Certification Tests – [Merchant Name]

Partner resource

Prepared by: Zip Integrations Team

pasted-image.tiff

# OVERVIEW

Each of the items in the below document should be completed for a successful integration. Once these tests have been completed by the Zip team, the Next Steps will be clearly outlined. If any items remain unclear, please reach out to our team and refer to the relevant sections of the API docs where all of these items are outlined:

Web Checkouts - <https://developers.zip.co/v2/docs>

Barcode - <https://developers.zip.co/v4/docs>

# TEST RESULTS (WEB CHECKOUTS)

## Miscellaneous

Once the customer is redirected to Zip, there are several possible outcomes for the user and each must to be handled appropriately. The below are the non-transactional outcomes at the checkout stage:

|  |  |
| --- | --- |
| **Use Case** | **Completed** |
| Cancelled order |  |
| Pressed “Back to …” button |  |

Notes

## Transactions

There are a number of possibilities when transacting with Zip at checkout, from existing customers to new customers or Zip Money to Zip Pay.

There are also several scenarios that require different items or values to be passed in the API calls. These will need to match our spec for their respective scenarios to achieve certification. Each of these are listed below and must be handled for a complete integration:

|  |  |
| --- | --- |
| **Use Case** | **Completed** |
| Existing Customer Zip Pay *(AU)* |  |
| Existing Customer Zip Money *(AU)* |  |
| New Approved CustomerZip Money *(AU)* |  |
| New Referred Customer Zip Money *(AU)* |  |
| New Declined Customer Zip Money *(AU)* |  |
| Order refund |  |
| Partial refund |  |
| Auth / Capture flow |  |
| Transaction data appears on Zip side |  |
| Item level data passed |  |
| Discount data passed |  |
| Zip-Version header |  |
| Idempotency-Key header |  |

Notes

## Product awareness and conversion optimisation

As a part of all Zip integrations, a key element has been identified as vital to increased checkout and conversions. As such, as a part of our partnerships we include this as a requirement when integrating Zip as an online payment method. There are number of pieces that are specifically called out in Zip agreements that are listed as below:

|  |  |
| --- | --- |
| **Use Case** | **Completed** |
| Zip landing page |  |
| Placement on the footer *(link)* |  |
| Placement on the footer *(Zip payment icon)* |  |
| Placement on the product detail page |  |
| Placement on the cart page |  |

Notes

# TEST RESULTS (IN-STORE BARCODE)

## Configuration

There are several variables that need to be able to be configured for each store location and POS. The ability to rollout of new stores easily and quickly switch off Zip temporarily if should something compromise customer experience is a must for a complete integration:

|  |  |
| --- | --- |
| **Use Case** | **Completed** |
| Store level credentials configurable (Location ID) |  |
| Kill switch (ability to turn Zip off easily) |  |

Notes

## Transactions

There are a number of possible outcomes when transacting with Zip at POS, from successful transactions to invalid store tokens, insufficient funds or invalid refunds. Each needing to be handled and are listed as below:

|  |  |
| --- | --- |
| **Use Case** | **Completed** |
| Purchase using Zip Pay |  |
| Purchase using Zip Money |  |
| Auth / Capture flow |  |
| Split purchase (part cash or card) |  |
| Refund with Zip receipt number manually |  |
| Refund by order lookup |  |
| Partial refund (Refunding 1 of 2 items on an order) |  |

Notes

## Exception scenarios

A number of non-success scenarios can result from a transaction attempt at POS. These are outlined below:

|  |  |
| --- | --- |
| **Use Case** | **Completed** |
| Invalid Zip token for purchase |  |
| Insufficient funds for purchase |  |
| Refund for higher value than original purchase |  |
| Invalid receipt ID for manual refund |  |

Notes

## Void / Reversal Scenarios

There are edge-case fail scenarios that need to be handled by a Zip integrated POS. These surround the below 3 scenarios:

* + A failure on the Zip side causing no response to be received at POS
  + A network failure to the POS or between the payment gateway and Zip
  + A Power failure at POS or server level

|  |  |
| --- | --- |
| **Use Case** | **Completed** |
| Void API call made for the order after no response is received from Zip for the POST /purchaserequests API call |  |
| Void API call made for the order after network issue when either:   * No API response is received from Zip for the POST /purchaserequests API call * No API response is received from Zip for the GET /purchaserequests API call * The response for the GET /purchaserequests API calls are still returning Pending |  |
| Void API call made for the order after power issue when either:   * No API response is received from Zip for the POST /purchaserequests API call * No API response is received from Zip for the GET /purchaserequests API call * The response for the GET /purchaserequests API calls are still returning Pending |  |

Notes