

Wirecard CEE Integration Documentation



Created: 2019-09-18 15:54

Core Feature Comparison in Detail

JAPI/SAPI

Java API/Serial API are the Java software library and the interface for the QTill Server. This table gives a basic feature overview. For more detailed information such as parameter definition, contact support.

Feature	Description
Initiate Payment	Mandatory parameters: <ul style="list-style-type: none"> • inittype (to initialize a specific payment method) • amount • currency • orderNumber • orderDescription 3-D Secure: <ul style="list-style-type: none"> • Specific inittype (3DSECURE) which implicitly checks for 3-D Secure enrollment. • If card is enrolled, merchant needs to send verifyPayment with additional 3-D Secure data received from ACS redirect.
Verify Payment	Follow up of initiatePayment. Checks payment status of a single payment. Mandatory parameter: <ul style="list-style-type: none"> • orderNumber (returned on initiatePayment) Result: <ul style="list-style-type: none"> • Success returns payment details • Error
Deposit	Transfer of authorization into deposit (capture). Mandatory parameters: <ul style="list-style-type: none"> • orderNumber (returned on initiatePayment) • paymentNumber (= orderNumber) • amount (amount to be deposited)
Recurring Payment	Creates new order and payment based on already existing order. Mandatory parameters: <ul style="list-style-type: none"> • orderNumber (new number from createOrder) • sourceOrderNumber (parent transaction) • amount • currency • orderDescription
Refund	Creates refund for already closed deposit. Only possible if deposit for and closing of order are already completed. Mandatory parameters: <ul style="list-style-type: none"> • orderNumber (orderNumber of order which is to be refunded) • amount

Query	<p>queryOrders: Returns information about all orders and their payment and refund status for given input parameters.</p> <p>Mandatory parameter:</p> <ul style="list-style-type: none"> • “For” to specify which orders to return (depending on status). <p>queryCredits: Returns all refunds.</p> <p>queryBatches: Returns information about daily batches - either closed or open (depending on request parameters).</p>
Approve Reversal	Reverses an authorization for a payment which has not yet been deposited.
Deposit Reversal	Reverses a deposit if daily closing has not yet been done.

Toolkit

[Click here for comprehensive Toolkit documentation.](#)

Feature	Description
Initiate Payment	Triggered by <ul style="list-style-type: none"> • WCS or WCP • JAPI/Serial API direct call
Verify Payment	queryOrders with orderNumber as primary key
Deposit	Transfer of authorization into deposit (capture). Mandatory parameters: <ul style="list-style-type: none"> • orderNumber (returned on initiatePayment) • paymentNumber (usually equal to orderNumber) • amount (amount to be deposited)
Recurring Payment	Available as add-on. Creates new order and payment based on already existing order. Mandatory parameters: <ul style="list-style-type: none"> • orderNumber (new number from createOrder) • sourceOrderNumber (parent transaction) • amount • currency • orderDescription
Refund	Creates refund for already closed deposit. Only possible if deposit and closing for order are already completed. Mandatory parameters: <ul style="list-style-type: none"> • orderNumber of order which is to be refunded • amount

Query	<p>queryOrders: Returns information about all orders and their payment and refund status for given input parameters.</p> <p>Mandatory parameter:</p> <ul style="list-style-type: none"> • “For” to specify which orders to return (depending on status). <p>queryCredits: Returns all refunds.</p> <p>queryBatches: Returns information about daily batches - either closed or open (depending on request parameters).</p>
Approve Reversal	Reverses an authorization for a payment which has not yet been deposited.
Deposit Reversal	Reverses a deposit if daily closing has not yet been done.

Toolkit light

Toolkit light documentation:

- Back-end operations
- Wirecard Checkout Page back-end example on GitHub
- Wirecard Checkout Seamless back-end example on GitHub

Feature	Description
Initiate Payment	Triggered in front-end by WCS or WCP
Verify Payment	Return all details for a given order: <ul style="list-style-type: none"> • WCP command: getOrderDetails • WCS endpoint: /seamless/backend/getOrderDetails
Deposit	Approval converted into capture: <ul style="list-style-type: none"> • WCP command: deposit • WCS endpoint: /seamless/backend/deposit
Recurring payment	Create payment by reusing information from previous order: <ul style="list-style-type: none"> • WCP command: recurPayment • WCS endpoint: /seamless/backend/recurPayment
Refund	Refund already closed order: <ul style="list-style-type: none"> • WCP command: refund • WCS endpoint: /seamless/backend/refund
Query	Not possible to query multiple orders; possible to get single order details by order number: <ul style="list-style-type: none"> • WCP command: getOrderDetails • WCS endpoint: /seamless/backend/getOrderDetails
Approve Reversal	Cancel of approval: <ul style="list-style-type: none"> • WCP command: approveReversal • WCS endpoint: /seamless/backend/approveReversal
Deposit Reversal	Cancel of deposit: <ul style="list-style-type: none"> • WCP command: depositReversal • WCS endpoint: /seamless/backend/depositReversal

Wirecard PHP Payment SDK

Detailed documentation, examples and download: <https://github.com/wirecard/paymentSDK-php/>

Feature	Description
Initiate Payment	Operational functions: <ul style="list-style-type: none"> • credit card UI for input fields generated via seamless Payment Page • <code>pay()</code>: initiates payment process or captures authorization • <code>reserve()</code>: authorization request 3-D Secure: <ul style="list-style-type: none"> • credentials for 3-D Secure added in configuration • limits added in configuration • <code>reserve()</code> and <code>pay()</code> functions check enrollment and redirect to 3-D Secure authentication if 3-D Secure is possible
Verify Payment	<ul style="list-style-type: none"> • Notification handling server-to-server request regarding changes in status • Server notification received after payment has been completed successfully
Deposit	<code>pay()</code> function: <ul style="list-style-type: none"> • initiates payment process or • captures an existing authorization (<code>reserve()</code>)
Recurring payment	A recurring payment is based on a previous transaction, which returns <code>tokenID</code> as response if the transaction was successful. It is performed with the function <code>pay()</code> , using the additional parameter <code>tokenID</code> . Recurring payments are available for: <ul style="list-style-type: none"> • Credit Card • Maestro SecureCode • SEPA Direct Debit
Refund	There are two different refund functions: <ul style="list-style-type: none"> • <code>cancel()</code>: requests to void a transaction. If voiding is no longer possible, a refund is initiated. • <code>credit()</code>: performs a <code>pay()</code> in the opposite direction (pays a specific amount to the consumer).
Query	You can query transactions by: <ul style="list-style-type: none"> • <code>transaction-id</code>: returns single transaction • <code>parent-transaction-id</code>: returns all corresponding transactions See sample source: <code>transaction.php</code>
Approve Reversal	<code>cancel()</code> : If a payment has been authorized but not yet captured, <code>cancel()</code> leads to a void of authorization.
Deposit Reversal	<code>cancel()</code> : If a credit card payment has already been captured, <code>cancel()</code> leads to a refund.

REST API

The REST API architecture differs completely from the JAPI/SAPI architecture. There are multiple options to call a certain functionality. The transaction type used always depends on the payment method to be implemented. This table provides an overview. For more information see the comprehensive documentation:

- REST API

• Transaction types

Feature	Description
Initiate Payment	Transaction types: <ul style="list-style-type: none"> • preauthorization • authorization • authorization-only • final-authorization • authorization-supplementary • get-url • debit • purchase 3-D Secure: <ul style="list-style-type: none"> • Check enrollment • Customer ACS Communication • Check Payers Response • Payment Request
Verify Payment	General API request to get status of asynchronous payment (example).
Deposit	Transaction types: <ul style="list-style-type: none"> • capture-authorization (takes funds from the card holder after authorization) • capture-preauthorization • capture-final-auth • deposit
Recurring Payment	Transaction types: <ul style="list-style-type: none"> • debit • authorization • referenced-purchase • referenced-authorization Additional elements in XML request: <ul style="list-style-type: none"> • periodic • periodic-type • sequence-type Recurring based on <ul style="list-style-type: none"> • merchant-tokenization-flag and tokenID • parent-transaction-id
Refund	Transaction types: <ul style="list-style-type: none"> • refund-capture (moves funds back to cardholder according to a corresponding capture) • refund-purchase (returns funds based on purchase) • refund-request & refund-debit • chargebacks: first-cb, second-cb
Query	Single queries are possible via REST API based on: <ul style="list-style-type: none"> • transaction-id • request-id • request-idandcategory • group-transaction-id
Approve Reversal	Transaction types: <ul style="list-style-type: none"> • void-authorization • void-preauthorization • void-final-authorization • void-auth-supplementary • void-debit • reversal-authorization

Deposit Reversal	Transaction types: <ul style="list-style-type: none">• void-capture• void-debit• void-purchase• reversal-purchase• void-capture-preauthorization• void-capture-final-auth
-------------------------	--

[Back to Switching to Wirecard Payment Gateway main page.](#)