



# QPAY Payment Page

## Integration Guide

Version 3.9.0

This integration Guide has been created by Wirecard AG. Its contents may be changed without prior notice. References (links) to external websites are provided for information only. Wirecard does not claim liability for access to and correctness of the referenced content.

## COPYRIGHT

The information contained in this Guide is confidential and/or privileged material and is intended only for the persons or the legal entities to which it is addressed. Any review, retransmission, dissemination or other use of, or taking of any action in reliance upon, this information by persons or entities other than the intended recipient is prohibited. If you received this Guide in error, please contact Wirecard AG and delete the material from any computer.

Copyright © 2009 Wirecard AG  
All rights reserved.

Printed in Germany / European Union  
Version 3.9.0  
Last Updated: October 2009

## TRADEMARKS

The Wirecard logo is a registered trademark of Wirecard AG. Other trademarks and service marks in this document are the sole property of the Wirecard AG or their respective owners.

## CONTACT INFORMATION

For questions relating to this document please contact:

Wirecard Technologies AG  
Bretonischer Ring 4  
D-85630 Grasbrunn

Tel: +43 316 81 36 81  
e-mail: [support@wirecard-cee.com](mailto:support@wirecard-cee.com)

# Contents

<b>1</b>	<b>General</b>	<b>4</b>
1.1	About QPAY	4
1.2	About this Guide	4
1.3	Revision History	4
1.4	Notations	5
<b>2</b>	<b>Integration</b>	<b>6</b>
2.1	Sending Data	6
2.2	Checkout Page and Data Acceptance	6
2.3	Test System and Live System	7
<b>3</b>	<b>Receiving Payments</b>	<b>8</b>
3.1	Editing the Checkout Page	8
3.2	Transferring Parameters	8
3.3	Parameters: Descriptions and Examples	10
3.4	Return Parameters	16
<b>4</b>	<b>Payment Confirmation</b>	<b>19</b>
<b>5</b>	<b>Traceability</b>	<b>20</b>
<b>6</b>	<b>Using the Payment Center</b>	<b>21</b>
<b>7</b>	<b>Frequently Asked Questions</b>	<b>22</b>
7.1	Online Transactions	22
7.2	Debits and Daily Closing Balance	22
7.3	orderDescription	23
7.4	Avoiding Double Payment	24
7.5	Demo Version / Live Version	24
7.6	Activation	24
7.7	MasterCard SecureCode / Verified by VISA	24
7.8	QPAY and Recurring Payments	25
7.9	QPAY and Call Centres	25
7.10	Guaranteed Payments	25
7.11	Foreign Currencies	26
7.12	Support	26

# 1 General

## 1.1 About QPAY

QPAY is a payment page for online shop retailers. It is extremely simple to integrate and easy to handle while maintaining maximum security and scalability.

The payment procedure for consumers is intuitive and easy to follow, which explains the very low ratio of cancelled sales. All payments are subjected to proven risk management checks and are processed without exception online and in real time.

## 1.2 About this Guide

This technical document is first and foremost addressed to Web designers and programmers engaged in programming and configuring online shops, and integrating the Wirecard payment platform in webshops. The user rights for the various product features described are dependent on the terms of service of your licence agreement with Wirecard. The standard version of QPAY supports the euro (€) currency, and the German and English languages.

This document describes how webshops are connected to the Wirecard payment platform using the QPAY. For further information see the inline documentation of our reference implementation which can be downloaded at no charge from the Wirecard Central and Eastern Europe website [www.wirecard-cee.com/download](http://www.wirecard-cee.com/download).

## 1.3 Revision History

These Integration Instructions are updated at regular intervals to provide you promptly with all relevant product improvements and functions. Changes of any kind to the contents will be noted in the version history below and marked in the text with a horizontal bar in the right margin.

Date	Version	Comment/Change
2009-05-04	3.4.0	new Wirecard layout.
2009-06-11	3.5.0	first English version: Data types of parameters incorporated in tables. Detailed description of parameters and values removed from tables and presented in a new Section. Several minor linguistic changes.
2009-07-02	3.6.0	Sections 3.1. and <a href="#">3.2</a> updated. New supported languages (Swedish and Finnish) added.
2009-09-23	3.7.0	New <i>paymenttype</i> ( <a href="#">PAYPAL</a> ) added, new <a href="#">languages</a> (Polish, Chinese, Japanese, Greek) incorporated and new <a href="#">return parameters</a> added for PayPal.
2009-10-08	3.8.0	Description of <a href="#">Notations</a> added. Description of parameter <i>duplicateRequestCheck</i> updated (3 hours changed to <a href="#">30 minutes</a> ). PayPal added to <a href="#">Section 3.4.2</a> .
2009-10-19	3.9.0	New language ( <a href="#">Turkish</a> ) added. Changes made to description of <a href="#">imageURL</a> and <a href="#">confirmURL</a> parameters.

## 1.4 Notations

The following notations define the data type formats of the QPAY parameters.

Notation	Description
a	alphabetic A-Z, a-z
n	numeric digits, 0-9
am	amount
an	alphanumeric
ans	alphanumeric and special characters
ns	numeric and special characters
bool	yes/no
enum	enumeration of possible values
3	Fixed length of 3 bytes
..3	Variable length up to a maximum of 3 bytes.

## 2 Integration

---

QPAY has been designed for easy integration in your online shop without the need for a local software installation. The payment page is called up by a form embedded in the **checkout page** of your online shop. The checkout page is the last page in your online shop located on your server, and is the page where customers can choose their method of payment. But QPAY can be configured such that customers do not have to select their preferred payment method until they reach the the payment page (see [Chap 3](#)). The payment page is residing on our server. It is the page where customers enter their payment data (credit card number, expiry date, etc.).

### 2.1 Sending Data

A few additional parameters must be entered when data is sent (see [Chap 3](#)). You may choose how the QPAY payment page is displayed, either as a separate page, as an inline frame (iFrame) or as a popup.

Remember that there is an increasing trend among Internet users towards popup blockers. Windows XP SP2 also activates a popup blocker. Consequently, we recommend not using a popup to display the payment page.

### 2.2 Checkout Page and Data Acceptance

Proceed as follows: Ensure that all required parameters are provided on the checkout page of your shopping cart.

Also indicate on your checkout page that connected payment platform is securely encrypted. This can be done by displaying a logo we provide in a visible minimum size of 127 x 50 pixels. You will have obtained this logo with the demo version of QPAY. The name of the file is "QENTA\_secured-payments.gif".

Recommendation: Store the order data of your online customers on your system before sending it to the payment page. If payment is successful but the customer is not directed to the success page for whatever reason, you can then still match the payment to its respective order.

To ensure flawless functionality of QPAY, a scripting language like CGI scripts, Perl, PHP, ASP, or an servlet (JAVA) must be employed.



---

**NOTE:** Remember that in accordance with our Terms of Service you have already agreed to implementing our logo in your webshop.

---

In addition to this Guide you will have received sample files for scripting languages ASP, Perl and PHP.

## 2.3 Test System and Live System

You will have received a demo version of QPAY from us, including a reference implementation in the scripting languages ASP, Perl and PHP. The customer number supplied (CUSTOMERID) stands for the demo merchant. QPAY operates in test mode for as long as you use the demo customer number.

The purpose of the test mode is to demonstrate whether you have correctly integrated QPAY with your online shop or not. When you will no longer receive error messages, you can reliably assume that the payment platform will also accept the tested parameters in production mode.

When operating in test mode you will not communicate with the servers of the financial services provider, but with the Wirecard server. This means that you will need an Internet connection for testing your QPAY connectivity.

### 2.3.1 Demo Card Numbers

QPAY comes with demo credit card numbers you can use to simulate online payments by credit card. Like the demo customer number these credit card numbers are available in the file "testdaten.txt". The number are purely fictitious and do not relate to any real credit card brand.

If any of these numbers are used in production mode later on, the error message "Invalid credit card number" will appear.

But you don't need to worry about using genuine credit card numbers, since no real transactions can be made in test mode. Also note that when operating in test mode for MasterCard SecureCode (MCSC) and Verified by Visa (VbV) no checks are made to verify if a credit card is registered for authentication or not.

To be able to switch at a later point from the demo version to the production version, you simply replace the demo customer number with your real customer number and your Demo Secret with your Live Secret.

## 3 Receiving Payments

This Chapter discusses the interface to the payment platform. The interface comes with a complete reference implementation that contains the entire QPAY payment process workflow.

Please also take a look at the inline documentation in the reference implementation.

### 3.1 Editing the Checkout Page

You post all required parameters simply by editing the "checkout.\*" file. Depending on the scripting language you use (ASP, Perl or PHP), the file will be called "checkout.asp", "checkout.cgi" or "checkout.php". Throughout this document the file name extension is replaced by an asterisk '\*'. Please provide the parameters described in Section 3.2 below.

### 3.2 Transferring Parameters

The payment data must be sent with a number of parameters as POST Request to: <https://secure.wirecard-cee.com/qpay/init.php>

#### 3.2.1 Using Parameter Data

Not all of the parameters which are sent as POST are always mandatory or are solely used for transferring payment data to QPAY. Some parameters are also used for generating fingerprints.

On the next page you find a [Table](#) outlining which data is used as parameter and which is used to generate a fingerprint. The column abbreviation "P" denotes that the data is to be sent as parameter while the column "FP" shows which data is used to generate a fingerprint. For further details see descriptions below.

#### Used as Parameter

Use	Description
<b>Mandatory</b>	Parameter <b>must</b> be sent to QPAY
<b>Optional</b>	Parameter <b>can</b> be sent to QPAY
<b>Conditional</b>	Parameter <b>must</b> be sent <b>under certain conditions</b> to QPAY
<b>Never</b>	Parameter should <b>never</b> be sent to QPAY

#### Used to Generate Fingerprint

Use	Description
<b>Mandatory</b>	Parameter <b>must</b> be used
<b>Optional</b>	Parameter <b>can</b> be used
<b>Conditional</b>	Parameter <b>must</b> be used <b>if stated</b>
<b>Never</b>	Parameter should <b>never</b> be used

All parameters marked with an asterisk (\*) should only be used if explicitly activated by Wirecard in the administration setup. Upper and lower case spellings of the variable and their values must be strictly adhered to! It must also be ensured that no special characters are used in order descriptions and the displayed text as this will lead to errors.

### 3.2.2 Parameters at a Glance

Parameter	P	FP	Data Type	Description
customerId	M	M	an 7	Your customer number.
secret	N	M	an..60	Preshared key for signing the transferred parameters.
shopId	O	M	an..16	Shop site identifier required when more than one licence is used for a single customer number.
amount	M	M	am	Payment amount.
amount_net	O	M	am	After tax payment amount.
currency	M	M	a 3 / n 3	ISO currency code
paymenttype	M	O	enum	Indicates the payment method used.
financialInstitution	O	O	enum	When using EPS the supporting bank can be selected in the webshop.
language	M	M	a 2	language abbreviation.
orderDescription	M	M	an..255	description of the order.
displayText	O	O	an	text displayed to the consumer along with the order data.
successURL	M	M	ans	URL connecting to a page indicating a successful purchase.
cancelURL	M	O	ans	URL connecting to a page indicating a cancelled purchase.
failureURL	M	O	ans	URL connecting to a page indicating a failed purchase.
serviceURL	M	O	ans..255	URL connecting to a page providing service information.
confirmURL	O	M	ans	URL for the server-to-server confirmation (not visible).
imageURL	O	O	ans	Link to your logo embedded on the QPAY pages.
requestFingerprintOrder	M	M	ans	List of parameters used for the fingerprint (separated by a comma), order of parameters.
requestFingerprint	M	N	an 32	Digital signature of the posted parameters.
windowName	C	N	an	Name of the window in which QPAY is started.
duplicateRequestCheck	O	C	bool	Activates a double-booking check.
customerStatement	O	C	an..254	Text which is to appear customer's statement.
orderReference	O	C	n..128	Unique transaction ID of the merchant.
autoDeposit *	O	C	bool	Activates automatic settling of a payment.
maxRetries *	O	C	n	Maximum number of payment attempts.
orderNumber *	O	C	n..9	order number assigned to the payment.
other parameters	O	O	ans	

### 3.3 Parameters: Descriptions and Examples

<b>customerid</b>	your customer number
-------------------	----------------------

To be integrated into a webshop, QPAY needs a customer number that uniquely identifies the merchant. In test mode, use the demo customer number provided with the file "testdaten.txt".  
Example: D2xxxxx

<b>secret</b>	presared key for signing the transfered parameters
---------------	--

To be integrated into a webshop, QPAY needs a so called *secret*, a unique alphanumeric string identifying the merchant. In test mode, use the demo secret supplied (in the file "testdaten.txt"). The secret must not be entered as a parameter.

<b>shopid</b>	shop site identifier, required when more than one licence is used for a single customer number
---------------	--

Only to be set if expressly demanded or authorised by Wirecard.

<b>amount</b>	payment amount
---------------	----------------

Decimal point "." and decimal comma "," can be used as separators. No separator characters for thousands are permitted. The maximum number of decimal places is in relation to the posted currency.

#### Examples:

- Euro = maximum of two decimal places (100 Cent = 1 Euro)
- US Dollar = maximum of two decimal places(100 Cent = 1 US Dollar)
- Yen = no decimal places
- Libyan Dinar = maximum of three decimal places (1000 Dirham = 1 Libyan Dinar)

<b>amount_net</b>	after tax payment amount
-------------------	--------------------------

This value is necessary for the MIA payment method. See also parameter "amount"

<b>currency</b>	ISO currency code
-----------------	-------------------

Note that only one currency is activated by default.

#### Examples:

- EUR / 978Euro
- USD / 840US Dollar
- CHF / 756 Swiss Franc
- JPY /392 Yen

<b>paymenttype</b>	Indicates the payment method used to make the transaction.
SELECT	The payment method is defined only in the e QPAY payment page.
CCARD	Payment by credit card, incl. "Verified by Visa" and "MasterCard Secure Code"
CCARD-MOTO	Payment by credit card without "Verified by VISA" or "MasterCard Secure Code" (only allowed with the consent of KKG and Wirecard)
MAESTRO	Payment by Maestro SecureCode
PBX	Payment by paybox
PSC	Payment by paysafecard
EPS	Online wire transfer. Further choices can be made in QPAY.
ELV	Payment by direct debit.
QUICK	Payment by electronic prepaid card @Quick
Mia	Payment through mobile phone invoicing / operator selection in QPAY.
IDL	iDEAL
GIROPAY	giropay
PAYPAL	PayPal

QPAY can be used with payment type CCARD-MOTO in offline mode. However, the buyer should not be able to select this payment method as this would bypass 3D Secure (VbV and MCSC) standards. Note that this feature has to be ordered separately. For details please contact your sales representative.

<b>financialInstitution</b>	preselection of payment type with several options
-----------------------------	---

See also Values for *financialInstitution* in [Section 3.4.2](#).

<b>language</b>	two-character language abbreviation
-----------------	-------------------------------------

The language in which all messages (including interim and error messages) are displayed on the Internet pages. Further languages can be included on request. Please note that only German and English are activated by default. The language is not case sensitive.

cz	Czech
da	Danish
de	German
el	Greek
en	English
es	Spanish
fi	Finnish
fr	French
hu	Hungarian

it	Italian
jp	Japanese
nl	Dutch
pg	Portuguese
pl	Polish
ru	Russian
se	Swedish
sk	Slovakian
sl	Slovenian
tr	Turkish
zh	Chinese

<b>orderDescription</b>	description of the order
-------------------------	--------------------------

This description must clearly refer to the customer's order. The value of this parameter may not include **special characters** or **line breaks**. No **HTML special characters** such as German umlauts, quotation marks (e.g. &auml;) may be used, as these may possibly have been already converted by the browser. In such a case it would be impossible for us to obtain the same starting string as you to determine the correct fingerprint. The result would be an error message such as "Invalid fingerprint".

**This text is reserved for payment centre messaging and will not appear on the customer's invoice.**

<b>displayText</b>	text displayed to the consumer along with the order data
--------------------	--

If this field is missing, the *orderDescription* field is shown. This text does NOT appear on the customer's invoice.

<b>successURL</b>	URL connecting to a page indicating a successful purchase
-------------------	---

This page is launched if the transaction was processed as intended. The consumer should be informed that payment was successful. This is where the posted data is checked for authenticity and the received data is stored.

<b>cancelURL</b>	URL connecting to a page indicating a cancelled purchase
------------------	--

This page is launched if the transaction was cancelled. The consumer should be informed that payment was cancelled.

<b>failureURL</b>	URL connecting to a page indicating a purchase attempt failed
-------------------	---

This page is launched if the transaction failed. The consumer should be informed that payment has been unsuccessful.

<b>serviceURL</b>	URL connecting to a page providing service information
-------------------	--

This page link is embedded behind the webshop logo. When clicked it directs the customer to the webshop's contact information and not to his shopping chart.

<b>confirmURL</b>	URL for the server-to-server confirmation (not visible)
-------------------	---

This page is called upon directly by QPAY before the consumer is directed back to your shop. This URL is not a visible page, but is launched in the background by QPAY.

The same data is posted as with the "successURL", "failureURL" and "cancelURL", but with the one difference that it is treated as a server-to-server request. The confirmURL can be reached only on port 80 (= standard http) or port 433 (= standard https).

Using the *confirmURL* can be very useful as it forwards the transaction data to the shop system in the event of a successful purchase. If for whatever reason the buyer is directed to the *successURL* but the transaction was still processed correctly the payment is reported as successful to your shop system on this *confirmURL*.



**NOTE:** Private addresses are NOT allowed! To find out which IP addresses are private and which are not, go to <http://www.iana.org/>. Do not use any resource provided with .htaccess protection (basic authentication). We will not provide any authentication data when sending confirmation! Make sure that no cookies are sent to the Confirm UR.

<b>imageURL</b>	link to your logo embedded on the QPAY pages.
-----------------	---

The logo with a recommended size of 95 x 65 pixels. If you do not want to show a logo, leave these parameters blank.

Images must have the format "gif", "jpg", "jpeg" or "png". The imageURL cannot contain a query parameter and can be reached only on port 80 (= standard http) or port 433 (= standard https).



**NOTE:** Please remember that cookies cannot be sent to the imageURL. Also, there is no "Get" parameter for the imageURL. Private addresses are NOT allowed! To find out which IP addresses are private and which are not, go to <http://www.iana.org/>. Do not use any resource provided with .htaccess protection (basic authentication). We will not provide any authentication data when copying the graphic!

<b>requestFingerprintOrder</b>	list of parameters used for the fingerprint (separated by a comma), order of parameters
--------------------------------	---

For generating a fingerprint the parameters can be arranged in any order.

**Example of requestfingerprintorder:**

"secret,customerId,amount,currency,language,orderDescription,displayText,successURL,customerField1,requestfingerprintorder".

<b>requestFingerprint</b>	digital signature of the posted parameters
---------------------------	--

This is the digital fingerprint of the data forwarded to QPAY. Used by our server to check whether the parameters sent by you have been modified.

MD5 hash of all the parameters sent with requestFingerprintOrder. Generation of the MD5 value: the parameter values must be added as a string in the specified order. An MD5 value is generated from this string.

Make sure that at least 1 selected parameter of the hash is unique in order to avoid the same hash with different orders. (see duplicateRequestCheck)

This hash value is used on the Init page of QPAY to check the authenticity of the parameters sent.

**Example of a generated fingerprint:**

62682e259f42aa4c2e7aaf727cc58900

<b>windowName</b>	name of the window in which QPAY is started
-------------------	---

The following parameter must be entered if QPAY is to be started inside an iFrame or as a popup.

The following rules apply if you want to assign a name of your own:

- It must not contain any blanks.
- It must consist of letters and numbers only, and the first character must be a letter. Upper and lower cases are permitted but the parameters are case-sensitive.
- It must not contain any German umlauts or special characters.
- The only special character that may be used is the underscore "\_".
- It must not be the same as a reserved word. (see <http://de.selfhtml.org/javascript/sprache/reserviert.htm> )

<b>duplicateRequestCheck</b>	activates a double-booking check
------------------------------	----------------------------------

Prevents unintentional initiation of repeated payments by your customers. It checks whether a specific request - or more accurately, a hash value - has already been used within the last 30 minutes.

Possible values: "yes", or no value. The check is carried out only if "yes" is entered.

By setting this parameter to yes, you will ensure that a buyer cannot send a second payment request to QPAY with the same order number thereby avoiding that the same request is generated twice.

<b>customerStatement</b>	text which is to appear customer's statement
--------------------------	--

Availability of this text on the customer's statement is dependent my vary by financial service provider. The actual data type length may vary by financial service provider.

<b>orderReference</b>	unique transaction ID of the merchant
-----------------------	---------------------------------------

The actual data type length is dependent on the length supported by the financial service provider.

<b>autoDeposit *</b>	activates automatic settling of a payment
----------------------	---

If the value `yes` is sent in the initial payment request, this payment is automatically settled immediately upon successful payment processing.

<b>maxRetries *</b>	maximum number of payment attempts
---------------------	------------------------------------

The following values exist:

- `maxretries < 0` = infinite number of retries (deactivated feature)
- `maxretries = 0` = no retries. Every failed attempt leads to a redirect to the `failureUrl` and a corresponding message (`failureMessage`) in the selected QPAY language.
- `maxretries > 1` = number of retries per buyer. payment attempts which are not counted include: missing parameter (empty input field), incorrect length, incorrect format (e.g. blanks in card number, letters/special characters in numeric fields), configuration errors by Wirecard or financial service provider, errors in communication with Wirecard or financial service provider (transaction server or gateway not visible).

<b>orderNumber *</b>	order number assigned to the payment
----------------------	--------------------------------------

The order number of the payment may be used only if the maximum number of payment attempts is set to zero (0).

Any order number posted to QPAY cannot be used a second time, even if the payment could not be successfully processed.

<b>other parameter</b>	
------------------------	--

Any other parameters (e.g. a session ID) that do not have the same name as one of the QPAY fields are sent unchanged to the `successURL`, `failureURL`, `cancelURL` and `confirmURL`.

Parameters entered in the query strings of the `successURL`, `failureURL`, `cancelURL` and `confirmURL` are sent unchanged.

## 3.4 Return Parameters

The following parameters are sent back by HTTPS POST to the URL of the Internet page to be accessed depending on the result of the payment process, i.e. either the "successURL" (S) and/or the "confirmURL", the "failureURL" (F) or the "cancelURL" (C).

The parameters listed in the table below are returned in the following instances:

- on completion of the payment process
- on cancellation of payment by the consumer
- on failure to process payment due to a fault.

The return parameters marked with an asterisk (\*) may only be used if activated in the administration by Wirecard:

- S = parameter is sent to the success page in the event of a success
- C = parameter is sent to the cancel page in the event of a cancellation
- C = parameter is sent to the error page in the event of a fault

Parameter	Return URL	Data Type	Description
paymentState	S,C,F	enum	Result of the payment attempt "SUCCESS", "CANCEL", "FAILURE"
amount	S	am	Payment amount .
currency	S	a 3/ n 3	Code for the currency in which the transaction was carried out.
paymentType	S	enum	Type of payment in which the authorisation was carried out.  Possible values are " CCARD ", " CCARD-MOTO", "EPS", "ELV", "QUICK", "MIA", " PBX" und "PSC".
financialInstitution	S	enum	See table in <a href="#">Section 3.4.2</a>
language	S	a 2	Language used.
orderNumber	S	n.. 9	Payment number that uniquely identifies the transaction.  It is important to store this order number together along with its associated order data in order to be able to create a unique correlation between the order and payment!
responseFingerprint	S	an 32	The fingerprint used to check the authenticity of the return parameters. This is an MD5 hash covering all parameters.
responseFingerprint Order	S	ans	See also requestFingerprint.
anonymousPan	S	n4	The last four digits of the card number (only for credit card transactions).
authenticated	S	bool	Has the cardholder successfully authenticated himself? (only for credit card transactions).
message	S, F	ans	Error message. It should be logged if possible.

Parameter	Return URL	Data Type	Description
expiry*	S	ns 7	This is the expiration date of the credit card using the format mm/yyyy.
cardholder*	S	a.. 30	The name of the credit card holder. Activated only if so required by processor/acquirer.
maskedPan*	S	ns 19	PCI compliant masked number. Shows several asterisks (*) between the first 6 digits + last 4 digits of the card number.
gatewayReferenceNumber*	S	ans..255	Reference number of the processor/acquirer
gatewayContractNumber*	S	ans..255	Contract number of the processor/acquirer
idealConsumerName*	S	a..255	Name of the account holder returned by the iDEAL issuer.
idealConsumerCity*	S	ans..255	City of the account holder returned by the iDEAL issuer.
idealConsumerAccountNumber*	S	an..255	Number of the account holder returned by the iDEAL issuer.
paypalPayerID *	S	an..13	ID of the consumer returned by Papal.
paypalPayerEmail *	S	ans..127	email address of the consumer returned by Papal.
paypalPayerLastName *	S	an..64	last name of the consumer returned by Papal.
paypalPayerFirstName *	S	an..64	first name of the consumer returned by Papal.
Additional Parameter	S;C;F	ans	Any other parameters (e.g. a session ID, customer data, ...) also sent by you will be returned unchanged by the payment platform to QPAY but not displayed in the Payment Centre.  Bear in mind: If you send parameters which include a dot, QPAY will change the dot into an underscore "_".

\* Parameters marked (\*) should be used only if explicitly activated by Wirecard in the administration setup.

### 3.4.1 Shop Encryption

It is entirely possible that when a customer leaves the QPAY payment page and is directed on to the shop owner's page, the following message appears: "Although this page is encrypted, the information entered by you will be sent over an unencrypted connection and can be easily read by third parties.

Should this information really be sent?" The reason for this is that if a shop is not encrypted, the change from the payment page to the shop page is a change from an encrypted page to an unencrypted page. Since the shop page is where the customer generally enters personal data, it is advisable for it to be encrypted as well,

### 3.4.2 Values for *financialInstitution*

Parameter-Type	Posted Parameter	Returned Parameter	Explanation
CCARD/ CCARD-MOTO	MC	MC	MasterCard
	Visa	Visa	Visa
	Amex	Amex	American Express
	Diners	Diners	Diners
	JCB	JCB	JCB
MAESTRO	MAESTRO	MAESTRO	Maestro SecureCode
PSC	paysafecard	paysafecard	Paysafecard oder cashticket
PBX	PBX	PBX	Paybox
MIA	ONE	ONE	
ELV	ELV	ELV	Elektronisches Lastschriftverfahren
QUICK	QUICK	QUICK	@Quick
BIE	bill-it-easy	bill-it-easy	bill-it-easy
C2P	C2P	C2P	Click2Pay
GIROPAY	GIROPAY	GIROPAY	Giropay
PAYPAL	PAYPAL	PAYPAL	PayPal
EPS	BA-CA	BA-CA	Bank Austria
	Spardat BB	Spardat	Bank Burgenland
	ARZ BAF	ARZ BAF	Bank für Ärzte und Freie Berufe
	ARZ BCS	ARZ BCS	Bankhaus Carl Spängler & Co. AG
	Bawag B	Bawag	BAWAG
	ARZ VB	ARZ VB	Die österreichischen Volksbanken
	Bawag E	Bawag	easybank
	Spardat EBS	Spardat	Erste Bank und Sparkassen
	ARZ GB	ARZ GB	Gärtnerbank
	ARZ HAA	ARZ HAA	Hypo Alpe-Adria-Bank AG
	ARZ HI	ARZ HI	HYPO Investmentbank AG
	Hypo-Racon O	Hypo-Racon	Hypo Oberösterreich
	Hypo-Racon S	Hypo-Racon	Hypo Salzburg
	Hypo-Racon St	Hypo-Racon	Hypo Steiermark
	ARZ HTB	ARZ HTB	Hypo Tirol Bank AG
	ARZ IB	ARZ IB	Immo-Bank
	ARZ IKB	ARZ IKB	Investkredit BANK AG
	ARZ NLH	ARZ NLH	Niederösterreichische Landesbank-Hypothekenbank
	ARZ AB	ARZ AB	Österreichische Apothekerbank
	Bawag P	Bawag	PSK Bank
	Racon	ELBA-Racon	Raiffeisen
	Bawag S	Bawag	Sparda Bank
	ARZ VLH	ARZ VLH	Vorarlberger Landes- und Hypothekenbank AG
IDL	ABNAMROBANK	ABNAMROBANK	ABN AMRO Bank
	FORTISBANK	FORTISBANK	Fortis Bank
	POSTBANK	POSTBANK	Postbank
	RABOBANK	RABOBANK	Rabobank
	SNSBANK	SNSBANK	SNS Bank

## 4 Payment Confirmation

---

QPAY lets you verify returned parameters for authenticity. This is important, since such a check effectively prevents misuse of the payment data and means that a payment cannot be simulated.

To use the QPAY hash check in test mode, proceed as follows:

- ❑ Copy the demo customer number supplied with the file "testdaten.txt" and enter it into the "checkout.\*" file between the two inverted commas as the value of the parameter CUSTOMERID.
- ❑ Copy the demo secret supplied with the file "testdaten.txt" and enter it into the "checkout.\*" file between the two inverted commas as the value of the parameter \$secret.
- ❑ Copy the demo secret supplied with the file "testdaten.txt" and enter it into the "return.\*" file between the two inverted commas as the value of the parameter \$secret.

Using this secret, known only to you and us, a digital fingerprint of the transmitted data is generated which can be used to test the sent data for authenticity. This means it is no longer possible to access your "successURL" directly to simulate a successful payment.

To allow your online application to process or store the order data returned by QPAY you must use a script language like CGI, Perl, PHP, ASP or a Servlet (JAVA).



---

**NOTE:** Please bear in mind that the *secret* consists of critical (sensitive) data. Consequently, never send the *secret* to us or any of your staff by e-mail. In doing so you will compromise the hash check security feature and with the result that we can no longer ensure that the secret will solely be known to you and us.

---

## 5 Traceability

---

To identify each transaction unequivocally and allocate the corresponding user session of your customer, you must be able to match the returned parameters and most importantly the order number to the appropriate user session (e.g. order data) of your customer.

Remember that you hand over control to us when your online customer is passed on to our payment page. If connection problems occur during or after the payment procedure, the data of the user session are lost. Even in the case of successful payment, it would no longer simply be possible to establish whether this payment exists in the Wirecard system for this user session, or the user session to which the payment belongs.

Depending on the online application, the user session can relate to a purchase in your online shop (shopping cart data), registration of software, or a booking in a ticket shop, etc.

We therefore recommend the following course of action:

1. Include a unique number for the user session in question (e.g. session ID, order number, debit number), depending on the nature of your online application, in the order description of the payment.
2. Store the relevant user session data of the online customer (i.e. the content of the session) at your facility before it is passed on to our payment page. Send at least one unique number for the user session, which will be returned unchanged by QPAY. You can also send any other additional parameters (other customer data or the entire session). They will be returned unchanged.
3. If the transaction is completed successfully, QPAY will return an order number to you (orderNumber parameter) with all the additional parameters you have sent.
4. Store the parameters returned by QPAY (essentially the order number) in your successURL together with the user session data. We will return the user session in question to you if you have sent it to us. You can record the values in a file or database, or send them by e-mail.

This means that an unequivocal link between your customers user sessions and their payments will always exist at all times. If the transfer of the customer to your successURL fails, the content of the user session is not lost, as it was stored in step 2.

A transfer can fail, for example, if the user closes the browser window or clicks on the update button, or if sudden connection problems arise. In such cases the order data would be lost if they were not previously stored, because the *successURL* was never reached.

To ensure that your online application processes or stores the order number returned by QPAY, you must use a scripting language (CGI Script, Perl, PHP, ASP or the like), or a servlet (JAVA).



---

**NOTE:** If you want to generate the link between user session and payment in the payment centre automatically or check the status of the online payment process, you can do this with the payment centre toolkit, controlled by the script. The toolkit<sup>1</sup> also allows you to copy all payments into your external accounting software and integrate some or all the functions of the payment centre into your accounting software.

---

## 6 Using the Payment Center

---

As part of activation, you have received your genuine access data to the payment centre that allow you to monitor transactions at any time. In the payment centre, authorisations can be debited and items cleared, among other things.

Further information about the payment centre can be found in "Payment Centre" under the menu option "Help". You can familiarize yourself with the payment centre before activation by logging on with the test access data, which you will find in the "testdaten.txt" file.

It is important to note that in the first step, the payment platform only authorises the credit card orders from your customers. This authorisation lapses at the credit card organisation within one week (MasterCard) or fourteen calendar days (all other credit card companies). Therefore, if you have not set your payment module to automatic debiting (which can be done at any time), you must debit these authorisations in the payment centre at regular intervals and prepare daily closing balances (at least once a week) so that you can receive the transfers. Wirecard does not accept any liability if you do not carry out these steps regularly and correctly.

Additional information on how to proceed with other methods of payment (paybox, paysafecard, etc.) can be found in the operating instruction for the payment centre.



---

**NOTE:** Before shipping the ordered goods ensure that a matching payment exists in the payment centre!

---

---

1. The use of the Payment Center toolkit comes at an additional charge. Please contact your sales representative for details

# 7 Frequently Asked Questions

---

## 7.1 Online Transactions

### **F: What happens exactly when an online payment is made by credit card?**

**A:** Once the customer has entered his credit card number and the expiry date, the server of the financial services provider is contacted and asked if the payment will be made, i.e. whether it is covered, among other things.

In the event of an authorisation (= result of a positive check), a payment (authorisation) is generated for the corresponding amount. The available limit of the cardholder is reduced by the amount in question, although initially his/her account is not debited. This limit can be fixed (a specific amount) or set individually. The card account is not charged until the payment is debited and cleared.

Until the contract company debits and clears this authorisation (see the next question), the cardholder is not charged. Once the authorisation has been debited and also cleared, the real money cash flow is initiated, and after a pause (which can be found in the relevant contract with the FDL), the amount is transferred to the merchant's account. Only now is the cardholder's credit account debited, and the cardholder finds a debit line on his next statement with the trading name of the online merchant setting out the appropriate amount.

## 7.2 Debits and Daily Closing Balance

### **F: How can payments be debited and deposited so that I can collect the money?**

**A:** The payments can either be debited and cleared manually in the payment centre, or alternatively the payment module can be set to fully automatic clearing. Manual debiting and clearing is then no longer possible. Another possibility is to use the payment centre tool kit. The tool kit is the complete function library of the payment centre, with which the manual user operations in the payment centre can be automated under the control of a script.

See the directions for use of the payment centre for further information on debiting and daily closing balances. You can download the latest version of this document from the payment centre at any time under Home ? Help ? Directions for Use.

You can also log in to the payment centre with the demo access data from the "testdaten.txt" file.

## 7.3 orderDescription

### F: What is the purpose of the orderDescription parameter?

A: The purposes for which you want to use the orderDescription parameter are left up to you. Based on many years of experience with our customers, we advise including the name and possibly the address of the consumer, or an invoice or order number in the merchandise control system enabling the order and therefore the cardholder to be unequivocally identified, in the order description.

Remember that the order description is only displayed in the payment centre.:

9 Genehmigungen gefunden, davon werden 9 angezeigt.

Zahlungsnummer	Betrag	gebucht	Währung	Erstellzeit	Code	Auftragsnummer	Auftragsbeschreibung	Marke
<input type="checkbox"/> 7678983	1.00	0.00	EUR	31.01.2004 08:00	123456	7678983	Max Mustermann, K-Nr: 12345	
<input type="checkbox"/> 4784721	1.00	0.00	EUR	30.01.2004 08:00	123456	4784721	Daniela E. Kommerz, K-Nr: 60245	
<input type="checkbox"/> 321	1.00	0.00	EUR	29.01.2004 08:00	123456	321	Lisa Kaufrausch, K-Nr: 54435	

The order description does not appear on the consumer's bank statement. Its purpose is to give a better overview of the transactions and orders.

Remember also that an order description has a maximum length of 255 characters. In the standard payment view (see screenshot below), only up to 100 characters of the order description can be displayed:

Zahlungsnummer	Betrag	gebucht	Währung	Erstellzeit	Code	Auftragsnummer	Auftragsbeschreibung	Mar
<input type="checkbox"/> 4398333	1.00	0.00	EUR	15.06.2004 09:46	055050	4398333	AAAAAAAA AAAAAA AAAAAA AAAAAA BBBBBB BBBBBB BBBB BB CCCCCC CCCCC CCCC CCCC DDDD DDDD DDD DDD DD E	

The complete order description is only shown in the detailed view of the payment:

Home
Genehmigungen
Abbuchungen
Gutschriften
Tagesabschluss
Suchen
Hilfe

### Genehmigungsinformationen/Zahlung stornieren

**Auftragsdaten**

**Auftragsnummer** 4398333

**Auftragsbeschreibung**  
 AAAAAAAAA AAAAAA AAAAAA AAAAAA BBBBBB BBBBBB  
 BBBB BB CCCCC CCCCC CCCC CCCC DDDD DDDD DDD  
 DDDD DD EEEE EEE EEE EEEE EEE FFF FFFF FFFF GGG GGG  
 GGG GGGG GG HHHH HHH HHHH HHH IIII IIII IIII II IIII  
 KKKK LL M NN OOOOOOOO PPPPPPPP QQQQQQQQQQ  
 RRRRRR SS

**Auftragsbetrag** 1.00

**Währung** EUR

**Zahlungsart**

**Erstellzeit** 15.06.2004 09:46:15

**Letzte Änderung** 15.06.2004 09:46:15

**Zahlungsdaten**

**Zahlungsnummer** 4398333

**Code** 055050

**Genehmigter Betrag** 1.00

**Gebuchter Betrag** 0.00

**Erstellzeit** 15.06.2004 09:46:15

**Letzte Änderung** 15.06.2004 09:46:15

**Tagesabschluss**

Genehmigung stornieren

Hier können die Zahlungen, die vom Kunden gemacht worden sind, storniert werden und verschwinden aus dem Payment Center.

Dies funktioniert bei MasterCard Zahlungen (nicht bei allen mögliche), American Express und Diners Club Zahlungen. Bei paybox Zahlungen ist dies auch möglich, aber NUR wenn Sie einen paybox PREMIUM Vertrag abgeschlossen haben.

Bei VISA Zahlungen ist das Stornieren der Zahlungen NICHT möglich. Dies ist aber kein Problem. Die Zahlungen fallen nach ca. 2 bis 3 Wochen aus dieser Ansicht heraus, wenn Sie NICHT abgebucht wurden.

If you use umlauts in the order description, these will simply be ignored. However, no error will be caused.

## 7.4 Avoiding Double Payment

**F: Sometimes a merchant discovers more than one payment for an order in the payment centre. How does this happen, and how can it be avoided?**

**A:** If a customer presses the Back button of his browser, he may under some circumstances unwittingly/inadvertently trigger a new payment.

You can easily prevent this by giving the parameter `DUPLICATEREQUESTCHECK` the value "yes". This has the following effect: For each transaction, QPAY checks whether the generated hash value has been used within the last three hours. If this is the case, the transaction is not carried out, as one already exists.

## 7.5 Demo Version / Live Version

**F: Once I've installed the demo version of QPAY: is there any way I can switch it to the production version, or do I have to install a separate production version of QPAY?**

**A:** You can switch your installed QPAY to the production version simply by replacing the demo customer number by your genuine customer number, and the demo secret by the live secret. All you have to do is enter your customer number as the parameter value in the `CUSTOMERID` parameter in the "checkout.\*" file. We will send you your customer number once the contract is concluded.

## 7.6 Activation

**F: What happens in activation?**

**A:** When we activate a webshop, we activate the account of the online merchant and the licenced payment modules, and send you (our contract partner, in other words the merchant) your customer number by e-mail and your live secret by fax.

You will also be faxed you access data (Internet address for login, user name, password) for the payment centre. This administrative program allows transactions to be debited, cleared, etc., but also cancelled and searched.

## 7.7 MasterCard SecureCode / Verified by VISA

**F: Where can I look at an MSCS or VbV transaction?**

**A:** You can watch demos of authenticated payments on the MasterCard and Visa website.

**MasterCard:**

[http://www.mastercard.com/us/personal/en/cardholderservices/securecode/how\\_it\\_works.html](http://www.mastercard.com/us/personal/en/cardholderservices/securecode/how_it_works.html)

Here you will find a summary of how a MasterCard SecureCode payment operates.

**VISA:**

<http://www.visaeurope.com/merchant/handlingvisapayments/cardnotpresent/verifiedbyvisa.jsp>

Click Visa Shopping Demo. A demo program will start, where you can follow the process of a VbV transaction.

## 7.8 QPAY and Recurring Payments

**F: I want to make recurrent payments without knowing the credit card numbers or expiry dates. How do I use QPAY for this?**

**A:** This can be done with the payment centre tool kit and the tool kit add-on (recurrent payments add-on). QPAY itself is less suitable as an application for pure webshop payments. With the tool kit and add-on, recurrent payments can be made without you having to know the credit card number of the cardholder.

## 7.9 QPAY and Call Centres

**F: I want to use QPAY as a call-centre application. The staff take orders over the phone and want to check in real time whether the accounts are covered. How do I do this?**

**A:** QPAY can be configured so that that call centre/back office transactions can be effected (see paymenttype CCARD-MOTO). Contact our sales department to have this option activated.

## 7.10 Guaranteed Payments

**F: I want to use QPAY so that only guaranteed payments can be made over it. Once the customer has bought something, I can keep the money whatever happens.**

**A:** Conventional credit card payments do not include any guarantee of payment. That means that the financial service provider credits the amount in question back to the cardholder's account if the latter subsequently disputes payment.

Guaranteed payments are possible with MasterCard and Visa, however. Even if the cardholder subsequently disputes payment, the money remains with the merchant whatever happens. This means that all MasterCard SecureCode payments are guaranteed. You should ask you acquirer about the conditions under which a Verified-by-Visa payment is guaranteed. The other credit card companies also offer payment guarantees under specific contractual provisions in individual cases.

If you only want to permit guaranteed payments in your webshop, contact us. We will amend your account accordingly.



---

**NOTE:** Not all cardholders have enrolled their cards for MasterCard SecureCode or Verified by VISA. The best thing is to draw the attention of these cardholders to the fact that they must enrol if they want to shop with you. You may possibly lose a lot of sales because some cardholders will not want to register their cards for MasterCard SecureCode or Verified.

---

## 7.11 Foreign Currencies

**F: Does QPAY support currencies other than the Euro and the US dollar?**

A: QPAY supports all currencies in accordance with the ISO 4217 standard:

<http://www.iso.org/iso/en/prods-services/popstds/currencycodeslist.html>

As of March 2007. Currencies introduced since March 2007 are currently not supported by QPAY. Contact our Support if necessary.

Please note that the financial service provided in question must also support transactions in the currency you desire. Please ask the appropriate financial service provider if necessary.

## 7.12 Support

**F: I still have questions about QPAY. Where can I get support?**

A: We will be happy to answer your questions at any time by e-mail and by phone. Please contact Wirecard Customer Support.

**E-Mail:** [support@wirecard-cee.com](mailto:support@wirecard-cee.com)

Phone: +43 316 81 36 81

Please remember when sending queries by e-mail to put your customer number (starting with D2) in the "Subject" line, and then your text for the subject, describing the nature of the problem.

**Example:**

D2XXXXX Questions re. QPAY

Please replace "XXXXX" with the last five (= right-hand) digits of your customer number. You can find your customer number on your invoices or welcoming mail, for example.

When enquiring by phone, please also have your customer number and customer code word ready.

[www.wirecard.com](http://www.wirecard.com)

© 2009 Wirecard AG  
All rights reserved.