Windcave

Windcave Hosted –PX Pay 2.0 Integration Guide

Version 3.4

Contents

Windcave

1 Overview

The PX Pay 2.0 interface is a platform independent Hosted Payments Page (HPP) solution provided by Windcave. The HPP provides a solution for the capturing credit card information securely without exposing the merchant to the sensitive data.

This is achieved by allowing the card holder to enter their card details into a page which is hosted by Windcave rather than the merchants own website. The major advantage of this approach is that the merchant does not see, and is not aware of, the card number at any point in the process. This is beneficial from a PCI DSS standpoint because the scope of PCI DSS requirements is likely to be reduced.

PCI DSS (Payment Card Industry Data Security Standard) is a set of comprehensive requirements created by card issuers American Express, Discover Financial Services, JCB International, MasterCard and Visa to ensure the security of credit card data online. All merchants, whether small or large, need to be PCI compliant. Windcave is registered as a PCI DSS compliant service provider; therefore a payment page solution hosted by Windcave meets all PCI DSS requirements.

1.1 Key features

A demonstration of PX Pay 2.0 can be found online at https://demo.windcave.com/SandboxMain.aspx

2 How It Works
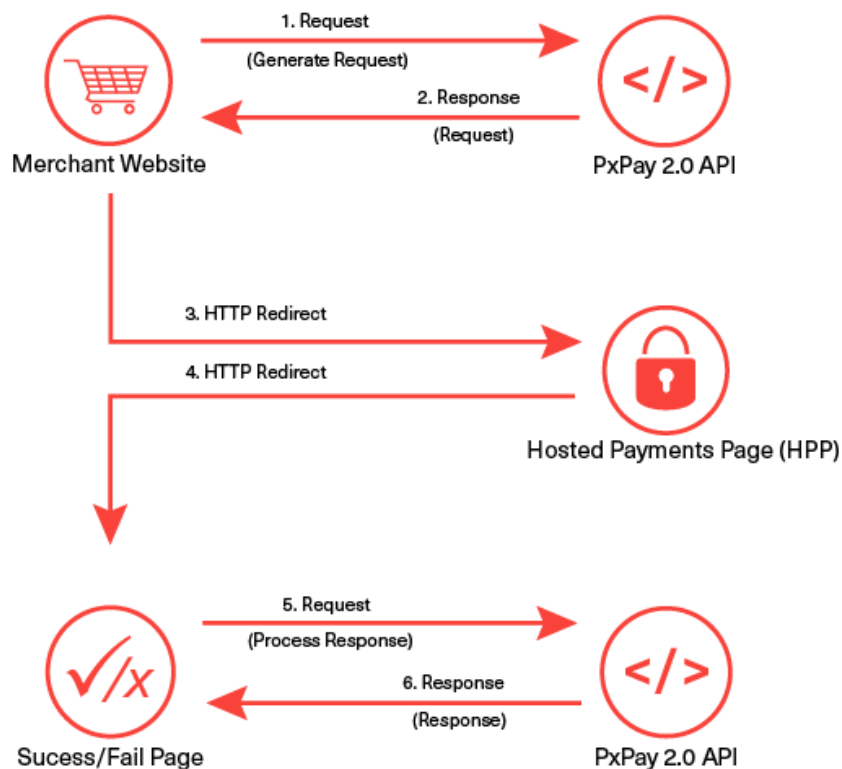
1. To process a transaction, PX Pay 2.0 allows merchants to send XML requests to Windcave® via HTTPS posts to https://sec.windcave.com/pxaccess/pxpay.aspx. PxPay Username & PxPay Key is required too.

2. Windcave® responds with a unique URI (encrypted URL) for an SSL secure payments page.

3. The merchant shopping cart uses the returned URI to redirect the customer to the secure Windcave hosted payments page.

4. The customer will be prompted to enter their credit card details and complete the transaction. The transaction is then sent to the merchant bank for authorisation. The result is displayed and the user is automatically redirected back to the merchant's website (success or fail page); i.e.https://demo.windcave.com/SandboxSuccess.aspx?result=0000840000185376f1519ff80a5ccd54&userid= SampleUserId

5. You take the "result" parameter value in the URL string i.e. 0000840000185376f1519ff80a5ccd54 along with the PX Pay username and PX Pay key; to send the response request (ProcessResponse) to Windcave and receive the XML response back.

6. The transaction results and other transaction details are decrypted and sent back to the merchant as a standard XML response. NOTE: In case a blank XML response is returned, please retry the ProcessResponse twice with an interval of 2 seconds.

3 Basic Communication

Character data sent via PX Pay 2.0 must be well formed XML.

The XML document must contain the required **opening** and **closing tags** that contains the whole document i.e. the root element.

Example: When generating the input XML document to begin a transaction request, the following GenerateRequest opening and closing tags must be present.

<**GenerateRequest**>
…
</**GenerateRequest**>

All tags must be **nested properly**. There must be an opening and a closing tag for all elements and the tags cannot overlap.

Example: Closing tags not complete.

</**AmountInput** - *has no closing angle bracket, therefore the tag is not complete.*

</**AmountInput**) - *has a wrong closing bracket, therefore the tag is not complete.*

The XML tags are **case sensitive** and **unique**. If a tag is submitted which is not recognized by Windcave and is not a required element, it will be ignored and will not be returned in the response. If the tag is for a required element, an error may occur and a response code will be returned.

Example: If the AmountInput tag was sent with a lowercase "i" instead of an uppercase "I" and error will occur the response code "IU – Invalid Amount" will be returned

<**Amountinput**>1.00</**Amountinput**> - *Incorrect*

<**AmountInput**>1.00</**AmountInput**> - *Correct*

If there is a possibility that a value will contain **invalid characters** (such as '&' and '>' in the cardholder name), please format the value using "HtmlEncoding", otherwise Windcave will be unable to read the XML and will return an error (i.e. "Not acceptable input XML").

Example, the following is invalid XML:

<**GenerateRequest**>

<TxnData1>Bill & Son</TxnData1>

<MerchantReference>Abc >> 123</MerchantReference>

</**GenerateRequest**>

The following is how it should be formatted.

<**GenerateRequest**>

<TxnData1>Bill & Son</TxnData1>

<MerchantReference>Abc >> 123</MerchantReference>

</**GenerateRequest**>

---

4   Integration Methods

Generally merchants implement a Windcave hosted payment page solution in one of two ways; either redirecting the user and their entire browser to the payment page or by presenting the payment page within an inline frame embedded in a page on their website.

4.1   Mobile Device

The PxPay 2.0 interface can also be integrated on a native mobile application as a payment method. The mobile application may utilise a webpage component to view the hosted payment page over HTTPs. The mobile platform being integrated with PxPay 2.0 should support HTTPS posts and XML data exchange.

The PxPay 2.0 payment automatically switches to a mobile format for mobile devices by default. If the PxPay 2.0 payment page is not responsive to the mobile device's screen as expected, please ensure the user agent string that the device's web browser is reporting is mobile specific within the first 255 characters of the string.

If the hosted payment page is still not responsive on the mobile device screen, please note the following:

- Please email devsupport@windcave.com and quote the exact user agent string of the relevant mobile device(s) used to send the transaction request.
- Also if a mobile web browser is used to redirect to the hosted payment page, please specify the exact mobile web browser and the version.

4.2   Redirect

The redirect integration method involves directing the user away from the merchant website to a Windcave-hosted page for the purposes of collecting credit card details. Once credit card details have been collected and a transaction processed the user is directed back to the merchant website. The image below demonstrates a payment page accessed using the redirection method.
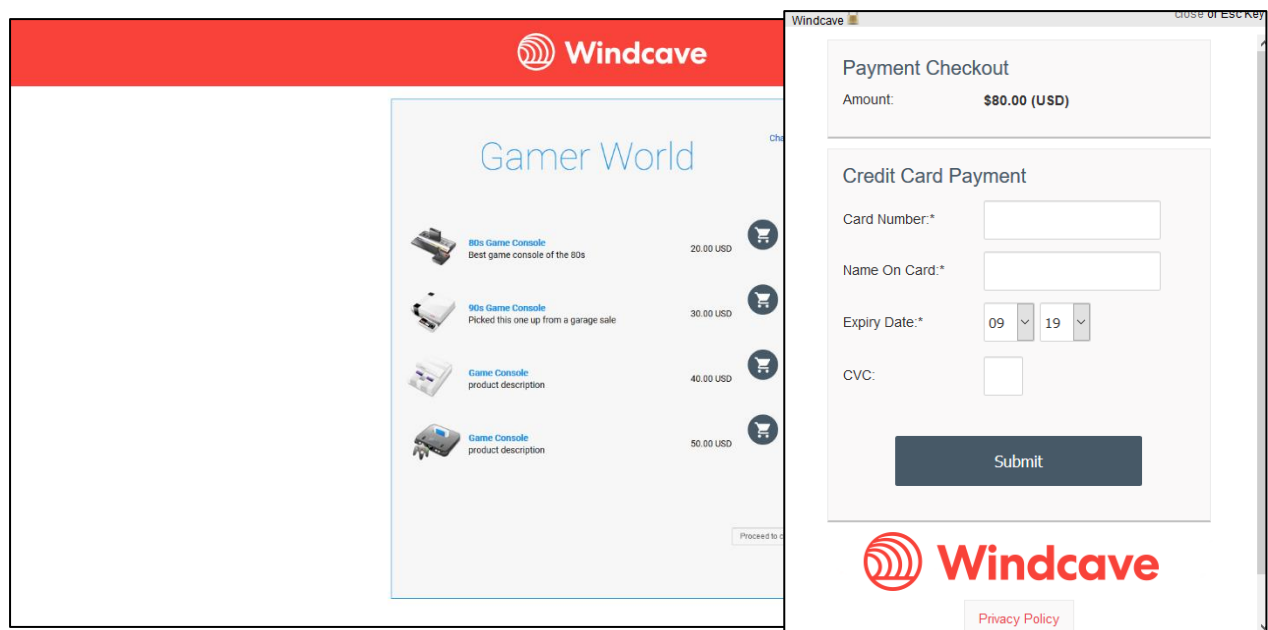
Example Merchant Website

Windcave Hosted Payments Page (HPP)

4.3 Iframe – Embedded Hosted Payment Page

The iframe integration method involves presenting the Windcave hosted payment page within the merchant website inside a frame. The iframe content can either be presented as the page loads or asynchronously (outside the normal page request flow) based upon user interaction. Note that this method of integration may increase the scope of applicable PCI-DSS requirements. Please speak to your acquirer to confirm their position on this particular implementation of the Windcave hosted payment page. Please note that the iframe window must be large enough to present the entire 400 pixel width by 470 pixel length for 3DS ACS page to display without any scrolling required.The images below demonstrate the iframe method of integration.
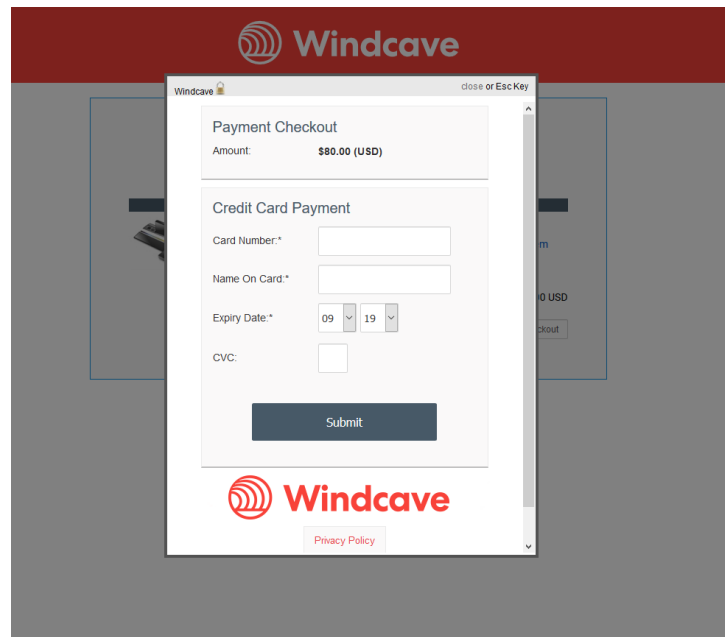
Windcave Hosted Payments Page
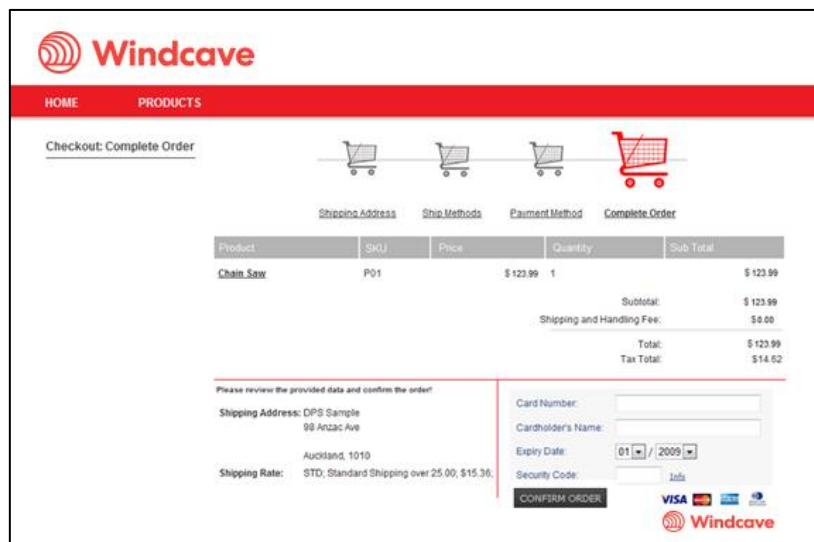
Windcave Hosted Payment Page (HPP)
embedded in an iframe

5 Preparation

To begin integration testing you will need the following:

- Windcave® PX Pay 2.0 development accounts – Contact our Ecommerce sales team to request a Dev account (Please refer to: https://www.windcave.com/contact), or apply online at https://sec.windcave.com/pxmi/apply.
- PX Pay 2.0 interface technical specification - www.windcave.com/developer-e-commerce-hosted-pxpay
- PX Pay 2.0 sample code - http://www.windcave.com/Downloads/PxPay2_SampleCode.zip

5.1 PX Pay 2.0 development account

A PX Pay 2.0 development account is usually setup within 1-3 business days. Each test account will be assigned to the Windcave test environment which simulates a connection to the merchant bank. To access the PX Pay 2.0 account, a UserId and Key will be provided.

Example: PxPayUserId: Sample2_Dev
            PxPayKey: abcdef1234567890abcdef1234567890abcdef1234567890abcdef1234567890

All PX Pay 2.0 accounts also come with a Payline account. Developers can use Payline to track down their test transactions, process transactions manually, and generate transaction reports. To access the Payline account, use the PXPayUserId along with a unique alphanumeric password setup just for Payline.

Payline login URL: https://sec.windcave.com/pxmi3/logon.

Example: Payline Username: Sample2_Dev (Same as PxPay2UserId)
            Payline Password: abcd1234

5.2 PX Pay 2.0 sample code

Sample code can be provided in the following languages:

- PHP cURL

- PHP OpenSSL

- ASP.Net 3.5 (C#)

- ASP.Net 3.5 (VB)

- Java

- ColdFusion

You will need to ensure the client library sending HTTP request is using TLSv1.2 for connectivity at all times.

All the sample codes can be downloaded from the link:

https://www.windcave.com/Downloads/PxPay2_SampleCode.zip

6 Transaction Request

GenerateRequest XML Document – To initiate a transaction the merchant posts the GenerateRequest to https://sec.windcave.com/pxaccess/pxpay.aspx
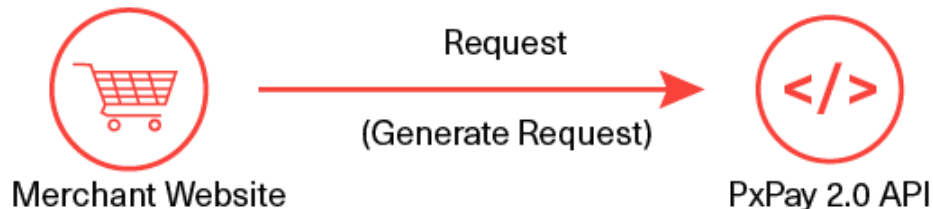

```
<GenerateRequest>

<PxPayUserId>SampleUserId</PxPayUserId>

<PxPayKey>abcdef1234567890abcdef1234567890abcdef1234567890abcdef1234567890</PxPayKey>

<TxnType>Purchase</TxnType>

<AmountInput>1.00</AmountInput>

<CurrencyInput>NZD</CurrencyInput>

<MerchantReference>Purchase Example</MerchantReference>

<TxnData1>John Doe</TxnData1>

<TxnData2>0211111111</TxnData2>

<TxnData3>98 Anzac Ave, Auckland 1010</TxnData3>

<EmailAddress>SampleUserId@windcave.com</EmailAddress>

<TxnId>ABC123</TxnId>

<BillingId>BillingId123xyz</BillingId>

<EnableAddBillCard>1</EnableAddBillCard>

<RecurringMode>single</RecurringMode>

<UrlSuccess>https://demo.windcave.com/SandboxSuccess.aspx</UrlSuccess>

<UrlFail>https://demo.windcave.com/SandboxSuccess.aspx</UrlFail>

<UrlCallback>https://InsertValidUrlForCallback</UrlCallback>

</GenerateRequest>
```

Note: Elements in blue text are optional

The following is a list of the inputs elements applicable for a GenerateRequest.

GenerateRequest (Input XML Document)

Input Element	Required	Datatype
PxPayUserId	Yes	Max 32 bytes
PxPayKey	Yes	Max 64 bytes
AmountInput	Yes	Max 13 characters
BillingId	No	Max 32 characters
CurrencyInput	Yes	Max 4 characters
EmailAddress	No	Max 255 bytes
EnableAddBillCard	No	Max 1 character
RecurringMode	No	String
MerchantReference	No	Max 64 bytes
DpsBillingId	No	Max 16 characters
TxnData1	No	Max 255 bytes
TxnData2	No	Max 255 bytes
TxnData3	No	Max 255 bytes
TxnType	Yes	Max 8 Characters
TxnId	Yes	Max 16 bytes
UrlFail	Yes	Max 255 bytes
UrlSuccess	Yes	Max 255 bytes
UrlCallback	No	Max 255 bytes
Opt	No	Max 64 bytes
ClientType	No	String
ForcePaymentMethod	No	String
DebtRepaymentIndicator	No	Number
InstallmentNumber	No	Number
InstallmentCount	No	Number

Request XML Document - Once the GenerateRequest has been processed a Request will be returned.
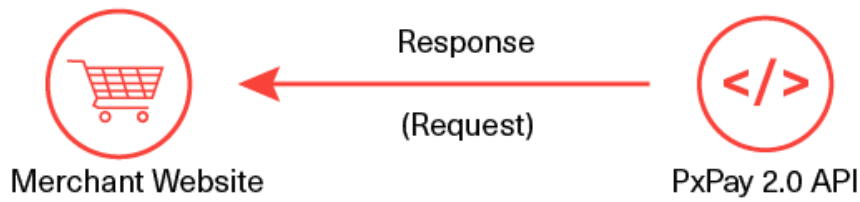
The URI returned can then be used to redirect the customer to the Windcave Hosted Payments Page.

The following is a list of the output elements applicable for a Request.

```
<Request valid="1">

<URI>https://sec.windcave.com/pxmi3/EF4054F622D6C4C1B4F9AEA59DC91CAD3654CD60ED7ED04110
CBC402959AC7CF035878AEB85D87223</URI>

</Request>
```

Request (Output XML Document)

Output Element	Datatype
valid [Attribute]	1 character
URI	String

7   Transaction Response

ProcessResponse XML Document – Once the user has submitted their credit card information and the transaction has been processed, the merchant's UrlSuccess or UrlFail or UrlCallback (if set in the request) will receive an <u>FPRN</u> with the GET HTTP message with the result value that needs to extracted and requested as below to obtain the transaction outcome and details.
To extract the transaction details from the encrypted URI string, the merchant sends the encrypted result string as Response value with their PXPayUsername and PXPaykey in the ProcessResponse (Input XML Document).

Success/Fail Page → Request (Process Response) → PxPay 2.0 API
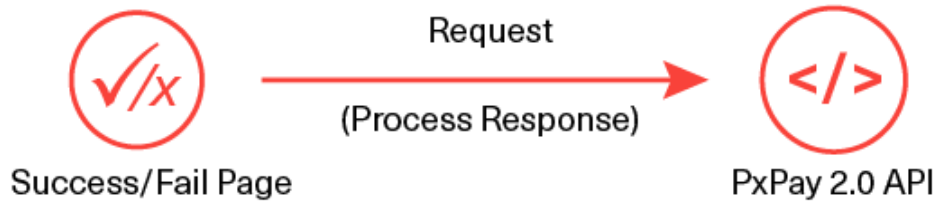
The following is a list of the inputs elements applicable to the ProcessResponse. A full description of all available elements can be found on <u>*Element Descriptions*</u> section.
. **Example:**

```
<ProcessResponse>

<PxPayUserId>SampleUserId</PxPayUserId>

<PxPayKey>abcdef1234567890abcdef1234567890abcdef1234567890abcdef1234567890</PxPayKey>

<Response>0000840000185376f1519ff80a5ccd54</Response>

</ProcessResponse>
```

ProcessResponse (Input XML Document)

Input Element	Required	Datatype
PxPayUserId	Yes	Max 32 bytes
PxPayKey	Yes	Max 64 bytes
Response	Yes	String

Response XML Document - The Response will return the decrypted transaction results for the merchant to interpret.
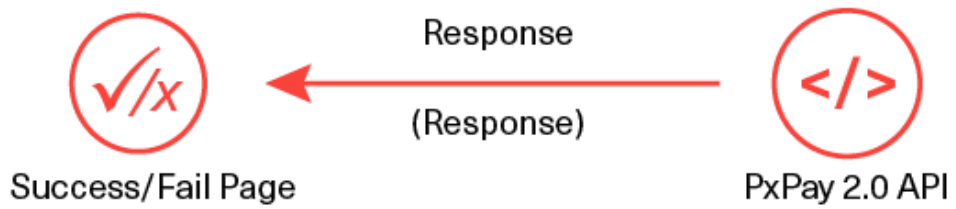
The following is a list of the output elements applicable to the Response. A full description of all available elements can be found on *Element Descriptions* section.

Response (Output XML Document)

Output Element	Datatype		Output Element	Datatype
valid [Attribute]	1 character		TxnData1	Max 255 bytes
AmountSettlement	Max 13 characters		TxnData2	Max 255 bytes
AuthCode	Max 22 characters		TxnData3	Max 255 bytes
CardName	Max 16 bytes		TxnType	Max 8 Characters
CardNumber	Max 16 bytes		CurrencyInput	Max 4 characters
DateExpiry	Max 4 characters		MerchantReference	Max 64 bytes
DpsTxnRef	Max 16 bytes		ClientInfo	Max 15 characters
Success	Long		TxnId	Max 16 bytes
ResponseText	Max 32 bytes		EmailAddress	Max 255 bytes
DpsBillingId	Max 16 characters		BillingId	Max 32 characters
CardHolderName	Max 64 bytes		TxnMac	Max 32 characters
CurrencySettlement	Max 4 characters		CardNumber2	Max 16 characters
PaymentMethod	String		Cvc2ResultCode	1 character

Windcave

8   Element Descriptions

PxPayUserId (input) Datatype: BSTR Max 32 bytes
The PxPayUserId is a unique username to identify your customer account. This username will be setup by our Windcave® Activations team and issued with your development account. The standard format used for Dev PxPayUserId is 'merchantname_dev' e.g. storename_dev.

PxPayKey (input) Datatype: BSTR Max 64 bytes
The PxPayKey is a unique 64 character key to identify customer account and used to encrypt the transaction request with 3DES to protect the transaction information. This key will be setup by our Windcave® Activations team and issued with your development account.

AmountInput (input) Datatype: BSTR Max 13 characters
The AmountInput field specifies the total Purchase or Auth amount.

Format is d.cc where d is dollar amount (no currency indicator) and cc is cents amount. For example, $1.80 (one dollar and eighty cents) is represented as "1.80", not "1.8". A string value is used rather than the conventional currency datatype to allow for easy integration with web applications. The maximum value allowable is "999999.99" however acquirer or card limits may be lower than this amount.

When submitting transactions for currencies with no decimal division of units such as the Japanese Yen (JPY), the AmountInput must be in an appropriate format e.g. "1000" for 1000 Yen. If "1000.00" is submitted in AmountInput with CurrencyInput set to JPY, an error will occur and response code "IU" will be returned.

BillingId (input/output) Datatype: BSTR Max 32 characters
If a token based billing transaction is to be created, a BillingId may be supplied. This is an identifier supplied by the merchant application that is used to identify a customer or billing entry and can be used as input instead of card number and date expiry for subsequent billing transactions. Token Billing is discussed in detail on Token Billing Section.

CurrencyInput (input/output) Datatype: BSTR Max 4 characters
The CurrencyInput field is used to specify the currency to be used e.g. "NZD" or "AUD". Possible range of currencies:

AUD	Australian Dollar		HKD	Hong Kong Dollar		SGD	Singapore Dollar
BND	Brunei Dollar		INR	Indian Rupee		THB	Thai Baht
CAD	Canadian Dollar		JPY	Japanese Yen		TOP	Tongan Pa'anga
CHF	Switzerland Franc		KWD	Kuwait Dinar		USD	United States Dollar
EUR	Euro		MYR	Malaysian Ringgit		VUV	Vanuatu Vatu
FJD	Fiji Dollar		NZD	New Zealand Dollar		WST	Samoan Tala

Windcave

FRF	French Franc		PGK	Papua New Guinean Kina		ZAR	South African Rand
GBP	United Kingdom Pound		SBD	Solomon Islander Dollar			

EmailAddress (input/output) Datatype: BSTR Max 255 bytes
The EmailAddress field can be used to store a customer's email address and will be returned in the transaction response. The response data along with the email address can then be used by the merchant to generate a notification/receipt email for the customer. This is an optional field.

EnableAddBillCard (input) Datatype: BSTR Max 1 character
The EnableAddbillCard field is used if subsequent billing is required. Setting this field to "1" will cause Windcave to store the credit card details for future billing purposes. A DpsBillingId (Windcave generated) or BillingId (merchant generated) is used to reference the card. More information on Token/Recurring billing can be found on Token Billing Section.

MerchantReference (input/output) Datatype: BSTR Max 64 bytes
The Merchant Reference field is a free text field used to store a reference against a transaction. The merchant reference allows users to easily find and identify the transaction in Payline transaction query and Windcave reports. The merchant reference is returned in the transaction response, which can be used interpreted by the merchant website. Common uses for the merchant reference field are invoice and order numbers. This is an optional field.

ForcePaymentMethod (input) Datatype: String
Allows to force the specific payment method on the hosted payment page to use if the PxPayUserId has multiple payment methods configured. Only one of the below string values should be set. Please note certain payment methods such as Account2Account, Alipay, UPOP5 (China UnionPay) and WeChat, only accept the Purchase transaction type.
Any other values are deemed invalid according to the response.
If this field is not set then by default, multiple payment method options configured would be available on the hosted payment page.
- Account2Account
- Alipay
- AmexExpressWallet
- Card
- GECard
- MasterPass
- MonerisIOP
- Oxipay
- PayPal
- UPOP5
- UPOP
- VisaCheckout
- WeChat

PaymentMethod (output) Datatype: String
The final transaction response can contain the PaymentMethod field to indicate the specific payment method used for the transaction. If this field does not exist in the final transaction response then a card payment method was used by default.

Windcave

ClientType (input) Datatype: BSTR
Allows the Client type to be passed through at transaction time. **Please note the feature may need to be enabled for the PxPay 2.0 user.** Recommended values for PxPay 2.0:
Internet = I
Recurring = R (if supported by your MID at your merchant bank expiry date will not be validated)
Please note the single letter input is case sensitive.
Please submit the full word or a single letter.

RecurringMode (input) Datatype: BSTR Max 255 bytes
In the RecurringMode request field, please set one of the card storage reason as the string listed below:
When **tokenising the card**:
- recurringinitial
- installmentinitial
- credentialonfileinitial
- unscheduledcredentialonfileinitial

When **rebilling the card** with a token returned on tokenising, set RecurringMode as one of:
- credentialonfile
- unscheduledcredentialonfile
- incremental
- installment
- recurring
- recurringnoexpiry
- resubmission
- reauthorisation
- delayedcharges
- noshow

Note: The RecurringMode string value should be set based on the merchant's business case or use case. Further details in Token Billing section.

InstallmentNumber (input) Datatype: Number
Number based value that is used to indicate the current payment number for an installment transaction. For example, if the consumer is making payment 1 of 12, then this value should be set to 1. Only used for installment based payments.

InstallmentCount (input) Datatype: Number
The number value is used to indicate the total number of payments for an installment transaction. For example, if the consumer is making 12 installment payments for total payment, then this value should be set to 12. Only used for installment based payments.

DebtRepaymentIndicator (input) Datatype: Number
Only send this field and set it to 1 to indicate that a debt repayment transaction is to be processed. Otherwise it will be set to 0 by default.

TxnData1, TxnData2, TxnData3 (input/output) Datatype: BSTR Max 255 bytes (each field)
The TxnData fields are free text fields that can be used to store information against a transaction. This can be used to store information such as customer name, address, phone number etc. This data is then returned in the transaction response and can also be retrieved from Windcave reports. This is an optional field.

Note: storing large strings of information in the TxnData fields can cause the URI returned in the Request (Response) to exceed the max URL length for some browsers.

Windcave

TxnType (input/output) Datatype: BSTR Max 8 characters
The TxnType can either be "Purchase" or "Auth".

A Purchase transaction type processes a financial transaction immediately and funds are transferred at next settlement cut-off. Basically a Purchase transaction debits the card in real time and if you need to cancel the transaction you will need to do a refund. An example of the Purchase transaction can be found on Purchase Transaction Example section.

The Auth transaction on the other hand, transfers no money. An Auth transaction type verifies that funds are available for the requested card and amount and reserves the specified amount. The amount is reserved for a period of up to 7 days (depending on merchant bank) before it clears. If you do not wish to take money from the customer you do nothing, and the reservation on the money will expire after 7 days. If you wish to process the payment a Complete transaction is sent to finalize the funds transfer.

For more information on the Auth and Complete transaction can be found on Auth Transaction Example section.

TxnId (input/output) Datatype: BSTR Max 16 bytes
On GenerateRequest XML it is a mandatory request field. Should always contain a unique, merchant application generated value that uniquely identifies the transaction. Used by Windcave® to check for a duplicate transaction generated from Merchant web site. If a duplicate TxnId in the request is detected per PxPay account, the hosted payment page URI will not be generated and an error will be returned in the XML response of the GenerateRequest XML request and the merchant site should be informed of the result. Please ensure that the value is always unique per transaction request.

UrlFail (input) Datatype: BSTR Max 255 bytes
Url of the page to redirect to if transaction failed. Please ensure the URL starts with the protocol, for example: "https:..." or "http:...". Query string characters are permitted. However percent-encoding the URL is not permitted. Please avoid using any custom ports for this URL.

UrlSuccess (input) Datatype: BSTR Max 255 bytes
Url of the page to redirect to if transaction successful. Please ensure the URL starts with the protocol, for example: "https:..." or "http:...". Query string characters are permitted. However percent-encoding the URL is not permitted. Please avoid using any custom ports for this URL.

UrlCallback (input) Datatype: BSTR Max 255 bytes
Externally accessible URL of page or resource to receive the notification on the conclusion of the HPP session when the transaction outcome is determined. Please ensure the URL starts with the protocol, for example: "https:..." or "http:...". Query string characters are permitted. Please avoid using any custom ports for this URL.

Opt (input) Datatype: BSTR Max 64 bytes
This optional parameter can be used to set a timeout value for the hosted payments page or block/allow specified card BIN ranges.

Timeout – A timeout (TO) can be set for the hosted payments page, after which the payment page will timeout and no longer allow a payment to be taken. Note: The default timeout of the created hosted payment page session is 72 hours. A specific timeout timestamp can also be specified via the request in Coordinated Universal Time (UTC). The value must be in the format "TO=yymmddHHmm" e.g. "TO=1010142221" for 2010 October 14th 10:21pm.
The merchant should submit the timeout value of when the payment page should timeout. One approach is to find the time (UTC) from when the GenerateRequest input XML document is generated and add on how much time you wish to give the customer before the payment page times out.

Example:
1. Customer finishes last stage of checkout process and clicks proceed to be redirected to the hosted payments page.
2. GenerateRequest Input XML document is generated with the required fields and the Opt field is populated with a timeout "TO=yymmddHHmm" that is 10 minutes later than the current time.
3. When the customer is redirected to the hosted payments page they will have 10 minutes to complete the payment before the page times out.

If the page times out the customer will be prompted with a "Declined – Transaction timed out" response, indicating that the page has timed out and payment was not processed. The customer will then be returned to the merchant's UrlFail page.

This feature is useful for merchants who wish control how much time a customer has to pay for the product e.g. concert tickets, where the merchant does not want people to be able to hold seats for an extended period of time due to limited availability.

Valid (output) Datatype: BSTR Max 1 character
The valid field indicates whether the initial request was valid or invalid i.e. "1" for valid and "0" for invalid.

URI (output) Datatype: BSTR
The URI field returns the URL including the encrypted transaction request that you will need to redirect the user to.

Response (input) Datatype: BSTR
The Response field should contain the encrypted URL response from Windcave, which can be obtained from the "result" parameter in the URL string that is returned to your response page.

AmountSettlement (output) Datatype: BSTR Max 13 characters
The amount of funds settled with your bank for the transaction.

AuthCode (output) Datatype: BSTR Max 22 characters
Authorisation code returned for approved transactions from the acquirer.

CardName (output) Datatype: BSTR Max 16 bytes
The card type used for the transaction. Please refrain from processing any logic based on the value of CardName as it is subject to change from Windcave. Please see examples of possible major card names below.

CardName	CardNumber	CardType
Visa	4111111111111111	<CardName>Visa</CardName>
MasterCard	5431111111111111	<CardName>MasterCard</CardName>
Amex	371111111111114	<CardName>Amex</CardName>
Diners	36000000000008	<CardName>Diners</CardName>

CardNumber (output) Datatype: BSTR Max 16 bytes
The card number used for the transaction. The full credit card number isn't shown, however the bin range is given (first 6 characters) depending on masking pattern set.

DateExpiry (output) Datatype: BSTR Max 4 bytes
The expiry date of the card used in the transaction.

DpsTxnRef (input/output) Datatype: BSTR Max 16 bytes
A DpsTxnRef is returned for every transaction. If the transaction was approved, DpsTxnRef can be used as input to a Refund transaction i.e. used to specify a transaction for refund without supplying the original card number and expiry date.

Success (output) Datatype: Long
Indicates success or failure of the transaction. A value of 0 indicates the transaction was declined or there was an error. A value of 1 indicates the transaction was approved.

ResponseText (output) Datatype: BSTR Max 32 bytes
Response Text associated with the response code of the transaction.

DpsBillingId (input/output) Datatype: BSTR Max 16 characters
When output, contains the Windcave generated BillingId. Only returned for transactions that are requested by the application with the EnableAddBillCard value is set to true indicating a token billing entry should be created. Information on Token Billing can be found on Token Billing section.

CardHolderName (output) Datatype: BSTR Max 64 bytes
The cardholder name as it appears on customer card.

CurrencySettlement (output) Datatype: BSTR Max 4 characters
Used to specify the currency that was used for the transaction: AUD, USD, NZD etc.

ClientInfo (output) Datatype: BSTR Max 15 characters
The IP address of the user who processed the transaction.

BillingId (input/output) Datatype: BSTR Max 32 characters
If a token based billing transaction is to be created, a BillingId may be supplied. This is an identifier supplied by the merchant application that is used to identify a customer or billing entry and can be used as input instead of card number and date expiry for subsequent billing transactions.

TxnMac (output) Datatype: BSTR Max 32 characters
Indication of the uniqueness of a card number.

CardNumber2 (output) Datatype: BSTR Max 16 characters
A token generated by Windcave when adding a card for recurring billing. CardNumber2 is a 16 digit number which conforms to a Luhn 'mod 10' algorithm and has a 1-to-1 relationship with the actual card number used. Please contact Windcave support if you would like to use this value.

Cvc2ResultCode (output) Datatype: BSTR 1 character
The result of CVC validation.

Response Code	Definition	Interpreting Response Codes
M	CVC matched.	You will want to proceed with transactions for which you have received an authorisation approval. A CVC match indicates the values provided matches the Issuing Banks details
N	CVC did not match.	You may want to follow up with the cardholder to verify the CVC value before completing the transaction, even if you have received an authorisation approval. The CVC details provided by the Cardholder do not match their Issuing Banks details
P	CVC request not processed.	Issuing Bank is unable to process CVC at this time
S	CVC should be on the card, but merchant has sent code indicating there was no CVC.	You may want to follow up with the cardholder to verify that the customer checked the correct location for the CVC. If the transaction is Approved you may also wish to consider not fulfilling the transaction
U	Issuer does not support CVC.	The card Issuing bank does not support CVC process.

Windcave

9   Common Scenarios

This section provides examples on the common PX Pay 2.0 scenarios.

All examples can be replicated on https://demo.windcave.com/SandboxMain.aspx using our PX Pay 2.0 Sandbox tool (https://demo.windcave.com/Sandbox.aspx).

9.1   Purchase Transaction Example

A Purchase transaction type processes a financial transaction immediately and funds are transferred at next settlement cut-off. Basically a Purchase transaction debits the card in real time and if you need to cancel the transaction you will need to do a refund.

Note additionally, if the PxPayUserId has multiple payment method configured by our activations team then you can force a specific payment method with the ForcePaymentMethod field as specified in the section Element Descriptions. Please note certain payment methods only accept the Purchase transaction type.

Step 1 - GenerateRequest (Input XML Document): Post to Windcave PX Pay 2.0 API (https://sec.windcave.com/pxaccess/pxpay.aspx).

```
<GenerateRequest>

<PxPayUserId>SampleUserId</PxPayUserId>

<PxPayKey>abcdef1234567890abcdef1234567890abcdef1234567890abcdef1234567890</PxPayKey>

<MerchantReference>Purchase Example</MerchantReference>

<TxnType>Purchase</TxnType>

<AmountInput>1.00</AmountInput>

<CurrencyInput>NZD</CurrencyInput>

<TxnData1>John Doe</TxnData1>

<TxnData2>0211111111</TxnData2>

<TxnData3>98 Anzac Ave, Auckland 1010</TxnData3>

<EmailAddress>SampleUserId@windcave.com</EmailAddress>

<TxnId>P03E57DA8A9DD700</TxnId>

<UrlSuccess>https://demo.windcave.com/SandboxSuccess.aspx</UrlSuccess>

<UrlFail>https://demo.windcave.com/SandboxSuccess.aspx</UrlFail>

</GenerateRequest>
```

Step 2 - Request (Output XML Document): Returned by Windcave PX Pay 2.0 API for merchant to use to redirect customer to Windcave Hosted Payments Page.

```
<Request valid="1">

<URI>https://sec.windcave.com/pxmi3/EF4054F622D6C4C1B4F9AEA59DC91CAD3654CD60ED7ED

04110CBC402959AC7CF035878AEB85D87223</URI>

</Request>
```

Step 3 – Redirection to HPP: Customer is redirected to the Hosted Payments Page to enter their credit card details

Page 1 – Hosted Payments Page (Enter CC details)   Page 2 - Transaction results

Note: Page 2 can be skipped so that redirection is done immediately after transaction is processed. Please refer to *HPP Customisation via Payline* for more information.

Step 4 – Redirection to UrlSuccess/UrlFail: Once transaction has been processed the customer is redirected back to the merchant's UrlSuccess of UrlFail depending on the transaction result. Transaction results are encrypted and returned in the "result" parameter of the URL string. You may also use the UrlCallback to receive the notification on your server side.

```
https://demo.windcave.com/SandboxSuccess.aspx?result=0000840000185416971799d38ef12345
&userid=SampleUserId
```

Step 5 - ProcessResponse (Input XML Document): Generated by the merchant using PxPayUserId, PxPaykey and Response details to decrypt the transaction result. The Response is the URI obtained from the "result" parameter in the URL string that is returned to the UrlSuccess/UrlFail page in step 4.

```
<ProcessResponse>

<PxPayUserId>SampleUserId</PxPayUserId>

<PxPayKey>abcdef1234567890abcdef1234567890abcdef1234567890abcdef1234567890</PxPayKey>

<Response>00008400001853747f2bc6ded6012345</Response>

</ProcessResponse>
```

Step 6 - Response (Output XML Document): Transaction results decrypted and returned to merchant.

```
<Response valid="1">

<Success>1</Success>

<TxnType>Purchase</TxnType>

<CurrencyInput>NZD</CurrencyInput>

<MerchantReference>Purchase Example</MerchantReference>

<TxnData1></TxnData1>

<TxnData2></TxnData2>

<TxnData3></TxnData3>

<AuthCode>113837</AuthCode>

<CardName>Visa</CardName>

<CardHolderName>CARDHOLDER NAME</CardHolderName>

<CardNumber>411111........11</CardNumber>

<DateExpiry>1111</DateExpiry>

<ClientInfo>192.168.1.111</ClientInfo>

<TxnId>P03E57DA8A9DD700</TxnId>

<EmailAddress></EmailAddress>

<DpsTxnRef>0000000604951234</DpsTxnRef>

<BillingId></BillingId>

<DpsBillingId></DpsBillingId>

<AmountSettlement>1.00</AmountSettlement>

<CurrencySettlement>NZD</CurrencySettlement>
```

```
<DateSettlement>20100924</DateSettlement>

<TxnMac>BDDDD619</TxnMac>

<ResponseText>APPROVED</ResponseText>

<CardNumber2></CardNumber2>

<Cvc2ResultCode>M</Cvc2ResultCode>

</Response>
```

9.2 Auth Transaction Example

An Auth transaction type verifies that funds are available for the requested card and amount and reserves the specified amount. The amount is reserved for a period of up to 7 days (depending on merchant bank) before it clears. No funds are transferred. If you do not wish to take money from the customer you do nothing, and the reservation on the money will expire after 7 days. If you wish to process the payment a Complete transaction is sent to finalize the funds transfer. This is discussed on Finalizing Auth Transactions section.

Step 1 - GenerateRequest (Input XML Document): Post to Windcave PX Pay 2.0 API (https://sec.windcave.com/pxaccess/pxpay.aspx).

```
<GenerateRequest>

<PxPayUserId>SampleUserId</PxPayUserId>

<PxPayKey>abcdef1234567890abcdef1234567890abcdef1234567890abcdef1234567890</PxPayKey>

<MerchantReference>Auth Example</MerchantReference>

<TxnType>Auth</TxnType>

<AmountInput>1.00</AmountInput>

<CurrencyInput>NZD</CurrencyInput>

<TxnData1>John Doe</TxnData1>

<TxnData2>0211111111</TxnData2>

<TxnData3>98 Anzac Ave, Auckland 1010</TxnData3>

<EmailAddress>SampleUserId@windcave.com</EmailAddress>

<TxnId>P03E890575E9D4A2</TxnId>

<UrlSuccess>https://demo.windcave.com/SandboxSuccess.aspx</UrlSuccess>

<UrlFail>https://demo.windcave.com/SandboxSuccess.aspx</UrlFail>

</GenerateRequest>
```
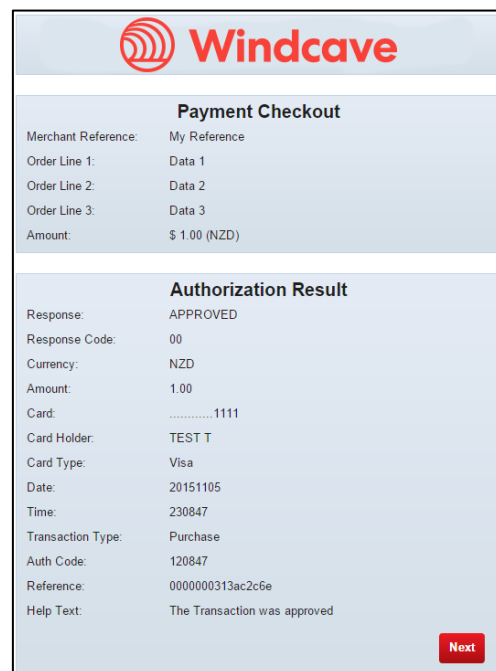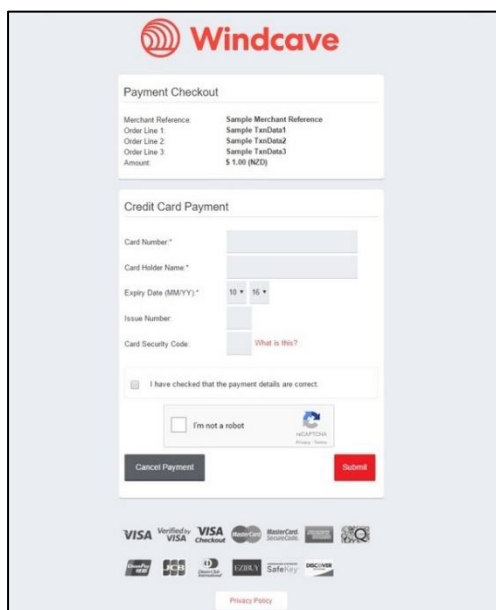
Step 2 - Request (Output XML Document): Returned by Windcave PX Pay 2.0 API for merchant to use to redirect customer to Windcave Hosted Payments Page.

<Request valid="1">

<URI>https://sec.windcave.com/pxmi3/EF4054F622D6C4C1B4F9AEA59DC91CAD3654CD60ED7ED04110CBC402959AC7CF035878AEB85D87223</URI>

</Request>

Step 3 - Redirection to HPP: Customer is redirected to the Hosted Payments Page to enter their credit card details.

Page 1 – Hosted Payments Page (Enter CC details)   Page 2 - Transaction results

Note: Page 2 can be skipped so that redirection is done immediately after transaction is processed. The Amount shown on page 1 can also be hidden.

Step 4 – Redirection to UrlSuccess/UrlFail: Once transaction has been processed the customer is redirected back to the merchant's UrlSuccess of UrlFail depending on the transaction result. Transaction results are encrypted and returned in the "result" parameter of the URL string. You may also use the UrlCallback to receive the notification on your server side.

https://demo.windcave.com/SandboxSuccess.aspx?result=00008400001854169717999d38ef12345&userid= SampleUserId

Step 5 - ProcessResponse (Input XML Document): Generated by the merchant using PxPayUserId, PxPaykey and Response details to decrypt the transaction result. The Response is the URI obtained from the "result" parameter in the URL string that is returned to the UrlSuccess/UrlFail page in step 4.

```
<ProcessResponse>

<PxPayUserId>SampleUserId</PxPayUserId>

<PxPayKey>abcdef1234567890abcdef1234567890abcdef1234567890abcdef1234567890</PxPayKey>

<Response>00008400001853747f2bc6ded6012345</Response>

</ProcessResponse>
```

Step 6 - Response (Output XML Document): Transaction results decrypted and returned to merchant.

```
<Response valid="1">

<Success>1</Success>

<TxnType>Auth</TxnType>

<CurrencyInput>NZD</CurrencyInput>

<MerchantReference>Auth Example</MerchantReference>

<TxnData1>John Doe</TxnData1>

<TxnData2>0211111111</TxnData2>

<TxnData3>98 Anzac Ave, Auckland 1010</TxnData3>

<AuthCode>101828</AuthCode>

<CardName>Visa</CardName>

<CardHolderName>CARDHOLDER</CardHolderName>

<CardNumber>411111........11</CardNumber>

<DateExpiry>1111</DateExpiry>

<ClientInfo>192.168.1.111</ClientInfo>

<TxnId>P03E890575E9D4A2</TxnId>

<EmailAddress>SampleUserId@windcave.com</EmailAddress>

<DpsTxnRef>00000006049ac588</DpsTxnRef>

<BillingId></BillingId>

<DpsBillingId></DpsBillingId>

<AmountSettlement>1.00</AmountSettlement>

<CurrencySettlement>NZD</CurrencySettlement>
```

Windcave

```
<DateSettlement>20100927</DateSettlement>

<TxnMac>BD43E619</TxnMac>

<ResponseText>APPROVED</ResponseText>

<CardNumber2></CardNumber2>

<Cvc2ResultCode>M</Cvc2ResultCode>

</Response>
```

9.3 Finalizing Auth Transactions

A Complete transaction is sent at a later date to finalize the funds transfer. The merchant does not need to supply the card number or expiry date, only the DpsTxnRef returned from the original Auth transaction.

The amount sent in the Complete transaction can be less, more, or identical to the original amount specified in the original Auth transaction.

- If the Complete transaction is sent when the funds are still reserved (i.e. within the reservation period) and the amount is identical or less than the original Auth amount, then the funds are guaranteed to be available for the merchant to charge.
- If the amount is more than the original Auth amount then the amount up to the original Auth amount is guaranteed, however the additional amount is not.
- If the Complete transaction is sent when funds are not reserved (i.e. after the reservation period) the transaction can still be processed, however the funds are not guaranteed.
- One Complete transaction can be done per Auth transaction. If a Complete transaction is processed for a lesser amount than the original Auth amount, the remaining balance is no longer reserved.

This transaction set is useful when the merchant needs to ensure that funds up to a certain limit are available but the actual total amount is not yet known or goods/services have not yet been delivered.

Please note that a complete transaction cannot be submitted via PX Pay 2.0, as the PX Pay 2.0 API is designed for the initial capture of card data only.

The Complete transactions will need to be processed manually using the Payline account (included with each PX Pay 2.0 account). Alternatively Windcave can setup a Batch Processor, PX Post, or Web Services account for complete transactions only.

The following options are discussed in the next section.

- Option 1: PX Pay 2.0 > Payline
- Option 2: PX Pay 2.0 > Batch Upload
- Option 3: PX Pay 2.0 > Batch Processor
- Option 4: PX Pay 2.0 > PX Post
- Option 5: PX Pay 2.0 > Web Services

Option 1: PX Pay 2.0 > Payline
The Auth transaction can be finalized by processing a Complete transaction via Payline.

1. Browse to https://sec.windcave.com/pxmi3/logon
2. Login using username and password.
3. Navigate to the Completions screen using the navigation menu on the left. Find the Auth transaction and click the "Complete" button.
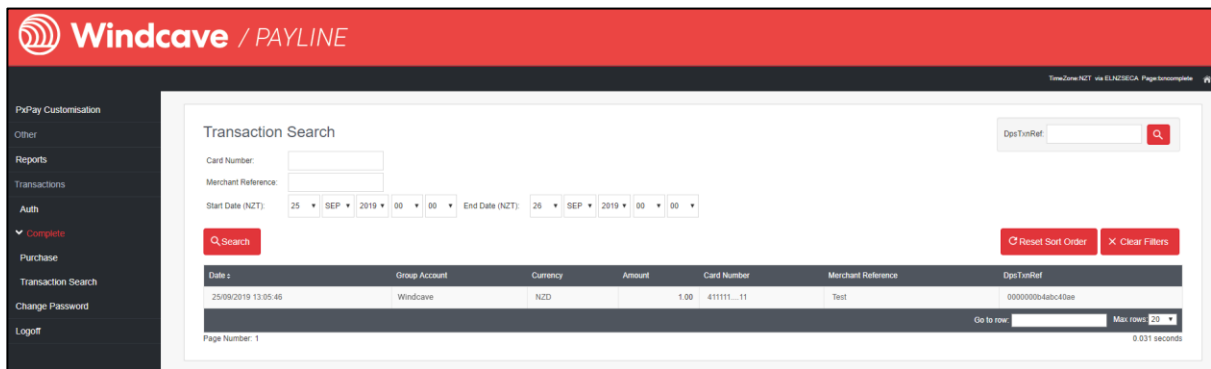
4. Enter the merchant reference and amount you wish to complete the transaction for, and click submit.
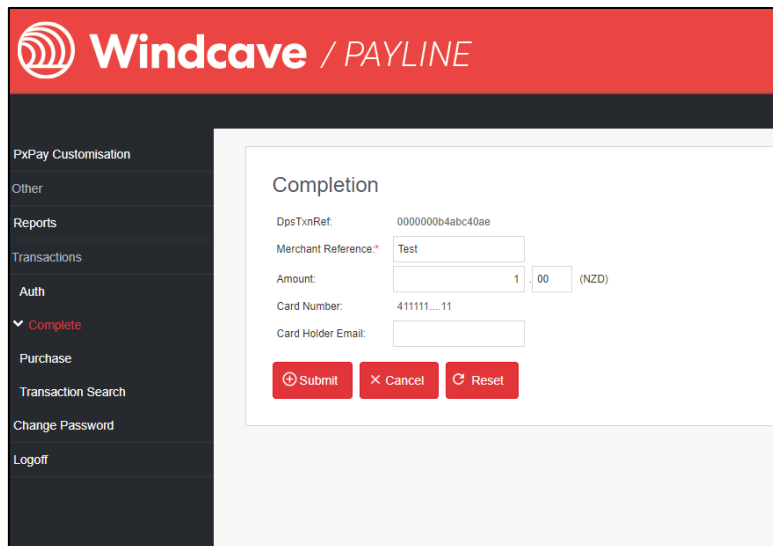
5. The transaction will be processed and the result will be returned. Clicking on the "Next Transaction" button will take you back to the Completions screen.
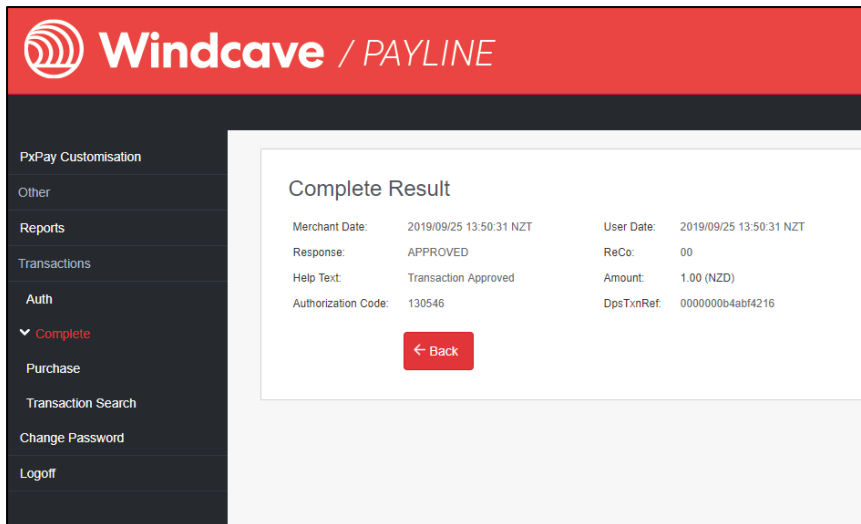
Option 2: PX Pay 2.0 > Batch Upload

The Auth transaction can be finalized by processing a Complete transaction via the Payline Batch Processor.
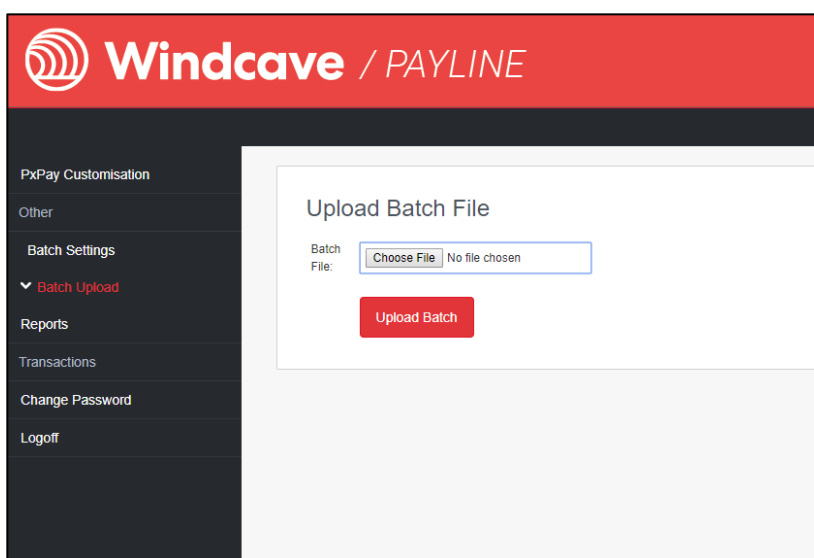
1. Process Auth transaction via PX Pay 2.0 and store the returned DpsTxnRef.

2. Create a batch input file according to specification. Technical documentation found here: https://www.windcave.com/developer-other-components-payline-batch

   Example batch input file contents:
   C,9997,MerchantReference,,,1.00,00000006049ac588,,CardHolderName

   The Complete transaction should include the DpsTxnRef returned from the original Auth. The DpsTxnRef in this example is "00000006049ac588".

3. Navigate to the Batch Upload screen using the navigation menu on the left. Click "Browse" to select your input file. Click "Upload New Batch File" to upload.

4. Once the batch file has been uploaded, the output file can be downloaded from the reports screen and transactions can be found in the transaction search screen.

Option 3: PX Pay 2.0 > Batch Processor

The Auth transaction can be finalized by processing a Complete transaction via the Windcave® Batch Processor. The Batch Processor is designed to process input files containing credit card payment information for authorisation. Communication is done using HTTPS posts.

Technical documentation found here:
https://www.windcave.com/developer-other-components-batch-processor

1. Process Auth transaction via PX Pay 2.0 and store the returned DpsTxnRef.

2. Create a batch input file according to specification.

    Example batch input file contents:

    C,9997,MerchantReference,,,1.00,00000006049ac588,,CardHolderName
    The Complete transaction should include the DpsTxnRef returned from the original Auth. The DpsTxnRef in this example is "00000006049ac588".

3. Drop the batch file into the "Input" folder to process the transaction.

4. Once the batch file has been uploaded, the output file will be generated and transactions can be found in the Payline transaction search screen.

Option 4-5: PX Pay 2.0 > PX Post/Webservice

The Auth transaction can be finalized by processing a Complete transaction via PX Post or Web Service.

PX Post: https://www.windcave.com/developer-ecommerce-pxpost#auth-completion

Web Service: https://www.windcave.com/developer-ecommerce-web-service#auth-complete

1. Process Auth transaction via PX Pay 2.0 and store the returned DpsTxnRef.
2. Send a Complete transaction via PX Post or Web Service with the DpsTxnRef.

10 Token Billing

Token Billing allows for regular billing of a cardholder card, under the control of the merchant, without requiring the merchant to either store sensitive card data securely or to obtain credit card details every time a transaction is made. This functionality is implemented by providing the ability for a merchant to request Windcave to capture and store a credit card number and expiry date and to link these stored details to a merchant supplied "BillingId" or Windcave generated "DpsBillingId".

The **BillingId** is a 32 character field that contains a reference that is unique to the merchant's customer that will be associated with the credit card information stored securely at Windcave. The BillingId is generated and supplied by merchant. This is undertaken during the Setup Phase.

The BillingId submitted does not have to be unique every time. If the BillingId of an existing card (stored at Windcave) is submitted for a new card, the BillingId will only reference the new card moving forward. The old card can still be referenced by using the unique DpsBillingId.

The **DpsBillingId** is the same as the BillingId, but is generated by Windcave not the merchant. A DpsBillingId will be generated for every transaction where the credit card information is to be stored. The returned value will be 16 characters in length and is unique. The merchant can choose to use the DpsBillingId or their own BillingId.

For subsequent charges to the card (Rebill Phase), the merchant does not need to supply the card number or expiry date, only the Token i.e. the BillingId or DpsBillingId originally associated during the Setup Phase.

10.1 Setup Phase
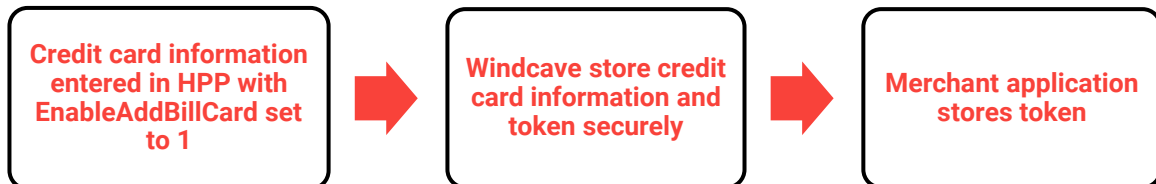
The setup phase consists of loading a card into Windcave® with a transaction. The transaction can be a Purchase or Validate or Auth transaction type. The transaction can be an online $0.00 or $1.00 Validate which will only process a non-financial (zero dollar or no hold) transaction that is used check that the card and cardholder's account is valid. If the processing bank or acquirer does not support Validate then the Validate transaction request will be converted to an Auth transaction automatically.

Alternatively a $1.00 Auth transaction type request will determine that the card is valid and not on hot or stolen card lists but depending on the processing bank or acquirer the transaction may incur a temporary financial hold of the transaction amount.

The Purchase transaction type is used if the card is to be charged with an amount and tokenised at the same time.

To add a card for future rebilling, send a PX Pay 2.0 transaction request including the following properties:

- EnableAddBillCard (Set to 1 when adding a card)
- RecurringMode (required)
- BillingId (optional)

In the **RecurringMode** request field, please set one of the card storage reason as the string listed below:

When tokenising the card, please set one of the following:

RecurringMode	Usage explanation
credentialonfileinitial	Cardholder will save card and for future orders the cardholder selects to reuse the saved card for the one-off payment.
unscheduledcredentialonfileinitial	Cardholder will save their card and for future order based on an event (such as topup) the merchant will reuse the saved card on behalf of the cardholder for the one-off payment.
recurringinitial	Cardholder will save their card and merchant will reuse the saved card on behalf of cardholder for the subscribed recurring payments.
installmentinitial	Cardholder will save their card and merchant will reuse the saved card on behalf of cardholder for the installment payments.

Please discuss with our Implementation and Sales team about your tokenisation use cases if you are unsure. The RecurringMode string value should be set based on the merchant's business case for tokenising.

You can supply your own billing ID in the BillingId field or leave it blank and use the DpsBillingId returned in the response. We also recommend sending the unique TxnId and a MerchantReference. This will make searching and identifying tokens/cards much easier.

Token Creation Example

Step 1 - GenerateRequest (Input XML Document): Post to Windcave PX Pay 2.0 API (https://sec.windcave.com/pxaccess/pxpay.aspx).

```
<GenerateRequest>

<PxPayUserId>SampleUserId</PxPayUserId>

<PxPayKey>abcdef1234567890abcdef1234567890abcdef1234567890abcdef1234567890</PxPayKey>

<MerchantReference>Create Token Example</MerchantReference>

<TxnType>Validate</TxnType>

<AmountInput>1.00</AmountInput>

<CurrencyInput>NZD</CurrencyInput>

<TxnData1>John Doe</TxnData1>

<TxnData2>0211111111</TxnData2>

<TxnData3>98 Anzac Ave, Auckland 1010</TxnData3>

<EmailAddress>SampleUserId@windcave.com</EmailAddress>

<TxnId>P03E9C7087908EB2</TxnId>

<EnableAddBillCard>1</EnableAddBillCard>
```

```
<RecurringMode>credentialonfileinitial</RecurringMode>

<UrlSuccess>https://demo.windcave.com/SandboxSuccess.aspx</UrlSuccess>

<UrlFail>https://demo.windcave.com/SandboxSuccess.aspx</UrlFail>

</GenerateRequest>
```

This transaction request has EnableAddBillCard set to 1. This will inform Windcave to store credit card information for this request. Please note that we will be using DpsBillingId in this example. If you would rather use your own BillingId, simply add the BillingId field to the GenerateRequest and populate the value with the unique reference.

e.g. <BillingId>abc12345xyz</BillingId>

Also the RecurringMode is set to indicate the reason of card storage as a token. It should be set according to business case for the card tokenisation. The possible string values for RecurringMode field are listed above.

Step 2 - Request (Output XML Document): Returned by Windcave PX Pay 2.0 API for merchant to use to redirect customer to Windcave Hosted Payments Page.

```
<Request valid="1">

<URI>https://sec.windcave.com/pxmi3/EF4054F622D6C4C1B4F9AEA59DC91CAD3654CD60ED7ED
04110CBC402959AC7CF035878AEB85D87223</URI>

</Request>
```

Step 3 - Redirection to HPP: Customer is redirected to the Hosted Payments Page to enter their credit card details.

Page 1 – Hosted Payments Page (Enter CC details)

Page 2 - Transaction results

Note: Page 2 can be skipped so that redirection is done immediately after transaction is processed. The Amount shown on page 1 can also be hidden.

Step 4 – Redirection to UrlSuccess/UrlFail: Once transaction has been processed the customer is redirected back to the merchant's UrlSuccess of UrlFail depending on the transaction result. Transaction results are encrypted and returned in the "result" parameter of the URL string. You may also use the UrlCallback to receive the notification on your server side.

> https://demo.windcave.com/SandboxSuccess.aspx?result=0000840000185416971799d38ef12345 &userid=SampleUserId

Step 5 - ProcessResponse (Input XML Document): Generated by the merchant using PxPayUserId, PxPaykey and Response details to decrypt the transaction result. The Response is the URI obtained from the "result" parameter in the URL string that is returned to the UrlSuccess/UrlFail page in step 4.

> ```
> <ProcessResponse>
>
> <PxPayUserId>SampleUserId</PxPayUserId>
>
> <PxPayKey>17ce70a6025d8b925373066d3f71704a1868ea30a8d0485b6d086f722fdd9997</PxPayKey>
>
> <Response>00008400001853747f2bc6ded6012345</Response>
>
> </ProcessResponse>
> ```

Step 6 - Response (Output XML Document)

> ```
> <Response valid="1">
>
> <Success>1</Success>
>
> <TxnType>Validate</TxnType>
>
> <CurrencyInput>NZD</CurrencyInput>
>
> <MerchantReference>Create Token
> Example</MerchantReference>
>
> <TxnData1>John Doe</TxnData1>
>
> <TxnData2>0211111111</TxnData2>
>
> <TxnData3>98 Anzac Ave, Auckland 1010</TxnData3>
>
> <AuthCode>103426</AuthCode>
>
> <CardName>Visa</CardName>
>
> <CardHolderName>CREATE TOKEN EXAMPLE</CardHolderName>
> ```

Windcave

```
<CardNumber>411111........11</CardNumber>

<DateExpiry>1111</DateExpiry>

<ClientInfo>192.168.1.111</ClientInfo>

<TxnId>P03E9C7087908EB2</TxnId>

<EmailAddress>SampleUserId@windcave.com</EmailAddress>

<DpsTxnRef>00000006049d1234</DpsTxnRef>

<BillingId></BillingId>

<DpsBillingId>0000060004444444</DpsBillingId>

<AmountSettlement>1.00</AmountSettlement>

<CurrencySettlement>NZD</CurrencySettlement>

<DateSettlement>20100928</DateSettlement>

<TxnMac>BD43E619</TxnMac>

<ResponseText>APPROVED</ResponseText>

<CardNumber2></CardNumber2>

<Cvc2ResultCode>M</Cvc2ResultCode>

</Response>
```

- The unique DpsBillingId is generated by Windcave and returned in this Response (Shown in the red text).

- The merchant can then store the DpsBillingId and use it for future billing purposes (Rebill phase).

10.2 Rebill Phase

```
┌─────────────────────┐        ┌─────────────────────┐        ┌─────────────────────┐
│ Merchant application│        │ Windcave interpret  │        │  Windcave send      │
│   sends token to    │  ──▶   │ token and process   │  ──▶   │   response of       │
│ Windcave securely   │        │    credit card      │        │ transaction back to │
│                     │        │    transaction      │        │ merchant application│
└─────────────────────┘        └─────────────────────┘        └─────────────────────┘
```

When the merchant wishes to bill a customer, the merchant application or Batch processor requests a new transaction and supplies the appropriate Token (BillingId or DpsBillingId), a MerchantReference, RecurringMode and the amount to be charged. The merchant does not need to supply the card number or expiry date, only the Token.

Windcave® uses the Token and retrieves the credit card number and expiry date stored in the Setup Phase and processes a transaction to the associated card.

PxPay 2.0 can be used to rebill the token and billing can also be processed manually using the Payline account (included with each PX Pay 2.0 account). Alternatively Windcave can setup a Batch Processor, PX Post or Web Service account for billing only.

The following options are discussed in the next section.

- Option 1: PX Pay 2.0 > PX Pay 2.0
- Option 2: PX Pay 2.0 > Batch Upload
- Option 3: PX Pay 2.0 > Batch Processor
- Option 4: PX Pay 2.0 > PX Post
- Option 5: PX Pay 2.0 > Web Services

Windcave

Option 1: PX Pay 2.0

PX Pay 2.0 Rebilling Example

Step 1 - GenerateRequest (Input XML Document): Post to Windcave PX Pay 2.0 API (https://sec.windcave.com/pxaccess/pxpay.aspx).

It is important to set the **RecurringMode** request field, please set one of the card storage reason as the string listed below:
When **rebilling the card with token**, please set one of the following.

RecurringMode	Usage explanation
credentialonfile	Cardholder selects their saved card to make the one-off rebill payment.
unscheduledcredentialonfile	Merchant initiated and event driven one-off rebilling with stored card (e.g. auto topups).
installment	Merchant initiated rebilling payments in installments with a stored card token.
incremental	Merchant initiated incremented transaction amount to rebill, e.g hospitality, rentals, etc.
recurring	Merchant initiated recurring transaction with a stored card token (e.g subscriptions).
recurringnoexpiry	Merchant initiated recurring transaction with a stored card token where no card expiry check needs to occur (e.g subscriptions).
resubmission	Merchant resubmits rebill with token where it requested an authorisation, but may have received a decline due to insufficient funds, but the order already delivered to the cardholder. Used with the token to get an outstanding payment from cardholder.
reauthorisation	Merchant initiated when the completion or conclusion of the original order or service extends beyond the authorisation validity. Common for retail (split or delayed shipments) and hospitality or rental services scenarios.
delayedcharges	Merchant initiated to process additional account rebill charge after original order and payment has been already processed and fulfilled.
noshow	Merchant initiated to charge the cardholder a penalty relevant to the merchant's cancellation policy. Common for guaranteed reservations scenarios, e.g. Hospitality.

Please discuss with our Implementation and Sales team about your rebilling use cases if you are unsure. The RecurringMode string value should be set based on the merchant's business case for rebilling the card.

Example PxPay request:

```
<GenerateRequest>

<PxPayUserId>SampleUserId</PxPayUserId>

<PxPayKey>abcdef1234567890abcdef1234567890abcdef1234567890abcdef1234567890</PxPayKey>

<MerchantReference>Rebill Token Example</MerchantReference>

<TxnType>Purchase</TxnType>

<AmountInput>100.00</AmountInput>

<CurrencyInput>NZD</CurrencyInput>

<TxnData1>John Doe</TxnData1>

<TxnData2>0211111111</TxnData2>

<TxnData3>98 Anzac Ave, Auckland 1010</TxnData3>

<EmailAddress>SampleUserId@windcave.com</EmailAddress>

<TxnId>P03E9C7087908EB2</TxnId>

<DpsBillingId>0000060004444444</DpsBillingId>

<RecurringMode>credentialonfile</RecurringMode>

<UrlSuccess>https://demo.windcave.com/SandboxSuccess.aspx</UrlSuccess>

<UrlFail>https://demo.windcave.com/SandboxSuccess.aspx</UrlFail>

</GenerateRequest>
```

This transaction request uses the DpsBillingID that was generated in the setup phase to recharge the card. Please note that we will be using DpsBillingId in this example. If you would rather use your own BillingId (that was generated in the setup phase), simply add the BillingId field to the GenerateRequest.

e.g. <BillingId>abc12345xyz</BillingId>

Step 2 - Request (Output XML Document): Returned by Windcave PX Pay 2.0 API for merchant to use to redirect customer to Windcave Hosted Payments Page.

```
<Request valid="1">

<URI>https://sec.windcave.com/pxmi3/EF4054F622D6C4C1B4F9AEA59DC91CAD3654CD60ED7ED
04110CBC402959AC7CF035878AEB85D87223</URI>

</Request>
```

Windcave

Step 3 - Redirection to HPP: Customer is redirected to the Hosted Payments Page to enter their Card Security
Code.

Page 1 – Hosted Payments Page
(Card Number, Card Holder name and Expiry Date
should be pre-defined)

Page 2 - Transaction results

Note: Page 2 can be skipped so that redirection is done immediately after transaction is processed. The Amount shown on page 1 can also be hidden. Refer to *Fail-Proof Result Notification* section.

Step 4 – Redirection to UrlSuccess or UrlFail: Once transaction has been processed the customer is redirected back to the merchant's UrlSuccess of UrlFail with the below GET message depending on the transaction result. Transaction results are encrypted and returned in the "result" parameter of the URL string. You may also use the UrlCallback to receive the same notification on your server side.

https://demo.windcave.com/SandboxSuccess.aspx?result=0000840000185416971799d38ef12345
&userid=SampleUserId

Step 5 - ProcessResponse (Input XML Document): Generated by the merchant using PxPayUserId, PxPaykey and Response details to decrypt the transaction result. The Response is the URI obtained from the "result" parameter in the URL string that is returned to the UrlSuccess/UrlFail page in step 4.

<ProcessResponse>

<PxPayUserId>SampleUserId</PxPayUserId>

<PxPayKey>17ce70a6025d8b925373066d3f71704a1868ea30a8d0485b6d086f722fdd9997</PxPayKey>

```
<Response>00008400001853747f2bc6ded6012345</Response>

</ProcessResponse>
```

Step 6 - Response (Output XML Document)

```
<Response valid="1">

<Success>1</Success>

<TxnType>Purchase</TxnType>

<CurrencyInput>NZD</CurrencyInput>

<MerchantReference> Rebill Token Example</MerchantReference>

<TxnData1>John Doe</TxnData1>

<TxnData2>0211111111</TxnData2>

<TxnData3>98 Anzac Ave, Auckland 1010</TxnData3>

<AuthCode>103426</AuthCode>

<CardName>Visa</CardName>

<CardHolderName> REBILL TOKEN EXAMPLE</CardHolderName>

<CardNumber>411111........11</CardNumber>

<DateExpiry>1111</DateExpiry>

<ClientInfo>192.168.1.111</ClientInfo>

<TxnId>P03E9C7087908EB2</TxnId>

<EmailAddress>SampleUserId@windcave.com</EmailAddress>

<DpsTxnRef>00000006049d1234</DpsTxnRef>

<BillingId></BillingId>

<DpsBillingId>0000060004444444</DpsBillingId>

<AmountSettlement>1.00</AmountSettlement>

<CurrencySettlement>NZD</CurrencySettlement>

<DateSettlement>20100928</DateSettlement>

<TxnMac>BD43E619</TxnMac>

<ResponseText>APPROVED</ResponseText>
```

The DpsBillingId will be the same to what was supplied by the merchant.

Windcave

```
<CardNumber2></CardNumber2>

<Cvc2ResultCode>M</Cvc2ResultCode>

</Response>
```

Option 2: PX Pay 2.0 > Batch Upload

Billing customers via Payline Batch Processor.

1. Merchant application will store the token returned in the PX Pay 2.0 Response.

2. Create a batch input file according to specification. Technical documentation found here:
    https://www.windcave.com/developer-other-components-payline-batch

    Example batch input file contents:
    B,9997,MerchantReference,0000060004444444,,1.00,,,CardHolderName

    The billing transaction should include the token. The token (DpsBillingId) in this example is
    "0000060004444444".

3. Navigate to the Batch Upload screen using the navigation menu on the left. Click "Browse" to select
    your input file. Click "Upload New Batch File" to upload.
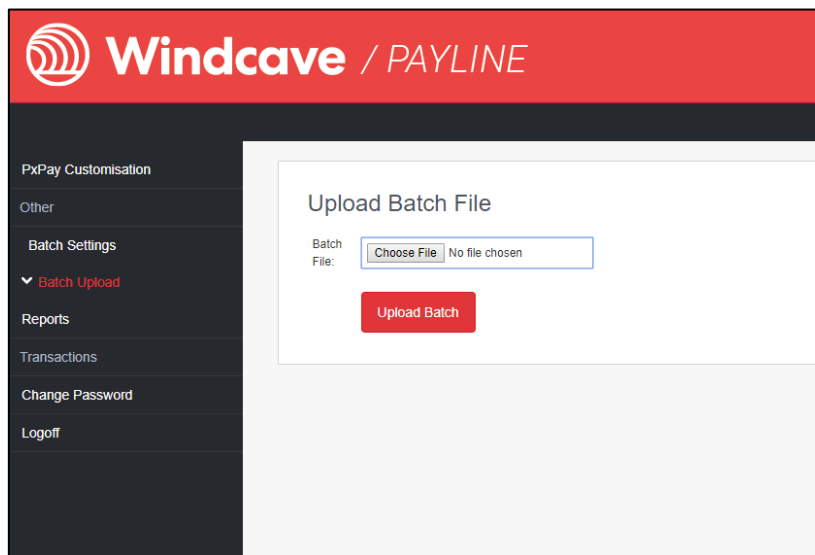
4. Once the batch file has been uploaded, the output file can be downloaded from the reports
    screen and transactions can be found in the transaction search screen.

Option 3: PX Pay 2.0 > Batch Processor

Billing can be done via the Windcave® Batch Processor. The Batch Processor is designed to process
input files containing credit card payment information for authorisation. Communication is done using
HTTPS posts.

Technical documentation found here:
https://www.windcave.com/developer-other-components-batch-processor

1. Merchant application will store the token returned in the PX Pay 2.0 Response.

2. Create a batch input file according to specification.

   Example batch input file contents:
   B,9997,MerchantReference,0000060004444444,,1.00,,,CardHolderName

   The billing transaction should include the token. The token (DpsBillingId) in this example is "0000060004444444".

3. Drop the batch file into the "Input" folder to process the transaction.

4. Once the batch file has been uploaded, the output file will be generated and transactions can be found in the Payline transaction search screen.

Option 4-5: PX Pay 2.0 > PX Post or Webservice
1. Merchant application will store the token returned in the PX Pay 2.0 Response.
2. Send a Purchase transaction via PX Post or Web Service with the token.
Please visit one of the API options for rebilling or charging the card with a token:

- Token Billing via PX Post
- Token Billing via Web Service

11 UPOP (UnionPay Online Payments)

PX Pay 2.0 is integrated with UPOP (UnionPay Online Payments). The integration is handled between Windcave and UnionPay and the payment functionality is made available to all integrators of PX Pay 2.0.

There is no additional development work required to enable UPOP on existing PX Pay 2.0 integrations – we enable it as a Payment Method at the Windcave end and the endpoint is the same as the standard PxPay 2.0 endpoint - however there are potential exceptions your application will need to handle. Because the UPOP payment method involves a redirect to a UnionPay hosted payment page for card holder card entry verification before redirection back to PxPay 2 with the transaction result, there is more potential for unusual navigation and unusual call back behaviour.

For this reason implementing PxPay 2.0's FPRN is essential. FPRN is of course enabled by default, but it requires development on your part to make use of the messages.

Additionally there is an exception scenario that has been noted with UPOP where the card holder after redirection from PxPay 2 to UPOP tries to click back in their browser to navigate back to PxPay 2. The transaction fails at this point but in some browsers a call back to your success / fail page may occur. If the result string from this call back is queried with a Process Response immediately then the Response will be as follows:

```
<Response valid="1">

  <Reco>N1</Reco>

  <ResponseText>No Transactions</ResponseText>

</Response>
```

Your application should be able to handle receiving this and to know that the transaction result can be ascertained later, either by waiting 10 minutes or so to re-query the result string, or by waiting to get the result via FPRN. Such transactions are almost always declined transactions.

Please note: UPOP transactions do not support tokenisation.

Merchants can request to add UPOP to their PX Pay 2.0 account by contacting their Windcave Account Manager to discuss on boarding process.

The following section shows the Cardholder journey via UPOP Authentication Interface.

PX Pay 2.0 Select Payment Method: Once on the PX Pay 2.0 Hosted Payments Page, the cardholder is prompted with various Payment Methods options. Selecting the "China UnionPay" option will provide the cardholder the option to continue (redirect) to UPOP.

Please note: Credit Card payments can be completely segmented out from the UnionPay flow, depending on your requirements.

UPOP Authentication: Once on the UPOP page, the cardholder is prompted to login or pay as guest. These screens will be familiar to existing UPOP users.

The amount, order number reference and merchant names will be displayed at the top of the page. These details remain displayed throughout the process.
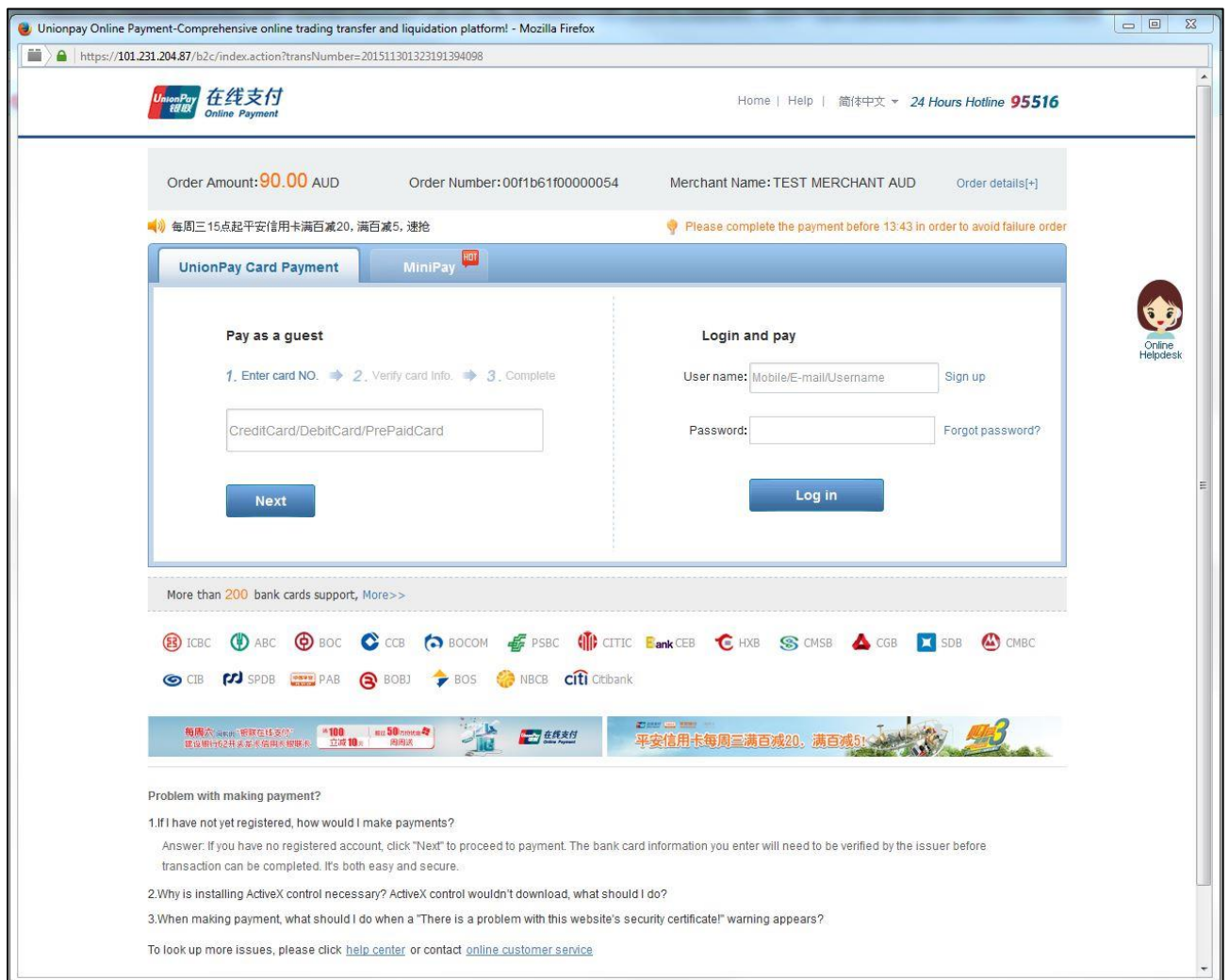
UPOP Payment: The cardholder can choose to login using their UPOP user/pass details OR pay as guest by entering their UnionPay card number.
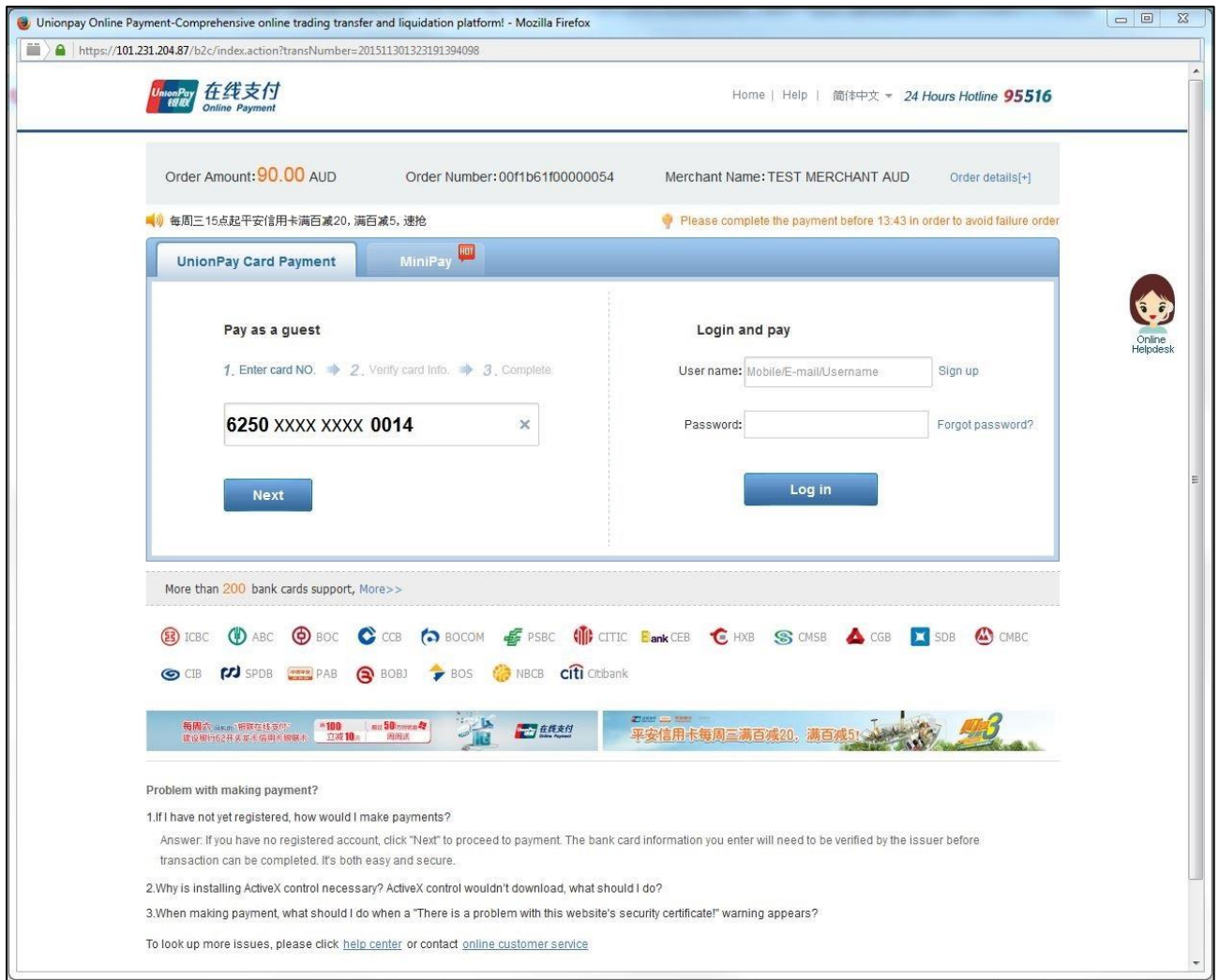
UPOP Payment Security: The cardholder is prompted to enter various security details to complete the payment process. Security details such as CVC, SMS Code, and Web Verification Code. This is part of the standard UPOP security process so all existing UPOP users should be familiar.

PX Pay 2.0 Transaction Result: On successful completion of the UPOP payment process, the cardholder is redirected back to PX Pay 2.0 Hosted Payments Page and will be shown the transaction result details. The cardholder can then click the next button to be redirected back to the merchant website.

Please note: this step can be skipped if the merchant elects to display the results page themselves. This is configured on the PX Pay 2.0 user.

12 HPP Customisation

Windcave has provides a service which allows merchants to maintain a consistent look and feel between their website and the Hosted Payment Page. Merchants can select one of a number of pre-set styles and apply images and logos, colours, buttons and dynamically populated fields to the payment page. PxPay 2.0 also support merchant submitted custom CSS which provide a greater flexibility in interface design.

- Ability to customise the look and feel of the hosted payments page.
- Pre-set style sheets available to customise the look and feel of the hosted payments page.
- Support for merchant submitted custom header logo and background images.
- Support for merchant submitted custom CSS.
- Display logos of accepted cards.
- Display or hide Payment Checkout information such as Amount, Merchant Reference and TxnData1-3 fields.
- Ability to customise all field labels & titles.
- Preview option available before deployment.
- Display or Skip Payment Result page.

Requirements

- A PxPay 2.0 development or live account.
- "Setup Custom Feature" on your Payment Manager logon at https://sec.windcave.com/pxmi3/logon

(Please contact sales@windcave.com if you do not have a Payline account and/or the Custom hosted screen available on your Payline portal.)

12.1 HPP Customisation via Payline

Windcave provides a service which allows merchants to maintain a consistent look and feel between their website and the Hosted Payment Page. Merchants can select one of a number of pre-set styles and apply images and logos, colours, buttons and dynamically populated fields to the payment page.

- Your Setup Custom Features screen will look like this:

For the above fields you can:

- Hide/unhide the label fields (please use these settings rather than hide/unhide via Custom CSS)
- Set a customised label for the fields

In the Payment Page tab, you can configure various settings related to the hosted payment page with the payment methods including fields related to the card payment form.

In the Result Page tab, if you wish to skip the second hosted page which displays the final transaction results, "Disable Result Page (SkipPage2)" can be used.

- You will be able change the interface styles on the Page Style tag. There are 1-6 legacy and newer 7-13 styles available.

- You will be able to preview your styles from the preview page.

HPP Page 1 Sample - Customisable Fields (Payment Details & Payment Page)

Images
Dps Header Logo
Background Logo
Customer Logo

Label Title:

Label Merchant Reference::	MerchantRef
Label TxnData1::	TxnData1
Label TxnData2::	TxnData2
Label TxnData3::	TxnData3
Label Amount::	1.00 (NZD)

Select Payment Method

Label Payment Method:

- Credit Card
- Pago
- Account2Account

Credit Card Payment

Label CardNumber::	
Label NameOnCard::	
Label ExpiryDate::	
Label CardSecurityCode::	What is this?

Label Submit Button:
Label Cancel Button:

Images
Dps Footer Logo

Payment Details
Label Title
Label Amount
Label TxnData1
Label TxnData2
Label TxnData3
Label Merchant Reference
Hide Currency
Hide Amount
Hide TxnData1
Hide TxnData2
Hide TxnData3
Hide Merchant Reference

Payment Page
Label CardNumber
Label NameOnCard
Label Payment Method
Label Submit Button
Label ExpiryDate
Label CardSecurityCode
Label IssueNumber
Label Cancel Button
Hide Cancel Button
Show IssueNumber
Show MasterCard Logo
Show MasterCard SecureCode Logo
Show JCB Logo
Show EzibuyGift Logo
Show Visa Logo
Show VerifiedByVisa Logo
Show Amex SafeKey Logo
Show Diners Logo
Show Cvc2 Help Link
Minimum Card Digits
Maximum Card Digits

PxPay Custom Hosted

Payment Details	Payment Page	Result Page	Payform Page	Images	Page Style	A2A Specific	Cardholder Receipt	Preview Page

Label Title:	Payment Checkout	Label Details Description:	
Hide Amount	☐	Label Amount:	Amount
Hide TxnData1	☐	Label TxnData1:	Order Line 1
Hide TxnData2	☐	Label TxnData2:	Order Line 2
Hide TxnData3	☐	Label TxnData3:	Order Line 3
Hide Merchant Reference	☐	Label Merchant Reference:	Merchant Reference
Label Merchant Authorized Name:	Merchant Name	Label Merchant Authorized Address:	Merchant Address
Hide Currency	☐		

✓ Apply   ⟳ Reset

HPP Page 1 Sample - Customisable Fields (Page Style)

Depending on the style you have selected you can configure each of the various colours properties. Please note that the colours you select must be web colours only and does not need a preceding '#' symbol entered. Once you have selected the colours you want, click on 'Preview Page' tab to view the sample output.

There are 6 preset CSS styles available for selection. The color set on the wizard will overtake the preset color in CSS.

HPP Page 2 Sample - Customisable Fields (Result Page)

Result Page
Label Title
Label Amount
Label Card
Label CardHolder Name
Label Card-Type
Label TxnDate
Label TxnTime
Label TxnType
Label ReCo
Label ResponseText
Label HelpText
Label AuthCode
Label Done Button
Label Cancel Button
Label Retry Button
Hide Currency
Hide Amount
Hide Card
Hide Card Holder Name
Hide Card-Type
Hide TxnDate
Hide TxnTime
Hide TxnType
Hide ReCo
Hide ResponseText
Hide HelpText
Hide AuthCode

PxPay Custom Hosted

Tabs: Payment Details | Payment Page | Result Page | Payform Page | Images | Page Style | A2A Specific | Cardholder Receipt | Preview Page

Field		Label Field	Value
Disable Result Page (SkipPage2)	☐	Label Title:	Authorization Result
Hide Currency	☐	Label Currency:	Currency
Hide Amount	☐	Label Amount:	Amount
Hide Card	☐	Label Card:	Card
Hide Card Holder Name	☐	Label CardHolder Name:	Card Holder
Hide Card-Type	☐	Label Card-Type:	Card Type
Hide TxnDate	☐	Label TxnDate:	Date
Hide TxnTime	☐	Label TxnTime:	Time
Hide Txn Datetime (User TZone)	☐	Label Txn Datetime (User TZone):	Transaction Date
Hide TxnType	☐	Label TxnType:	Transaction Type
Hide ReCo	☐	Label ReCo:	Response Code
Hide ResponseText	☐	Label ResponseText:	Response
Hide HelpText	☐	Label HelpText:	Help Text
Hide AuthCode	☐	Label AuthCode:	Auth Code
Hide DpsTxnRef	☐		
Label Payment Result Mail Hasbeen Sent:	The transaction result mai	Label Send Payment Result to Cardholder:	Send me an email receipt
Label Email:	Email	Label Done Button:	Next
Label Retry Button:	Retry	Label Cancel Button:	Cancel Payment
Label Result Description:			

✓ Apply   ⟳ Reset

Windcave

12.2 Merchant Submitted CSS

The merchant can also modify the CSS code from the default CSS template and customise the payment page interface to suit their purpose. For sample CSS files, please see our website.

The following diagram outlined the most common Class (.) and ID (#) selectors that can be adjusted using CSS.
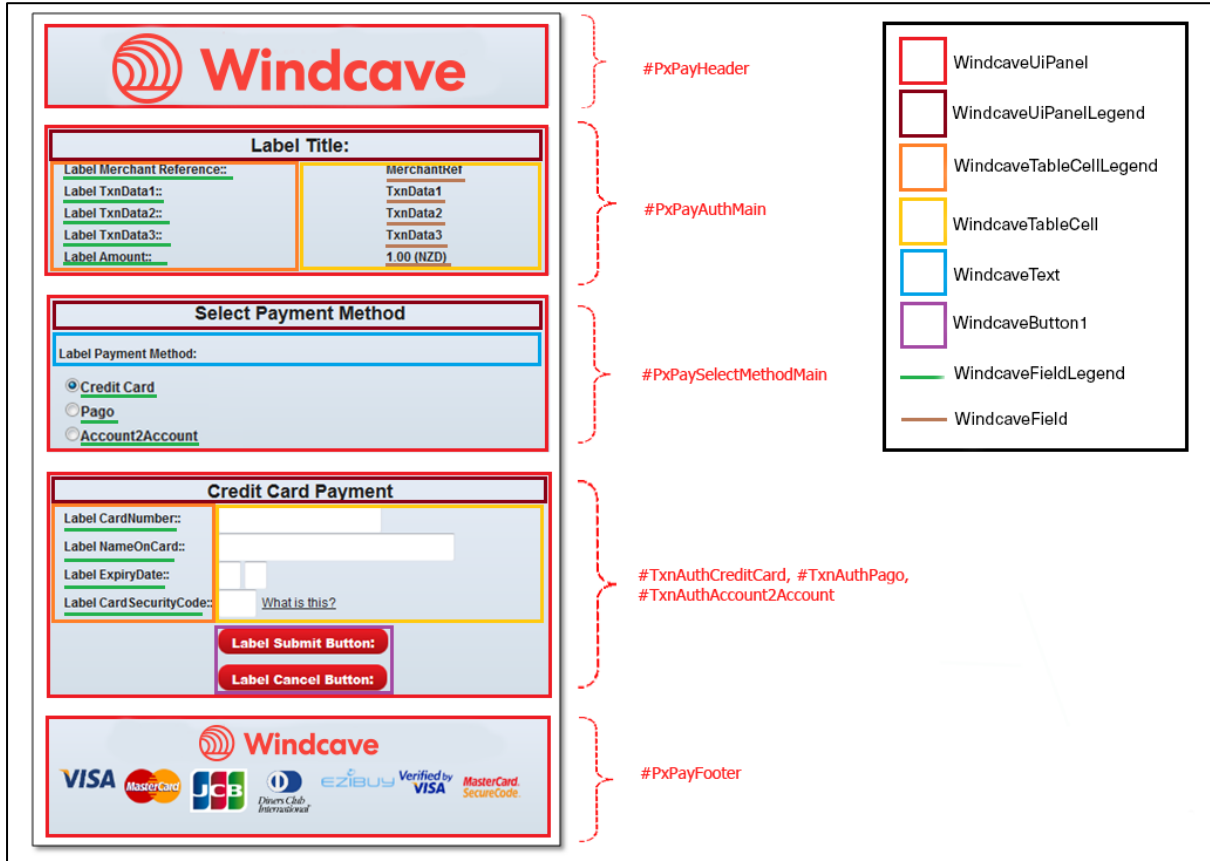
Please ensure the CSS file ONLY contains CSS code. No HTML and Java Script is allowed in the file.

Please use the custom hosted settings to hide some elements in the payment page such as label fields or buttons. Please avoid hiding elements from the CSS itself.

On the preview page, you can simulate a test transaction by using test credit card credentials (credit card number: 4111111111111111, using any expiry date in the future. To update changes you have made click 'Update'.

- Upload a merchant logo or background images:

Please email your images to support@windcave.com quoting your PXPay ID in the email. Please send image files in web format only (.jpg, .jpeg, .gif, .png). Please ensure image file is less than 40KB in file size.

- Submit a customized CSS file:

Please email your CSS file to support@windcave.com quoting your PXPay ID in the email. You can minify CSS.

Please ensure the CSS file is less than 48KB and contains CSS code only. No custom HTML and Javascript are allowed.

- **Apply Changes and bring your changes live**

To activate the customised hosted payments page that you have just created (bring the changes live) please email support@windcave.com quoting your PXPay ID in the email.

13 Fail Proof Result Notification (FPRN)

Fail Proof Result Notification (FPRN) is a server side service that provides additional assurance that the merchant website will receive notification regarding the outcome of transactions completed via the Windcave Hosted Payment Page (HPP).

FPRN helps cater for the possibility that a user may not successfully navigate to the nominated success or failure URL enabling the merchant web application to acknowledge the outcome of the transaction. The user could close their browser or otherwise navigate away from the Windcave hosted payment page once they have been informed of the transaction outcome. The merchant's web server may be temporarily unavailable as the transaction is completed and therefore unable to recognise that a transaction has taken place. Using the FPRN service the merchant website is virtually guaranteed to receive notification of the each and every transaction.

FPRN is highly recommended by Windcave and is enabled on all new accounts by default. The service ensures that the following processes occur for every transaction performed via hosted payment page:

1. Once the transaction outcome is determined for the HPP session, a background process at Windcave makes an HTTP GET request to the merchant nominated URL set on the GenerateRequest fields:
   UrlSuccess, UrlFail or UrlCallback. The URLs should resolve to a HTTP 200, 302 or 303 response externally.

2. If the merchant nominated URL is unreachable or returns any <u>HTTP status code other than 200 or 404</u> the notification (HTTP GET) is retried up to a maximum of six times.
   For example, a 500 HTTP status code, indicating a temporary problem at the client site, will cause a retry.

3. Our FPRN service will stop sending any notifications immediately on receiving a 404 (Not Found) or 502 (Bad Gateway) HTTP status code.

We can configure the FPRN HTTP status code behaviour further per API user if required. Also per API user we can configure additional query string parameter key value pair for the FPRN GET requests only for the specific API user account. This may help lock a static authentication value for server side notifications. Please contact support@windcave.com to get assistance with this. Otherwise we recommend to simply set the UrlCallback field per GenerateRequest.

In order to ensure that the web application is in the best position to acknowledge the outcome of each and every transaction certain guidelines should be followed.

The merchant web application should not:

- Filter or base any conditional logic upon the originating IP address as this can vary.
- Depend upon receiving one and only one HTTP GET request for the success, fail or callback URL from the Windcave FPRN system (multiple requests may be sent).

The merchant web application should:

- Rely on the UrlSuccess, UrlFail or UrlCallback field to receive the notification GET message. Note: UrlCallback can be used to receive the server side notification (regardless of transaction outcome) separately to the same GET redirects coming from a client side such as a web browser.
- Get the 'result' query string parameter value from the GET notification request to UrlSuccess, UrlFail or UrlCallback. Your URL will have that appended 'result' query string parameter containing

Windcave

the encrypted key for the transaction details. The query string also contains the 'userid' to identify the API user associated to the transaction outcome.

- Use the userid as PxPayUserId, include the PxPayKey stored securely in your server side and result parameter as Response value for the final ProcessResponse request to get the final decrypted transaction outcome details.
- Determine if a database operation or some form of communication such as generating an order record or sending an email is required. Generally this will mean that the application needs to be aware if these actions have been taken previously for the particular transaction or not. TxnId should be used for this purpose.

Note: The URL at which the merchant website will process FPRN requests must be exposed via standard internet ports i.e. port 80 or port 443 for SSL/TLS traffic. When specifying UrlSuccess, UrlFail or UrlCallback field values do not specify a non-standard port number within the URL.

Please note FPRN is turned on by default.

14 3D Secure

As part of Windcave' continuing efforts to promote best practice solutions, the Hosted Payments Package allows for merchants to participate in the Verified by Visa and MasterCard's SecureCode schemes at no additional cost.
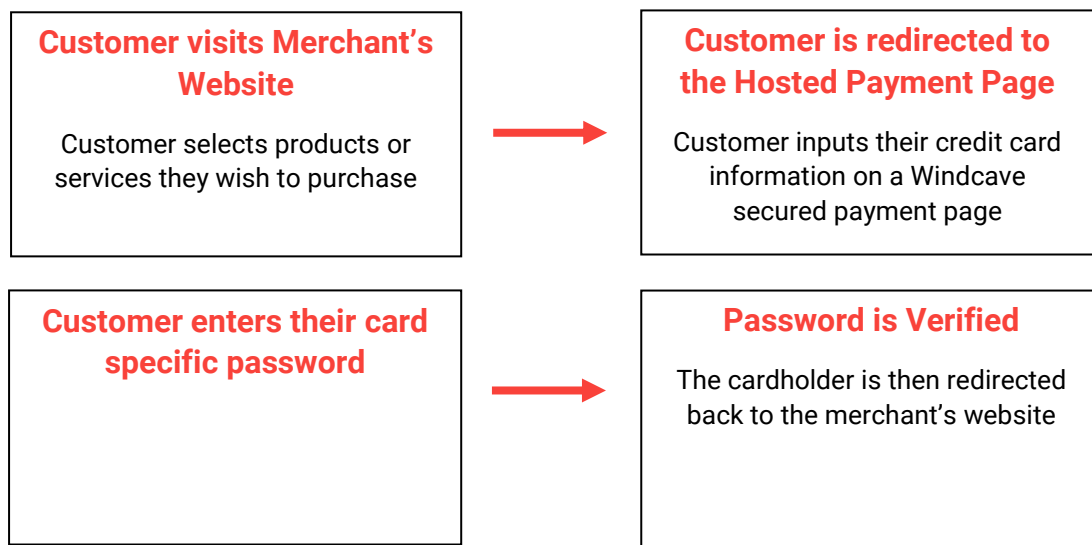
These latest schemes allow merchants to enjoy:

- Fraud reduction
- Reduced liability
- Reduced operational costs
- Increase sales and
  marketing opportunies

These schemes allow merchants' to prove to their potential customers that they are serious about protecting the identity and integrity of their customers' sensitive card details. In doing so, the merchant can potentially increase sales generated through their web services, by converting 'window-shoppers' to 'shoppers'.

These schemes require the card holder to enter a password (only if the card holder is enrolled) as an additional step to entering their credit card number and expiry date that is verified by the issuing bank. Once the issuing bank has verified the password, the user is then redirected back to the merchant's web site.

Customer visits Merchant's Website

Customer selects products or services they wish to purchase

Customer is redirected to the Hosted Payment Page

Customer inputs their credit card information on a Windcave secured payment page

Customer enters their card specific password

Password is Verified

The cardholder is then redirected back to the merchant's website

Windcave have implemented a MPI (Merchant Plug-In) that securely allows for these schemes to be implemented with no development cost or hassle to the merchant whatsoever.

The MPI is compliant with the latest 3D secure technology and Visa estimates that with these new security mechanisms in place, merchants will experience an increase in sales generated and lower operational costs – having to deal with fewer charge backs.

15 Troubleshooting & FAQs

Q − "I have set the response pages URLs in my transaction request form, but I'm not getting redirected back to the pages I asked for."

A - You need to use a fully qualified address for example http://www.yourwebsite.com/response.xxx. You cannot just use response xxx etc.

Q − "How do I get my logo to show on the Payments Page?"

A - To upload a merchant logo to your payments page, please email support@windcave.com quoting your PX Pay 2 Id in the email.

Q - "What happens if the customer doesn't click next on the payment page and closes their window, how will the response message get back to my success/failure pages?"

A - There is a background post as well that sends the response to your site that you requested and the customer has also seen the response by that time. Clicking the next button to return to your site will also send a redundant post.

Q - "Why do I get an error page stating that the 'Root element is missing'?"

A - When the URL is provided as part of an XML document certain characters such as the ampersand must be escaped in order to maintain the integrity of the document. Before directing the browser to the URL provided the escaped characters should be unescaped.

Q − "Why can't Internet Explorer display my success or failure pages?"

A - Internet Explorer limits URLs to 2083 characters, above which it will display a page stating that 'Internet Explorer cannot display the webpage'. The lengths of the query strings used on the payment pages and of those appended to the success and failure URLs can sometimes cause the URLs to exceed the maximum allowable in Internet Explorer. This will generally only occur when large amounts of transaction data are being passed to Windcave and are subsequently encrypted and become part of the query string. Please contact the Windcave support team to ask about URL compression or reduce the amount of transaction data being passed in optional fields.

Q − "I am unable to process Japanese currency, every time I submit the transaction request I receive the error code "IU". I have no problems processing in NZD and AUD."

A - When submitting transactions for currencies with no decimal division of units such as the Japanese Yen (JPY), the AmountInput must be in an appropriate format e.g. "1000" for 1000 Yen. If "1000.00" is submitted in AmountInput with CurrencyInput set to JPY, an error will occur and response code "IU" will be returned.

Q − "Can I process refunds via the Hosted Payments Page?"

A - No, the HPP solution is designed for initial capture of card data only. Refunds can be done via Payline or our merchant hosted API solutions (PX Post, Web Service, DPSAuth SSL).

Windcave

16 Request Response Codes

During development if the URI is not received after a transaction request you may receive a Request XML document (response data) with children elements Reco and ResponseText from Windcave. The Reco (Response code) and ResponseText provides an indication of the error (see an **example of the XML below**). Common error codes and the potential cause for the error are listed below. Please note a Reco may be received if the integration is not receiving the hosted payment page URI. To debug a possible request issue, please ensure your integration is logging the response data after a GenerateRequest is sent to Windcave.

Please note these error codes are related to the functionality of PxPay 2.0 interface itself and are different to the ReCo related to the host, bank or acquirer that you receive in the final transaction response XML document.

```
<Request valid="1">

<Reco>IP</Reco>

<ResponseText>Invalid Access Info</ResponseText>

</Request>
```

Please note that error codes are subject to change and under no circumstances should you base any application logic on error codes returned. If you cannot identify the issue from the list above please contact support@windcave.com

Code	Description
IC	Invalid Key or Username. Also check that if a TxnId is being supplied that it is unique.
ID	Invalid transaction type. Ensure that the transaction type is either Auth or Purchase.
IK	Invalid UrlSuccess. Ensure that the URL being supplied does not contain a query string.
IL	Invalid UrlFail. Ensure that the URL being supplied does not contain a query string.
IM	Invalid PxPayUserId.
IN	Blank PxPayUserId.

IP	Invalid Access Info. Ensure the PxPayID and/or key are valid.
IQ	Invalid TxnType. Ensure the transaction type being submitted is either "Auth" or "Purchase".
IT	Invalid currency. Ensure that the CurrencyInput is correct and in the correct format e.g. "USD".
IU	Invalid AmountInput. Ensure that the amount is in the correct format e.g. "20.00".
NF	Invalid Username.
NK	Request not found. Check the key and the mcrypt library if in use.
NL	User not enabled. Contact Windcave.
NM	User not enabled. Contact Windcave.
NN	Invalid MAC.
NO	Request contains non ASCII characters.
NP	Closing Request tag not found.
NQ	User not enabled for PxPay 2.0. Contact Windcave.
NT	Key is not 64 characters.

17 Go Live

When testing is done and you are ready to start accepting live payments we will link your Windcave account to your bank/acquirer merchant account(s).

In order for Windcave to process live payments, please complete the following steps:

- Register for the Windcave software at https://sec.windcave.com/pxmi3/apply
- Once applied; a Windcave consultant will make contact with you and explain the setup procedure and put you in contact with your chosen bank.
- You will need to apply for an "ecommerce merchant account" with your bank and specify that payments will be processed through "Windcave".
- Once you have completed the process with the bank, they will issue you with a merchant ID / Merchant number. Windcave will need this in order to activate your account for live payments.
- Once Windcave have been provided with the Merchant ID and confirmation that your website is ready to go live; we can activate your payment gateway with 3 working days' notice.

If unsure about any step above, please contact Windcave on the details provided below for your region: http://www.windcave.com/contact