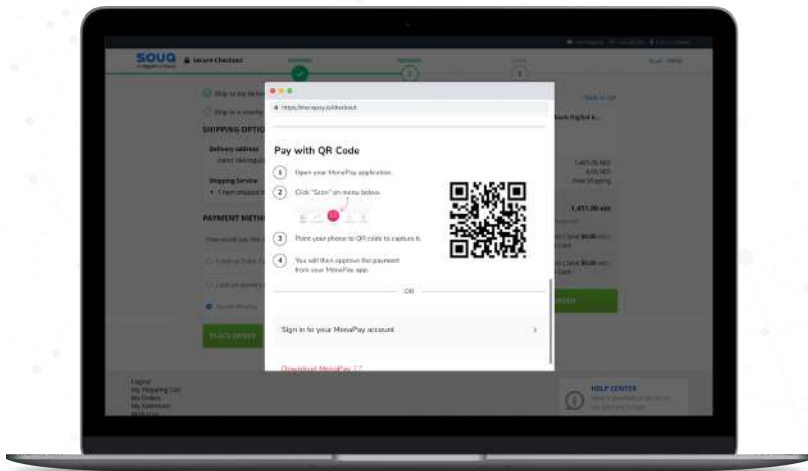


MenaPay E-commerce Site Payment API



MenaPay E-commerce Site Payment API is an online payment solution for those who want to offer MenaPay platform as a payment service. Using the power of blockchain MenaPay real-time international payment solution enables merchants to accept payment without borders. Merchants who are registered and approved by MenaPay Foundation is entitled to use MenaPay E-commerce Site Payment API. Merchants will be able to receive payments from MenaPay Wallet owners as MenaCash by integrating MenaPay API to their websites.

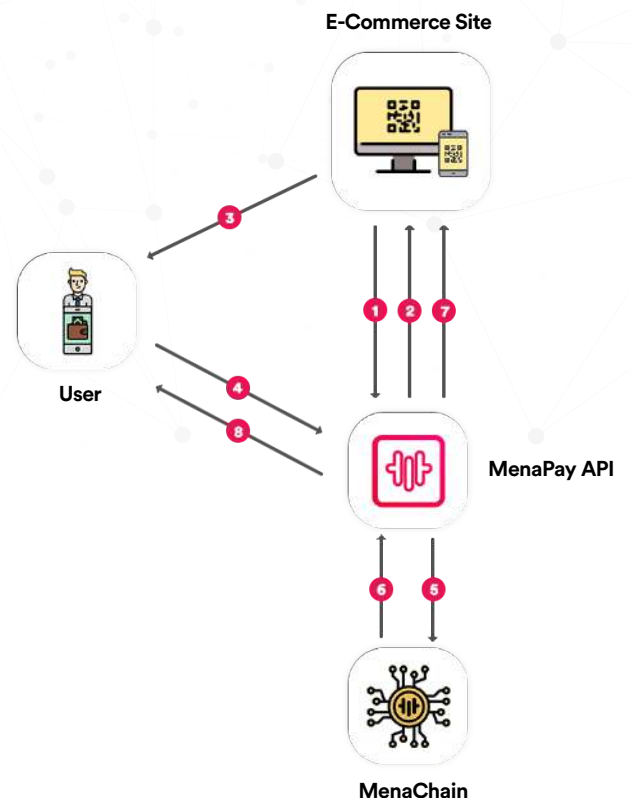


INTEGRATION

Menapay Ecommerce Site Payment API is easy to integrate. After the merchant completes the application process, the documentation will be accessible through MenaPay Merchant Dashboard for those who are willing to integrate MenaPay Ecommerce Payment API to their sites.

PRODUCT FLOW

- 1 Merchant Request QR code, and Provides a Unique Call-Back URL for this transaction
- 2 MenaPay API generates and Sends the QRCode for this transaction
- 3 Customer scans the QrCode
- 4 Customer Approves Transaction
- 5 MenaPay API sends signed transaction to the MenaChain
- 6 MenaChain performs the transaction and returns response
- 7 MenaPay API informs e-commerce site about the transaction result
- 8 MenaPay API informs the user about the transaction result





MenaPay API Document

Audience and Purpose

This document is written for application developers who want to integrate Menapay payment solution to their system. This integration requires additional software development to call required APIs to make a payment.

PAYMENT FLOW

If you integrate successfully with Menapay payment solution, an example payment will be performed as below:

- Buyer chooses “Pay with Menapay” option from several payment options.
- A QR code representing his payment is shown on merchant system.
- Buyer logs in to the Menapay Payment App on his/her mobile device.
- Buyer scans the QR code using Menapay Payment App.
- Menapay Payment App displays the payment details.
- Buyer approves the payment.
- Menapay informs both buyer and merchant system about the payment result.

Integration

Authentication

In order to use the services provided by Menapay, you need to authenticate first. This authentication is done using the credentials of your merchant account on Menapay Business App. An authentication token is returned as response to a successful authentication. This token will be used in further requests and API calls. Token has an expire time and should be refreshed to continue making new requests.



API List

1 Server URL

Production Environment: <https://api.menapay.io/>

2 API Request

POST /v1/auth/corporate/login

3 Request Parameters

○ Request Body:

```
{
  "username": "string",
  "password": "string",
  "userType": "PERSONAL"
}
```

○ Response Body:

```
{
  "data": {
    "accessToken": "string",
    "idToken": "string",
    "refreshToken": "string",
    "user": "object"
  },
  "message": "string",
  "code": "number",
  "error": "boolean"
}
```

Request Body:

username (required): Username of the merchant account on Menapay Business App. It can be of any account that is registered for that merchant.

password (required): Password of the merchant account on Menapay Business App. It can be of any account that is registered for that merchant.

userType (required): Can be one of the following values according to the account type on Menapay Business App.

MERCHANT (for merchant users)

ECS (for e-commerce site users)

Response Body:

accessToken: Access token for a user to be used in some APIs. (Not necessary in our context)

idToken: Id token for a user to be used in all APIs to provide authentication.

refreshToken: Refresh token for a user to be used for refreshing the id token. (Not necessary in our context)

user: User details of the token owner. (May be required to obtain public key of the account)

message: Success or error message if exists.

code: Error code if exists.

error: Boolean value representing if it is an error response or not.



Error Codes:

Please note that these error codes are returned in message body. HTTP status code is 400 for these cases. Message fields in responses may contain additional information about error.

828 - Incorrect username or password: Provided credentials are not valid.

803 - User not confirmed: User did not validate his/her email/phone that is used as username yet.

804 - Required/Invalid parameter: The request has empty/invalid argument.

837 - New password required: User should reset his/her password using Menapay Business App first.

500 - Unexpected error: There is an unexpected behavior, contact the system administrator.

Create QR Code for Payment API

1 Server URL

Production Environment: <https://api.menapay.io/>

2 API Request

POST /v1/transaction/ecs/qr

3 Preconditions

- ⦿ There should be an active merchant or e-commerce site Menapay Business Account.
- ⦿ Menapay Business Account should have an approved KYC process.
- ⦿ Token should be retrieved via Authentication API.

4 Request Parameters

- Request Header:

Authorization (required): Id token provided in the Authentication API response.

userType (required): Can be one of the following values according to the account type on Menapay Business App.

MERCHANT (for merchant users)

ECS (for e-commerce site users)



○ Request Body:

```
{
  "address": "string",
  "amount": "number",
  "callbackUrlError": "string",
  "callbackUrlSuccess": "string",
  "currency": "string",
  "isMenaCashToCash": "boolean"
}
```

○ Response Body:

```
{
  "data": "string",
  "message": "string",
  "code": "number",
  "error": "boolean"
}
```

Request Body:

address (required): Public key of the merchant account (Obtainable from the login response, inside user object)

amount (required): Cash or MenaCash amount of user basket.

isMenaCashToCash (required): true if MenaCash amount is provided above, false viceversa.

currency (required): Currency code for the basket, if cash amount is provided.

Supported currencies: DZD, BHD, EGP, IQD, JOD, KWD, LBP, LYD, MAD, OMR, SAR, SYP, TND, AED, YER, SDG, TRY, QAR, USD

callbackUrlSuccess (required): Callback URL for the basket to be called by MenaPay after this transaction is completed successfully. (Callbacks are performed using POST requests)

callbackUrlError (required): Callback URL for the basket to be called by MenaPay if this transaction is failed. (Callbacks are performed using POST requests)

Response Body:

data: Base64 encoded QR code data. It can be used by displaying as an image.

message: Success or error message if exists.

code: Error code if exists.

error: Boolean value representing if it is an error response or not.

Error Codes:

HTTP 401 Unauthorized: Token is missing or expired. Token should be refreshed by the Authentication API.

Please note that these error codes are returned in message body. HTTP status code is 400 for these cases. Message fields in responses may contain additional information about error.

804 - Required/Invalid parameter: The request has empty/invalid argument.

500 - Unexpected error: There is an unexpected behavior, you might need to contact the system administrator.



Receiving Payment Callbacks

There are 2 ways that can be used to receive payment callbacks. First one is the callback URLs that are provided while requesting the QR code. And the second way is getting notified via web sockets.

*** Note that even though there are two possible ways, the first one is mandatory. That means you should provide a success and an error callback URL even if you listen to the relevant web socket.

For the first case, Menapay Server performs a POST request to the corresponding callback URL according to the result of payment operation.

- Payment is successful → **POST request to callbackUrlSuccess**
- Error → callbackUrlError → **POST request to callbackUrlError**

For the second case, which is optional, Menapay Server sends the payment output thorough a web socket. (via Stomp over web sockets or Stomp over SockJS)

- Clients can connect to the socket using the following URL: **"https://api.menapay.io/websocket"**
- Clients then subscribes to the topic: **"/payment-complete/{publicKey}/{paymentId}"**

If the payment is successful, the following response is sent by Menapay Server:

HTTP 200 OK:

```
{
  "data": true,
  "message": null,
  "code": null,
  "error": false
}
```

If the payment is not successful, the following response is returned by Menapay Server:

HTTP 400 Bad Request:

```
{
  "data": false,
  "message": "This transaction
could not be performed.",
  "code": 839,
  "error": true
}
```