User

Clicks →

Partner Screen

Pay With FIB

Body Request Parameters:

- grant_type (client_credentials)
- client_id (Provided by Fib)
- client_secret (Provided by FIB)

First Get the Token ↓

Getting Authorized
POST

Response Parameters:

- access_token
- expires_in (expires in 60 sec)
- token_type (Bearer)

access_token ↑

FIB Payment Creating
POST

Body Request Parameters:

- Amount (No commas or points)
- Currency (IQD)
- StatusCallbackURL (Webhook)
- description (Payment description)
- expiresIn (Payment expiration)

Header Authorization:

- Bearer (access_token)

Response Parameters:

- PaymentID
- QR Code
- readableCode
- Redirection Link
- validUntil

Payment Screen

QR Code
or
Redirection Link

Back Button

FIB

Cancel (Back Button) →

Payment in progress (QR Scanning, Link) →

FIB Personal Application

Query Request Parameters:

- paymentId

Header Authorization:

- Bearer (access_token)

Cancel Payment (Declined)
POST

Declined After 15 Minutes

15 minutes Progress

Confirm within 15 minutes →

No →

Check balance availability

Yes ↓

Query Request Parameters:

- paymentId

Header Authorization:

- Bearer (access_token)

Confirmed Payment

Receive → Callback/Webhook to StatusCallBackURL

Updated: "Paid"
POST

After initiating a payment, wait for 10, 15 seconds for a callback update from FIB.

If Payment PAID

Updated: "Paid"

Refund Request
POST

If no callback is received withing 10s,15s, use the Check Payment Status Endpoint to query the status.

- paymentId

Header Authorization:

- Bearer (access_token)

Check Payment Status
GET

Response Parameters:

- pamentId
- status
- PaidAt
- amount
- PaidBy (name, Iban)

Response Parameters

- id (PaymentId of the payment)
- status (Status of the payment)

Repeat the callback and check payment status process until either:
- Receive a callback from FIB.
- The Payment gets paid
- The payment expires.

Please refer to page **(2)** for more clarification and notes ⬇

Notes:

Be careful whenever you're sending requests to each of these endpoints:

1. **Create Payment**
2. **Cancel Payment**
3. **Check Payment Status**
4. **Refund Payment**

- You will need to attach "access_token" as a Bearer token type in the Authorization Header of each of the endpoints requests mentioned above.

- And in order to get "access_token" you need to send a request to Authorization endpoint using your client_id & client_secret.

- once you got access_token it expires in 60 seconds which is a short time.
- what you will need to do is that to get a new token each time you're sending a request to the endpoints mentioned since it gets expired each 60 seconds.

- When you create a payment it lasts for 15 minutes the user can pay whether using QRCode or Redirection link or using the readableCode which is a manual code the user can copy and put it into:

  FIB App "QuickPay" $\longrightarrow$ ManualCode $\longrightarrow$ Paste the code.

- If the user did not pay the payment during this 15 minutes
- then the payment will automatically get declined.
- you can implement a cancel button to send a request to the "Cancel Payment" endpoint to decline the payment before the payment has been paid
- once the user paid the payment it cannot be declined however you as the business can implement the "Refund" endpoint to refund a payment after it is PAID.

Endpoints usage:

1. Authorization (POST Request): Used to get access_token which allows you to get access and send requests to other endpoints. it requires three body parameters: client_credentials, client_id, client_secret

2. Create Payment (POST Request): Used to create payments access_token is needed to access this endpoint it requires these body parameters: (amount, currency, statusCallbackUrl, description, expiresIn).

3. Cancel Payment (POST Request): Used to decline payments access_token is needed to access this endpoint it requires paymentId as a query parameter: (amount, currency, statusCallbackUrl, description, expiresIn).

4. Check Payment Status (GET Request): Used to to check the status of the payments by default the status of each payment is "UNPAID" access_token is needed to access this endpoint it requires paymentId as a query parameter: (amount, currency, statusCallbackUrl, description, expiresIn). in the response it will provide valuable information about the user who paid the payment such as (Name, IBAN) and more... it has these different statuses: (PAID, DECLINED, UNPAID, REFENDED, REFEND_REQUESTED)

5. Refund (POST Request): Used to refund a payment that is PAID. when Refunding the amount of money where the user paid before will get deducted from you the business who received the payment and it will be sent to the user who paid the payment. it takes a couple of minutes until it is processed and checked on our side then it receives to the user as "Bank Operation".

note: if you refund a payment which is not PAID then you will get an error.

access_token is needed to access this endpoint. it it requires paymentId as a query parameter.