



Express Checkout V4.3

Express Check Out Integration Guide
Version 4.3



Revision History

Version	Description of Change	Date
1.0	- New Document	24/03/2015
1.1	- Added Section: Troubleshoot your Installation	05/11/2015
1.2	- Added new parameter: country_code - Amended statement, only shipping details can be hidden, not billing details	07/07/2015
1.3	- Added Express Checkout Screen Shot and Diagram	15/09/2015
2.0	- Added shipping first name and shipping last name parameters	09/02/2016
2.1	- Changed formatting of Document - Changed "Troubleshooting" section to "Frequently Asked Questions" - Added FAQ (2 to 7) - Added "Verify Transaction" API Call - Updated Transaction Reports API - Added IPN section - Added API for Refunding Transactions - Added section "Payment Statuses" under Verify Payment	04/04/2016
3.0	- Option to use Custom CSS File - Option for redirect URL on rejected payment - Option to remove shipping sections from Express Checkout display. - Option to remove billing sections from Express Checkout display. - Option to remove customer info sections from Express Checkout display. - Optional Secure Sign Check - Option to use custom "Pay Now" button	26/04/2016
3.1	- Corrected verify_payment_transaction request / response variables.	05/10/2016
3.2	- Corrected country code for UAE in format examples.	28/12/2016
3.3	- Added IPN sample responses	06/04/2017
3.4	- Added Screenshot for Secret Key	07/05/2017
3.5	- Updated secure_sign code	29/05/2017
3.6	-Updated FAQ for Express checkout -Added FAQs for Express checkout-Iframe -Changed transaction_id parameter to Required	28/08/2017
3.6.1	-Update Test Cards Expiry Date	20/02/2018
4.0	-Added Tokenization. -Updated Response codes and sample code. -Added Token parameters on Return Url	18/04/2018



4.1	<ul style="list-style-type: none">- Added New Refund API response for successful refund processed- Added New IPN response for rejected refund request- Added url_cancel and is_canceled_pt param for express checkout	04/06/2018
4.2	<ul style="list-style-type: none">- Add new section for processing MADA transactions	09/08/2018
4.3	<ul style="list-style-type: none">- Add new section for processing pre-auth transactions- Add new variable is_self in section 3.1- Update IPN Sample responses	08/10/2018



Table of Contents

1	OVERVIEW	5
1.1	TEST MERCHANT ACCOUNT v/s LIVE MERCHANT ACCOUNT.....	5
1.2	TARGET AUDIENCE	5
1.3	ASSISTANCE	5
1.4	DOCUMENTATION FEEDBACK	5
1.5	EXPRESS CHECKOUT PAYMENT FLOW.....	6
1.6	PREREQUISITES FOR EXPRESS CHECKOUT	6
1.7	HIGHLY RECOMMENDED SETTINGS	6
1.7.1	HTTPS over HTTP.....	6
1.7.2	Hosting of JavaScript file.....	6
2	CONFIGURING THE BUTTON CODE FOR PAYTABS EXPRESS CHECKOUT	6
3	BUTTON CODE FOR EXPRESS CHECKOUT.....	8
3.1	SETTINGS.....	8
3.1.1	iFrame.....	11
3.1.2	CSS Customization	11
3.2	CUSTOMER INFORMATION	12
3.3	BILLING ADDRESS	13
3.4	SHIPPING ADDRESS.....	14
3.5	CHECKOUT BUTTON.....	15
3.6	PAY BUTTON.....	15
3.7	SAMPLE BUTTON CODE FOR MINIMAL EXPRESS CHECKOUT	16
3.8	SAMPLE CODE WITH ALL MANDATORY AND OPTIONAL PARAMETERS:	17
3.9	POST VARIABLES ON RETURN URL	19
3.9.1	is_canceled_pt	20
3.9.2	Secure Sign.....	20
4	VERIFY PAYMENT	23
4.1	REQUEST	23
4.2	RESPONSE	23
4.3	RESPONSE CODES.....	24
4.4	PAYMENT STATUSES	25
5	TOKENIZATION.....	27
5.1	SAMPLE CODE WITH ALL MANDATORY AND OPTIONAL PARAMETERS:	27
6	PROCESS PAYMENT WITH EXISTING TOKEN	29
6.1	REQUEST	29
6.2	RESPONSE	30
6.3	RESPONSE CODES.....	31
7	PROCESS MADA TRANSACTIONS.....	32
7.1	SAMPLE CODE FOR MADA TRANSACTIONS	32
8	REFUND TRANSACTIONS	33
8.1	REQUEST	36
8.2	RESPONSE	36
8.3	RESPONSE CODES.....	37
9	PAYTABS IPN (INSTANT PAYMENT NOTIFICATION)	37
9.1	HOW IPN WORKS	37



9.2	SET UP YOUR IPN LISTENER.....	39
9.3	IPN EVENTS.....	39
9.4	CONTENTS / DATA OF THE IPN	39
9.5	RESPONSE CODES.....	41
9.6	SAMPLE IPN RESPONSES	41
9.6.1	<i>IPN sample response for successful express checkout transactions</i>	<i>41</i>
9.6.2	<i>IPN sample response for rejected express checkout transactions</i>	<i>42</i>
9.6.3	<i>IPN sample response for rejected express checkout transactions by fraud monitoring</i>	<i>43</i>
9.6.4	<i>IPN sample response for force accepted express checkout transactions.....</i>	<i>43</i>
9.6.5	<i>IPN sample response for refunded express checkout transactions.....</i>	<i>43</i>
10	TRANSACTIONS REPORTS.....	45
10.1	REQUEST	45
10.2	RESPONSE	45
10.3	RESPONSE CODES.....	47
10.4	SAMPLE RESPONSE.....	48
11	TEST CREDIT CARD NUMBERS	49
12	EXPRESS CHECKOUT SCREEN SHOTS.....	50
13	FREQUENTLY ASKED QUESTIONS	52
	FAQ -EXPRESS CHECKOUT – IFRAME	54



1 Overview

PayTabs Express Checkout is a solution that provides an efficient checkout process for online shoppers which keeps them on the merchant's website while making a payment and even after the payment is complete.

It has an option to collect the information from the checkout form present on the merchant's website like personal information, billing address and shipping address. If your website does not support payments that include shipping address you can customize the Express Checkout form to remove/hide these fields. But billing address is mandatory. Moreover, you can even use your own button image/design for the express Checkout button on your website.

1.1 Test Merchant Account v/s Live Merchant Account

PayTabs does not offer an explicit Sandbox / Testing environment. But by using a demo account, you can freely test all the functionalities and integrate our API and Plugin without any hassle. In order to move to LIVE PRODUCTION environment, you will need to activate your LIVE MERCHANT ACCOUNT by clicking on GO LIVE through your PayTabs Merchant Account. PayTabs requires some personal information, documents and other details related to your business in order to validate your business. Once your Merchant Account is activated, you will have been moved to the LIVE PRODUCTION environment.

1.2 Target Audience

This document is intended for developers who want to write applications that interact with PayTabs.com.

This document assumes that you understand general networking and programming ideas. The Express CheckOut supports only PHP platforms and is not intended for use with E-commerce Platforms.

1.3 Assistance

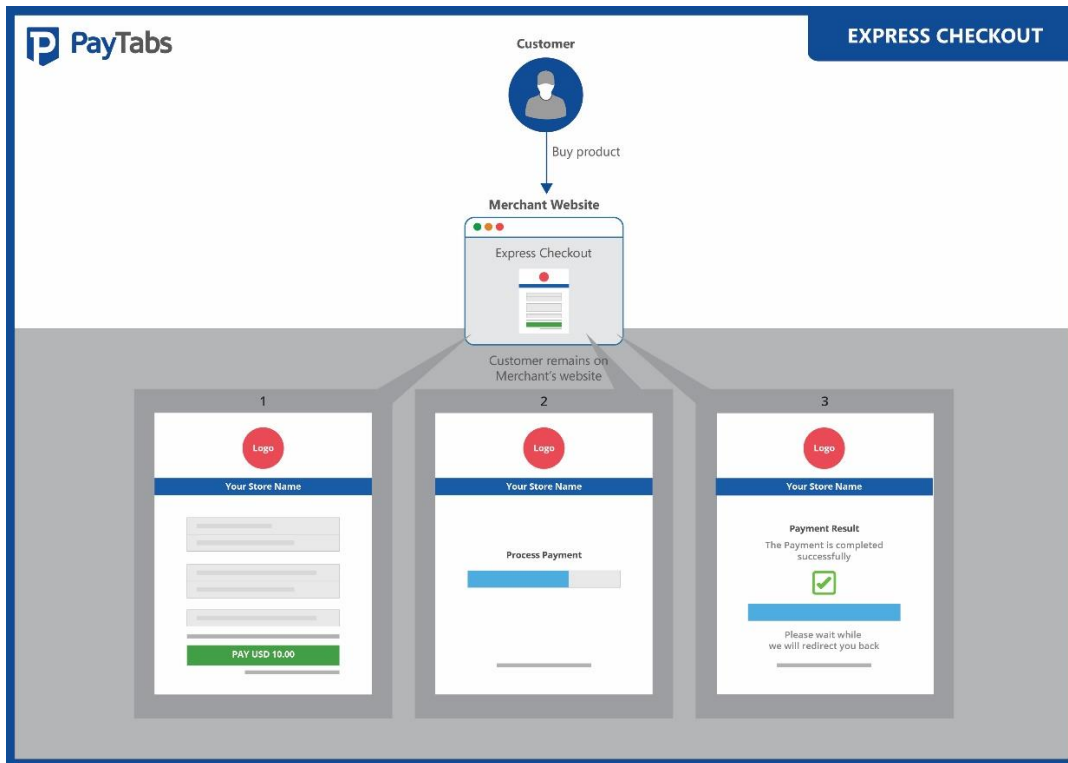
If you require assistance, please check out our [Frequently Asked Questions](#) on the website or the [Frequently asked questions](#) at the end of the document or email our technical staff at support@paytabs.com.

1.4 Documentation Feedback

Help us improve the PayTabs Express CheckOut documentation, by sending your feedback to support@paytabs.com.



1.5 Express Checkout Payment Flow



1.6 Prerequisites for Express CheckOut

There are no mandatory prerequisites but to include a custom CSS file, it must be hosted on a secure HTTPS connection.

1.7 Highly recommended Settings

1.7.1 HTTPS over HTTP

All submissions of payment info using Express CheckOut are made via a secure HTTPS connection. However, in order to protect yourself from certain forms of man-in-the-middle attacks, we suggest that you also serve the page containing the payment form with HTTPS as well. This means that any page that an Express CheckOut form may exist on should start with `https://` rather than just `http://`.

1.7.2 Hosting of JavaScript file

PayTabs JavaScript file should always be hosted on `paytabs.com`. If PayTabs makes any changes or updates in the JavaScript file, you will not receive the updates which will result in your transactions being considered as fraudulent and consequently they will be rejected.

2 Configuring the Button Code for PayTabs Express Checkout

PayTabs Express CheckOut V 3.0 has various options to allow you as the merchant to adapt the Express CheckOut solution to meet your exact requirements.



The Express CheckOut has some mandatory fields that need to be entered and the other fields are optional.

1. Go to <https://www.paytabs.com/developers> again, then go to Express CheckOut V 3.0 and copy the button code.
2. Copy the button code to the place where you want the checkout button.
3. Set up the variables according to your requirement. Do not comment any field in the button code. If you do not wish to use a certain field, kindly leave it blank.
4. You can use the test cards, to test a Transaction on your DEMO merchant account. Test cards will only give successful payment results, if your merchant account is in DEMO or TEST mode. Please do not use test cards on your LIVE PayTabs Merchant Account. This may cause your Merchant Account to be blocked or suspended.
5. Please remember that CheckOut is a FINAL step. Express checkout cannot be used for multiple products placed in the same page. It should be used as a final payment step after the checkout.



3 Button Code for Express CheckOut

PayTabs require that your website where you are integrating express checkout should be the same as the website/URL registered in your PayTabs Merchant Account. The merchant can see this SITE URL on his/her dashboard, under “My Profile”.

For Demo Users:

You can edit your site URL by clicking going to “My Profile” and clicking on edit, enter your correct site URL and click on Save.

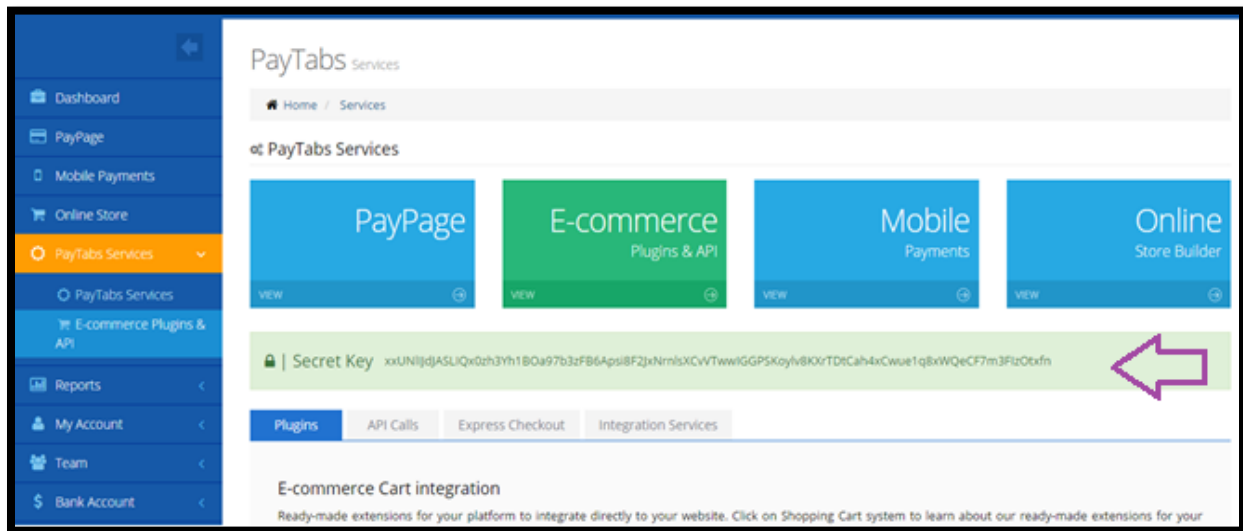
For Live Merchants:

You can use the website that you have submitted in the Go-Live application. If you need to edit/change the site URL, you can send a request to account manager.

3.1 Settings

The following button code should be placed on the page where you want the CheckOut button to appear.

Fig: To find your secret key, go to PayTabs Services >> E-commerce Plugins and API.



The button code has various parameters that are mandatory to achieve a successful payment via PayTabs. These parameters are passed through the settings object.



<i>Element</i>	<i>Description</i>	<i>Format</i>
merchant_id	<u>Required</u> Merchant ID found at the top-right corner of the Merchant Dashboard.	String E.g.: 100001646
secret_key	<u>Required</u> SECRET Key found on the merchant dashboard > Mobile Payment > Secret Key	String E.g.: <i>Mao8sdhasfyahosdap9sd</i>
amount	<u>Required</u> Amount of the products & other charges	Decimal Up to 3 Decimal places E.g.: 123.399
currency	<u>Required</u> Currency of the amount stated. 3 character ISO currency code	String E.g.: <i>BHD for Bahrain</i> <i>SAR for Saudi</i> <i>AED for UAE</i>
title	<u>Required</u> Name of the customer who requested the order	String 32 characters E.g.: <i>Mr. John Doe</i>
product_names	<u>Required</u> Product names linked to this order. For multiple products use comma, separated products	String 170 characters E.g.: <i>IPhone, Samsung S5, Samsung S4</i>
order_id	<u>Required</u> Order ID generated on merchant's Website.	String 40 characters E.g.: <i>Abc-5566</i>
url_redirect	<u>Required</u> Call back URL, after payment is successful. (or rejected, if enabled)	String E.g.: <i>http://yourwebsite.com/</i>
url_cancel	<u>Optional</u> In case the user clicks the close buttons of the express checkout popup and cancels the payment, the order_id and secure_sign will be posted in "url_cancel" parameter	String E.g.: <i>http://yourwebsite.com/cancel_url</i>



is_self	<u>Optional</u> If you wish to not be redirected on successful payment to redirect_url and remain on the same page you can pass this parameter as 1.	E.g: 1
display_billing_fields	<u>Optional</u> By default, value is 1. If you are already collecting this information on your website, then you can hide it, but you must pass it to PayTabs in the background through the express checkout code in order for your payment to be processed.	String E.g.: 0 or 1
display_shipping_fields	<u>Optional</u> By default value is 1. If your website does not require shipping then set this parameter to 0.	String E.g.: 0 or 1
display_customer_info	<u>Optional</u> By default value is 1. If you are already collecting this information on your website, then you can hide it, but you must pass it to PayTabs in the background through the express checkout code in order for your payment to be processed	String E.g.: 0 or 1
language	<u>Optional</u> Default value is en for English. If you wish to set it to Arabic, use ar.	String E.g.: en or ar
redirect_on_reject	<u>Optional</u> In case you wish to redirect the user to url_cancel even if the payment is rejected then set this value to 1. By default it is 0, which means the customer is not redirected in case the payment is rejected.	String E.g.: 0 or 1



3.1.1 iFrame

If you wish to use the express checkout like an iframe instead of pop up then pass the object `is_iframe` inside settings object.

Object name is **is_iframe**. The parameters are:

<i>Element</i>	<i>Description</i>	<i>Format</i>
load	<u>Required</u> This option defines the iframe load type. Two values can be passed. 'onbodyload' if merchant wishes to load iframe on page load. onbuttonclick if merchant wishes to load iframe on checkout button click.	String E.g.: <i>onbodyload</i> or <i>onbuttonclick</i>
show	<u>Required</u> Set value as 1 in order to use express checkout like an iframe. By default it is 0.	String E.g.: 0 or 1

3.1.2 CSS Customization

If merchant wishes to pass their own CSS then use the object style inside settings object and set the following options which are mandatory. If you want to include the custom CSS file then it must be hosted on a secure HTTPS connection.

Object name is **style**. The parameters are:

<i>Element</i>	<i>Description</i>	<i>Format</i>
css	<u>Required</u> Set the value to "custom"	E.g.: <i>custom</i>
linktocss	<u>Required</u> The URL of the CSS file hosted on an SSL Connection	E.g.: <i>https://www.YOURSTORE.com/css/style.css</i>



3.2 Customer Information

In case `display_customer_info` is 1, then passing this object will pre-fill customer information section for the Express Checkout form on your website. This option is mandatory in the code in case you set `display_customer_info` to 0. Otherwise your payment will not be processed.

Object name is **customer_info**. The parameters are:

<i>Element</i>	<i>Description</i>	<i>Format</i>
<code>first_name</code>	<u>Required</u> First Name of the Customer	String 32 characters E.g.: <i>John</i>
<code>last_name</code>	<u>Required</u> Last Name of the Customer	String 32 characters E.g.: <i>Doe</i>
<code>country_code</code>	<u>Required</u> international dialing code for Phone Number of the Customer Please enter country code without '0' or '+'	String E.g.: <i>973 for Bahrain</i> <i>971 for UAE</i> <i>966 for Saudi</i>
<code>phone_number</code>	<u>Required</u> Phone Number of the Customer	String Minimum 6 numbers up to 15 numbers E.g.: <i>3312345678</i>
<code>email_address</code>	<u>Required</u> Email of the customer	String 32 characters E.g.: <i>customer@domain.com</i>



3.3 Billing Addresss

This option is mandatory in the code in case you set `display_billing_details` to 0. Otherwise your payment will not be processed

In case `display_billing_details` is 1 then passing this object will pre-fill billing address for the Express Checkout form on your website.

Object name is **billing_address**. The parameters are:

Options	Description	Format
full_address	<u>Required</u> Complete Address of the customer. Multiple address lines will be merged into one single line.	String 40 characters E.g.: <i>Flat 11 Building 222 Block 333 Road 444 Manama Bahrain</i>
city	<u>Required</u> Name of the billing city selected by customer	String 50 characters E.g.: <i>Manama</i>
state	<u>Required</u> Billing State entered by the customer	String 32 characters E.g.: <i>Manama</i> 2 characters for US and Canada E.g.: if state is New York then the state field should be NY
country	<u>Required</u> Country of the customer	String 3 character ISO country code E.g.: <i>BHR for Bahrain</i> <i>ARE for UAE</i> <i>SAU for Saudi Arabia</i>
postal_code	<u>Required</u> Billing Postal code provided by the customer.	String From 5 to 9 characters E.g.: <i>12345</i>



3.4 Shipping Address

In case `display_shipping_details` is 1 then passing this object will pre-fill shipping address for the Express Checkout form on your website. This option enables you to prefill the shipping address for the Express Checkout form on your website. This option is mandatory in the code in case you set `display_shipping_details` to 0. Otherwise your payment will not be processed.

Object name is **shipping_address**. The parameters are:

Options	Description	Format
<code>shipping_first_name</code>	<u>Required</u> First Name of the customer.	String 40 characters E.g.: "Jane"
<code>shipping_last_name</code>	<u>Required</u> Last Name of the customer.	String 40 characters E.g.: "Abdulla"
<code>full_address_shipping</code>	<u>Required</u> Complete address of the customer. Multiple address lines will be merged into one single line.	String 40 characters E.g.: Flat 11 Building 222 Block 333 Road 444 Manama Bahrain
<code>city_shipping</code>	<u>Required</u> Name of the billing city selected by customer	String 50 characters E.g.: Manama
<code>state_shipping</code>	<u>Required</u> Billing State entered by the customer	String 32 characters E.g.: Manama 2 characters for US and Canada E.g.: if state is New York then the state field should be NY
<code>country_shipping</code>	<u>Required</u> Country of the customer	String 3 character ISO country code E.g.: BHR for Bahrain ARE for UAE SAU for Saudi Arabia
<code>postal_code_shipping</code>	<u>Required</u> Shipping Postal code provided by the customer	String From 5 to 9 characters E.g.: 12345



3.5 Checkout Button

This option enables you to use your own checkout button image and design for the Express Checkout buttons on your website otherwise the default button image from PayTabs is used.

Object name is **checkout_button**. The parameters are:

<i>Element</i>	<i>Description</i>	<i>Format</i>
width	Define the width of the image you are using for your button. By default the width is set to 150px.	E.g.: 150
height	Define the height of the image you are using for your button. By default the height is set to 30px.	E.g.: 30
img_url	Define the URL of the image that you want to use as your button. By default, the image is retrieved from PayTabs.	E.g.: https://www.mywebsite.com/mypaybutton

3.6 Pay Button

This option enables you to use your own button image and design for the Express Checkout pay button on your website, otherwise the default button image from PayTabs is used.

Object name is **pay_button**. The parameters are:

<i>Element</i>	<i>Description</i>	<i>Format</i>
width	Define the width of the image you are using for your button. By default the width is set to 150px.	E.g.: 150
height	Define the height of the image you are using for your button. By default the height is set to 30px.	E.g.: 30
img_url	Define the URL of the image that you want to use as your button. By default, the image is retrieved from PayTabs.	E.g.: https://www.mywebsite.com/mypaybutton



3.7 Sample Button Code for Minimal Express CheckOut

```
<link rel="stylesheet" href="https://www.paytabs.com/theme/express_checkout/css/express.css">
<script src="https://www.paytabs.com/theme/express_checkout/js/jquery-1.11.1.min.js"></script>
<script src="https://www.paytabs.com/express/express_checkout_v3.js"></script>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
<!-- Button Code for PayTabs Express Checkout -->
<div class="PT_express_checkout"></div>
<script type="text/javascript">
  Paytabs("#express_checkout").expresscheckout({
    settings:{
      merchant_id: "11009133",
      secret_key: "vO54JlCQSEOrZcccZ7fccPwpdvQcUSBJoGN7",
      amount : "10.00",
      currency : "USD",
      title : "Mr. John Doe",
      product_names: "Product1,Product2,Product3",
      order_id: 25,
      url_redirect: "http://mywebsite.com/paymentresult ",
      display_customer_info:1,
      display_billing_fields:1,
      display_shipping_fields:0,
      language: "en",
      redirect_on_reject: 0,
    }
  });
</script>
```



3.8 Sample code with all mandatory and optional parameters:

```
<link rel="stylesheet" href="https://www.paytabs.com/theme/express_checkout/css/express.css">
<script src="https://www.paytabs.com/theme/express_checkout/js/jquery-1.11.1.min.js"></script>
<script src="https://www.paytabs.com/express/express_checkout_v3.js"></script>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
<!-- Button Code for PayTabs Express Checkout -->
<div class="PT_express_checkout"></div>
<script type="text/javascript">
  Paytabs("#express_checkout").expresscheckout({
    settings:{
      merchant_id: "11009133",
      secret_key: "vO54JlCQSEOrZcccZ7fccPwpdvQcUSBJoGN7",
      amount : "10.00",
      currency : "USD",
      title : "Mr. John Doe",
      product_names: "Product1,Product2,Product3",
      order_id: 25,
      url_redirect: "http://mywebsite.com/paymentresult ",
      url_cancel: "http://mywebsite.com/cancel ",
      display_customer_info:1,
      display_billing_fields:1,
      display_shipping_fields:0,
      language: "en",
      redirect_on_reject: 1,
      style: {
        css: "custom",
        linktocss: "https://www.YOURSTORE.com/css/style.css",
      },
      is_iframe: {
        load: "onbuttonclick", //onbodyload
        show: 1,
      },
    },
    customer_info:{
      first_name: "John",
      last_name: "Smith",
      phone_number: "5486253",
      email_address: "john@test.com",
      country_code: "973"
    },
    billing_address:{
      full_address: "Manama, Bahrain",
      city: "Manama",
      state: "Manama",
      country: "BHR",
    }
  });
}
```



```
    postal_code: "00973"
  },

  shipping_address:{
    shipping_first_name: "Jane",
    shipping_last_name: : "Abdulla",
    full_address_shipping: "Manama, Bahrain",
    city_shipping: "Manama",
    state_shipping: "Manama",
    country_shipping: "BHR",
    postal_code_shipping: "00973"
  },
  checkout_button:{
    width: 150,
    height: 30,
    img_url: "https://www.YOURSTORE.com/image/yourimage.jpg"
  }
  pay_button:{
    width: 150,
    height: 30,
    img_url: "https://www.YOURSTORE.com/image/yourimage.jpg"
  }
  });
</script>
```



3.9 Post variables on return URL

The PayTabs Express CheckOut V 3.0, returns certain variables as entered by the customer, back to the merchant's website to assist the merchant in order fulfillment and delivery.

The post variables can be accessed on the redirect URL by using the following command:

print_r (\$_REQUEST) or print_r (\$_POST)

The POST variables returned on the redirect URL are as follows:

<i>POST Variables</i>	<i>Description</i>	<i>Format</i>
order_id	Order ID as per the merchant's website.	alphanumeric
transaction_id	PayTabs Transaction ID	integer
response_code	PayTabs response code for transaction completion	integer
response_message	PayTabs description for the response code	string
transaction_amount	Amount of the transaction	integer
transaction_currency	Currency of the transaction	3 character ISO currency code
customer_name	Full customer name	string
customer_email	Email of the customer	string
customer_phone	Mobile Number of the customer	string
last_4_digits	Last 4 digits of the credit card used by the customer	integer
first_4_digits	First 4 digits of the credit card used by the customer	integer
card_brand	Brand of the credit card used by the customer i.e. Visa, MasterCard etc.	string
trans_date	Date & time of the transaction. The format is DD-MM-YYYY H:M:S AM/PM	string
secure_sign	A secure sign key in order for merchant to validate if the payment is returning from PayTabs	alphanumeric

If display_shipping_fields is set to 1, then there will be POST variables in addition to the above, as follows:



<i>POST Variables</i>	<i>Description</i>
shipping_first_name	First name as entered by the customer.
shipping_last_name	Last name as entered by the customer.
address_shipping	Full shipping address as entered by the customer.
city_shipping	Shipping city as entered by the customer.
state_shipping	Shipping state as entered by the customer.
country_shipping	Shipping country as entered by the customer.
postal_code_shipping	Shipping postal code as entered by the customer.

In order to see which items have been purchased, you will have to save the products on your own server, generate an order id for reference and pass it through the express checkout code. We will return a transaction id against that order id. Other transaction related details will be available in the Merchant Dashboard.

3.9.1 is_canceled_pt

In case the user closes the popup during the transactions and "url_cancel" parameter is not passed in the merchant's Express CheckOut button script, a query param is_canceled_pt=1 is passed in the return link

3.9.2 Secure Sign

Secure sign is calculated based on the parameters sent in the response in order to verify the payment is coming from PayTabs. Merchant can calculate secure sign at their end and verify. This step is optional.

The following parameters are passed as an array to the function named **create_secure_hash()**.

- order_id
- response_code
- customer_name
- transaction_currency
- last_4_digits
- customer_email

The default secure sign key is '**secure@paytabs#@aes11%%**'. The merchant will be able to set their own key from the merchant dashboard > My Profile.





To use create the secure sign on your own, you can use the following code:

```
$params = array('order_id' => 123456,
                'response_code' => 100,
                'customer_name' => "name of customer",
                'transaction_currency' => "AED",
                'last_4_digits' => "0002",
                'customer_email' => "test@email.com"
                );

function create_secure_hash($params = array()) {
    $string = ""; // SHA phrase
    $shain_phrase = secure@paytabs#@aaes11%% //This is the default value
    ksort($params); // Sorting the params

    foreach ($params as $keys => $values)
    {
        $string .= strtoupper($keys) . '=' . $values . $shain_phrase;
        // Combining the field values in one string
    }

    // SHA Converting here
    $secure_sign = sha1($string); // Converting the values to SHA1 format
    return $secure_sign;
}
```



4 Verify Payment

This method verifies the result of the payment. When a payment is processed, you will receive a transaction id in the response. When the customer completes a payment and is redirected back to your app, the transaction_id is used to verify the status of the payment whether it is a successful transaction or a failed transaction.

PATH	https://www.paytabs.com/apiv2/verify_payment_transaction
REQUEST METHOD	POST
PRODUCTION	LIVE

4.1 Request

<i>Element</i>	<i>Description</i>	<i>Format</i>
merchant_email	<u>Required</u> Merchant email that you use to sign up and/or login into PayTabs Merchant Dashboard	String Up to 128 characters E.g.: foof@foo.com
secret_key	<u>Required</u> SECRET Key generated by merchant dashboard	String Up to 128 characters E.g.: Mao8sdhasfyahosdap9sd
transaction_id	<u>Required</u> This is the transaction_id that is returned in the response of the payment.	String Up to 12 characters E.g.: 130017
order_id	<u>Optional</u> This is the order id from your system, you can use that instead of transaction_id parameter	32 characters E.g.: 4578

4.2 Response

<i>Element</i>	<i>Description</i>	<i>Format</i>
result	The string representation for result, whether transaction has succeeded or failed and the reason why your transaction failed.	String E.g.: Payment is completed.



response_code	The response code / error code of the API call. A complete list of response codes can be found in Merchant Dashboard > Help > Reason Codes	String Up to 4 characters E.g.: 4002
pt_invoice_id	The invoice ID/PayPage ID of the transaction. This is available only for PayPage and API integration. It will be NULL in case of mobile SDK	String NULL E.g.: NULL
amount	The transaction amount.	Decimal Up to 3 Decimal places E.g.: 123.399
currency	The currency of the transaction amount.	String 3 character ISO country code E.g.: BHR
transaction_id	The ID of the latest transaction that occurred for this invoice	String Up to 12 characters E.g.: 90094
order_id	The order id from your system to be able to map your order's to PayTabs transactions	32 characters E.g.: 4578

4.3 Response Codes

Error Code	Description
4001	Missing parameters
4002	Invalid Credentials
4003	There are no transactions available.
0404	You don't have permissions
100	Payment is completed Successfully.
481 or 482	This transaction may be suspicious, your bank holds for further confirmation. Payment Provider has rejected this transaction due to suspicious activity; Your bank will reverse the dedicated amount to your card as per their policy. Important Note: These transactions are rejected by our fraud monitoring tools; however the transactions are only authorized (Not



settled). In case you verify the transaction is genuine you can contact us to force accept the transaction after checking if it is not fraud and we might ask verification documents like ID card and front side of credit card used showing only name and last 4 digits, otherwise the transaction amount will be released back to the cardholder according to the issuing bank policy. In case you are using IPN, it will post the response once the transaction is force Accepted from our side

****** The Complete list of response codes can be found on the Merchant Dashboard > Help > Reason Codes

4.4 Payment Statuses

Each transaction at PayTabs can have one of the following statuses:

<i>Status</i>	<i>Description</i>	<i>Reason Code</i>
Payment Approved	Payment is Approved and Captured	100
Payment Authorized	Payment is Authorized	111
Payment Rejected	Payment is rejected	Refer to Reason code section for a list of all reason codes
Authorization Fully Captured	Payment is authorized and the fully captured	113
Authorization Partially Captured	Partial amount of the authorized amount is captured (i.e: This transaction can be captured again using capture API till the full amount is captured)	112
Authorization Expired	Authorization is expired because the authorization period is completed and the authorized amount will be released back to the card holder	114
Authorization Partially captured remaining expired	Partial amount of the authorized amount is captured, the remaining amount cannot be captured as the authorization period is complete and the remaining amount from authorization will be released back to the card holder	115
Authorization Fully reversed	The authorized amount is reversed and released back to the cardholder	116



Payment Rejected (by fraud monitoring tools)	Payment is rejected by our fraud monitoring tools; however, the transactions are only authorized (Not settled). In case the merchant can verify the transaction is genuine, the merchant needs to contact PayTabs to force accept the transaction after checking if it is not fraud and we might ask verification documents like ID card and front side of credit card used showing only name and last 4 digits, otherwise the transaction amount will be released back to the cardholder according to the issuing bank policy. In case you are using IPN, it will post the response once the transaction is force Accepted from our side	481 or 482
Payment Force Accepted	Payment is Approved by PayTabs Operations team, upon merchant request and after verifying it's not fraud. Note: In this case the Payment Status is only updated but the reason code remains the same	481 or 482
Payment refunded	For Payments that you submit a refund request for either from Merchant's dashboard or using API for submitting refund requests In case you are using IPN, it will post the response once the request is approved from our side Note: In this case the Payment Status is only updated but the reason code remains the same	



```
},  
  
  billing_address:{  
    full_address: "Manama, Bahrain",  
    city: "Manama",  
    state: "Manama",  
    country: "BHR",  
    postal_code: "00973"  
  },  
  
  shipping_address:{  
    shipping_first_name: "Jane",  
    shipping_last_name: "Abdulla",  
    full_address_shipping: "Manama, Bahrain",  
    city_shipping: "Manama",  
    state_shipping: "Manama",  
    country_shipping: "BHR",  
    postal_code_shipping: "00973"  
  },  
  
  checkout_button:{  
    width: 150,  
    height: 30,  
    img_url: "https://www.YOURSTORE.com/image/yourimage.jpg"  
  },  
  
  pay_button:{  
    width: 150,  
    height: 30,  
    img_url: "https://www.YOURSTORE.com/image/yourimage.jpg"  
  },  
});  
</script>
```

Note: Once the transaction is successful user can get the additional Token parameters on his return url as shown below, which can be used to process tokenized payments.

Array

```
(  
  [pt_customer_email] => XXXX@paytabs.com  
  [pt_customer_password] => XXXAX1  
  [pt_token] => K4vf2XXXXXXXXXXFaQ7rBBJTwttuvd  
)
```



6 Process Payment with Existing Token

This method uses an existing token for a returning customer to process payment. You will need to pass the details (token, customer email and customer password) of the token that you previously created to perform a transaction.

PATH	https://www.paytabs.com/apiv3/tokenized_transaction_prepare
REQUEST METHOD	POST
PRODUCTION	LIVE

6.1 Request

You need to pass all the parameters as stated below:

<i>Element</i>	<i>Description</i>	<i>Format</i>
merchant_email	<u>Required</u> Merchant email id used to login into PayTabs Merchant Dashboard.	String E.g.: testing@test.com
secret_key	<u>Required</u> SECRET Key generated by merchant dashboard	String E.g.: <i>Mao8sdhasfyahosdap9sd</i>
title	<u>Required</u> Description or title of the transaction done by the customer	String 32 characters E.g.: <i>Order # 3321</i>
cc_first_name	<u>Required</u> First Name of the Customer	String 32 characters E.g.: <i>John</i>
cc_last_name	<u>Required</u> Last Name of the Customer	String 32 characters E.g.: <i>Doe</i>
order_id	<u>Required</u> Merchant's Website order ID, this is same as the reference_no value posted	String 32 characters E.g.: <i>Dx2365</i>
product_name	<u>Required</u> Product title of the product.	String 256 characters E.g.: <i>SamsungS5</i>
customer_email	<u>Required</u> Email of the customer	String 32 characters E.g.: <i>customer@domain.com</i>



phone_number	<u>Required</u> Phone Number of the Customer	String 32 characters E.g.: 3312345678
amount	<u>Required</u> Amount of the products and other charges, it should be equal to: amount = (sum of all products' (unit_price * quantity)) + other_charges This field will be displayed in the invoice as the sub total field	Float Up to 3 Decimal places E.g.: 123.399
currency	<u>Required</u> Currency of the amount stated. 3 character ISO currency code	String E.g.: BHD
pt_token	<u>Required</u> This pt_token is received in the API post response after completing the payment; it will be redirected to return_url. While returning back to that URL, it will send a POST request	String E.g.: R7ANsPK1q91fv5Q0bmQ3
pt_customer_email	<u>Required</u> The customer email linked to the tokenization profile used, by default when the profile is created at the first successful payment, it will use customer_email value sent in the API to link it to the token	String E.g.: customer@domain.com
pt_customer_password	<u>Required</u> This pt_customer_password is received in the API post response after completing the payment; it will be redirected to return_url. While returning back to that URL, it will sen	String E.g.: 1q91fv5Q0b

6.2 Response

Element	Description	Format
result	The string representation for result, whether transaction has succeeded or failed and the reason why your transaction failed.	String E.g.: <i>Payment is completed.</i>



response_code	The response code / error code of the API call. Complete list of response codes can be found in Merchant Dashboard > Help > Reason Codes	String Up to 4 characters E.g.: 4002
pt_invoice_id	The invoice/paypage ID of the transaction.	String Up to 12 characters E.g.: 90094
amount	The transaction amount.	Float Up to 3 Decimal places E.g.: 123.399

6.3 Response Codes

Code	Description
4001	Missing parameters
4002	Invalid Credentials
0404	You don't have permissions
4091	There are no transactions available.
100	Payment is completed.
481 482	This transaction may be suspicious. If this transaction is genuine, please contact PayTabs customer service to enquire about the feasibility of processing this transaction.
Any response code other than 100, 481, 482	Any other response than 100, 482, 481 means that the transaction has been rejected and the rejection reason is the description of the corresponding response code



7 Process MADA Transactions

MADA card is a debit card issued by local banks in Saudi Arabia that allows a cardholder to access funds in their accounts. In order to accept MADA transactions, this service needs to be enabled to your merchant account. Please contact your account manager for further details

7.1 Sample code for MADA transactions

```
<link rel="stylesheet" href="https://www.paytabs.com/theme/express_checkout/css/express.css">
<script src="https://www.paytabs.com/theme/express_checkout/js/jquery-1.11.1.min.js"></script>
<script src="https://www.paytabs.com/express/express_checkout_v3.js"></script>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
<!-- Button Code for PayTabs Express Checkout -->
<div class="PT_express_checkout"></div>
<script type="text/javascript">
  Paytabs("#express_checkout").expresscheckout({
    settings:{
      merchant_id: "11009133",
      secret_key: "vO54]ICQSEOrZcccZ7fccPwpdvQcUSBJoGN7",
      amount : "10.00",
      currency : "SAR",
      title : "Mr. John Doe",
      product_names: "Product1,Product2,Product3",
      order_id: 25,
      url_redirect: "http://mywebsite.com/paymentresult ",
      display_customer_info:1,
      display_billing_fields:1,
      display_shipping_fields:0,
      language: "en",
      redirect_on_reject: 0,
      mada: {
        is_mada: 1
      }
    }
  });
</script>
```



8 Pre-Authorized transactions

The Authorization feature will allow merchants to authorize transactions for a certain period of days. During this period, merchants will be able to capture the authorized transaction via multiple partial captures or fully capture the amount or void(reverse) it at any time until the authorization period ends where the transaction will be voided (reversed) automatically. You need to contact your account manager to enable this feature for your account.

8.1 Sample code for pre-authorized transactions

```
<link rel="stylesheet" href="https://www.paytabs.com/theme/express_checkout/css/express.css">
<script src="https://www.paytabs.com/theme/express_checkout/js/jquery-1.11.1.min.js"></script>
<script src="https://www.paytabs.com/express/express_checkout_v3.js"></script>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
<!-- Button Code for PayTabs Express Checkout -->
<div class="PT_express_checkout"></div>
<script type="text/javascript">
    Paytabs("#express_checkout").expresscheckout({
        settings:{
            merchant_id: "11009133",
            secret_key: "v054JlCQSEOrZcccZ7fccPwpdvQcUSBJoGN7",
            amount : "10.00",
            currency : "SAR",
            title : "Mr. John Doe",
            product_names: "Product1,Product2,Product3",
            order_id: 25,
            url_redirect: "http://mywebsite.com/paymentresult ",
            display_customer_info:1,
            display_billing_fields:1,
            display_shipping_fields:0,
            language: "en",
            redirect_on_reject: 0,
            is_preauth: 1,
        }
    });
</script>
```

8.2 Response code

The response_code and response_message returned as post variables in the return_url along other variables will be as follows:

[response_code] => 111

[response_message] => The transaction is authorized successfully



9 Capture, Partial Capture or Reverse authorized transactions

When you are ready to fulfill a customer's order and transfer funds from the customer's bank to your bank, capture the authorization for that order.

When fulfilling only part of a customer's order, you can capture only partial amount of the authorized payment. The full authorization reversal service releases the hold that the authorization placed on the customer's credit card funds. Use this service to reverse an unnecessary or undesired authorization only for an authorization that has not been captured (fully or partially) and settled.

This feature can be used only if the authorization period is not expired. Most authorizations expire within five to seven days, but the bank or company that issued the card decides how long an authorization lasts. Also, PayTabs determines for each merchant account the allowed authorization period. To know the authorization period assigned for your account please contact your account manager.

PATH https://www.paytabs.com/apiv3/release_capture_preauth

REQUEST METHOD POST

PRODUCTION LIVE

9.1 Request

<i>Element</i>	<i>Description</i>	<i>Format</i>
merchant_email	<u>Required</u> Merchant email from PayTabs Merchant Dashboard.	<i>String</i> E.g.: johndoemail.com
secret_key	<u>Required</u> SECRET Key generated by merchant dashboard	<i>String</i> E.g.: Mao8sdhasfyahosdap9sd
transaction_id	<u>Required</u> The ID of the latest transaction that occurred for this invoice	<i>String</i> Up to 12 characters E.g.: 90094
capture_amount	<u>Optional</u> Pass the amount you want to capture, you can either pass the full authorized amount or a partial amount Pass this value as 0 in case you want to void the transaction Note: You can do multiple partial captures till the full amount is captured	<i>E.g: 100</i>
void_transaction	<u>Optional</u> Pass this value as 1 in case you want to reverse the authorized transaction	<i>E.g: 1</i>

9.1.1 Response Parameters

<i>Element</i>	<i>Description</i>	<i>Format</i>
result	The String representation for result about the supplied data	<i>String</i> E.g.: Transaction has been partially



		captured
response_code	The response code of the processor, if the parameters are accepted.	<i>String</i> Min 3 characters E.g.: 100

9.1.2 Response Codes

Code	Description
112	Transaction has been partially captured
113	Transaction has been fully captured
116	Transaction has been fully voided successfully
4004	The selected transaction can't be captured due to one of those reason: Transaction doesn't exist. Transaction was not marked as pre-auth. Transaction was not authorized. Transaction already captured.
4005	The requested amount is bigger than the transaction amount and or captured amount
4006	Please choose amount as 0 if you want to void the transaction
4007	Can't void this transaction because it has partial capture!
4010	Expired Authorization
114	

10 Refund Transactions

This method enables you to submit the refund requests using the below API function. Refund requests will be processed in around 24 hours and if you are using IPN service the IPN will post a response once the refund request is approved or rejected. Partial refunds are available. You can submit a single refund request for the whole amount of Multiple Partial refunds for the same transaction. However, if your refund request for a certain transaction is still pending for approval, you can't submit a second refund request for the same transaction.

Note: This feature is also available from Merchant Dashboard > Bank Account > Refunds

Note: This API can be used by Active accounts only; it's not available for Demo merchants

PATH	https://www.paytabs.com/apiv2/refund_process
REQUEST METHOD	POST
PRODUCTION	LIVE



10.1 Request

<i>Element</i>	<i>Description</i>	<i>Format</i>
merchant_email	<u>Required</u> Merchant email that you use to sign up and/or login into PayTabs Merchant Dashboard	String Up to 128 characters E.g.: foof@foo.com
secret_key	<u>Required</u> SECRET Key can be found at merchant dashboard	String Up to 128 characters E.g.: Mao8sdhasfyahosdap9sd
transaction_id	<u>Required</u> The transaction id of the successful payment that you wish to refund	String Up to 12 characters E.g.: 130017
order_id	<u>Optional</u> The order id of the payment from your system. You can either use this or use the paypage_id This the same as reference_no value posted in Create PayPage call	String Up to 32 characters E.g.: 4578
refund_amount	<u>Required</u> The amount you wish to refund. It can either be partial amount or full amount.	Up to 3 Decimal places E.g.: 134.999
refund_reason	<u>Required</u> The reason for refunding the transaction	String Up to 256 Characters

10.2 Response

<i>Element</i>	<i>Description</i>	<i>Format</i>
result	The string representation for result, whether transaction has succeeded or failed and the reason why your transaction failed.	String E.g.: Payment is completed.
response_code	The response code / error code of the API call. A complete list of response codes can be found in Merchant Dashboard > Help > Reason Codes	String Up to 4 characters E.g.: 4002



10.3 Response Codes

<i>Error Code</i>	<i>Description</i>
4001	Missing parameters
4002	Invalid Credentials
810	You already requested Refund for this Transaction ID
811	Amount is above or below the invoice and also the minimum balance
812	Refund request is sent to Operation for Approval. You can track the Status. To know the result of your request you need to integrate the IPN service mentioned in the following section.
813	You are not authorized to view this transaction
814	Refund request is processed successfully

11 PayTabs IPN (Instant Payment Notification)

11.1 How IPN Works

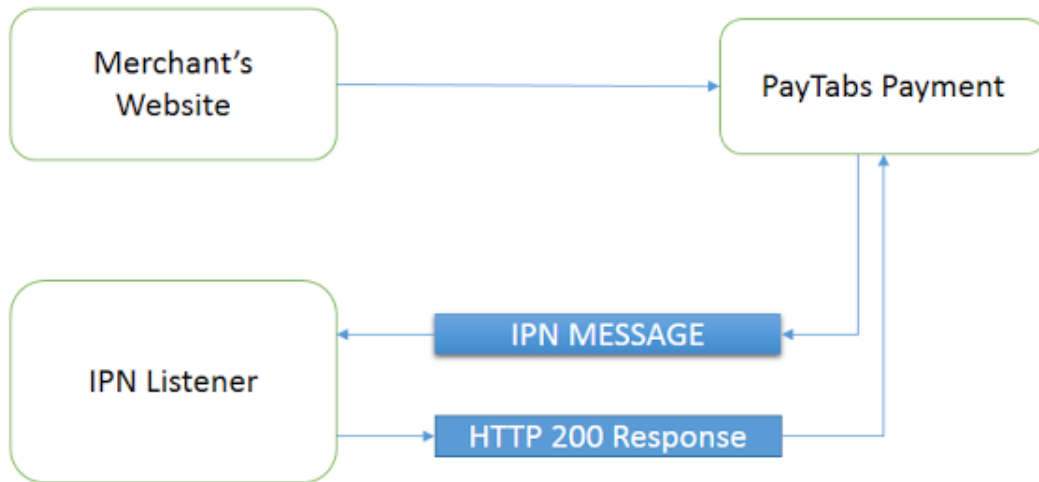
When a customer makes a payment, the transaction is processed by PayTabs and a confirmation is sent back to the customer on the merchant's application. Additionally, a notification email will be sent to the merchant.

This scenario, leave no scope for any related ERP or E-commerce or Accounting Systems to be updated. This is where IPN comes in.

As soon as a transaction is created or altered, PayTabs IPN will instantly send a notification to the merchant's specified IPN Listener URL, which can have any code that will assist in updating ERP or dependent systems. Alternatively, the merchant may wish to save this information in his local database.



IPN FLOW DIAGRAM





11.2 Set up your IPN Listener

1. Create an IPN Listener that receives and processes the IPN messages sent by PayTabs.
2. Log into your PayTabs Merchant Account > Go to “My Profile” and click on “Edit”.
3. Enter the URL of your IPN Listener and click on “Submit”. IPN will not work with a local host IPN Listener URL. Testing the IPN Message Service is only possible on a publicly hosted URL.

The screenshot shows a form with the following fields and controls:

- Paypage Currency:** A dropdown menu currently showing 'USD'.
- IPN Listener URL:** A text input field, which is highlighted with a red rectangular border.
- IPN Listener URL for Transactions Reports:** A text input field.
- Payment Timeout (Seconds):** A checkbox labeled 'No Timeout' which is checked.
- Buttons:** 'Submit' (in blue) and 'Cancel' (in grey) buttons at the bottom right.

Now your IPN Service is ready to go!

11.3 IPN Events

Your listener will be triggered on the following events:

1. Successful Transaction
2. Rejected Transaction
3. Force Accepted Transaction by PayTabs
4. Refunded Transaction

11.4 Contents / Data of the IPN

The Instant Payment Notification will return the following information depending on the type of event that has occurred.

<i>Element</i>	<i>Description</i>	<i>Format</i>
transaction_id	PayTabs Transaction ID	String E.g.: 12345
order_id	Merchant's Website order ID, this the same as reference_no value posted in Create PayPage call	String Up to 12 characters E.g.: 90094
response_code / transaction_response_code	The response code / error code of the API call. A complete list of response codes can be found in	String Up to 4 characters E.g.: 4002



customer_name	Name of the Customer	String 32 characters E.g.: John
customer_email	Email of the customer	String 32 characters E.g.:customer@domain.com
customer_phone	Phone number of the customer	String Minimum 6 numbers up to 32 numbers E.g.: 3312345678
transaction_amount	The transaction amount.	Decimal Up to 3 Decimal places E.g.: 123.399
transaction_currency	Currency of the amount stated. 3 character ISO currency code	String E.g.: BHD
first_4_digits	The first 4 digits of the credit card number used in the transaction.	String E.g.: 5387
last_4_digits	The last 4 digits of the credit card number used in the transaction.	String E.g.: 6908
card_brand	The card brand of the credit card used in the transaction.	String E.g.: MasterCard
secure_sign	A secure sign key for merchant to validate if the payment is returning from PayTabs	Alphanumeric E.g.:1f074e6f8b61d4f0af3042ce118e82535f8cb447
datetime	Date/Time of the transaction This value is returned in case of rejected or approved transactions	String dd-mm-yyyy hh:mm:ss 25-04-2016 12:25:26 PM
force_accept_datetime	Date/Time of when the transaction is force accepted. This value is returned instead of “datetime” value for transactions when they are force accepted only	String dd-mm-yyyy hh:mm:ss 25-04-2016 12:25:26 PM
refund_req_amount	Refunded amount for an approved refund request. This value is returned once a refund request for a transaction is approved	Decimal Up to 3 Decimal places E.g.: 123.399



Refund_req_id	Refunded id of refund request.	E.g: 12322
	This value is returned once a refund request for a transaction is approved or rejected	

11.5 Response Codes

Response Code	Description
100	Payment has been accepted successfully
5001	
5002	Payment has been rejected
481 or 482	Transaction rejected by fraud monitoring tools. These transactions can be force accepted upon merchant's request
5002	Payment has been forcefully accepted
5003	Payment has been refunded
5004	Your refund request has been rejected
** The Complete list of response codes can be found on the Merchant Dashboard > Help > Reason Codes	

Important Notes

1. When merchants use Verify transaction API, the API will send a response for successful or rejected payment. The IPN feature will also give the same response. The merchant should be cautious and accept both notifications as one transaction and not separate transactions. Moreover, **ONLY** the IPN will post updated information about force accepted and refunded transactions.

11.6 Sample IPN responses

11.6.1 IPN sample response for successful express checkout transactions

```
Array
(  
    [transaction_id] => 263074  
    [order_id] => 25
```



```
[response_code] => 100
[transaction_response_code] => 5003
[customer_name] => John Doe
[customer_email] => john@email.com
[transaction_amount] => 1.00
[transaction_currency] => USD
[customer_phone] => 973 123456
[last_4_digits] => 1111
[first_4_digits] => 4111
[card_brand] => Visa
[payment_type] => CREDIT CARD
[secure_sign] => 7ec0ac0890acbe33139efe989d32e97e061f93f4
[datetime] => 06-04-2017 12:01:40 PM
[address_shipping] => Shipping address Manama"
[city_shipping] => Manama city shipping
[state_shipping] => Manama state shipping"
[country_shipping] => EGY
[postal_code_shipping] => 00200
[pt_customer_email] => abc@accept.com
[pt_customer_password] => TnwUfjN3s4
[pt_token] => AgC3ojtcQZt6CB5apU5RsZfYTvlGybL9
[detail] => Transaction has been Successful
)
```

11.6.2 IPN sample response for rejected express checkout transactions

Array

```
(
    [transaction_id] => 263075
    [order_id] => 25
    [response_code] => 203
    [transaction_response_code] => 203
    [customer_name] => John Doe
    [customer_email] => john@email.com
    [transaction_amount] => 1.00
    [transaction_currency] => USD
    [customer_phone] => 973 123456
    [last_4_digits] => 1111
    [first_4_digits] => 4111
    [card_brand] => Visa
    [secure_sign] => 7ec0ac0890acbe33139efe989d32e97e061f93f4
    [datetime] => 06-04-2017 12:01:40 PM
    [address_shipping] => Shipping address Manama"
    [city_shipping] => Manama city shipping
    [state_shipping] => Manama state shipping"
    [country_shipping] => EGY
    [detail] => The card used in this transaction has been declined. Please contact your bank for
    further clarification.
)
```



11.6.3 IPN sample response for rejected express checkout transactions by fraud monitoring

Array

```
(
    [transaction_id] => 263076
    [order_id] => 25
    [response_code] => 481
    [transaction_response_code] => 481
    [customer_name] => John Doe
    [customer_email] => john@email.com
    [transaction_amount] => 1.00
    [transaction_currency] => USD
    [customer_phone] => 973 123456
    [last_4_digits] => 1111
    [first_4_digits] => 4111
    [card_brand] => Visa
    [secure_sign] => 7ec0ac0890acbe33139efe989d32e97e061f93f4
    [datetime] => 06-04-2017 12:01:40 PM
)
```

11.6.4 IPN sample response for force accepted express checkout transactions

Array

```
(
    [transaction_id] => 263076
    [order_id] => 25
    [response_code] => 5002
    [transaction_response_code] => 481
    [customer_name] => John Doe
    [customer_email] => john@email.com
    [transaction_amount] => 1.00
    [transaction_currency] => USD
    [customer_phone] => 973 123456
    [last_4_digits] => 1111
    [first_4_digits] => 4111
    [card_brand] => Visa
    [secure_sign] => 7ec0ac0890acbe33139efe989d32e97e061f93f4
    [force_accept_datetime] => 06-04-2017 12:01:40 PM
)
```

11.6.5 IPN sample response for refunded express checkout transactions

Array

```
(
    [transaction_id] => 263076
    [order_id] => 25
    [response_code] => 5003
    [transaction_response_code] => 100
    [customer_name] => John Doe
    [customer_email] => john@email.com
    [transaction_amount] => 100.00
)
```



```
[transaction_currency] => USD
[customer_phone] => 973 123456
[last_4_digits] => 1111
[first_4_digits] => 4111
[card_brand] => Visa
[secure_sign] => 7ec0ac0890acbe33139efe989d32e97e061f93f4
[refund_req_amount] => 50
[refund_req_id] =>12322
)
```



12 Transactions Reports

This API call will post all transactions that have taken place within the specified time interval to a URL.

Before calling this API, you will need to set the listener URL for the reports. Login to your merchant dashboard and edit your profile. In the 'IPN Listener for Transactions Reports' Enter the URL where you would like PayTabs to post your transactions and click on Save.

Once you call the transaction_reports API, all the transactions will be posted to the listener URL and the response will be contain the number of transactions that will be posted.

PATH	https://www.paytabs.com/expressv2/transaction_reports/
REQUEST METHOD	POST
PRODUCTION	LIVE

12.1 Request

<i>Element</i>	<i>Description</i>	<i>Format</i>
merchant_id	<u>Required</u> Merchant email id	String E.g.: foo@foo.com
secret_key	<u>Required</u> Secret Key from merchant dashboard.	String E.g.: Mao8sdhasfyahosdap9sd
startdate	<u>Required</u> Start Date	String mm-dd-yy HH:MM:SS E.g.: 02/01/2016 12:00:00s
enddate	<u>Required</u> End Date	String mm-dd-yy HH:MM:SS E.g.: 02/20/2016 12:00:00

12.2 Response

<i>Element</i>	<i>Description</i>	<i>Format</i>
Response_code	Response code of the API call	String E.g.: 4001, 4090, 4091
transaction_count	Total number of transactions.	String E.g.: 23



details	The details of all the retrieved transactions	String
status	Payment Status	String E.g.: <i>Approved/ Rejected / Refunded / Force Accepted</i>
response_code	Payment Result Response Code 100 for successful/ refunded payment 481: DM declined/ Force Accepted Payment other: Rejected	String E.g.: 475
transaction_id	PayTabs Transaction ID	String E.g.: 12345
order_id	Merchant's Website order ID	String E.g.: 1234
auth_code	Authorization Code	String E.g.: AhjzbwSR8VsHlol4QVCEcgSegdUxd
transaction_title	Title of the transaction	String
amount	Transaction Amount	String E.g.: 199.287
currency	Transaction Currency	String 3 Character ISO country Code E.g.: AED
net_amount	The equivalent of the original amount converted to merchant dashboard Currency	String E.g.: 199.287
net_amount_currency	Merchant Dashboard Currency	String 3 Character ISO country Code E.g.: AED
net_amount_credited	Net Amount Credit to Merchant's Balance after deducting PayTabs fees	String E.g.: 199.287
net_amount_credited_currency	Merchant Dashboard Currency	String 3 Character ISO country Code E.g.: AED
transaction_datetime	Date and time of the transaction	String E.g.: 23-03-2016 01:39:16 pm



force_accept_datetime	Date and time of force accept (If applicable)	String E.g.: 23-03-2016 01:39:16 am
-----------------------	---	--

12.3 Response Codes

<i>Response Code</i>	<i>Description</i>
4001	Missing parameters
4002	Invalid Credentials
4006	Your time interval should be less than 60 days
4090	Data Found
4091	Transaction Count is 0



12.4 Sample Response

Below is a sample of the response received from the API call in JSON format:

```
{
  "transaction_count": 2,
  "response_code": "4090",
  "details": [
    {
      "status": "Payment Approved",
      "response_code": "100",
      "transaction_id": "130044",
      "order_id": "1234567",
      "auth_code": "AhjzbwSR8Re008y6YqFo3AKfvJwhpLVUHyL4C7ekmW6QHFHlgDAAxQlz",
      "transaction_title": "Mr. John Doe",
      "amount": "1.00",
      "currency": "SAR",
      "net_amount": "3.73",
      "net_amount_currency": "SAR",
      "net_amount_credited": "3.58",
      "net_amount_credited_currency": "SAR",
      "transaction_datetime": "20-03-2016 06:07:34 pm",
      "force_accept_datetime": "Not Applicable"
    },
    {
      "status": "Payment Approved",
      "response_code": "481",
      "transaction_id": "130043",
      "order_id": "1234567",
      "auth_code": "AfvzbwSR8RcpB85hKYko3AKfvmhqlu1UHyL4C7ekmW6QHFHlgDAA2gfW",
      "transaction_title": "Mr. John Doe",
      "amount": "1.00",
      "currency": "SAR",
      "net_amount": "3.73",
      "net_amount_currency": "SAR",
      "net_amount_credited": "3.58",
      "net_amount_credited_currency": "SAR",
      "transaction_datetime": "20-03-2016 05:43:49 pm",
      "force_accept_datetime": "20-03-2016 06:43:49 pm"
    }
  ]
}
```



13 Test Credit Card Numbers

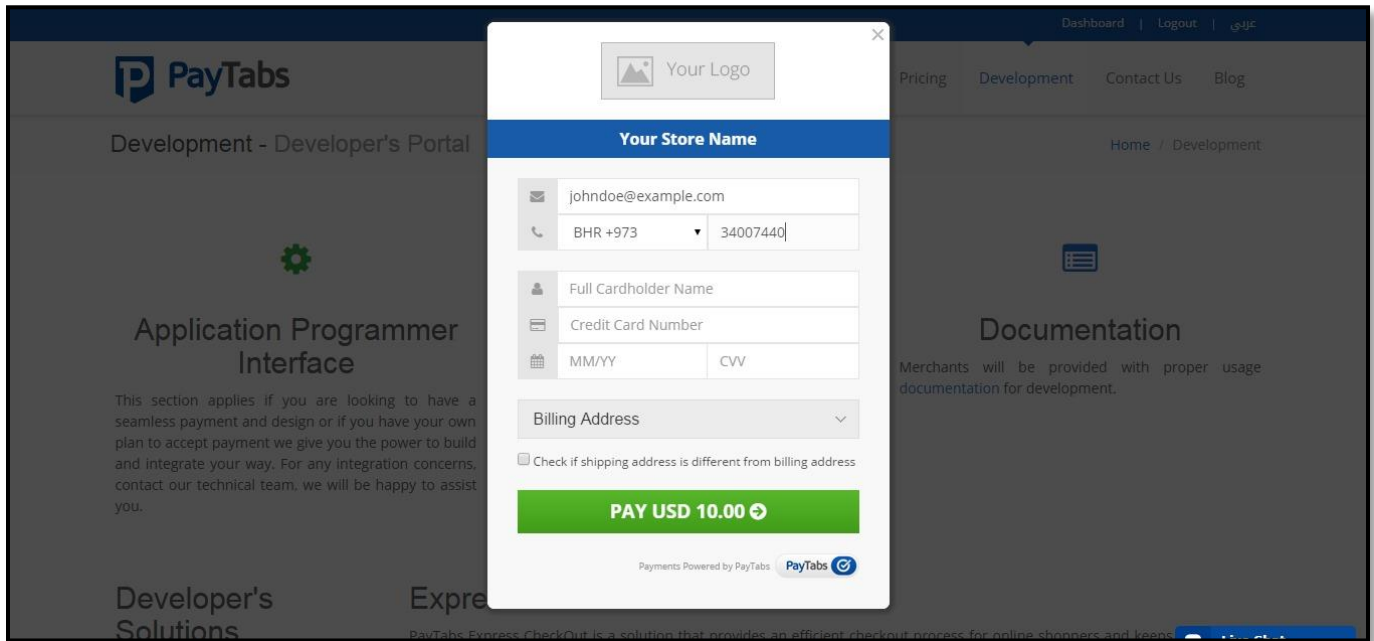
To test your payment process, you can typically use any valid credit card number or you can use the ones listed below. Please do not use test cards in a live environment. This will cause your transactions to be rejected.

Card Number	Card Type	Description	CVV	Expiry Date
40000000000000051	VISA	Without authentication window	123	12/22
40000000000000002	VISA	With authentication window (3D secure)	123	12/22
40000000000000127	VISA	Card enrollment option during purchase process	123	12/22
52000000000000007	MasterCard	With authentication window (3D secure)	123	12/22
52000000000000114	MasterCard	Without authentication window	123	12/22
52000000000000122	MasterCard	Card enrollment option during purchase process	123	12/22

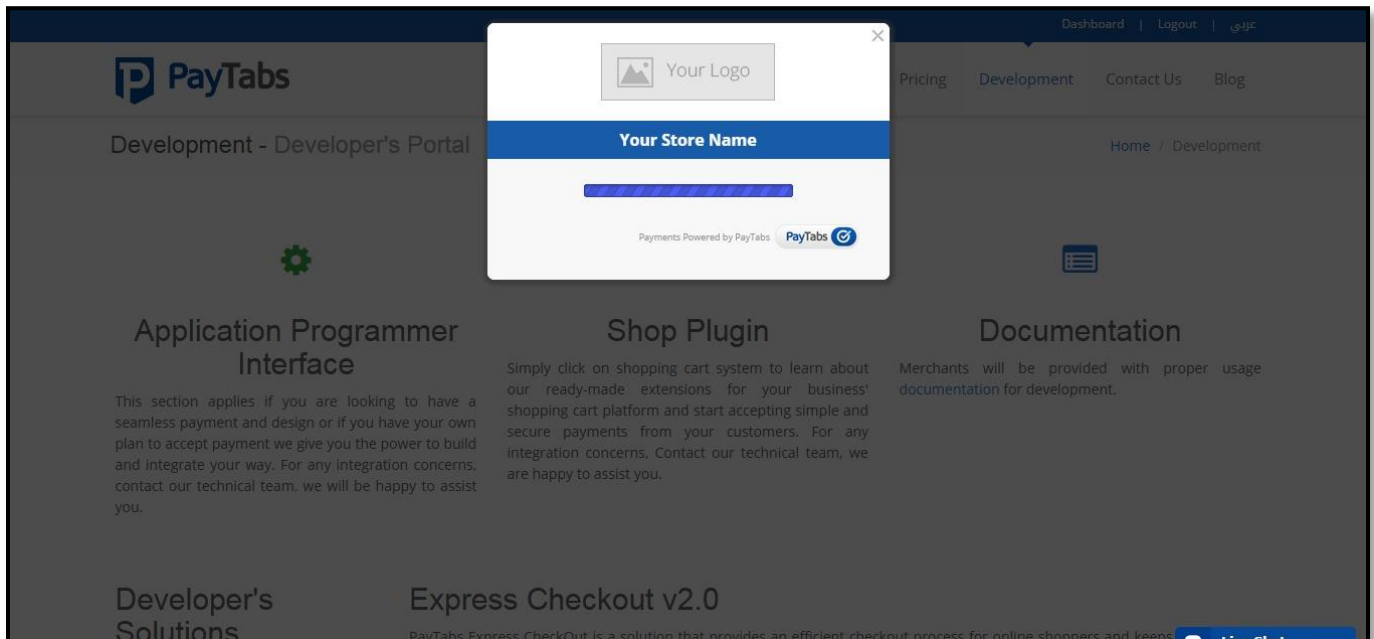


14 Express Checkout Screen Shots

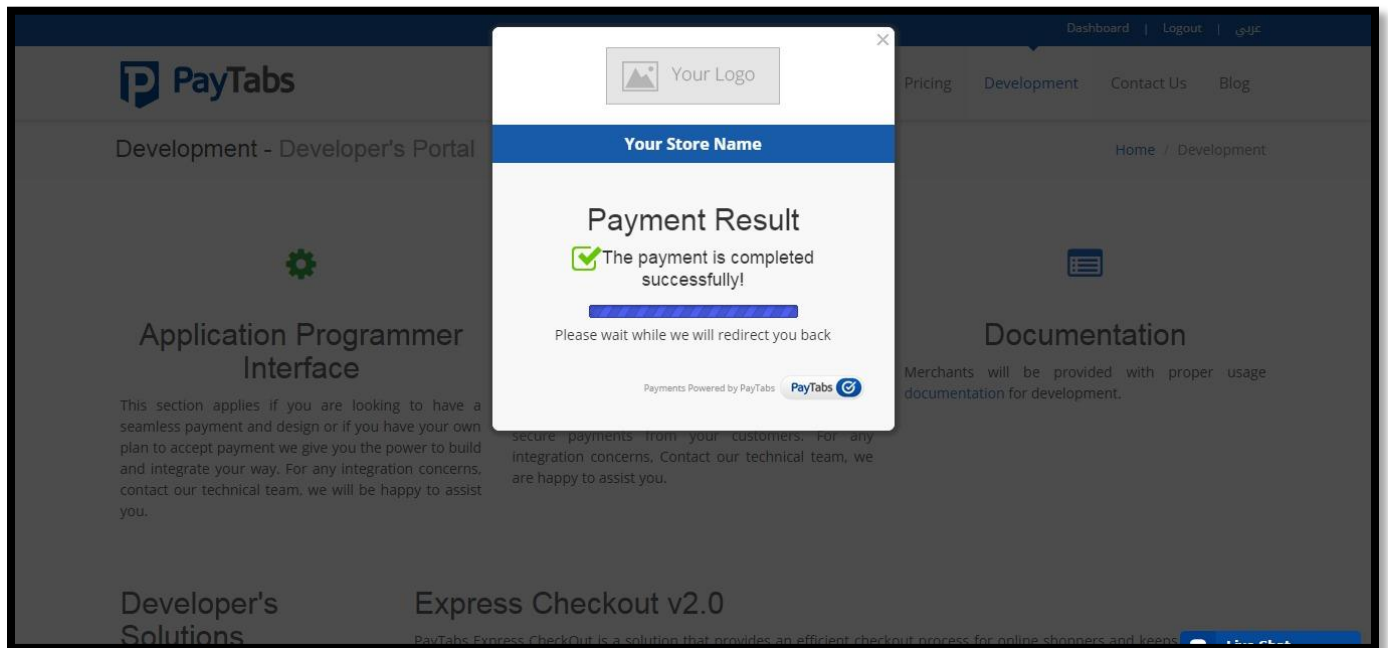
1. Express Checkout will appear as prefilled and customer needs to enter credit card details and click Pay.



2. Payment is processed without redirection from Merchant's Website.



3. Payment Result is displayed on Express Checkout Popup.





15 Frequently Asked Questions

Q1. I am getting an error on Express CheckOut Button: “Notice: Trying to get property of non-object”.

This is usually caused due to some incorrect configuration on your server. You will need to re-check that your server has been properly configured with all the pre-requisites required for Express CheckOut V2.0

- cURL
- PHP 5.0 or greater

Q2. I can see a progress/loading bar that never stops loading! What do I do?

After integration, if you get a progress/loading bar that never stops loading, kindly debug your page. The following are the possible causes and their solutions:

1. Error on authentication

- Check for any leading or trailing spaces in your secret key or email address and remove them.

2. 404: Page not found error

- Make sure to use the complete and exact site URL
- Make sure to add a trailing “/” to the site URL.
- Ensure that your siteurl and url_redirect begin with “https” only if you are using an SSL Certificate. If you do not have an SSL certificate installed, then kindly begin the URLs with “http”.
- Do not rename the file, do not change the case of the file name / Check with your hosting company.

Q3. Does the Express CheckOut support all languages?

The Express CheckOut only supports English and Arabic. Setting the language variable to anything else will default to English.

Q4. Express CheckOut does not give me any result, just a blank screen! What do I do?

Make sure you are using the latest version of Express CheckOut.

PayTabs releases updates to the various services from time to time. Information of these updates is sent to merchants via the blog and monthly newsletter.

Q6. I am getting an error “Error: No 'Access-Control-Allow-Origin' header is present on the requested resource.”

Make sure that you have entered the correct links for the siteurl and url_redirect variables. Use “https” only if you have an SSL certificate installed. Also make sure to add the “www”. Your URL structure should look similar to this:

With SSL : https://www.mywebsite.com/

Without SSL: http://www.mywebsite.com/



Q7. I am getting following warning Session_Start()

Express checkout warning means that your PHP script already sent the HTTP headers, and as such it can't make modifications to them now. Check that you don't send ANY content before calling session_start.

Better yet, just make session_start the first thing you do in your PHP file (so put it at the absolute beginning, before all HTML etc.).

<http://stackoverflow.com/questions/8812754/cannot-send-session-cache-limiter-headers-already-sent>

Q8. I cannot see the custom button image when I enter a few parameters in Arabic. How do I fix this?

Make sure the following line is added to your button code, so that the UTF-8 support is enabled.

```
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
```

Q9. The close button on express checkout is redirecting to a 404 page on PayTabs. What do I do?

Ensure that there are no quotation marks or apostrophe in the express checkout button code

```
Paytabs("#express_checkout").expresscheckout({
settings: {
secret_key:"QM0JVBlg7cEfb9HqKATI5OzRSRSIkFKarKTnlfEsriICsp2dZWW5KvXCnPkIv4TXONk9
2ZCthk6Kj9XJ5TOPpLrNBLv6Qo3EboOd",
merchant_id: "10010864",
amount: "10",
currency: "USD",
title: "My Express Checkout",
display_shipping_fields: 0,
product_names: "Leather Garment",
order_id: 25,
redirect_on_reject: 0,
url_redirect: "http://myexpresscheckout.ae/Thankyou.aspx/"
}
});
```

10. How can I get the product names from Verify Payment API & IPN response from Express Checkout?

The product names can be stored in your database against your own invoice id/reference id/order Id. Once the payment is made, we will send the same id back to your server in the verify_payment / IPN response. You can then compare the IDs and retrieve the product names from your own database.



11. I am receiving a “403 Forbidden” error when I try to use Express CheckOut. How do I fix this?

Ensure that your site URL is accepting POST requests. When the customer completes the payment, PayTabs send request to return URL by post method. But the return URL specified may not be accepting post requests.

12. Can I customize the Express CheckOut close (X) button?

The close button redirects to checkout page and cannot be customized.

13. I am getting the following error “Transaction Error: The currency is not a valid ISO-Certified currency. Please use 3 digit ISO-Certified currency codes.” But my currency code is correct! What’s the problem?

1. Make sure that you have entered the correct links for the site_url and url_redirect variables. Use “https” only if you have an SSL certificate installed. Also make sure to add the “www”. Your URL structure should look similar to this “

With SSL : https://www.mywebsite.com/
Without SSL : http://www.mywebsite.com/

2. Check and confirm whether you are using the correct ISO Currency Code.

For example:

ISO Currency Code for Bahraini Dinar is BHD and not BD,

ISO Currency Code for Saudi Arabian Riyal is SAR and not SR.

Merchant needs to pass correct 3-character ISO codes to country and shipping_country fields. We have 4 parameters accepting 3-character ISO code:

For example, the ISO codes for United Arab Emirates (UAE) are:

country	: ARE
shipping_country	: ARE
currency	: AED
cc_phone_number	: 00971

FAQ -Express CheckOut – iframe

1. The iFrame window is very small, how can I enlarge it to show the whole window without scroll?

Change the following line in iFrame code from:

```
<div class="PT_express_checkout">
```

to

```
<div class="PT_express_checkout" style="height: 300px;">
```



2. I am receiving a phishing attack notification from my anti-virus software for the area where the PayTabs iFrame is supposed to be. What do I do?

This is a false positive. You can whitelist/exclude the following links in the anti-virus software you are using such as Kaspersky and the phishing alert will disappear.

https://www.paytabs.com/expressv2/*

https://www.paytabs.com/expressv3/*

3. Are there any existing merchants using the IFrame solution? Can you provide sample?

A few of the merchants integrating IFrame are as follows: (This list is subject to change)

	Merchant ID	Store Name	Payment Channel
1	10009676	تحاضیر	Express Checkout
2	10012808	Chemindex2016	Express Checkout
3	10011200	دار سیبویه	Express Checkout
4	10011038	biohome.org	Express Checkout
5	10008411	Okaz	Express Checkout
6	10011683	toov	Express Checkout
7	10011268	George VI VIP	Express Checkout



Express Checkout Integration Guide – [Version 4.1](#)

www.paytabs.com