BankStore XML Integration

Version 1.0

01/01/2012

PAYTPV

DOCUMENT CONTENT

1. DOCUMENT CHANGES

Author	Date	Version	Modifications
(W.E.D.) Web Engineering Department	2012-01-01	V1.0	First version
(W.E.D.)	2012-02-03	V1.0b	Small changes in *create_subscription* and *edit_subscription*

2. INTRODUCTION

The purpose of this document is to serve as a reference during the integration process of a trade with **PAYTPV** payment gateway, using the system that stores bank data of customers called: **BankStore**.

It also includes an appendix of reference and is available on the website of **PAYTPV** a document with the error codes returned by the system to simplify the debugging process.

XML integration through service enables access to **BankStore** payment transactions without direct user intervention, extending the individual collection operational management of customer bank details of trade. The requests are based on the trade server and processed by the payment gateway.

The chosen technology for performing operations is **SOAP**, XML-based and allows safe passage through firewalls and through the network.

BankStore service is differentiated by its trade and communication between the gateway combining information on customer data. Trade stores personal data of your customer and bank information **PAYTPV**. Once the trade send the banking information of the customer, **PAYTPV** returns a handle formed by a unique customer number and a token.

This trade has a common identifier PAYTPV to make arrangements on your account as:

- User registration (***add_user***)
- User information (***info_user***)
- User removal (***remove_user***)
- Purchase execution (***execute_purchase***)

In the same service has been added operational subscription. This operation is completely parallel to the storage of customer bank details. The management of subscriptions will be made with the following functions:

- Subscription registration (***create_subscription***)
- Subscription edition (***edit_subscription***)
- Subscription removal (***remove_subscription***)

Thus two concepts are divided in the same service that is necessary to clarify their purpose and differences:

BankStore	***BankStore Subscription***
Storing client data bank trade	Managing subscriptions with data stores trade customer bank
Ideal for businesses with the same client habitual purchase without having to re-enter card details	Ideal for subscriptions to a service or installment payments. Important: PAYTPV not be responsible for refusal of payment that may occur in assessments.
Refunds can be made via XML SOAP	Refunds must be made from the Control Panel
The registration of the user can only be made via XML SOAP	The registration of a subscription can be done through XML SOAP or WEB POS (IFRAME or FULLSCREEN).

3. TRACK PURCHASE PROCESS

The buying process is conducted by the payment gateway and the resolution of the operation is returned directly in the response message. However you can set through the **PAYTPV** Control Panel additional notification methods to maintain knowledge of the operation to users or launch additional mechanisms.

These notifications are identical to those submitted by other payment gateway products, such as Web gateway or Telephone Sales. This makes it possible for the customer to maintain a unified control of the sales made by trade.

For proper notification of the payment process, you can configure the notification system to notify you of the status of the operation conducted by the gateway, by e-mail, call to a URL in the background (independent of the response process service) or both and globally via SMS.

3.1. Email

The information contained in a notification email is as follows:

There was a sale by the following parameters:

Account identifier: 0gs265nc

Kind of transaction: Authorization (1)

Card Country: ES

Time stamp of transaction (yyyymmddhhmmss): 20101027110536

Order: 2010102711053676

Response: OK

Error ID: 0

Error Description:

Authentication number: 802335/120098123810102711053606007000

Currency: EUR

Amount (euros): 10.00

Amount (original): 1000

Language: es

Product ID: 25

Hash:

3.2. URL Callback

The destination URL is passed to the notification parameters with the POST method according to the following table:

TransactionType	Numeric	Operation Type (ANNEX III –OPERATION TYPES)
TransactionName	Alphanumeric	Operation Type
CardCountry	Alphanumeric	Card country, can be void
BankDateTime	Alphanumeric	Operation Time stamp
Signature	Alphanumeric	Hash
Order	Alphanumeric	Order by trade
Response	Alphanumeric	Response of the operation
ErrorID	Numeric	Error code. See appendix.
ErrorDescription	Alphanumeric	Error description (optional)
AuthCode	Alphanumeric	Authcode from bank (optional)
Currency	Alphanumeric	Operation currency
Amount	Numeric	Operation amount
AmountEur	Numeric	Operation amount in €
Language	Alphanumeric	Language

AccountCode	Alphanumeric	Account code
TpvID	Numeric	Product identifier

For complete control of the payment process will be a good idea use a processed notifications URL. However, it is possible to maintain manual control of collections received by the customer control panel (https://www.paytpv.com/clientes.php).

5. POS CONFIGURATION

To use the payment gateway **PAYTPV**, the trade must be in possession of the necessary configuration parameters. These can be obtained through the customer management platform of **PAYTPV**, in https://www.paytpv.com/clientes.php.

Once inside the PAYTPV platform, you can see the configuration in the option *Mis productos→Configurar producto.*

Once clicked *"Editar"* ( ) button of the selected product, you will see a panel with the basic information of your product under *"Configuración técnica del TPV WEB"*. The technical data to configure your integration will be:

- Contraseña (*Password*)
- Número de terminal (*Terminal ID*)
- Código de cliente (*Account code*)
- URL del servicio (***https://www.paytpv.com/gateway/xml_bankstore.php***) (Service URL)
- URL del archivo de descripción del servicio web (WSDL) a modo de referencia (***https://www.paytpv.com/gateway/xml_bankstore.php?wsdl*) *(WSDL URL)***

6.

operations. There is broad support for the implementation of SOAP calls for major programming languages used in web environments.

The requests are made through the HTTPS transport protocol, so you must ensure that your system is capable of performing correctly requests and manage security certificates returned by the platform for proper use.

There are several operations that can be launched from the same service.

The available operations are described below.

6.1 ADDING A USER IN THE SYSTEM

Function: (**add_user**)

The variables required to register a user with his bank data are (in this order):

Element	Content	Description
DS_MERCHANT_MERCHANTCODE	[A-Za-z0-9]{1,8}	Required. Account code
DS_MERCHANT_TERMINAL	[0-9]{1,4}	Required. Terminal ID
DS_MERCHANT_PAN	[0-9]{16,19}	Required. PAN without spaces
DS_MERCHANT_EXPIRYDATE	[0-9]{4}	Required. Expiration date in “mmyy” (2 characters month and 2 characters year)
DS_MERCHANT_CVV2	[0-9]{3,4}	Required. CVC2 code
DS_MERCHANT_MERCHANTSIGNATURE	[a-zA-Z0-9]{40}	Required. See **HASH CALCULATION**
DS_ORIGINAL_IP	A.B.C.D	Required. IP Address of the final customer.

The description of the column "Content" refers to the regular expression that must meet the information in the corresponding element (except in DS_ORIGINAL_IP, who means IP Address “192.168.1.254”). For example “[A-Z]” means from “A” to “Z” and numbers inside brackets means string length. For example “1{2,4}” validate the numbers “11”, “111” and “1111”.

The service response to the request is done by returning an array formatted in XML with the different elements are described in the following table:

DS_IDUSER	[0-9]{1,13}	Unique identifier of the user logged into the system. Return empty on error.
DS_TOKEN_USER	[A-Za-z0-9]{1,20}	Token code associated to DS_IDUSER.
DS_ERROR_ID	[0-9]{1,5}	On error, here come the error code generated. If no error will be 0 or empty. The error codes are

		specified in the document: **Appendix I - error codes**, available in the "Documentation" on the control panel.

If the request was an error of some sort (wrong signature, wrong amount, user not found, etc.) all fields are empty, except "DS_ERROR_ID" containing the error code.

If the transaction is successful, the service will return the field DS_TOKEN_USER and DS_IDUSER that must be stored by trade, associating it with their client, later to make charges on his credit / debit card.

6.2 USER INFORMATION

Function: (**info_user**)

This function is used to confirm to customers the card which will perform the recovery. This step is optional but it is convenient to avoid mistrust.

The variables required to request information from a user are (in this order):

Element	Content	Description
DS_MERCHANT_MERCHANTCODE	[A-Za-z0-9]{1,8}	Required. Account code
DS_MERCHANT_TERMINAL	[0-9]{1,4}	Required. Terminal ID
DS_IDUSER	[0-9]{1,13}	Required. Unique identifier of the registered user in the system.
DS_TOKEN_USER	[A-Za-z0-9]{1,20}	Required. Token code associated to DS_IDUSER.
DS_MERCHANT_MERCHANTSIGNATURE	[a-zA-Z0-9]{40}	Required. **See** **HASH CALCULATION**
DS_ORIGINAL_IP	A.B.C.D	Required. IP Address of the final customer.

The description of the column "Content" refers to the regular expression that must meet the information in the corresponding element (except in DS_ORIGINAL_IP, who means IP Address "192.168.1.254"). For example "[A-Z]" means from "A" to "Z" and numbers inside brackets means string length. For example "1{2,4}" validate the numbers "11", "111" and "1111".

The service response to the request is done by returning an array formatted in XML with the different elements are described in the following table:

DS_MERCHANT_PAN	[0-9]{16,19}	Card PAN without spaces. The string will be masked with the last 4 digits. The rest will be filled with X.
DS_ERROR_ID	[0-9]{1,5}	On error, here come the error code generated. If no error will be 0 or empty. The error codes are

		specified in the document: **Appendix I - error codes**, available in the "Documentation" on the control panel.

If the request was an error of some sort (wrong signature, wrong amount, user not found, etc.) all fields are empty, except "DS_ERROR_ID" containing the error code.

If the transaction is successful, the service will return the field DS_MERCHANT_PAN containing credit card / debit masked, leaving visible only the last 4 digits.

This operation is very useful to show the customer's trade with which card is to make the transaction.

6.3 USER REMOVAL

Function: (**remove_user**)

This function is used to delete a trade user account.

The variables required to request removal of a user are (in order):

Element	Content	Description
DS_MERCHANT_MERCHANTCODE	[A-Za-z0-9]{1,8}	Required. Account code
DS_MERCHANT_TERMINAL	[0-9]{1,4}	Required. Terminal ID
DS_IDUSER	[0-9]{1,13}	Required. Unique identifier of the registered user in the system.
DS_TOKEN_USER	[A-Za-z0-9]{1,20}	Required. Token code associated to DS_IDUSER.
DS_MERCHANT_MERCHANTSIGNATURE	[a-zA-Z0-9]{40}	Required. **See HASH CALCULATION**
DS_ORIGINAL_IP	A.B.C.D	Required. IP Address of the final customer.

The description of the column "Content" refers to the regular expression that must meet the information in the corresponding element (except in DS_ORIGINAL_IP, who means IP Address "192.168.1.254"). For example "[A-Z]" means from "A" to "Z" and numbers inside brackets means string length. For example "1{2,4}" validate the numbers "11", "111" and "1111".

The service response to the request is done by returning an array formatted in XML with the different elements are described in the following table:

DS_RESPONSE	[0-1]{1}	Result of the operation. **0** or **void** means error and **1** successful operation.
DS_ERROR_ID	[0-9]{1,5}	On error, here come the error code generated. If no

		error will be 0 or empty. The error codes are specified in the document: **Appendix I - error codes**, available in the "Documentation" on the control panel.

If the request was an error of some sort (wrong signature, wrong amount, user not found, etc.) all fields are empty, except "DS_ERROR_ID" containing the error code.

If the transaction is successful, the service will return DS_RESPONSE field whose value is **0** or **empty** on failure and **1** on success in eliminating.

6.4 EXECUTING A PURCHASE

Function: (**execute_purchase**)

Once the user is stored into the system, they can make charges to your account by sending their credentials and transaction data.

The variables required to make a payment to a user registered in the system are (in this order):

Element	Content	Description
DS_MERCHANT_MERCHANTCODE	[A-Za-z0-9]{1,8}	Required. Account code
DS_MERCHANT_TERMINAL	[0-9]{1,4}	Required. Terminal ID
DS_IDUSER	[0-9]{1,13}	Required. Unique identifier of the registered user in the system.
DS_TOKEN_USER	[A-Za-z0-9]{1,20}	Required. Token code associated to DS_IDUSER.
DS_MERCHANT_AMOUNT	[0-9]{1,8}	Required. Amount of the operation in int format (not float) 1,00 EURO = 100, 4,50 EUROS = 450...
DS_MERCHANT_ORDER	[A-Za-z0-9]{1,20}	Required. Order of the trade. Must to be unique in valid transactions.
DS_MERCHANT_CURRENCY	[EUR][USD][GBP] [JPY]	Required. Currency of the transaction. See more information in **ANNEX I**
DS_MERCHANT_MERCHANTSIGNATURE	[a-zA-Z0-9]{40}	**Required. See HASH CALCULATION**
DS_ORIGINAL_IP	A.B.C.D	Required. IP Address of the final customer.
DS_MERCHANT_PRODUCTDESCRIPTION	[a-zA-Z0-9]{40}	Optional. Product description
DS_MERCHANT_OWNER	[a-zA-Z0-9]{40}	Optional. Transaction description

The description of the column "Content" refers to the regular expression

that must meet the information in the corresponding element (except in DS_ORIGINAL_IP, who means IP Address "192.168.1.254"). For example "[A-Z]" means from "A" to "Z" and numbers inside brackets means string length. For example "1{2,4}" validate the numbers "11", "111" and "1111".

The service response to the request is done by returning an array formatted in XML with the different elements are described in the following table:

DS_MERCHANT_AMOUNT	[0-9]{1,8}	Amount of the operation in int format (not float) 1,00 EURO = 100, 4,50 EUROS = 450...
DS_MERCHANT_ORDER	[A-Za-z0-9]{1,20}	Order of the trade.
DS_MERCHANT_CURRENCY	[EUR][USD][GBP][JPY]	Currency of the transaction. See more information in **ANNEX I**
DS_MERCHANT_AUTHCODE	[a-zA-Z0-9]{1,40}	Authcode of the transaction (needed to make refunds).
DS_MERCHANT_CARDCOUNTRY	[0-9]{1,3}	Country of the card in ISO3 code (ex: 724 = Spain). Can be empty.
DS_RESPONSE	[0-1]{1}	Result of the operation. **0** or **void** means error and **1** successful operation.
DS_ERROR_ID	[0-9]{1,5}	On error, here come the error code generated. If no error will be 0 or empty. The error codes are specified in the document: **Appendix I - error codes**, available in the "Documentation" on the control panel.

If the request was an error of some sort (wrong signature, wrong amount, user not found, etc.) all fields are empty, except "DS_ERROR_ID" containing the error code.

If the transaction is successful, the service will return all fields (except DS_MERCHANT_CARDCOUNTRY which is optional) to be stored by trading ahead of a possible refund. In this case the field will be returned DS_ERROR_ID empty.

6.5 REFUNDS

Function: (**execute_refund**)

You can execute refunds with this function. You will need the data identifying the user and the bank authorization code.

The variables required to perform a returning operation are (in order):

Element	Content	Description
DS_MERCHANT_MERCHANTCODE	[A-Za-z0-9]{1,8}	Required. Account code
DS_MERCHANT_TERMINAL	[0-9]{1,4}	Required. Terminal ID

DS_IDUSER	[0-9]{1,13}	Required. Unique identifier of the registered user in the system.
DS_TOKEN_USER	[A-Za-z0-9]{1,20}	Required. Token code associated to DS_IDUSER.
DS_MERCHANT_AUTHCODE	[a-zA-Z0-9]{1,40}	Required. Original AuthCode from the transaction
DS_MERCHANT_ORDER	[A-Za-z0-9]{1,20}	Required. Original order of the transaction.
DS_MERCHANT_CURRENCY	[EUR][USD][GBP] [JPY]	Currency of the transaction. See more information in **ANNEX I**
DS_MERCHANT_MERCHANTSIGNATURE	[a-zA-Z0-9]{40}	Required**. See HASH CALCULATION**
DS_ORIGINAL_IP	A.B.C.D	Required. IP Address of the trade system.

The description of the column "Content" refers to the regular expression that must meet the information in the corresponding element (except in DS_ORIGINAL_IP, who means IP Address "192.168.1.254"). For example "[A-Z]" means from "A" to "Z" and numbers inside brackets means string length. For example "1{2,4}" validate the numbers "11", "111" and "1111".

The service response to the request is done by returning an array formatted in XML with the different elements are described in the following table:

DS_MERCHANT_ORDER	[A-Za-z0-9] {1,20}	Order of the trade.
DS_MERCHANT_CURRENCY	[EUR][USD] [GBP][JPY]	Currency of the transaction. See more information in **ANNEX I**
DS_MERCHANT_AUTHCODE	[a-zA-Z0-9] {1,40}	Authcode of the transaction.
DS_RESPONSE	[0-1]{1}	Result of the operation. **0** or **void** means error and **1** successful operation.
DS_ERROR_ID	[0-9]{1,5}	On error, here come the error code generated. If no error will be 0 or empty. The error codes are specified in the document: **Appendix I - error codes**, available in the "Documentation" on the control panel.

If the request was an error of some sort (wrong signature, wrong amount, user not found, etc.) all fields are empty, except "DS_ERROR_ID" containing the error code.

If the transaction is successful, the service will return all fields (except DS_MERCHANT_CARDCOUNTRY which is optional) to be stored by trading ahead of a possible refund. In this case the field will be returned DS_ERROR_ID empty.

6.6 CREATE A SUBSCRIPTION IN THE SYSTEM

Function: (**create_subscription**)

The subscription involves a register of a user in the system of **PAYTPV BankStore**. This process is completely independent from the single authorization to a trade customer.

The variables required to register a user with subscription are (in this order):

Element	Content	Description
DS_MERCHANT_MERCHANTCODE	[A-Za-z0-9]{1,8}	Required. Account code
DS_MERCHANT_TERMINAL	[0-9]{1,4}	Required. Terminal ID
DS_MERCHANT_PAN	[0-9]{16,19}	Required. PAN without spaces
DS_MERCHANT_EXPIRYDATE	[0-9]{4}	Required. Expiration date in “mmyy” (2 characters month and 2 characters year)
DS_MERCHANT_CVV2	[0-9]{3,4}	Required. CVC2 code
DS_SUBSCRIPTION_STARTDATE	[YYYY-MM-DD]	Required. Start date of subscription. If empty the start date will be today.
DS_SUBSCRIPTION_ENDDATE	[YYYY-MM-DD]	Required. End date of subscription.
DS_SUBSCRIPTION_ORDER	[A-Za-z0-9]{1,20}	Required. First characters of the trade Order. **IMPORTANT**: You can't include “**[**“ or “**]**”, because will be used to recover information of the user by the commerce.
DS_SUBSCRIPTION_PERIODICITY	[0-9]{3}	Required. Periodicity of the payments. The numbers are **Days**.
DS_SUBSCRIPTION_AMOUNT	[0-9]{1,8}	Required. Amount of the operation in int format (not float) 1,00 EURO = 100, 4,50 EUROS = 450...
DS_SUBSCRIPTION_CURRENCY	[EUR][USD][GBP] [JPY]	Required. Currency of the transaction. See more information in **ANNEX I**
DS_MERCHANT_MERCHANTSIGNATURE	[a-zA-Z0-9]{40}	Required**. See HASH CALCULATION**
DS_ORIGINAL_IP	A.B.C.D	Required. IP Address of the final customer.

The description of the column "Content" refers to the regular expression that must meet the information in the corresponding element (except in DS_ORIGINAL_IP, who means IP Address “192.168.1.254”). For example

"[A-Z]" means from "A" to "Z" and numbers inside brackets means string length. For example "1{2,4}" validate the numbers "11", "111" and "1111".

IMPORTANT: The field DS_SUBSCRIPTION_ORDER should be the first characters of the purchase reference. Followed DS_IDUSER data will be attached by brackets [] + the day of the transaction format: YYYYMMDD. Thus the reference is always the same but will change when performing periodic transactions.

Example:

DS_SUBSCRIPTION_ORDER = Luis_3268314

Purchase of the user DS_IDUSER 32 in the day 23th December of 2030 the system will create the DS_SUBSCRIPTION_ORDER:

DS_SUBSCRIPTION_ORDER = Luis_3268314[23]20301223

The service response to the request is done by returning an array formatted in XML with the different elements are described in the following table:

DS_IDUSER	[0-9]{1,13}	Unique identifier of the user logged into the system. Return empty on error.
DS_TOKEN_USER	[A-Za-z0-9]{1,20}	Token code associated to DS_IDUSER.
DS_SUBSCRIPTION_AMOUNT	[0-9]{1,8}	Amount of the operation in int format (not float) 1,00 EURO = 100, 4,50 EUROS = 450...
DS_SUBSCRIPTION_ORDER	[A-Za-z0-9]{1,20}	Original order of the transaction + **[DS_IDUSER]** + date in format **YYYYMMDD**.
DS_SUBSCRIPTION_CURRENCY	[EUR][USD][GBP][JPY]	Currency of the transaction. For more information see **ANNEX I**
DS_MERCHANT_AUTHCODE	[a-zA-Z0-9]{1,40}	Authcode of the transaction
DS_MERCHANT_CARDCOUNTRY	[0-9]{1,3}	Country of the card in ISO3 code (ex: 724 = Spain). Can be empty.
DS_ERROR_ID	[0-9]{1,5}	On error, here come the error code generated. If no error will be 0 or empty. The error codes are specified in the document: **Appendix I - error codes**, available in the "Documentation" on the control panel.

If the request was an error of some sort (wrong signature, wrong amount, user not found, etc.) all fields are empty, except "DS_ERROR_ID" containing the error code.

If the transaction is successful, the service will return DS_TOKEN_USER and DS_IDUSER fields that must be stored by trade, associating it with their client, and subsequently modify or terminate subscription.

If execution of the first quota system finish unsuccessfully all fields will be empty unless DS_ERROR_ID.

If the execution of the first installment is flawed for several reasons (balance, validity of the card, etc ...),

the subscription will be canceled having to create another new subscription. In this case only return DS_ERROR_ID with specific error code.

6.7 SUBSCRIPTION MODIFICATION

Function: (**edit_subscription**)

If a user renew your subscription or just want to increase the amount of the purchase, we offer a subscription service change. In this case you can not change the currency or banking data of the customer. This process is completely independent from the single purchase customer.

The variables required for modifying a subscription are (in order):

Element	Content	Description
DS_MERCHANT_MERCHANTCODE	[A-Za-z0-9]{1,8}	Required. Account code
DS_MERCHANT_TERMINAL	[0-9]{1,4}	Required. Terminal ID
DS_IDUSER	[0-9]{1,13}	Required. Unique identifier of the registered user in the system.
DS_TOKEN_USER	[A-Za-z0-9]{1,20}	Required. Token code associated to DS_IDUSER.
DS_SUBSCRIPTION_STARTDATE	[YYYY-MM-DD]	Required. Start date of subscription. If empty the start date will be today.
DS_SUBSCRIPTION_ENDDATE	[YYYY-MM-DD]	Required. End date of subscription.
DS_SUBSCRIPTION_PERIODICITY	[0-9]{3}	Required. Periodicity of the payments. The numbers are **Days**.
DS_SUBSCRIPTION_AMOUNT	[0-9]{1,8}	Required. Amount of the operation in int format (not float) 1,00 EURO = 100, 4,50 EUROS = 450...
DS_MERCHANT_MERCHANTSIGNATURE	[a-zA-Z0-9]{40}	Required**. See** **HASH CALCULATION**
DS_EXECUTE	[0-1]{1}	Required. If the registration process involves the collection of the first installment DS_EXECUTE value must be **1**. If you only want the modification of the subscription without the payment of the first installment its value must be **0**.
DS_ORIGINAL_IP	A.B.C.D	Required. IP Address of the final customer.

The description of the column "Content" refers to the regular expression

that must meet the information in the corresponding element (except in DS_ORIGINAL_IP, who means IP Address "192.168.1.254"). For example "[A-Z]" means from "A" to "Z" and numbers inside brackets means string length. For example "1{2,4}" validate the numbers "11", "111" and "1111".

The service response to the request is done by returning an array formatted in XML with the different elements are described in the following table:

DS_IDUSER	[0-9]{1,13}	Unique identifier of the user logged into the system. Return empty on error.
DS_TOKEN_USER	[A-Za-z0-9]{1,20}	Token code associated to DS_IDUSER.
DS_SUBSCRIPTION_AMOUNT	[0-9]{1,8}	Amount of the operation in int format (not float) 1,00 EURO = 100, 4,50 EUROS = 450...
DS_SUBSCRIPTION_ORDER	[A-Za-z0-9]{1,20}	Original order of the transaction + **[DS_IDUSER]** + date in format **YYYYMMDD**.
DS_SUBSCRIPTION_CURRENCY	[EUR][USD][GBP][JPY]	Currency of the transaction. For more information see **ANNEX I**
DS_MERCHANT_AUTHCODE	[a-zA-Z0-9]{1,40}	Authcode of the transaction
DS_MERCHANT_CARDCOUNTRY	[0-9]{1,3}	Country of the card in ISO3 code (ex: 724 = Spain). Can be empty.
DS_ERROR_ID	[0-9]{1,5}	On error, here come the error code generated. If no error will be 0 or empty. The error codes are specified in the document: **Appendix I - error codes**, available in the "Documentation" on the control panel.

If the request was an error of some sort (wrong signature, wrong amount, user not found, etc.) all fields are empty, except "DS_ERROR_ID" containing the error code.

If execution of the first quota is successfully, system will return all fields filled except DS_ERROR_ID who will be empty.

If the execution of the first installment is flawed for several reasons (balance, validity of the card, etc ...), the subscription will be canceled having to create another new subscription. In this case only return DS_ERROR_ID with specific error code.

6.8 SUBSCRIPTION CANCELLATION

Function: (**remove_subscription**)

This function is used to remove a subscription.

The variables required to request removal of a subscription are (in order):

Element	Content	Description

DS_MERCHANT_MERCHANTCODE	[A-Za-z0-9]{1,8}	Required. Account code
DS_MERCHANT_TERMINAL	[0-9]{1,4}	Required. Terminal ID
DS_IDUSER	[0-9]{1,13}	Required. Unique identifier of the registered user in the system.
DS_TOKEN_USER	[A-Za-z0-9]{1,20}	Required. Token code associated to DS_IDUSER.
DS_MERCHANT_MERCHANTSIGNATURE	[a-zA-Z0-9]{40}	Required**. See** **HASH CALCULATION**
DS_ORIGINAL_IP	A.B.C.D	Required. IP Address of the final customer.

The description of the column "Content" refers to the regular expression that must meet the information in the corresponding element (except in DS_ORIGINAL_IP, who means IP Address "192.168.1.254"). For example "[A-Z]" means from "A" to "Z" and numbers inside brackets means string length. For example "1{2,4}" validate the numbers "11", "111" and "1111".

The service response to the request is done by returning an array formatted in XML with the different elements are described in the following table:

DS_RESPONSE	[0-1]{1}	Result of the operation. **0** or **void** means error and **1** successful operation.
DS_ERROR_ID	[0-9]{1,5}	On error, here come the error code generated. If no error will be 0 or empty. The error codes are specified in the document: **Appendix I - error codes**, available in the "Documentation" on the control panel.

If the request was an error of some sort (wrong signature, wrong amount, user not found, etc.) all fields are empty, except "DS_ERROR_ID" containing the error code.

If the transaction is successful, the service will return DS_RESPONSE field whose value is **0** or **empty** on failure and **1** on success in eliminating.

ANNEX I – CURRENCY AND AMOUNT

The amount must be formatted depending on the currency selected by the trade. It depends of the currency if need to be represented as an integer (whole numbers) or will be multiplied by 1 or 100.

The currencies available by the payment platform are:

Currency	Code	Observations	Value of Amount
Euro	EUR	You can always work with Euros	Cents of euro: "123" = 1,23€
US Dollar	USD	Must be enabled multicurrency	Cents of dollar: "123" = $1,23
Pound Sterling	GBP	Must be enabled multicurrency	Penny: "123" = £1,23
Japan Yen	JPY	Must be enabled multicurrency	Yen: "123" = ¥123
Brazilian real	BRL	Multi-currency must be enabled	Real cents: "123" = $1.23
Polish złoty	PLN	Multi-currency must be enabled	Groszy: "123" = zł1.23
Cape Verdean escudo	CVE	Multi-currency must be enabled	Escudo Cents: "123" = $1.23

Charging currencies other than the euro must be specifically enabled by **PAYTPV**. You can contact us via the customer control panel on the menu "*Soporte→Notificación de incidencia*".

ANNEX II – HASH CALCULATION

The hash shall be calculated on the trade server and relate to the main parameters of the call to verify the integrity of data over the Internet and the shopper's browser.

The encryption algorithm used for this purpose is SHA1, which allows us to encrypt a text string. Such one-way algorithms prevents obtaining initial parameter from the result.

The firm will be calculated in different ways depending on the function used to verify the integrity of data delivery platform.

To facilitate the generation of the signature in languages without the tools to calculate SHA1 algorithms have included a number of libraries. You can download the files from *Soporte→Documentación* in the customer Control Panel.

Some of these libraries have been made by persons or entities other than **PAYTPV**. See the contents of each file for more information.

HASH CALCULATION DEPENDING OF THE FUNCTION USED

Function: (**add_user**)

The hash send to the gateway will be calculated as follows (in pseudo-code):

```
SHA1(DS_MERCHANT_MERCHANT_CODE + DS_MERCHANT_PAN + DS_MERCHANT_CVV2 +
DS_MERCHANT_TERMINAL + PASSWORD)
```

Function: (**info_user**)

The hash send to the gateway will be calculated as follows (in pseudo-code):

```
SHA1(DS_MERCHANT_MERCHANT_CODE + DS_IDUSER + DS_TOKEN_USER +
DS_MERCHANT_TERMINAL + PASSWORD)
```

Function: (**remove_user**)

The hash send to the gateway will be calculated as follows (in pseudo-code):

```
SHA1(DS_MERCHANT_MERCHANT_CODE + DS_IDUSER + DS_TOKEN_USER +
DS_MERCHANT_TERMINAL + PASSWORD)
```

Function: (**execute_purchase**)

The hash send to the gateway will be calculated as follows (in pseudo-code):

```
SHA1(DS_MERCHANT_MERCHANT_CODE + DS_IDUSER + DS_TOKEN_USER +
DS_MERCHANT_TERMINAL + DS_MERCHANT_AMOUNT + DS_MERCHANT_ORDER + PASSWORD)
```

Function: (**execute_refund**)

The hash send to the gateway will be calculated as follows (in pseudo-code):

```
SHA1(DS_MERCHANT_MERCHANT_CODE + DS_IDUSER + DS_TOKEN_USER +
```

DS_MERCHANT_TERMINAL + DS_MERCHANT_AUTHCODE + DS_MERCHANT_ORDER + PASSWORD)

Function: (**create_subscription**)

The hash send to the gateway will be calculated as follows (in pseudo-code):

SHA1(DS_MERCHANT_MERCHANT_CODE + DS_MERCHANT_PAN + DS_MERCHANT_CVV2 + DS_MERCHANT_TERMINAL + DS_SUBSCRIPTION_AMOUNT + DS_SUBSCRIPTION_CURRENCY + PASSWORD)

Function: (**edit_subscription**)

The hash send to the gateway will be calculated as follows (in pseudo-code):

SHA1(DS_MERCHANT_MERCHANT_CODE + DS_IDUSER + DS_TOKEN_USER + DS_MERCHANT_TERMINAL + DS_SUBSCRIPTION_AMOUNT + PASSWORD)

Function: (**remove_subscription**)

The hash send to the gateway will be calculated as follows (in pseudo-code):

SHA1(DS_MERCHANT_MERCHANT_CODE + DS_IDUSER + DS_TOKEN_USER + DS_MERCHANT_TERMINAL + PASSWORD)

ANNEX III –OPERATION TYPES

The operation types that will be notified are detailed in the following table:

ID	Operation
1	Authorisation
2	Refund
106	Chargeback