* API requirements for Integrating Payment
* **API Response**

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Type** | **Description** | **Sample** |
| Status | boolean | API response success or failure status | True/false |
| Message | string | Error message when API request failed |  |
| Data | Object | API return result.(Look in every api method below) | Look in every api method below. |

* **Environemnt**

|  |  |
| --- | --- |
| Environment | Url |
| Staging | <https://api-dev.gopayment.com.my/swagger/index.html> |

|  |  |
| --- | --- |
| Environment | Url |
| Production | To be confirm |

* API Request and Repsonse
* **Swagger**

Kindly refer to swagger below,

<https://api-dev.gopayment.com.my/swagger/index.html>

* Encryption method
* /api/Sales/create\_sale

Request

{MerchantId}{MerchantTransactionNo}{PaymentGateway}{Currency}{Amount.Value.ToString("0.##")}{ExpirySecond}{Today.ToString("yyyyMMdd")}

Response

{SalesTransactionNo}{MerchantTransactionNo}{Amount.ToString("0.##")}{Today.ToString("yyyyMMdd")}

* /api/Sales/query\_transaction

Request

{MerchantId}{SalesTransactionNo}{PaymentGateway}{Currency}{Amount.Value.ToString("0.##")}{Today.ToString("yyyyMMdd")}

Response

{MerchantId}{MerchantTransactionNo}{PaymentGateway}{Currency}{RefundAmount.Value.ToString("0.##")}{Today.ToString("yyyyMMdd")}

* /api/Sales/cancel\_transaction

Request

{MerchantId}{MerchantTransactionNo}{PaymentGateway}{Currency}{Amount.Value.ToString("0.##")}{Today.ToString("yyyyMMdd")}

Response

{Status}{Description}{MerchantTransactionNo}{Today.ToString("yyyyMMdd")}

* /api/Sales/refund\_transaction

Request

{MerchantId}{MerchantTransactionNo}{PaymentGateway}{Currency}{RefundAmount.Value.ToString("0.##")}{Today.ToString("yyyyMMdd")}

Response

{Status}{Description}{MerchantTransactionNo}{Today.ToString("yyyyMMdd")}

* Push Notification for Sales Result

Request

{MerchantTransactionNo}{SalesTransactionNo}{Description}{Today.ToString("yyyyMMdd")}

* Key Description
* Transaction Status

|  |  |
| --- | --- |
| Key | Description |
| new | New transaction request |
| payment\_gateway\_req\_fail | Fail to initialize with payment gateway |
| init | Initialized with payment gateway. |
| paying | In the process for user key in payment information |
| expired | Transaction had been expired |
| paid\_fail | User fail to pay |
| paid | User successfully paid |
| cancel\_by\_api | Transaction cancelled by api |
| refunded\_by\_failed\_sales | Transaction had been refunded due to transaction expired/cancelled |
| refunded\_by\_api | Refunded by api |
| refunded\_by\_merchant | Refunded from Merchant Portal |
| refunded\_by\_admin | Refunded from Admin Portal |
| manual\_refund | Manual Refund perform by admin |
| other | Other error |
| record\_not\_found | Record not found in System |
| cancel\_and\_refund\_by\_api | Transaction cancelled or Refund by API |

* TerminalType

|  |  |
| --- | --- |
| Key | Description |
| Web | Web Browser |
| MobileBrowser | Mobile In App browser |
| MobileApp | Mobile Application |
| Kiosk | Kiosk |

Base 64 string to image

WPF

<https://www.csharp-console-examples.com/general/c-base64-string-to-png-image/>

  string base64String = File.ReadAllText(@"C:\samples\base64Image.txt");

            byte[] imgBytes = Convert.FromBase64String(base64String);

            BitmapImage bitmapImage = new BitmapImage();

            MemoryStream ms = new MemoryStream(imgBytes);

            bitmapImage.BeginInit();

            bitmapImage.StreamSource = ms;

            bitmapImage.EndInit();

            //Image object

            image.Source = bitmapImage;

HTML

<!DOCTYPE html>

<html>

<head>

<title>Display Image</title>

</head>

<body>

<div>

<p>Testing QR</p>

<img src="data:image/png;base64, {replace base64 string here}" alt="Qr payment" />

</div>

</body>

</html>