



**COLLECTING SOLUTION**

# **Adding the payment button Paylib**

## **Implementation guide**

Document version 2.4

# Contents

<b>1. HISTORY OF THE DOCUMENT.....</b>	<b>3</b>
<b>2. OBTAINING HELP.....</b>	<b>4</b>
<b>3. PAYLIB: PAYING WITH AN E-WALLET.....</b>	<b>5</b>
3.1. Viewing the PAYLIB payment process.....	5
<b>4. USING THE PAYLIB PAYMENT METHOD.....</b>	<b>6</b>
4.1. Prerequisites.....	6
4.2. Signing in to the Lyra Collect Back Office.....	6
4.3. Declaring a Paylib contract in the Expert Back Office.....	7
4.4. Viewing the progress of a PAYLIB payment on the payment page.....	9
<b>5. ESTABLISHING INTERACTION WITH THE PAYMENT GATEWAY.....</b>	<b>11</b>
<b>6. GENERATING A PAYMENT FORM.....</b>	<b>12</b>
6.1. Creating an immediate payment with Paylib.....	12
6.2. Using additional features.....	14
Transmitting shipping details.....	14
Transmitting order details.....	15
Transmitting buyer details.....	17
Managing the payment methods offered to the buyer.....	18
<b>7. COMPUTING THE SIGNATURE.....</b>	<b>19</b>
<b>8. ANALYZING THE PAYMENT RESULT.....</b>	<b>21</b>
8.1. Processing data specific to the response of a wallet payment.....	21
<b>9. IDENTIFYING OPERATIONS AUTHORIZED ON TRANSACTIONS.....</b>	<b>23</b>
9.1. Change a transaction.....	23
9.2. Cancel a transaction.....	24
9.3. Capturing a transaction.....	24
9.4. Edit the order reference.....	25
9.5. Resending transaction confirmation e-mail to the buyer.....	25
9.6. Resending the transaction confirmation e-mail to the merchant.....	25
9.7. Making a refund.....	26
9.8. Duplicating a transaction.....	26
9.9. Reconciling a transaction manually.....	27

# 1. HISTORY OF THE DOCUMENT

---

Version	Author	Date	Comment
2.4	Lyra Collect	16/07/2019	The creation process of a merchant ID has been updated.
2.3	Lyra Collect	01/10/2018	Initial version

This document and its contents are confidential. It is not legally binding. No part of this document may be reproduced and/or forwarded in whole or in part to a third party without the prior written consent of Lyra Collect. All rights reserved.

## 2. OBTAINING HELP

---

Need some help? Please, check the FAQ on our website

<https://lyra.com/doc/fr/collect/faq/sitemap.html>

For technical inquiries or support, you can reach us from Monday to Friday, between 9 a.m. and 6 p.m.

by phone at:

**0811900475**

Service fee 0.06 € / min  
+ call charge

by e-mail:

[support-ecommerce@lyra-collect.com](mailto:support-ecommerce@lyra-collect.com)

and via your Expert Back Office, menu **Help > Contact support**

## 3. PAYLIB: PAYING WITH AN E-WALLET

**PAYLIB** is an innovative **e-Wallet** service that is available on the payment gateway. It is the result of collaboration of large French banks (BNP Paribas, La Banque Postale, la Société Générale and Arkea). Its goal is to simplify e&m-commerce payments.

**PAYLIB** was designed designed in the best interest of the buyer and the merchant. It allows to reinforce the clients' trust when they make purchases on the Internet and secure their payments.

### Advantages for the buyers:

- They make online payments without having to enter their bank card details.
- The service is available on cell phones, tablets and computers.
- The login and password are personal. The customer journey becomes simpler and more fluid.
- The banking details are stored by their bank.
- The transaction can be traced using their bank statement.

### Advantages for your merchant website:

- Integration is simple. In order to use this service, all you need to do is sign up the bank contract for **PAYLIB**.
- This will be a guarantee of safety. The buyer's bank performs all the necessary security checks before validating a transaction.
- **PAYLIB** has the same level of guarantee as 3D Secure and includes liability shift.

### 3.1. Viewing the PAYLIB payment process

Here is what the process of an online payment with **PAYLIB** looks like:



*Figure 1: The PAYLIB payment process*

This payment system is simple, fast and easy to use.

## 4. USING THE PAYLIB PAYMENT METHOD

---

The **Paylib** service allows to make single or deferred payments with a term of strictly less than 7 days.

Recurring payments, deferred payments with a term of strictly more than 6 days or installment payments are not supported.

Transactions made via the Paylib service are visible via **Management > Transactions > Transactions in progress tab**.

### 4.1. Prerequisites

---

#### For the merchant

Adding the PAYLIB payment option to your merchant website is very simple: it consists in activating the **Payment via Paylib** function.

Please contact your sales contact person to subscribe to the PAYLIB option in your Lyra Collect offer.

#### Characteristics of a PAYLIB contract

The special feature of a PAYLIB contract is the enrollment. The chapter **Associating a CB contract with a PAYLIB contract** provides a step-by-step instruction on integrating it within your payment page.

#### For the buyer

The subscription to the PAYLIB service is done directly via the bank. The buyer can enable it via the website or via their banking mobile app in a few clicks:

- Create a PAYLIB account by entering an e-mail address and a password.
- Select a bank card and associate it with PAYLIB without mentioning its number.
- Create a second password that will be used for validating transactions.

#### **Note**

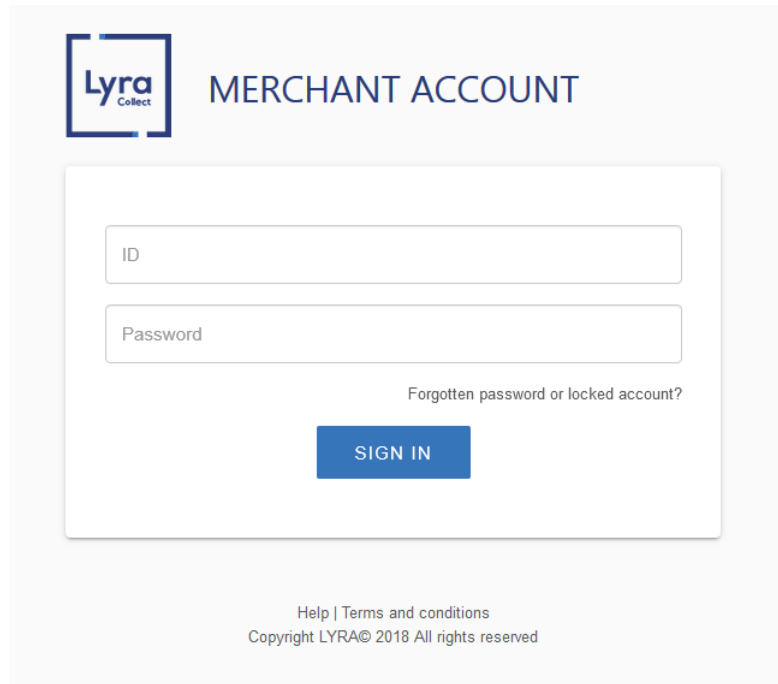
*A buyer with accounts in the 4 partner banks will have to open a PAYLIB account in each of the 4 banks.*

### 4.2. Signing in to the Lyra Collect Back Office

---

Sign in the Back Office:

<https://secure.lyra.com/portal/>



1. Enter your login.

2. Enter your password.

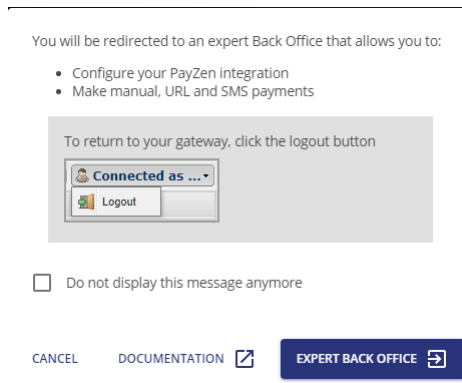
3. Click **Login**.

In case of an entry error of the login and/or password, the error message *“Invalid username or password”* will appear.

You can correct your entry or click on the link **Forgotten password or locked account**.

4. Click **Other actions**.

The following window appears:



5. Click on **Expert Back Office** to access your Expert Back Office

### 4.3. Declaring a Paylib contract in the Expert Back Office

---

Via the menu **Settings > Company > Merchant IDs**:

1. Click **Create a Merchant ID (MID)**.

The **Selection of Merchant ID type** dialog box appears.

2. Select **Paylib**.

The merchant ID creation assistant opens.



3. Select the CB contract to be associated with the Paylib contract.
4. Select the desired mode.
  - **Test**: allows to make test transactions within the Paylib test environment.
  - **Production**: allows to make real transactions.
5. Click **Terminate**.
6. In the **Edition of the Paylib MID** section situated at the bottom of the screen, click the **“Paylib enrollment”** tab, then the **Enroll Merchant ID** button.
7. Click **OK**.

Once the Merchant ID has been created, click the **“Associate with a shop”** button and associate it with a shop of your choosing.



## 4.4. Viewing the progress of a PAYLIB payment on the payment page

---

1. The buyer validates the shopping cart.
2. The merchant website redirects the buyer to the payment gateway. The redirection is done via an HTML POST form using HTTPS. The parameters of the form are described in the chapter **Generating a payment form**.
3. When the parameters and their signature have been verified, the payment method selection page appears.

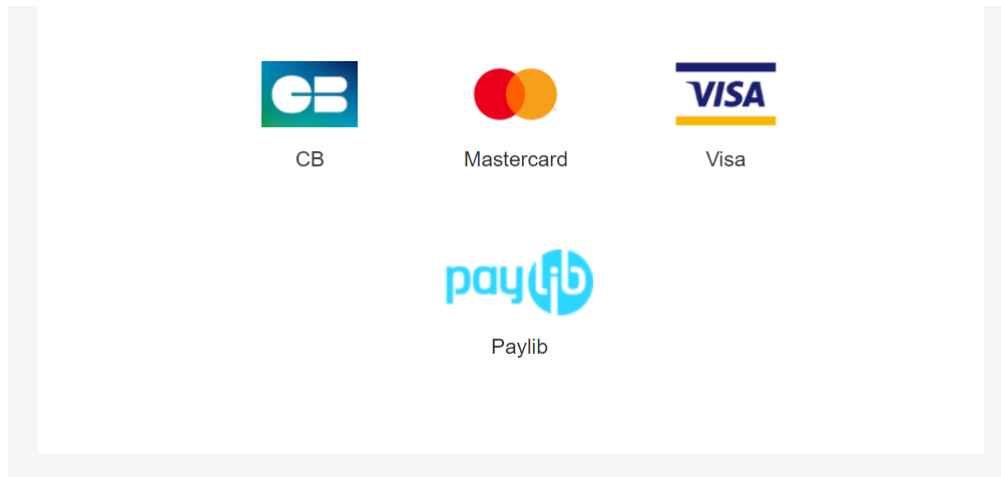


Figure 2: Selecting a payment method

If the payment method has been specified in the form, the buyer moves on directly to step 6.

4. The buyer selects the **Paylib** payment method.  
The gateway redirects him to the Paylib authentication interface.
5. The buyer specifies his or her Paylib identifier and password to sign in.
6. The buyer confirms by clicking **Validate**.
7. He selects his card.
8. An authorization request is sent to the issuer (buyer's bank), in addition to internal fraud verification on the payment gateway.
9. In case of success, a summary page is presented to the buyer resuming the transaction details.  
A link at the bottom of the page allows to return to the shop.


Your payment request has been  
successfully recorded.

REMINDER: this transaction was made in TEST mode.

#### Payment details

---

Transaction identifier :	907278
SHOP :	Lynx 8888
URL address :	http://www.lynx-ems.com
Shop ID :	9138881

 **Mastercard : 10.00 EUR**

---

Wallet :	Paylib
Date / Time :	09-09-2019 / 11:42:16 (GMT+2)
Card number :	XXXXXXXXXXXX1199
Authorization number :	3fa9e7
Merchant ID :	4567654 001
Type :	DEBIT VADS
CB transaction number :	136392
Usage :	Debit
External reference :	75625357



Verified by  
**VISA** 

*Figure 3: Summary page*

In case of failure, a message is displayed. The buyer is informed of the rejection of the payment request. A link at the bottom of the page allows to cancel and return to the shop.

## 5. ESTABLISHING INTERACTION WITH THE PAYMENT GATEWAY

The interaction with the payment gateway is described in the *Hosted Payment Page Implementation guide*, available in the website.

The merchant website and the payment gateway interact by exchanging data.

To create a payment, this data is sent in an HTML form via the buyer's browser.

At the end of the payment, the result is transmitted to the merchant website in two ways:

- automatically by means of notifications called Instant Notification URLs (also known as IPN for Instant Payment Notification).
- by the browser when the buyer clicks the button to return to the merchant website.

To guarantee the security of the exchange, the data is signed with a key only known to the merchant and the payment gateway.

## 6. GENERATING A PAYMENT FORM

To generate a payment request, you must create an HTML form as follows:

```
<form method="POST" action="https://secure.lyra.com/vads-payment/">
  <input type="hidden" name="parameter1" value="value1" />
  <input type="hidden" name="parameter2" value="value2" />
  <input type="hidden" name="parameter3" value="value3" />
  <input type="hidden" name="signature" value="signature"/>
  <input type="submit" name="pay" value="Pay"/>
</form>
```

It contains:

The following technical elements:

- The `<form>` and `</form>` tags that allow to create an HTML form.
- The `method="POST"` attribute that defines the method used for sending data.
- The `action="https://secure.lyra.com/vads-payment/"` attribute that defines where to send the form data.

Form data:

- The shop ID.
- Information about the payment depending on the use case .
- Additional information, depending on your requirements.
- Signature that certifies the integrity of the form.

This data is added to the form by using the `<input>` tag:

```
<input type="hidden" name="parameter1" value="value1" />
```

For setting the `name` and `value` attributes, see the **dictionary** chapter of the *Hosted Payment PageImplementation guide*.

All the data in the form must be encoded in **UTF-8**.

Special characters (accents, punctuation marks, etc.) will then be correctly interpreted by the payment gateway. Otherwise, the signature will not be computed correctly and the form will be rejected.

The **Pay** button that will allow to send data:

```
<input type="submit" name="pay" value="Pay"/>
```

### 6.1. Creating an immediate payment with Paylib

In the immediate payment mode, the buyer pays the total amount for the purchase at once. The payment is captured by the bank on the same day.

1. Use all the fields presented in the table below to create your payment form.

Field name	Description	Value
vads_site_id	Shop ID	E.g.: 12345678
vads_ctx_mode	Operating mode	<b>TEST</b> or <b>PRODUCTION</b>
vads_trans_id	Transaction number	E.g.: 123456

Field name	Description	Value
vads_trans_date	Date and time of the payment form in UTC format	E.g.: 20140129130025
vads_amount	Payment amount (in the smallest currency unit)	E.g.: 3000 for 30,00 EUR
vads_currency	Code of the currency used for the payment	E.g.: 978 for euro (EUR)
vads_action_mode	Card data acquisition mode	<b>INTERACTIVE</b>
vads_page_action	Action to perform	<b>PAYMENT</b>
vads_version	Version of the exchange protocol	<b>V2</b>
vads_payment_config	Payment type	<b>SINGLE</b>
vads_payment_card	Payment method	<b>PAYLIB</b>
vads_capture_delay	Capture delay	<b>0</b>

Table 1: Field list - Immediate payment

2. Populate the **vads\_payment\_config** field with **SINGLE**.
3. Set the **vads\_payment\_card** field to **PAYLIB** if you wish to show only the PAYLIB payment method on your payment page.
4. Set the **vads\_capture\_delay** field to **0**.
5. Set **vads\_currency** with the desired currency code.

The table below shows a non-exhaustive list of available currencies. For more information, refer to the Hosted payment page implementation guide available in our online documentation archive.

<https://lyra.com/doc/fr/>

Currency	ISO 4217 encoding
US Dollar	840
Euro	978
Swiss Franc	756
Pound Sterling	826

Table 2: Extract from the currency table

6. Add optional fields according to your requirements (see chapter **Using additional features**).
7. Compute the value of the **signature** field using all the fields of your form that start with **vads\_** (see chapter **Computing the signature**).

### Example of a form for immediate payment:

```
<form method="POST" action="https://secure.lyra.com/vads-payment/">
<input type="hidden" name="vads_action_mode" value="INTERACTIVE" />
<input type="hidden" name="vads_amount" value="3000" />
<input type="hidden" name="vads_capture_delay" value="0" />
<input type="hidden" name="vads_ctx_mode" value="TEST" />
<input type="hidden" name="vads_currency" value="978" />
<input type="hidden" name="vads_page_action" value="PAYMENT" />
<input type="hidden" name="vads_payment_config" value="SINGLE" />
<input type="hidden" name="vads_site_id" value="91335531" />
<input type="hidden" name="vads_trans_date" value="20190418101407" />
<input type="hidden" name="vads_trans_id" value="239848" />
<input type="hidden" name="vads_version" value="V2" />
<input type="hidden" name="signature" value="0WaYrONo3L0VZqMcvyVf8vT/g8KfZKJ+1jqAs3Ehiw=" />
<input type="submit" name="pay" value="Pay" />
</form>
```

## 6.2. Using additional features

To obtain a custom form adapted to your needs, you can use additional optional features from the list below:

- Manage the payment methods offered to the buyer on the payment page
- Transmit shipping details (address, etc.)
- Transmit order details (reference, shopping cart contents, etc.)
- Transmit buyer details (title, e-mail address, etc.)

### Transmitting shipping details

The merchant can transmit the buyer's shipping details (e-mail address, title, phone number, etc.).

This information can be found in the transaction details in the Expert Back Office (**Shipping** tab).

1. Use the fields required for your use case (see chapter **Generating a payment form** ) to create your payment form.
2. Use optional fields according to your requirements.

These fields will be returned in the response and will include the value transmitted in the form.

Field name	Description	Format	Value
vads_ship_to_city	City	an..128	E.g.: Bordeaux
vads_ship_to_country	Country code in compliance with the ISO 3166 standard.	a2	E.g.: FR
vads_ship_to_district	District	ans..127	E.g.: La Bastide
vads_ship_to_first_name	First name	ans..63	E.g.: Albert
vads_ship_to_last_name	Name	ans..63	E.g.: Durant
vads_ship_to_legal_name	Legal name	an..100	E.g.: D. & Cie
vads_ship_to_phone_num	Phone number	ans..32	E.g.: 0460030288
vads_ship_to_state	State / Region	ans..127	E.g.: Nouvelle Aquitaine
vads_ship_to_status	Allows to specify the type of the shipping address.	enum	<b>PRIVATE</b> : for shipping to a private individual <b>COMPANY</b> : for shipping to a company
vads_ship_to_street_number	Street number	ans..64	E.g.: 2
vads_ship_to_street	Postal address	ans..255	E.g.: Rue Sainte Catherine
vads_ship_to_street2	Second line of the address	ans..255	
vads_ship_to_zip	Zip code	an..64	E.g.: 33000

Table 3: Field list - Shipping details

3. Compute the value of the **signature** field using all the fields of your form starting with **vads\_** (see chapter **Computing the signature**).

### **Transmitting order details**

The merchant can indicate in their payment form if they wish to transfer the order details (order reference, description, shopping cart contents, etc.).

This information can be found in the transaction details in the Expert Back Office (**Shopping cart** tab).

1. Use the fields required for your use case (see chapter **Generating a payment form** ) to create your payment form.
2. Use optional fields according to your requirements. These fields will be returned in the response and will include the value transmitted in the form.

Field name	Description	Format	Value
<b>vads_order_id</b>	Order ID	ans..64	E.g.: 2-XQ001
<b>vads_order_info</b>	Complementary order info	an..255	E.g.: Door phone code 3125
<b>vads_order_info2</b>	Complementary order info	an..255	E.g.: No elevator
<b>vads_order_info3</b>	Complementary order info	an..255	E.g.: Express
<b>vads_nb_products</b>	Number of items in the cart	n..12	E.g.: 2
<b>vads_product_labelN</b>	Item name. N corresponds to the index of the item (0 for the first one, 1 for the second one, etc.).	an..255	E.g.: vads_product_label0 = "tee-shirt" vads_product_label1 = "Biscuit" vads_product_label2 = "sandwich"
<b>vads_product_amountN</b>	Item amount. N corresponds to the index of the item (0 for the first one, 1 for the second one, etc.).	n..12	E.g.: vads_product_amount0 = "1200" vads_product_amount1 = "800" vads_product_amount2 = "950"
<b>vads_product_typeN</b>	Item type. N corresponds to the index of the item (0 for the first one, 1 for the second one, etc.).	enum	E.g.: vads_product_type0 = "CLOTHING_AND_ACCESSORIES" vads_product_type1 = "FOOD_AND_GROCERY" vads_product_type2 = "FOOD_AND_GROCERY"
<b>vads_product_refN</b>	Item reference. N corresponds to the index of the item (0 for the first one, 1 for the second one, etc.).	an..64	E.g.: vads_product_ref0 = "CAA-25-006" vads_product_ref1 = "FAG-B5-112" vads_product_ref2 = "FAG-S9-650"
<b>vads_product_qtyN</b>	Quantity of items. N corresponds to the index of the item (0 for the first one, 1 for the second one, etc.).	n..12	E.g.: vads_product_qty0 = "1" vads_product_qty1 = "2" vads_product_qty2 = "2"

Table 4: Field list - Order details

3. Populate the **vads\_nb\_products** field with the number of items contained in the cart.

#### **Note:**

*This field becomes mandatory for the shopping cart to be taken into account.*

*When it is populated, the **Shopping cart** tab becomes available in the transaction details in the Expert Back Office.*

*However, if the other fields that start with **vads\_product\_** are not populated, the tab will not include any information. For this reason, when populating the **vads\_nb\_products** field, it becomes mandatory to populate the other fields that start with **vads\_product\_**.*

4. Populate the **vads\_product\_amountN** field with the amount for the items in the cart, using the smallest currency unit.

N corresponds to the index of the item (0 for the first one, 1 for the second one, etc.).

**5. Populate `vads_product_typeN` with the value corresponding to the item type.**

N corresponds to the index of the item (0 for the first one, 1 for the second one, etc.).

Value	Description
FOOD_AND_GROCERY	Food and grocery
AUTOMOTIVE	Cars / Moto
ENTERTAINMENT	Entertainment / Culture
HOME_AND_GARDEN	Home and gardening
HOME_APPLIANCE	Household appliances
AUCTION_AND_GROUP_BUYING	Auctions and group purchasing
FLOWERS_AND_GIFTS	Flowers and presents
COMPUTER_AND_SOFTWARE	Computers and software
HEALTH_AND_BEAUTY	Health and beauty
SERVICE_FOR_INDIVIDUAL	Services for individuals
SERVICE_FOR_BUSINESS	Services for companies
SPORTS	Sports
CLOTHING_AND_ACCESSORIES	Clothes and accessories
TRAVEL	Travel
HOME_AUDIO_PHOTO_VIDEO	Sound, image and video
TELEPHONY	Telephony

*Table 5: Values associated with `vads_product_type0`*

**6. Populate `vads_product_labelN` with the name of each item contained in the cart.**

N corresponds to the index of the item (0 for the first one, 1 for the second one, etc.).

**7. Populate `vads_product_qtyN` with the quantity of each item contained in the cart.**

N corresponds to the index of the item (0 for the first one, 1 for the second one, etc.).

**8. Populate `vads_product_refN` with the reference of each item contained in the cart.**

N corresponds to the index of the item (0 for the first one, 1 for the second one, etc.).

**9. Check the value of the `vads_amount` field. It must correspond to the total amount of the order.**

**10. Compute the value of the `signature` field using all the fields of your form starting with `vads_` (see chapter **Computing the signature**).**



### Example of the payment form with cart description:

```
<form method="POST" action="https://secure.lyra.com/vads-payment/">
<input type="hidden" name="vads_action_mode" value="INTERACTIVE" />
<input type="hidden" name="vads_amount" value="11000" />
<input type="hidden" name="vads_capture_delay" value="0" />
<input type="hidden" name="vads_ctx_mode" value="PRODUCTION" />
  <input type="hidden" name="vads_currency" value="978" />
<input type="hidden" name="vads_nb_products" value="2"/>
<input type="hidden" name="vads_product_amount0" value="5000" />
<input type="hidden" name="vads_product_label0" value="produit1" />
<input type="hidden" name="vads_product_qty0" value="2" />
<input type="hidden" name="vads_product_ref0" value="ref1" />
<input type="hidden" name="vads_product_amount1" value="1000" />
<input type="hidden" name="vads_product_label1" value="produit2" />
<input type="hidden" name="vads_product_qty1" value="1" />
<input type="hidden" name="vads_product_ref1" value="ref2" />
<input type="hidden" name="vads_order_id" value="CD100000857" />
<input type="hidden" name="vads_page_action" value="PAYMENT" />
<input type="hidden" name="vads_payment_config" value="SINGLE" />
<input type="hidden" name="vads_site_id" value="12345678" />
<input type="hidden" name="vads_trans_date" value="20140327145218" />
<input type="hidden" name="vads_trans_id" value="571381" />
<input type="hidden" name="vads_version" value="V2" />
<input type="hidden" name="signature" value="xYw1UnU3BACGhf3UEyqbQzpwuvZDEkCAWAE5fgbtfxI=" />
<input type="submit" name="pay" value="Pay"/></form>
```

### Transmitting buyer details

The merchant can specify the buyer's billing details (e-mail address, title, phone number, etc.). This information will be used to create the invoice.

All the data transmitted via the payment form can be viewed in the transaction details in the Expert Back Office (**Buyer** tab).

1. Use the fields required for your use case (see chapter **Generating a payment form** ) to create your payment form.
2. Use optional fields according to your requirements. *These fields will be returned in the response and will include the value transmitted in the form.*

Field name	Description	Format	Value
vads_cust_email	Buyer's e-mail address	ans..150	E.g.: abc@example.com
vads_cust_id	Buyer reference on the merchant website	an..63	E.g.: C2383333540
vads_cust_title	Buyer's title	an..63	E.g.: M.
vads_cust_status	Status	enum	<b>PRIVATE:</b> for a private individual <b>COMPANY:</b> for a company
vads_cust_first_name	First name	ans..63	E.g.: Laurent
vads_cust_last_name	Name	ans..63	E.g.: Durant
vads_cust_legal_name	Buyer's legal name	an..100	E.g.: D. & Cie
vads_cust_cell_phone	Cell phone number	an..32	E.g.: 06 12 34 56 78
vads_cust_address_number	Street number	ans..64	E.g.: 109
vads_cust_address	Postal address	ans..255	E.g.: Rue de l'innovation
vads_cust_address2	Second line of the address	ans..255	E.g.:
vads_cust_district	District	ans..127	E.g.: Downtown
vads_cust_zip	Zip code	an..64	E.g.: 31670
vads_cust_city	City	an..128	E.g.: Labège
vads_cust_state	State / Region	ans..127	E.g.: Occitanie
vads_cust_country	Country code in compliance with the ISO 3166 standard	a2	E.g.: "FR" for France, "PF" for French Polynesia, "NC" for New Caledonia, "US" for the United States

3. Compute the value of the **signature** field using all the fields of your form that start with **vads\_** (see chapter **Computing the signature**).

4. Compute the value of the **signature** field using all the fields of your form that start with **vads\_** (see chapter **Computing the signature**).

Example of payment form with buyer details:

```
<form method="POST" action="https://secure.lyra.com/vads-payment/">
<input type="hidden" name="vads_action_mode" value="INTERACTIVE" />
<input type="hidden" name="vads_amount" value="4000" />
<input type="hidden" name="vads_capture_delay" value="0" />
<input type="hidden" name="vads_ctx_mode" value="PRODUCTION" />
<input type="hidden" name="vads_currency" value="978" />
<input type="hidden" name="vads_cust_country" value="FR" />
<input type="hidden" name="vads_cust_email" value="smith.john@example.com" />
<input type="hidden" name="vads_cust_first_name" value="John" />
<input type="hidden" name="vads_cust_last_name" value="Smith" />
<input type="hidden" name="vads_cust_title" value="Mr" />
<input type="hidden" name="vads_page_action" value="PAYMENT" />
<input type="hidden" name="vads_payment_config" value="SINGLE" />
<input type="hidden" name="vads_site_id" value="12345678" />
<input type="hidden" name="vads_trans_date" value="20140327133115" />
<input type="hidden" name="vads_trans_id" value="522754" />
<input type="hidden" name="vads_version" value="V2" />
<input type="hidden" name="signature" value="7896adcaf7338930db9715afa123531f42"/>
<input type="submit" name="pay" value="Pay"/>
</form>
```

### **Managing the payment methods offered to the buyer**

It is possible to customize the payment methods offered to the buyer.

1. Use all the fields required for your use case (see chapter **Generating a payment form on page 12**) to create your payment form.
2. Set the **vads\_payment\_cards** field.
  - with one single value, if you do not wish to show the page of payment method selection.
  - with a list of values separated by ";" to show the page of payment method selection.

For more information on field values, see chapter **Data dictionary** of the *Hosted Payment Page - Implementation guide*

3. Compute the value of the **signature** field using all the fields of your form that start with **vads\_** (see chapter **Computing the signature on page 19**).

Example of a payment form with payment method selection:

```
<form method="POST" action="https://secure.lyra.com/vads-payment/">
<input type="hidden" name="vads_action_mode" value="INTERACTIVE" />
<input type="hidden" name="vads_amount" value="30000" />
<input type="hidden" name="vads_capture_delay" value="0" />
<input type="hidden" name="vads_ctx_mode" value="PRODUCTION" />
<input type="hidden" name="vads_currency" value="978" />
<input type="hidden" name="vads_page_action" value="PAYMENT" />
<input type="hidden" name="vads_payment_cards" value="VISA;MASTERCARD;PAYLIB" />
<input type="hidden" name="vads_payment_config" value="SINGLE" />
<input type="hidden" name="vads_site_id" value="12345678" />
<input type="hidden" name="vads_trans_date" value="20140526101407" />
<input type="hidden" name="vads_trans_id" value="239848" />
<input type="hidden" name="vads_version" value="V2" />
<input type="hidden" name="signature" value="qppxF6z1+Ri5jtkHNVDCCJulxxpJYehrfP1OLwJ4Ysg="/>
<input type="submit" name="pay" value="Pay"/>
</form>
```

## 7. COMPUTING THE SIGNATURE

---

To be able to compute the signature, you must have:

- all the fields that start with **vads\_**
- the signature algorithm chosen in the shop configuration
- the **key**

The value of the key is available in your Expert Back Office via **Settings > Shop > Keys** tab.

The signature algorithm is defined in your Expert Back Office via **Settings > Shop > Configuration** tab.

**For maximum security, it is recommended to use HMAC-SHA-256 algorithm and an alphanumeric key.**

To compute the signature:

1. Sort the fields that start with **vads\_** alphabetically.
2. Make sure that all the fields are encoded in UTF-8.
3. Concatenate the values of these fields separating them with the "+" character.
4. Concatenate the result with the test or production key separating them with a "+".
5. According to the signature algorithm defined in your shop configuration:
  - a. if your shop is configured to use "SHA-1", apply the **SHA-1** hash function to the chain obtained during the previous step.
  - b. if your shop is configured to use "HMAC-SHA-256", compute and encode in Base64 format the message signature using the **HMAC-SHA-256** algorithm with the following parameters:
    - the SHA-256 hash function,
    - the test or production key (depending on the value of the **vads\_ctx\_mode** field) as a shared key,
    - the result of the previous step as the message to authenticate.
6. Save the result of the previous step in the **signature** field.

## Example of parameters sent to the payment gateway:

```
<form method="POST" action="https://secure.lyra.com/vads-payment/">
<input type="hidden" name="vads_action_mode" value="INTERACTIVE" />
<input type="hidden" name="vads_amount" value="5124" />
<input type="hidden" name="vads_ctx_mode" value="TEST" />
<input type="hidden" name="vads_currency" value="978" />
<input type="hidden" name="vads_page_action" value="PAYMENT" />
<input type="hidden" name="vads_payment_config" value="SINGLE" />
<input type="hidden" name="vads_site_id" value="12345678" />
<input type="hidden" name="vads_trans_date" value="20170129130025" />
<input type="hidden" name="vads_trans_id" value="123456" />
<input type="hidden" name="vads_version" value="V2" />
<input type="hidden" name="signature" value="ycA5Do5tNvsnKdc/eP1bj2xa19z9q3iWPy9/rpesfS0= " />

<input type="submit" name="pay" value="Pay"/>
</form>
```

This sample form is analyzed as follows:

### 1. The fields whose names start with **vads\_** are sorted **alphabetically**:

- vads\_action\_mode
- vads\_amount
- vads\_ctx\_mode
- vads\_currency
- vads\_page\_action
- vads\_payment\_config
- vads\_site\_id
- vads\_trans\_date
- vads\_trans\_id
- vads\_version

### 2. The values of these fields are concatenated using the "+" character:

```
INTERACTIVE+5124+TEST+978+PAYMENT+SINGLE+12345678+20170129130025+123456+V2
```

### 3. The value of the test key is added at the end of the chain and separated with the "+" character. In this example, the test key is **1122334455667788**

```
INTERACTIVE+5124+TEST+978+PAYMENT+SINGLE+12345678+20170129130025+123456+V2+1122334455667788
```

### 4. If you use the SHA-1 algorithm, apply it to the obtained chain.

The result that must be transmitted in the signature field is:  
**59c96b34c74b9375c332b0b6a32e6deec87de2b**

### 5. If your shop is configured to use "HMAC-SHA-256", compute and encode in Base64 format the message signature using the **HMAC-SHA-256** algorithm with the following parameters:

- the SHA-256 hash function,
- the test or production key (depending on the value of the **vads\_ctx\_mode** field) as a shared key,
- the result of the previous step as the message to authenticate.

The result that must be transmitted in the signature field is:

**ycA5Do5tNvsnKdc/eP1bj2xa19z9q3iWPy9/rpesfS0=**

## 8. ANALYZING THE PAYMENT RESULT

The analysis of the payment result is described in the **Hosted payment page Implementation Guide** available in our online documentation archive (<https://lyra.com/doc/fr/>).

This document describes only the processing of data specific to the response of a **Paylib** wallet.

### 8.1. Processing data specific to the response of a wallet payment

The data returned in the response can be specific to the acquirer contract declared for the shop.

The Instant Payment Notification URL will contain the following information:

1. Identify the mode (TEST or PRODUCTION) that was used for creating the transaction by analyzing the value of the **vads\_ctx\_mode** field.

2. Identify the order by retrieving the value of the **vads\_order\_id** field if you have transmitted it to the payment gateway. Sinon, récupérez la valeur du champ **vads\_trans\_id**.

En cas de rejeu de l'IPN, vérifiez que le statut de la commande n'a pas déjà été mis à jour.

3. Identify the payment result by retrieving the value of the **vads\_trans\_status** field.

The value returned in the **vads\_trans\_status** field is specific to the acquirer contract of the shop.

Value	Description
AUTHORISED	The transaction has been accepted and will be automatically captured at the bank on the expected date.
CAPTURED	Captured The transaction has been captured by the bank.
ABANDONED	payment abandoned by the buyer. The transaction has not been created, and <b>therefore cannot be viewed in the Expert Back Office</b> .
CANCELLED	Canceled Cancellation by the buyer, the acquirer or the payment gateway.
REFUSED	Declined
AUTHORISED_TO_VALIDATE	To be validated and authorized
EXPIRED	Expired The expiry date of the authorization request has passed and the merchant has not validated the transaction. The account of the cardholder will, therefore, not be debited.

For more information, see the *Hosted Payment Page Implementation guide* available in our website.

4. Store the type of the card used for the payment by retrieving the value of the **vads\_card\_brand** field.

Value of the vads_card_brand field	Description
CB	Payment made with a CB card.
MASTERCARD	Payment made with a Mastercard card.
VISA	Payment made with a Visa card.
...	...

Table 6: Examples of **vads\_card\_brand** values

5. You can record the used payment wallet type by retrieving the value of the **vads\_wallet** field.

Value of the vads_wallet field	Wallet type
PAYPAL	Paypal live

Value of the vads_wallet field	Wallet type
PAYPAL_SBX	Paypal Sandbox

The **vads\_wallet** is present only when a wallet was used for the payment.

6. Check the amount and the used currency. To do this, retrieve the values of the following fields:

Field name	Description
vads_amount	Payment amount in the smallest currency unit.
vads_currency	Code of the currency used for the payment.

7. Check the value of the **vads\_result** field to obtain the transaction status.

The possible values are presented in the following table. They are always the same regardless of the payment method. In case of a failure, you can also check the value of **vads\_auth\_result** to obtain additional information on the reason of the failure.

Value	Description
00	Action successfully completed.
02	The merchant must contact the cardholder's bank Deprecated.
05	Action rejected.
17	Action canceled by the buyer.
30	Request format error. To associate with the value of the vads_extra_result field.
96	Technical error.

8. Check the value of the **vads\_auth\_result** field to identify the result of the authorization request.

This information is also available in the Expert Back Office. To see it, view the transaction details. The return details (code and label) are displayed amongst the details related to the authorization.

The return code associated with the **vads\_auth\_result** field depends on the used payment method and the network to which it belongs.

9. Record the number of the contract associated with the transaction by retrieving the value of the **vads\_contract\_used** field.

10. Retrieve all the order, buyer and shipping details.

These details will be provided in the response only if they have been transmitted in the payment form.

Their values are identical to the ones submitted in the form.

11. Proceed to order update.

## 9. IDENTIFYING OPERATIONS AUTHORIZED ON TRANSACTIONS

In the Expert Back Office, there are several operations that can be performed with transactions.

In the list of transactions:

1. Select a transaction.
2. Right-click to view the list of authorized operations.

The operations authorized for a transaction in progress are:

- Modify
- Cancel
- Capture manually
- Edit the order reference
- Resending transaction confirmation e-mail to the buyer
- Resending the transaction confirmation e-mail to the merchant

The operations authorized for a captured transaction are:

- Manual reconciliation

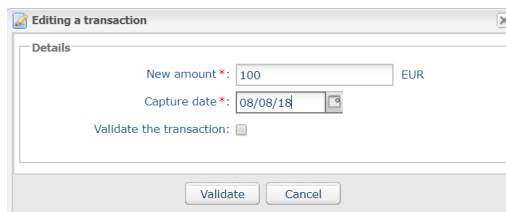
### 9.1. Change a transaction

The **Modify** option is only available for transactions with the following statuses:

- To be validated
- To be validated and authorized
- Waiting for authorization
- Waiting for capture

In order to modify a transaction:

1. Right-click the transaction.
2. Select **Modify**.



3. Enter a new amount.

Unlike in the case of an immediate payment, as long as the transaction has not reached the presentation date, the merchant can modify the amount. The new amount must be lower than the initial amount.

4. Specify the presentation date.

The presentation date can be changed to an earlier or a later date.

- For transactions with **Waiting for capture** or **To be validated** status, the possible presentation range is 3 days. The authorization request valid for 3 days has already been made.
- For transactions with the **To be validated and authorized** or **Waiting for authorization** status, you can enter the presentation date of your choice.

It is also possible to validate a transaction with the **To be validated** or **To be validated and authorized** status by checking **Validate the transaction**.

5. Click **Validate**.

If you wish, you may view the transaction details to see the applied changes (right-click the edited transaction **Display transaction details**).

## 9.2. Cancel a transaction

---

The **Cancel** option is only available for the transactions that have not been captured.

1. Select a transaction with a right-click.
2. Select **Cancel**.
3. Confirm that you wish to definitively cancel the selected transaction.

The transaction status changes to **Canceled**.

### **Note**

*It is possible to **cancel** several transactions at the same time.*

*For this, select all the transactions to be canceled. Press and hold down the **Ctrl** key and **click** for selecting multiple transactions.*

*After the selection, you can click **Cancel** using right-click or via the menu bar and confirm your choice.*

*The transaction statuses will change to **Canceled**.*

## 9.3. Capturing a transaction

---

***This operation is available during test phase. It is not available in production environment.***

The **Capture** option is only available for transactions that have not reached the presentation date.

To manually capture a transaction:

1. Display the tab: **Transactions in progress**
2. Select a transaction with a right-click.
3. Select **Capture manually**.
4. Confirm that you wish to definitively capture the selected transaction.



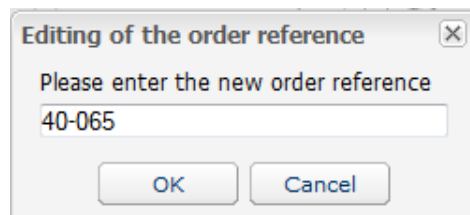
## 9.4. Edit the order reference

---

This operation allows the merchant to change the order reference.

To edit the order reference of a transaction:

1. Right-click the transaction.
2. Select **Edit the order reference**.



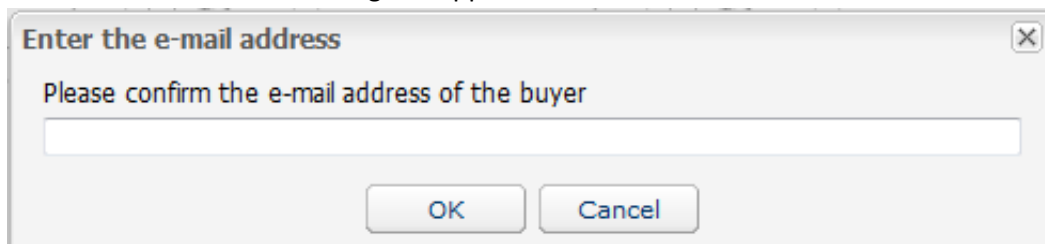
3. Enter the new order reference.
4. Click **OK**.

## 9.5. Resending transaction confirmation e-mail to the buyer

---

To resend the transaction confirmation e-mail to the buyer in case of non-receipt or correction of the email address.

1. Look for the transaction.
2. Right-click the transaction.
3. Right-click the transaction and click **Resending transaction confirmation e-mail to the buyer**.  
The Enter e-mail address dialog box appears.



4. Enter the e-mail address.
5. Click **OK**.

## 9.6. Resending the transaction confirmation e-mail to the merchant

---

To resend the transaction confirmation e-mail to the merchant:

1. Look for the transaction.
2. Right-click the transaction and click **Resending the transaction confirmation e-mail to the merchant**.  
A confirmation message appears.
3. Click **OK**.

## 9.7. Making a refund

---

This operation is available only for transactions present in the **Captured transactions** tab.

The number of the refund depends on the acquirer. The refund amount can only be less than or equal to the displayed amount.

1. Right click on a transaction to refund.
2. Select **Activate refund** in the context menu.

The dialog box **Transaction refund** appears.

3. Enter the amount to be refunded.

4. Click **Perform refund**.

The detail of this operation appears.

## 9.8. Duplicating a transaction

---

This function allows to create a new transaction with the exact same characteristics (e.g. card number) as the transaction that was used for duplication.

A duplicated transaction has the same characteristics as all the other transactions, and it can be duplicated itself.

During duplication of a transaction, a new authorization request is made with the card number that corresponds to the original transaction. This transaction does not have a payment guarantee.

The payment receipt will be sent to the buyer if the e-mail exists on the transaction used for duplication and if the notification rule associated with sending an e-mail to the buyer is active.

Transactions that can be subject to duplication must have one of the following status(es) :

- Captured
- Expired
- Canceled
- Declined

The duplication of refused transactions made with Mastercard cards (Mastercard, Maestro, Mastercard Debit) is forbidden when the reason for refusal is mentioned on the list below:

- 04 - Capture card
- 41 - Lost card
- 14 - Invalid card number
- 43 - Stolen card
- 15 - Invalid issuer
- 54 - Expired card

1. Select the transaction.

2. Click **Duplicate**.

The dialog box **Duplication of the transaction** appears. All of the fields are pre-populated.

Duplication of the transaction 493291

Duplication details

Order reference: pbb-6216

Amount\*: 75.72

Currency\*: USD

Requested capture date\*: 23/11/17

Manual validation:

Duplicate Cancel

You can modify:

- The order reference
- The amount.
- The currency.

If the selected currency is not supported the following message is displayed: **Unsupported currency on this Merchant ID (MID) and/or shop.**

If the selected currency is supported and multi currency possible in your contract, the conversion rate will be applied. The details of the new transaction will be displayed with both currencies: local currency and new currency.

Example

Details of a transaction in progress: 908839

Details Buyer Risk assessment Advanced risks assessment Event log

Transaction identification

Transaction : 908839

Transaction UUID : 7360aa61248c4d5c8801a698be0f723b

UUID Original transaction : ffa80e59f219484bb6f9db02c498c8dc ( [Display transaction](#) )

Order reference : pbb-6216

Shop : [Redacted]

Current amount : EUR 56.01

Amount in currency : USD 75.72

Exchange rate : 1.3518

Type : Debit

- The requested capture date  
It can not be earlier than the transaction modification date.
- The validation mode by (un)checking **Manual validation** if you wish

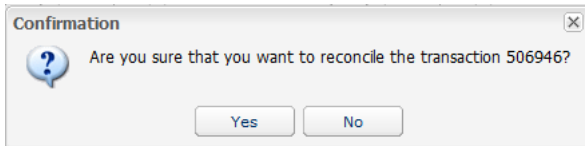
3. Click **Duplicate** to continue or **Cancel** to cancel the duplication.

The transaction can be viewed in the **Transactions in progress** tab.

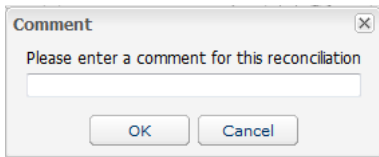
## 9.9. Reconciling a transaction manually

This operation allows you to manually reconcile merchant's payments from an account statement.

1. From the **Captured transactions** tab, look for the relevant transaction.
2. Right-click the transaction.
3. Select **Manual reconciliation**.
4. Click **Yes** to confirm the manual reconciliation of the selected transaction.



The **Comment** dialog box appears.



5. Enter a comment for this reconciliation.

6. Click **OK**.

The transaction status changes to **Reconciled**.