­­­­

|  |
| --- |
| **Merchant Integration Guide** |
|  |



Authors:

* Syed Danyal Hassan (Service Delivery Executive)

Reviewed by:

* Syed Faisal Alam(Chief Technology Officer)

Revision History:

|  |  |  |  |
| --- | --- | --- | --- |
| **Date** | **Revised By** | **Revision Section** | **Incorporated Changes** |
| 8 June 2021 | Syed Danyal Hassan |  | Creating Merchant On Boarding Document |
| 9 July 2021 | Syed Danyal Hassan | 3.4 Instant Payment notification | Adding Instant payment notification content. |

Table of Contents

[1 Preface 4](#_Toc40108046)

[1.1 About 4](#_Toc40108047)

[1.2 Intended Audience 4](#_Toc40108048)

[1.3 Integration scope 4](#_Toc40108049)

[2 Introduction 4](#_Toc40108050)

[2.1 Payment Flow: 4](#_Toc40108051)

[2.2 Pay2m Checkout Flow 4](#_Toc40108052)

[2.3 Merchant Logo/Name setup on PAY2M gateway page 5](#_Toc40108054)

[2.4 Getting started with PAY2M 6](#_Toc40108055)

[3 Technical integration requirements 7](#_Toc40108061)

[3.1 Getting an ACCESS TOKEN 7](#_Toc40108062)

[3.2 Initiate Transaction via Form POST 8](#_Toc40108063)

[For Redirection implementation 8](#_Toc40108064)

[3.3 Description of mandatory field 14](#_Toc40108065)  
3.4 Instant Payment Notification (IPN) 15

[4 Error Codes & Description 15](#_Toc40108066)

[5 PAY2M checkout Page 17](#_Toc40108067)

[5.1 For redirection implementation 16](#_Toc40108068)

[5.1.1 For Master/Visa 16](#_Toc40108068)

[5.1.2 For NAPS 19](#_Toc40108068)

# **Preface**

## About

This document contains detail explanation about how to integrate with PAY2M Using Our Hosted Checkout and I-frame based transactions functionality. This document contains the details for online transactions.

## Intended Audience

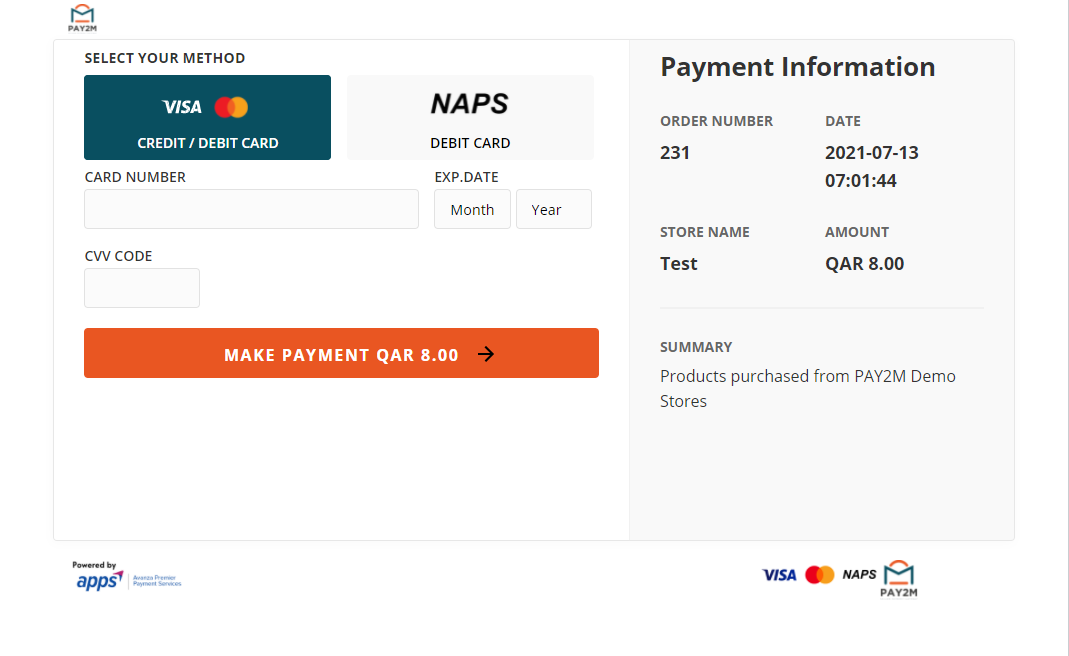
This document is for merchants, acquirers and developers who want to integrate with PAY2M to perform Our Hosted Checkout and I-frame based transactions.

## Integration scope

The merchant will implement all the ecommerce functionality. PAY2M service will be used only for payment processing.

# **Introduction**

PAY2M payment gateway is a payment service developed for merchants to accept payments from the customers using banks’ instruments.



Transaction starts from the merchant’s web based ecommerce portal, after this the customer checks out and is redirected to PAY2M payment gateway web checkout page where customer will enter their credit, debit, wallet and, account number Buyer Send the correct OTP (one-time password), and completes the payment process, and customer will have redirected towards the same merchant’s ecommerce portal where the customer order is confirmed by merchant if payment is successful.

## Payment Flow:

A generic payment flow

In a typical PAY2M payment flow, a buyer:

1. Buyer (customer) adds items in the shopping cart on merchant’s ecommerce portal.
2. Buyer reaches checkout page and select the PAY2M as the payment option.
3. Buyer is redirected to PAY2M payment gateway web checkout page (For Details see section “3” PAY2M Checkout Flow).
4. Buyer Send the correct OTP (one-time password), and completes the payment process. In addition, customer will have redirected towards the same merchant’s ecommerce portal where merchant confirms the customer order if payment is successful.

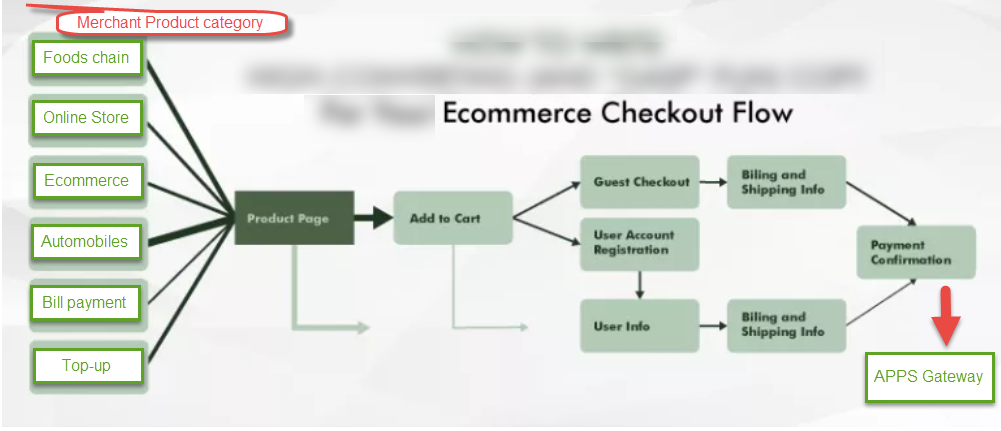
## Pay2m Checkout Flow

In the PAY2M checkout flow, the buyer:

1. Logged IN user and clicks, “Submit my order” button on the merchant’s ecommerce portal.

### For redirection implementation

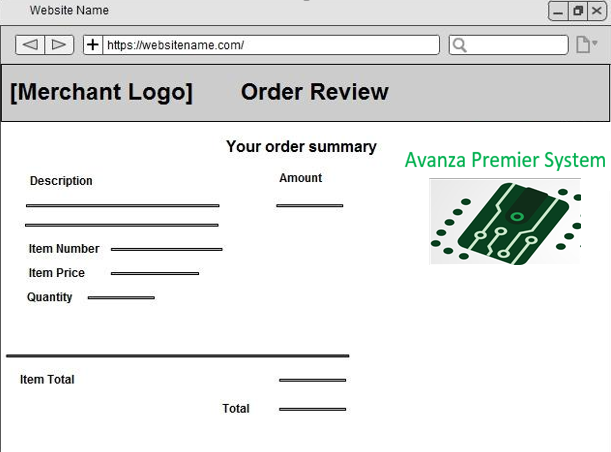
1. Order request forward to the PAY2M checkout page.
2. Buyer select banks and enter their credit, debit, wallet and account number, and proceed payment.
3. After clicking validate button, transaction request to the selected bank for authentication via OTP from customer.
4. After receiving OTP, customer can enter the OTP which is provided by bank on SMS/USSD channel.
5. Returns to Merchant site and confirms the order then receives confirmation.



## Merchant Logo/Name setup on PAY2M gateway page

Before the transaction is processed the buyer is taken to a Merchant Logo/Name on PAY2M payment gateway page, where the buyer’s information is taken, and after submission the order information is displayed. Upon continuation, this page is used to process the transaction with the PAY2M platform.

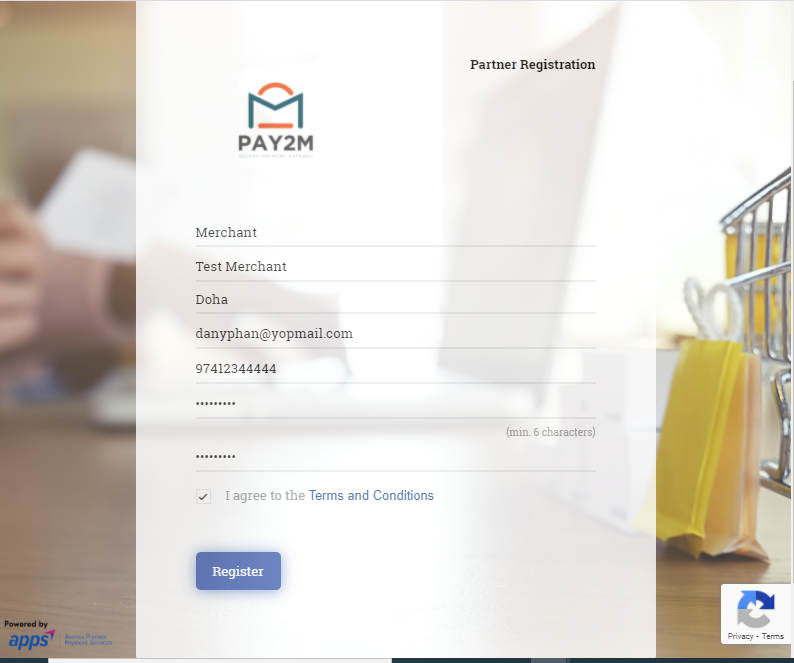
The buyer is subsequently taken to the merchant website order confirmation page.



## Getting started with PAY2M DMS

## 

Digital Merchant Sign Up is an automated process for signing up merchants. Digital Merchant Sign up acts as virtual entity that takes to-be-merchant information like MDR, product(s) to be used like pay fast payment gateway and payment link (invoice) and further details like business details, documents, contact details, bank info, UAT sign off and approving terms and conditions. Once information is saved in DMS, operation would take the merchant data for further analysis with respect to risk and data analysis, once merchant is validated with all aspects then operation would permit the merchant to go live. A notification would be sent to merchant regarding successful registration at DMS and then merchant would login into the portal and enable the live mode by clicking on Go live button. Operation would be allowed to create merchant as well as approving/rejecting the recently on boarded merchant. Operation would have the feasibility to pin point the mistakes or any document mismatch detection by sending objection along with details to merchant.



# 3. Technical integration requirements

## 3.1. Getting an ACCESS TOKEN

A web hook has to be called to get the access token. Merchant ID and Secure Key are the two values will be sent to fetch the access token. TXNAMT and BASKET\_ID are optional parameters for generating access token. Moreover, Empty user agent not allowed. Please see below table 1.1.

|  |  |
| --- | --- |
| Table 1.0 |  |
| URL | https://payments.pay2m.com/Ecommerce/api/Transaction/GetAccessToken |
| Method | POST |
| Parameters | MERCHANT\_ID |
|  | SECURED\_KEY |
|  | BASKET\_ID |
|  | TXNAMT |
| Response Type | JSON |
| Response | {  “MERCHANT\_ID”:”<<MERCHANT\_ID>>”   “ACCESS\_TOKEN”:”<<ACCESS\_TOKEN>>”  “NAME”:”sha”  “GENERATED\_DATE\_TIME”:”<<GENERATED\_DATE\_TIME>>”  } |

## 3.2 Initiate Transaction via Form POST

For Redirection implementation  
  
after getting an access token, ecommerce will send a **FORM POST** along with following parameters to PAY2M payment gate.

|  |  |
| --- | --- |
| Table 1.1 |  |
| URL for redirection option | https://payments.pay2m.com/Ecommerce/api/Transaction/PostTransaction |
| Method | POST |
| Form Parameters | **Description** |
| MERCHANT\_ID | A unique ID provided by PAY2M |
| MERCHANT\_NAME | Brand name |
| TOKEN | Access token fetched by web hook as mentioned in Table 1.0 |
| PROCCODE | 00 |
| TXNAMT | The amount which needs to be paid by the customer |
| CUSTOMER\_MOBILE\_NO | Customer’s mobile number |
| CUSTOMER\_EMAIL\_ADDRESS | Customer’s email address |
| SIGNATURE | A random string value |
| VERSION | A random string value |
| TXNDESC | Short description of the goods purchased by customer |
| SUCCESS\_URL | URL where customer will be redirected to in the browser if payment is successful. |
| FAILURE\_URL | URL where customer will be redirected to in the browser if payment is unsuccessful. |
| BASKET\_ID | A unique order id from the merchant, which will represent customer’s order. |
| ORDER\_DATE | Date of the order. (Format: YYYY-MM-DD e.g. 2018-01-01) |
| CHECKOUT\_URL | URL payment information is pushed to the merchant at the backend. |
| STORE\_ID | Alphanumeric string which would come from merchant portal’s outlet section after outlet creation <optional> |
| CURRENCY\_CODE | QAR <optional> |
| TRAN\_TYPE | Transaction type for settlement <optional> |

**Working Code Example**

|  |
| --- |
| <html>  <head>  <link rel="stylesheet" href="https://stackpath.bootstrapcdn.com/bootstrap/4.4.1/css/bootstrap.min.css" integrity "sha384-Vkoo8x4CGsO3+Hhxv8T/Q5PaXtkKtu6ug5TOeNV6gBiFeWPGFN9MuhOf23Q9Ifjh" crossorigin="anonymous">  </head>  <body>  <h1>pay2m Example Code For Redirection Payment Request</h1>  <?php  $merchant\_id = 1003;  $secured\_key = 'pfYu8vPrUmYD6JZJAKhl';  $basket\_id = 'Basket Item-1';  $trans\_amount = 1;  if (count($\_GET) > 0) {  processResponse($merchant\_id, $basket\_id, $trans\_amount, $\_GET);  }  $token = getAccessToken($merchant\_id, $secured\_key, $basket\_id, $trans\_amount);  /\*\*  \* get access token with merchant id, secured key, basket id, transaction amount  \*  \*/  function getAccessToken($merchant\_id, $secured\_key, $basket\_id, $trans\_amount)  {  $tokenApiUrl = 'https://payments.pay2m.com/Ecommerce/api/Transaction/GetAccessToken';  $urlPostParams = sprintf(  'MERCHANT\_ID=%s&SECURED\_KEY=%s&TXNAMT=%s&BASKET\_ID=%s',  $merchant\_id,  $secured\_key,  $trans\_amount,  $basket\_id  );  $ch = curl\_init();  curl\_setopt($ch, CURLOPT\_URL, $tokenApiUrl);  curl\_setopt($ch, CURLOPT\_RETURNTRANSFER, 1);  curl\_setopt($ch, CURLOPT\_POST, 1);  curl\_setopt($ch, CURLOPT\_POSTFIELDS, $urlPostParams);  curl\_setopt($ch, CURLOPT\_USERAGENT, 'CURL/PHP pay2m Example');  $response = curl\_exec($ch);  curl\_close($ch);  $payload = json\_decode($response);  $token = isset($payload->ACCESS\_TOKEN) ? $payload->ACCESS\_TOKEN : '';  return $token;  }  /\*\*  \* process response coming from pay2m  \*  \*/  function processResponse($merchant\_id, $original\_basket\_id, $txnamt, $response)  {  /\*\*  \* following parameters sent from pay2m after success/failed transaction  \*  \*/  $trans\_id = $response['transaction\_id'];  $err\_code = $response['err\_code'];  $err\_msg = $response['err\_msg'];  $basket\_id = $response['basket\_id'];  $order\_date = $response['order\_date'];  $response\_key = $response['Response\_Key'];  $payment\_name = $response['PaymentName'];  $secretword = ''; // No secret code defined for merchant id 102, secret code can be entered in merchant portal.  $response\_string = sprintf("%s%s%s%s%s", $merchant\_id, $original\_basket\_id, $secretword, $txnamt, $err\_code);  $response\_hash = hash('MD5', $response\_string);  if (strtolower($response\_hash) != strtolower($response\_key)) {  echo "<br/>Transaction could not be varified<br/>";  return;  }  if ($err\_code == '000' || $err\_code == '00') {  echo "<strong>Transaction Successfully Completed. Transaction ID: " . $trans\_id . "</strong><br/>";  echo "<br/>Date: " . $order\_date;  return;  }  echo "<br/>Transaction Failed. Message: " . $err\_msg;  }  ?>  <!-- For data integrity purpose, transaction amount and basket\_id should be the same as the ones sent in token request -->  <!--  Actual Payment Request  -->  <div class="container">  <div class ="alert alert-success"> Reload page to get new token</div>  <div class="card">  <div class="card-body">  <div class="card-header">  Pay2m Web Checkout - Example Code  </div>  <form id='pay2m\_payment\_form' name='pay2m-payment-form' method='post' action=" https://payments.pay2m.com/Ecommerce/api/Transaction/PostTransaction">  Currency Code: <input type="TEXT" name="CURRENCY\_CODE" value="QAR" /><br />  Merchant ID: <input type="TEXT" name="MERCHANT\_ID" value="<?php echo $merchant\_id; ?>" /><br />  Merchant Name: <input type="TEXT" name="MERCHANT\_NAME" value="UAT Demo Merchant " /><br />  Token: <input type="TEXT" name="TOKEN" value="<?php echo $token; ?>" /><br />  Success URL: <input type="TEXT" name="SUCCESS\_URL" value="http://merchant-site-example.com" /><br />  Failure URL: <input type="TEXT" name="FAILURE\_URL" value="http://merchant-site-example.com" /><br />  Checkout URL: <input type="TEXT" name="CHECKOUT\_URL" value="http://merchant-site-example.com" /><br />  Customer Email: <input type="TEXT" name="CUSTOMER\_EMAIL\_ADDRESS" value="some-email@example.com" /><br />  Customer Mobile: <input type="TEXT" name="CUSTOMER\_MOBILE\_NO" value="+974-34312767" /><br />  Transaction Amount: <input type="TEXT" name="TXNAMT" value="<?php echo $trans\_amount; ?>" /><br />  Basket ID: <input type="TEXT" name="BASKET\_ID" value="<?php echo $basket\_id; ?>" /><br />  Transaction Date: <input type="TEXT" name="ORDER\_DATE" value="<?php echo date('Y-m-d H:i:s', time()); ?>" /><br />  Signature: <input type="TEXT" name="SIGNATURE" value="SOME-RANDOM-STRING" /><br />  Version: <input type="TEXT" name="VERSION" value="MERCHANT-CART-0.1" /><br />  Item Description: <input type="TEXT" name="TXNDESC" value="Item Purchased from Cart" /><br />  Proccode: <input type="TEXT" name="PROCCODE" value="00" /><br />    <br/>    Store ID/Terminal ID : <input type="TEXT" name="STORE\_ID" value='' /><br />  <input type="SUBMIT" value="SUBMIT">  </form> </div></div>  <script src="https://code.jquery.com/jquery-3.4.1.slim.min.js" integrity="sha384-J6qa4849blE2+poT4WnyKhv5vZF5SrPo0iEjwBvKU7imGFAV0wwj1yYfoRSJoZ+n" crossorigin="anonymous"></script>  <script src="https://cdn.jsdelivr.net/npm/popper.js@1.16.0/dist/umd/popper.min.js" integrity="sha384-Q6E9RHvbIyZFJoft+2mJbHaEWldlvI9IOYy5n3zV9zzTtmI3UksdQRVvoxMfooAo" crossorigin="anonymous"></script>  <script src="https://stackpath.bootstrapcdn.com/bootstrap/4.4.1/js/bootstrap.min.js" integrity="sha384-wfSDF2E50Y2D1uUdj0O3uMBJnjuUD4Ih7YwaYd1iqfktj0Uod8GCExl3Og8ifwB6" crossorigin="anonymous"></script> </div></body></html> |
|  |

In either case, payment gateway will send additional parameters (along with those sent already by merchant) containing transaction information. These parameters are as follows:

|  |  |
| --- | --- |
| **Table 1.2** | |
| **Method** | GET |
| **URL** | Merchant’s supplied URL |
| **transaction\_id** | Transaction ID processed at payment gateway |
| **err\_code** | Status code of the transaction.  (‘000’ is for successful transaction) |
| **err\_msg** | Descriptive message in case of failed transaction |
| **basket\_id** | Order ID sent by merchant. |
| **order\_date** | Date of order sent by merchant. |
| **responseKey** | An SHA256 hash of combination of multiple fields, to verify authenticated response:  merchant\_id + basket\_id + <secret word>+ txnamt + err\_code |

## 3.3. Description of mandatory field

|  |  |
| --- | --- |
| Mandatory Field | Description |
| MERCHANT\_ID | A unique ID provided by PAY2M |
| Merchant\_Name | Your brand name |
| TOKEN | Access token fetched by web hook as mentioned in Table 1.0 |
| PROCCODE | 00 |
| TXNAMT | The amount which needs to be paid by the customer |
| CUSTOMER\_MOBILE\_NO | Customer’s mobile number |
| CUSTOMER\_EMAIL\_ADDRESS | Customer’s email address |
| SIGNATURE | A random string value |
| VERSION | A random string value |
| TXNDESC | Short description of the goods purchased by customer |
| SUCCESS\_URL | URL where customer will be redirected to in the browser if payment is successful. |
| FAILURE\_URL | URL where customer will be redirected to in the browser if payment is unsuccessful. |
| BASKET\_ID | A unique order id from the merchant that will represent customer’s order. |
| ORDER\_DATE | Date of the order. (Format: YYYY-MM-DD e.g. 2018-01-01) |
| CHECKOUT\_URL | URL payment information is pushed to the merchant at the backend. |
| STORE\_ID | Store identifier will be added in this field that can be created from Merchant portal OR Outlet APIs |

The code being flexible is easily embedded on the merchant payment page. The buyer payment information will need to be parsed to the PAY2M gateway page via this relay form.

## 3.4. Instant Payment Notification (IPN) Instant payment notification would be enabled from merchant’s end which would allow the merchant to get payment status on his/her end easily and hassle-free.

|  |  |
| --- | --- |
| IPN Status | Description |
| 0 | IPN is not configured |
| 1 | Receive IPN status in real time |
| 2 | Receive IPN status in a delay |

## Please note that delay can be in between 5-15 minutes. Once the transaction has been performed and customer is redirected to transaction status page then if IPN is 0, then transaction id, basket id, amount paid and transaction status would not be sent to merchant. If IPN is 1 then transaction id, basket id, amount paid and transaction status would be sent to merchant that he/she would be showing on his/her checkout page. If IPN is 2 then transaction id, basket id, amount paid and transaction status would be pushed to merchant in a delay of 10-15 minutes.

# 4. Error Codes & Description

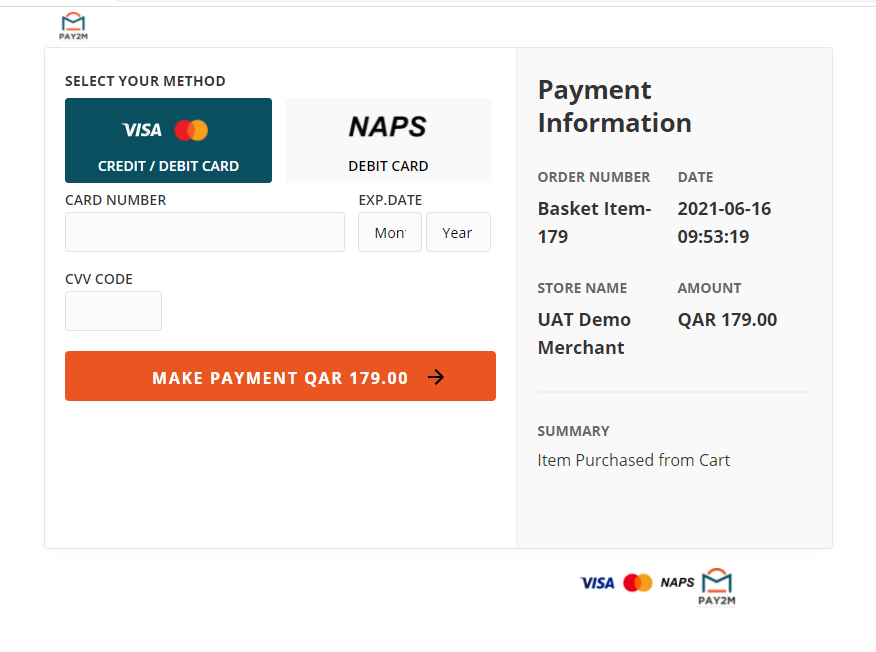
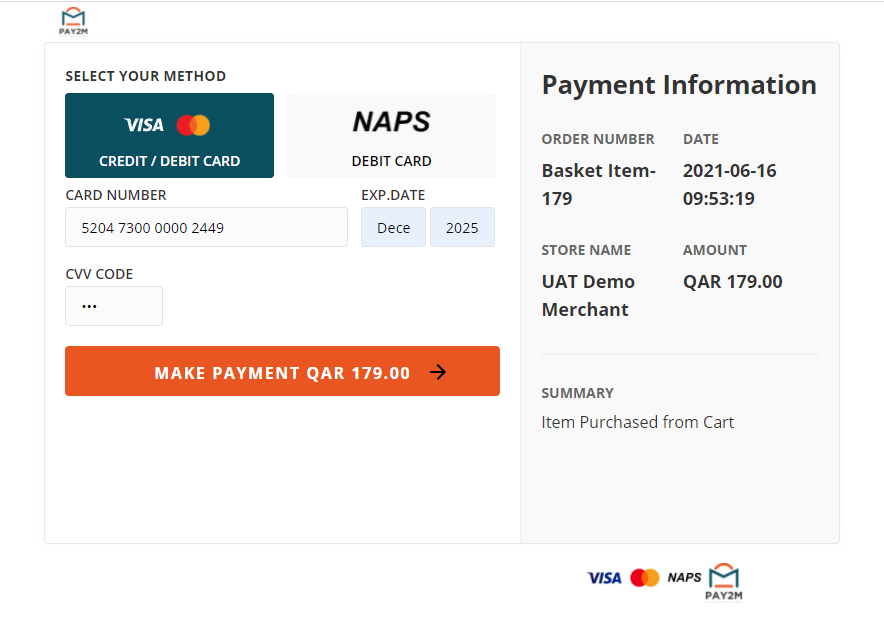
|  |  |
| --- | --- |
| ERROR CODES | ERROR DESCRIPTIONS |
| 00 | Processed OK |
| 002 | Time Out |
| 97 | Dear Customer, You have an Insufficient Balance to proceed |
| 106 | Dear Customer, Your transaction Limit has been exceeded please contact your bank |
| 03 | You have entered an Inactive Account |
| 104 | Entered details are Incorrect |
| 55 | You have entered an Invalid OTP/PIN |
| 54 | Card Expired |
| 13 | You have entered an Invalid Amount |
| 126 | Dear Customer your provided Account details are Invalid |
| 75 | Maximum PIN Retries has been Exceeded |
| 14 | Dear Customer, You have entered an In-Active Card number |
| 15 | Dear Customer, You have entered an In-Active Card number |
| 42 | Dear Customer, You have entered an Invalid CNIC |
| 423 | Dear Customer, We are unable to process your request at the moment please try again later |
| 41 | Dear Customer, entered details are Mismatched |
| 801 | {0} is your Pay2m OTP (One Time Password). Please do not share with anyone. |
| 802 | OTP could not be sent. Please try again later. |
| 803 | OTP has been sent to your email address. |
| 804 | OTP has been sent to your mobile number. |
| 805 | OTP Verified |
| 806 | OTP could not be verified. |
| 807 | Too many attempts. Please try again later in few minutes. |
| 808 | Passwords do not match |
| 809 | Invalid Password |
| 810 | Password could not be changed |
| 811 | Password changed successfully |
| 812 | Request could not be validated. Please try again. |
| 813 | Email address already registered |
| 9000 | FRMS Rejected |
| 9010 | FRMS Error |
| 308 | Dear Customer your provided Account details are Invalid |
| 600 | OTP Already Expired |
| 309 | You have entered Invalid OTP Length |
| 853 | Dear Customer, Your provided Account details are Invalid |
| 04 | You have entered a Closed Account |
| 537 | You have entered a Dormant Account |
| 359 | Blocked Account |
| 8811 | Password changed successfully |
| 880 | Transaction declined due to non-activation of local ecommerce session. Please call your bank. |
| 881 | Your transaction is declined due to insufficient funds. Please call your bank for assistance |
| 882 | Your transaction is declined due to consumption of daily local epayment transaction limit/count. |
| 883 | Your transaction is declined. Please call your bank to activate local epayment service |

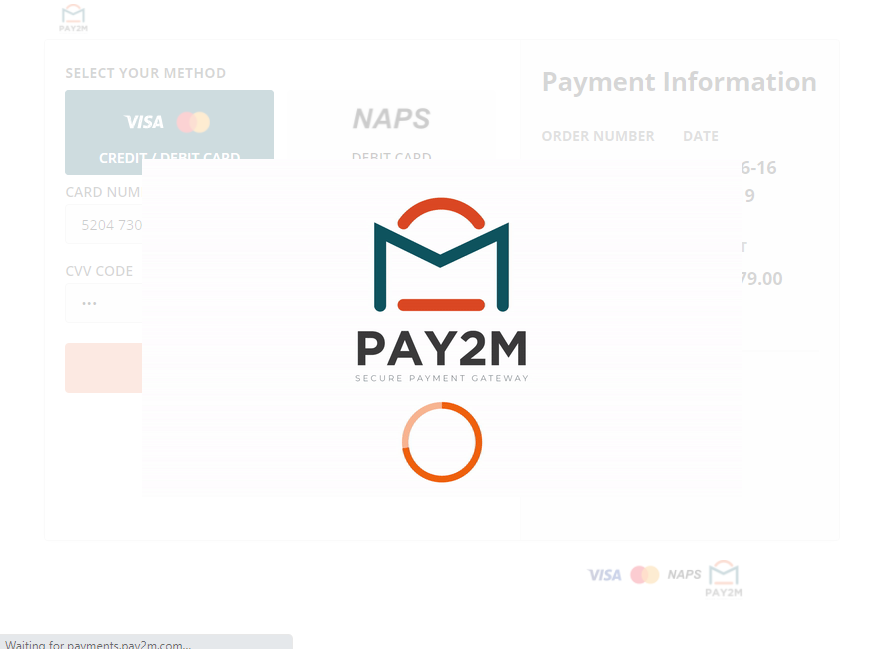
# 

# **5. PAY2M checkout Page**

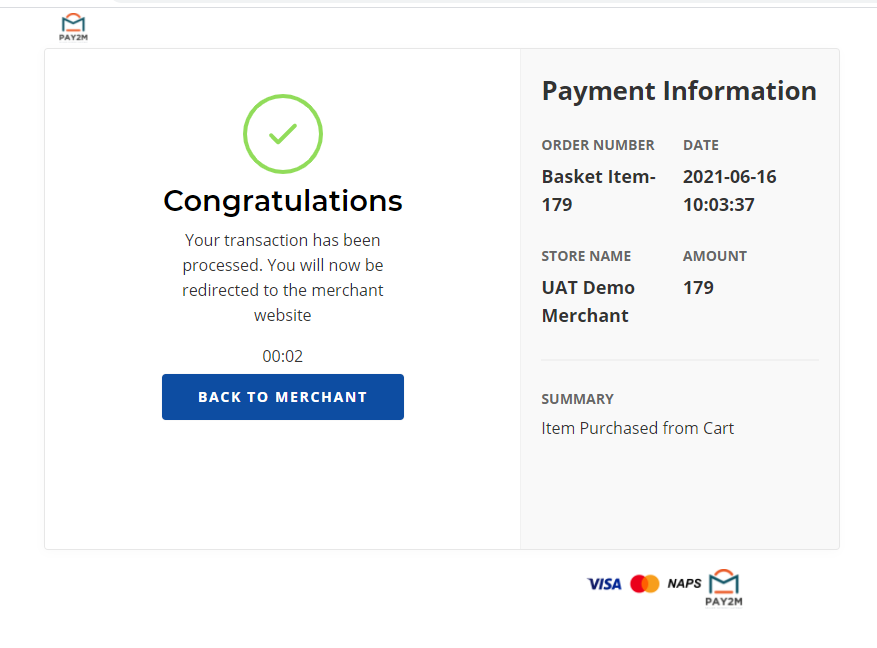
## **5.1. For redirection implementation**

On posting the form data, customer will be redirected to the following page of the PAY2M web ecommerce portal.

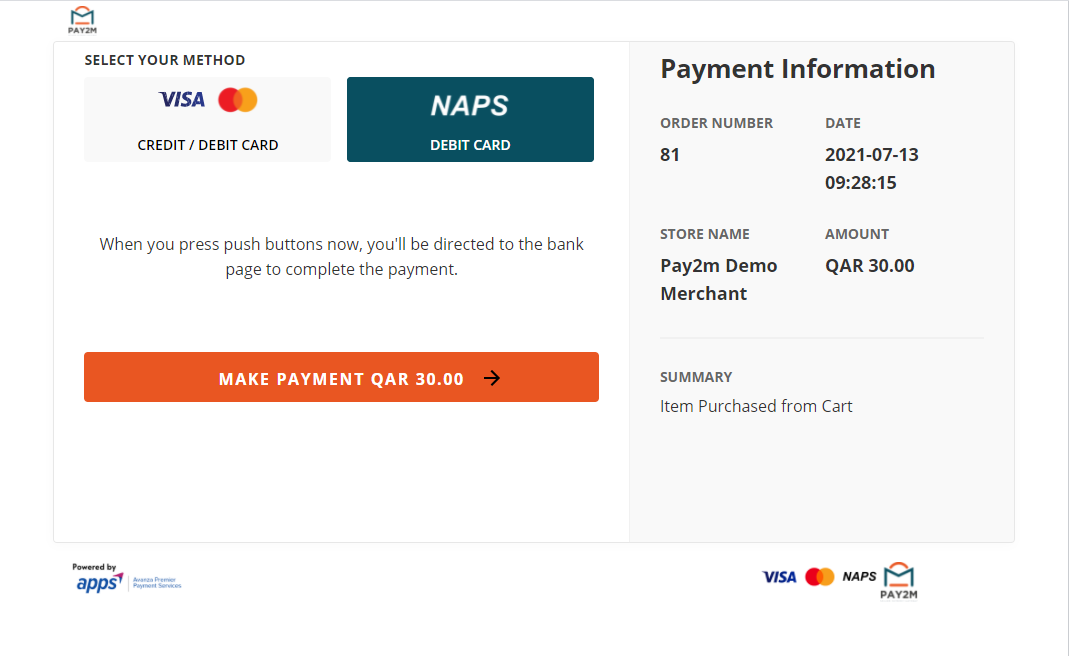
**5**.1.1 For Master/Visa  
For Master/visa transaction, proceed with clicking on credit/debit card option in payment gateway page  
  
Add your card details like card number, expiry date and year and cvv code.  
  


Click on Make payment button.   
  


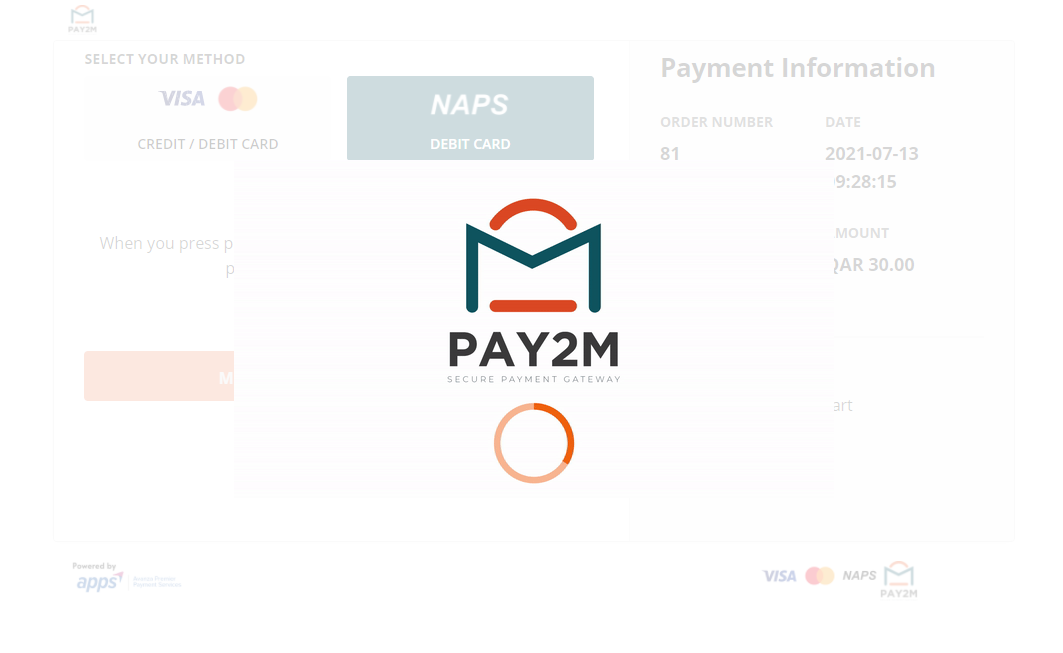
Click on submit button and you will be navigated to successful page.



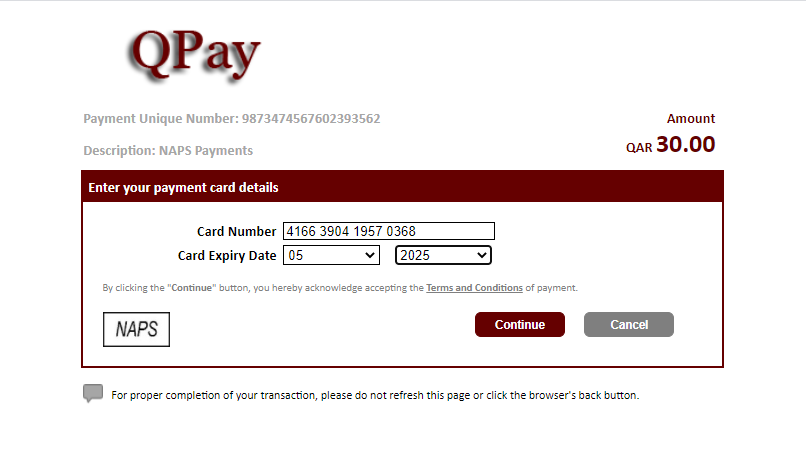
**5.1.2. NAPS**After switching on NAPS tab, click on make payment button.



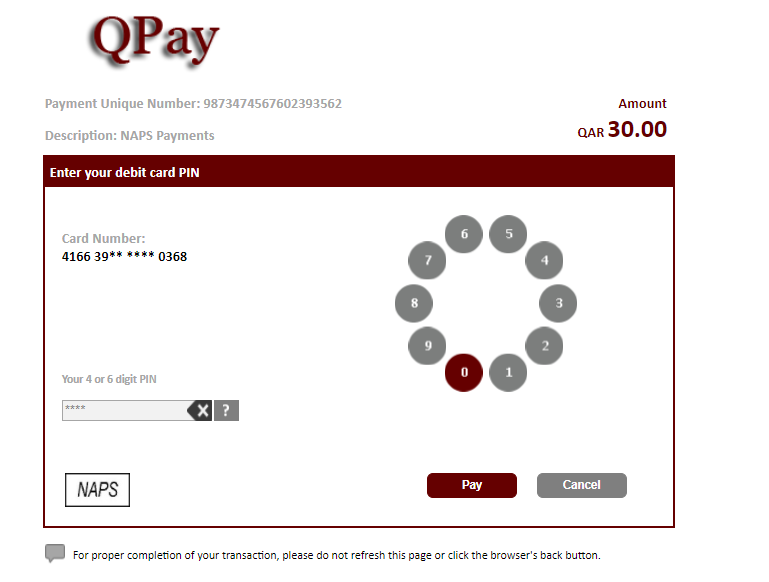
After clicking on make payment button, a loader would appear which would navigate the customer to external QPay interface.

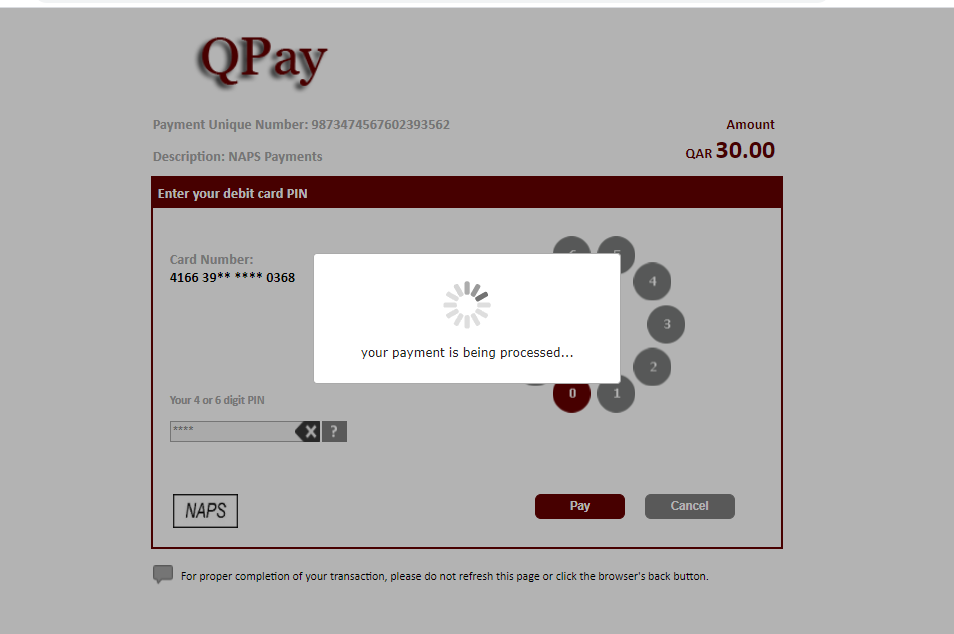


Once customer is navigated to Qpay external interface, Add card number and expiry and then proceed with clicking on Continue button.



Add PIN 1234 and click on Pay button.



Payment processing pop up would appear before customer and customer would then be redirected to pay2m payment gateway page with notification of transaction status.

