance	-
LoyaltyTransactionId	-



Merchant application actions:

- Display of a message of transaction approval to the user without information regarding the yellows
- Saving of the values of the SupportReferenceID, MerchantReference, TransactionID, StatusFlag, ResponseCode, ResponseDescription, ApprovalCode, PackageNo parameters
- Update of the merchant's application on the successful transaction



Test Case 5: APPROVED TRANSACTION – NON-PARTICIPATING CARD IN YELLOW PROGRAM (VISA)

MANDATORY

Scenario: Successful sale or pre-authorization (without installments) with a Visa card



When it applies:

When ResultCode=0 and StatusFlag=Success



Note:

The redemption of yellows in pre-authorizations is not possible



Input parameters:

Parameter	Value
ExpirePreauth for purchase	0
ExpirePreauth for pre-authorization	30
Currency	978
Installments	0
CardType	VISA or UNKNOWN
CardNumber for purchase	490845555555557
CardNumber for pre-authorization	4020680000000098
ExpirationMonth	01
ExpirationYear	Any future
CVV2	123



Response parameters:

Parameter	Value
ResultCode	0
ResponseCode	00



Merchant application actions:

- Display of a message of transaction approval on the user's page
- Saving of the values of the SupportReferenceID, MerchantReference, TransactionID, StatusFlag, ResponseCode, ResponseDescription, ApprovalCode, PackageNo parameters
- Update of the merchant's application on the successful transaction



Test Case 6: DECLINED TRANSACTION

MANDATORY

Scenario: Declined transaction



When it applies:

When ResultCode=0 and StatusFlag=Failure



Input parameters:

Parameter	Value
ExpirePreauth for purchase	0
ExpirePreauth for pre-authorization	30
Currency	978
Installments	0
CardType	VISA or UNKNOWN
CardNumber for purchase	490845555555557
CardNumber for pre-authorization	4020680000000098
ExpirationMonth	02
ExpirationYear	<i>Any future</i>
CVV2	123



Response parameters:

Parameter	Value
ResultCode	0
ResponseCode	12



Merchant application actions:

- Display of a message of transaction decline from the issuing Bank to the user's page
- Saving of the values of the SupportReferenceID, MerchantReference, ResponseCode, ResponseDescription parameters
- Update of the merchant's application on the decline of the transaction



Test Case 7: RECHARGE ATTEMPT

MANDATORY

Scenario: Attempt to double charge a transaction (a request sent with a value for the "MerchantReference" already used in an approved transaction)



When it applies:

When StatusFlag = Failure & ResultCode=1048



Input parameters:

Parameter	Value
ExpirePreauth for purchase	0
ExpirePreauth for pre-authorization	30
Currency	978
Installments	0
CardType	VISA or UNKNOWN
CardNumber for purchase	490845555555557
CardNumber for pre-authorization	4020680000000098
ExpirationMonth	03
ExpirationYear	<i>Any future</i>
CVV2	123



Response parameters:

Parameter	Value
StatusFlag	Failure
ResultCode	1048



Merchant application actions:

- Display of a message of transaction approval on the user's page
- Saving of the values of the SupportReferenceID, MerchantReference, ResultCode, ResponseDescription parameters
- Update of the merchant's application on the attempt of double charging (if deemed necessary, in order to be checked in more detail)



Test Case 8: COMMUNICATION ERROR

MANDATORY

Scenario: Incapacity to execute a transaction due to a (technical) problem of communication with the system of transaction processing



When it applies:

When ResultCode = 50x (that is 500, 501 etc.)



Input parameters:

Parameter	Value
ExpirePreauth for purchase	0
ExpirePreauth for pre-authorization	30
Currency	978
Installments	0
CardType	VISA or UNKNOWN
CardNumber for purchase	4908455555555557
CardNumber for pre-authorization	4020680000000098
ExpirationMonth	04
ExpirationYear	<i>Any future</i>
CVV2	123



Response parameters:

Parameter	Value
ResultCode	500
ResponseCode	



Merchant application actions:

- Display of a message of incapacity to execute a transaction on the user's page (with encouragement to try again later)
- Saving of the values of the SupportReferenceID, MerchantReference, ResultCode, ResultDescription parameters
- Update of the merchant's application on the incapacity to execute the transaction



Test Case 9: INVALID CARD NUMBER

MANDATORY

Scenario: Incapacity to execute a transaction due to inserting wrong card details or details of a card not supported by the system



When it applies:

When ResultCode = 981



Input parameters:

Parameter	Value
ExpirePreauth for purchase	0
ExpirePreauth for pre-authorization	30
Currency	978
Installments	0
CardType	VISA or UNKNOWN
CardNumber for purchase	490845555555557
CardNumber for pre-authorization	4020680000000098
ExpirationMonth	05
ExpirationYear	<i>Any future</i>
CVV2	123



Response parameters:

Parameter	Value
ResultCode	981
ResponseCode	



Merchant application actions:

- Display of a message of incapacity to execute a transaction on the user's page (with encouragement to try again and check their card details or insert a new card)
- Saving of the values of the SupportReferenceID, MerchantReference, ResultCode, ResultDescription parameters
- Update of the merchant's application on the incapacity to execute the transaction



Test Case 10: UNDER-PROCESS TRANSACTION WAS RE-SENT

MANDATORY

Scenario: Attempt to send a transaction with the same "MerchantReference" as the one of a transaction already being processed by epay eCommerce (a response from the issuing Bank may have not been received or a problem may have emerged in the processing system of the transactions resulting to the transaction "getting stuck")



When it applies:

When ResultCode = 1045



Input parameters:

Parameter	Value
ExpirePreauth for purchase	0
ExpirePreauth for pre-authorization	30
Currency	978
Installments	0
CardType	VISA or UNKNOWN
CardNumber for purchase	490845555555557
CardNumber for pre-authorization	4020680000000098
ExpirationMonth	06
ExpirationYear	Any future
CVV2	123



Response parameters:

Parameter	Value
ResultCode	1045
ResponseCode	



Merchant application actions:

- Display of a message of incapacity to execute the transaction on the user's page (with encouragement to try again later)
- Saving of the values of the SupportReferenceID, MerchantReference, ResultCode, ResultDescription parameters
- Update of the merchant's application on the incapacity to execute the transaction and encouragement of the business to investigate the transaction's status through the epay eCommerce AdminTool



Note:

The encouragement of the user to try again is recommended, because, if the initial transaction is successfully executed at the end, then, on the next try, the case of Test case 7 will be reproduced.



Test Case 11: BATCH IS CLOSING

MANDATORY

Scenario: Non-execution of a transaction due to the process of clearing of the transactions of the current package (package closing) taking place at that moment



When it applies:

When ResultCode = 1072



Input parameters:

Parameter	Value
ExpirePreauth for purchase	0
ExpirePreauth for pre-authorization	30
Currency	978
Installments	0
CardType	VISA or UNKNOWN
CardNumber for purchase	4908455555555557
CardNumber for pre-authorization	4020680000000098
ExpirationMonth	07
ExpirationYear	<i>Any future</i>
CVV2	123



Response parameters:

Parameter	Value
ResultCode	1072
ResponseCode	



Merchant application actions:

- Display of a message of incapacity to execute a transaction on the user's page (with encouragement to try again later)
- Saving of the values of the SupportReferenceID, MerchantReference, ResultCode, ResultDescription parameters
- Update of the merchant's application on the incapacity to execute the transaction (if deemed necessary)



Test Case 12: GENERAL ERROR

MANDATORY

Scenario: Non-execution of a transaction due to a temporary technical problem



When it applies:

When ResultCode = 1



Input parameters:

Parameter	Value
ExpirePreauth for purchase	0
ExpirePreauth for pre-authorization	30
Currency	978
Installments	0
CardType	VISA or UNKNOWN
CardNumber for purchase	490845555555557
CardNumber for pre-authorization	4020680000000098
ExpirationMonth	08
ExpirationYear	<i>Any future</i>
CVV2	123



Response parameters:

Parameter	Value
ResultCode	1
ResponseCode	



Merchant application actions:

- Display of a message of incapacity to execute a transaction due to a temporary technical problem on the user's page (with encouragement to try again later)
- Saving of the values of the SupportReferenceID, MerchantReference, ResultCode, ResultDescription parameters
- Update of the merchant's application on the incapacity to execute the transaction (if deemed necessary)



Test Case 13: APPROVED TRANSACTION WITH INSTALLMENTS

OPTIONAL

Scenario: Approval of a transaction with installments



When it applies:

When ResultCode=0 and StatusFlag=Success



Input parameters:

Parameter	Value
ExpirePreauth for purchase	0
ExpirePreauth for pre-authorization	30
Currency	978
Installments	3
CardType	VISA or UNKNOWN
CardNumber for purchase	490845555555557
CardNumber for pre-authorization	4020680000000098
ExpirationMonth	09
ExpirationYear	Any future
CVV2	123
Amount	Over 90



Response parameters:

Parameter	Value
ResultCode	0
ResponseCode	00



Merchant application actions:

- Display of a message of transaction approval on the user's page in "x" installments (where "x" the number of installments sent)
- Saving of the values of the SupportReferenceID, MerchantReference, TransactionID, StatusFlag, ResponseCode, ResponseDescription, ApprovalCode, PackageNo parameters
- Update of the merchant's application on the successful transaction



Test Case 14: APPROVED TRANSACTION (MASTERCARD)

OPTIONAL

Scenario: Approval of transaction (without installments) with a Mastercard card



When it applies:

When ResultCode=0 and StatusFlag=Success



Input parameters:

Parameter	Value
ExpirePreauth for purchase	0
ExpirePreauth for pre-authorization	30
Currency	978
Installments	0
CardType	MasterCard or UNKNOWN
CardNumber for purchase and Pre-authorization	5194993333333335
ExpirationMonth for purchase	01
ExpirationMonth for pre-authorization	02
ExpirationYear	Any future
CVV2	123

On the screen displayed, type "Yes" for the transaction to be completed.



Response parameters:

Parameter	Value
ResultCode	0
ResponseCode	00



Merchant application actions:

- Display of a message of transaction approval on the user's page
- Saving of the values of the SupportReferenceID, MerchantReference, TransactionID, StatusFlag, ResponseCode, ResponseDescription, ApprovalCode, PackageNo parameters
- Update of the merchant's application on the successful transaction



Test Case 15: APPROVED TRANSACTION (DINERS)

OPTIONAL

Scenario: Approval of transaction (without installments) with a Diners card



When it applies:

When ResultCode=0 and StatusFlag=Success



Input parameters:

Parameter	Value
ExpirePreauth for purchase	0
ExpirePreauth for pre-authorization	30
Currency	978
Installments	0
CardType	DinersClub or UNKNOWN
CardNumber for purchase	36131111111119
CardNumber for pre-authorization	36131100000000
ExpirationMonth	01
ExpirationYear	Any future
CVV2	123



Note:

Transactions with a Diners/Discover card are sent with a "null" value in the "AuthInfo" element (the parameters Eci, Xid, Enrolled, PAResStatus, SignatureVerification are included – see section 6).



Response parameters:

Parameter	Value
ResultCode	0
ResponseCode	00



Merchant application actions:

- Display of a message of transaction approval on the user's page
- Saving of the values of the SupportReferenceID, MerchantReference, TransactionID, StatusFlag, ResponseCode, ResponseDescription, ApprovalCode, PackageNo parameters
- Update of the merchant's application on the successful transaction



Test Case 16: APPROVED TRANSACTION (DISCOVER)

OPTIONAL

Scenario: Approval of transaction (without installments) with a Discover card



When it applies:

When ResultCode=0 and StatusFlag=Success



Input parameters:

Parameter	Value
ExpirePreauth for purchase	0
ExpirePreauth for pre-authorization	30
Currency	978
Installments	0
CardType	DinersClub or UNKNOWN
CardNumber for purchase	601111111111117
CardNumber for pre-authorization	6011000000000004
ExpirationMonth	01
ExpirationYear	Any future
CVV2	123



Note:

Transactions with a Diners/Discover card are sent with a "null" value in the "AuthInfo" element (the parameters Eci, Xid, Enrolled, PAREsStatus, SignatureVerification are included – see section 6).



Response parameters:

Parameter	Value
ResultCode	0
ResponseCode	00



Merchant application actions:

- Display of a message of transaction approval on the user's page
- Saving of the values of the SupportReferenceID, MerchantReference, TransactionID, StatusFlag, ResponseCode, ResponseDescription, ApprovalCode, PackageNo parameters
- Update of the merchant's application on the successful transaction



Test Case 17: APPROVED TRANSACTION (AMERICAN EXPRESS)

OPTIONAL

Scenario: Approval of transaction (without installments) with an American Express card



When it applies:

When ResultCode=0 and StatusFlag=Success



Input parameters:

Parameter	Value
ExpirePreauth for purchase	0
ExpirePreauth for pre-authorization	30
Currency	978
Installments	0
CardType	AMEX or UNKNOWN
CardNumber for purchase	375537111111116
CardNumber for pre-authorization	375537000000008
ExpirationMonth	01
ExpirationYear	Any future
CVV2	1234



Note:

Transactions with an American Express card are sent with a "null" value in the "AuthInfo" element (the parameters Eci, Xid, Enrolled, PResStatus, SignatureVerification are included – see section 6).



Response parameters:

Parameter	Value
ResultCode	0
ResponseCode	00



Merchant application actions:

- Display of a message of transaction approval on the user's page
- Saving of the values of the SupportReferenceID, MerchantReference, TransactionID, StatusFlag, ResponseCode, ResponseDescription, ApprovalCode, PackageNo parameters
- Update of the merchant's application on the successful transaction



Test Case 18: APPROVED TRANSACTION (GBP)

OPTIONAL



Attention!

For each different currency, a different test and live account is required

Scenario: Approval of transaction (without installments) in GBP



When it applies:

When ResultCode=0 and StatusFlag=Success



Input parameters:

Parameter	Value
ExpirePreauth for purchase	0
ExpirePreauth for pre-authorization	30
Currency	826
Installments	0
CardType	VISA or UNKNOWN
CardNumber for purchase and pre-authorization	4908456666666663
ExpirationMonth for purchase	01
ExpirationMonth for pre-authorization	02
ExpirationYear	<i>Any future</i>
CVV2	123

On the screen displayed, type "Yes" for the transaction to be completed.



Response parameters:

Parameter	Value
ResultCode	0
ResponseCode	00



Merchant application actions:

- Display of a message of transaction approval on the user's page
- Saving of the values of the SupportReferenceID, MerchantReference, TransactionID, StatusFlag, ResponseCode, ResponseDescription, ApprovalCode, PackageNo parameters
- Update of the merchant's application on the successful transaction



Test Case 19: APPROVED TRANSACTION (USD)

OPTIONAL



Attention!

For each different currency, a different test and live account is required

Scenario: Approval of transaction (without installments) in USD



When it applies:

When ResultCode=0 and StatusFlag=Success



Input parameters:

Parameter	Value
ExpirePreauth for purchase	0
ExpirePreauth for pre-authorization	30
Currency	840
Installments	0
CardType	VISA or UNKNOWN
CardNumber for purchase and pre-authorization	4908456666666663
ExpirationMonth for purchase	03
ExpirationMonth for pre-authorization	04
ExpirationYear	Any future
CVV2	123

On the screen displayed, type "Yes" for the transaction to be completed.



Response parameters:

Parameter	Value
ResultCode	0
ResponseCode	00



Merchant application actions:

- Display of a message of transaction approval on the user's page
- Saving of the values of the SupportReferenceID, MerchantReference, TransactionID, StatusFlag, ResponseCode, ResponseDescription, ApprovalCode, PackageNo parameters
- Update of the merchant's application on the successful transaction

10. Security Requirements

With regard to the security requirements that must be met by the merchant system, the following should be considered:

- According to the Visa/Mastercard organizations specifications, no card detail (i.e. card number, expiry date, cvv2) must be stored in the merchant system.
- If transactions are performed via a site, then SSL encryption with min. 128-bit key size should be used on the page where the user enters his/her card details, so that they may be transferred securely.



Note:

In order for the live account's data to be sent, the SSL must have been used on the live site in order for it to be checked by Euronet Merchant Services.

- The use of ssl on the card details entry page of the merchant site must be visible to the user through the relevant icons used by the various browsers. Therefore, the card details entry page may not be in any kind of frame (FrameSet, IFrame) because the secure address of the page with the respective browser symbols indicating its validity and security are suppressed. On the contrary, in this case the Frame parent page address is displayed, which might not be secure thus creating a wrong impression to the user.
- For transactions via a site, characters should not be visible when typed by the user in the cvv2 field (e.g. asterisks should be displayed instead).



11. Use of Icons

In case the transactions are executed through a website, it is important that the necessary icons are used – in accordance with the specifications of the Visa and Mastercard Organizations.

All relevant material can be downloaded from the following link:

<https://paycenter.piraeusbank.gr/services/Manuals/Icons/Icons.zip>

More specifically:

Icons of supported cards

The icons of the cards supported are included in the folder (Icons/CardsIcons) and are the following:

Visa (<i>Visa.png</i>)
Mastercard (<i>Mastercard.png</i>)
Maestro (<i>Maestro.png</i>)

If Diners/Discover or/and American Express cards are also supported, then, the respective icons should be included:

Diners (<i>Diners.jpg</i>)
Discover (<i>Discover.jpg</i>)
American Express (<i>Amex.jpg</i>)

The above icons should be displayed on the homepage of the website.

3D Secure icons

In the case of online transactions through a website, the following 3D Secure icons should be displayed on the website homepage, on the page where card details are entered and on the security information page:

- **Visa Secure service:**
One of the icons located in (Icons/Visa Secure) should be displayed.
- **Mastercard Identity Check:**
One of the icons located in (Icons/Mastercard Identity Check) should be displayed.

epay Logo

Logo of epay may optionally be displayed on the merchant site. The relevant icons are included in the (Icons/epay) folder.

12. Tips

Below, some remarks – tips are provided, that have to be taken into account:




- 💡 A preauthorization transaction can be completed by the merchant (either via the epay eCommerce AdminTool or by calling the Transaction Web Service) within the days defined via the "ExpirePreauth" parameter (maximum 30 days). After that period of time, the preauthorization cannot be settled.
- 💡 Refund transactions can be carried out through either the epay eCommerce AdminTool (web application provided to all merchants) or a Web Service call.
- 💡 According to the Visa/Mastercard organizations specifications, no card detail (i.e. card number, expiry date, cvv2) must be stored in the merchant system.
- 💡 The "Password" parameter in the "Transaction Web Service" (see section 5) must be sent encrypted with the MD5 hashing algorithm.
- 💡 The "**MerchantReference**" parameter in the "Transaction Web Service" should have a unique/different value in each new sale or pre-authorisation transaction. Even if the transaction fails and a new attempt is made (i.e. a new transaction), the 3D Secure process should be repeated (if supported) and the Transaction Web Service call should contain a new "Merchant Reference" value.
- 💡 It is important that the "**MerchantReference**" parameter has a value that has a special meaning and is known to the merchant (e.g. order number, contract number, etc.). This value, uniquely designating every successful transaction, appears in the "AdminTool" provided by Euronet Merchant Services to merchants to monitor their transactions. Using the "AdminTool" merchants can find transactions using the "**MerchantReference**" value as search criterion.
- 💡 For better merchant support by Euronet Merchant Services, the "**SupportReferenceID**" parameter should be stored with every attempt and be available to the merchant managers, so that it may be used in the communication with Euronet Merchant Services to solve potential problems. The same parameter must be sent by technicians to Euronet Merchant Services in the event of issues during test transactions.
- 💡 Interest-free installments are only supported by certain cards issued by Greek banks (depending on BIN, i.e. the first 6 digits of the card number). Euronet Merchant Services provides the "**BIN Web Service**" API which can be used in order to check if a card supports installments without sending a charge transaction. In case of interest, the technical specifications should be requested from Euronet Merchant Services.
- 💡 If communication with epay eCommerce is interrupted and no response is received by the merchant system, a refund request may be submitted (RequestType = "REFUND") sending a value in the "**MerchantReference**" parameter instead of the "**TransactionReferenceID**" parameter. This

functionality is only provided for transaction cancellations (i.e. cancellation of transactions included in an open package).



13. Implementation Checklist

S/N	TASK
1.	⇒ SIGNING OF CONTRACT Signing of acquiring contract for the “Web Service” solution between the company and Euronet Merchant Services.
2.	⇒ TECHNICAL IMPLEMENTATION Implementation of: <ul style="list-style-type: none">■ Strong customer authentication process for online card transactions with Visa, Mastercard and Maestro cards through a website (“3d-secure” - see section 6)■ Software calling the “LoyaltyMemberBalance Service” and the “Transaction Web Service”
3.	⇒ SENDING INFORMATION FOR TEST ACCOUNT Sending to Euronet Merchant Services the necessary information for the creation of a test account (see section 3)
4.	⇒ IMPLEMENTATION OF TEST TRANSACTIONS <ul style="list-style-type: none">■ Euronet Merchant Services forwards the following test account details:<ul style="list-style-type: none">■ AcquirerID■ MerchantID■ PosID■ User■ Password■ ChannelType■ <u>Only for systems supporting online transactions through a website:</u> Execution of all test cases of the 3D Secure process (see relevant documentation).■ Implementation of test transactions with the “Transaction Web Service” (see section 9).
5.	⇒ USE OF ICONS <u>For systems sending transactions through a website:</u> Posting the necessary icons on the merchant website (see section 11)


6.	<p> COMPLETION OF TEST TRANSACTIONS</p> <ul style="list-style-type: none"> ■ Notification of Euronet Merchant Services regarding the successful completion of the test transactions, the use of icons and ssl and sending the data of the test transactions to be checked by Euronet Merchant Services. More specifically, the following shall be sent: <ul style="list-style-type: none"> ▪ The "MerchantReference" value for each test case of the 3D Secure process (see relevant documentation). ▪ The "SupportReferenceID" value for each test case of calling the Transaction Web Service (see section 9). ■ Check of the test transactions by Euronet Merchant Services and notification of the technical manager for the result within a week. ■ Sending to Euronet Merchant Services the IP address of the server from which real transactions will be sent (the business's production system). ■ Sending to Euronet Merchant Services an e-mail address of the business for receiving e-mails regarding epay eCommerce.
7.	<p> RECEIPT OF LIVE ACCOUNT</p> <ul style="list-style-type: none"> ■ Euronet Merchant Services forwards live account details: <ul style="list-style-type: none"> ▪ AcquirerID ▪ MerchantID ▪ PosID ▪ User ▪ Password ▪ ChannelType ■ Replace the test account details with the live account. <div data-bbox="368 1279 1362 1464" style="border: 1px solid black; padding: 10px; margin-top: 10px;"> <p> Note: The URL of the Transaction Web Service is the same both for the test and for the real transactions: https://paycenter.piraeusbank.gr/services/paymentgateway.asmx</p> </div>

> Annex 1


Below, the most frequent values for the parameters "ResultCode" (that is the technical problems that may emerge) and "ResponseCode" are presented (that is the most frequent responses sent by the issuing banks).

ResultCode FREQUENT VALUES			
ResultCode	ResultDescription	Explanation	Action
1	An error occurred. Please check your data or else contact epay eCommerce administrator	General error message displayed in case of a technical problem at epay eCommerce.	Try again later when the problem has been rectified
100	Authentication Error	A wrong value has been sent for the "Username" and/or "Password" parameter.	The business's technician must make the necessary corrections in order for the correct values to be sent.
130	Field "x" contains invalid characters	The field "x" contains non-acceptable characters	The business's technician must make the necessary corrections in order for the right value to be sent for the field "x".
151	Check that field "x" contains data	There is no value in field "x".	The business's technician must make the necessary corrections in order for a value to be sent for the field "x".
215	AMEX cards require 4-digit cvv2	Concerns a transaction with an American Express card. Cvv2 shall consist of 4 digits.	The transaction has to be re-sent with the correct Cvv2.
216	Wrong cvv2	Not acceptable value in the cvv2 parameter ("Cvv2"), e.g., characters.	The business's technician must make the necessary corrections in order for an acceptable value to be sent for cvv2 (e.g. only numerical characters accepted) or the card holder must repeat the transaction sending the correct Cvv2.

50x (e.g. 500, 501 etc.)	Communication Error	Communication problem with the transaction processing system	Try again later when the problem has been rectified
981	Invalid Card number/Exp Month/Exp Year	Insertion of wrong card details (e.g. wrong number, wrong card type, old expiration date) or of details of a card not supported by the system.	The card details must be filled out correctly
1006	Unknown BIN	Concerns a transaction with installments. The user's card does not participate in the program of interest-free installments.	Another card must be used or the transaction must be repeated without installments
1007	Merchant does not support given bin	Concerns a transaction with installments. The card's bin (that is the first 6 digits) cannot be used in a transaction with installments in the specific business.	Another card must be used or the transaction must be repeated without installments
1010	Wrong original transaction	It concerns settlement transactions (SETTLEMENT), preauthorization cancellations (VOIDREQUEST), refund transactions (REFUND) or follow up requests (FOLLOW_UP). The request is rejected because there is no successful transaction for which the settlement, the preauthorization cancelation, the refund transaction or the follow up is asked.	The initial transaction must be checked and the correct value must be sent for the "TransactionReferenceID".

1012	Original transaction already settled, or being settled	It concerns a preauthorization settlement transaction («SETTLEMENT»). A settlement is requested for a preauthorization which has already been settled or is being settled.	It shall be checked through the AdminTool whether the pre-authorization is indeed completed.
1014	Refunding amount cannot exceed remaining amount of the original transaction	Concerns a transaction of refund ("REFUND"). The refunding amount exceeds the amount of the initial charging transaction.	<p>The refund must be repeated with the correct amount.</p> <div>  Note: Many partial refunds can be made, provided that the total of the amounts of all the refunds does not exceed the amount of the initial charging transaction. </div>
1017	Preorder date has expired	It concerns a preauthorization settlement transaction («SETTLEMENT»). The settlement cannot be carried out because the preauthorization has expired.	A new pre-authorization must be executed.
1019	Too many installments asked	The number of installments used is greater than the maximum amount allowed for this business.	Fewer installments must be used
1026	Merchant does not support instalments	Installments were used in the transaction but the merchant does not support installments.	Communication with Euronet Merchant Services is required in order to activate installments.
1034	Terminal does not support given card type	The transaction was sent with a non-supported card type.	It shall be checked whether the correct card type was sent and communication

			with Euronet Merchant Services is required.
1040, 1041	"Error validating IP address. Contact sysadmin." (1040), "Invalid IP address." (1041)	The transaction was sent from an IP address different than the one declared by the technical manager.	The IP address from which the requests are sent shall be checked and if it needs to be changed, communication with Euronet Merchant Services is required.
1042	Refund maximum allowed period exceeded	An attempt for refund (REFUND) was made after the allowed period of <u>365 days</u> .	Communication with Euronet Merchant Services is required
1045	Duplicate transaction references are not allowed	Concerns a transaction of sale or pre-authorization. The transaction was sent with a value for "MerchantReference" which is used in another transaction being processed at that moment.	A new transaction shall be executed later so that the transaction being processed is completed. If the first transaction was finally approved, error 1048 will return, otherwise the new transaction will be normally executed. Alternatively, one can check through the AdminTool whether the initial transaction being processed has been approved.
1048	Transaction already processed and completed	A successful transaction has already been executed with the specific "MerchantReference", i.e. an attempt of double charging was made	A new transaction with a different "MerchantReference" shall be sent.
1072	Pack is still closing	The batch settlement process (batch closing) takes place, during which the processing of transactions is not possible.	New attempt later after the batch has been closed.
1802	Wrong amount value	Non-acceptable value for the parameter of the amount ("Amount"), e.g. zero amount.	The business's technician must make the necessary corrections in order for an acceptable value to be sent for the amount.

7001	<Code of the anti-fraud rule activated>	The request was rejected due to anti-fraud checks. The «ResultDescription» parameter contains the code of the rule that was fired-up. <u>The zero value (0) means that the card number is included in a black list.</u> If special anti-fraud rules have been agreed with the merchant, Euronet Merchant Services will provide the relevant rule codes that may be returned.	<p>Encouragement of the for another form of payment or ask for a different card.</p> <div>  Attention! The final user shall not be notified that the transaction was declined due to anti-fraud checks. </div>
-------------	---	--	---

ResponseCode FREQUENT VALUES				
ResponseCode	ResponseDescription	Explanation	Action	Approval of transaction
05	Declined	The transaction was declined by the issuing bank	Communication of the card holder with their Bank or use of another card	No
12	Declined	The transaction was declined by the issuing bank	Communication of the card holder with their Bank or use of another card	No
51	Declined	The transaction was declined by the issuing bank	Communication of the card holder with their Bank or use of another card	No

34 43	Lost card Stolen card, pick-up	The transaction was declined by the issuing bank	Communication of the card holder with their Bank or use of another card	No
54	Expired card	The card has expired and has not been renewed	Use of another card	No
62	Restricted Card	The transaction was declined by the issuing bank	Communication of the card holder with their Bank or use of another card	No
92	Declined	Communication problem with the the payment Organisation (Visa, Mastercard, etc.)	A new attempt shall be made later	No
12	Installment amount bellow allowed minimum	Concerns a transaction with installments and the value of each installment is lower than the minimum value allowed	The transaction shall be repeated with fewer installments	No



Note:

More values may be returned in addition to the ones listed in the tables above.

> Annex 2

The codes of the supported currencies are the following:

Currency code	Currency
008	ALBANIAN LEK (ALL)
032	ARGENTINA PESO (ARS)
036	AUSTRALIAN DOLLAR (AUD)
124	CANADIAN DOLLAR (CAD)
152	CHILEAN PESO (CLP)
156	CHINESE YUAN (CNY)
170	COLOMBIAN PESO (COP)
191	CROATIAN KUNA (HRK)
203	CZECH KORUNA (CZK)
208	DANISH KRONE (DKK)
344	HONG KONG DOLLAR (HKD)
348	FIORINT (HUF)
356	INDIAN RUPEE (INR)
360	RUPIAH (IDR)
376	ISRAELI NEW SHEQEL (ILS)
392	YEN (JPY)
398	TENGE (KZT)
410	WON (KRW)
414	KUWAITI DINAR (KWD)
440	LITHUANIAN LITAS (LTL)
446	PATACA (MOP)
458	MALAYSIAN RINGGIT (MYR)
484	MEXICAN PESO (MXN)
504	MORROCAN DIRHAM (MAD)
554	NEW ZEALAND DOLLAR (NZD)
578	NORWEGIAN KRONE (NOK)
604	NUEVO SOL (PEN)
608	PHILIPPINE PESO (PHP)
643	RUSSIAN ROUBLE (RUB)
682	SAUDI RIYAL (SAR)
702	SINGAPORE DOLLAR (SGD)
710	RAND (ZAR)

752	SWEDISH KRONA (SEK)
756	SWISS FRANC (CHF)
764	BAHT (THB)
784	UNITED ARAB EMIRATES DIRHAM (AED)
818	EGYPTIAN POUND (EGP)
826	POUND STERLING (GBP)
840	US DOLLAR (USD)
933	BELARUSIAN RUBLE (BYN)
937	BOLIVAR FUERTE (VEF)
941	SERBIAN DINAR (RSD)
946	ROMANIAN LEU (RON)
949	TURKISH LIRA (TRY)
975	BULGARIAN LEV (BGN)
978	EURO (EUR)
980	UKRAINIAN HRYVNIA (UAH)
985	POLISH ZLOTY (PLN)
986	BRAZILIAN REAL (BRL)



Glossary

3d-secure	The name of the protocol used in the strong customer authentication process ("Visa Secure" and "Mastercard Identity check" by Visa and Mastercard respectively).
Acquirer	An organization enabling merchants to execute card transactions. Euronet Merchant Services in this case.
BIN	The first 6 digits of a card number identifying the Issuer bank.
Live account	The merchant account through which live transactions are executed. It comprises the following: <ul style="list-style-type: none">▪ AcquirerID▪ MerchantID▪ PosID▪ User▪ Password▪ ChannelType
Merchant id	The " <i>merchant code</i> " corresponding to the business.
Pos id	The " <i>pos identification</i> " (P oint O f S ale) of the merchant.
Test account	The test account provided by Euronet Merchant Services through which test transactions are carried out. It comprises the same elements as the "live account" but has different values.
Transaction Web Service	Euronet Merchant Services SOAP Web Service via which transactions are submitted to epay eCommerce.
epay eCommerce	The electronic payment system of Euronet Merchant Services.