Braspag  
Custom Payment Gateway

Oracle Cloud Marketplace Listing **Test Plan Questionnaire**

Complete this documentation to describe the scope of the integration testing performed with the Oracle Cloud.

|  |  |  |  |
| --- | --- | --- | --- |
| Version | Date | Author | Comments |
| 1.0 | 08 November 2019 | Rogerio Santos Almeida | Initial Version |
|  |  |  |  |

### 

# Test Scope and Use Cases

Outline the scope of testing covered by the test plan and list the primary test uses cases.

|  |
| --- |
| The main tests are: Payments without fraud analysis and payments with fraud analysis |

# Test Environment

### Environments used for the testing

List the Cloud Services and the URLs where the solution has been deployed for testing. In some cases, Oracle may request access to the environments for additional verification

|  |
| --- |
| Test environment: <https://ccadmin-z9ta.oracleoutsourcing.com/> |

### Initial data set up requirements

Describe any test-specific data that is required to perform the testing

|  |
| --- |
| The data to make tests are:  **For Credit Card Payment - With Anti fraud:**   * CPF, date of birth, shipping address, billing address, credit card info (cvv, brand, number, holder, expireMonth, expireYear), gateway settings * Valid credit card number * Last name in the shipping address form should be one of this values: Accept, Review or Reject   **For Credit Card Payment - Without Anti fraud:**   * CPF, date of birth, shipping address, billing address, credit card info (cvv, brand, number, holder, expireMonth, expireYear), gateway settings * Valid credit card number   **For Tokenized Card Payment - Without Anti fraud:**   * CPF, date of birth, shipping address, billing address, credit card info (cvv, brand), gateway settings * Valid card token   **For Cash Payment - Without Anti fraud:**   * CPF, date of birth, shipping address, billing address, gateway settings |

# Test Plan

### Test Use Cases

Describe each process flow being tested and expected result – a live demonstration to Oracle may be required to confirm whether the flow passed or failed. For each process flow list any additional set up, the detailed steps, expected results and the test result/status that can be verified by Oracle.

|  |  |
| --- | --- |
| **Payment without anti fraud - Credit Card - Success** | |
| Pre-requisites & Test Setup: | * Environment full configured. (Gateway Settings, Checkout widget and custom properties configured in checkout widget js) |
| Process Flow: | 1. The shopper fill form fields on checkout page 2. The shopper submit data and confirm order 3. Credit card webhook sends the request to custom gateway 4. Custom gateway just authorize or authorize and capture the payment and responses credit card webhook 5. Order submit webhook sends the order details to custom gateway and this update OCC or wait the capture or refund payment by admin panel 6. If the merchant don’t use automatic capture, the custom gateway waits the payment capture or payment refund by admin panel. When this will be done, custom payment gateway receive an notification and will be update OCC |
| Expected result: | Order Status: SUBMITTED  Payment Status: SETTLED |
| Test result: | Order Status: SUBMITTED  Payment Status: SETTLED |

|  |  |
| --- | --- |
| **Payment without anti fraud - Credit Card - Fail** | |
| Pre-requisites & Test Setup: | * Environment full configured. (Gateway Settings, Checkout widget and custom properties configured in checkout widget js) |
| Process Flow: | 1. The shopper fill form fields on checkout page 2. The shopper submit data and confirm order 3. Credit card webhook sends the request to custom gateway 4. Custom gateway just authorize or authorize and capture the payment and responses credit card webhook 5. Order submit webhook sends the order details to custom gateway and this update OCC or wait the capture or refund payment by admin panel 6. If the merchant don’t use automatic capture, the custom gateway waits the payment capture or payment refund by admin panel. When this will be done, custom payment gateway receive an notification and will be update OCC |
| Expected result: | Order Status: FAILED  Payment Status: SETTLE\_FAILED |
| Test result: | Order Status: FAILED  Payment Status: SETTLE\_FAILED |

|  |  |
| --- | --- |
| **Payment without anti fraud - Cash - Success** | |
| Pre-requisites & Test Setup: | * Environment full configured. (Gateway Settings, Checkout widget and custom properties configured in checkout widget js) |
| Process Flow: | 1. The shopper fill form fields on checkout page 2. The shopper submit data and confirm order 3. Generic webhook sends the request to custom gateway 4. Custom gateway just authorize the payment and responses generic webhook 5. The payment provider waits the bank confirm payment and sends a notification to custom gateway when it receive a confirmation. 6. The custom gateway receives the notification and capture or refund the payment and update OCC |
| Expected result: | Order Status: SUBMITTED  Payment Status: SETTLED |
| Test result: | Order Status: SUBMITTED  Payment Status: SETTLED |

|  |  |
| --- | --- |
| **Payment without anti fraud - Cash - Fail** | |
| Pre-requisites & Test Setup: | * Environment full configured. (Gateway Settings, Checkout widget and custom properties configured in checkout widget js) |
| Process Flow: | 1. The shopper fill form fields on checkout page 2. The shopper submit data and confirm order 3. Generic webhook sends the request to custom gateway 4. Custom gateway just authorize the payment and responses generic webhook 5. The payment provider waits the bank confirm payment and sends a notification to custom gateway when it receive a confirmation. 6. The custom gateway receives the notification and capture or refund the payment and update OCC |
| Expected result: | Order Status: FAILED  Payment Status: SETTLE\_FAILED |
| Test result: | Order Status: FAILED  Payment Status: SETTLE\_FAILED |

|  |  |
| --- | --- |
| **Payment with anti fraud - Credit Card - Success** | |
| Pre-requisites & Test Setup: | * Environment full configured. (Gateway Settings, Checkout widget and custom properties configured in checkout widget js) |
| Process Flow: | 1. The shopper fill form fields on checkout page 2. The shopper submit data and confirm order 3. Credit card webhook sends the request to custom gateway 4. Custom gateway just pre authorize the payment in this first step and responses credit card webhook 5. Order submit webhook sends the order details to custom gateway and the gateway authorize and analyse the payment and response to order submit webhook. 6. After process the order, custom gateway waits a notification from antifraud provider to decide if will capture or refund the payment |
| Expected result: | Order Status: SUBMITTED  Payment Status: SETTLED |
| Test result: | Order Status:SUBMITTED  Payment Status: SETTLED |

|  |  |
| --- | --- |
| **Payment with anti fraud - Credit Card - Fail** | |
| Pre-requisites & Test Setup: | * Environment full configured. (Gateway Settings, Checkout widget and custom properties configured in checkout widget js) |
| Process Flow: | 1. The shopper fill form fields on checkout page 2. The shopper submit data and confirm order 3. Credit card webhook sends the request to custom gateway 4. Custom gateway just pre authorize the payment in this first step and responses credit card webhook 5. Order submit webhook sends the order details to custom gateway and the gateway authorize and analyse the payment and response to order submit webhook. 6. After process the order, custom gateway waits a notification from antifraud provider to decide if will capture or refund the payment |
| Expected result: | Order Status: FAILED  Payment Status:   * AUTHORIZE\_FAILED: Used when an error has occurred while the payment authorization has processed or fraud analysis returns reject status * SETTLE\_FAILED: Used for canceled or refunded transactions after has authorized or captured * CREDIT\_FAILED: Used for transactions with a status diffent of authorized, captured, refunded,canceled and transactions rejected by anti fraud |
| Test result: | Order Status: FAILED  Payment Status:   * AUTHORIZE\_FAILED: Used when an error has occurred while the payment authorization has processed or fraud analysis returns reject status * SETTLE\_FAILED: Used for canceled or refunded transactions after has authorized or captured * CREDIT\_FAILED: Used for transactions with a status diffent of authorized, captured, refunded,canceled and transactions rejected by anti fraud |

|  |  |
| --- | --- |
| **Payment without anti fraud - Card Tokenized - Success** | |
| Pre-requisites & Test Setup: | * Environment full configured. (Gateway Settings, Checkout widget and custom properties configured in checkout widget js) |
| Process Flow: | 1. The shopper fill form fields on checkout page 2. The shopper submit data and confirm order 3. Generic webhook sends the request to custom gateway 4. Custom gateway just authorize or authorize and capture the payment and responses credit card webhook 5. Order submit webhook sends the order details to custom gateway and this update OCC or wait the capture or refund payment by admin panel 6. If the merchant don’t use automatic capture, the custom gateway waits the payment capture or payment refund by admin panel. When this will be done, custom payment gateway receive an notification and will be update OCC |
| Expected result: | Order Status: SUBMITTED  Payment Status: SETTLED |
| Test result: | Order Status: SUBMITTED  Payment Status: SETTLED |

# Additional Information

Provide links or references to any additional supporting material including any pre-recorded demonstration viewlets or videos. Recorded demo videos are highly recommended for including as part of Marketplace Listing content.

|  |
| --- |
|  |