



Adding the SEPA Direct Debit payment button

Implementation Guide

Document version 3.3

Contents

1. HISTORY OF THE DOCUMENT.....	4
2. PRESENTATION.....	5
3. TECHNICAL INFORMATION.....	7
4. PREREQUISITES.....	8
5. PAYMENT PROCESS.....	9
6. SEPA MANDATES.....	10
6.1. The Unique Mandate Reference.....	11
6.2. Presenting a new mandate for signature.....	12
6.3. Functioning of signatures.....	12
6.4. Importing existing direct debit mandates.....	12
6.5. Storing and archiving signed mandates.....	12
6.6. Amendments to the SEPA direct debit mandate.....	13
6.7. Customizing the direct debit motive.....	14
7. HOW SEPA DIRECT DEBIT WORKS.....	15
Additional information on SEPA Direct Debit.....	16
8. CAPTURE FILES.....	20
9. RESTITUTION OF THE DIFFERENT DATES IN THE MERCHANT BACK OFFICE.....	21
10. CREATING AN SDD PAYMENT.....	23
10.1. Viewing the progress of a one-off direct debit.....	23
10.2. Implementation.....	24
11. CREATING AN SDD RECURRING PAYMENT.....	26
11.1. Recurring payment with flexible amounts and/or due dates.....	26
11.1.1. Implementation.....	27
11.1.2. Issuing a direct debit request.....	28
11.2. Recurring payment with fixed amounts and due dates.....	29
11.2.1. Viewing the progress of a recurring direct debit.....	29
11.2.2. Implementation.....	31
11.2.3. Creating a recurring payment via the Merchant Back Office.....	32
11.2.4. Creating a recurring payment via Web services.....	33
11.3. 1-Click SDD payment.....	35
11.3.1. Implementation.....	35
12. IMPORTING EXISTING MANDATES.....	36
12.1. Importing an existing mandate from the Merchant Back Office.....	37
12.2. Importing several existing mandates at a time.....	40
13. UPDATING A RECURRING MANDATE.....	41
13.1. Updating a mandate via the Merchant Back Office.....	41
13.2. Updating a mandate via the payment page.....	43
14. FOLLOWING THE PROCESSING OF YOUR DIRECT DEBITS.....	44
15. MANAGING CHARGEBACKS.....	45
15.1. Setting up the chargeback flow.....	45
15.2. Viewing chargebacks in the Merchant Back Office.....	45

15.3. Table of reasons for chargeback.....	46
15.4. Reissuing a direct debit request.....	49
16. ESTABLISHING INTERACTION WITH THE PAYMENT GATEWAY.....	50
17. SETTING UP NOTIFICATIONS.....	51
17.1. Notifying the debtor.....	52
17.1.1. A pre-notification e-mail.....	52
17.1.2. An e-mail to notify the payment.....	52
17.1.3. Enabling pre-notification.....	53
17.1.4. Customizing the pre-notification e-mail.....	53
17.1.5. Updating the e-mail address.....	53
17.2. Setting up the Instant Payment Notification.....	53
17.3. Setting up notifications in case of abandoned or canceled payments.....	54
17.4. Setting up a notification upon creating a recurring payment.....	54
17.5. Instant Payment Notification URL on an operation coming from the Back Office.....	55
17.6. Setting up the notification for validating SEPA files sent to the bank.....	56
17.7. Configuring e-mails sent to the buyer.....	57
18. GENERATING A PAYMENT FORM.....	58
18.1. Creating a one-off direct debit request.....	60
18.2. Creating a recurring mandate signature request.....	63
18.3. Registering a recurring mandate and making a payment.....	65
18.4. Registering a recurring mandate and creating a recurring payment with fixed due dates and amounts.....	67
18.5. Creating a one-click payment.....	70
18.6. Updating a mandate.....	72
18.7. Updating a mandate and making a payment.....	74
19. COMPUTING THE SIGNATURE.....	76
20. SENDING THE PAYMENT REQUEST.....	78
20.1. Redirecting the buyer to the payment page.....	78
20.2. Processing errors.....	78
21. ANALYZING THE PAYMENT RESULT.....	79
21.1. Analyzing the result of an SDD payment.....	81
21.2. Analyzing the result of the mandate creation.....	83
21.3. Analyzing the result of a flat-rate subscription request.....	84
21.4. Analyzing the result of a mandate update with payment.....	86
21.5. Analyzing the result of an installment payment.....	88
21.6. Analyzing the result of a one-click payment.....	90
22. MANAGING SEPA DIRECT DEBIT TRANSACTIONS FROM THE MERCHANT BACK OFFICE.....	92
22.1. Identifying unavailable operations.....	92
22.2. Identifying authorized operations.....	92
22.3. Viewing SEPA transactions in the Merchant Back Office.....	92
22.4. Viewing mandate details in the Merchant Back Office.....	93
23. OBTAINING HELP.....	94

1. HISTORY OF THE DOCUMENT

Version	Author	Date	Comment
3.3	La Banque Postale	2/23/2023	<p>Addition of the chapter:</p> <ul style="list-style-type: none">• Functioning of signatures <p>Addition of precisions in the following chapters:</p> <ul style="list-style-type: none">• <i>Presentation</i>• <i>Presenting a new mandate for signature</i>• <i>Storing and archiving signed mandates.</i>• <i>Creating a one-off direct debit request.</i>
3.2	La Banque Postale	6/8/2021	<ul style="list-style-type: none">• Addition of the vads_payment_config field for use cases related to the creation of a transaction.• Addition of the possibility to issue a direct debit request via REST Web Services in SILENT mode.• Update of the chapter <i>Technical information</i>.• Addition of a JSON request example in the chapter <i>Creating a subscription via Web Services</i>.• Addition of the fields vads_occurrence_type and vads_acquirer_network in the response analysis for use cases related to the creation of a transaction.
3.1	La Banque Postale	2/2/2021	<ul style="list-style-type: none">• Addition of the chapter <i>Customizing the direct debit motive</i>.• Addition of the chapter <i>Restitution of the different dates in the Merchant Back Office</i>.• Addition of the chapter <i>Capture files</i>.• Update of chapters relative to notification configuration.• Update of the chapter <i>Technical information</i>.
3.0	La Banque Postale	5/26/2020	Document overhaul following the merging of “SEPA Direct Debit (SDD) - One-Off” and “SEPA Direct Debit (SDD) - Recurring”.

This document and its contents are confidential. It is not legally binding. Any reproduction and / or distribution of all or part of this document or its content to a third party is strictly prohibited or subject to prior written authorization from La Banque Postale. All rights reserved.

2. PRESENTATION



SEPA (Single Euro Payments Area) Direct Debit (SDD) is a European project whose aim is to create payments in euro (transfers, direct debits and card payments) across all European countries.

This payment is offered by a creditor (merchant) to a debtor (buyer) in order to automatically pay recurring or one-time invoices.

SEPA Direct Debit is based on a unique form: the "SEPA Direct Debit Mandate".



Supported currencies

- EUR



Supported countries

Member countries of the European Union and Countries of the European Economic Area:

- Germany
- Austria
- Belgium
- Bulgaria
- Cyprus (Greek part)
- Bulgaria
- Denmark
- Spain¹
- Estonia
- Finland²
- France³
- Greece
- Hungary
- Ireland
- Iceland
- Italy
- Latvia
- Liechtenstein
- Lithuania
- Luxembourg
- Malta
- The Netherlands
- Norway
- Poland
- Portugal⁴
- Czech republic
- Romania
- Slovakia
- Slovenia
- Sweden

Countries outside the European Economic Area:

- Andorra
- Monaco
- San Marino
- Switzerland
- United Kingdom⁵
- Vatican City State

Territories outside the European Economic Area:

- Mayotte
- Saint Pierre and Miquelon
- Guernsey
- Jersey
- Isle of man



Additional information

Requires EBICS T functionality.

Caption

1. Spain, including the Canary Islands, Ceuta and Melilla.
2. Finland, including the Aland Islands.

3. France, including Guadeloupe, Martinique, French Guiana, Réunion, the French part of Saint Martin and Saint Barthélemy.
4. Portugal, including the Azores and Madeira.
5. United Kingdom, including Gibraltar.

3. TECHNICAL INFORMATION

Value of the vads_card_brand field	Supported currencies	Supported countries	Authorization validity period	Payment process
SDD	EUR	AD;AT;AX;BE;BG;BL;CH; CY;CZ;DE;DK;EA;EE;ES; FI;FR;GB;GF;GG;GI;GP; GR;HR;HU;IC;IE;IM;IS; IT;JE;LI;LT;LU;LV;MC; MF;MQ;MT;NL;NO;PL; PM;PT;RE;RO;SE;SI;SK; SM;VA;YT	15 days	Deferred capture

Sales channel	
e-commerce	✓
m-commerce	✗
MOTO payment	✓
Interactive Voice Server	✗
Payment order by e-mail/SMS	✓

Operations with transactions	
Cancellation	✓
Refund by wire transfer	✓
Modification	✗
Validation	✓
Duplication	✗
Manual reconciliation	✓

Type of integration	
Redirection	✓
Iframe	✓
JavaScript Client	✗
Data collection form	✓
API Web Services	✓
Mobile SDK	✗
Back Office	✓

Payment type	
Immediate payment	✗
Deferred payment	✓
Payment in installments	✓
Payment by subscription	✓
Payment by file (token or batch)	✓
One-click payment	✓
Payment by wallet	✗

Miscellaneous	
Reporting	✓
Transaction settlement	✗
Chargeback settlement	✓
Extra payment attempts	✗

4. PREREQUISITES

To use this payment method, the merchant must:

- Conclude an **EBICS T** recurring payment contract with his or her bank or ask for a new EBICS user to be created for his or her existing EBICS connection.

The creditor must confirm the direct debit orders via his or her banking interface. Please get in touch with your bank to know more about the validation process.



*The **EBICS TS** protocol is not supported by the payment gateway.*

- Select the **Payment by SEPA Direct Debit** option via the gateway.
- Communicate the **ICS** to the payment gateway.
- Send a request for direct debit file validation process to the bank.

To receive the chargeback reconciliation report, the merchant must:

- Sign up to the outstanding payment flow at their bank.

In order to implement refunds via SEPA transfers:

- Enable the SDD, SCT and RDD flow within your EBICS connection via your bank.

Note

The terminology used for naming the flows may differ depending on the client, the acquirer or the gateway.

Example: RDD flow = chargeback flow.

Please refer to the technical names below in order to request the right flow:

- *SDD = pain.008*
- *SCT = pain.001*
- *RDD = camt.054*

5. PAYMENT PROCESS

Here is the general exchange process for a SEPA Direct Debit:

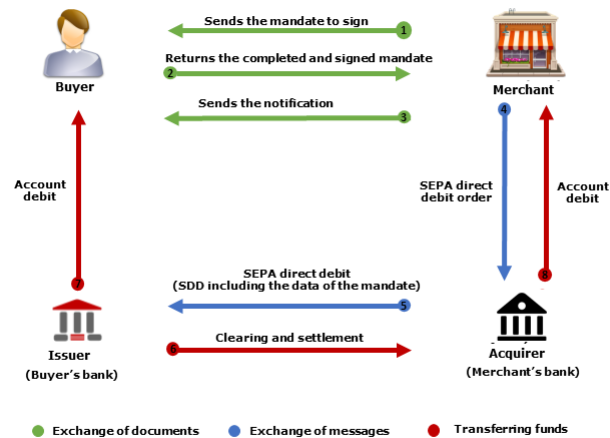


Figure 1: Overview of the data exchange process for SEPA direct debit

Stages 1, 2 and 3 – Document exchange

SEPA Direct Debit is based on a SEPA mandate. Your debtors must sign this mandate in order to prove that they consent to the creditor debiting their account. Each mandate is identified by a **Unique Mandate Reference** that allows to identify the contract to which the SEPA direct debits are attributable.

For each SEPA direct debit, the creditor must send the debtor a pre-notification, 14 calendar days before the direct debit expiry date, to warn him or her about the direct debit.

Stages 4 and 5 – Message exchange

The creditor sends the SEPA Direct Debit order to his or her bank. The bank will contact the debtor's bank in order to proceed to SEPA interbank direct debit.

Stages 6, 7 and 5 – Transfer of funds

Funds from the debtor's bank account are debited and transferred to the creditor's bank account.

6. SEPA MANDATES

There are two types of mandates:

- **One-off**

The signed mandate applies to a single direct debit. It expires after this single direct debit is made.

- **Recurring**

The signed mandate applies to a series of direct debits. It becomes null after a period of 36 months with no payments or after a "FNAL" direct debit.

The SEPA mandate contains the following elements:

- The type of direct debit of the mandate: one-off or recurring.
- The **UMR** (Unique Mandate Reference): unique reference that identifies each signed mandate.
- The coordinates (name) of the debtor.
- The **IBAN** (International Bank Account Number) of the debtor: the account number serves to reliably identify the recipient of a transfer or a direct debit.
The **BIC** (Business Identifier Code) that used to complete the IBAN during identification has become optional.
- The coordinates (name and address) of the creditor.
- The **ICS** (SEPA Creditor Identifier) of the creditor: a unique code that identifies each company with a permission to make direct debits. The ICS is assigned by the Banque de France via a financial institution.
- The date of mandate signature.
- The signature of the debtor authorized by his or her bank to operate the account where the direct debits take place.

Example of a direct debit mandate

SEPA Direct-Debit Mandate

Creditor identifier **FR39 226** Mandate reference **97fc9e29** c0

Name of the debtor

SWIFT BIC

Account number - IBAN **90 158**

Type of payment **Recurrent payment**

By signing this mandate form, you authorise to send instructions to your bank to debit your account and your bank to debit your account in accordance with the instructions from

As part of your rights, you are entitled to a refund from your bank under the terms and conditions of your agreement with your bank. A refund must be claimed within 8 weeks starting from the date on which your account was debited.

Date **21 February 2020** 14:07 21/02/2020

Note: Your rights regarding the above mandate are explained in a statement that you can obtain from your bank.

Tech & Trust by DOCAPOSTE

The SEPA direct debit works in "sequences". The following table lists the possible sequences for direct debit, each code corresponding to a type of mandate.

Mandate type	Possible sequence for the direct debit	Code
One-off	ONE OFF: unique direct debit, not followed by other direct debits within the same mandate.	OOFF
Recurring	FIRST ¹ : 1st direct debit of the series.	FRST
	RECURRING: direct debits following the 1st one of a series.	RCUR
	FINAL: last direct debit of a series.	FNAL

Table 1: Possible sequences depending on the type of direct debit

1. As of 20 November 2016, the FIRST sequence is optional. The first SEPA direct debit of a series can be issued either with the "RCUR" presentation sequence or with the "FRST" presentation sequence.

The payment gateway handles sequence codes when issuing direct debit files to the creditor's bank.

Revocation

The SEPA direct debit mandate can be revoked at any time by the debtor or the creditor.

In case where the revocation is initiated by the debtor, the bank where the account is located does not inform the payment gateway. The potential future transactions that concern this mandate will be considered as chargebacks. They can be identified in the chargeback logs by the return code MD01 (absent mandate).

Exp. date

A mandate becomes null after 36 months without a payment or following a "FNAL" direct debit. In this case, the payment gateway will refuse to create new transactions with this mandate (therefore, there will be no presentation in the bank).

If new direct debits are to be made with this client, a new mandate must be signed.

6.1. The Unique Mandate Reference

By default the UMR is generated by the payment gateway.

However, the creditor can define its value according to the following requirements:

- The UMR must be unique for each direct debit mandate.
Any request for creating one-off or recurring direct debit mandate made with an existing UMR will lead to an error and a technical error message will appear.
- The length must not exceed 35 characters.
- The authorized characters are:
 - a b c d e f g h i j k l m n o p q r s t u v w x y z
 - A B C D E F G H I J K L M N O P Q R S T U V W X Y Z
 - 0 1 2 3 4 5 6 7 8 9 / - ? : () . , ' +
 - Space is not an authorized character.

6.2. Presenting a new mandate for signature

Mandates can only be created via the payment form.

They are signed on the payment page.

Once signed, the mandate is sent to the debtor by e-mail.

Their storage is ensured by the payment gateway (see chapter [Storing and archiving signed mandates](#) on page 12).



The mandate signature is not available via the REST API or via the JavaScript client.

6.3. Functioning of signatures

- **Signature by SMS**

By default, mandates are signed by entering a code received by SMS.

This signature mode is automatically applied if none of the following 2 options is checked: "**Signature by e-mail**" or "**Signature using a checkbox**". In any case, these two signature modes should not be activated at the same time. It's either one or the other.

In case of problems, contact the E-Banking Merchant Support Service.

- **Signature by e-mail**

The mandate is signed via a code received by e-mail.

It is possible to disable the mandate signature by SMS or by e-mail. For this, please contact the E-Banking Merchant Support Service.

- **Signature using a checkbox**

The mandate is signed by ticking the checkbox "**By checking this box, you agree that you have read and understood the details of this mandate and that you append your signature to it**".

6.4. Importing existing direct debit mandates

If the Merchant has already signed direct debit mandates, there are several solutions for declaring these mandates and being able to use them with the payment gateway (see chapter [Importing existing mandates](#) on page 36).

Storing and archiving imported mandates remains the only responsibility of the Merchant.

In the event of a dispute initiated by the debtor, the Merchant must be able to provide the signed mandate.

6.5. Storing and archiving signed mandates

The location for storing and archiving mandates depends on the signature mode:

- **Signature by SMS**

By default, mandates are signed by entering a code received by SMS.

The mandate benefits from a legally binding archiving. The payment gateway is responsible for transmitting the signed mandate for storing and archiving in the electronic safe of the Deposit and Consignment Office.

In case of consignment, this signature serves as guarantee that no changes have been made to the mandate by a third party. The Deposit and Consignment Office provides lines of evidence that cannot be challenged or objected.

- **Signature by e-mail**

The mandate is signed via a code received by e-mail. Legally binding archiving is also applied within this signature mode.

If you wish, you can disable the mandate signature by SMS or by e-mail. For this, please contact the E-Banking Merchant Support Service.

- **Signature using a checkbox**

The mandate is signed by ticking the checkbox **“By checking this box, you agree that you have read and understood the details of this mandate and that you append your signature to it”**.

The payment gateway is responsible for storing and archiving the signed mandate.

The creditor (merchant) also has the possibility to store and archive mandates. However, in this case, the merchant will need to import the UMRs of mandates to the gateway in order to make direct debit requests (by form or by file).

6.6. Amendments to the SEPA direct debit mandate

In case the data that constitutes the SEPA mandate changes, the standard provides for the creation of amendments. This process allows to modify the information without interfering with the recurrence sequence or the need for a new mandate signature.

The following information results in amendments:

- **Creditor:**
 - Modification of the UMR (Unique Mandate Reference)
 - Modification of the ICS (SEPA Creditor Identifier)
 - Modification of the legal name
- **Debtor:**
 - Modification of banking details (BIC, IBAN)

The ICS or the legal name can only be modified upon request sent to the customer service.

Other information can be edited by the Merchant (see chapter [Updating a recurring mandate](#) on page 41).

The changes will appear in your Merchant Back Office:

1. Display the list of tokens in **Management > Recurring payments > Token** tab.
2. Double-click the desired transaction to view the **Token details**.
3. In the **Details** tab, click **See the list of amendments**.

6.7. Customizing the direct debit motive

This information is visible:

- on the debtor's account statement,
- on the creditor's remittance slip.

By default, the debit motive is populated as follows:

<Shop ID> <Company name> <ICS> <UMR>.

The maximum length for the debit motive is 140 characters.

The accepted characters are defined by the ISO 20022 standard.

The personalization of the debit motive consists in modifying the way this information is presented using the following data:

- Shop reference
- Company name
- RUM
- ICS
- Order ID
- A prefix or a string of characters. This piece of data is non-variable.

The merchant can choose the details and their display order.

If the customization chosen by the merchant results in an empty value, then the default valuation will be applied.

To respect the maximum length, the following data can be truncated:

- Order ID
- Company name
- Shop reference

Any request for customization of the direct debit motive must be sent to the technical support.

7. HOW SEPA DIRECT DEBIT WORKS

A pre-notification must be sent at least 14 calendar days before the due date of the direct debit. As of 20 November 2016, upon bilateral agreement, this delay can be reduced but the pre-notification must precede the exchange between the banks of the direct debit orders, i.e.:

- **1 bank business day** before the due date.

Applies both to **one-off SEPA direct debit** (OOFF) and to a **recurring SEPA direct debit** (Recurring RCUR or Final FNAL)

A **bank working day** is a day that is not Saturday, Sunday or public holiday (according to the TARGET 2 calendar published by the CFONB).



If the pre-notification delay is **1 bank business day** before the due date, the capture delay of SEPA direct debit orders by the merchant/creditor must take into account the delay for processing the SEPA direct debit by the creditor's bank (i.e. **1 extra business day**).

The pre-notification delay can only be changed upon request sent to customer service.

All in all, in the Merchant Back Office, the merchant must:

- Pre-notify his or her buyer 2 to 14 days before the due date,
- Submit his or her payment forms 2 business days before the due date of the SEPA direct debit, otherwise the due date will be recalculated by the bank.

Example of SEPA direct debit submitted in December:

With its public holidays, December is the perfect month for illustrating the delays for collecting a SEPA direct debit.

A SEPA direct debit on January 1st (due date) must be presented to the debtor's bank:

- On December 18th at the earliest, that is **14 calendar days** before the due date.
- On December 31st at the latest, that is **1 bank business day** before the due date for a direct debit captured by the merchant at the bank on December 30.
- On December 31st at the latest, that is **1 bank business day** before the due date for a **recurring direct debit** captured by the merchant at the bank on December 30.

January 1st is a holiday. This due date does not correspond to a bank business day. The SEPA direct debit order will be executed on the following bank business day.

Additional information on SEPA Direct Debit

This chapter provides additional information on the variables influencing the payment presentation date when it comes to SEPA Direct Debit.

When creating a SEPA Direct Debit transaction, several elements can have an effect on the date of payment presentation:

- The creation date of the transaction.
- The due date requested by the Merchant in the recurring payment definition.
- The pre-notification delay, specified in the agreement.
- The capture delay, configured in the shop.
- The delay of presentation at the bank, set to 1 bank business day for all SEPA direct debits (as of 20 November 2016).

If the due date requested by the merchant does not respect the delays, the payment gateway will shift this date by as many days as necessary.

If the merchant does not provide a due date, the gateway will calculate the earliest possible date, in accordance with the rules. The calculated date will be returned in the responses to transaction creation (file, web service response, etc.)

The **pre-notification delay** is the minimum number of days that must pass between the moment the pre-notification e-mail is sent to the client and the moment when the funds are transferred. This delay can be redefined but it is incompressible: the delay before the desired fund transfer date will remain the same.

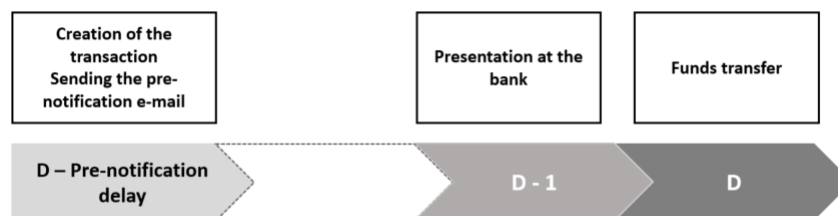
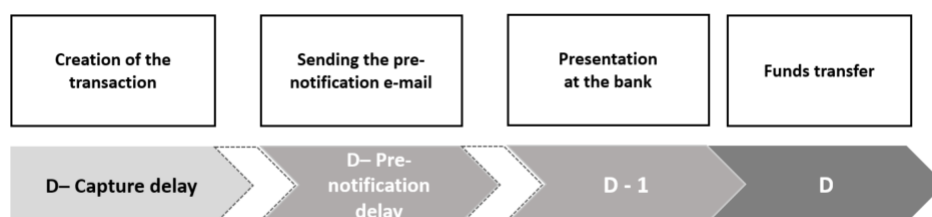


Figure 2: Diagram - pre-notification delay

The **capture delay** is the minimum number of days that must pass between the moment the transaction is created and the moment when the funds are transferred. In the case of SEPA payment, it does not have any influence if it is shorter than the pre-notification delay (see the case of the diagram above). If it is greater than this delay, it will shift the fund transfer date and add the delay between creating the transaction and sending the pre-notification e-mail:



In the following examples, the capture delay is set to zero (default value).

1. Calculation of the minimum fund transfer date

- If the pre-notification delay is shorter than or equal to 2 days:

Minimum fund transfer date = Date of transaction creation + Delay for sending the capture file (the following night) + Presentation delay (set to 1 day)



Either a fund transfer date set to minimum 2 days after the transaction creation (first bank business day starting from the calculated minimum presentation date).

- If the the delay for pre-notification is greater than 2 days

Minimum fund transfer date = Date of transaction creation + Pre-notification delay

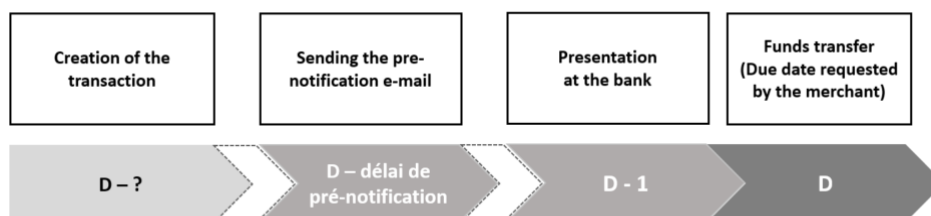
Or for a pre-notification delay of 14 days, a fund transfer date set to minimum 14 days after the transaction creation date (first bank business day starting from the calculated minimum presentation date).

2. Comparison with the due date requested by the merchant

If the merchant has requested a due date when creating the transaction, the requested due date and the minimum fund transfer date calculated during the previous steps are compared.

If the requested due date is greater than the minimum fund transfer date, then:

Fund transfer date = Due date requested by the merchant



Conclusion

In case the merchant wants the payments to be made on a specific date, the requirements are:

- The requested due date must be on a bank business day.
- The delay between the transaction creation date and the desired due date must be greater than the configured pre-notification delay (or the capture delay if it is greater than the pre-notification delay).
- In case the pre-notification delay is shorter than or equal to 2 days, the minimum delay between the date of transaction creation and the desired fund transfer date is two days.

Additional note

In case payment files are used, they are processed twice daily at 7 a.m. and at 1 p.m., every day of the week. If the merchant sends his or her payment file after 1 p.m., the corresponding transactions will be created the following morning at 7 a.m.

Examples:

Due date not specified (or specified on the same day as the date of transaction creation)

Date of transaction creation	Requested date due	Configured pre-notification delay	Date of presentation to the bank	Date of fund transfer
Mon 02/01/2017	Mon 02/01/2017	0	Tue 03/01/2017	Wed 04/01/2017 ⁽¹⁾
Mon 02/01/2017	Mon 02/01/2017	1	Tue 03/01/2017	Wed 04/01/2017 ⁽¹⁾
Mon 02/01/2017	Mon 02/01/2017	2	Tue 03/01/2017	Wed 04/01/2017 ⁽¹⁾
Mon 02/01/2017	Mon 02/01/2017	3	Thu 05/01/2017	Fri 06/01/2017
Mon 02/01/2017	Mon 02/01/2017	5	Fri 06/01/2017	Fri 09/01/2017 ⁽²⁾
Mon 02/01/2017	Mon 02/01/2017	7	Fri 06/01/2017	Mon 09/01/2017
Mon 02/01/2017	Mon 02/01/2017	14	Fri 13/01/2017	Mon 16/01/2017

1. Since the minimum processing delay is two days, the request for transfer of funds cannot be sent to the bank before 04/01/2017.
2. In accordance with the the 5-day pre-notification delay, the transaction fund transfer date will be Saturday 07/01/2017. Since it must be a bank business day, it is therefore shifted to Monday 09/01/2017.

Desired due date set in the future (e.g. On the 6th day of the month)

Date of transaction creation	Requested date due	Configured pre-notification delay	Date of presentation to the bank	Date of fund transfer
Mon 02/01/2017	Fri 06/01/2017	0	Thu 05/01/2017	Fri 06/01/2017 ⁽¹⁾
Mon 02/01/2017	Fri 06/01/2017	1	Thu 05/01/2017	Fri 06/01/2017 ⁽¹⁾
Mon 02/01/2017	Fri 06/01/2017	2	Thu 05/01/2017	Fri 06/01/2017 ⁽¹⁾
Mon 02/01/2017	Fri 06/01/2017	3	Thu 05/01/2017	Fri 06/01/2017 ⁽²⁾
Mon 02/01/2017	Fri 06/01/2017	5	Fri 06/01/2017	Fri 09/01/2017 ⁽³⁾
Mon 02/01/2017	Fri 06/01/2017	7	Fri 06/01/2017	Fri 09/01/2017 ⁽³⁾
Mon 02/01/2017	Fri 06/01/2017	14	Fri 13/01/2017	Mon 16/01/2017 ⁽³⁾

1. Since the delay between the date of transaction creation and the presentation date requested by the merchant is **greater** than the processing delay, the request for the fund transfer is sent to the bank on 06/01/2017 as desired.
2. Since the delay between the date of transaction creation and the due date requested by the merchant is **greater** than the pre-notification delay, the request for the fund transfer is sent to the bank on 06/01/2017 as desired.
3. Since the delay between the date of transaction creation and the due date requested by the merchant is **shorter** than the pre-notification delay, the fund transfer date is determined based on the pre-notification delay.

Desired due date not on a bank business day

Date of transaction creation	Requested date due	Configured pre-notification delay	Date of presentation to the bank	Date of fund transfer
Tue 01/08/2017	Sat 05/08/2017	0	Fri 04/08/2017	Mon 07/08/2017 ⁽¹⁾

Date of transaction creation	Requested date due	Configured pre-notification delay	Date of presentation to the bank	Date of fund transfer
Tue 01/08/2017	Sat 05/08/2017	1	Fri 04/08/2017	Mon 07/08/2017 ⁽¹⁾
Tue 01/08/2017	Sat 05/08/2017	2	Fri 04/08/2017	Mon 07/08/2017 ⁽¹⁾
Tue 01/08/2017	Sat 05/08/2017	3	Fri 04/08/2017	Mon 07/08/2017 ⁽¹⁾
Tue 01/08/2017	Sat 05/08/2017	5	Fri 04/08/2017	Mon 07/08/2017 ⁽¹⁾
Tue 01/08/2017	Sat 05/08/2017	7	Mon 07/08/2017	Tue 08/08/2017 ⁽²⁾
Tue 01/08/2017	Sat 05/08/2017	14	Mon 14/08/2017	Wed 16/08/2017 ⁽³⁾

1. Since the fund transfer must occur on a bank business day, the date is shifted to Monday 07/08/2017. As a result, in accordance with the pre-notification and presentation delays, the fund transfer is made on 07/08/2017 as "requested" by the merchant.
2. Since the delay between the date of transaction creation and the due date requested by the merchant is shorter than the pre-notification delay, the presentation date is determined with respect of the pre-notification delay.
3. In accordance with the the 14-day pre-notification delay, the transaction fund transfer date will be Tuesday 15/08/2017. Since the fund transfer must occur on a bank business day, it is therefore shifted to Wednesday 16/08/2017.

8. CAPTURE FILES

Direct debit requests are grouped by “categories” before the presentation of capture files.

Several lines of captures are visible in the Merchant Back Office (see chapter [Restitution of the different dates in the Merchant Back Office](#) on page 21).

Direct debit requests are grouped:

- By due date: a capture always corresponds to one due date.
- By ICS: if the ICS is updated, then on the same day you can have captures containing the direct debits made with the old ICS and other captures containing the direct debits made with the new ICS.
- By area of issue of the payment method: direct debits outside the EEA zone are separated from direct debits within the EEA zone.
- By mandate type: a capture contains either one-off or recurring direct debits.
- By operation type: refunds (or SCT transfer) are transferred separately.
- The last category groups all the direct debits that were impacted by a mandate amendment that occurs between 2 installments of a recurring direct debit.

Note

For technical reasons, each capture file is limited to 1 000 transactions.

After 1 000 transactions, a new capture file is created.

9. RESTITUTION OF THE DIFFERENT DATES IN THE MERCHANT BACK OFFICE

The capture files are visible via the **Captures** tab of the transaction table.

Each line corresponds to a capture file that can group several direct debit requests presented in chapter [Capture files](#) on page 20.

Among the details, you can see:

- The total capture amount, in the **Debit** column.
- The total amount of the SCT transfers (refunds), in the **Credit** column.
- The presentation date to the bank, in the **Sent on** column.
- The number of direct debits in the capture, in the **Reconciliation** column.

As a reminder, the date of presentation to the bank must occur 1 bank business day before the due date.

Double-click a capture line to display the list of transactions in the capture.

For each transaction line, you can see:

- The direct debit amount in the **Payment amount** column.
- The date when the transaction is created by the payment gateway in the **Payment date** column.
- The date of the funds transfer in the **Expected date of fund transfer** column.
- The UMR in the **Token** column.
- The recurring payment reference in the **Recurring payment** column.

Note

The **Expected date of fund transfer** column is not displayed by default.

To add this column to the transaction table (in progress or captured):

1. Click the **Customize** button at the bottom of the page.
2. Select the **Expected date of fund transfer** column in the **Non-displayed columns** area.
3. Click **Display** or drag-and-drop the column to **Displayed columns**.
4. Click the **Validate** button to save the changes.

As a reminder, the date of the funds transfer is calculated by the gateway so that it is as close as possible to the due date requested by the merchant, when it is not a bank business day.

Double-click a transaction line to display the transaction details.

Among the details, you can see:

- The payment amount
- The type (debit or credit)
- The transaction status
- The creation date of the payment
- The due date requested by the merchant, possibly recalculated by the payment gateway based on business days.

- Retrieve the details of the used payment method:

By clicking on the **History** tab, you will be able to see the details of the operations related to the transaction, such as:

- The date when the pre-notification was sent to the buyer.
- The date when the capture file was sent.

10. CREATING AN SDD PAYMENT

During a SDD payment, a unique direct debit mandate is created.

The Buyer enters their banking details on the payment page and gives their authorization for the direct debit.

The mandate is signed on the payment page.

The mandate is automatically terminated at the end of payment.

The date of funds transfer is calculated on the basis of the pre-notification delay and the capture delay.

It is transmitted in the response via the **vads_presentation_date** field.

10.1. Viewing the progress of a one-off direct debit

The one-off mandate is used for one-off direct debit and does not result in payment in installments.

It is not possible to make other direct debits using the same mandate.

Here are the steps of a direct debit transaction:

1. The debtor (buyer) selects **SEPA Direct Debit**.
2. The next page displays the transaction summary (**Merchant ID**, **Transaction identifier** and **Amount**). The debtor must populate the fields **Last name**, **First name**, **e-mail**, **IBAN** and **phone number** (if it has not been pre-filled).

Identifiant du marchand : [blurred]

Numéro de transaction : 901671

Montant : 65,00 EUR

L'adresse de ce site de paiement préfixée par https indique que vous êtes sur un site sécurisé et que vous pouvez régler votre achat en toute tranquillité.

SEPA Direct Debit

Informations bancaires

Civilité: Mme/Mlle/M. ▼

Nom *

Prénom *

Raison sociale ?

E-mail *

Coordonnées bancaires du compte à prélever

Vous devez être le titulaire de ce compte, ou disposer des autorisations nécessaires pour agir en son nom.

IBAN *

VALIDER

Figure 3: Filling in the debtor's bank account details



Another form of signature is available (signature using a checkbox). Please contact the administrator of your payment gateway for more information.

3. Once the fields are filled in, the debtor clicks **Validate**.

4. Several tests are made to verify the validity of the specified IBAN (such as compatibility with SEPA).

If the data is incorrect, the following message appears: **The specified bank account is not compatible with this payment method.**

If the IBAN test is completed successfully, the SEPA direct debit mandate is displayed with all the required information. The debtor must verify this information.

Identifiant du marchand :


Numéro de transaction :

901671

Montant :

65,00 EUR

L'adresse de ce site de paiement préfixée par https indique que vous êtes sur un site sécurisé et que vous pouvez régler votre achat en toute tranquillité.



Informations bancaires

Mandat de prélèvement SEPA

Créancier : Leclerc 100 rue de l'Industrie, 91070 Lisses, France

Identifiant du créancier : FR 13001 54

Nom du débiteur : Dupont

BIC : CRL

IBAN : FR76 1234 5678 9010 7890 158

Réf. mandat (RUM) : 4c04c 0833d1

Type de paiement : Paiement ponctuel

En signant ce formulaire de mandat, vous autorisez Leclerc à envoyer des instructions à votre banque pour débiter votre compte, et votre banque à débiter votre compte conformément aux instructions de Leclerc.

Vous bénéficiez d'un droit à remboursement par votre banque selon les conditions décrites dans la convention que vous avez passée avec elle. Toute demande de remboursement doit être présentée dans les 8 semaines suivant la date de débit de votre compte.

Code SMS :

Merci de renseigner le code qui vient de vous être envoyé par SMS au numéro +33 6 *** 23 14.

☐ En cochant cette case, vous certifiez avoir lu et compris les informations présentes dans ce mandat et y apposez votre signature.

Figure 4: Validation of the mandate

- In order to be able to sign this mandate, the debtor must:
 - Enter the **Code** received by SMS (or e-mail, depending on the shop option).
 - Check **By checking this box, you agree that you have read and understood the details of this mandate and that you append your signature to it.**
- The debtor clicks **Validate**.
- The overview of the details will appear.

The screenshot displays a payment confirmation interface. At the top left, there is a blue button labeled "RETOURNER À LA BOUTIQUE". The main heading states "Votre demande de paiement a été enregistrée avec succès." followed by a red warning: "RAPPEL : Cette transaction a été effectuée en mode TEST." Below this, the section "Détails du paiement" lists the following information:

- BOUTIQUE :
- Adresse URL : http://www.scellius.com
- Identifiant du marchand : 7100000010
- Numéro de transaction : 901671

A horizontal separator line follows. Below it, the "SEPA" logo is shown next to the text "Prélèvement SEPA : 65,00 EUR". Another horizontal separator line is present. The bottom section contains the following details:

- Date / Heure : 05-08-2019 / 13:48:18 (GMT+2)
- IBAN à prélever : FR76 3000 0000 0000 0000 0000 7890 158
- Titulaire du compte : Dupont

At the bottom center, there is a blue button with a document icon and the label "TICKET".

10.2. Implementation

Step 1: establish interaction with the payment gateway.

Please see chapter *Establishing interaction with the payment gateway* on page 50.

Step 2: set up notifications.

You must configure at least the notification rule at the end of payment and in case of an abandoned payment.

Please see chapter [Setting up notifications](#) on page 51.

Step 3: generate the payment form.

The procedure is described in the chapter [Creating a one-off direct debit request](#) on page 60.

Step 4: redirect the buyer to the payment page.

Please follow the instructions in the chapter [Redirecting the buyer to the payment page](#) on page 78.

Step 5: analyze the notification.

The specific data transmitted during the notification are indicated in the chapter [Analyzing the result of an SDD payment](#) on page 81.

Step 6: follow up on direct debit processing.

Please see chapter [Following the processing of your direct debits](#) on page 44.

11. CREATING AN SDD RECURRING PAYMENT

You can use a recurring mandate to create:

1. **Recurring payments with variable amounts and/or due dates:** after signing the mandate or importing an existing mandate, you issue a direct debit request on each due date.

Please see chapter [Recurring payment with flexible amounts and/or due dates](#) on page 26.

2. **Recurring payments with fixed amounts and due dates:** you use the Recurring payment management service to create a recurring payment. You define the recurrence rule to be applied and the start date of the recurring payment.

Depending on the constraints of different deadlines, the payment gateway calculates the date on which the transaction must be created so that the funds transfer date is as close as possible to the due date defined in the recurrence rule.

The start date must be at least 14 days after the current date.

Once the recurring payment start date has been reached, it is no longer possible to change the installment amounts.

Please see chapter [Recurring payment with fixed amounts and due dates](#) on page 29.

3. **1-Click SDD payments:** no more need to re-enter the IBAN for each order on your merchant website. Instead, you will use the payment form and transmit the UMR to be debited.

Once the Buyer is redirected to the payment page, they simply need to confirm to record their payment.

Please see chapter [1-Click SDD payment](#) on page 35.

11.1. Recurring payment with flexible amounts and/or due dates

This solution takes 2 steps:

1. You proceed to signing the mandate using the payment form.

You can also import your existing mandates (see chapter [Importing existing direct debit mandates](#) on page 12).

2. You issue a direct debit request upon each installment.

With this solution:

- You can change the amount of the direct debit for each installment.
- You are free to not issue a direct debit, for instance, if the client did not use a service over a period of time.

11.1.1.1. Implementation

Step 1: Present a new mandate for signature.

The mandate does not exist yet. You must proceed to the signature of a new mandate by following these steps:

1. Establish interaction with the payment gateway.

Please see chapter [Establishing interaction with the payment gateway](#) on page 50.

2. Configure notification rules.

You must configure at least the notification rule at the end of payment and in case of an abandoned payment.

Please see chapter [Setting up notifications](#) on page 51.

3. Generate the payment form.

The procedure is described in the chapter [Creating a recurring mandate signature request](#) on page 63.

If you want to create a single payment on the day the mandate is signed, see chapter [Registering a recurring mandate and making a payment](#) on page 65.

4. Redirect the Buyer to the payment page.

Please follow the instructions in the chapter [Redirecting the buyer to the payment page](#) on page 78.

5. Analyze the notification.

The specific data transmitted during the notification are indicated in the chapter [Analyzing the result of the mandate creation](#) on page 83.

If the mandate already exists (see chapter [Importing existing mandates](#) on page 36), continue to step 2.

Step 2: Issue a direct debit request.

The available solutions are listed in the chapter [Issuing a direct debit request](#) on page 28.

Step 3: Follow up on direct debit processing.

Please see chapter [Following the processing of your direct debits](#) on page 44.

11.1.2. Issuing a direct debit request

In order to issue a direct debit request, you can:

- **Use the Merchant Back Office**

This solution consists in creating a manual payment using an existing token via the Merchant Back Office.

The Merchant defines the direct debit amount and the desired due date.

Depending on the constraints related to various deadlines, the payment gateway calculates which funds transfer date will be the closest possible to the desired due date.

- **Use the Charge/CreatePayment method of the REST Web Service API**

This solution consists in making a call to the createPayment Web Service **in SILENT mode** by transmitting the UMR to be debited in the **paymentMethodToken** attribute.

To enable the SILENT mode, set the **formAction** attribute to "SILENT". Any other value will cause the request to be rejected.

Depending on the constraints related to different deadlines, the Merchant decides when to issue a direct debit request.

Therefore, the Merchant controls the date of funds transfer.

The amount of direct debits may vary from one due date to another.

If the operation is successful, the Web Service returns a **Payment** object. The fund transfer date is returned in the **transaction.transactionDetails.cardDetails.expectedCaptureDate** attribute.

Example of a request:

```
{
  "amount": "3990",
  "currency": "EUR",
  "formAction": "SILENT",
  "paymentMethodToken": "0d75bd5f1fef4e779a4154e65abb39ca",
  "orderId": "myOrderId-605811",
  "customer": {
    "email": "sample@example.com",
    "reference": "12345678"
  }
}
```

11.2. Recurring payment with fixed amounts and due dates

This solution allows the Merchant to delegate recurrence management to the payment gateway.

By using the payment form, the Merchant can simultaneously proceed to signing the mandate and creating a recurring payment.

The implementation of this solution is specified in chapter [Implementation](#) on page 31.

If a recurring mandate already exists for the debtor, it is possible to define a recurring payment:

- **either via the Merchant Back Office**

The merchant creates a subscription from an existing mandate since his Merchant Back Office.

For more information, see the related documentation [Managing tokens and recurring payments](#).

- **or via Web Services**

For more information, please see chapter [Creating a recurring payment via Web services](#) on page 33.

11.2.1. Viewing the progress of a recurring direct debit

The recurring mandate applies to a series of direct debits.

Here are the steps of payment collection via direct debit:

1. The debtor selects **SEPA Direct Debit**.
2. The transaction information summary page (**Merchant ID** and **Amount**) is displayed. The debtor must populate the fields **Last name**, **First name**, **e-mail**, **IBAN** and **phone number** (if it has not been pre-filled).

Figure 5: Filling in the debtor's bank account details



Another form of signature is available (signature using a checkbox). Please contact the administrator of your payment gateway for more information.

3. Once the fields are filled in, the debtor clicks **Validate**.

4. Several tests are made to verify the validity of the specified IBAN (such as compatibility with SEPA).

If the data is incorrect, the following message appears: **The specified bank account is not compatible with this payment method.**

If the IBAN test is completed successfully, the SEPA direct debit mandate is displayed with all the required information. The debtor must verify this information.

Shop ID : [redacted] 8

Amount per installment : 36.00 EUR

The address of this payment gateway prefixed with https indicates that you are on a secure site and you can safely enter your credit card details.

Token registration

SEPA Direct-Debit Mandate

Creditor : [redacted]
Creditor identifier : FR8 [redacted] 54

Name of the debtor : Francis [redacted] naise
SWIFT BIC : CRLYF [redacted] XX
Account number - IBAN : FR76 3000 [redacted]
Mandate reference : 53bd0665 [redacted] 976ad4
Type of payment : Recurrent payment

By signing this mandate form, you authorise [redacted] to send instructions to your bank to debit your account and your bank to debit your account in accordance with the instructions from [redacted].

As part of your rights, you are entitled to a refund from your bank under the terms and conditions of your agreement with your bank. A refund must be claimed within 8 weeks starting from the date on which your account was debited.

SMS code : [input type="text"] Please enter the code you received by SMS at +33 6 ** ** 23 14.

☐ By checking this box, you agree that you have read and understand the information in this mandate and that you append your signature to it

SUBMIT

Figure 6: Example of mandate validation with SMS validation

5. In order to be able to sign this mandate, the debtor must:
 - Enter the **Code** received by SMS (or e-mail, depending on the shop option).
 - Check **By checking this box, you agree that you have read and understood the details of this mandate and that you append your signature to it.**
6. The debtor clicks **Validate**.
7. The overview of the details will appear.

RETURN TO SHOP

Your registration request with recurring payment has been successfully recorded.

REMINDER: this transaction was made in TEST mode.

Details of the associated payment method

SHOP : [redacted]
URL address : [redacted]
Shop ID : [redacted] 8
Token : 53bd0665 [redacted] 3b976ad4
Amount per installment : 36.00 EUR

SEPA direct debit

Date / Time : 26-06-2020 / 10:51:55 (GMT+2)
Mandate reference (UMR) : 53bd0665 [redacted] 3b976ad4
IBAN to debit : FR76 3000 [redacted]
Account holder : Francis [redacted] naise

RECEIPT

11.2.2. Implementation

Step 1: Present a new mandate for signature.

The mandate does not exist yet. You must proceed to the signature of a new mandate by following these steps:

1. Establish interaction with the payment gateway.

Please see chapter [Establishing interaction with the payment gateway](#) on page 50.

2. Configure notification rules.

You must configure at least the notification rule at the end of payment and in case of an abandoned payment.

Please see chapter [Setting up notifications](#) on page 51.

3. Generate the payment form.

The procedure is described in the chapter [Registering a recurring mandate and creating a recurring payment with fixed due dates and amounts](#) on page 67.

4. Redirect the Buyer to the payment page.

Please follow the instructions in the chapter [Redirecting the buyer to the payment page](#) on page 78.

5. Analyze the notification.

The specific data transmitted during the notification are indicated in the chapter [Analyzing the result of a flat-rate subscription request](#) on page 84.

Note:

The subscription can be defined via the payment form by setting the vads_page_action field to SUBSCRIBE. However, since interaction with the Buyer is limited, this use case is of little interest and is not described in this document. Instead, use the methods presented below.

Step 2: Create the direct debit order.

The payment gateway automatically creates transactions on the date that is most suitable for respecting the due date defined by the recurrence rule.

To be notified when the transaction is created (and receive the funds transfer date), you must:

1. Configure the appropriate notification rule

Please see chapter [Setting up a notification upon creating a recurring payment](#) on page 54.

2. Analyze the notification

The specific data transmitted during the notification are indicated in the chapter [Analyzing the result of an installment payment](#) on page 88.

Step 3: Follow up on direct debit processing.

Please see chapter [Following the processing of your direct debits](#) on page 44.

11.2.3. Creating a recurring payment via the Merchant Back Office

If the mandate already exists (e.g. you have imported an existing mandate), you can create recurring payments via the Merchant Back Office.

In the Back Office:

- A direct debit mandate is represented by a **token**.
- Recurring payments can only be created with recurring mandates that have not been canceled.
- The **Type** column allows you to identify one-off and recurring mandates.

1. Via **Management > Recurring payments** menu, search for the mandate to update.

You can search for a mandate using the unique mandate reference, the e-mail address or the name of the debtor.

You can use the search term **Payment method** by selecting the value **SEPA Direct Debit**.

2. Right-click the mandate.

3. Click **Create a recurring payment**.

The screenshot shows a dialog box titled "Creation of a recurring payment for aac6b16ed49c465b99dc4623cfca1053 (TEST mode)". It is at "Step 1 of 4: Order details". The dialog has a "Shop" section with a dropdown menu. Below it is the "Order details" section with fields for "Order reference:", "Order details:", and "Validation *:" (set to "Automatic"). At the bottom are "Next >" and "Cancel" buttons.

4. Select the shop.

5. Select the **Validation mode** (Automatic or Manual).

6. Click **Next**.

The screenshot shows the same dialog box at "Step 2 of 4: Effective date and amounts". The "Effective date" section has a date field set to "04/04/19" and a note: "The recurring payment will be enabled on this date (not necessarily the date of the first installment)". The "Amounts" section has fields for "Fixed amount of the recurring payment *:" (highlighted in red), "Currency *:" (dropdown), "Initial amount (if different from the fixed amount):", and "Number of installments of the initial amount:". At the bottom are "< Previous", "Next >", and "Cancel" buttons.

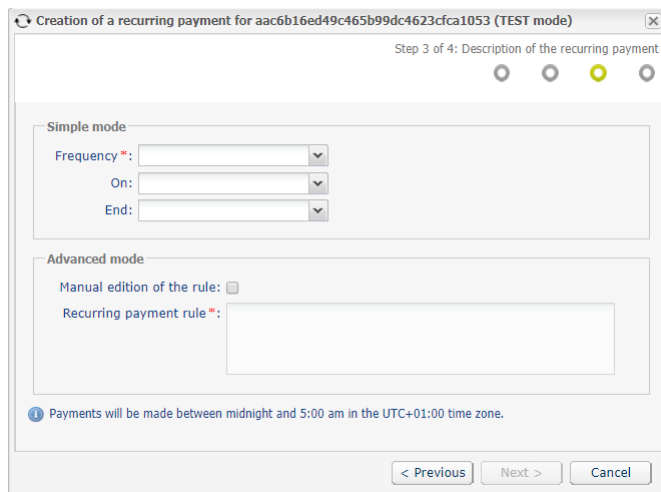
7. Specify the effective date (or the start date of the recurring payment).

The start date must be at least 14 days after the current date.

8. Enter the **fixed amount** and the **currency** of the recurring payment.

It is possible to define additional optional parameters (**Initial amount** if it is different from the **fixed amount** and the **number of installments with the initial amount**).

9. Click **Next**.



10. Specify the recurrence rule in **Simple mode** or in **Advanced mode**.

Check the box **Manual edition of the rule** if you wish to enable the advanced mode and specify the desired recurrence rule.

The rule must respect the iCalendar (Internet Calendar) specification, described in the RFC5545 (see <http://tools.ietf.org/html/rfc5545>).

Example of the rule in advanced mode:

RRULE:FREQ=YEARLY;BYMONTHDAY=-1;BYMONTH=1,4,7,10;UNTIL=20221231

11. Click **Next** to display the summary of the recurring payment.

12. Click **Create** to complete the process.

11.2.4. Creating a recurring payment via Web services

Use the **Charge/CreateSubscription** method of the REST web Service / API

This solution consists in making a call to the **Charge/CreateSubscription** Web Service by transmitting the UMR to be debited in the **paymentMethodToken** attribute.

The Merchant defines the recurrence rule via the **rrule** attribute and the recurring payment start date via the attribute **effectDate**.

The start date must be at least 14 days after the current date.

If the operation is successful, the Web Service returns the reference of the subscription created in the **subscriptionId** attribute.

Example of a request:

```
{
  "amount": "3990",
  "currency": "EUR",
  "effectDate": "2021-06-14T00:00:00+01:00",
  "paymentMethodToken": "0d75bd5f1fef4e779a4154e65abb39ca",
  "rrule": "RRULE:FREQ=MONTHLY;BYMONTHDAY=1;INTERVAL=1",
  "orderId": "myOrderId-803362"
}
```

Example of a response:

```
"answer": {  
  "subscriptionId": "20210531gh5h5R",  
  "_type": "V4/SubscriptionCreated"  
}
```

11.3. 1-Click SDD payment

1-Click payment (or payment by token) requires a signed recurring mandate.

With each payment, the Merchant transmits the UMR to be debited in his or her payment form.

The payment gateway retrieves the mandate information and displays it to the Buyer.

The Buyer confirms the bank details and receives a direct debit notification.

The date of funds transfer is calculated on the basis of the pre-notification delay and the capture delay.

It is transmitted in the response via the **vads_presentation_date** field.

11.3.1. Implementation

Step 1: check notification rule configuration.

Make sure you configure at least the notification rule at the end of payment and the rule in case of abandoned payment.

Please see chapter [Setting up notifications](#) on page 51.

Step 2: generate the payment form.

The procedure is described in the chapter [Creating a one-click payment](#) on page 70.

Step 3: redirect the buyer to the payment page.

Please follow the instructions in the chapter [Redirecting the buyer to the payment page](#) on page 78.

Step 4: analyze the notification.

The specific data transmitted during the notification are indicated in the chapter [Analyzing the result of a one-click payment](#) on page 90.

Step 5: follow up on direct debit processing.

Please see chapter [Following the processing of your direct debits](#) on page 44.

12. IMPORTING EXISTING MANDATES

If you had signed paper direct debit mandates or if you change your technical service providers for your online payments, you may need to reuse these mandates with the payment gateway.

You can import it:

- Manually from the Merchant Back Office.

Please see chapter [*Importing an existing mandate from the Merchant Back Office*](#) on page 37.

- By exchanging files with the payment gateway.

Please see chapter [*Importing several existing mandates at a time*](#) on page 40.

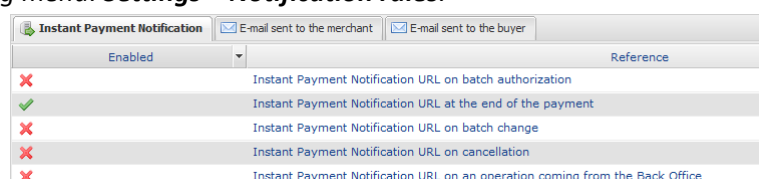
12.1. Importing an existing mandate from the Merchant Back Office

Prerequisites

The notification rules must be configured.

The configuration of notification rules allows to receive notifications once a mandate has been imported. If you have several shops, only one of the shops will receive the notifications, but the created mandate can be used for payments made in all of the shops of the same company (legal name).

1. Go to the following menu: **Settings > Notification rules.**



Instant Payment Notification	
<input checked="" type="checkbox"/> E-mail sent to the merchant <input checked="" type="checkbox"/> E-mail sent to the buyer	
Enabled	Reference
<input checked="" type="checkbox"/>	Instant Payment Notification URL on batch authorization
<input checked="" type="checkbox"/>	Instant Payment Notification URL at the end of the payment
<input checked="" type="checkbox"/>	Instant Payment Notification URL on batch change
<input checked="" type="checkbox"/>	Instant Payment Notification URL on cancellation
<input checked="" type="checkbox"/>	Instant Payment Notification URL on an operation coming from the Back Office

2. Make sure that the **“Instant Payment Notification URL on an operation coming from the Back Office”** rule is present and enabled.
3. If the rule is not enabled, make a right-click and select **Enable the rule.**
4. Double-click the rule and make sure that the URL in TEST mode and the URL in PRODUCTION mode are specified.

For more information on configuring notification rules, see chapter [Instant Payment Notification URL on an operation coming from the Back Office](#) on page 55.

Once the configuration of the notification rule is complete, you can proceed to importing the mandate.

In the Back Office:

- A direct debit mandate is represented by a **token**.
- It is only possible to import recurring mandates.

1. Display the **Management > Recurring payments** menu.

The **Token** tab appears by default.

2. Click **Create**.

3. Select the **Shop to notify** from the list if you have several shops.

The token creation window appears.



*If the **Notifications** item appears on the display of the token creation window, make sure that the steps listed in **Prerequisites** have been completed correctly.*

4. Enter the **Buyer e-mail** address.

5. An Unique Mandate Reference is generated by default in the **Token ID** field. You can click on the button **Generate a new ID** if you wish.

You also can enter your own UMR. You must, however, make sure it is unique.

6. Select the **SEPA Direct Debit** payment method from the list.

The presented payment methods depend on the MIDs associated with your shop.

7. Enter the buyer's IBAN

8. Click **Next**

The buyer detail entry page appears.

The **Token** section reminds you of the specified e-mail and the unique mandate reference.

Creation of a token with the shop

Step 2 of 2: Additional information

Token

Buyer e-mail: [redacted]

Token ID: 991e5b699e6d4c258d19f919499742bf

Buyer details

Buyer reference: [text box]

Legal name: [text box]

Title: [text box]

First name: [text box]

Last name: [text box]

Buyer details: [text box]

Language: [dropdown menu]

Contact details

Country: [dropdown menu]

State: [text box]

City: [text box]

ZIP code: [text box]

Address: [text box]

Additional address details: [text box]

Phone: [text box]

Cell phone: [text box]

< Previous Create Cancel

9. Fill in the information about the buyer.

These details are useful for buyer identification.

Fields marked with an asterisk (*) are required.

The buyer's first and last names are mandatory when creating a SEPA mandate.

10. Click **Create** to complete the process.

If all the payment method verification processes have been successfully completed, the token detail window appears.

Details of the token: 750a4438a8d040f6af140c51b47c2002

Details **Buyer details**

Main

Creation date : 18/06/2019 18:05:22

Termination date :

Token ID : 750a4438a8d040f6af140c51b47c2002

Bank account

BIC : COBADEFFXXX

IBAN : [redacted] 532013000

Type : Recurring

Amendments : [See the list of amendments](#)

Creditor

Legal name : [redacted]

SCI : FR83ZZZ459654

Address : [redacted] Rue [redacted], France

Close

If the “**Confirmation e-mail of recurring payment sent to the merchant**” notification rule is configured and enabled, you will also receive:

- the confirmation of buyer’s banking detail registration,
- the buyer’s token that they can later use for another financial operation.

These details can be sent to the buyer (see chapter [Configuring e-mails sent to the buyer](#) on page 57).

The processing of a mandate import request results in the creation of a **VERIFICATION** transaction type, visible in the Merchant Back Office, that has the following characteristics:

- its amount is 0.00EUR,
- its status is either "Accepted" (vads_trans_status=ACCEPTED) or "Refused" (vads_trans_status=REFUSED),
- it is never captured and remains in the "Transactions in progress" tab.



The mandate (token) will not be created if the IBAN verification fails.

12.2. Importing several existing mandates at a time

This solution allows you to import several mandates.

Here are the main stages of this process:

- The Merchant creates a file containing the data necessary for creating mandates (Unique Mandate Reference, first and last name, postal address, BIC, IBAN, etc.).
- The file is encrypted with a protocol to be determined by the various parties involved.
- The file is transferred to our SFTP server (login identifiers will be sent upon request).
- Keys or passwords are exchanged to decrypt the file.
- The file is imported into the database.

For more information, please contact the [technical support](#).

13. UPDATING A RECURRING MANDATE

You have the possibility to update the following data:

- Bank details: BIC and IBAN,
- The buyer's contact details: e-mail, postal address, telephone numbers.

Updating a mandate can be done:

- Manually from the Merchant Back Office.

The procedure to be followed is described in the chapter [Updating a mandate via the Merchant Back Office](#) on page 41.

- By the Buyer from the payment page.

The implementation of this solution is presented in chapter [Updating a mandate via the payment page](#) on page 43.

13.1. Updating a mandate via the Merchant Back Office

There are two different operations for updating a mandate:

- **Update buyer details**

This operation allows to update information about the buyer. It also gives the possibility to update the unique mandate reference.

- **Replace the payment method**

This operation makes it possible to modify the BIC and the IBAN of the debtor. It also gives the possibility to update the unique mandate reference.

In the Back Office:

- A direct debit mandate is represented by a token.
- A terminated mandate (or token) cannot be modified.
- The **Type** column allows you to identify one-off and recurring mandates.
- A one-off mandate is automatically terminated at the end of payment.

In order to **update buyer details**

1. Via **Management > Recurring payments** menu, search for the mandate to update.

You can search for a mandate using the unique mandate reference, the e-mail address or the name of the debtor.

You can use the search term **Payment method** by selecting the value **SEPA Direct Debit**.

2. Right-click the mandate.
3. Click **Modify**.

4. Click **Update buyer details**.

The mandate modification page appears. Only editable fields are available for input.

5. Edit the e-mail address, if needed.

6. Generate a new token (UMR) if you wish to replace the old one.

7. Complete or edit the information about the buyer.

Only editable fields can be updated.

The buyer's first and last names that are specified during the creation of a SEPA mandate cannot be changed.

8. Click the **Validat** button to save the changes.

To update the debtor's IBAN

1. Via **Management > Recurring payments** menu, search for the mandate to update.

You can search for a mandate using the unique mandate reference, the e-mail address or the name of the debtor.

You can use the search term **Payment method** by selecting the value **SEPA Direct Debit**.

2. Right-click the mandate.

3. Click **Replace the payment method**

The token update page appears.

4. Generate a new token (UMR) if you wish to replace the old one.

5. Complete the new IBAN.

6. Click the **Validat** button to save the changes.

The processing of a mandate update request results in the creation of a VERIFICATION transaction type, visible in the Merchant Back Office, that has the following characteristics:

- its amount is 0.00EUR,
- its status is either "Accepted" ((vads_trans_status=ACCEPTED)) or "Refused" ((vads_trans_status=REFUSED)),
- it is never captured and remains in the "Transactions in progress" tab.



The mandate (token) will not be updated if IBAN checks fail.

In any case, the merchant will be notified about this update if the **Instant Payment Notification URL on an operation coming from the Back Office** rule is enabled.

13.2. Updating a mandate via the payment page

Step 1: check notification rule configuration.

Make sure you configure at least the notification rule at the end of payment and the rule in case of abandoned payment.

Please see chapter [Setting up notifications](#) on page 51.

Step 2: generate the payment form.

The procedure is described in the chapter [Updating a mandate](#) on page 72.

If you want to make an SDD payment at the same time, see chapter [Updating a mandate and making a payment](#) on page 74.

Step 3: redirect the buyer to the payment page.

Please follow the instructions in the chapter [Redirecting the buyer to the payment page](#) on page 78.

Step 4: analyze the notification.

The specific data transmitted during the notification are indicated in the chapter [Analyzing the result of a mandate update with payment](#) on page 86.

14. FOLLOWING THE PROCESSING OF YOUR DIRECT DEBITS

Step 1: Validate the direct debit file.

Each time a direct debit order is transmitted to the creditor's bank, you must validate it via your banking interface.

Without this action, the direct debit will not be made.

Files are normally valid for 15 days. The longer you wait before validating them, the later the funds will be transferred.

The payment gateway can notify you each time a direct debit file is sent.

See chapter [Setting up the notification for validating SEPA files sent to the bank](#) on page 56 to configure notification sending.

We recommend to enter a generic e-mail address rather than a personal address to ensure that in the event of the recipient's absence, the files will still be validated.

What should you do if you have not validated your direct debit orders in time?

The payment gateway can return non-validated files.

You must first notify your bank of this omission and ask if we can return the same capture file.

Then, contact technical support to ask for the capture file to be resent.

Step 2: Enable the transmission of Chargeback reconciliation report.

The chargeback reconciliation log lists the rejected SEPA direct debit capture operations.

Example: debtor account format error, closed account, insufficient funds, unknown client.

Check that the Report frequency is correctly configured:

1. In your Merchant Back Office, open the **Settings > Shop > Reports** menu and select the **Chargeback reconciliation report** tab.
2. In the **General Settings** section, select the frequency from the **Report frequency** drop-down list:

Make sure you have entered a generic e-mail address to ensure that in the absence of the recipient, the chargebacks will still be processed (see chapter [Managing chargebacks](#) on page 45).

Step 3: Reconcile transactions.

When sending the direct debit file, the transaction appears in the Captured transactions tab of the Merchant Back Office. Its reconciliation status is **To be analyzed**.

After 3 interbank business days without a chargeback notification, the payment can be considered as debited.

You can then manually reconcile it by right-clicking on the transaction.

15. MANAGING CHARGEBACKS

If a problem occurs following a direct debit, your bank will return a code stating the reason for rejection.

The payment gateway regularly retrieves information and presents it to you in the Merchant Back Office and in the chargeback logs (the reason appears in the **UNPAID_CODE** field). For more information on this field, see the **Description of reporting** available in our online document archive.

15.1. Setting up the chargeback flow

On the bank side:

Enable the **SEPA Direct Debit chargeback flow** via your bank.

On the Scellius side:

Please ensure that your Scellius offer includes the Visual reconciliation of chargebacks.

It will be set up automatically. Once the **EBICS** connection and the **SEPA Direct Debit chargeback flow** are enabled, your payment gateway will provide you with all the information related to chargebacks.

15.2. Viewing chargebacks in the Merchant Back Office

1. Select **Management > Transactions** menu.
2. Select the **Captured transactions** tab.
Chargebacks are visible via the **Captured transactions** tab.
3. Use the search tool to locate the **Disputes** criteria.
The **Disputes** criteria is situated in the **Status** group.
4. Select **Yes** in order to filter the results.
5. Click **Search**.
6. Double-click a SEPA transaction to see its details.
The content of the **Details** tab is displayed by default.
7. Go to **Transaction lifecycle**, see the line with **Reasons for chargeback**.
A return code is associated with the chargeback.
See the table below to analyze the reason of this chargeback and identify the steps to be made.

15.3. Table of reasons for chargeback

ISO code	Reason - Description	Required action
AC01	Account Identifier incorrect <ul style="list-style-type: none"> The IBAN or BIC code provided by the buyer is incorrect. The account is not in euros. 	Contact the buyer to confirm the correct IBAN. Via the Merchant Back Office, update the token in order to update the BIC and/or the IBAN. Re-submit the payment (see next chapter).
AC04	Account closed The account has been closed. The buyer may have used an old account number or closed the account since the mandate was implemented.	Contact the buyer to obtain the details of the new account. Via the Merchant Back Office, update the token in order to update the BIC and/or the IBAN. Re-submit the payment (see next chapter).
AC06	Account blocked The account has been blocked for direct debits: <ul style="list-style-type: none"> either by the bank or by the buyer 	Contact the buyer to request a replacing payment account. Via the Merchant Back Office, update the token in order to update the BIC and/or the IBAN. Re-submit the payment (see next chapter).
AG01	Transaction forbidden on this type of account The SEPA direct debit cannot be implemented for this type of account. E.g.: savings account.	Contact the buyer for information about the account to be used. Via the Merchant Back Office, update the token in order to update the BIC and/or the IBAN. Re-submit the payment (see next chapter).
AG02	Invalid bank operation code Technical error	Contact the payment gateway.
AM04	Insufficient funds The debtor's bank could not cover the direct debit due to insufficient funds.	Contact the buyer so that he or she credits the account, and re-submit the payment (see next chapter).
DNOR	BIC code of the debtor's bank unknown or does not match the IBAN.	This may occur if the IBAN corresponds to an account in Switzerland but the debtor's address has not been transmitted.
AM05	Duplicate collection Collection has already been made. This could either be due to an incorrect transaction ID, or if two transactions have been submitted.	Check whether the collection is really duplicate.
BE05	Identifier of the Creditor Incorrect The provided Creditor ID is incorrect or invalid.	Send the the ICS (Creditor Identifier) to the administrator of your payment gateway.
FF01 (Previously MD03)	Invalid file format Technical error	Contact your payment gateway.
FF05	Invalid local instrument code Technical error	Contact your payment gateway.
MD01	No mandate	Check the expiration date in the Merchant Back Office.

ISO code	Reason - Description	Required action
	The mandate for the buyer's account is no longer valid. The mandate may have been canceled by the buyer. It is also used when a customer requests a refund, stating that the transaction was not authorized. This can occur up to 13 months after the settlement.	In case of a refund, contact the buyer for more information.
MD02	Mandate data missing or incorrect Technical error.	Contact your payment gateway.
MD03	Invalid file format Technical error.	Contact your payment gateway.
MD06	Return of funds requested by end customer The buyer has requested a refund of an authorized transaction. May be received up to 8 weeks after the settlement.	Contact the buyer.
MD07	End customer deceased You have attempted to set up a mandate for the account of a deceased individual. Extremely rare.	Terminate the agreement.
MS02	No specified reason by customer The buyer rejects the direct debit. This code may be received pre- or post-settlement, depending on how quickly the buyer's bank responds to the rejection.	Contact the buyer.
MS03	Non specified reason by agent One of the most common error codes. Often used when banks choose to not provide more specific explanatory codes, such as MD07 and AM04, due to data protection reasons.	Contact the buyer to request if he or she can contact their bank to find out the reason for rejection.
RC01	Invalid BIC The BIC code provided by the buyer is incorrect.	Contact the buyer to confirm the correct BIC. Via the Merchant Back Office, change the token in order to update the BIC. Re-submit the payment (see next chapter).
RR01	Missing debtor account or identification Technical error.	Contact your payment gateway.
RR02	Missing debtor name or address Technical error.	Contact your payment gateway.
RR03	Missing creditor name or address Technical error.	Contact your payment gateway.
RR04	Regulatory reason R04 can be used for a regulatory reason different from RR01, RR02 or RR03.	Contact your bank and request that they contact the buyer's bank to find out the reason of the rejection.
SL01	Specific service offered by debtor agent The request clashes with specific instructions that the buyer has configured for his or her account. E.g.: the creditor is blacklisted.	Contact the buyer.

ISO code	Reason - Description	Required action
TM01	Cut off time The time limit has passed.	Contact your payment gateway.

Table 2: ISO codes and reasons for chargeback

15.4. Reissuing a direct debit request

The payment gateway will not automatically re-submit a chargeback, regardless of its reason. It is up to you to determine if a new payment request must be made, taking into account the return code of the chargeback.

Payments created in this way will become new transactions, unrelated to the failed payments. Their lifecycle will be the same as for any new transaction (sending a pre-notification, calculation of delays, etc.).

In order to re-submit a direct debit request for a rejected payment, make a MOTO payment.

1. Select the **Management > Transactions** menu.
2. Click on the **MOTO payment** button located in the upper part of the transaction details view.
3. Select the SEPA payment method.
4. Enter the UMR in the token field.
5. Populate the necessary fields to complete the payment.



The upcoming payments will automatically use the new banking details.

16. ESTABLISHING INTERACTION WITH THE PAYMENT GATEWAY

The interaction with the payment gateway is described in the *Implementation Guide Hosted Payment Page*, available in our online documentation archive.

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 or Instant Payment Notification).
- Via 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 known only to the merchant and the payment gateway.

17. SETTING UP NOTIFICATIONS

The Merchant Back Office provides several types of notifications.

- Instant Payment Notification URL call
- E-mail sent to the merchant
- E-mail sent to the buyer
- SMS sent to the merchant
- SMS sent to the buyer

They allow to manage the events (payment accepted, payment abandoned by the buyer, payment canceled by the merchant, etc.) that will trigger a notification sent to the merchant website, the merchant or the buyer.



The notifications of Instant Payment Notification URL call type are very important as they represent the only reliable way for the merchant website to obtain the payment result.

If the payment gateway is unable to access the URL of your page, an e-mail will be sent to the shop administrator.

It contains:

- The HTTP code of the encountered error
- Parts of error analysis
- Its consequences
- Instructions via the Merchant Back Office to resend the request to the previously defined URL.

To access notification rule management:

Go to the following menu **Settings > Notification rules**.

Instant Payment Notification		<input checked="" type="checkbox"/> E-mail sent to the merchant	<input checked="" type="checkbox"/> E-mail sent to the buyer
Enabled		Reference	
✗	Instant Payment Notification URL on batch authorization		
✓	Instant Payment Notification URL at the end of the payment		
✗	Instant Payment Notification URL on batch change		
✗	Instant Payment Notification URL on cancellation		
✗	Instant Payment Notification URL on an operation coming from the Back Office		

17.1. Notifying the debtor

Two e-mails can be sent to the debtor by the payment gateway:

17.1.1. A pre-notification e-mail

The pre-notification e-mail is disabled by default (see chapter [Enabling pre-notification](#) on page 53).

The payment gateway may send an e-mail notification to the debtor maximum 14 business days before the due date.

The two parties may decide to reduce this delay (see chapter [How SEPA direct debit works](#) on page 15).

Pre-notification e-mail template:

SHOP : ■■■■■
URL address : ■■■■■
IDENTIFIER : a6f3e3e04f254bb89c10863af758dd46

Hello Mrs./Ms./Mr. ■■■■■ ,

This e-mail informs you that a direct debit of 1.00 EUR will occur on 07/05/2015 on your bank account. Think about having the necessary funds available on this account.

■■■■■ thanks you for your confidence.

Note:

The pre-notification e-mail template can be customized (see chapter [Customizing the pre-notification e-mail](#) on page 53).

17.1.2. An e-mail to notify the payment

The payment gateway notifies the debtor by e-mail to confirm that his or her request for direct debit has been registered.

Example of an e-mail for this notification:

SHOP : ■■■■■
URL address : ■■■■■
IDENTIFIER : a6f3e3e04f254bb89c10863af758dd46

Hello Mrs./Ms./Mr. ■■■■■ ,

This e-mail confirms that your bank details have been recorded on the payment platform for the ■■■■■ shop.

Your buyer ID is: a6f3e3e04f254bb89c10863af758dd46

The copy of the associated SEPA direct debit order is attached to this message.

Associated payment method details

Payment method : SEPA direct debit
Card number : DE97XXXXXXXXXXXX3000_COBADEFFXXX

PayZen thanks you for your confidence.

17.1.3. Enabling pre-notification

In order to enable the pre-notification e-mail:

1. In your Merchant Back Office, go to the following menu: **Settings > Notification rules**.
2. Select the **E-mail sent to the buyer** tab.
3. Right-click the **Pre-notification e-mail of SEPA direct debit to the buyer** label.
4. Select **Enable the rule**.

17.1.4. Customizing the pre-notification e-mail

The pre-notification e-mail template can be customized. To do this:

1. From your Merchant Back Office, click **Settings > Notification rules > E-mail sent to the buyer** tab.
2. Right-click the **Pre-notification e-mail of SEPA direct debit to the buyer** label.
3. Select **Manage the rule** in the menu.
4. In **Buyer e-mail settings**, select the language of the e-mail and click **Restore some default values** to customize the contents of the e-mail.
Before saving your modifications, you can preview the pre-notification e-mail by clicking the **E-mail preview** button situated in the bottom of the **Management of the rule** dialog box.
5. Click **Save**.

17.1.5. Updating the e-mail address

1. Select the **Management > Recurring payments** menu from your Merchant Back Office
2. In the **Token** tab, select the buyer
3. Click the **Modify** button in the top left corner of the table
4. Enter the new e-mail address

Note

The e-mail address can also be updated via the mandate update web service.

17.2. Setting up the Instant Payment Notification

This rule allows to notify the merchant website in the following cases:

- Payment accepted
- Payment refused
- Token creation or update
- Creation of a recurring payment

This notification is required for communicating the result of a payment request, token or recurring payment creation.

It will inform the merchant website of the result even if the buyer has not clicked the “Return to the shop” button.

1. Right-click **Instant Payment Notification URL at the end of the payment**.
2. Select **Manage the rule**.
3. Enter the **E-mail address(es) to notify in case of failure** field in the **General settings** section.
To specify several e-mail addresses, separate them with a semi-colon.
4. Check the box **Automatic retry in case of failure** if you wish to authorize the gateway to automatically resend the notification in case of a failure (can be done up to 4 times).
5. In the **Instant Payment Notification URL of the API form V1, V2** section, specify the URL of your page in the fields **URL to notify in TEST mode** and **URL to notify in PRODUCTION mode** if you wish to receive notifications in the API form format.
6. Save the changes.

17.3. Setting up notifications in case of abandoned or canceled payments

This rule allows to notify the merchant website in the following cases:

- When the buyer abandons/cancels a payment - via the **Cancel and return to shop** button.
- When the buyer has not completed the payment and the payment session has expired.

The maximum length of a payment session is 10 minutes.

This rule is **disabled by default**.

1. Right-click **Instant Payment Notification URL on cancellation**.
2. Select **Manage the rule**.
3. Enter the **E-mail address(es) to notify in case of failure** field in the **General settings** section.
To specify several e-mail addresses, separate them with a semi-colon.
4. Check the box **Automatic retry in case of failure** if you wish to authorize the gateway to automatically resend the notification in case of a failure (can be done up to 4 times).
5. In the **Instant Payment Notification URL of the API form V1, V2** section, specify the URL of your page in the fields **URL to notify in TEST mode** and **URL to notify in PRODUCTION mode** if you wish to receive notifications in the API form format.
6. Save the changes.
7. Enable the rule by right-clicking **Instant Payment Notification URL on cancellation** and select **Enable the rule**.

17.4. Setting up a notification upon creating a recurring payment

This rule allows to notify the merchant website in the following cases:

- When the payment gateway creates a new installment date of a recurring payment.
- Upon each new payment attempt, after a recurring payment installment has been refused.

Requires the activation of the anticipated authorization option.

This rule is **disabled by default**.

1. Right-click **Instant Payment Notification URL when creating a recurring payment**.
2. Select **Manage the rule**.
3. Enter the **E-mail address(es) to notify in case of failure** field in the **General settings** section.
To specify several e-mail addresses, separate them with a semi-colon.
4. Check the box **Automatic retry in case of failure** if you wish to authorize the gateway to automatically resend the notification in case of a failure (can be done up to 4 times).
5. In the **Instant Payment Notification URL of the API form V1, V2** section, specify the URL of your page in the fields **URL to notify in TEST mode** and **URL to notify in PRODUCTION mode** if you wish to receive notifications in the API form format.
6. In the **REST API Instant Payment Notification URL** section, specify the URL of your page in the fields **Target URL of the IPN to notify in TEST mode** and **Target URL of the IPN to notify in PRODUCTION mode** if you are using the JavaScript client.
7. Save the changes.
8. Enable the rule by right-clicking **Instant Payment Notification URL when creating a recurring payment** and select **Enable the rule**.

17.5. Instant Payment Notification URL on an operation coming from the Back Office

This rule allows to notify the merchant website about every operation made via the Merchant Back Office. See the chapter [Managing SEPA Direct Debit transactions from the Merchant Back Office](#) on page 92 for more information.

1. Right-click **Instant Payment Notification URL on an operation coming from the Back Office**.
2. Select **Manage the rule**.
3. Enter the **E-mail address(es) to notify in case of failure** field in the **General settings** section.
To specify several e-mail addresses, separate them with a semi-colon.
4. Check the box **Automatic retry in case of failure** if you wish to authorize the gateway to automatically resend the notification in case of a failure (can be done up to 4 times).
5. In the **Instant Payment Notification URL of the API form V1, V2** section, specify the URL of your page in the fields **URL to notify in TEST mode** and **URL to notify in PRODUCTION mode**.
6. Save the changes.
7. Enable the rule by right-clicking **Instant Payment Notification URL on an operation coming from the Back Office** and select **Enable the rule**.

17.6. Setting up the notification for validating SEPA files sent to the bank

The payment gateway can systematically notify the merchant when a SEPA Debit/Credit Transfer order file is sent to its bank.

The purpose of this notification is to inform the merchant that he or she must validate the transfer file via his or her bank in order to receive the money. The merchant must first get in touch with his or her bank to know more about the validation process.

To send the notification:

- Setting up the SMS notification

1. Select the **SMS sent to the merchant** tab.
2. Right click on **SMS delivery confirmation of a direct debit/credit transfer file**.
3. Select **Manage the rule**.
The **Management of the rule** dialog box appears.
4. Enter the phone number to notify in the **Settings** tab.
5. Click **Save**.

- Setting up the e-mail notification

1. Select the **E-mail sent to the merchant** tab.
2. Right-click **E-mail delivery confirmation of a direct debit/credit transfer file**.
3. Select **Manage the rule**.
The **Management of the rule** dialog box appears.
4. Enter the e-mail address to notify in the **Settings** tab.

We recommend to enter a generic e-mail address rather than a personal address to ensure that in the event of the recipient's absence, the files will still be validated.

5. Click **Save**.

Note:

These labels and the content of these messages can be customized.

The names of variables are provided for information purposes. You must comply with the naming conventions if you wish to use them.

17.7. Configuring e-mails sent to the buyer

From the **E-mail sent to the buyer** tab:

1. Right-click the rule to be modified and select **Enable the rule**.
2. Right-click the rule again and select **Manage the rule**.
The rule management wizard appears.
3. In the General settings section, you can customize the label of the rule.
4. To customize the e-mail content:
 - a. Click **Buyer e-mail settings**.
 - b. Select the template of the e-mail to apply.
 - c. Select the language that you would like to update.
 - d. Click **Customize default text values** if you wish to edit the body and the subject of the “default” e-mail message.
 - e. Click on **Fields to include** to display the list of fields available for e-mail customization.
 - f. Select the fields that you wish to include. A detailed summary of the request processing will be added to the body of the e-mail.



To preview the changes, click **Preview the e-mail** at the bottom of the dialog box.

5. In order to change the events that trigger the notification:
 - a. Click the **Rule conditions** tab.
A condition is composed of a variable, a comparison operator and a reference value.
Example: "mode = TEST", "amount exceeding 1000". During the execution of a rule, the value of a variable is retrieved and compared to the reference value.
 - b. Double-click on an existing condition to edit it.
 - c. Click **Add** to create a new condition.
All the conditions must be validated for the rule to be executed.
6. Click **Save**.

18. GENERATING A PAYMENT FORM

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

```
<form method="POST" action="https://scelliuspaiement.labanquepostale.fr/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://scelliuspaiement.labanquepostale.fr/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 needs.
 - The signature that ensures 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 **Data dictionary** chapter also available in the online document archive.

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

This will allow for the special characters (accents, punctuation marks, etc.) to be correctly interpreted by the payment gateway. Otherwise, the signature will be computed incorrectly and the form will be rejected.

- The **Pay** button for submitting the data:

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

Different use cases are presented in the chapters below. They will allow you to adapt your payment form to your needs.


The following table lists the different formats that you can encounter when building your form.

Notation	Description
a	Alphabetic characters (from 'A' to 'Z' and from 'a' to 'z')
n	Numeric characters
s	Special characters
an	Alphanumeric characters
ans	Alphanumeric and special characters (except '<' and '>')
3	Fixed length of 3 characters
..12	Variable length up to 12 characters
json	<p>JavaScript Object Notation.</p> <p>Object containing key/value pairs separated by commas.</p> <p>It starts with a left brace "{" and ends with a right brace "}".</p> <p>Each key / value pair contains the name of the key between double-quotes followed by ":", followed by a value.</p> <p>The name of the key must be alphanumeric.</p> <p>The value can be:</p> <ul style="list-style-type: none"> • a chain of characters (in this case it must be framed by double-quotes) • a number • an object • a table • a boolean • empty <p>Example: {"name1":45,"name2":"value2", "name3":false}</p>
bool	Boolean. Can be populated with the true or false value.
enum	<p>Defines a field with a complete list of values.</p> <p>The list of possible values is given in the field definition.</p>
Enum list	<p>List of values separated by a ";".</p> <p>The list of possible values is given in the field definition.</p> <p>Example: vads_available_languages=fr;en</p>
map	<p>List of key / value pairs separated by a ";".</p> <p>Each key / value pair contains the name of the key followed by "=", followed by a value.</p> <p>The value can be:</p> <ul style="list-style-type: none"> • a chain of characters • a boolean • a json object • an xml object <p>The list of possible values for each key/value pair is provided in the field definition.</p> <p>Example: vads_theme_config=SIMPLIFIED_DISPLAY=true;RESPONSIVE_MODEL=Model_1</p>

18.1. Creating a one-off direct debit request



In case of one-off direct debit, the buyer authorizes the merchant to withdraw the whole amount of the purchase in one go.

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

Field name	Description	Value
vads_action_mode	Acquisition mode for bank data.	INTERACTIVE
vads_amount	Payment amount in the smallest currency unit	E.g.: 4525 for EUR 45.25
vads_capture_delay	Capture delay	0
vads_ctx_mode	Operating mode	TEST or PRODUCTION
vads_currency	Currency code	978
vads_cust_email	Buyer's e-mail address.  If the buyer's e-mail address is pre-filled on the payment page, the buyer can modify it when signing the mandate. If the buyer's e-mail address is unknown, the buyer will be prompted to specify it on the payment page.	E.g.: mymail@example.com
vads_page_action	Action to perform	PAYMENT
vads_payment_cards	Payment method	SDD
vads_payment_config	Payment type	SINGLE
vads_site_id	Shop ID	E.g.: 12345678
vads_trans_date	Date and time of the payment form in UTC format	E.g.: 20200516080441
vads_trans_id	Transaction number	E.g.: 3N2kj2
vads_version	Version of the exchange protocol	V2

2. Populate the **vads_capture_delay** field with **0** for the fund transfer to be executed maximum 14 calendar days after the purchase.

3. It is recommended to transmit the buyer's full address. Some banks check whether this information is present before accepting a direct debit request. Moreover, when this information is specified, the corresponding fields will be pre-filled on the mandate signature page.

Field name	Description	Value
vads_cust_address	Buyer's address <div>  <p>The buyer's postal address is required if the client has a bank account in the following departments, territories or countries: Switzerland, Monaco, San Marino, Mayotte, St. Pierre and Miquelon, Guernsey, Jersey, Isle of Man.</p> </div>	
vads_cust_cell_phone	Buyer's cell phone number for sending the security code (by SMS). <div>  <p>If the buyer's cell phone number (the phone number registered during the subscription) is pre-filled on the payment page, the buyer can modify it when signing the mandate. Only GSM-users having recorded this phone number at the moment during their subscription can receive an SMS with this access code for single use. If the buyer's cell phone number is unknown, the buyer will be prompted to specify it on the payment page.</p> </div>	E.g.: 0612345678
vads_cust_country	Allows to specify the country code in compliance with the ISO 3166 standard.	E.g.: FR
vads_cust_first_name	Allows to specify the buyer's first name.	E.g.: John
vads_cust_last_name	Allows to specify the buyer's last name.	E.g.: Smith
vads_cust_title	Allows to specify the buyer's title.	E.g.: <ul style="list-style-type: none"> • Mr • Mrs • Ms



*If the value of the **vads_cust_cell_phone** field is provided in international format (+336xxxxxxx) or if the fields **vads_cust_cell_phone** and **vads_cust_country** are both populated, the security code will be systematically sent by SMS regardless of the signature mode configured in the shop.*

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 a form:


```
<form method="POST" action="https://scelliuspaiement.labanquepostale.fr/vads-payment/">
<input type="hidden" name="vads_action_mode" value="INTERACTIVE" />
<input type="hidden" name="vads_amount" value="2990" />
<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_cust_country" value="FR" />
<input type="hidden" name="vads_cust_cell_phone" value="0612345678" />
<input type="hidden" name="vads_cust_email" value="mymail@example.com" />
<input type="hidden" name="vads_cust_first_name" value="Paul" />
<input type="hidden" name="vads_cust_last_name" value="Juve" />
<input type="hidden" name="vads_cust_title" value="Mr" />
<input type="hidden" name="vads_page_action" value="PAYMENT" />
<input type="hidden" name="vads_payment_cards" value="SDD" />
<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="20200516080441" />
<input type="hidden" name="vads_trans_id" value="3N2kj2" />
<input type="hidden" name="vads_validation_mode" value="0" />
<input type="hidden" name="vads_version" value="V2" />
<input type="hidden" name="signature" value="RR0vv+fsP4U27wB2cqm67eDUv0gMXrvCSD051KcWI28=" />
<input type="submit" name="pay" value="pay" />
</form>
```



18.2. Creating a recurring mandate signature request

To request the signature of a recurring SEPA mandate:

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

Field name	Description	Value
vads_action_mode	Acquisition mode for bank data.	INTERACTIVE
vads_ctx_mode	Operating mode	TEST or PRODUCTION
vads_cust_email	Buyer's e-mail address.  If the buyer's e-mail address is pre-filled on the payment page, the buyer can modify it when signing the mandate. If the buyer's e-mail address is unknown, the buyer will be prompted to specify it on the payment page.	E.g.: mymail@example.com
vads_page_action	Action required to register a mandate and make a payment	REGISTER
vads_payment_cards	Payment method	SDD
vads_site_id	Shop ID	E.g.: 12345678
vads_trans_date	Date and time of the payment form in UTC format	E.g.: 20200519130128
vads_version	Version of the exchange protocol	V2

2. It is recommended to transmit the buyer's full address. Some banks check whether this information is present before accepting a direct debit request. Moreover, when this information is specified, the corresponding fields will be pre-filled on the mandate signature page.

Field name	Description	Value
vads_cust_address	Buyer's address  The buyer's postal address is required if the client has a bank account in the following departments, territories or countries: Switzerland, Monaco, San Marino, Mayotte, St. Pierre and Miquelon, Guernsey, Jersey, Isle of Man.	
vads_cust_cell_phone	Buyer's cell phone number for sending the security code (by SMS).  If the buyer's cell phone number (the phone number registered during the subscription) is pre-filled on the payment page, the buyer can modify it when signing the mandate. Only GSM-users having recorded this phone number at	E.g.: 0612345678

Field name	Description	Value
	<p>the moment during their subscription can receive an SMS with this access code for single use.</p> <p>If the buyer's cell phone number is unknown, the buyer will be prompted to specify it on the payment page.</p>	
vads_cust_country	Allows to specify the country code in compliance with the ISO 3166 standard.	E.g.: FR
vads_cust_first_name	Allows to specify the buyer's first name.	E.g.: John
vads_cust_last_name	Allows to specify the buyer's last name.	E.g.: Smith
vads_cust_title	Allows to specify the buyer's title.	E.g.: <ul style="list-style-type: none"> • Mr • Mrs • Ms



*If the value of the **vads_cust_cell_phone** field is provided in international format (+336xxxxxxx) or if the fields **vads_cust_cell_phone** and **vads_cust_country** are both populated, the security code will be systematically sent by SMS regardless of the signature mode configured in the shop.*

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 76).


Example of a form:

```
<form method="POST" action="https://scelliuspaiement.labanquepostale.fr/vads-payment/">
<input type="hidden" name="vads_action_mode" value="INTERACTIVE" />
<input type="hidden" name="vads_ctx_mode" value="TEST" />
<input type="hidden" name="vads_cust_email" value="firstName.lastName@example.com" />
<input type="hidden" name="vads_page_action" value="REGISTER" />
<input type="hidden" name="vads_payment_cards" value="SDD" />
<input type="hidden" name="vads_site_id" value="12345678" />
<input type="hidden" name="vads_trans_date" value="20200519130128" />
<input type="hidden" name="vads_version" value="V2" />
<input type="hidden" name="signature" value="jv08cn/Qa5jRA8c180jzN/TlhNG2oUY8b4sJwNXF/70=" />
<input type="submit" name="pay" value="Pay"/>
</form>
```



18.3. Registering a recurring mandate and making a payment


In order to register a recurring mandate and make a payment:

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

Field name	Description	Value
vads_action_mode	Acquisition mode for bank data.	INTERACTIVE
vads_amount	Payment amount in the smallest currency unit	E.g.: 4525 for EUR 45.25
vads_ctx_mode	Operating mode	TEST or PRODUCTION
vads_currency	Currency euro code in compliance with the ISO 4217 standard.	E.g.: 978 for euro (EUR)
vads_cust_email	Buyer's e-mail address.  If the buyer's e-mail address is pre-filled on the payment page, the buyer can modify it when signing the mandate. If the buyer's e-mail address is unknown, the buyer will be prompted to specify it on the payment page.	E.g.: mymail@example.com
vads_page_action	Action required to register a mandate and make a payment	REGISTER_PAY
vads_payment_cards	Payment method	SDD
vads_payment_config	Payment type	SINGLE for immediate payment MULTI for installment payment
vads_site_id	Shop ID	E.g.: 12345678
vads_trans_date	Date and time of the payment form in UTC format	E.g.: 20200519130128
vads_trans_id	Transaction number	E.g.: V8y52j
vads_version	Version of the exchange protocol	V2

2. It is recommended to transmit the buyer's full address. Some banks check whether this information is present before accepting a direct debit request. Moreover, when this information is specified, the corresponding fields will be pre-filled on the mandate signature page.

Field name	Description	Value
vads_cust_address	Buyer's address  The buyer's postal address is required if the client has a bank account in the following departments, territories or countries: Switzerland, Monaco, San Marino, Mayotte, St. Pierre and Miquelon, Guernsey, Jersey, Isle of Man.	

Field name	Description	Value
vads_cust_cell_phone	<p>Buyer's cell phone number for sending the security code (by SMS).</p> <div>  <p>If the buyer's cell phone number (the phone number registered during the subscription) is pre-filled on the payment page, the buyer can modify it when signing the mandate. Only GSM-users having recorded this phone number at the moment during their subscription can receive an SMS with this access code for single use.</p> <p>If the buyer's cell phone number is unknown, the buyer will be prompted to specify it on the payment page.</p> </div>	E.g.: 0612345678
vads_cust_country	Allows to specify the country code in compliance with the ISO 3166 standard.	E.g.: FR
vads_cust_first_name	Allows to specify the buyer's first name.	E.g.: John
vads_cust_last_name	Allows to specify the buyer's last name.	E.g.: Smith
vads_cust_title	Allows to specify the buyer's title.	E.g.: <ul style="list-style-type: none"> • Mr • Mrs • Ms



*If the value of the **vads_cust_cell_phone** field is provided in international format (+336xxxxxxx) or if the fields **vads_cust_cell_phone** and **vads_cust_country** are both populated, the security code will be systematically sent by SMS regardless of the signature mode configured in the shop.*

3. 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:

```
<form method="POST" action="https://scelliuspaiement.labanquepostale.fr/vads-payment/">
<input type="hidden" name="vads_action_mode" value="INTERACTIVE" />
<input type="hidden" name="vads_amount" value="2990" />
<input type="hidden" name="vads_ctx_mode" value="TEST" />
<input type="hidden" name="vads_currency" value="978" />
<input type="hidden" name="vads_cust_email" value="nom.prenom@exemple.com" />
<input type="hidden" name="vads_page_action" value="REGISTER_PAY" />
<input type="hidden" name="vads_payment_cards" value="SDD" />
<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="20200519130128" />
<input type="hidden" name="vads_trans_id" value="V8y52j" />
<input type="hidden" name="vads_version" value="V2" />
<input type="hidden" name="signature" value="oXN8x9nciau7mmN4Y8KKo4zOB234CbHY/I5qR9KJgTk=" />
<input type="submit" name="pay" value="Pay"/>
</form>
```

18.4. Registering a recurring mandate and creating a recurring payment with fixed due dates and amounts



To register a recurring mandate and create a recurring payment with fixed amounts and due dates:

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

Field name	Description	Value
vads_action_mode	Acquisition mode for bank data.	INTERACTIVE
vads_ctx_mode	Operating mode	TEST or PRODUCTION
vads_cust_email	<p>Buyer's e-mail address.</p> <div>  <p>If the buyer's e-mail address is pre-filled on the payment page, the buyer can modify it when signing the mandate. If the buyer's e-mail address is unknown, the buyer will be prompted to specify it on the payment page.</p> </div>	E.g.: mymail@example.com
vads_page_action	Action to perform	REGISTER_SUBSCRIBE
vads_payment_cards	Payment method	SDD
vads_site_id	Shop ID	E.g.: 12345678
vads_sub_amount	Installment amount in the smallest currency unit	E.g.: 4525 for EUR 45.25
vads_sub_currency	Currency euro code in compliance with the ISO 4217 standard.	Possible value: 978
vads_sub_desc	<p>Rule for recurring payments to apply according to the iCalendar RFC5545 specification. Recurring payments can be made daily, weekly or monthly. You can specify the day or month number. For example:</p> <ul style="list-style-type: none"> the 10th of each month: FREQ=MONTHLY;BYMONTHDAY=10, every 3 months: FREQ=MONTHLY;INTERVAL=3, every month except August: FREQ=YEARLY;BYMONTH=1,2,3,4,5,6 <p><i>Note: the string must not contain space characters.</i></p> <p>It is recommended to always choose a fixed date (for example 05 or 15 of the month) or a fixed day of the week (for example Wednesday) in order to</p>	<p>Examples:</p> <ul style="list-style-type: none"> To program installment payments for the 15th of each month during 12 months: <pre>RRULE:FREQ=MONTHLY; COUNT=12;BYMONTHDAY=15</pre> To program installment payments for the 4th tuesday of each month: <pre>RRULE:FREQ=MONTHLY;BYDAY=4TU</pre> In order to define a weekly recurring payment: <pre>RRULE:FREQ=WEEKLY</pre> In order to define a recurring payment every two weeks,

Field name	Description	Value
	facilitate the validation of the direct debit files.	<p>on the same day and every 7 days:</p> <pre>RRULE:FREQ=WEEKLY;INTERVAL=2</pre> <ul style="list-style-type: none"> To program installment payments for the last day of each month for 12 months: <pre>RRULE:FREQ=MONTHLY; BYMONTHDAY=28,29,30,31; BYSETPOS=-1;COUNT=12</pre>
vads_sub_effect_date	Recurring payment start date. For technical reasons, the due date must be at least 14 calendar days after the submission date of the form (vads_trans_date).	E.g.: 20161002
vads_trans_date	Date and time of the payment form in UTC format	E.g.: 20200526125112
vads_version	Version of the exchange protocol	V2

2. It is recommended to transmit the buyer's full address. Some banks check whether this information is present before accepting a direct debit request. Moreover, when this information is specified, the corresponding fields will be pre-filled on the mandate signature page.

Field name	Description	Value
vads_cust_address	<p>Buyer's address</p> <div>  <p>The buyer's postal address is required if the client has a bank account in the following departments, territories or countries: Switzerland, Monaco, San Marino, Mayotte, St. Pierre and Miquelon, Guernsey, Jersey, Isle of Man.</p> </div>	
vads_cust_cell_phone	<p>Buyer's cell phone number for sending the security code (by SMS).</p> <div>  <p>If the buyer's cell phone number (the phone number registered during the subscription) is pre-filled on the payment page, the buyer can modify it when signing the mandate. Only GSM-users having recorded this phone number at the moment during their subscription can receive an SMS with this access code for single use. If the buyer's cell phone number is unknown, the buyer will be prompted to specify it on the payment page.</p> </div>	E.g.: 0612345678
vads_cust_country	Allows to specify the country code in compliance with the ISO 3166 standard.	E.g.: FR

Field name	Description	Value
vads_cust_first_name	Allows to specify the buyer's first name.	E.g.: John
vads_cust_last_name	Allows to specify the buyer's last name.	E.g.: Smith
vads_cust_title	Allows to specify the buyer's title.	E.g.: <ul style="list-style-type: none"> • Mr • Mrs • Ms



If the value of the **vads_cust_cell_phone** field is provided in international format (+336xxxxxxx) or if the fields **vads_cust_cell_phone** and **vads_cust_country** are both populated, the security code will be systematically sent by SMS regardless of the signature mode configured in the shop.

3. If you wish to set up a recurring payment for which the amount of the first installment(s) would differ from the ones configured by the **vads_sub_amount** field, use the fields below:

Field name	Description	Value
vads_sub_init_amount	Amount of the first installments. The number of the first installments is defined by vads_sub_init_amount_number . Can't be set to 0. To define a recurring payment where the n first months are free, all you need to do is set the start date (vads_sub_effect_date) to n months later.	2500
vads_sub_init_amount_number	Number of installments to which the amount, defined by vads_sub_init_amount , will be applied.	3

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 a form:


```
<form method="POST" action="https://scelliuspaiement.labanquepostale.fr/vads-payment/">
<input type="hidden" name="vads_action_mode" value="INTERACTIVE" />
<input type="hidden" name="vads_ctx_mode" value="TEST" />
<input type="hidden" name="vads_cust_email" value="name.firstname@example.com" />
<input type="hidden" name="vads_page_action" value="REGISTER_SUBSCRIBE" />
<input type="hidden" name="vads_payment_cards" value="SDD" />
<input type="hidden" name="vads_site_id" value="12345678" />
<input type="hidden" name="vads_sub_amount" value="2990" />
<input type="hidden" name="vads_sub_currency" value="978" />
<input type="hidden" name="vads_sub_desc" value="RRULE:FREQ=MONTHLY;BYMONTHDAY=15" />
<input type="hidden" name="vads_sub_effect_date" value="20200609" />
<input type="hidden" name="vads_trans_date" value="20200526125112" />
<input type="hidden" name="vads_version" value="V2" />
<input type="hidden" name="signature" value="moN40Cp08Kp8WCe80HyebvEfSv2uzTajPHefcB9iIHk=" />
<input type="submit" name="pay" value="Pay"/></form>
```

18.5. Creating a one-click payment


This use case requires an already existing and valid mandate.


To make one-click payment:

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

Field name	Description	Value
vads_action_mode	Acquisition mode for bank data.	INTERACTIVE
vads_amount	Payment amount in the smallest currency unit (cents for euro)	E.g.: 3000 for €30.00
vads_ctx_mode	Operating mode	TEST or PRODUCTION
vads_currency	Currency euro code in compliance with the ISO 4217 standard.	978
vads_cust_email	Buyer's e-mail address. <div> If the buyer's e-mail address is pre-filled on the payment page, the buyer can modify it when signing the mandate. If the buyer's e-mail address is unknown, the buyer will be prompted to specify it on the payment page.</div>	E.g.: e-mail@example.com
vads_identifier	Unique Mandate Reference	E.g.: 22e75ce992afe18da10
vads_page_action	Action required to register a mandate and make a payment	PAYMENT
vads_payment_cards	Payment method	SDD
vads_payment_config	Payment type	SINGLE for immediate payment MULTI for installment payment
vads_site_id	Shop ID	E.g.: 12345678
vads_trans_date	Date and time of the payment form in UTC format	E.g.: 20200526130412
vads_trans_id	Transaction number	E.g.: 3N2kj2
vads_version	Version of the exchange protocol	V2

2. Populate the **vads_identifier** field with the already existing and valid mandate identifier (UMR).
3. It is recommended to transmit the buyer's full address. Some banks check whether this information is present before accepting a direct debit request. Moreover, when this information is specified, the corresponding fields will be pre-filled on the mandate signature page.

Field name	Description	Value
vads_cust_address	Buyer's address <div> The buyer's postal address is required if the client has a bank account in the following departments, territories</div>	

Field name	Description	Value
	or countries: Switzerland, Monaco, San Marino, Mayotte, St. Pierre and Miquelon, Guernsey, Jersey, Isle of Man.	
vads_cust_cell_phone	<p>Buyer's cell phone number for sending the security code (by SMS).</p> <p> If the buyer's cell phone number (the phone number registered during the subscription) is pre-filled on the payment page, the buyer can modify it when signing the mandate. Only GSM-users having recorded this phone number at the moment during their subscription can receive an SMS with this access code for single use.</p> <p>If the buyer's cell phone number is unknown, the buyer will be prompted to specify it on the payment page.</p>	E.g.: 0612345678
vads_cust_country	Allows to specify the country code in compliance with the ISO 3166 standard.	E.g.: FR
vads_cust_first_name	Allows to specify the buyer's first name.	E.g.: John
vads_cust_last_name	Allows to specify the buyer's last name.	E.g.: Smith
vads_cust_title	Allows to specify the buyer's title.	E.g.: <ul style="list-style-type: none"> • Mr • Mrs • Ms



*If the value of the **vads_cust_cell_phone** field is provided in international format (+336xxxxxxx) or if the fields **vads_cust_cell_phone** and **vads_cust_country** are both populated, the security code will be systematically sent by SMS regardless of the signature mode configured in the shop.*

4. Compute the value of the **signature** field using all the fields of your form that start with **vads_** (see chapter [Computing the signature](#)).

18.6. Updating a mandate

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

Field name	Description	Value
vads_action_mode	Acquisition mode for bank data.	INTERACTIVE
vads_ctx_mode	Operating mode	TEST or PRODUCTION
vads_cust_email	Buyer's e-mail address.	E.g.: mymail@example.com
vads_identifier	Reminding the token ID to be updated.	E.g.: 22e75ce992afe18da10
vads_page_action	Action required to update a mandate and make a payment	REGISTER_UPDATE
vads_site_id	Shop ID	E.g.: 12345678
vads_trans_date	Date and time of the payment form in UTC format	E.g.: 20200526130921
vads_version	Version of the exchange protocol	V2

2. Set **vads_page_action** to **REGISTER_UPDATE**.

Note:

Not to be confused with **REGISTER_UPDATE_PAY** and **REGISTER_UPDATE**.

REGISTER_UPDATE allows to only update a mandate without making a payment.

3. If the debtor's details are updated, the new details must be transmitted via the merchant website. The payment page will not provide entry fields. Use the fields below:

Field name	Description	Value
vads_cust_address	Buyer's address.	
vads_cust_cell_phone	Buyer's cell phone number for sending the security code (by SMS).	E.g.: 0612345678
vads_cust_country	Allows to specify the country code in compliance with the ISO 3166 standard.	E.g.: FR
vads_cust_first_name	Allows to specify the buyer's first name	E.g.: John
vads_cust_last_name	Allows to specify the buyer's last name	E.g.: Smith
vads_cust_title	Allows to specify the buyer's title.	E.g.: <ul style="list-style-type: none">• Mr• Mrs• Ms

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 a form:

```
<form method="POST" action="https://scelliuspaiement.labanquepostale.fr/vads-payment/">
<input type="hidden" name="vads_action_mode" value="INTERACTIVE" />
<input type="hidden" name="vads_ctx_mode" value="TEST" />
<input type="hidden" name="vads_cust_email" value="surname.name@example.com" />
<input type="hidden" name="vads_cust_address" value="rue de l'innovation" />
<input type="hidden" name="vads_cust_zip" value="31670" />
<input type="hidden" name="vads_cust_city" value="Labège" />
<input type="hidden" name="vads_cust_country" value="FR" />
<input type="hidden" name="vads_identifier" value="22e75ce992afe18da10" />
<input type="hidden" name="vads_page_action" value="REGISTER_UPDATE" />
<input type="hidden" name="vads_site_id" value="12345678" />
<input type="hidden" name="vads_trans_date" value="20200526130921" />
```




```
<input type="hidden" name="vads_version" value="V2" />
<input type="hidden" name="signature" value="k+qSipEEba8hsXXmHHc02ulUB9hSHnLt7myuPpBUhxY=" />
<input type="submit" name="pay" value="Pay"/>
</form>
```

18.7. Updating a mandate and making a payment

In order to update a recurring mandate and make a payment:

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

Field name	Description	Value
vads_action_mode	Acquisition mode for bank data.	INTERACTIVE
vads_amount	Payment amount in the smallest currency unit	E.g.: 4525 for EUR 45.25
vads_ctx_mode	Operating mode	TEST or PRODUCTION
vads_currency	Currency euro code in compliance with the ISO 4217 standard.	Possible value: 978 .
vads_cust_email	Buyer's e-mail address. <div> If the buyer's e-mail address is pre-filled on the payment page, the buyer can modify it when signing the mandate. If the buyer's e-mail address is unknown, the buyer will be prompted to specify it on the payment page.</div>	E.g.: mymail@example.com
vads_identifier	Unique Mandate Reference to update.	E.g.: 22e75ce992afe18da10
vads_page_action	Action required to update a mandate and make a payment	REGISTER_UPDATE_PAY
vads_payment_config	Payment type	SINGLE for immediate payment MULTI for installment payment
vads_site_id	Shop ID	E.g.: 12345678
vads_trans_date	Date and time of the payment form in UTC format	E.g.: 20200526131250
vads_trans_id	Transaction number	E.g.: PmZ34k
vads_version	Version of the exchange protocol	V2


2. Set **vads_page_action** to **REGISTER_UPDATE_PAY**.


Note:

Not to be confused with **REGISTER_UPDATE_PAY** and **REGISTER_UPDATE**.

REGISTER_UPDATE allows to only updating a mandate without making a payment.

3. It is recommended to transmit the buyer's full address. Some banks check whether this information is present before accepting a direct debit request. Moreover, when this information is specified, the corresponding fields will be pre-filled on the mandate signature page.

Field name	Description	Value
vads_cust_address	Buyer's address <div> The buyer's postal address is required if the client has a bank account in the following departments, territories</div>	

Field name	Description	Value
	or countries: Switzerland, Monaco, San Marino, Mayotte, St. Pierre and Miquelon, Guernsey, Jersey, Isle of Man.	
vads_cust_cell_phone	<p>Buyer's cell phone number for sending the security code (by SMS).</p> <p> If the buyer's cell phone number (the phone number registered during the subscription) is pre-filled on the payment page, the buyer can modify it when signing the mandate. Only GSM-users having recorded this phone number at the moment during their subscription can receive an SMS with this access code for single use.</p> <p>If the buyer's cell phone number is unknown, the buyer will be prompted to specify it on the payment page.</p>	E.g.: 0612345678
vads_cust_country	Allows to specify the country code in compliance with the ISO 3166 standard.	E.g.: FR
vads_cust_first_name	Allows to specify the buyer's first name.	E.g.: John
vads_cust_last_name	Allows to specify the buyer's last name.	E.g.: Smith
vads_cust_title	Allows to specify the buyer's title.	E.g.: <ul style="list-style-type: none"> • Mr • Mrs • Ms



*If the value of the **vads_cust_cell_phone** field is provided in international format (+336xxxxxxx) or if the fields **vads_cust_cell_phone** and **vads_cust_country** are both populated, the security code will be systematically sent by SMS regardless of the signature mode configured in the shop.*

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 a form:

```
<form method="POST" action="https://scelliuspaiement.labanquepostale.fr/vads-payment/">
<input type="hidden" name="vads_action_mode" value="INTERACTIVE" />
<input type="hidden" name="vads_ctx_mode" value="TEST" />
<input type="hidden" name="vads_cust_email" value="nom.prenom@exemple.com" />
<input type="hidden" name="vads_identifiant" value="22e75ce992afe18da10" />
<input type="hidden" name="vads_page_action" value="REGISTER_UPDATE_PAY" />
<input type="hidden" name="vads_site_id" value="12345678" />
<input type="hidden" name="vads_amount" value="2990" />
<input type="hidden" name="vads_currency" value="978" />
<input type="hidden" name="vads_payment_config" value="SINGLE" />
<input type="hidden" name="vads_trans_id" value="PmZ34k" />
<input type="hidden" name="vads_trans_date" value="20200526131250" />
<input type="hidden" name="vads_version" value="V2" />
<input type="hidden" name="signature" value="rxFdxmHsYrj6OqBy6iXaipdS11ihAI1NfRno5XkhtXY=" />
<input type="submit" name="pay" value="Pay"/>
</form>
```

19. 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 key value is available in your Merchant Back Office via **Settings > Shop > Keys** tab.

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



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

The use of SHA-1 algorithm is deprecated but maintained for compliance reasons.

To compute the signature:

1. Sort the fields whose name begins with **vads_** alphabetical order.
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 the "+" character.
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. **Deprecated.**
 - 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 field **vads_ctx_mode**) 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 field **signature**.

Example of parameters sent to the payment gateway:

```
<form method="POST" action="https://scelliuspaiement.labanquepostale.fr/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. We sort in **alphabetical** order the fields whose name begins with **vads_** :

- 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. We concatenate the value of these fields with 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 field **vads_ctx_mode**) 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=

20. SENDING THE PAYMENT REQUEST

The buyer will be able to finalize his/her purchase once he/she is redirected to the payment page.

The buyer's browser must transmit the payment form data.

20.1. Redirecting the buyer to the payment page

The URL of the payment gateway is:

<https://scelliuspaiement.labanquepostale.fr/vads-payment/>

Example of parameters sent to the payment gateway:

```
<form method="POST" action="https://scelliuspaiement.labanquepostale.fr/vads-payment/">
<input type="hidden" name="vads_action_mode" value="INTERACTIVE" />
<input type="hidden" name="vads_amount" value="1315" />
<input type="hidden" name="vads_currency" value="978" />
<input type="hidden" name="vads_cust_id" value="1234" />
<input type="hidden" name="vads_cust_email" value="jg@sample.com" />
<input type="hidden" name="vads_ctx_mode" value="TEST" />
<input type="hidden" name="vads_order_id" value="CMD012859" />
<input type="hidden" name="vads_page_action" value="PAYMENT" />
<input type="hidden" name="vads_payment_cards" value="VISA;MASTERCARD" />
<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="20200326101407" />
<input type="hidden" name="vads_trans_id" value="362812" />
<input type="hidden" name="vads_version" value="V2" />
<input type="hidden" name="signature" value="NM25DPLKEbtGEHCDHn8MBT4ki6aJI/ODaWhCzCnAfvY=" />
<input type="submit" name="pay" value="Pay"/>
</form>
```

20.2. Processing errors

If the payment gateway detects an error while receiving the form, an error message will appear and the buyer will not be able to proceed to the payment.

In TEST mode

The message indicates the source of the error and provides a link to the error code description to help you fix it.

In PRODUCTION mode

The message simply indicates to the buyer that a technical problem has occurred.

In both cases the merchant receives a notification e-mail.

It contains:

- the source of the error,
- a link to possible causes to facilitate its analysis,
- all the fields of the form.

A description of the error codes with their possible causes is available on our website

<https://scelliuspaiement.labanquepostale.fr/doc/fr-FR/error-code/error-00.html>

21. ANALYZING THE PAYMENT RESULT

To process the payment result, the merchant website must have a separate page with a script (e.g. `analyze_payment.php`).

This page will be automatically called after each payment (whether it has been accepted or declined): the parameters linked to the payment result are sent in POST mode by the payment gateway.

Prerequisites:

- URL of the page that analyzes the payment result must be specified in the Merchant Back Office). See chapter **Setting up notifications**.
- The merchant has to make sure that this URL is available via the payment gateway without redirection. Using redirection leads to losing data presented in POST.
- In case some restrictions are set up by the merchant, the **194.50.38.0/24** IP address range must be authorized.

*Notifications are sent from an IP address in the **194.50.38.0/24** range in Test and Production mode.*

- HTML should not be visible on the page.

Access to images or CSS may slow down the exchange between the payment gateway and the merchant website.

Furthermore, the payment gateway systematically reads the 512 first characters returned by the merchant website.

These characters can be viewed in the transaction history.

- Avoid integrating time-consuming tasks, such as invoice generation or sending e-mails in the script.

The processing time has a direct influence on how long it takes to display the payment summary page. The longer the processing of the notification, the greater the delay for displaying the page.

After 35 seconds, the payment gateway will consider that the call has failed (timeout).

Failed notification (IPN)

In case the call to IPN fails, a notification e-mail is sent to the address specified in the Merchant Back Office (see chapter **Setting up notifications**).

It contains:

- The HTTP code of the encountered error,
- Analysis parts depending on the error,
- Instructions for resending the notification from the Merchant Back Office.

Writing the processing script

The processing script must include at least the following steps:

- Retrieve the field list sent with the POST response
- Compute the signature
- Compare the computed signature with the received signature
- Analyze the nature of the notification
- Retrieve the payment result

The script may check the order status (or any information of your choice) to see if it has not been already updated.

Once these steps are completed, the script can update the database (new order status, stock update, registration of payment information, etc.).

21.1. Analyzing the result of an SDD payment

Below is an excerpt of the data transmitted in the IPN.

Field	Description
vads_acquirer_network	Acquirer network code. Populated with SEPA.
vads_amount	Amount of the direct debit in cents.
vads_capture_delay	Delay before the capture (in days). It is recalculated by the payment gateway according to the required interbank business days. Will be valued at 2 a minimum.
vads_card_brand	Payment method, set to SDD .
vads_card_number	IBAN followed by BIC.
vads_currency	Numeric code of the currency, valued at 978 .
vads_identifier	Unique Mandate Reference (UMR).
vads_occurrence_type	Type of sequence. Populated with UNITAIRE .
vads_presentation_date	Date of funds transfer, in UTC, YYYYMMDDHHMMSS format.
vads_trans_id	Transaction identifier transmitted by the Merchant.
vads_trans_status	Transaction status (see table below).
vads_trans_uuid	Transaction identifier generated by the gateway.

The transaction status is returned in the **vads_trans_status** field:

Value	Description
AUTHORISED	Accepted, waiting for capture A pre-notification has been sent. The direct debit is waiting to be presented at the bank.
ABANDONED	Payment abandoned by the buyer The transaction was not created, and is therefore not visible in the Merchant Back Office .
REFUSED	Refused Status sent in case of 1-click payment when some rules have been enabled in the Risk assessment tab or when the mandate is expired.
AUTHORISED_TO_VALIDATE	To be validated The direct debit request has been made in "manual validation" mode. The merchant must manually validate the transaction in the Merchant Back Office to authorize the transfer of the mandate to the bank. The pre-notification will only be sent after the transaction has been validated. An automatic e-mail is processed every day, several times per day (every 3 hours) that does not require any intervention on your behalf. The transaction can be validated as long as the expiration date of the authorization request has not passed. If the authorization validity period has passed, the payment status changes to EXPIRED . The Expired status is final.

Value	Description
WAITING_AUTHORISATION	<p>Waiting for authorization</p> <p>The pre-notification has not yet been sent.</p> <p>The direct debit request has been made with a capture delay greater or equal to 7 days for a single payment while the pre-notification delay in the contract is 14 days.</p>
WAITING_AUTHORISATION_TO_VALIDATE	<p>To be validated and authorized</p> <p>The capture delay in the bank exceeds the authorization validity period.</p> <p>The Merchant must manually validate the transaction for the authorization request and the capture to occur.</p>

21.2. Analyzing the result of the mandate creation

Mandate signature results in the creation of a VERIFICATION type transaction.

It is not necessary to analyze the status of this transaction.

1. Analyze the result of mandate registration by testing the value of the **vads_identifier_status** field.

Value	Description
ABANDONED	Mandate registration has been abandoned by the debtor (buyer). The mandate has not been created and cannot be viewed in the Merchant Back Office. No direct debits can be made.
CREATED	The mandate has been successfully registered.
NOT_CREATED	The mandate has not been created and cannot be viewed in the Merchant Back Office. No direct debits can be made.
UPDATED	The mandate has been successfully updated.

2. Retrieve mandate details:

Field	Description
vads_card_number	Returns the IBAN and BIC, separated by a “_”. Since the BIC is optional, it may be left out.
vads_expiry_month	Expiration month of the mandate.
vads_expiry_year	Expiration year of the mandate.
vads_identifier	Unique Mandate Reference (UMR).

Example of notification received:

```
vads_amount=0 vads_auth_mode=MARK vads_auth_number= vads_auth_result=00
vads_capture_delay=0 vads_card_brand=SDD
vads_card_number=FR7617515900001234567890135_CEPAPRPP751 vads_payment_certificate=
vads_ctx_mode=TEST vads_currency=978 vads_effective_amount=0
vads_effective_currency=978 vads_site_id=70258842 vads_trans_date=20200518140923
vads_trans_id=509630 vads_trans_uuid=242110375cd543edb90db511cb1e6ac0
vads_validation_mode=0 vads_version=V2 vads_warranty_result=NO vads_payment_src=EC
vads_order_id=111111111 vads_cust_email=françoise.lyonnaise@example.net
vads_cust_name=Françoise+Lyonnaise vads_cust_first_name=Françoise
vads_cust_last_name=Lyonnaise vads_cust_country=FR vads_cust_cell_phone=0612345678
vads_sequence_number=1 vads_contract_used=FR7610107002130032013109105
vads_trans_status=ACCEPTED vads_expiry_month=12 vads_expiry_year=2099 vads_pays_ip=FR
vads_effective_creation_date=20200518140925 vads_identifier=59ecb199110145338c5704505760ec31
vads_identifier_status=CREATED vads_operation_type=VERIFICATION
vads_threeds_enrolled= vads_threeds_auth_type= vads_threeds_cavv= vads_threeds_eci=
vads_threeds_xid= vads_threeds_cavvAlgorithm= vads_threeds_status=
vads_threeds_sign_valid= vads_threeds_error_code=4 vads_threeds_exit_status=4
vads_result=00 vads_extra_result=00 vads_card_country=FR vads_language=fr
vads_hash=ec634c4cebc5e284adae11d895f935f38dd006d80c58a31aa100e22fe8198b6
vads_url_check_src=PAY vads_action_mode=INTERACTIVE vads_payment_config=SINGLE
vads_page_action=REGISTER signature=2c51b0ef96d77488df18f28cf8a6a0f32f3fadac
```

21.3. Analyzing the result of a flat-rate subscription request

1. Analyze the result of mandate registration by testing the value of the **vads_identifier_status** field.

Value	Description
ABANDONED	Mandate registration has been abandoned by the debtor (buyer). The mandate has not been created and cannot be viewed in the Merchant Back Office. No direct debits can be made.
CREATED	The mandate has been successfully registered.
NOT_CREATED	The mandate has not been created and cannot be viewed in the Merchant Back Office. No direct debits can be made.
UPDATED	The mandate has been successfully updated.

2. Retrieve mandate details:

Field	Description
vads_card_number	Returns the IBAN and BIC, separated by a “_”. Since the BIC is optional, it may be left out.
vads_expiry_month	Expiration month of the mandate.
vads_expiry_year	Expiration year of the mandate.
vads_identifier	Unique Mandate Reference (UMR).

3. Analyze the result of recurring payment creation by testing the value of the **vads_recurrence_status** field.

Value	Description
ABANDONED	Creation of the recurring payment has been abandoned by the debtor (buyer). The recurring payment has not been created and cannot be viewed in the Merchant Back Office. No direct debits can be made.
CREATED	The recurring payment has been successfully created. The recurring payment details are visible in the Merchant Back Office.
NOT_CREATED	The recurring payment has not been created and cannot be viewed in the Merchant Back Office. No direct debits can be made.

4. Retrieve the details of the recurring payment:

Field name	Description
vads_sub_amount	Amount of each installment (in cents) of the recurring payment, except the ones that will be eventually defined by the vads_sub_init_amount_number
vads_sub_currency	Numeric code of the EUR currency, valued at 978 .
vads_sub_desc	Recurring payment rule to apply according to the iCalendar RFC5545 specification.
vads_sub_effect_date	Recurring payment start date.
vads_sub_init_amount	Amount of the <u>first installments</u> of the recurring payment.
vads_sub_init_amount_number	Number of installments to which the vads_sub_init_amount amount should be applied.
vads_subscription	Recurring payment ID

Example of notification received:

```
vads_amount =  
vads_auth_mode = MARK  
vads_auth_number =  
vads_auth_result =  
vads_capture_delay = 0  
vads_card_brand = SDD  
vads_card_number = FR7617515900001234567890135_CEPAFRPP751  
vads_payment_certificate =  
vads_ctx_mode = TEST  
vads_currency = 978  
vads_effective_amount =  
vads_site_id = 91335531  
vads_trans_date = 20160925090433  
vads_trans_id =  
vads_validation_mode = 0  
vads_version = V2  
vads_warranty_result = NO  
vads_payment_src = EC  
vads_cust_email = firstname.lastname@mymail.com  
vads_cust_last_name = Last name  
vads_contract_used = FR7630002005701234567890158  
vads_recurrence_status = CREATED  
vads_identifier_status = CREATED  
vads_expiry_month = 9  
vads_expiry_year = 2017  
vads_pays_ip = FR  
vads_identifier = 20160925fRbwlu  
vads_subscription = 20160925FRwRw9  
vads_threeds_enrolled =  
vads_threeds_cavv =  
vads_threeds_eci =  
vads_threeds_xid =  
vads_threeds_cavvAlgorithm =  
vads_threeds_status =  
vads_threeds_sign_valid =  
vads_threeds_error_code = 4  
vads_threeds_exit_status = 4  
vads_result = 00  
vads_extra_result =  
vads_card_country =  
vads_language =FR  
vads_hash = 8c094f21dca9f0279108a87efc0d5bc49337a8613451271e201d1cfcdbbbbeb3  
vads_url_check_src = PAY  
vads_action_mode = INTERACTIVE  
vads_payment_config = SINGLE  
vads_page_action = REGISTER_SUBSCRIBE  
vads_sub_amount = 2990  
vads_sub_currency = 978  
vads_sub_desc = RRULE:FREQ=MONTHLY;BYMONTHDAY= 1  
vads_sub_effect_date = 20161101  
signature = a5026f230479eb17739609cb98c0e70e6516a006
```

21.4. Analyzing the result of a mandate update with payment

1. Analyze the result of the mandate update by testing the value of the **vads_identifier_status** field.

Value	Description
ABANDONED	Mandate registration has been abandoned by the debtor (buyer). The mandate has not been created and cannot be viewed in the Merchant Back Office. No direct debits can be made.
CREATED	The mandate has been successfully registered.
NOT_CREATED	The mandate has not been created and cannot be viewed in the Merchant Back Office. No direct debits can be made.
UPDATED	The mandate has been successfully updated.

2. Retrieve mandate details:

Field	Description
vads_card_number	Returns the IBAN and BIC, separated by a “_”. Since the BIC is optional, it may be left out.
vads_expiry_month	Expiration month of the mandate.
vads_expiry_year	Expiration year of the mandate.
vads_identifier	Unique Mandate Reference (UMR).

3. Retrieve the transaction details:

Field name	Description
vads_acquirer_network	Acquirer network code. Populated with SEPA.
vads_amount	Payment amount (in cents).
vads_currency	Numeric code of the EUR currency, valued at 978 .
vads_occurrence_type	Type of sequence. Populated with RECURRENT_INTERMEDIAIRE .
vads_presentation_date	Returns the requested due date.
vads_trans_id	Transaction identifier transmitted by the Merchant.
vads_trans_status	Transaction status (see table below).
vads_trans_uuid	Transaction identifier generated by the gateway.

The transaction status is returned in the **vads_trans_status** field:

Value	Description
AUTHORISED	Accepted, waiting for capture A pre-notification has been sent. The direct debit is waiting to be presented at the bank.
ABANDONED	Payment abandoned by the buyer The transaction was not created, and is therefore not visible in the Merchant Back Office .
REFUSED	Refused Status sent in case of 1-click payment when some rules have been enabled in the Risk assessment tab or when the mandate is expired.
AUTHORISED_TO_VALIDATE	To be validated

Value	Description
	<p>The direct debit request has been made in “manual validation” mode. The merchant must manually validate the transaction in the Merchant Back Office to authorize the transfer of the mandate to the bank.</p> <p>The pre-notification will only be sent after the transaction has been validated.</p> <p>An automatic e-mail is processed every day, several times per day (every 3 hours) that does not require any intervention on your behalf.</p> <p>The transaction can be validated as long as the expiration date of the authorization request has not passed. If the authorization validity period has passed, the payment status changes to EXPIRED. The Expired status is final.</p>
WAITING_AUTHORISATION	<p>Waiting for authorization</p> <p>The pre-notification has not yet been sent.</p> <p>The direct debit request has been made with a capture delay greater or equal to 7 days for a single payment while the pre-notification delay in the contract is 14 days.</p>
WAITING_AUTHORISATION_TO_VALIDATE	<p>To be validated and authorized</p> <p>The capture delay in the bank exceeds the authorization validity period.</p> <p>The Merchant must manually validate the transaction for the authorization request and the capture to occur.</p>

21.5. Analyzing the result of an installment payment

1. You have requested the creation of a recurring payment with fixed amounts and due dates. You will receive a notification for each due date, identified by the value of the **vads_url_check_src** field:

Value	Description
REC	The REC value is sent only for recurring payments if the merchant has configured the Instant Payment Notification URL when creating recurring payments rule.

2. Identify the mandate and the recurring payment that correspond to the installment by testing the following fields:

Field	Description
vads_identifier	Mandate reference (UMR)
vads_subscription	Recurring payment ID
vads_sequence_number	Sequence number of the recurring payment

3. Retrieve the transaction details:

Field name	Description
vads_acquirer_network	Acquirer network code. Populated with SEPA.
vads_amount	Payment amount (in cents).
vads_currency	Numeric code of the EUR currency, valued at 978 .
vads_occurrence_type	Type of sequence. Populated with RECURRENT_INTERMEDIAIRE .
vads_presentation_date	Returns the requested due date.
vads_trans_id	Transaction identifier transmitted by the Merchant.
vads_trans_status	Transaction status (see table below).
vads_trans_uuid	Transaction identifier generated by the gateway.

The transaction status is returned in the **vads_trans_status** field:

Value	Description
AUTHORISED	Accepted, waiting for capture A pre-notification has been sent. The direct debit is waiting to be presented at the bank.
ABANDONED	Payment abandoned by the buyer The transaction was not created, and is therefore not visible in the Merchant Back Office .
REFUSED	Refused Status sent in case of 1-click payment when some rules have been enabled in the Risk assessment tab or when the mandate is expired.
AUTHORISED_TO_VALIDATE	To be validated The direct debit request has been made in "manual validation" mode. The merchant must manually validate the transaction in the Merchant Back Office to authorize the transfer of the mandate to the bank.

Value	Description
	<p>The pre-notification will only be sent after the transaction has been validated.</p> <p>An automatic e-mail is processed every day, several times per day (every 3 hours) that does not require any intervention on your behalf.</p> <p>The transaction can be validated as long as the expiration date of the authorization request has not passed. If the authorization validity period has passed, the payment status changes to EXPIRED. The Expired status is final.</p>
WAITING_AUTHORISATION	<p>Waiting for authorization</p> <p>The pre-notification has not yet been sent.</p> <p>The direct debit request has been made with a capture delay greater or equal to 7 days for a single payment while the pre-notification delay in the contract is 14 days.</p>
WAITING_AUTHORISATION_TO_VALIDATE	<p>To be validated and authorized</p> <p>The capture delay in the bank exceeds the authorization validity period.</p> <p>The Merchant must manually validate the transaction for the authorization request and the capture to occur.</p>

21.6. Analyzing the result of a one-click payment

1. Retrieve mandate details:

Field	Description
vads_card_number	Returns the IBAN and BIC, separated by a “_”. Since the BIC is optional, it may be left out.
vads_expiry_month	Expiration month of the mandate.
vads_expiry_year	Expiration year of the mandate.
vads_identifier	Unique Mandate Reference (UMR).

2. Retrieve the transaction details:

Field name	Description
vads_acquirer_network	Acquirer network code. Populated with SEPA.
vads_amount	Payment amount (in cents).
vads_currency	Numeric code of the EUR currency, valued at 978 .
vads_occurrence_type	Type of sequence. Populated with RECURRENT_INTERMEDIAIRE .
vads_presentation_date	Returns the requested due date.
vads_trans_id	Transaction identifier transmitted by the Merchant.
vads_trans_status	Transaction status (see table below).
vads_trans_uuid	Transaction identifier generated by the gateway.

The transaction status is returned in the **vads_trans_status** field:

Value	Description
AUTHORISED	Accepted, waiting for capture A pre-notification has been sent. The direct debit is waiting to be presented at the bank.
ABANDONED	Payment abandoned by the buyer The transaction was not created, and is therefore not visible in the Merchant Back Office .
REFUSED	Refused Status sent in case of 1-click payment when some rules have been enabled in the Risk assessment tab or when the mandate is expired.
AUTHORISED_TO_VALIDATE	To be validated The direct debit request has been made in “manual validation” mode. The merchant must manually validate the transaction in the Merchant Back Office to authorize the transfer of the mandate to the bank. The pre-notification will only be sent after the transaction has been validated. An automatic e-mail is processed every day, several times per day (every 3 hours) that does not require any intervention on your behalf. The transaction can be validated as long as the expiration date of the authorization request has not passed. If the authorization validity period

Value	Description
	has passed, the payment status changes to EXPIRED . The Expired status is final.
WAITING_AUTHORISATION	Waiting for authorization The pre-notification has not yet been sent. The direct debit request has been made with a capture delay greater or equal to 7 days for a single payment while the pre-notification delay in the contract is 14 days.
WAITING_AUTHORISATION_TO_VALIDATE	To be validated and authorized The capture delay in the bank exceeds the authorization validity period. The Merchant must manually validate the transaction for the authorization request and the capture to occur.

22. MANAGING SEPA DIRECT DEBIT TRANSACTIONS FROM THE MERCHANT BACK OFFICE

22.1. Identifying unavailable operations

The following operations are not allowed in case of a SDD payment:

- **Duplicate**
- **Modify**

22.2. Identifying authorized operations

The authorized operations are:

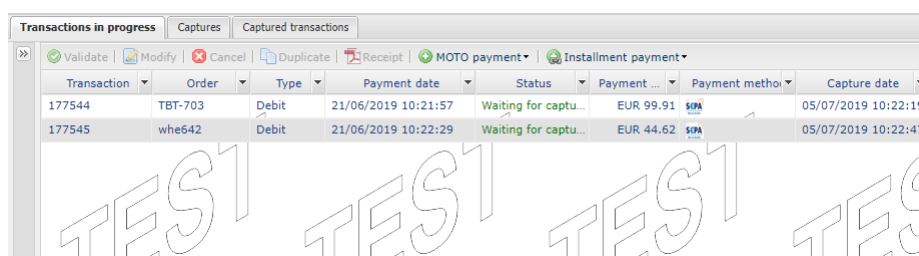
- **Cancel**
- **Validat**
- **Edit the order reference**
Allows to modify the order number.
- **Send the Instant Payment Notification**
Allows to return a notification at the end of payment to the merchant website when the initial call has failed.
- **Resending the transaction confirmation e-mail to the merchant and Resending the transaction confirmation e-mail to the buyer**
Allows to send payment confirmation e-mails to the merchant or the buyer.
- **Refund by wire transfer**
Allows to proceed to refund by wire transfer from the Merchant Back Office.
To do so, the SCT flow must be activated within your EBICS connection via your bank.
- **Terminate**

22.3. Viewing SEPA transactions in the Merchant Back Office

Transactions can be viewed in the Merchant Back Office via the **Management > Transactions** menu.

As long as the due date has not arrived, transactions can be viewed in the **Transactions is progress** tab.

Once the interbank payment presentation has been made, transactions appear in the **Captured transactions** tab.



Transaction	Order	Type	Payment date	Status	Payment ...	Payment method	Capture date
177544	TBT-703	Debit	21/06/2019 10:21:57	Waiting for captu...	EUR 99.91	SDD	05/07/2019 10:22:19
177545	whe642	Debit	21/06/2019 10:22:29	Waiting for captu...	EUR 44.62	SDD	05/07/2019 10:22:47

Double-click a SEPA transaction to see its details.

Details of a transaction in progress: 908610

Details | Buyer | Advanced risks assessment | Event log

Transaction identification

Transaction : 908610

Transaction UUID : b13e8c855fa54310a880d10a2bc77684

Shop : L'Oréal Paris - Paris (391461600)

Current amount : EUR 1.02

Type : Debit

Transaction life cycle

Status : Waiting for capture

Creation date : 20/06/2017 15:52:44

Requested capture date : 04/07/2017 15:52:44

Payment method

Payment method : IBAN

BIC/IBAN : CEPAFRPP75 // FR7617517900001234567890123

Acceptor

Merchant ID (MID) : FR7630003041230000004403264

Source details

Payment source : E-mail payment order

Source user details : s12345

Payment made on : Firefox 53.0. (Windows 7)

Validate | Modify | Cancel | Duplicate | Report | Close

22.4. Viewing mandate details in the Merchant Back Office

The details of the completed and signed mandates are accessible via the Merchant Back Office.

All you need to do is select a **Token** via the **Management > Recurring payments > Token** tab.

You may also receive the mandate by e-mail (right-click on the token > **Receive the mandate by e-mail**).

The mandate is translated only into the languages of the European Union: French, German, English, Spanish, Italian, Dutch, Polish, Portuguese and Swedish.

23. OBTAINING HELP

Looking for help? Check our FAQ on our website

<https://scelliuspaiement.labanquepostale.fr/doc/en-EN/faq/sitemap.html>

For any technical inquiries or if you need any help, contact [technical support](#).

In view of facilitating the processing of your requests, please specify your shop ID (an 8-digit number) in your query.

This information is available in the “registration of your shop” e-mail or in the Merchant Back Office (**Settings > Shop > Configuration**).